Universidade do Minho
Escola de Engenharia

José António Silva de Carvalho Campos e Matos

Uncertainty Evaluation of Reinforced Concrete and Composite Structures Behavior

Avaliação de Incertezas no Comportamento de Estruturas de Betão Armado e Mistas

Abril de 2013

José António Campos e Matos Uncertainty Evaluation of Reinforced Concrete and Composite Structures Behavior

UMinho | 2013

Universidade do Minho
Escola de Engenharia

José António Silva de Carvalho Campos e Matos

Uncertainty Evaluation of Reinforced Concrete and Composite Structures Behavior

Avaliação de Incertezas no Comportamento de Estruturas de Betão Armado e Mistas

Tese de Doutoramento
Engenharia Civil

Trabalho efectuado sob a orientação de
Professor Doutor Paulo Jorge de Sousa Cruz
Professora Doutora Maria Isabel Brito Valente

Abril de 2013

To all of you that make this work possible …

Acknowledgments

The presented work was possible due to the contribution of several people. Therefore, I express my gratitude to all of you that, in some way, helped me, especially:

- To my supervisors, Professor Paulo Cruz and Isabel Valente, who coordinated the research work, giving all the support with their guidance, scientific knowledge and trust;

- To the technicians of the Laboratory of Civil Engineering of Minho University, for their help in the development of the laboratory tests;

- To Professor Luis Neves, for the help on the development of the probabilistic analysis module;

- To all my colleagues of the Structural Group of Civil Engineer Department, for their helpful suggestions;

- To all my friends, for being with me during this work.

I would also like to address my thankfulness to:

- Minho University, for providing the conditions and facilities to develop this work;

- FCT – Science and Technology Portuguese Foundation for the support awarded through the grant SFRH/BD/10705/2002, co-funded by the Operational Programme for Science and Innovation 2010 (POCI 2010) and by the European Social Fund (ESF);

- Cervenka Consulting, for their support in numerical model development;

- BRISA, for providing the design project of Sousa River Bridge;

- Teixeira Duarte, for providing data values obtained from in-field tests of Sousa River Bridge.

And, finally,

- To my family, particularly to my grandmother Bé and to my sisters Ana Raquel and Joana, for giving me all the support;

- To my mum and dad whose care and support are always there;

- To Ana who was always there with such care and attention;

- To all of you who are not physically present at this moment but who were very important for me, namely, to my grandfather's Zé and Armando, to my grandmother Dicha, to my cousin Ana, to my aunt Cristina and to Noca.

Abstract

The quantity of civil engineering infrastructures increased in a substantial manner within the last few decades. As a result of this, countries become responsible for an enormous set of infrastructures to be maintained during its whole life. However, and according to OECD (Organization for Economic Co-operation and Development), the dispended budget in this field is still lower than the recommended one.

Accordingly, it becomes necessary to develop advanced tools that, with the support of data from diverse observation systems, would permit to obtain a real image of the analysed structure. This way, some researchers produced, in the last few years, different structural assessment frameworks. While some introduced the model identification techniques, others presented the probabilistic assessment algorithms with Bayesian inference updating.

Within this thesis an advanced probabilistic assessment algorithm, that comprises both these techniques of model identification and Bayesian inference, is developed. A two-step updating procedure is then established. The developed algorithm is combined with commercial nonlinear structural analysis software (ATENA®). This will allow the structure evaluation both in service and failure region.

Additionally, it is intended to incorporate within this algorithm all different uncertainty sources that coexist in a structural assessment procedure. Some of these sources are introduced during the identification process, while others are incorporated during the probabilistic analysis. The final purpose is to obtain a reliability index, based in the comparison between loading and resistance curves, which provides a real measure of structural safety.

The developed algorithm is, posteriorly, validated with different sets of laboratory tested structures which were loaded up to failure. These sets include both reinforced concrete and composite beams. Furthermore it is tested on a composite bridge submitted to a load test (Sousa River Bridge). Obtained results permit, in some situations, to identify an additional structural capacity which was not known at beginning.

Resumo

O número de infraestruturas cresceu substancialmente nas últimas décadas, fruto de um crescimento económico desmedido. Como resultado deste fenómeno, os países ficaram detentores de um vasto parque de infraestruturas, a manter durante o seu ciclo de vida. Contudo, e de acordo com a OCDE (Organização para a Cooperação e Desenvolvimento Económico), o valor despendido nesta matéria encontra-se, ainda, muito aquém do recomendado.

Desta forma foi necessário desenvolver ferramentas que, com o apoio dos dados provenientes dos diversos sistemas de observação, permitissem a obtenção de uma imagem real da estrutura em análise. Neste sentido, vários investigadores desenvolveram, nos últimos anos, diversos algoritmos de avaliação estrutural. Alguns incidiram sobre técnicas de identificação, enquanto outros focaram as análises probabilísticas com atualização inferencial.

Nesta tese, desenvolveu-se uma ferramenta de avaliação probabilística da segurança das estruturas que combina ambas as técnicas de identificação e inferência Bayesiana. Esta ferramenta é, posteriormente, implementada num programa comercial de análise não linear de estruturas (ATENA®), possibilitando, deste modo, a sua avaliação quer em serviço quer em rotura.

Em paralelo, pretende-se, com este algoritmo, a incorporação das diferentes fontes de incerteza existentes num processo de avaliação estrutural. Enquanto algumas são introduzidas durante o processo de identificação estrutural, outras são incorporadas durante a própria análise probabilística. O objetivo final será a obtenção de um índice de fiabilidade, baseado numa comparação entre curvas de resistência e de carga, que forneça uma informação fidedigna da segurança estrutural.

O algoritmo é, posteriormente, validado através de uma aplicação a um conjunto de estruturas ensaiadas até à rotura em laboratório. Este conjunto incorpora vigas em betão armado e mistas aço-betão. Seguidamente, será testado com uma ponte mista aço-betão submetida a um ensaio de carga (Ponte sobre Rio Sousa). Os resultados obtidos permitem, em algumas situações, a identificação de uma capacidade estrutural adicional não prevista em projeto.

Keywords / Palavras-chave

Keywords	Palavras-chave
Optimization algorithm	Algoritmo de otimização
Bayesian inference	Inferência Bayesiana
Uncertainty sources	Fontes de incerteza
Model identification	Identificação numérica
Probabilistic assessment	Avaliação probabilística

List of contents

Chapter 3. Bayesian Inference

Chapter 4. Probabilistic Assessment of Structures

Chapter 5. Reinforced Concrete Beams

Chapter 7. Sousa River Bridge

Chapter 8. Conclusions

References

Appendices

List of figures

List of tables

1 Introduction

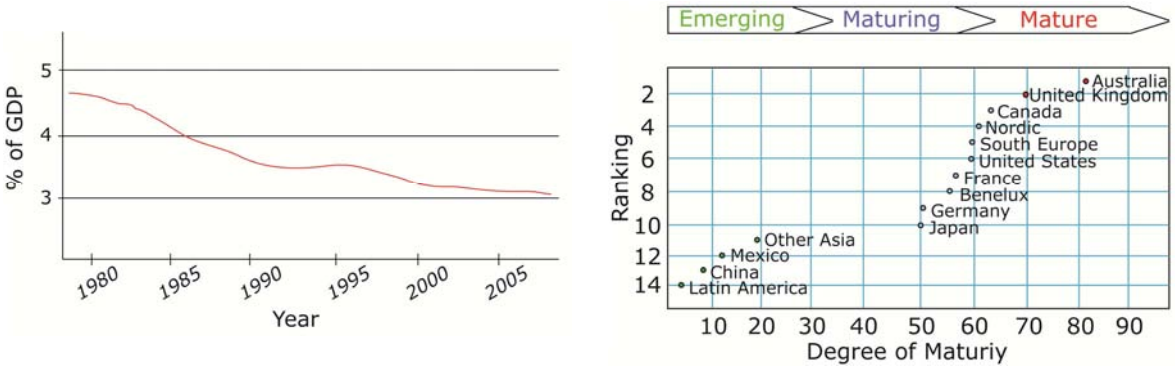
1.1. Introduction and motivation

The investment in infrastructure assets increased in the second half of the 20th century due to an excessive economic growth. As a result, developed countries become responsible for the maintenance and replacement, due to service and safety issues, of a set of infrastructures, namely for transport, health, energy and water resources, which are now ageing [141]. However, maintaining current infrastructure quality is not expected to be economically sustainable.

According to the Organization for Economic Co-operation and Development (OECD) the need for maintenance and replacement of public infrastructure represents, annually, 3.5% of each country gross domestic product (GDP). Figure 1.1a indicates the average GDP percentage of OECD countries dedicated to infrastructure since the year 1980 [142]. This reflects a general global trend of rarefaction of resources dedicated to infrastructure.

Figure 1.1b reports the result of a recent study on the maturity of the market in infrastructure in several countries [140]. The respective rating is based in the country risk (including legal and regulatory risk along with political economic and financial risk) together with the value of completed deals in the last 24 months as a percentage of GDP (reflecting a country's experience with private involvement in infrastructure projects). It shows that most western countries have infrastructure in advanced stage of maturity.

The OECD noted that by 2030 "... a larger effort will need to be directed towards maintenance and upgrading of existing infrastructure and to getting infrastructure to work more efficiently" [32]. Therefore, it is extremely important for countries to prioritize their budget expenditures and investments in this topic by improving the way structures are currently being evaluated. Accordingly, several researchers developed structural assessment frameworks that will provide an accurate image of analyzed structure [6, 41, 52, 165].



a) b)

Figure 1.1. OECD report results: a) average infrastructure investments in OECD countries (adapted from [142]); b) world infrastructure market maturity (adapted from [140]).

Some of these researchers introduced model identification techniques [6, 70, 152, 165], based in both static and dynamic data. The objective of these frameworks is to obtain an accurate representation of evaluated structure. Several optimization algorithms are thus implemented in order to obtain this model. The effectiveness of model identification will depend of considered optimization algorithm.

Recently, some researchers have developed identification algorithms that consider both modeling and measured data errors [6, 70, 152]. However, different ways of introducing such errors exist. Some authors incorporate them as a bias in model parameter values [6]. Others introduced them in an optimization algorithm criterion [70, 152].

The use of full probabilistic structural assessment algorithms is also recent [14, 41, 43, 52]. Although promising, as they permit to incorporate randomness into structural models, these algorithms were constantly avoided due to their high computational cost. This is due to the fact of being supported in sampling techniques, which obliges to compute, for several times, a specific numerical model. Fortunately, the evolution of computers in the last decades overcomes this important obstacle.

The use of permanent monitoring systems in critical infrastructures increased in the last few years [14, 43]. Moreover, on non-critical structures, it is recommended, due to durability

problems, to carry out scheduled nondestructive evaluations. Obtained data is so important to describe the real structural behavior. A Bayesian inference algorithm [15] may be used to update probabilistic numerical models through the use of such data [41, 52].

All these structural assessment algorithms will allow combining, at different stages, different uncertainty sources. However, there still exist many doubts regarding the correct moment an identification algorithm should be applied or a probabilistic model should be used, what errors should be considered within an analysis, etc. The algorithm which is developed within this thesis, described in Figure 1.2, combines different uncertainty sources. In order to validate it, the algorithm is tested on two laboratory tested beams and on a real bridge.

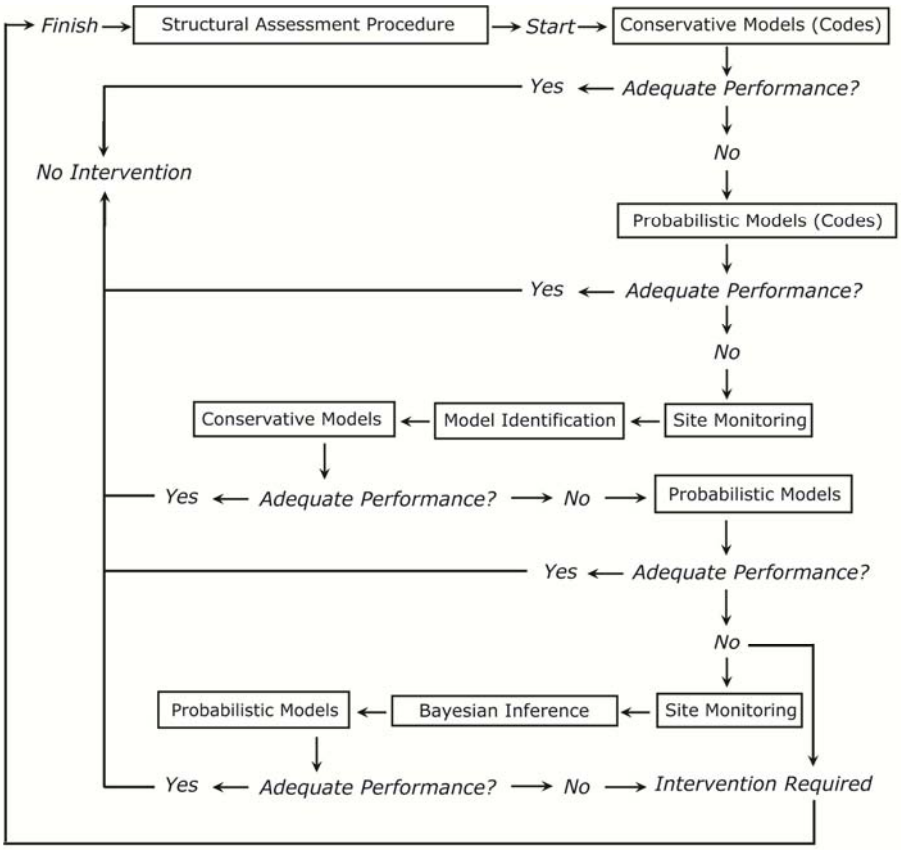


Figure 1.2. Organization chart of developed algorithm.

Traditionally, structural assessment is developed according to existent codes in which conservative or even probabilistic models are provided. However, there are some situations in which the studied structure presents a higher capacity reserve. This fact becomes more evident in existent structures. Accordingly, in those situations, site monitoring becomes important in order to obtain real data from the evaluated structure. This data is then used to predict the structural parameters, through the application of a model identification algorithm. The same models may be used to evaluate the structural performance but, now, considering the assessed parameters.

However, in some cases, the structural performance is still considered to be inadequate. In those situations, full probabilistic assessment algorithms are recommended as they allow to introduce randomness in model parameters and to evaluate the structural performance from a probabilistic point of view. Sometimes, additional tests are important to characterize, in a more realistic way, some parameters. In such situations it is recommended to automatically update the developed model through the use of a Bayesian inference procedure.

The structural performance evaluation may be so developed in an automatic basis. Therefore, within this thesis, a two-step updating algorithm is developed, which can be used with success to evaluate the structural performance, see Figure 1.2. The developed algorithm is then validated with reinforcement and composite steel-concrete structures.

This thesis focuses on the development of an advanced probabilistic assessment algorithm. The main result of this application, both in service and failure region, is an updated resistance curve for the assessed structure. The decision regarding the most adequate intervention and its cost is a topic that will be held for future developments.

1.2. Objectives

The main objective of this thesis is to present an advanced probabilistic structural assessment framework, which considers both model identification and Bayesian inference algorithms. Such framework will be applied with existent reinforced concrete and composite structures. Accordingly, it is first necessary to study existing procedures for structural assessment and find out their advantages and drawbacks.

The proposed methodology aims to incorporate all uncertainty sources within the analysis. This is important as most of the algorithms do not take them into account. Therefore, a detailed description of those uncertainty sources and of how they can be considered within the analysis becomes necessary. These sources include both modeling and measurement errors as also randomness in model parameters.

It is also planned to implement the developed algorithm with a nonlinear structural analysis software. This will allow the evaluation of the structural behavior both under service and failure loads. Obtained results from the analysis under service loads, which is typical from structure load tests, are further extrapolated for an analysis up to the structure collapse. Finally, it becomes necessary to validate the proposed probabilistic assessment algorithm. This will be done with both laboratory and real structures.

Thus, it is possible to identify the following objectives to attain within this thesis: (1) develop a probabilistic based algorithm for structural assessment, which can be automatic updated with

obtained data; (2) implement the developed structural assessment algorithm with a nonlinear structural analysis framework, which will allow more accurate predictions of structural behavior under both service and failure loads; (3) incorporate different uncertainty sources within the structural assessment algorithm, giving a special attention to measurement and modeling errors; (4) validate the developed structural assessment algorithm with both reinforced concrete and composite beams loaded up to failure in laboratory controlled conditions; (5) apply the developed structural assessment algorithm with a real case of a composite bridge, submitted to a load test; (6) help operators to better characterize the real state of their infrastructure and, especially, on the decision regarding the more appropriate maintenance strategy; (7) contribute to the success of investment of countries in maintenance and replacement of their infrastructure.

1.3. Outline of thesis

This thesis is composed by eight chapters which are complementary. The organization chart of this thesis is schematized at Figure 1.3.

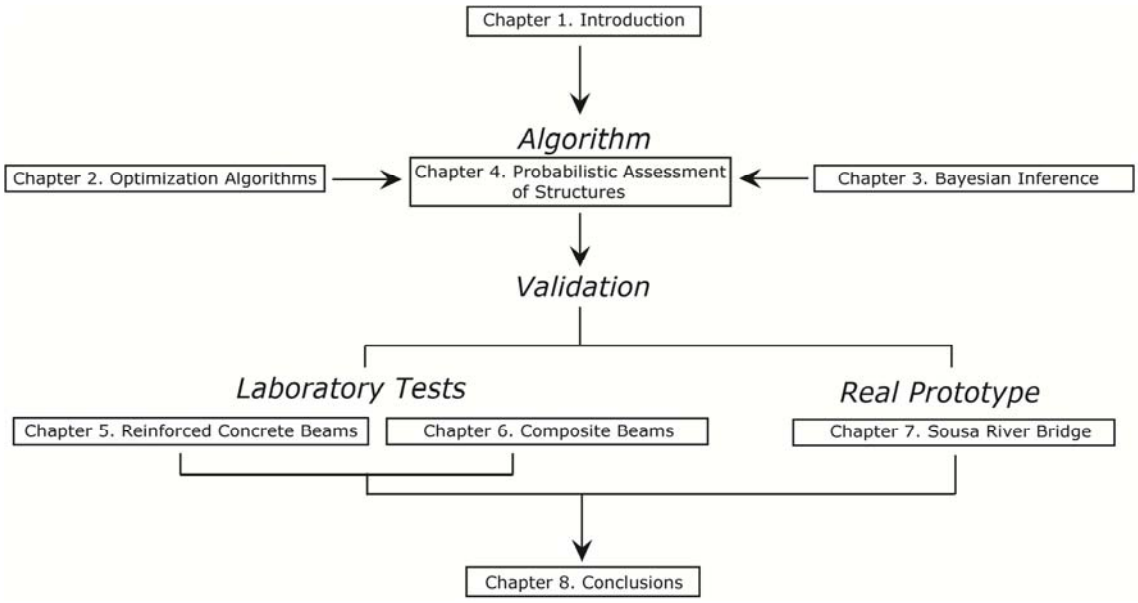


Figure 1.3. Outline of developed thesis.

The first chapter presents the scope and main motivations of this thesis. It specifically focuses the OECD needs in maintenance and replacement infrastructure investments and contrasts it with the financial support that countries actually provide. As a result, it indicates the needs of more efficient assessment algorithms, in order to obtain a reliable image of the studied infrastructure. It also presents the main objectives of this thesis.

In the second chapter, various optimization algorithms that might be used with an identification methodology are presented. Both local and global algorithms, as well as, the

advantages and drawbacks of each, are presented. These algorithms are firstly tested with traditional optimization problems. Secondly, they are tested with a problem in which the numerical model parameters of a reinforced concrete beam are adjusted in order that obtained results best fit experimental data. Obtained results are then compared in terms of computational cost and improvement in fitness function.

Chapter three describes the Bayesian inference method. This procedure is important to update probabilistic models with acquired data. Its mathematical formulation as also some simple and useful applications, are presented. These examples cover the inference of Normal, Lognormal and Weibull probability density functions (PDF) which include most significant input parameters. This method is relevant as it allows to control and eventually to reduce the statistical uncertainty.

Chapter four presents the developed algorithm for probabilistic assessment of structures. It begins with a description of structural assessment algorithms indicated in the literature and of most important uncertainty sources. Then, a description of the developed model identification algorithm is presented. Special attention is given to modeling and measurement errors. The probabilistic framework, which includes a sampling algorithm, is then introduced. Finally, a review of how reliability indexes are computed and of proposed target indexes, by different authors, is provided.

In chapter five the proposed algorithm for probabilistic structural assessment is validated and tested with a set of reinforced concrete beams that were loaded up to failure in laboratory. A description of developed tests, numerical model and obtained results is provided.

In chapter six, the algorithm is tested with a set of composite beams. These structures present a higher degree of uncertainty due to the difficulty in characterizing the steel-concrete connection. The methodology is then validated, being presented the developed tests, the numerical model and obtained results.

A final application of this methodology with a case study of a composite bridge is provided in chapter seven. The load test, the developed numerical model and obtained results from assessment algorithm are discussed in detail. This will allow to validate the algorithm and also to confirm the real condition of the evaluated structure.

A summary of obtained results from the application of developed probabilistic structural assessment algorithm with reinforced concrete and composite structures, both in laboratory and in-situ, is presented at chapter eight. It is also indicated the future developments of this research.

2 Optimization Algorithms

2.1. Introduction

In civil engineer field, when the need to interpret the behavior of any particular structure arises, numerical models are commonly used. In an initial phase, such models are only a simple representation of the real structure. The experimental data, despite the related errors, is at such phase, more accurate. Then, these models may be updated within a procedure in which the numerical results are adjusted to existent experimental data. Such procedure is usually designated by model identification [6, 7].

This process is generally based in a mathematical optimization algorithm. The selection of such algorithm is recognized as a very important step. It is verified that the majority of structural numerical models are multi-parametric, as they depend on more than one parameter, and nonlinear, due to the structural nonlinear behavior. Due to these factors, the objective function might present several minima, becoming the process of finding the global optimum more complex.

The optimization algorithms can be divided in local and global procedures (Figure 2.1) [184]. The former do not always find the global minimum of the objective function, that is, the minimum of all local minima of the function. This minimum is very useful to identify in many practical applications but it is usually a hard task to perform it. Also, it is very difficult to know beforehand if the objective function contains besides global minima, also local minima, since this requires the knowledge of the overall shape of the function.

On the other hand, the global minima, as several local minima, may not be from a physical point of view the value we are seeking for. In fact, the determined global minimum is only a pure mathematic value. An engineering judgment is so necessary to evaluate if obtained minima presents any logic from an engineer point of view. Sometimes the value we are searching for may be a local and not a global minimum. Consequently, the use of any optimization search procedure does not dispense an engineer evaluation of obtained results.

Local algorithms start from one initial point, generating in an iterative way a sequence of improved estimates until the solution is reached. These procedures are generally based in the gradient of the objective function. Although very popular, as they are very fast, these methods do not guarantee to find the global minimum since they can be easily trapped into a local solution. Global algorithms are more robust, as the choice of the starting point has little influence on the final result, and are more likely to detect a global minimum. The main drawback of such algorithms is the fact of requiring a large number of function evaluations, since they are based on probabilistic searching without the use of any gradient information.

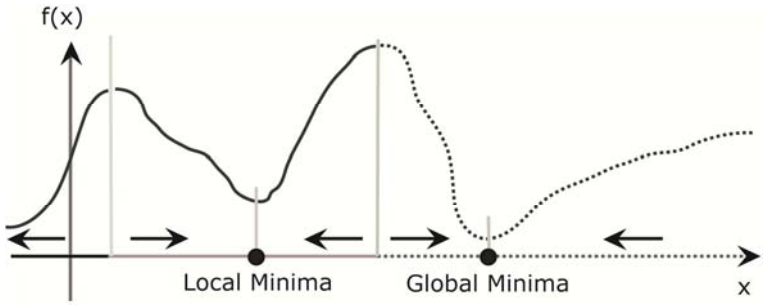


Figure 2.1. Local and global minimum.

The first model identification procedures were developed by using local search optimization algorithms [107], as the sequential quadratic programming (SQP) [56, 65, 132, 150]. These techniques were applied with static [11, 12, 78, 161, 199] or dynamic [63, 88, 89, 112, 162] measurements for model parameter estimation and, sometimes, in damage detection. Some authors developed an improved formulation based on both measurement types [74, 143].

Global search algorithms were later used in model identification. Different techniques were developed within the last decades. From those techniques it is important to mention the ant system (ASO), introduced in 1991 by Dorigo [34, 35, 36], and the particle swarm (PSO), presented in 1995 by Kennedy and Eberhart [39, 95]. These are population based algorithms that mimic the social behavior of animals in a flock. Raphael and Smith [151] Robert-Nicoud *et al.* [156, 157, 158] and Smith and Saitta [169] present a multiple-model identification algorithm, designated as probabilistic global search Lausanne (PGSL). An improvement of

PGSL, designated as candidate model search for system identification (CMS4SI), was developed by Goulet *et al.* [70, 72] and Goulet and Smith [71].

Others significant global search algorithms are the simulated annealing (SA), the genetic algorithms (GA) and the evolutionary strategies (ES). The first was conceived by Kirkpatrick *et al.* [96, 191] and is based in an analogy between the annealing of solids and the problem of solving optimization problems. The second one was first introduced by Holland [79] and is defined by an algorithm that acts in a population of search points and generates new populations using randomized operators that mimic those of natural evolution as the selection, crossover and mutation. Several applications of GA in model identification can be found in existent bibliography [27, 81, 82, 83, 168, 198]. The third algorithm, presented first by Rechenberg [153] and later by Schwefel [167], is also a search procedure that mimics the evolution of species in natural systems. Franco [57, 58] presents an application of ES for model identification. Both GA and ES are considered as evolutionary algorithms (EA) as they are based on Darwin's theory of evolution. However, ES works directly with real representations of decision variables being the defined transition rules, in particular the ones related with selection, deterministic [9, 16].

The coupled local minimizers (CLM), is a hybrid method proposed by Suykens *et al*. [178, 179]. In this method, a cooperative search mechanism is set up by performing a number of local optimization runs simultaneously coupled by information exchange. The global search process is directed by the gradients in each search point. In this way, CLM combines the advantage of local with global algorithms. Teughels [184] uses such algorithm for dynamic model identification. Other hybrid algorithms were developed by different authors. Koh *et al.* [98] used a GA framework coupled to a suitable local search method for identification of structural parameters of large structural systems. Lagaros *et al.* [100] investigate the efficiency of various hybrids EA as GA-SQP and ES-SQP when applied to large-scale structural problems.

This chapter describes different local and global optimization algorithms with the main objective of identifying the most suitable for civil engineering model identification. In these procedures, the used objective functions are nonlinear and dependent of several variables. In a first step, optimization algorithms are applied to analytical optimization problems, being further used in a real application, a reinforced concrete beam which was submitted to a laboratory test up to failure. The computational cost, the number of function evaluations and the obtained fitness value are compared.

2.2. Optimization algorithms

Optimization algorithms are used to find a set of optimal parameters *x*. In simple cases, this might be the minimization or maximization of some system that is dependent on *x*. In a more advanced formulation, the objective function *f(x)* to be minimized or maximized might be subjected to constraints. A general optimization problem may be stated as, (2.1),

$$min_x\ f(x) \tag{2.1}$$

subjected to (2.2),

$$g_i(x) = 0,\ i = 1,\ldots,m_e \tag{2.2}$$

and to (2.3),

$$g_i(x) \leq 0,\ i = m_e+1,\ldots,m \tag{2.3}$$

where *x* is the design parameters vector, *f(x)* is the objective function which returns a scalar value, and $g_i(x)$ a vector function containing the values of equality and inequality constraints. An efficient and accurate solution to this problem depends, not only on the size of the problem, in terms of the number of constraints and design variables, but also on characteristics of the objective function and constraints.

2.3. Local optimization methods

2.3.1. Sequential quadratic programming

The sequential quadratic programming (SQP) is a local direct search method where constraints are handled explicitly during the whole procedure. Within this method, the solution is found by solving a sequence of quadratic programming (QP) problems [184]. SQP can be considered as a generalization of Newton method for unconstrained optimization [18, 75, 146, 147] as it finds a step away from the current point by minimizing a quadratic model of the problem. Given the general problem described in expressions (2.1) to (2.3), the main concept of SQP method is the formulation of a QP problem based on a quadratic approximation of the following Lagrangian function (2.4),

$$L(x, \lambda) = f(x) + \sum_{i=1}^{m} \lambda_i \cdot g_i\,(x) \tag{2.4}$$

being λ_i the Lagrange multiplier. The QP problem is thus obtained by linearizing the nonlinear constraints, (2.5),

$$\min_{d_k \in R^n} \frac{1}{2} {d_k}^T H_k d_k + \nabla f(x_k)^T d_k \tag{2.5}$$

subjected to (2.6),

$$\nabla g_i(x_k)^T d_k + g_i(x_k) = 0, i = 1, \dots, m_e \tag{2.6}$$

and to (2.7),

$$\nabla g_i(x_k)^T d_k + g_i(x_k) \leq 0, i = m_e + 1, \dots, m \tag{2.7}$$

being d_k the search direction vector and H_k the Hessian matrix of the Lagrangian function. This problem can be solved by using a QP algorithm. Therefore, the SQP implementation consists of three main stages: (1) updating the Hessian matrix; (2) quadratic programming solution; (3) line search and objective function.

2.3.1.1. Updating the Hessian matrix

A positive definite quasi-Newton approximation of the Hessian of the Lagrangian function (2.4) is computed each iteration (2.8),

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T s_k^T s_k H_k}{s_k^T H_k s_k} \tag{2.8}$$

where $s_k = x_{k+1} - x_k$, and q_k is obtained by expression (2.9).

$$q_k = \left(\nabla f(x_{k+1}) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x_{k+1}) \right) - \left(\nabla f(x_k) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x_k) \right) \tag{2.9}$$

Powell [146, 147] recommends keeping the Hessian positive definite within the whole procedure, although it might be positive indefinite at solution points. A positive definite Hessian is maintained providing that ${q_k}^T s_k$ is positive at each update and that H_k is initialized with a positive definite matrix. When ${q_k}^T s_k$ is not positive, q_k is modified on an element-by-element basis so that ${q_k}^T s_k > 0$. The aim of this modification is to distort the elements of q_k, which contribute to a positive definite update, as little as possible. Therefore, in an initial phase of the modification, the most negative element of ${q_k}^T s_k$ is repeatedly halved. This procedure is continued until ${q_k}^T s_k$ is greater than or equal to a small negative tolerance. If, after this procedure, ${q_k}^T s_k$ is still not positive, q_k is modified by adding a vector v multiplied by a constant scalar w, that is, $q_k = q_k + w\,v$. Vector v is computed through expression (2.10).

$$v_i = \nabla g_i(x_{k+1}) g_i(x_{k+1}) - \nabla g_i(x_k) g_i(x_k) \tag{2.10}$$

if $(q_k)_i\, w < 0$ and $(q_k)_i\, (s_k)_i < 0$, for $i = 1, \dots, m$, otherwise $v_i = 0$. This value is increased systematically until ${q_k}^T s_k$ becomes positive.

2.3.1.2. Quadratic programming solution

A QP problem of the following form, obtained from expressions (2.5) to (2.7), is solved each iteration, (2.11),

$$\min_{d_k \in R^n} q(d_k) = \frac{1}{2} {d_k}^T H d_k + c^T d_k \tag{2.11}$$

subjected to (2.12),

$$A_i d_k = b_i,\ i = 1, \ldots, m_e \tag{2.12}$$

and (2.13),

$$A_i d_k \leq b_i,\ i = m_e+1, \ldots, m \tag{2.13}$$

where A_i refers to the *i*-th row of the *m*-by-*n* matrix *A*. An active strategy method may be used to solve this problem [65]. This procedure involves two phases. The first one comprises the computation of a feasible point, while the second corresponds to the generation of an iterative sequence of feasible points that converges to the solution. In this method an active set $\bar{A}_k$, that is an estimate of the active constraints at solution point, is maintained. The active set is updated each iteration *k* and this is used to form a basis for the search direction d_k. The search direction d_k is computed and minimizes the objective function, while remaining on any active constraint boundaries.

The feasible subspace for d_k is formed from a basis Z_k whose columns are orthogonal to the estimate of the active set $\bar{A}_k$. Thus a search direction, which is formed from a linear summation of any combination of the columns of Z_k, is guaranteed to remain on the boundaries of the active constraints. The matrix Z_k is formed from the last $m - l$ columns of the decomposition of the matrix ${\bar{A}_k}^T$, where *l* is the number of active constraints or those which are on the constraint boundaries. Once Z_k is found, a new search direction d_k is sought that minimizes *q(d)*, where d_k is in the null space of the active constraints. Therefore d_k is a linear combination of the columns of Z_k: $d_k = Z_k\, p$, for some vector p. Then by viewing expression (2.11) as a function of *p*, it results in expression (2.14).

$$q(p) = \frac{1}{2} p^T Z_k^T H Z_k p + c^T Z_k p \tag{2.14}$$

Differentiating this with respect to *p* yields expression (2.15).

$$\nabla q(p) = Z_k^T H Z_k p + Z_k^T c \tag{2.15}$$

In which this differentiation is referred to as the projected gradient of the quadratic function because it is the gradient projected in the subspace defined by Z_k. Assuming the Hessian matrix to be positive definite, then the minimum of the function *q(p)*, in the subspace defined

by Z_k, occurs when the projected gradient is null, which corresponds to the solution of the following system of linear equations (2.16).

$$Z_k^T H Z_k p = -Z_k^T c \tag{2.16}$$

A step is then taken of the form $x_{k+1} = x_k + \alpha d_k$. At each iteration, due to the quadratic nature of the objective function, there are only two choices of step length (α).A step of unity along d_k is the exact step to the minimum of the function, restricted to the null space of $\bar{A}_k$. If such a step can be taken, without violation of the constraints, then this is the problem solution. Otherwise, the step along d_k to the nearest constraint is less than unity and a new constraint is included in the active set at the next iteration. The distance to the constraint boundaries in any direction d_k is given by (2.17),

$$\alpha = \min_{i \in \{1,\ldots,m\}} \left\{ \frac{-(A_i x_k - b_i)}{A_i d_k} \right\} \tag{2.17}$$

which is defined for constraints not in the active set. When n independent constraints are included in the active set, without location of the minimum, the Lagrange multipliers (λ_k) are computed in order to satisfy the nonsingular set of linear equations (2.18).

$$\overline{A_k^T} \lambda_{\mathrm{k}} = \mathrm{c} \tag{2.18}$$

If all elements of λ_k are positive, than x_k is the optimal solution of the QP problem. However, if any component of λ_k is negative, and the component does not correspond to an equality constraint, then the corresponding element is deleted from the active set and a new iterate is sought.

2.3.1.3. Line search and objective function

The solution to the QP problem produces a vector d_k which is used to form a new iterate, (2.19).

$$x_{k+1} = x_k + \alpha d_k \tag{2.19}$$

The step length parameter α_k is determined in order to produce a sufficient decrease in the objective function. The objective function used by Han [75] and Powell [146, 147] presents the following form, (2.20),

$$\varphi(x) = f(x) + \sum_{i=1}^{m_e} r_i g_i(x) + \sum_{i=m_e+1}^{m} r_i max[0, g_i(x)] \tag{2.20}$$

being r_i the penalty parameter obtained by expression (2.21).

$$r_i = (n_{k+1})_i = max_i \left\{ \lambda_i, \frac{(n_k)_i + \lambda_i}{2} \right\}, i = 1, \dots, m \quad (2.21)$$

This allows positive contribution from constraints that are inactive in the QP solution but were recently active.

Although similar to other traditional active-set algorithms [184], SQP presents some differences as: (1) strict feasibility with respect to bounds. The SQP algorithm takes every step in the region constrained by bounds; (2) robustness. During iterations the SQP algorithm can attempt to take a step that fails. In this situation the algorithm attempts to take a smaller step; (3) refactored linear algebra routines. The SQP algorithm uses a different set of linear algebra routines to solve QP problems. These routines are more efficient in both memory usage and speed than traditional active-set routines; (4) reformulated feasibility routines. SQP algorithm has two new approaches to the solution when constraints are not satisfied: (a) the SQP algorithm combines the objective and constraint functions into an objective function. This modified problem can lead to a feasible solution. This approach has more variables than the original problem and this can slow the solution of the QP problem [172, 186]; (b) the SQP considers an attempt step that causes the constraint violation to grow. The SQP algorithm attempts to obtain feasibility using a second-order approximation to the constraints. This technique can slow the solution by requiring more evaluations of the nonlinear constraint functions.

2.4. Global optimization methods

2.4.1. Simulated annealing

Simulated annealing (SA) is a global search method, based in the annealing process of heating up a solid and then cooling it down slowly until it crystallizes. The atoms in material have high energies at high temperature and have more freedom to arrange them. As the temperature is reduced, the atomic energy decreases. A crystal with regular structure is obtained at the state where the system has minimum energy. If the cooling is carried out very quickly, known as rapid quenching, widespread irregularities and defects can be found in the crystal structure. The system does not reach the minimum energy state and ends in a polycrystalline state which has a higher energy.

The analogy between the annealing process and optimization is as follows. The state of physical substance corresponds to the value of the design vector in optimization, the physical energy is represented by the objective function, the temperature is introduced as a control parameter and finding the lowest energy state corresponds to find the global minimum. The

same way atoms find their way to build a perfect crystal structure through random movements, global minimum is reached through a search within randomly generated candidates for the design vector.

The algorithm starts from an initial point and generates in an iterative process consecutive points. At each iteration, a new random point is generated in the neighborhood of the current point according to a neighborhood function. If the new candidate has a smaller objective function value (downhill move), this point is accepted and replaces the old one. However, in the opposite case (uphill move), the candidate may either be rejected or accepted depending on the control parameter, defined in terms of the Metropolis criterion [124]. If $\Delta f = f(x_{new}) - f(x_{current}) \leq 0$, in which f is the objective function, then accept the new point unconditionally. Otherwise, accept the new point with a probability of, (2.22),

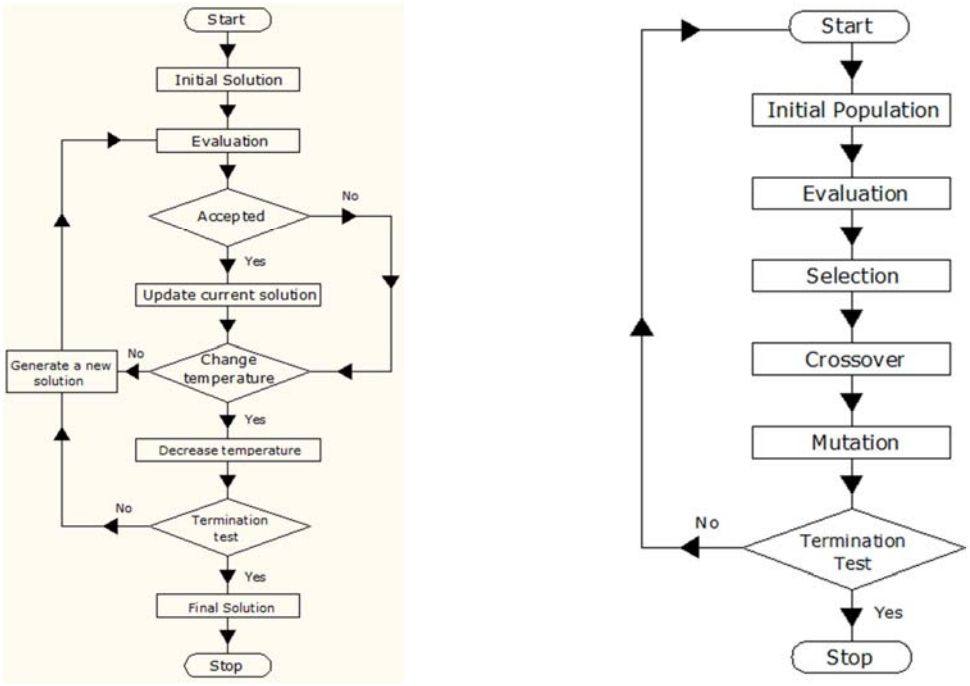
$$p_{Metr}(\Delta f) = e^{-\frac{\Delta f}{kT}} \tag{2.22}$$

where k is the Boltzmann constant and T is the control parameter or temperature.

When this algorithm is implemented, attention must be paid to the following choices: (1) representation of solutions; (2) definition of the objective function; (3) definition of the generation mechanism for the neighbors; (4) designing a cooling schedule. In designing the cooling schedule for a simulated annealing algorithm, four parameters should be specified. These are initial temperature, temperature updating rule, number of iterations to be performed at each temperature step and a stopping criterion for the search.

The SA process starts at a high temperature T_0. A sequence of iterates is then generated until the equilibrium is reached, that is when the average value of f remains stable as the number of iterations increase. The temperature T is then reduced by a cooling schedule and a new sequence of moves is made until again thermal equilibrium is obtained. The process is repeated until a sufficiently low temperature is reached, at which very few new moves are accepted. The algorithm flowchart is present in Figure 2.2a. The probability of accepting uphill moves is higher at initial stages of optimization, due to the initial higher temperatures, and is reduced later on, according to decreasing temperatures. Furthermore, the temperature is reduced slowly so as not to get trapped into a local minimum.

Various implementations of SA exist, based on different neighborhood functions and cooling schedules. A selection of four neighborhood functions is given by Levin and Lieven [103]. In the line adjustment, random moves are made along each coordinate direction x_i. The new coordinate values are uniformly distributed within the coordinate's valid range. In the fixed radius adjustment, a new point x_{new} is generated on a hyper sphere that is a fixed radius from the current point $x_{current}$. This approach requires an extra parameter, the radius.



a) b)

Figure 2.2. Optimization algorithm: a) simulated annealing; b) genetic algorithm.

Several cooling schedules have been studied in literature [73, 96, 191]. Levin and Lieven [103] describes a standard cooling schedule for SA that simply reduces the current temperature by a factor ρ ($0 < \rho < 1$). The choice of initial temperature T_0 depends on both objective and neighborhood function. An appropriate choice of T_0 can be made based on initial acceptance ratio, which is the ratio between the accepted moves over the total moves made. If the ratio is high, a significant part of the SA will be spent in the molten state implying unnecessary wasted effort. If this value is too low, then the risk of getting trapped in a local minimum increases. The success of SA relies on occasional acceptance of uphill moves avoid getting stuck in a local minimum.

2.4.2. Genetic algorithm

The basic genetic algorithm (GA) is a global search method, based on Darwin's natural evolution and the concept of survival of the fittest. In natural evolution, members of a population compete with each other to survive and reproduce successfully. If the genetic makeup of an individual member of a population gives that individual an advantage over its rivals, then it is more likely to breed successfully. The combination of genes that confer this advantage is likely to spread across the population. This is a natural optimization process that may also be simulated.

In GA the design vectors are represented as strings of binary digits, 0 and 1 [31, 125]. If each design variable x_i, i=1,…,n is coded in a string of length n_b, a design vector is represented with a string of total length n x n_b. In the standard binary encoding, a binary number given by $b_{n_b-1} \dots b_2 b_1 b_0$ where b_i = 0 or 1, denotes a decimal number x (integer), (2.23).

$$x = \sum_{i=0}^{n_b-1} 2^i b_i \tag{2.23}$$

Several bits are so needed to represent a real valued search variable. The used number of bits depends on the number range and needed precision. The sum of all bits, which represent a search variable, is called “gen”. The sum of all “genes” collected in a binary vector is called “chromosome”. The binary code leads to very long vectors combined with large search spaces when coding high precision variables in high dimensional problems. However, it is possible to apply real coding. In this situation, the search variable (“gene”) is saved as real number and subsequently collected in vectors (“chromosomes”). The disadvantage of this coding is that the classical procedure of crossover cannot be performed.

The mathematic optimization method acts on a population of “chromosomes”. Each “chromosome” is a representation of a design vector and its fitness value is given by the objective function. GA consists in generating a new population of “chromosomes” from the old population using three randomized operators that mimic those of natural evolution: selection, crossover and mutation. The nature of these operators is such that each subsequent generation tends to have an average fitness level higher than the previous one. A flowchart of GA is given in Figure 2.2b.

There exist two methods to define the initial solution of GA. The first consists of randomly producing solutions based in a uniform probability density function (PDF) within the given bounds. This method is preferable when no prior knowledge exists. The second method uses prior knowledge in such a way a set of requirements is obtained, and solutions which satisfy those requirements are collected to form an initial population. So, the GA starts the optimization with a set of approximately known solutions converging to an optimal solution in less time than the previous method.

Selection is a process in which the “chromosomes” are selected based on their fitness values relatives to that of population. During this process, each individual “chromosome” is assigned a probability of being selected (p_s) for copying as (2.24),

$$p_s \propto \frac{f_i}{\sum_{j=1}^{t} f_j} \tag{2.24}$$

where f_j is the objective function value of the j-th "chromosome" and t is the population size. The fittest "chromosomes" may typically be selected two or three times for mating and subsequent genetic action. This process reflects the principle of the survival of the fittest.

Two selection procedures [195] can be used: (1) proportional; (2) ranking-based. Proportional is usually called as "roulette wheel" due to the fact of the mechanism being reminiscent from the operation of a roulette wheel. Fitness values of individuals represent the widths of slots on the wheel. After a random spinning of the wheel, to select an individual for next generation, individuals in slots with large widths, representing higher fitness values, will have a higher chance to be selected. In ranking-based production procedure, each individual generates an expected number of offspring based on the rank of its fitness value [10].

Crossover is the operator that generates descendants based on proved individuals from last generation. The crossover operator mixes genetic information amongst the population and is implemented in two steps. First, the "chromosomes" are randomly paired together. Next, a crossover point is randomly selected along the string length of each pair of "chromosomes" and the binary digits following the crossover point are swapped between both "chromosomes". This recombination can be performed according to different crossover schemes (single-, point-, multipoint-, and shuffle-crossover). The crossover operator is applied with a specific probability (p_c). This operator has the potential of joining successful genetic fragments together to form fitter individuals.

The mutation operator randomly reverses the individuals bit values, according to a specific rate. Unlike crossover, this is an operator in which each child string is produced from a single parent string. The main objective is to find a new region of the search space and to avoid the convergence to a suboptimum. In real coding each "gene" is mutated according to a uniform PDF. The mutation operator has the potential to reintroduce genetic information that has been lost from the population. After crossover has occurred, each binary digit of each "chromosome" has a small probability of mutating ($p_m \approx 0.01 - 0.001$). A binary digit that mutates is simply inverted.

The population size is generally kept constant. Therefore it is necessary to decide which individuals should survive or substituted for next generation. There are different substitution schemes (substitution of complete generation, elitism, slight elitism, cancelation of n worst individuals, and many others). A direct method is to reprobate all invalid solutions with the disadvantage of increased computational time. An alternative is the implementation of the so-called "penalty functions".

Important control parameters include the population size, crossover and mutation rate. A large population size means the simultaneous handling of many solutions and increases the computational time. However, since many samples from the search space are used, the probability of convergence to a global optimum is higher. The crossover rate determines the frequency of crossover operation. It is useful at the start of optimization to discover a promising region. A low value decreases the speed of convergence to such an area. If this value is too high, it may lead to saturation around a solution. The mutation operator is controlled by the mutation rate. A higher value introduces high diversity in population and might cause instability. On the other hand, it becomes very difficult to find a global optimal solution with a too low value.

The capability of finding the global minimum, mainly due to the simultaneous search by a whole population of search points using randomized operators, such that the search space is widely explored, is an advantage of using GA. Moreover, the information exchange between selected pairs, directs the process towards the optimal point. The major drawback is that it requires a huge number of function evaluations.

2.4.3. Evolutionary strategies

Evolutionary strategies (ES) is a global search algorithm based on Darwin's natural evolution and in the concept of survival of the fittest. It consists in selecting a set of μ candidates for the optimization problem solution and applying the rules of evolution until an optimal solution is obtained. A typical candidate or an individual consists of a pair of vectors, one containing the parametric solution of the system (x) and other containing a vector of standard deviations (σ) which controls the evolution of the individual in subsequent steps. From these μ solutions, a batch of λ offspring's are generated according to the mechanisms of recombination and mutation. Then, these are evaluated according to the objective function value or optimizations criterion and they are ranked from best to worst. The best are then chosen, following a selection method to form the next parent population and the process is iterated.

2.4.3.1. Initial parent population

The first step of this algorithm is to define the initial parent population. This is carried out by picking μ individuals at random from the search space. The initial population $a^k = (x,\sigma)^k$ (k=1, …, μ) is defined by (2.25),

$$x_i^k = x_{min,i} + U_i(0,1) \cdot (x_{max,i} - x_{min,i}) \tag{2.25}$$

and (2.26),

$$\sigma_i^k = \left| x_i^k - \left(x_{min,i} + \frac{x_{max,i} - x_{min,i}}{2} \right) \right| \cdot \frac{1}{\sqrt{n}} \tag{2.26}$$

where $U_i(0,1)$ denotes a random number following an uniform PDF in interval [0,1] and n is the problem dimension . The initialization of the standard deviation is obtained by using expression $\sigma_i = \Delta x_i / \sqrt{n}$ where Δx_i denotes the estimated parametric distance to the optimum. The values of these deviations will evolve during the optimization process, making the choice of their initial values not of critical importance. However, such a choice must be reasonable as a too large value might make the algorithm diverge for a long time. Analogously, too small values will slow down the process until the deviations become large enough to make significant improvements in fitness.

2.4.3.2. Recombinant operator

Schwefel [167] has reported a remarkable acceleration in the search process, as well as, the facilitation of self-adaptation of parameters by introducing a recombination operator. Basically, it consists on recombining a set of chosen parents to find a new solution. A given number ρ of parents ($1 \leq \rho \leq \mu$) are randomly chosen for recombination. Thus, the nomenclature for ES can now be extended to ($\mu/\rho+\lambda$)-ES or ($\mu/\rho,\lambda$)-ES, where ρ represents the number of parents involved in the procreation of an offspring. These parameters (μ, ρ and λ) are designated by “exogenous strategy parameters” as they are kept constant during the evolution run.

This operation will distinguish ES from standard technique of GA. Unlike standard crossover in GA where two parents produces two offspring, the application of standard ES recombination operator to a parent family of size ρ produces only an offspring. Two recombination types exist: (1) intermediate; (2) discrete. The former consists in taking an individual from the parent population and holding it fixed while other parents are chosen to recombine with it. For each parameter, one mating individual is picked at random from the parent population. The parameters of the fixed and the mating parent are then weighed with a random factor from the interval [0,1] so that new offspring’s parameter might be at any intermediate point between its parent’s parameter values. This process is mathematically explained by (2.27),

$$x_i' = x_{R,i} + U_i(0,1) \cdot (x_{Q,i} - x_{R,i}) \tag{2.27}$$

and (2.28),

$$\sigma_i' = \sigma_{R,i} + U_i(0,1) \cdot (\sigma_{Q,i} - \sigma_{R,i}) \tag{2.28}$$

where an apostrophe denotes the result of applying the genetic operator. *Q* and *R* denote the fixed and the mating parent. Discrete recombination means that each component of the offspring is chosen from one of the ρ parents at random.

2.4.3.3. Mutation operator

In order to introduce new information into the population pool, the mutation operator is used. It consists in slightly perturbing the parameters of the offspring individuals after they have been generated by the recombination procedure. During the search, the step sizes for mutation are adapted. Several self-adaptation schemes are possible. A possibility is to update the standard deviation σ_i for each decision variable. Expressed in mathematical terms, the mutation operator (Gaussian mutation) for the *i*-th parameter is defined by (2.29),

$$\sigma_i' = \sigma_i exp[\tau' N(0,1) + \tau N_i(0,1)] \tag{2.29}$$

and (2.30),

$$x_i' = x_i + \sigma_i' N_i(0,1) \tag{2.30}$$

where $N(0,1)$ denotes a random sample from a Normal PDF with zero mean and a unitary standard deviation. The values of τ and τ' appear to be rather robust and they can be picked as (2.31),

$$\tau \propto \left(\sqrt{2\sqrt{n}}\right)^{-1} \tag{2.31}$$

and (2.32),

$$\tau' \propto \left(\sqrt{2n}\right)^{-1} \tag{2.32}$$

where the proportional constants are usually unitary. To guarantee a minimum variation in the mutation of the parameters, all σ_i are required to remain above a certain threshold value. As suggested by Schwefel [167], this minimum deviation should be expressed as a percentage of the parameter value, (2.33).

$$\sigma_i \geq \%|x_i| \tag{2.33}$$

The standard deviations σ_i are updated using the 1/5 success rule. This rule can be formulated in the following manner: “from time to time, during the optimum search, obtain the frequency of successes or the ratio of the number of successes to the total number of trials. If the ratio is greater than 1/5, increase the variance, if it is less than 1/5, decrease the variance”. Assuming this rule to be applied periodically, for each *p* generations, it can be expressed as, (2.34),

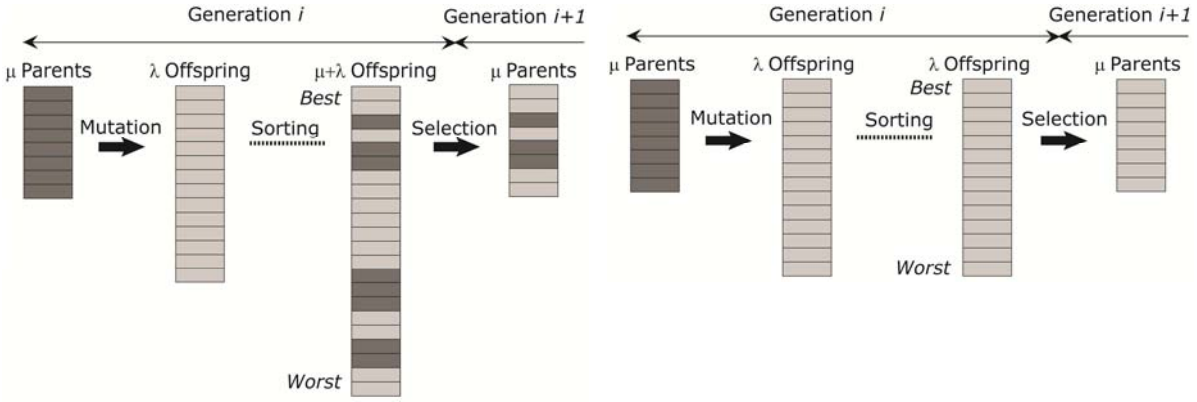
$$\sigma_i^{(k+1)} = \begin{cases} c_{dec}\sigma_i^{(k)} \leftarrow Pr_{suc}(p) < \frac{1}{5} \\ c_{inc}\sigma_i^{(k)} \leftarrow Pr_{suc}(p) > \frac{1}{5} \\ \sigma_i^{(k)} \leftarrow Pr_{suc}(p) = \frac{1}{5} \end{cases} \tag{2.34}$$

where $Pr_{suc}(p)$ is the success rate during the last p generations and $c_{dec} < 1$ and $c_{inc} > 1$ are the decreasing and increasing factors of the standard deviations σ_i. Mutation is an internal parameter which is self-adapted, and optimized, during the evolution, together with other parameters. This intrinsic characteristic differentiates ES from standard GA in which the mutation rate is a fixed external parameter [29, 30].

2.4.3.4. Selection operator

The selection operator in ES is deterministic, contrasting with some selection mechanisms often used in GA. Mainly, it is necessary to distinguish between elitist and non-elitist selection operators. Elitism, or (μ+λ)-ES, dictates that the old parent individuals will be pooled together with the new-offspring individuals and then the ranking of all μ+λ individuals will be performed according to their fitness value. The best fitted μ individuals, selected from the pool, will substitute the old parent population. This technique guarantees survival of the best adapted individuals but also hinders evolution if these apparently well-adapted individuals approach a local optimum instead of the global one. In contrast, non-elitism, or (μ,λ)-ES, dictates the ranking to be performed only on the offspring population of λ individuals and the best μ of these will substitute the old parent population. Thus, this selection procedure might lose well adapted parent individuals but it provides also the power to leave local optima in search for the global optimum.

More refined selection methods have been introduced, out of which, the fitness-based-reinsertion combined with the elitist selection is recommended. In this method, the λ offspring's are ranked in terms of their fitness and the best μ-γ ($\gamma < \mu$) are selected to become part of the next parent generation. The remaining γ slots in the parent population to complete the μ individuals are filled by the best parents of the older generation. This technique represents an intermediate scenario between elitist and non-elitist selection and it aims to capture the best of both methods. Figure 2.3a illustrates the (μ+λ)-ES while Figure 2.3b presents the (μ,λ)-ES selection procedure.



a) b)

Figure 2.3. Evolutionary strategies: a) (μ+λ); b) (μ,λ).

2.4.3.5. Tolerance criteria

In order to halt the process and accepts the best found individual as the solution to the optimization problem, one or several criteria have to be established. As tolerance criteria we may have: (1) maximum number of generations; (2) maximum computational time; (3) convergence in the space of fitness value; (4) convergence in object parameter; (5) convergence in strategy parameter. In respect to objective function, this algorithm evaluates its minimum value for generation *i* (f_i) and *i*+*n* (f_{i+n}), and determines its improvement (Δf). A threshold value (ε), commonly designated by algorithm precision, is then defined for this criterion through expression (2.35).

$$\Delta f = |f_{i+n} - f_i| \leq \varepsilon \rightarrow f_{i+n} \leq f_i \pm \varepsilon \tag{2.35}$$

The gap between generations is computed by the product of the maximum number of generations (MaxGen) by a tolerance value (see Figure 2.4).

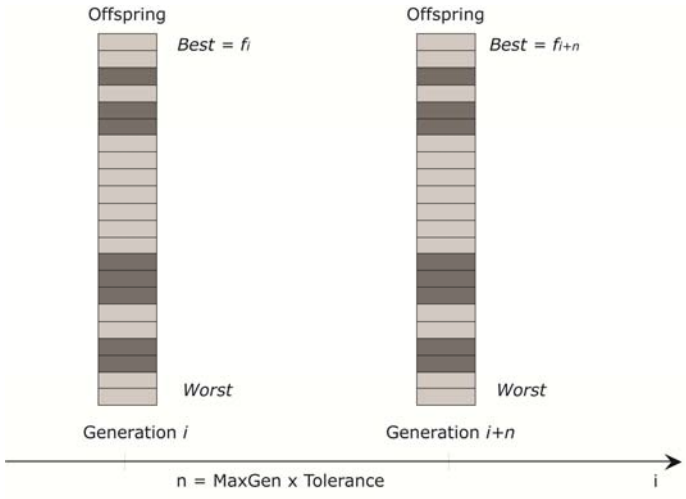


Figure 2.4. Objective function criterion.

2.5. Example

The previous presented optimization algorithms were tested with three analytical functions by assuming four different starting points or starting point sets. For each initial point or set, five runs were developed. An average fitness value, the number of function evaluations and the required computational time is then determined. Regarding ES both (+) and (,) selection operators were considered. Table 2.1 presents the algorithm parameters.

Table 2.1. Algorithm parameters.

Parameters	SQP	SA	GA	ES (+)	ES (,)
Maximum number of function evaluations	1000	1000	*	*	*
Maximum number of iterations	1000	1000	*	*	*
Maximum number of generations	*	*	1000	1000	1000
Initial temperature (T_0)	*	100	*	*	*
Mutation probability (p_m)	*	*	$1.00*10^{-3}$	*	*
Crossover probability (p_c)	*	*	0.80	*	*
Parent population (μ) (number of individuals)	*	*	10	10	10
Parent for recombination (ρ) (number of individuals)	*	*	*	10	10
Offspring population (λ) (number of Individuals)	*	*	*	50	50

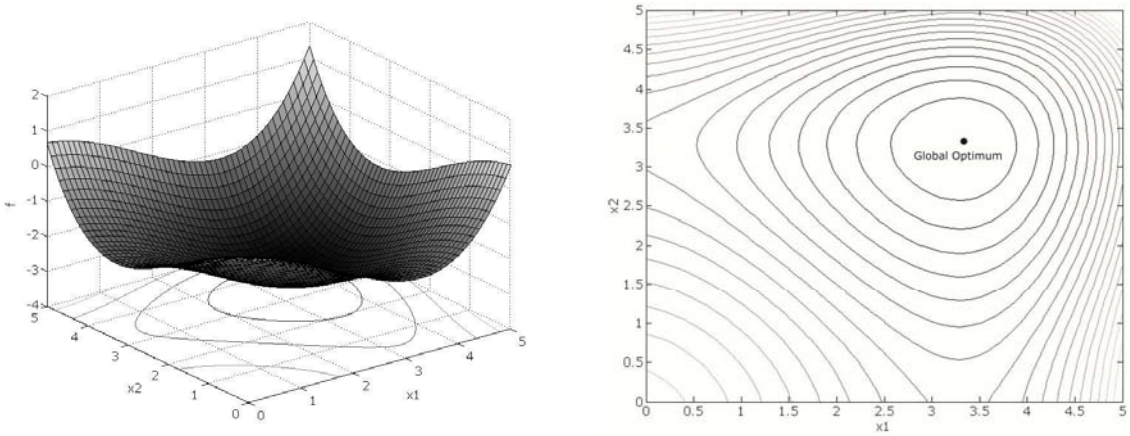
* Not applicable.

2.5.1. Function 1

The first test function is a 2D well-shaped function with a clear global minimum defined as, (2.36),

$$f(x_1, x_2) = 0.01 \cdot \sum_{i=1,2} \left[(x_i + 0.5)^4 - 30 \cdot x_i^2 - 20 \cdot x_i\right] \tag{2.36}$$

with $-5 \leq x_i \leq 5$ ($i = 1, 2$). This function is presented in Figure 2.5a. The global minimum of this function f is at $x^* = (3.29; 3.29)$ and is represented in contour plot in Figure 2.5b. The obtained value for this point is of -3.68.



a) b)

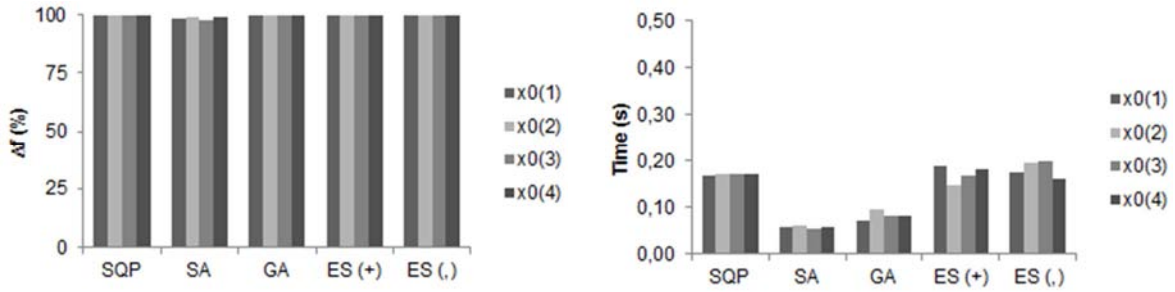
Figure 2.5. Function 1: a) 3D plot; b) contour plot.

The average of results obtained for each algorithm is presented in Table 2.2. The second column shows the gap (Δf) between the fitness and the real value. Figure 2.6 indicates obtained results, for each initial point or set $x_0(i)$, represented by a plot bar graphic.

Table 2.2. Obtained results.

Optimization algorithm	Δf [%]	Number of function evaluations [-]	Computational time [*s*]
SQP	100.00	30	0.17
SA	98.71	193	0.06
GA	100.00	1020	0.08
ES (+)	100.00	1010	0.17
ES (,)	100.00	1010	0.18

Obtained results indicate that all the four algorithms reached the global minimum. In respect to required computational time, it is verified that local search techniques present a higher value due to the need to compute the numerical derivatives of objective function in each step. Identically, ES presents a high value as they perform the search in a wider range of pool of points. SA is the algorithm that requires less computing time as it is based in a more guided search procedure.



a) b)

Figure 2.6. Function 1: a) fitness value gap (%); b) computational time (*s*).

2.5.2. Function 2

The second test function is a 2D polar coordinate composite function with several local minima and a clear global minimum. This function is described by (2.37),

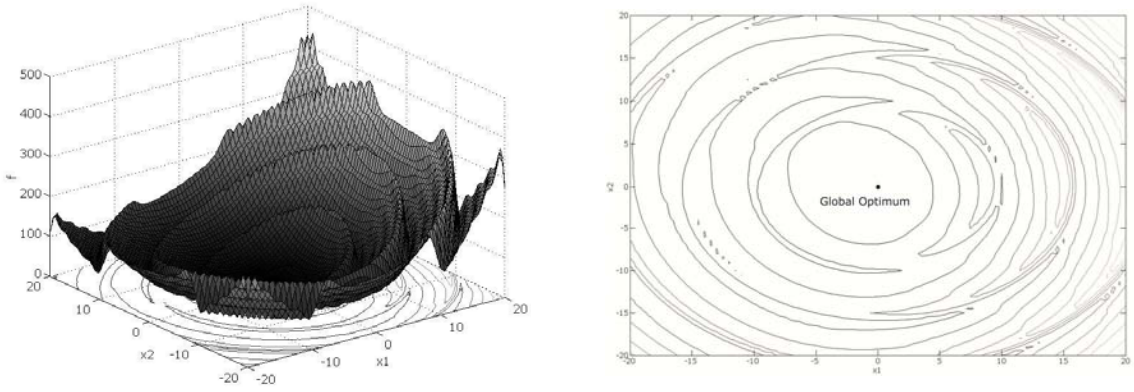
$$f(x_1, x_2) = g(x_1) \cdot h(x_2) \tag{2.37}$$

when $g(x_1)$ and $h(x_2)$ are respectively described by (2.38) and (2.39),

$$g(x_1) = (\sin x_1 - \sin 2x_1/2 + \sin 3x_1/3 - \sin 4x_1/4 + 4) \cdot {x_1}^2/(x_1 + 1) \tag{2.38}$$

$$h(x_2) = 2 + \cos x_2 + (\cos 2x_2 - 1/2)/2 \tag{2.39}$$

with -20 ≤ x_i ≤ 20 (i = 1, 2). This function is plotted in Figure 2.7a. The global minimum of function *g* is obtained for x_1 = 0, with a value of 0, while function *h* presents two local minima, one of them, global. The global minimum of function *f*, indicated at contour plot in Figure 2.7b, is at x^* = (0; 0) with a value of 0.



a) b)

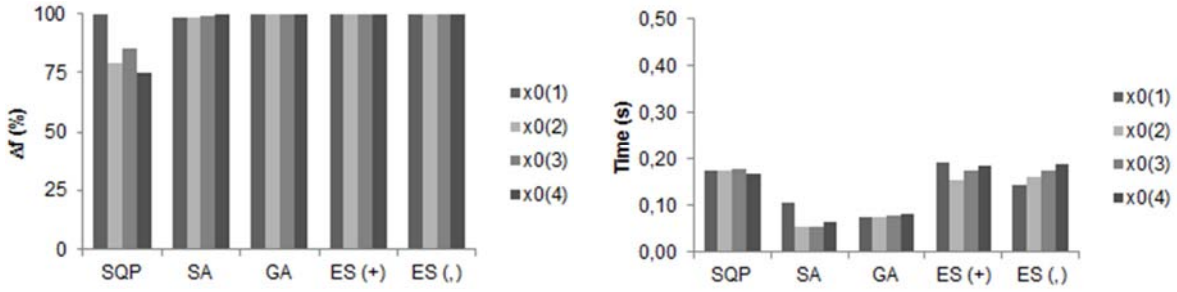
Figure 2.7. Function 2: a) 3D plot; b) contour plot.

Table 2.3 presents obtained average results for each algorithm. The second column shows the gap (Δf), between the fitness and the real value. Figure 2.8 indicates the results, determined for each initial point or set $x_0(i)$, and illustrated as a plot bar graphic.

Table 2.3. Obtained results.

Optimization algorithm	Δf [%]	Number of function evaluations [-]	Computational time [*s*]
SQP	84.87	78	0.17
SA	98.86	217	0.07
GA	100.00	1024	0.08
ES (+)	99.99	1010	0.18
ES (,)	99.96	1010	0.17

From a first analysis it is possible to indicate that all global search algorithms present very good results. The performance of used local search technique was not as good as the others as, in some situations, a local minimum is identified. Additionally, this algorithm is too much dependent of considered starting point. In terms of computational time all the analyzed algorithms present identical values. SA is the one that involves less computing time due to a direct search procedure.



a) b)

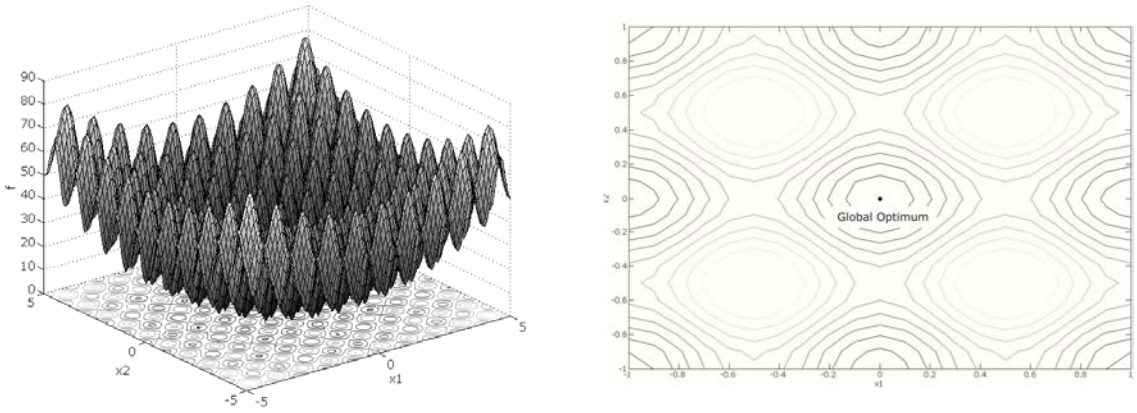
Figure 2.8. Function 2: a) fitness value gap (%); b) computational time (*s*).

2.5.3. Function 3

The third function to be analyzed is a 2D function commonly designated as Rastrigin's Function [187]. This function has several local minima and one global. It is described by the following expression, (2.40),

$$f(x_1, x_2) = 20 + \sum_{i=1,2} \left[x_i^2 - 10 \cdot (\cos 2\pi x_i)\right] \tag{2.40}$$

with $-5 \leq x_i \leq 5$ (i = 1, 2). This function is plotted in Figure 2.9a. The global minimum of function f, indicated in the contour plot in Figure 2.9b, is at $x^* = (0; 0)$ with a value of 0.



a) b)

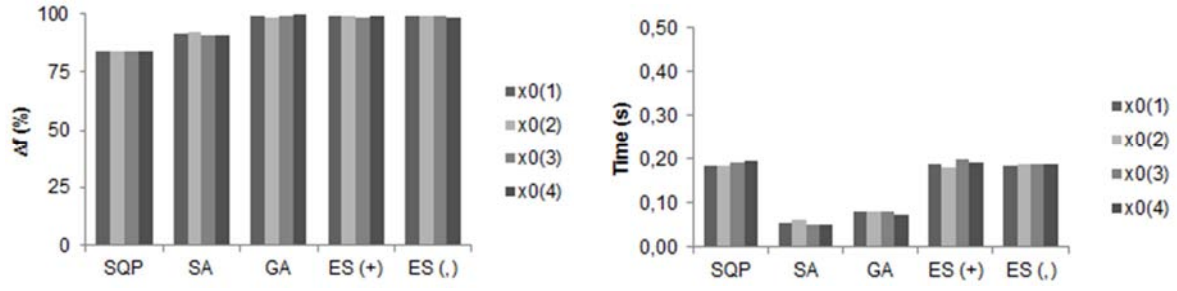
Figure 2.9. Function 3: a) 3D plot; b) contour plot.

Table 2.4 presents obtained average results for each algorithm. The second column shows the gap (Δf), between the fitness and the real value. Figure 2.10 indicates the results, determined for each initial point or set $x_0(i)$, and illustrated as a plot bar graphic.

Table 2.4. Obtained results.

Optimization algorithm	Δf [%]	Number of function evaluations [-]	Computational time [s]
SQP	84.08	50	0.19
SA	91.21	199	0.06
GA	99.00	1030	0.08
ES (+)	99.05	1010	0.19
ES (,)	99.05	1010	0.19

The results pointed out the main limitation of local search techniques, as, for the analyzed function, the SQP converges to the solution close to the starting basin. Therefore, the performance of this algorithm is directly dependent of the chosen starting point. SA did not find the global minimum either but it presents better results. In fact the pool of points in which SA performs it search is not as large as it should be. However, SA presents a reduced computational cost. All the EA present very good results as they all found the global optimum. In respect to computing costs, GA presents a lower value. For global search techniques, the obtained results are independent of considered starting point.



a) b)

Figure 2.10. Function 3: a) fitness value gap (%); b) computational time (*s*).

2.6. Real application

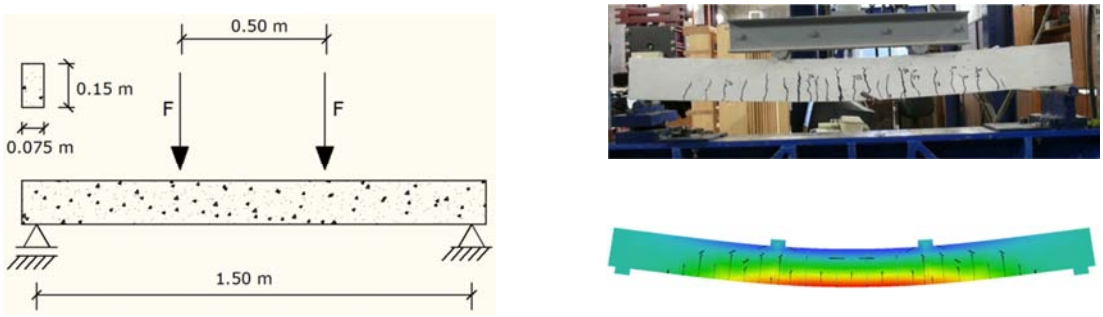
In order to study the applicability of these algorithms to structural problems, an example, a reinforced concrete beam loaded up to failure in laboratory is presented [114, 115, 116, 117]. In this example, the optimization algorithms are used to minimize the difference between numerical and experimental results, as (2.41),

$$f = \sum_{i=1}^{n} \left| y_i^{num} - y_i^{\exp} \right| \Big/ \max\left(y^{\exp}\right) \cdot 1/n \,[\%] \tag{2.41}$$

where y^{num} and y^{exp} are the numerical and experimental values and n the number of evaluated points. Such minimization procedure, usually designated as model identification [6, 7], is developed by continuously changing the model critical parameters, the ones that present a higher influence in the structural behavior. This behavior is highly nonlinear due to the constitutive materials. In this situation the optimization function f is multi-parametric and nonlinear. Used algorithm parameters are indicated at Table 2.1. Each algorithm was processed for 10 times, being represented the obtained best result from the whole population of models.

2.6.1. Experimental test

A set of two pinned-pinned reinforced concrete beams, characterized by a longitudinal reinforcement of 3ϕ6 (A_s = 0.848 cm^2) and a transversal reinforcement of ϕ4@0.10 (A_{sw}/s = 2.513 cm^2/m), were loaded in laboratory up to failure. In design phase, C25/30 concrete and S500B steel reinforcement were specified, according to EN 1992-1-1 [48]. The tested beams present a rectangular section of 75 x 150 mm^2 and a 1.50 m span length (see Figure 2.11a). The concrete cover was considered to be 10 mm in all sides. Applied loads were located at 1/3 and 2/3 of the span length.



a) b)

Figure 2.11. Real application: a) scheme; b) collapse mechanism (top: experimental; bottom: numerical).

Each beam was supported in two points. While one of those elements restricts only the vertical displacement, the other restricts both vertical and horizontal one. The laboratory test was developed with displacement control. During the test the applied load and the middle span displacement were monitored. A bending failure mode with concrete crushing and yielding of longitudinal steel reinforcement was obtained. The collapse mechanism, presented in Figure 2.11b, is characterized by a plastic hinge located at middle span. An average value of 24.79 kN is obtained for failure load (F_R).

2.6.2. Numerical model

A nonlinear numerical model of the tested beams was developed in ATENA® [22, 23]. A uniform finite element mesh, composed by quadrilateral elements, was used. The steel reinforcement was considered to be completely embedded in concrete elements. Two different load cases were considered, one in which the supports were inserted and other in which the applied load is represented. A steel plate was introduced in both supports and applied load points, to avoid concentration of stresses. The middle span displacement and the applied load were measured during the analysis.

2.6.3. Obtained results

Obtained results from running those different optimization algorithms are presented further. The collapse mechanism is, for all algorithms, characterized by a plastic hinge at beam middle span (see Figure 2.11b). A bending failure mode, with concrete crushing and yielding of longitudinal steel reinforcement, is obtained. In Figure 2.12 the applied load is plotted against the middle span displacement for experimental and numerical results, considering each optimization algorithm. The objective of these algorithms is to give the curve that best fits the experimental data. In order to do so, they start with an initial vector, equal for all

algorithms, which correspond to the values considered in design (nominal values), and iteratively change these values. The optimal vector, obtained from each algorithm, and also the initial one are given in Table 2.5. In the same table, between brackets, the bias factor, which represents the ratio between the identified and the nominal value for each variable, is also presented.

In this situation the identified model parameters are: (1) concrete elasticity modulus (E_c); (2) concrete tensile strength (f_t); (3) concrete compressive strength (f_c); (4) reinforcing steel elasticity modulus (E_s); (5) reinforcing steel yield strength (σ_y); (6) reinforcing steel area (A_s); (7) section width (b); (8) section height (h); (9) inferior concrete cover (c_{inf}).

Table 2.5. Model parameters.

Numerical model					Nominal value	Optimization algorithm				
						SQP *	SA *	GA *	ES (,) *	ES (+) *
Parameter	Material	Concrete	E_c	[GPa]	31.00	30.97 (1.00)	29.10 (0.94)	30.42 (0.98)	31.54 (1.02)	31.44 (1.01)
			f_t	[MPa]	2.60	2.60 (1.00)	2.53 (0.97)	2.44 (0.94)	2.64 (1.02)	2.54 (0.98)
			f_c	[MPa]	33.00	32.95 (1.00)	34.36 (1.04)	33.98 (1.03)	31.09 (0.94)	29.87 (0.91)
		Longitudinal steel reinforcement	E_s	[GPa]	200.00	197.78 (0.99)	156.99 (0.78)	172.56 (0.86)	174.70 (0.87)	186.73 (0.93)
			σ_y	[MPa]	500.00	499.71 (1.00)	486.92 (0.97)	529.91 (1.06)	520.32 (1.04)	535.83 (1.07)
			A_s	[cm^2]	0.85	0.85 (1.00)	0.93 (1.09)	0.87 (1.02)	0.90 (1.06)	0.91 (1.07)
	Geometry		c_{inf}	[cm]	1.00	1.00 (1.00)	1.04 (1.04)	0.97 (0.97)	1.02 (1.02)	0.99 (0.99)
			b	[cm]	7.50	7.49 (1.00)	7.14 (0.95)	7.11 (0.95)	6.90 (0.92)	7.27 (0.97)
			h	[cm]	15.00	14.96 (1.00)	15.23 (1.02)	14.87 (0.99)	14.70 (0.98)	14.28 (0.95)

* Bias factor is presented between brackets.

In a first analysis it is possible to conclude that global search techniques give better results than the local one. In fact, the latter provides a result that does not differentiate from the starting point. By analyzing the results from Table 2.5 it is important to identify the following: (1) concrete parameter values are close to each other and to the nominal ones; (2) reinforcing steel elasticity modulus (E_s) is always inferior to the nominal value; (3) reinforcing steel yield strength (σ_y) is, in general, higher than the nominal value;

(4) reinforcing steel area (A_s) is always higher than the nominal value; (5) section dimensions are, in general, inferior to nominal values; (6) inferior concrete cover (c_{inf}) values are close to each other and to the nominal one.

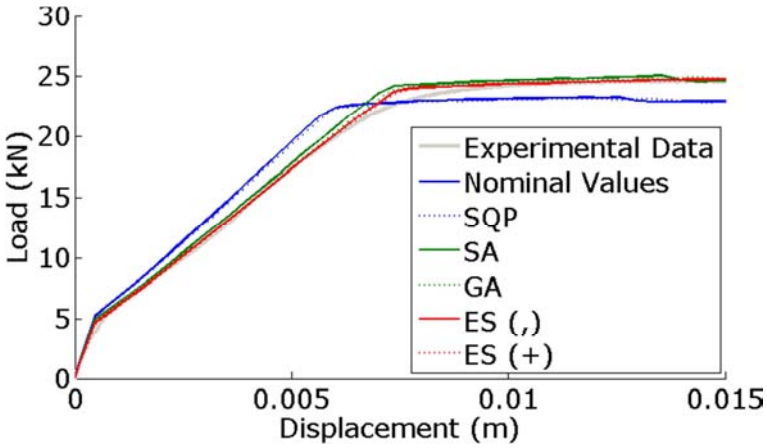


Figure 2.12. Obtained results.

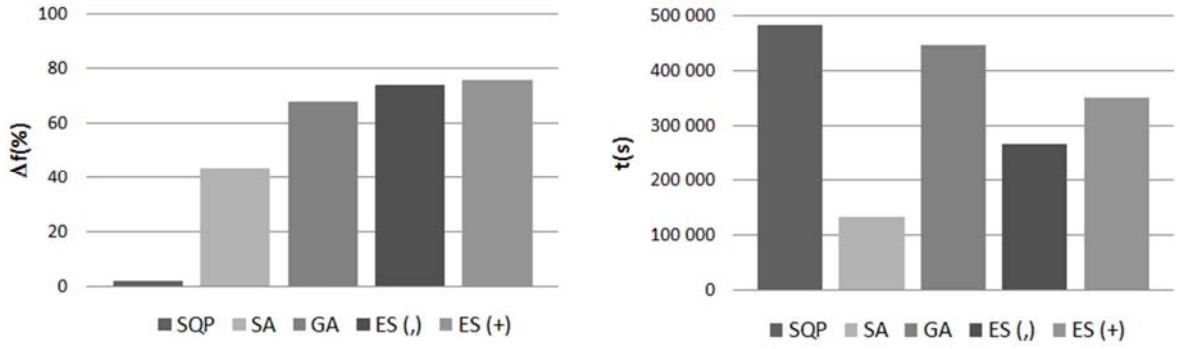
Two bar graphics, one representing the improvement in fitness value (Δf), according to expression (2.41), and other the computational time, are respectively presented in Figure 2.13a and 2.13b. The fitness value, obtained from the application of each algorithm, and the respective improvement are indicated in Table 2.6.

Table 2.6. Obtained results.

Numerical model		Fitness function	
		Value [%]	Improvement [%]
Nominal values		7.70	-
Optimization algorithm	SQP	7.54	2.06
	SA	4.34	43.66
	GA	2.47	67.92
	ES (,)	2.01	73.91
	ES (+)	1.86	75.90

From the analysis of obtained results it is possible to conclude that: (1) local search techniques fail to find global minimum. Therefore, these techniques cannot be used in nonlinear multi-parametric models; (2) SA gives poor results but presents a very low computational time. This algorithm, although belonging to the global search techniques, is based in a directional search procedure reducing so the number of searching points; (3) It is

verified that the performance of EA is extremely good for nonlinear multi-parameter optimization algorithms, being, the best results obtained with ES (+) algorithm; (4) GA present a high computational cost reducing so the respective efficiency for this kind of problems. It is important to note that by applying EA, an improvement of around 70% is achieved for the fitness value. Table 2.7 indicates the failure load (F_R) obtained from the application of different optimization algorithms. EA provide the most suitable results, being, the best results obtained with ES (+) algorithm.



a) b)

Figure 2.13. Obtained results: a) fitness value improvement (%); b) computational time (*s*).

Table 2.7. Failure load (F_R).

Numerical model		Failure Load	
		Value [kN]	Error [%] *
Nominal values		23.28	6.13
Optimization algorithm	SQP	23.22	6.37
	SA	25.03	0.93
	GA	24.84	0.16
	ES (,)	24.89	0.36
	ES (+)	24.84	0.16

* Comparing with the real failure load.

2.7. Conclusions

The use of optimization algorithms in structural engineering analysis grew within the last years, especially in the model identification field. Within this procedure, an objective function is minimized in order to obtain a numerical curve that best fits the acquired experimental data. In order to do so, model input parameters are iteratively changed so that an optimum

combination of values is identified. Once this global minimum is identified, engineers become able to interpret the structural behavior in a more reliable way.

However, this objective function is usually nonlinear and multi-parametric, which makes model identification a complex problem. Therefore, the identification of an appropriate optimization algorithm is the core issue of a correct model identification procedure. These algorithms can be classified into local or global ones. The former usually requires the computation of the objective function derivative to determine its minimum, while the latter are stochastically based and only dependent of the fitness value.

Within this chapter it is studied the sequential quadratic programming (SQP), which belongs to the local search algorithms. From the set of global search algorithms, the simulated annealing (SA), the genetic algorithms (GA) and the evolutionary strategies (ES) are analyzed. It is also evaluated the performance of two ES selection procedures, respectively, the ES (+) and the ES (,) algorithms.

These algorithms are then tested with three simple analytical functions in order to point out their advantages and disadvantages. As the objective of this chapter is to analyze the applicability of each algorithm in a structural model identification problem, a real application, a reinforced concrete beam loaded up to failure in laboratory, is respectively introduced.

Obtained results revealed that global search methods are, in general, more robust than local ones, as the choice of the starting position has little influence on the final result, and they can find the global solution with a high probability. Accordingly, in these methods the risk of being trapped into local minima is reduced. For typical optimization problems the computational effort of global search techniques is higher, which makes local search techniques more appropriate when time is an important concern. However, this is not true when the number of parameters to optimize and the objective function nonlinearity increase. In such situations the number of necessary derivatives in local search methods increases and, consequently, the respective computational time.

In civil engineer applications, specially, in structural model identification, in which the objective function depends from several parameters and is highly nonlinear, global search techniques are so more suitable. From those algorithms SA is the one that involves less computing time but, at same time, it might not find the global minimum due to the reduced pool of points in which performs its search. ES are the most efficient algorithms in terms of computational cost. This is confirmed with the results obtained by Costa and Oliveira [29, 30]. In this real application the best results are obtained with ES (+) algorithm.

3 Bayesian Inference

3.1. Introduction

The data analysis principles and methodologies that derive from subjective uncertainty sources are often mentioned as Bayesian statistics and they were studied in the last decades by several authors [15, 19, 28, 145]. These authors present the Bayesian inference of several probabilistic distributions and examples of its application. One of these examples is in the updating of probabilistic numerical models. This may be further incorporated into structural assessment algorithms, in which data acquired from permanent monitoring systems is used to update numerical models [80, 149, 175].

Two interesting works were developed in Portugal in this field. One of them was developed by Miranda [127], who presented a Bayesian framework for updating geomechanical parameters. This framework is applied to the deformability modulus updating of large underground structures. The second is the work of Jacinto [87], who used Bayesian inference techniques to reduce the statistical uncertainty of structural assessment algorithms. For this author, although statistic uncertainty may be reduced with additional collected data, it is always important to evaluate its impact in structural assessment in order to study if collecting such data will change results at all.

The Bayesian perspective of probability is different from the frequentist one. While the latter takes the perspective that probability is an objective concept, the former indicates that probability is the individual degree of belief that a given event will occur. Frequentist

interprets each parameter as a fixed but unknown quantity, while Bayesian regards it as a distribution of possible values. For the latter, the probability function reflects the degree of belief on where the true parameters may be. If this distribution is narrow then the confidence about the parameter location is high.

The frequentist data analysis methodologies are computationally simpler and this is why its use is more widespread. However, Bayesian techniques allow the update of random parameters in a sequential way, when new data is available. This process is divided in [33]: (1) setting up a joint probability distribution for all parameters, consistent with knowledge; (2) computing the conditional posterior distribution of relevant parameters, given new observed data; (3) evaluating the model fit to such data, analysing if conclusions are reasonable and how sensitive they are to modelling assumptions. The obtained posterior distribution is a compromise between prior information and the one contained in new data. This compromise is controlled by the data sample size. The posterior is thus the updated random parameter distribution.

In a probabilistic numerical model the material, geometric and physic parameters are usually characterized by random distributions. These distributions are introduced to represent physical uncertainties. In Bayesian approach, and unlike the frequentist one, the distribution parameters are also considered to be unknown, being represented by random distributions too [127]. Those distributions reflect the statistical uncertainty or, in other words, the degree of belief of each distribution parameter. Therefore, they can be updated given the data and used to infer each probabilistic numerical model parameter distribution.

Figure 3.1 presents a general scheme concerning the stages where an updating procedure can be applied to revaluate any structural parameter. In this case, the procedure is particularly applied to the concrete compressive strength (f_c). In the initial stages, this parameter can be evaluated based on preliminary research (f_{c1}). During construction more information may be gathered from laboratory tests which can be used to update the prediction ($f_{c\ actual\ stage}$). This parameter can be incorporated in probabilistic numerical model, at any time, for structural assessment purposes. During exploitation phase new information concerning this parameter can be obtained from several sources (e.g. monitoring systems). This information may be used in a dynamic process that improves such parameter prediction (f_{ci}) as the quantity of data increases ($f_{c\ actual\ stage}$). Therefore, the probabilistic numerical model and obtained results from structural assessment are continuously updated.

A general Bayesian framework for updating numerical model parameters is further presented. This framework is applied to update the initial prediction on lightweight concrete compressive strength (f_c). This updating procedure is based in data obtained from laboratory

characterization tests. In this approach, this parameter is considered as a random variable with a Normal or Lognormal distribution. Different levels of initial information, uncertainty levels and degree of belief in initial assumptions were considered, being the obtained results compared to evaluate its sensitivity to prior concerns. In order to overcome the problem of choosing a given probability distribution function, an alternative Bayesian methodology that uses a Weibull distribution, is also presented.

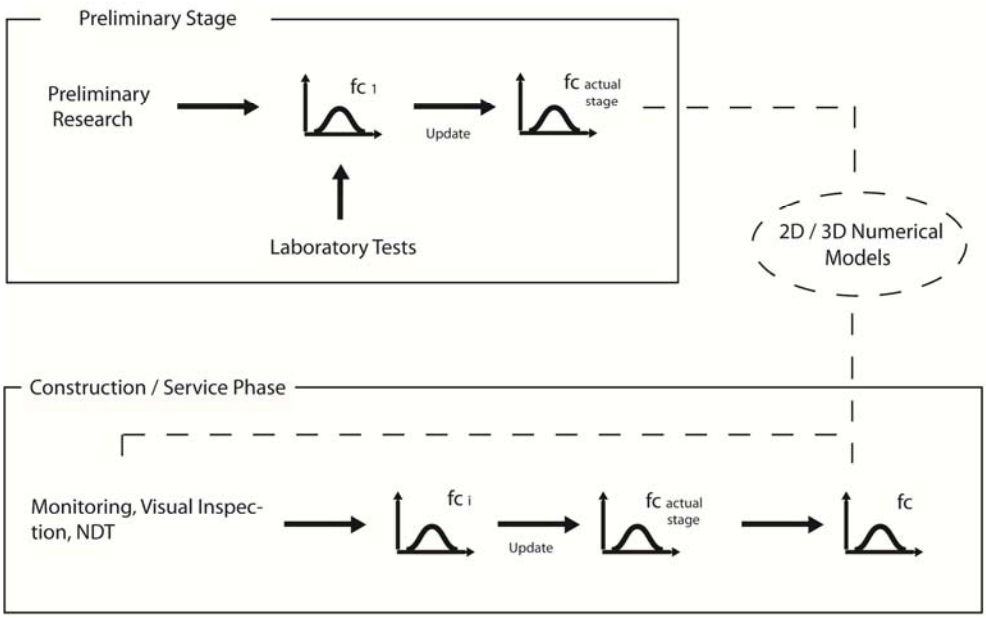


Figure 3.1. Updating procedure for structure compressive strength (f_c) (random distributions are merely indicative).

3.2. Bayes theorem

Bayesian methods provide tools to incorporate external information into data analysis process [15]. This process starts with a given distribution, designated as prior, whose parameters may be chosen or estimated based on bibliography, experience or from previous experimental results. This distribution represents the uncertainty of a given variable. As more data is collected, Bayesian analysis is used to update the prior distribution into a posterior distribution. The Bayes theorem, which weights the prior information with evidence provided by new data, is the basic tool for the updating procedure. This process is shown in Figure 3.2.

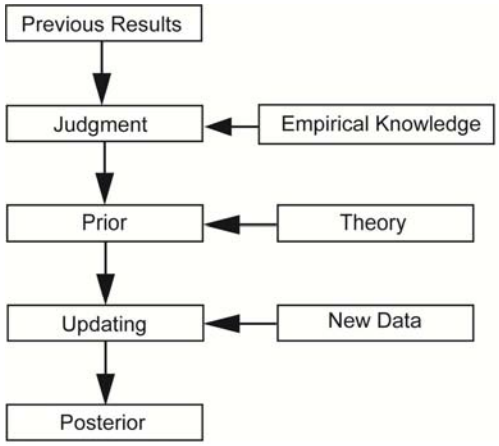


Figure 3.2. Updating procedure scheme, adapted from Faber [53].

The discrete version of Bayes theorem is given in expression (3.1). This is only possible when the prior distribution of parameter Θ with n possible outcomes is discrete, and the new information x comes from a discrete model too,

$$P\left(\theta|x\right)=\frac{P\left(x|\theta_i\right)\cdot P\left(\theta_i\right)}{\sum_{i=1}^{n}P\left(x|\theta_i\right)\cdot P\left(\theta_i\right)} \tag{3.1}$$

where $P(\Theta)$ is the prior distribution, which indicates the prior beliefs about the parameter values, $P(x|\Theta)$ is the conditional probability or likelihood of data given Θ and $P(\Theta|x)$ is the posterior distribution of Θ given observed data x.

The more usual form of this theorem, in terms of continuous variables, is indicated at expression (3.2). In this situation, the prior and the posterior distributions of Θ are represented by density functions, respectively, $p(\Theta)$ and $p(\Theta|x)$.

$$p\left(\theta|x\right)=\frac{p\left(x|\theta\right)\cdot p\left(\theta\right)}{\int_{\theta}p\left(x|\theta\right)\cdot p\left(\theta\right)d\theta},\ \theta\in\Theta \tag{3.2}$$

The joint probability distribution of the data and the parameter is given by $p(x|\Theta)$, which is called the likelihood, $L(\Theta)$, and is defined by expression (3.3),

$$p\left(x|\theta\right)=L\left(\theta\right)=\prod_{i}p\left(x_i\mid\theta\right) \tag{3.3}$$

being assumed that all observations x_i are independent. The integral of expression (3.2) is a normalizing constant. Therefore, this expression can be written as (3.4),

$$p\left(\theta\mid x\right)\propto p\left(x\mid\theta\right)\cdot p\left(\theta\right) \tag{3.4}$$

Accordingly, the Bayes theorem consists in multiplying the prior and the likelihood and then normalizing it in order to get the posterior distribution.

3.3. Prior distributions

One of the main issues of the Bayesian approach and part of the modeling process is the choice of a prior distribution. Several alternatives exist for the prior distribution, but it is important to verify the impact this choice in the stability of the posterior distribution. If the posterior distribution is highly dependent on the prior, then acquired data may not contain sufficient information. On the other hand, if it is relatively stable over a choice of priors, it means that data contains significant information. The prior should reflect [53]: (1) known observations of random variables, from which parameter estimates of prior distribution can be computed; (2) subjective knowledge on parameters distribution.

It is possible to choose a prior distribution that presents a range of situations from high (small standard deviation) to reduced knowledge (large standard deviation) and even no knowledge. In this latter situation, the prior is designated as non-informative, as it does not have any impact on the parameter posterior distribution. In this case it can be represented by a constant, as expression (3.5),

$$p(\theta) = c = \frac{1}{b-a} \quad for \quad a < \theta < b \tag{3.5}$$

assuming *a*, *b* and *c* as any value from the real space. In this situation, and considering expression (3.4), the posterior is just one constant times the likelihood, (3.6),

$$p(\theta \mid x) \propto L(\theta) \tag{3.6}$$

The use of a non-informative prior is often useful but it is always necessary to check if the obtained posterior distribution is proper [145]. A distribution is proper if it integrates to unity. It is verified that traditional non-informative priors are improper, (3.7),

$$\int p(\theta)\, d\theta = \infty \tag{3.7}$$

An improper prior may, or not, result in an improper posterior. Thus it is always important to check if the posterior presents a finite integral. A common non-informative prior is the Jeffrey's prior. In order to define this function it is important to define first the Fisher information, *I(Θ)*, through expression (3.8),

$$I(\theta) = -E\left(\frac{\partial^2 LogL(\theta)}{\partial \theta^2}\right) \tag{3.8}$$

This is the negative expectation of the second derivative of the log-likelihood, *LogL(Θ),* and measures the curvature of the likelihood function. The flatter the likelihood is, the less information it provides about parameter values. Jeffrey's prior is then defined by expression (3.9).

$$p(\theta) \propto \sqrt{I(\theta)} \tag{3.9}$$

The Jeffrey's rule allows finding prior distributions which are invariant under reparametrizations. Additionally, in most situations, although these priors are improper, computed posterior distributions are proper.

When there is any information regarding the interest parameter, an informative prior may be used instead. This prior is not dominated by the likelihood and has an impact on posterior distribution. However, this prior should be specified with caution.

The property that the posterior distribution follows the same parametric form as the prior distribution is called conjugacy. Conjugate families are mathematical convenient as the posterior distribution follows a known parametric form. Obtained results are easy to understand and can be often represented in its analytical form. However, a more realistic prior may be used if there is information that contradicts the conjugate family [145].

3.4. Bayesian inference

The Bayesian inference procedure involves passing from a prior to a posterior distribution using the likelihood data function. Considering a Normal likelihood has the advantage of either conjugate or non-informative priors resulting in proper posteriors [33]. Within the Bayesian approach, interest parameters are assumed to follow certain probability distributions. Such distributions are defined by one or more unknown parameters. These parameters are also considered to have given distributions. They are further updated given the data and will be used to infer each interest parameter.

The simplest model is the consideration of mean as an unknown random variable with known deterministic variance. A more complex approach involves the consideration of both mean and variance as unknown. Accordingly, the following cases will be further analyzed: (1) Normal data with unknown mean (μ) and known variance (σ^2): Jeffrey's prior; (2) Normal data with unknown mean (μ) and known variance (σ^2): conjugate prior; (3) Normal data with unknown mean (μ) and variance (σ^2): Jeffrey's prior; (4) Normal data with unknown mean (μ) and variance (σ^2): conjugate prior.

3.4.1. Normal data with unknown mean (μ) and known variance (σ^2): Jeffrey's prior

The Jeffrey's prior for mean parameter is an improper uniform distribution over the real space [15], (3.10),

$$p(\mu) = c, -\infty < \mu < \infty \tag{3.10}$$

being c an arbitrary constant. Obtained posterior distribution is proper. Considering that data x follows a Normal distribution, and after dropping all constants, it can be given by [15], (3.11),

$$p(\mu \mid x) \propto \exp\left[-\frac{n}{2\cdot\sigma^2}\cdot(\mu-\bar{x})^2\right] \tag{3.11}$$

where n is the number of samples and $\overline{x}$ the data set mean value. The posterior distribution of the mean, given the data x, is thus a Normal with mean $\overline{x}$ (μ_1) and variance σ^2/n (σ_1^2). The posterior population follows a Normal distribution with mean μ_1 (μ_{pop}), and variance (σ_{pop}^2), given by (3.12),

$$\sigma_{pop}^2 = \left(\frac{n+1}{n}\right)\cdot\sigma^2 \tag{3.12}$$

3.4.2. Normal data with unknown mean (μ) and known variance (σ^2): conjugate prior

The conjugate prior for the mean, an informative prior, follows a Normal distribution with known initial mean (μ_0) and variance (σ_0^2) [15], (3.13),

$$p(\mu) \propto \exp\left[-\frac{1}{2\cdot\sigma_0^2}\cdot(\mu-\mu_0)^2\right] \tag{3.13}$$

Therefore, the posterior is also a Normal distribution with mean (μ_1) and variance (σ_1^2) computed according to prior and likelihood values [15], (3.14),

$$p(\mu) \propto \exp\left[-\frac{1}{2\cdot\sigma_1^2}\cdot(\mu-\mu_1)^2\right] \tag{3.14}$$

The updated mean is then analytically determined through (3.15),

$$\mu_1 = \frac{\frac{1}{\sigma_0^2} \cdot \mu_0 + \frac{n}{\sigma^2} \cdot \bar{x}}{\frac{1}{\sigma_0^2} + \frac{n}{\sigma^2}} \tag{3.15}$$

Precision is the inverse of variance. Hence, the posterior mean is the weighted average of prior and data set mean, with weights proportional to precision. The variance is computed according to (3.16),

$$\frac{1}{\sigma_1^2} = \frac{1}{\sigma_0^2} + \frac{n}{\sigma^2} \tag{3.16}$$

The posterior population follows a Normal distribution with mean μ_1 (μ_{pop}), and variance (σ_{pop}^2), given by (3.17),

$$\sigma_{pop}^2 = \frac{\frac{1}{\sigma_1^2} + \frac{1}{\sigma^2}}{\frac{1}{\sigma^2} \cdot \frac{1}{\sigma_1^2}} \tag{3.17}$$

3.4.3. Normal data with unknown mean (μ) and variance (σ^2): Jeffrey's prior

In this situation, the simplest joint prior is obtained by assuming that mean and variance can be estimated independently and that a vague prior distribution may be used for each. A common pair of vague priors for the Normal model is given by [15], (3.18),

$$p(\mu) \propto c, -\infty < \mu < \infty \tag{3.18}$$

and by (3.19),

$$p(\sigma^2) \propto \frac{1}{\sigma^2}, \sigma^2 > 0 \tag{3.19}$$

which is equivalent to the joint Jeffrey's improper prior for (μ,σ^2), (3.20),

$$p(\mu,\sigma^2) \propto \frac{1}{\sigma^2}, -\infty < \mu < \infty, \sigma^2 > 0 \tag{3.20}$$

In order to infer the unknown parameters it becomes necessary to derive the posterior distribution from Bayes theorem, given all observations *x*. The respective posterior is a proper distribution that takes the following form [15], (3.21),

$$p\left(\mu,\sigma^2 \mid x\right) \propto \left(\frac{1}{\sigma^2}\right)^{1/2} \cdot \exp\left[-\frac{1}{2}\cdot\left(\frac{\mu-\bar{x}}{\sigma/\sqrt{n}}\right)^2\right]\cdot\left(\frac{1}{\sigma^2}\right)^{\frac{(n-1)+1}{2}} \exp\left[-\frac{1}{2}\cdot\frac{S}{\sigma^2}\right] \tag{3.21}$$

being $S=\Sigma(x_i-\overline{x})$. The form of $p(\mu,\sigma^2|x)$ indicates that the conditional posterior distribution of μ conditional on σ^2 is a Normal distribution with mean $\overline{x}$ and variance σ^2/n, (3.22),

$$\mu \mid \sigma^2, X \rightarrow N\left(\bar{x},\frac{\sigma^2}{n}\right) \tag{3.22}$$

Being the marginal posterior distribution of $1/\sigma^2$ an inverse χ^2 distribution, (3.23),

$$\frac{(n-1)\cdot s^2}{\sigma^2} \rightarrow \chi^2_{n-1} \tag{3.23}$$

where $s=\Sigma(x_i-\overline{x})^2/(n-1)$ is the sample variance. As σ^2 appears in conditional distribution $\mu|\sigma^2$ (3.22) this means that μ and σ^2 are necessarily interdependent. In this case the main parameter distributions can be obtained by simulation, through the application of expressions (3.22) and (3.23), or by analytical solutions [15]. Therefore, the mean value of the distribution mean (μ_1) is equal to the sample mean ($\overline{x}$), being its variance ($\sigma(\mu_1)^2$) computed through (3.24),

$$\sigma\left(\mu_1\right)^2 = \frac{1}{n-3}\cdot s \; n>3 \tag{3.24}$$

The precision distribution mean ($1/\sigma_1^2$) is given by (3.25),

$$\frac{1}{\sigma_1^2} = \frac{n-1}{n\cdot s} \tag{3.25}$$

while its variance ($\sigma(1/\sigma_1^2)$) is given by (3.26),

$$\sigma\left(\frac{1}{\sigma_1^2}\right)^2 = \frac{2\cdot(n-1)}{n^2\cdot s^2} \tag{3.26}$$

The standard deviation distribution mean (σ_1) and variance ($\sigma(\sigma_1)^2$) is then computed through expressions (3.25) and (3.26) [15]. The posterior population mean is equal to μ_1 (μ_{pop}), while its variance (σ_{pop}^2) is obtained through (3.27),

$$\sigma_{pop}^2 = \frac{n+1}{n-3}\cdot s,\; n>3 \tag{3.27}$$

3.4.4. Normal data with unknown mean (μ) and variance (σ^2): conjugate prior

When both mean and variance are unknown, the natural conjugate prior has the form [15], (3.28),

$$p\left(\mu,\sigma^2\right) \propto \left(\frac{n_0}{\sigma^2}\right)^{1/2} \cdot \exp\left[-\frac{n_0}{2\cdot\sigma^2}\cdot\left(\mu-\mu_0\right)^2\right] \cdot \left(\frac{1}{\sigma^2}\right)^{\nu_0/2+1} \cdot \exp\left[-\frac{S_0}{2\cdot\sigma^2}\right] \tag{3.28}$$

where n_0 is the initial sample size and S_0 is the prior value of S. The prior is a Normal-Gamma distribution or, in other words, the product of an inverted Gamma distribution, with argument σ^2, and ν_0 ($\nu_0=n_0-1$) degrees of freedom, by a Normal distribution with argument μ, being the variance proportional to σ^2. The conditional prior distribution of μ conditional on σ^2 is thus a Normal with mean μ_0 and variance σ^2/n_0 ($\sigma(\mu_0)^2$), (3.29),

$$\mu \mid \sigma^2 \rightarrow N\left(\mu_0, \frac{\sigma^2}{n_0}\right) \tag{3.29}$$

Being the prior distribution of $1/\sigma^2$ a Gamma with parameters $\nu_0/2$ and $S_0/2$, (3.30),

$$\frac{1}{\sigma^2} \rightarrow gamma\left(\frac{\nu_0}{2}, \frac{S_0}{2}\right) \tag{3.30}$$

The conditional posterior distribution of μ conditional on σ^2 is a Normal distribution with mean μ_1 and variance σ^2/n_1 ($\sigma(\mu_1)^2$) [15], (3.31),

$$\mu \mid \sigma^2 \rightarrow N\left(\mu_1, \frac{\sigma^2}{n_1}\right) \tag{3.31}$$

Being the marginal posterior distribution of $1/\sigma^2$ a Gamma, (3.32),

$$\frac{1}{\sigma^2} \mid x \rightarrow gamma\left(\frac{\nu_1}{2}, \frac{S_1}{2}\right) \tag{3.32}$$

being, (3.33),

$$\nu_1 = \nu_0 + n \tag{3.33}$$

and, (3.34),

$$S_1 = S_0 + n\cdot s^2 + \frac{n_0\cdot n}{n_0+n}\cdot\left(\overline{x}-\mu_0\right)^2 \tag{3.34}$$

Accordingly, the posterior sum of squares (S_1) combines the prior (S_0) and the sample (s) sums, with the additional uncertainty given by the difference between the sample and the

prior mean. As σ^2 appears in conditional distribution $\mu|\sigma^2$ (3.31) this means that μ and σ^2 are necessarily interdependent. In this case, the main parameter distributions can be obtained by simulation, through the application of expressions (3.31) and (3.32), or by analytical solutions [15]. Thus, the posterior mean μ_1 is computed through a weighted average of prior (μ_0) and sample data ($\overline{x}$) mean values, with weights determined by the relative precision of two pieces of information, (3.35),

$$\mu_1 = \frac{n_0}{n_1} \cdot \mu_0 + \frac{n}{n_1} \cdot \bar{x} \tag{3.35}$$

being, (3.36),

$$n_1 = n_0 + n \tag{3.36}$$

The posterior distribution parameters combines thus prior information with the one contained in new data. The mean distribution variance ($\sigma(\mu_1)^2$) is given by (3.37),

$$\sigma\left(\mu_1\right)^2 = \frac{1}{\left(n+n_0\right)} \cdot \frac{S_1}{\left(\nu_1-2\right)},\ \nu_1 > 2 \tag{3.37}$$

The precision distribution mean ($1/\sigma_1^2$) is given by (3.38),

$$\frac{1}{\sigma_1^2} = \frac{\nu_1}{S_1} \tag{3.38}$$

while its variance ($\sigma(1/\sigma_1^2)$) is given by (3.39),

$$\sigma\left(\frac{1}{\sigma_1^2}\right)^2 = \frac{2 \cdot \nu_1}{S_1^2} \tag{3.39}$$

The standard deviation distribution mean (σ_1) and variance ($\sigma(\sigma_1)^2$) is then computed through expressions (3.38) and (3.39) [15]. The posterior population mean is equal to μ_1 (μ_{pop}), while its variance (σ_{pop}^2) is obtained through (3.40),

$$\sigma_{pop}^2 = \frac{\left(n+n_0+1\right)}{\left(n+n_0\right)} \cdot \frac{S_1}{\left(\nu_1-2\right)},\ \nu_1 > 2 \tag{3.40}$$

3.5. Posterior simulation

The main objective of Bayesian inference analysis is to obtain the posterior distribution. In several situations, it is enough to obtain point estimates that summarize the overall information (e.g. mean and variance parameters). Sometimes, this can be performed by

using analytical closed form solutions, especially if prior distributions are properly chosen. One other alternative is to infer it from simulated distributions.

There are several algorithms to simulate the posterior distribution. One of these algorithms is the Markov Chain Monte Carlo (MCMC), introduced by Metropolis *et al.* [124]. MCMC is a general method based on a sequential draw of sampled values with the distribution of sampled draw only dependent on the last value. It is thus a sequence of random variables or samples, $\theta_1,\ldots,\theta_n$, for which, at any instant t, the distribution θ_t is only dependent of θ_{t-1}. Hastings [76] presents a detailed description of this algorithm and Tierney [185] introduced several MCMC techniques for sampling posterior distributions.

One of those sampling procedures, implemented in WinBugs® [111], is the Gibbs sampler. Other technique is the Metropolis algorithm [185]. The Gibbs sampler is the most popular and is normally chosen for simulation in conditionally conjugate models. Gelfand *et al.* [62] presents the Gibbs sampler as a method for computing the Bayesian marginal posterior distribution. The Metropolis is generally used in models which are not conditionally conjugate. In this situation, the Gibbs sampler is recommended because standard forms are used for posterior distributions.

The following example is used to better explain this algorithm. Within this example, two parameters θ_1 and θ_2, with known conditional distributions $p(\theta_1|\theta_2)$ and $p(\theta_2|\theta_1)$, are considered. In this situation it is necessary to compute one or both marginal distributions, $p(\theta_1)$ or $p(\theta_2)$. This sampler starts with an initial value of θ_2^{t-1} and obtains θ_1^t from the conditional distribution $p(\theta_1|\theta_2=\theta_2^{t-1})$. Then the sampler uses θ_1^t to generate a new value θ_2^t computed from the conditional distribution $p(\theta_2|\theta_1=\theta_1^t)$. Accordingly, the samplers are taken from those two conditional distributions through the following sequence, by considering (3.41),

$$\theta_1^t \rightarrow p\left(\theta_1 \mid \theta_2 = \theta_2^{t-1}\right) \tag{3.41}$$

and (3.42),

$$\theta_2^t \rightarrow p\left(\theta_2 \mid \theta_1 = \theta_1^t\right) \tag{3.42}$$

This sequence is a Markov chain as the values from step t are only dependent of values from step $t-1$. If this sequence is long enough, the distribution of current values converges to the simulated distribution.

In the situation of the Normal model with conjugate prior for unknown mean and variance, a simulation procedure, based in MCMC and in Gibbs sampler, is recommended. Therefore, it becomes necessary to obtain draws from the marginal posterior distribution of precision,

expression (3.32), and then to simulate the mean value from the conditional posterior distribution of the mean given the variance, expression (3.31).

Accordingly, in this situation, the sampler starts with an initial value of v_0 and S_0, and obtains $1/\sigma^2$ from the marginal posterior distribution (3.43),

$$\frac{1}{\sigma^{2(0)}} \mid x \rightarrow gamma\left(\frac{v_0}{2}, \frac{S_0}{2}\right) \tag{3.43}$$

Then the sampler uses the $1/\sigma^2$ value to generate a new μ value, computed from the conditional posterior distribution of the mean, given the variance (3.44),

$$\mu^{(1)} \mid \sigma^2 \rightarrow N\left(\mu_0, \frac{\sigma^{2(0)}}{n_0}\right) \tag{3.44}$$

The samplers are then taken from those two conditional distributions through the following sequence (3.45),

$$\frac{1}{\sigma^{2(t-1)}} \mid x \rightarrow gamma\left(\frac{v_{t-1}}{2}, \frac{S_{t-1}}{2}\right) \tag{3.45}$$

and (3.46),

$$\mu^{(t)} \mid \sigma^2 \rightarrow N\left(\mu_{t-1}, \frac{\sigma^{2(t-1)}}{n_{t-1}}\right) \tag{3.46}$$

where v_{t-1}, S_{t-1}, μ_{t-1} and n_{t-1} are respectively computed through expressions (3.33), (3.34), (3.35) and (3.36). This process finishes when convergence is reached. At the end of it, a final point estimate for mean and variance is obtained.

3.6. An application of Bayesian inference framework

The developed Bayesian framework is applied to infer the compressive strength value (f_{lc}) of lightweight concrete material (LC 50/55) [48], used in a composite beam that was tested in laboratory up to failure. This parameter is then introduced in the developed probabilistic numerical model [118, 120, 121]. The first information for this parameter is gathered from bibliography [93, 188]. In this situation, complementary characterization tests (e.g. uniaxial compressive strength tests) were executed at laboratory [188], Figure 3.3. This data is used to update the respective parameter distribution.

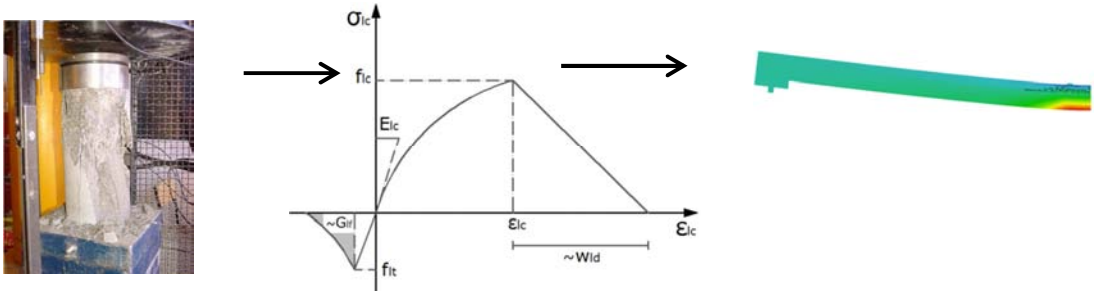


Figure 3.3. Compressive strength test and numerical model updating procedure.

The studied parameter is considered as a random variable. The initial distribution of the population is not known. Typically, in probabilistic approaches the concrete compressive strength follows a Normal or Lognormal distribution. Accordingly, both distributions were considered in order to evaluate its impact on final results. The Normal distribution presents some drawbacks as the possibility of considering negative parameter values which, in this particular situation, is physically impossible. However, this distribution is computationally convenient and good results are usually obtained with it.

When using a Lognormal distribution, three steps are considered [127]: (1) Proceed to the logarithmic transformation of data and compute the distribution parameters (mean and standard deviation); (2) Apply the Bayesian inference formula for Normal distribution, to update such parameters; (3) Transform the updated parameters (*Y*) into their equivalent ones on the Lognormal distribution (*X*) by using expression (3.47),

$$\mu_X = \exp\left(\mu_Y + \frac{\sigma_Y^2}{2}\right) \tag{3.47}$$

and(3.48),

$$\sigma_X^2 = \exp\left(2\mu_Y + \sigma_Y^2\right)\cdot\left(\exp\left(\sigma_Y^2\right) - 1\right) \tag{3.48}$$

Bayesian updating is developed in this example by considering several situations of initial information, uncertainty levels and degree of belief in initial assumptions. The scope of this study is to represent what is happening in reality. Therefore, two different uncertainty levels are considered: (i) unknown mean and known variance; (ii) unknown mean and variance. For each case, both non-informative (Jeffrey's prior) and informative (conjugate prior) priors were considered.

Accordingly, eight computations, outlined in Figure 3.4, were developed and compared, being chosen the one that provides the highest degree of belief. The posterior is obtained for all these analysis by simulation, through the use of WinBugs® [111]. In all these analysis, the

mean, standard deviation and 95% confidence interval of each posterior distribution will be computed [15].

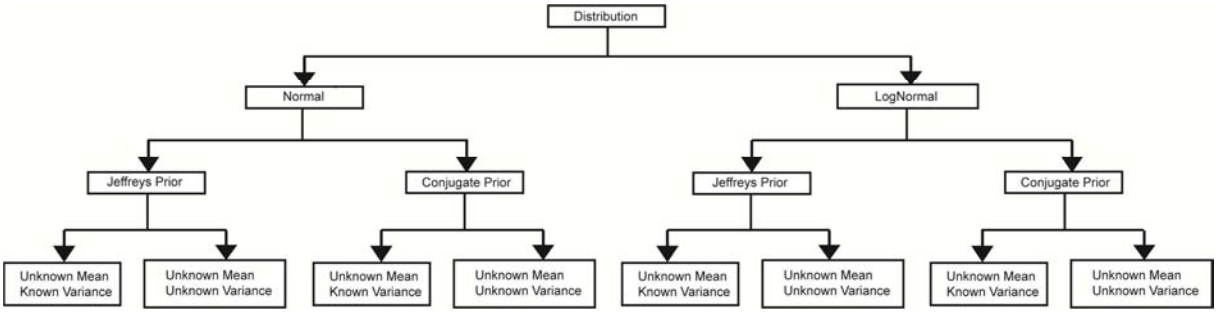


Figure 3.4. Scheme of performed computation for Bayesian inference.

3.6.1. Statistical analysis of data

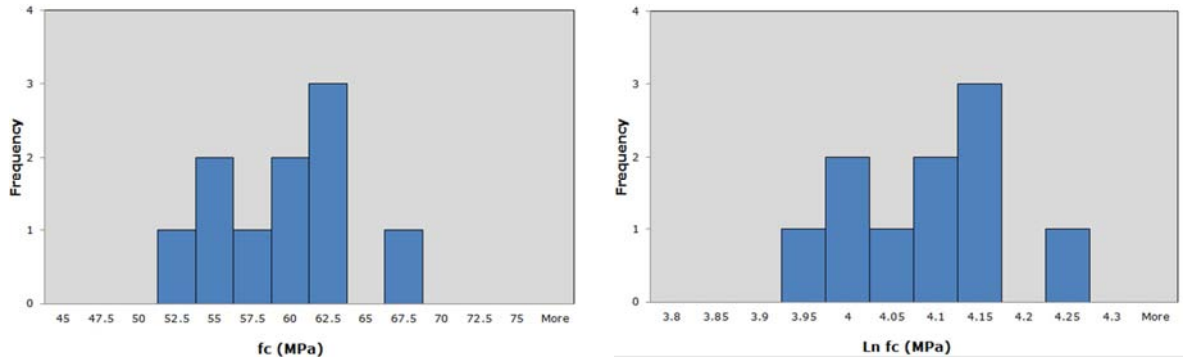
The conjugate prior distribution is defined from bibliography [93, 188]. According to JCSS [93], a Normal distribution with mean 58.00 MPa and standard deviation 5.80 MPa is recommended. A Lognormal distribution, with mean 4.056 MPa and standard deviation 0.101 MPa, is therefore adjusted to this distribution.

In this situation, experimental data is obtained from uniaxial compressive tests [188]. 10 tests were performed, resulting in the histogram presented at Figure 3.5a. Figure 3.5b indicates the histogram of the logarithmic transformation (to be used in the Lognormal case). The mean and standard deviation values for these two situations are indicated at Table 3.1. Obtained mean value from experimental tests [188] is close to the one from bibliography [93]. This validates the initial estimate.

Table 3.1. Distribution parameters for experimental data.

Parameter		Normal distribution	Lognormal distribution
μ	[MPa]	57.96	4.06
σ	[MPa]	4.64	0.08
σ^2	[MPa^2]	21.57	0.01

The histograms of Figure 3.5 present a Normal trend for the data. Accordingly, and in order to evaluate it, the Shapiro-Wilk normality test was performed to test this hypothesis [33]. This test shows that for a confidence level of 95% both distributions may be considered as normally distributed. This observation is important for the inference procedure.



a) b)

Figure 3.5. Obtained histograms for experimental data, considering: a) raw data; b) logarithmic transformation.

3.6.2. Normal data with unknown mean (μ) and known variance (σ^2)

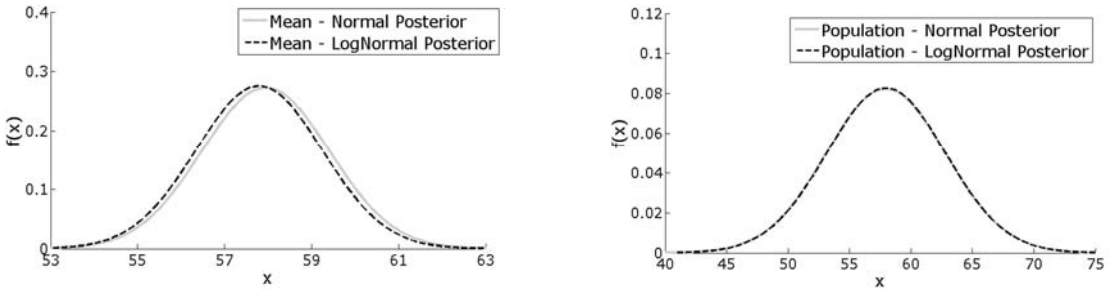
The first updating procedure corresponds to the situation of unknown mean and known variance, considering the Jeffrey's prior. Obtained results for the posterior distribution of the mean are closer for both Normal and Lognormal distribution. By analyzing Figure 3.6a it can be observed that the Lognormal distribution presents a slightly lower mean value than the Normal distribution. The population simulated values are also identical, as shown in Figure 3.6b. For both cases the simulated standard deviation value is lower than the initial one, which means that uncertainty is reduced with inference.

Table 3.2. Posterior estimates for the mean value, considering the Jeffrey's prior.

Parameter		Normal distribution	Lognormal distribution
μ_1	[MPa]	57.94	57.79
σ_1	[MPa]	1.46	1.45
95% CI for the mean	[MPa]	55.08 – 60.81	54.95 – 60.63
μ_{pop}	[MPa]	57.94	57.97
σ_{pop}	[MPa]	4.85	4.82
95% CI for the population mean	[MPa]	48.43 – 67.45	48.52 – 67.43

In order to use the conjugate informative prior, it was necessary to define a standard deviation for the initial mean. The adopted procedure, both for Normal and Lognormal distribution, was to use the obtained value from a non-informative prior (Jeffrey's prior). Table 3.3 provides the prior and posterior distributions. The posterior results are slightly different from those obtained using the Jeffrey's prior. In fact, a diminishment on the standard

deviation is verified. This point out the impact that the information provided by the used conjugate prior has in posterior results.



a) b)

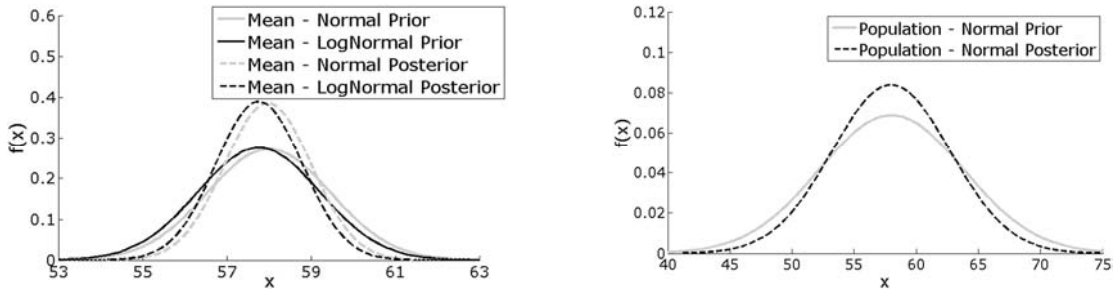
Figure 3.6. Posterior distribution, considering Jeffrey's prior, for: a) mean value of f_{lc}; b) simulated values of f_{lc}.

Table 3.3. Prior and posterior estimates for mean value, considering conjugate prior.

Parameter		Normal distribution	Lognormal distribution
μ_0	[MPa]	58.00	57.76
σ_0	[MPa]	1.46	1.45
μ_1	[MPa]	57.98	57.76
σ_1	[MPa]	1.03	1.02
95% CI for the mean	[MPa]	55.95 – 60.00	55.76 – 59.77
μ_{pop}	[MPa]	57.98	57.95
σ_{pop}	[MPa]	4.76	4.73
95% CI for the population mean	[MPa]	48.65 – 67.30	48.68 – 67.21

Figure 3.7a presents both prior and posterior distributions for the mean parameter. It can be observed the effect of experimental tests on obtained results. The uncertainty in mean value decreases since its standard deviation is reduced for both Normal and Lognormal case. Additionally, it can be observed that the Lognormal distribution presents a slightly lower mean value than the Normal distribution for both prior and posterior distributions.

Figure 3.7b shows a plot of prior and posterior population distribution for the Normal case. It is possible to conclude that updating the parameter mean presents a higher influence on simulated population. In fact, the posterior distribution is steeper than the prior one which means a reduction in the uncertainty value.



a) b)

Figure 3.7. Prior and posterior distribution, considering the conjugate prior, for: a) mean value of f_{lc}; b) simulated values of f_{lc}.

Figure 3.8 compares the posterior distributions for the population considering different priors and both Normal and Lognormal distribution. Practically, it is verified that all these approaches lead to similar posterior distributions.

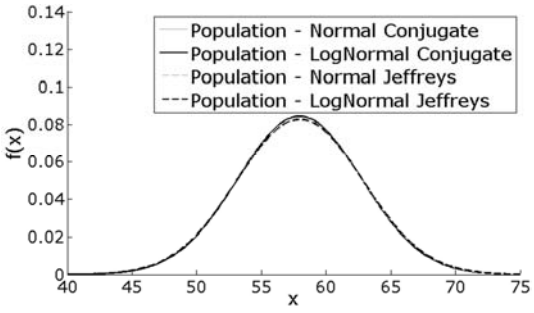


Figure 3.8. Posterior distribution for simulated values of f_{lc}, considering different prior distributions.

3.6.3. Normal data with unknown mean (μ) and variance (σ^2)

In this section, the parameter updating methodology, considering both mean and variance as unknown parameters, is presented. In a first step, a non-informative distribution, the Jeffrey's prior, is considered. The main results are presented in Table 3.4. Obtained results for the posterior distribution of the mean are identical for both Normal and Lognormal case. Simulated population mean values are closer to previous ones and they do not differ from Normal to Lognormal approach. This can be observed at Figure 3.9a.

The posterior values of the mean are closer to previous case, where variance is considered as a known parameter. In this situation the variance is only dependent from experimental tests, namely, from obtained variance and number of specimens. Therefore, due to the low number of specimens, obtained value is higher than the one obtained when variance is considered as a known parameter.

Table 3.4. Posterior estimate for mean value, considering the Jeffrey's prior.

Parameter		Normal distribution	Lognormal distribution
μ_1	[MPa]	57.97	57.83
$\sigma(\mu_1)$	[MPa]	1.74	1.76
σ_1	[MPa]	5.34	1.10
$\sigma(\sigma_1)$	[MPa]	1.44	0.03
95% CI for the mean	[MPa]	57.29 – 58.65	57.15 – 58.51
μ_{pop}	[MPa]	57.97	58.10
σ_{pop}	[MPa]	5.82	5.90
95% CI for the population mean	[MPa]	55.71 – 60.23	55.81 – 60.39

In the case of a conjugate prior, the initial distribution, obtained from bibliography [48, 188] is used to compute both mean and variance priors. In this case, the variance of the mean and of the variance is determined by applying first the inference considering a non-informative prior (Jeffrey's prior). The conditional posterior distribution for the mean and the marginal posterior for the variance are then obtained through the Bayes theorem. The main results are indicated at Table 3.5.

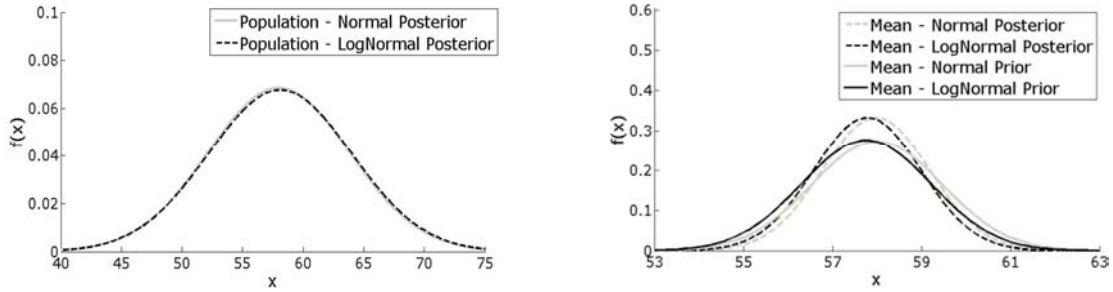
Table 3.5. Prior and posterior estimates for mean value, considering conjugate prior.

Parameter		Normal distribution	Lognormal distribution
μ_0	[MPa]	58.00	57.76
$\sigma(\mu_0)$	[MPa]	1.46	1.45
μ_1	[MPa]	57.98	57.78
$\sigma(\mu_1)$	[MPa]	1.20	1.20
σ_1	[MPa]	5.28	1.10
$\sigma(\sigma_1)$	[MPa]	1.07	0.02
95% CI for the mean	[MPa]	57.51 – 58.45	57.31 – 58.25
μ_{pop}	[MPa]	57.98	58.03
σ_{pop}	[MPa]	5.50	5.53
95% CI for the population mean	[MPa]	55.82 – 60.13	55.86 – 60.20

An insignificant variation on the mean value from prior to posterior estimates can be observed. In fact, the initial mean value is already close to the results provided by

experimental tests. This confirms that prior assumptions provide a good estimate of the analyzed parameter. Obtained distributions for the mean parameter are presented in Figure 3.9b. A clearly uncertainty reduction can be observed with inference, due to the fact that experimental tests provide lower standard deviation than prior distribution.

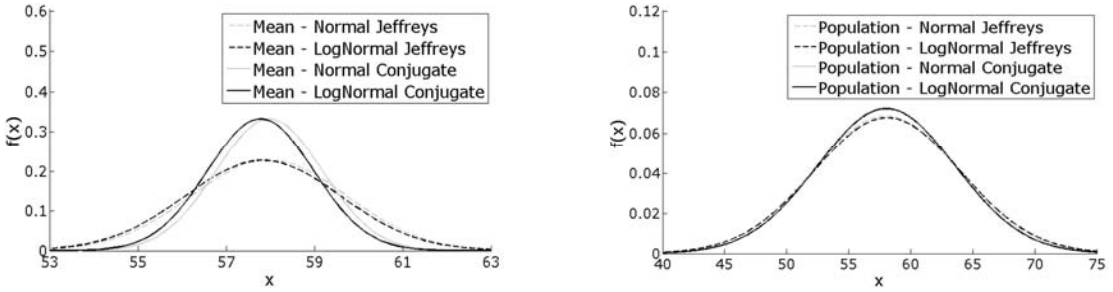
Using simulation, it is possible to infer the population parameter values. The updating procedure practically did not change the mean value. However, a reduction in uncertainty is verified due to a reduction in standard deviation value. This point out the impact that the information provided by the used conjugate prior has in posterior results.



a) b)

Figure 3.9. Obtained distributions for: a) simulated values of f_{lc}, considering Jeffrey's prior; b) mean value of f_{lc}, considering conjugate prior.

Figure 3.10a compares the posterior distribution for the mean value considering different priors. It is possible to observe that the conjugate prior provides a distribution with a lower uncertainty. Additionally, it is possible to observe that Normal distribution provides a higher mean value than the Lognormal distribution. Figure 3.10b compares the posterior distribution for the population, considering different priors. It is verified that the conjugate prior provides a distribution with a lower uncertainty.



a) b)

Figure 3.10. Posterior distribution, considering both priors, for: a) mean value of f_{lc}; b) simulated values of f_{lc}.

In this situation, while the number of experimental tests is known (10 values), the number of initial samples is adopted according to the degree of belief in initial assumptions. For instance, in the previous analysis 10 samples were considered in order to equilibrate the belief in prior distribution and in registered data from experimental tests. A study on the impact of different prior samples (n_0) in posterior distribution is thus developed. Accordingly, and for this situation, this value varies from 1 (no belief in prior model) to 10'000 (total belief in prior model). Obtained results are presented at Table 3.6.

Table 3.6. Posterior population distribution, considering different weights for initial assumptions.

n_0	μ_{pop}	σ_{pop}	95% CI for the population mean
	[MPa]	[MPa]	[MPa]
1	57.96	5.61	55.76 – 60.16
10	57.98	5.50	55.82 – 60.13
100	58.00	5.39	55.88 – 60.11
1000	58.00	5.37	55.89 – 60.10
10'000	58.00	5.37	55.89 – 60.10

Accordingly, as the number of prior samples increases, the posterior distribution mean converges from data sample to prior distribution mean value. At same time, the standard deviation decreases with this value. This is essentially due to the fact that as the prior sample increases, its weight in posterior distribution increases too and, at same time, the degree of belief on sampled data. Figure 3.11 shows these distributions.

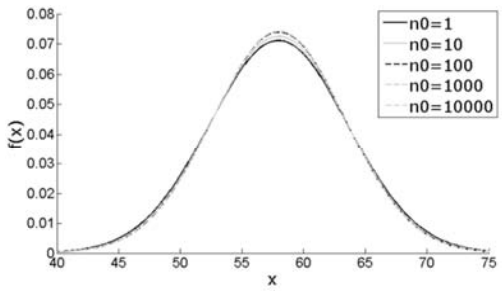


Figure 3.11. Simulated values of f_{lc} considering different weights for initial assumptions.

Bayesian methods present an inherent flexibility due to the incorporation of multiple uncertainty levels, the ability to combine information from different sources and the possibility of considering different degrees of belief in initial assumptions. The major advantage of this

framework is to deal, in a rational way, with uncertainty. Accordingly, random variables are updated through this methodology as new data is acquired.

The first approach considered the mean value as a random unknown variable and the variance known and deterministic. Good results were obtained for updating the mean parameter, providing the conjugate prior a higher reduction in uncertainty. In both situations, the population standard deviation values are lower than the initial ones. Obtained results also showed to be less sensitive to distribution type assumed for the data. The major drawback of this approach is related to the computation of each parameter characteristic value, necessary for design purposes. As the population variance is considered to be constant, the characteristic values are kept almost unchanged.

The second approach considers both mean and variance as unknown variables, even computationally costly, allows overcoming this problem. In this situation, the population variance is also updated. This approach allows a more global treatment of uncertainty. Obtained probabilistic distributions are almost identical for both Normal and Lognormal case and for both priors. An important note is that the mean value practically did not change with conjugate prior, which means that sampled data and prior assumptions are closer. However, the new data led to a significant decrease in parameter uncertainty.

Concluding, the unknown mean and known variance approach is simpler and can be used when the parameter of interest is the mean value. However, in structural parameters the prior information regarding the variance usually presents a high uncertainty and using it to define the deterministic variance may compromise the updating procedure. The more complex problem of considering both mean and variance as unknowns, allows overcoming this problem. It deals with uncertainty in a global way, reducing it in several dimensions. In both situations, posterior inferences showed stability to the choice of different priors.

The Bayesian framework provides a consistent way of treating data from different sources, in order to increase the reliability of structural parameters. In this situation, this approach revealed to be little sensitive to distribution type assumed for the data. This is essentially due to the fact of not being obtained any negative value for this parameter and thus there is no need to truncate any value when using the Lognormal distribution. However, for parameters in which the probability of obtaining a negative value is higher, Lognormal distributions are recommended.

3.7. Alternative updating methodology using Weibull distribution

Previous presented Bayesian framework showed good results in uncertainty treatment. However this approach might be sensitive to the distribution type assumed for the data. Therefore, Miranda [127] proposed a new methodology to overcome this problem. In this methodology, a two-parameter Weibull distribution is used to model both prior and likelihood. This distribution is, in fact, more flexible and easily adaptable to available data transforming it into a Normal or any other distribution according to its configuration.

Since Weibull data is not conjugate with prior, there is no closed form solution to the problem as there is for Normal distribution [66, 94]. Thus, to avoid heavy computations which would make the method difficult to implement in a practical sense, some acceptable simplifications are considered. Accordingly, a heuristic approximation to produce a fast method of estimating Weibull parameters assuming that they are normally distributed is proposed by Miranda [127].

3.7.1. The Weibull distribution

The Weibull distribution is one of the most widely distributions used in reliability engineering. The two-parameter Weibull distribution is characterized by a probability density function given by (3.49),

$$f(x) = \frac{\beta}{\alpha}\left(\frac{x}{\alpha}\right)^{\beta-1} \exp\left(-\frac{x}{\alpha}\right)^{\beta};\ x \geq 0 \tag{3.49}$$

where β>0 is the shape parameter and α>0 is the scale parameter. The cumulative distribution function is given by (3.50),

$$F(x) = 1 - \exp\left(-\frac{x}{\alpha}\right)^{\beta} \tag{3.50}$$

The reliability of a distribution is one minus the cumulative distribution function, (3.51),

$$R(x) = \exp\left(-\frac{x}{\alpha}\right)^{\beta} \tag{3.51}$$

To obtain the value (x) corresponding to a certain reliability degree, expression (3.52) can be used,

$$x = \alpha\left\{-\ln\left[R(x)\right]\right\}^{1/\beta} \tag{3.52}$$

$R(x)$ = 0.50 corresponds to the median value. The mean (μ) of a Weibull random variable is given by (3.53),

$$\mu = \alpha \cdot \Gamma\left(\frac{1}{\beta} + 1\right) \tag{3.53}$$

and the standard deviation (σ) is given by (3.54),

$$\sigma = \alpha \sqrt{\Gamma\left(\frac{2}{\beta} + 1\right) - \Gamma\left(\frac{1}{\beta} + 1\right)^2} \tag{3.54}$$

in which Γ is a Gamma distribution defined by (3.55),

$$\Gamma(n) = \int_0^\infty e^{-x} x^{n-1} dx \tag{3.55}$$

This distribution is very versatile and flexible since it adapts to the data and can easily mimic the behavior of other distribution types, based on the shape parameter value [33]. There are many methods to estimate Weibull parameters from a specific data set. One of the most appropriate methods is to perform the curve fitting to the histogram of data.

A measure of good fitness of a statistical model is the AIC (Akaike Information Criteria) [4], defined by (3.56),

$$AIC = 2k - 2\ln(L) \tag{3.56}$$

where k is the number of parameters in the statistical model and L is the maximized value of the likelihood function for the estimated model. Given a set of models, the most appropriate is the one which presents the lowest AIC value.

3.7.2. The proposed methodology

In the methodology proposed by Miranda [127], it is assumed that a prior distribution is already known. Accordingly, both prior and new data are modeled with Weibull distributions to take advantage of its flexibility. However, this may be very complex as prior parameters are not conjugate with data and thus there is no analytical formula for the posterior probability distribution of parameters [66, 94].

To overcome this problem a simple heuristic, in which the Weibull distribution parameters are considered to be random variables that follow a multivariate Normal distribution and can be analytically updated as such, is proposed. The main disadvantages of this method are that it does not use formal Bayesian procedures and assumes that parameter estimates are normally distributed [127].

Brennan and Kharroubi [20] considered identical simplifications in a similar Weibull approach. The accuracy of the method is context dependent. In fact, if the prior is weak (e.g.

based on little knowledge) and data is strong (e.g. large sample size), or in inverse case, the result can be almost equivalent to formal Bayesian updating, but, between these extremes, this simplification can lead to a reduced accuracy.

The methodology starts with the computation of Weibull parameters from prior data. Accordingly, it is assumed that uncertainty in these parameters can be correctly characterized by a multivariate Normal distribution (Prior$\rightarrow N(\mu_0,\sigma_0)$). The standard deviation value was considered to be related to the 95% confidence interval of the Weibull analysis regression parameters. Therefore, it is computed as the distance between the mean and the upper or lower bound of the confidence interval.

The same procedure is then applied to the data, considering a multivariate Normal distribution to characterize the parameters uncertainty (Data$\rightarrow N(\mu,\sigma)$). The Bayesian updating formula for multivariate Normal distribution is then used to compute the posterior parameter estimates, (3.57),

$$\mu_1 = \frac{\frac{1}{\sigma_0^2}\cdot\mu_0 + \frac{1}{\sigma^2}\cdot\mu}{\frac{1}{\sigma_0^2}+\frac{1}{\sigma^2}} \tag{3.57}$$

and, (3.58),

$$\frac{1}{\sigma_1^2} = \frac{1}{\sigma_0^2} + \frac{1}{\sigma^2} \tag{3.58}$$

A simulation procedure is then used to obtain the population distribution. Accordingly, 10’000 samples of Weibull distribution parameters are generated from their Normal distributions. These values are then reused to generate Weibull random values which are again fitted to a Weibull distribution, considered to be the population distribution. With this distribution it is possible to obtain the respective parameters. Figure 3.12 presents a scheme of the proposed methodology.

3.7.3. Obtained results

This methodology is applied with the data samples obtained from experimental tests. Accordingly, a Weibull analysis is applied to initial data and to likelihood (LLH). The posterior is then computed through the application of a Bayesian framework. Table 3.7 indicates the obtained distribution parameter values. Additionally, it also provides the AIC values from curve fitting to such data.

The population values are then generated through simulation. The parameters of the Weibull fit to simulated data are indicated at Table 3.8. The AIC values indicate that the fitting procedure is excellent for all distributions. Obtained inference values for the posterior are closer to both prior and likelihood, as expected.

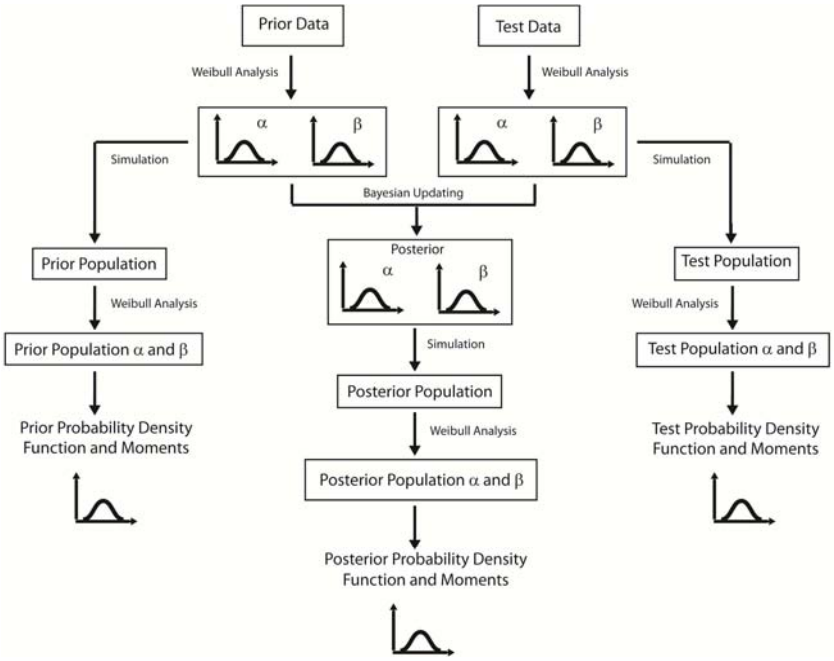


Figure 3.12. Bayesian updating scheme using Weibull distribution, adapted from Miranda [127] (random distributions are merely indicative).

Table 3.7. Mean and standard deviation of Weibull distribution parameters.

-	*α*		*β*		AIC
	Mean	Standard deviation	Mean	Standard deviation	
	[MPa]	[MPa]	[MPa]	[MPa]	[-]
Prior	60.62	0.01	10.61	0.01	$6.43 * 10^{6}$
LLH	60.16	3.11	12.74	5.98	$6.45 * 10^{1}$
Posterior	60.70	1.55	10.40	1.51	-

Figure 3.13 presents the probability density functions for simulated prior, LLH and posterior populations, according to Table 3.8. A clear uncertainty reduction can be observed with inference. Additionally, a slightly increment of the population mean value is verified.

The Weibull distribution allows using reliability concepts in parameter computations. Accordingly, it is possible to compute values with certain reliability levels for simulated distributions. The reliability of a certain value is interpreted as the probability that the

parameter true value has of being higher than it. Table 3.9 presents the obtained values for prior and posterior distributions considering different reliabilities.

Table 3.8. Weibull fit parameters for simulated population values.

-	α	β	AIC
	[MPa]	[MPa]	[-]
Prior	60.63	10.61	$6.53 * 10^6$
LLH	60.16	12.73	$6.17 * 10^6$
Posterior	61.89	11.09	$5.89 * 10^6$

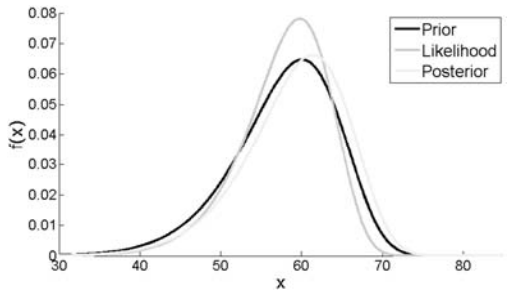


Figure 3.13. Weibull distributions for simulated values of f_{lc}.

Table 3.9. Values of parameter f_{lc} for different reliability levels.

Reliability	f_{lc}	
	[MPa]	
	Prior	Posterior
0.010	39.28	40.86
0.025	42.86	44.20
0.050	45.81	46.93
0.100	49.02	49.89
0.500	58.56	58.56
0.900	65.57	64.85
0.950	67.22	66.31
0.975	68.55	67.50
0.990	70.00	68.78

This alternative methodology tries to overcome the sensitivity of chosen distribution, through the use of a more flexible distribution that can easily adapt to available data. In this case the

two-parameter Weibull distribution is used to avoid more rigid Normal and Lognormal distributions. As it is not possible to find a conjugate prior for this distribution, some simplifications are considered. The methodology is based in a heuristic procedure, in which the model parameters are random variables that follow a Normal distribution.

This methodology should be applied when there is little information regarding the most recommended distribution. However, it is suggested to process it with all parameters in order to evaluate if it provides a higher degree of belief than the other models. Hence, the model that provides the lowest uncertainty value should be chosen.

3.8. Conclusions

This chapter presents the Bayesian inference framework that is used to updated structural parameters with acquired experimental data. It begins with a brief presentation of the Bayes theorem and the relation between prior, likelihood and posterior probabilistic distributions. A detailed description of prior distributions is also provided. The choice of the prior distribution is discussed, as well as the advantages of using an informative prior.

The Bayesian inference procedure is then clarified. Therefore, eight different inference situations are studied. These cases vary in considered probabilistic distribution for data, Normal or Lognormal, in the uncertain distribution parameters, mean (μ) and variance (σ^2) or both, and in the chosen prior, Jeffrey's or conjugate. For each case, it is indicated how the posterior distribution is obtained. Sometimes, it is necessary to use specific simulation algorithms to obtain such distributions. One of those algorithms is the Markov Chain Monte Carlo (MCMC), which is also described.

These eight cases are then tested with a real example of the inference applied on the compressive strength of lightweight concrete (f_{lc}). Accordingly, the statistical analysis of data obtained in uniaxial compressive tests is first indicated. Some additional studies are made in order to evaluate the impact of different levels of initial information, uncertainty levels and degree of belief in initial assumptions.

An alternative updating methodology for Weibull distributions, initially proposed by Miranda [127], is also provided. This tool is interesting, as this distribution type is very versatile and can be adopted for different numerical input parameters. Its formulation is presented, together with an application on the compressive strength (f_{lc}) of lightweight concrete that corresponds to the same example presented before. The reliability concept is also introduced.

4 Probabilistic Assessment of Structures

4.1. Introduction

The assessment of structures comprises all necessary activities to evaluate their condition for future use. A general flowchart of this procedure is provided in ISO 13822 [86]. A review of the philosophy, theoretical concepts and tools necessary to carry out this procedure is provided by Faber [52]. During structural assessment, several sources of uncertainty may be identified. In order to consider them, reliability algorithms are commonly used. It was verified that the use of these algorithms on the assessment of structures resulted in substantial cost savings [101].

Safety assessment is based on the established relation between the assessed structure resistance and the applied load. A reliability index is obtained from the probabilistic-based safety assessment [123, 134]. This index characterizes the failure probability of the assessed structure, through the comparison of both resistance and loading probability density functions (PDF). This index represents a more objective mean to assess the structural performance [59]. Several authors have been using probabilistic-based safety assessment procedures within the last few decades [41, 77, 113, 197].

Some authors proposed a probabilistic-based safety assessment method that considers both resistance and loading PDF as time-variant quantities [5, 40, 43, 130]. Recently, Bayesian inference was introduced to update the resistance PDF with collected data from real

structure [14, 80, 133]. Strauss *et al.* [175] proposed a methodology for continuous safety assessment that incorporates data from permanent monitoring systems, through a Bayesian inference algorithm.

The use of nonlinear analysis software in structural assessment procedures, although computational costly, enables a realistic estimation of the structural response both in service and failure region. Bergmeister *et al.* [14] introduced a probabilistic-based safety assessment concept for concrete structures that integrates nonlinear finite element analysis software with reliability algorithms.

Some authors used model identification techniques for parameter estimation of structures. A review of these procedures was provided in chapter two. However, the combination of these techniques with a probabilistic-based safety assessment framework is still scarce. More recently, Novák *et al.* [133] and Strauss *et al.* [174, 176, 177] developed a complex methodology for assessment of structures which combines structural analysis and reliability algorithms with new modules for model identification.

A multilayered probabilistic-based structural assessment framework [118, 119, 121], composed by both model identification and reliability algorithms, is described in this chapter. This methodology aims to incorporate all the uncertainty sources inherent to a structural assessment procedure. This chapter begins so with a summary of these uncertainty sources. Then the developed framework is presented. It is given an emphasis to sensitivity analysis as a measure to reduce computational costs.

A description of implemented model identification technique is further presented. Within this procedure, model parameters are adjusted in a way that obtained numerical results best fit the experimental data. A convergence criterion, based in a combination of measurement and modeling errors, is therefore established for incorporated optimization algorithms. An engineering judgment procedure is finally introduced to select the best model from a pool of optimal models.

The probabilistic model is obtained by considering randomness in identified model parameters. A Bayesian inference algorithm, described in chapter three, is used to update each model parameter PDF with acquired data. Obtained results, from probabilistic analysis, are statistically processed. An updated resistance PDF is thus obtained. Structural performance is evaluated through specific indexes. Some of these indexes are especially important as they give an indication of the structure safety. At the end, some considerations are made from the proposed methodology.

4.2. Uncertainty sources

Uncertainty is always present in structural assessment procedures. Accordingly, and from a pragmatic standpoint, it is useful to categorize it within this process in order to clarify which uncertainties have the potential of being reduced. Kiureghian and Ditlevsen [97] introduced categorization in probabilistic assessment of structures. It is shown that, for proper formulation of reliability, careful attention should be paid to the categorization of uncertainties. Failing to do so may result in underestimation or overestimation of failure probability. Uncertainties are generally characterized as epistemic, if the modeler sees a possibility to reduce them by gathering more data or by refining models, and as aleatoric, if the modeler does not foresee the possibility of reducing them.

A more detailed classification procedure consists on the following [77]: (1) modeling errors, that result from theoretical approximations from the real behavior of materials and from simplifications considered in loading and on its effects; (2) physic uncertainty, due to randomness in structural parameters (material, geometry and physic sources) and loading; (3) statistic uncertainty, due to the uncertainty in estimating the PDF parameters through a finite number of proofs; (4) human errors, which results from human involvement during the whole procedure. The measurement errors, due to transducer precision, cable losses, and data acquisition and transmission equipment's, is considered by some authors as an uncertainty source too [87].

4.3. Safety assessment levels

Different levels of structural safety assessment can be defined according to the way the uncertainty sources are processed. The following division is established by different authors [77, 87, 196]: (1) level 0, purely deterministic analysis where all parameters present deterministic values and a global safety factor is used; (2) level 1, semi-probabilistic method as the randomness in resistance and loading parameters is considered through representative values (nominal and characteristic) and partial safety factors. The characteristic values are defined through the specified mean value, coefficient of variation (CV) and PDF; (3) level 2, probabilistic method as a PDF is established for each model parameter. Such PDF is defined by statistic measures that describe the central tendency (e.g. mean value) and dispersion (e.g. variance). A failure probability (reliability index) is therefore computed, considering these PDF and a correlation matrix; (4) level 3, full probabilistic method that considers the joint PDF of all parameters for computing the failure probability (reliability index).

Other structural safety assessment classifications are also presented in literature. Ryall [160] introduced several safety assessment levels. Some of these levels use reliability algorithms and others combine numerical and measured data. The concept of rating factor, which relates resistance and loading, is introduced. A more complex factor that combines the assessed and design reliability index is also present. Wisniewski *et al.* [197] suggests the use of simplified probabilistic-based safety assessment methods. Accordingly, the use of level 2 assessment techniques increased in the last years [14, 41, 43, 197]. Maljaars *et al.* [113] present a variant of this level, which incorporates additional measurements at critical parts of the structure for determining its safety.

4.4. Probabilistic assessment

The proposed probabilistic assessment methodology [118, 119, 121] is also a variant of level 2 assessment technique, as it combines a simplified probabilistic-based approach with measured data. The main result of this methodology is an updated resistance PDF of the assessed structure. This updating procedure is executed within the structure lifetime, as data is collected from the structure. Once the updated resistance PDF is obtained, the safety assessment can be developed through a comparison with loading PDF. This methodology is divided into the following steps, indicated in Figure 4.1.: (1) step 1: model identification; (2) step 2: probabilistic analysis.

The main objective of model identification is to obtain the most likely values of model parameters. Within this procedure the numerical results are fitted to collected data from real structure, by adjusting the model parameter values. Obtained results are then compared to measured data and the distance between these two values (or fitness function) is minimized with an optimization algorithm. The main result of this procedure is an updated deterministic numerical model. In order to compute the numerical model the nonlinear structural analysis software ATENA® [24] is used. A Matlab® algorithm [25, 129] is developed to: (1) automatic generate and compute the numerical models; (2) compute the distance between numerical and experimental data; (3) optimize this distance. This procedure presents a high computational cost. It is usually developed when the uncertainty about the evaluated structure is high and an in-depth assessment becomes necessary. Figure 4.2a indicates a flowchart of this algorithm.

The main objective of probabilistic analysis is to include uncertainty in numerical model. Therefore, randomness is considered in model parameters through a PDF. Each prior PDF is then updated with complementary data (or likelihood) through a Bayesian inference procedure. The posterior PDF is therefore computed and an updated probabilistic numerical

model is obtained. The nonlinear structural analysis software ATENA® [24] is used to compute the numerical model. A Matlab® algorithm [25, 129] is developed to: (1) random generation of model parameter values based on Iman and Conover algorithm [85, 134]; (2) automatic compute the pool of generated numerical models; (3) process the obtained results through a statistical analysis. The Bayesian inference is performed with WinBugs® [111]. In this case, a Matlab® algorithm [25, 129] is developed to: (1) generate the prior PDF; (2) compute the posterior PDF. This procedure presents a low computational cost. Figure 4.2b gives a flowchart of this algorithm.

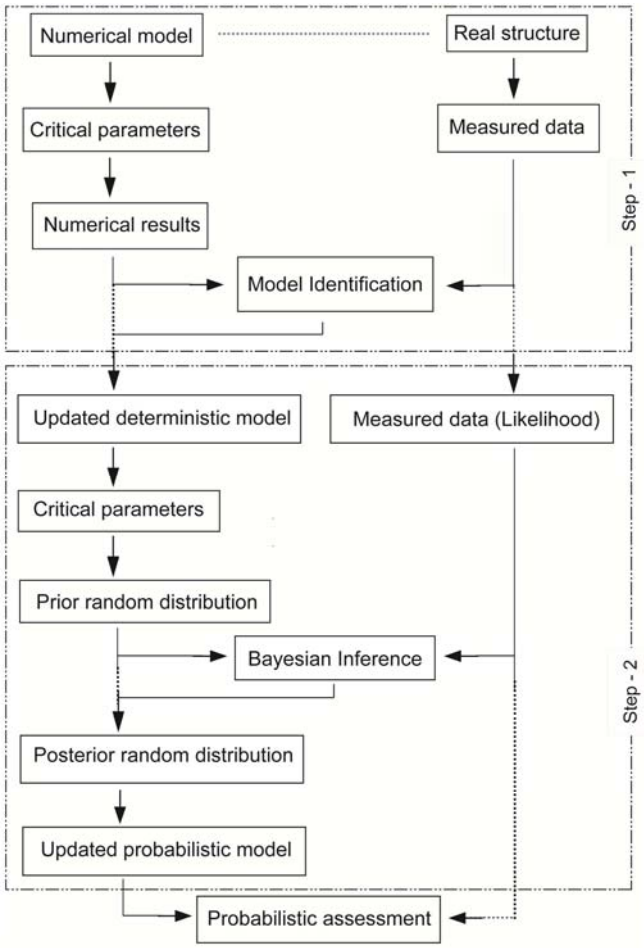
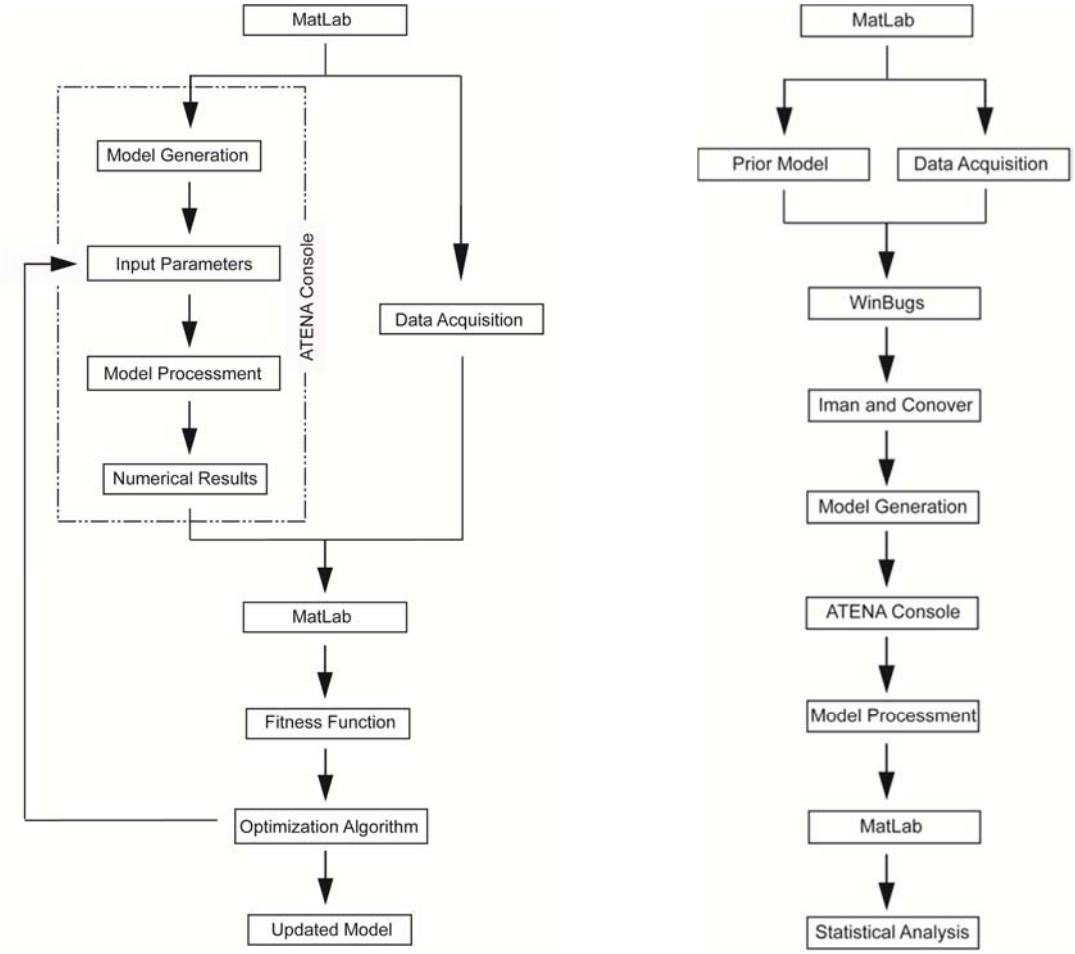


Figure 4.1. Probabilistic assessment algorithm [118, 119, 121].

The developed probabilistic assessment methodology considers all previous identified uncertainty sources. These uncertainties are: (1) measurement and modeling errors, that are introduced when defining the model identification convergence criterion; (2) physic uncertainty, that is incorporated when considering randomness in model parameters (material, geometric and physical); (3) statistical uncertainty, that is reduced with Bayesian inference as complementary data is obtained; (4) human errors, that are usually identified with model identification.



a) b)

Figure 4.2. Implemented algorithm: a) model identification; b) probabilistic analysis.

4.5. Sensitivity analysis

The main drawback of the proposed methodology is its computational cost. This cost is proportional to the number of input parameters. Accordingly, before applying the respective methodology, it is recommended to perform a sensitivity analysis with the developed numerical model.

The main objective of this analysis is to obtain the critical parameters. Critical parameters are those that present a higher influence on the overall structural response. In each sensitivity analysis it is possible to evaluate the influence of material (e.g. concrete strength), geometric (e.g. section dimensions) and physic (e.g. spring support stiffness) properties of analyzed structure. This analysis can be performed either in service or failure region. In fact, parameters that present a higher influence in service may not present an identical impact in failure.

Accordingly, the following procedure is recommended: (1) to develop the deterministic numerical model of analyzed structure using mean values for input parameters; (2) to divide the structural parameters by category (material, geometric, physic) and subcategory (concrete, steel, interface, support, …); (3) for each parameter, to determine the most suitable CV; (4) to vary each parameter by adding or subtracting a standard deviation value, keeping all the other parameters fixed; (5) for each set of parameter values, to proceed to the numerical analysis with a nonlinear structural analysis software; (6) to apply the following expression (4.1) [33],

$$b_k = \sum_{i=1}^{n} (\Delta y_k / y_m) / (\Delta x_{ik} / x_{im}) \cdot CV \ [\%] \tag{4.1}$$

being b_k the importance measure of parameter k, Δy_k the variation in structural response due to a deviation of Δx_k in relation to the parameter mean value x_m, y_m the average response and n the number of generated parameters; (7) to determine the maximum importance measure value; (8) to normalize all values in relation to the maximum; (9) to plot the standardized importance measures in a bar plot; (10) to define a limit for the importance measure (b_{lim}). The critical parameters are those that present an importance measure higher than this value.

4.6. Model identification

In a first step, and according to Figure 4.1, the proposed probabilistic assessment methodology performs the model identification in order to obtain an updated deterministic numerical model. During this procedure, model parameters are assessed through an automatic adjustment procedure to measured data. The set of procedures necessary to determine an updated model are designated by model updating. The organization chart of the developed algorithm is indicated at Figure 4.3.

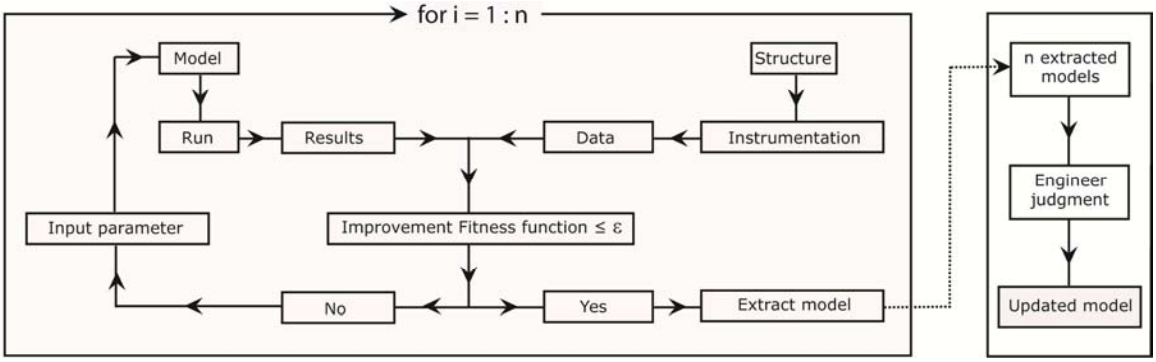


Figure 4.3. Model identification procedure.

The algorithm runs the nonlinear numerical model, for each set of input parameters obtaining, from each run, the respective numerical results. At same time, experimental data is collected by specific instrumentation, from real structure. A comparison that consists in determining the cumulative difference between numerical and experimental data is performed for each monitoring point (Figure 4.4a). The objective function value, to be minimized, is then obtained. When there is a population of models, this value is computed for all models. However, for a matter of convergence criterion the considered value is the minimum.

Accordingly, the basis of this methodology is an objective function, designated by fitness function (*f*), that describes the approximation between numerical results and experimental data, (4.2),

$$f = \sum_{i=1}^{n} \left| y_i^{num} - y_i^{\exp} \right| \Big/ \max\left(y^{\exp}\right) \cdot 1/n \ [\%] \tag{4.2}$$

in which y^{num} and y^{exp} are the numerical and experimental values and n is the number of evaluated points. Therefore, for each analyzed point, the ratio between the absolute difference of numerical and experimental results and obtained maximum measured value is computed. The sum of these values is then divided by the number of analyzed points.

It is important to note that this function is normalized and so, it can be used with different transducers, placed in different regions of the structure, measuring different parameters. Additionally, it can be also applied when more than a structure is tested, in identical conditions, and/or when more than a load case is implemented. In this situation, the fitness function is obtained by using the following expression (4.3),

$$f = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{o} \sum_{l=1}^{p} f_{i,j,k,l}^2} \tag{4.3}$$

being $f_{i,j,k,l}$ the fitness function value computed from expression (4.2) for each measurement *i* in each position *j* at tested structure *k* for load case *l*. According to Figure 4.3, an established stop criteria for model identification algorithm is the fitness function convergence criterion, given by expression (4.4),

$$\Delta f = \left| f_{i+n} - f_i \right| \leq \varepsilon \tag{4.4}$$

being *f* the fitness function for generation *i* and *i* + *n*, and *n* the defined gap between two generations. If the difference between these two values is less than or equal to a pre-

specified threshold value (ε), than the algorithm stops and the population of models is extracted. However, if this criterion is not accomplished, the developed algorithm will rearrange the input parameters set and the procedure restart. The extracted models are then analyzed by an engineering judgment procedure, based in the probability of occurrence of each model. The most likely model is the updated model.

When using a multi-parameter model identification methodology, there are two sources of errors, related to experimental measurements and to numerical analysis, which should be considered during the analysis. Consequently, when computing the difference between numerical and experimental data, according to fitness function (4.2), used points should not be considered as deterministic but as an interval of values, all of them with the same probability (uniform PDF). This is exemplified in Figure 4.4b. Therefore, the threshold value (ε) is a combination of both measurement and modeling errors.

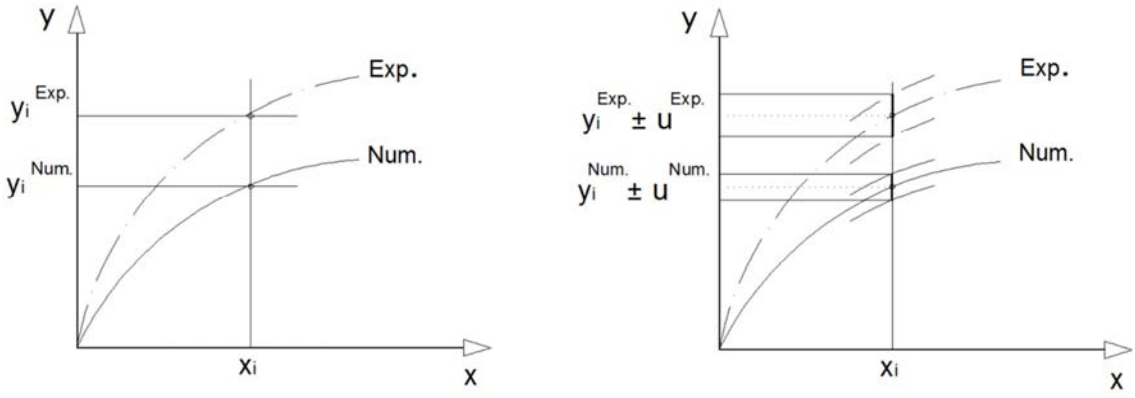


Figure 4.4. Fitness function: a) definition; b) convergence criterion.

A similar procedure is presented by Goulet *et al.* [67, 69]. For these authors, the fitness function is defined as the root-mean-square of the difference between measured values and model predictions. When this value is less than a certain threshold value, the model is classified as a candidate model. The threshold is evaluated by assuming reasonable values for modeling and measurement errors. Accordingly, a model involving the right set of assumptions and correct parameter values has a fitness function value that is less than or equal to this threshold.

These authors introduce the concept of reliability in model identification. The identification presents 100% reliability when the following three conditions are met: (1) all possible models are considered in the set of models; (2) there are sufficient measurement data to filter out wrong models; and (3) all errors are zero. Although, it is almost impossible to completely fulfill these three conditions, it is possible to assume that the first two conditions are

commonly met. In fact, for many structures the use of a stochastic algorithm is enough to generate all possible models. Additionally, enough measurement data is available to filter out wrong models.

During model identification procedure, human errors (e.g. due to bad concreting) are usually detected. This procedure is commonly used in: (1) assessment of existent structures in order to determine if those structures present any additional carrying capacity; (2) in forensic engineering, when it is necessary to determine any human error developed during construction (e.g. an inferior concrete cover). Therefore, model identification permits to identify this source of errors.

Within the model identification procedure, some Matlab® routines were developed, as indicated at Figure 4.2a [25, 129]. These routines are provided in Appendix A, respectively:

(1) b*ackanalysis.m*: run the model identification procedure. It defines the optimization algorithm parameters (e.g. threshold value of the convergence criterion) and also the lower and upper bound limits of each variable;

(2) *fobj.m*: compute the fitness function of the model identification procedure;

(3) *es.m* [29]: evolutionary strategies algorithm, used to compute the best model from the population of models. This algorithm was modified in order to give all the population of models, necessary for the engineering judgment procedure;

(4) *run_num.m*: run the structural analysis software;

(5) *parameter.m*: define each model parameter value;

(6) *data.m*: generate the data file which will be read and then processed in structural analysis software;

(7) *processment.m*: process the generated data file and convert the obtained results into Matlab® [25, 129];

(8) *matrix_num.m*: convert the obtained results from the structural analysis software into the same unities of experimental data, necessary to compute the fitness function value;

(9) *test_num.m*: divide the numerical results into specific steps in order to guarantee that the same points of experimental data are compared in the fitness function (4.2);

(10) *run_exp.m*: read experimental data and convert it into Matlab® [25, 129];

(11) *matrix_exp.m*: convert the experimental data into the same unities of numerical results, necessary to compute the fitness function value;

(12) *test_exp.m*: divide the experimental data into specific steps in order to guarantee that the same points of numerical results are compared in the fitness function (4.2);

(13) *constraint_value.m*: introduce, when necessary, specific constraint values;

(14) *limit.m*: determine limits for service and failure region evaluation.

Additionally, the nonlinear structural analysis software ATENA® [24] is used in this situation. The following executable files are thus incorporated: (1) *atenaconsole.exe*: run the developed numerical model using the finite element methodology; (2) *gawk.exe* [3]: convert obtained results from ATENA® into readable data for Matlab® processing.

4.6.1. Optimization algorithm

In order to minimize the distance between measured and numerical data, given by the fitness function (4.2), it will be necessary to use an optimization algorithm. There are several algorithms that can be used. After a detailed study on some of those algorithms, presented in previous chapter two, it was decided that the most reasonable algorithm is the evolutionary strategies in its plus version. This algorithm permit to identify a population of models once the convergence criterion is achieved. A Matlab® version of this algorithm already exists and was adapted for this methodology [29]. The respective algorithm (*es.m*) is presented in Appendix A of this thesis.

4.6.2. Errors

Errors play a major role in model identification. They may arise from many sources, being the most significant the measurement and modeling errors. Measurement errors can result from equipment as well as on-site installation faults. Some authors like Banan *et al.* [11, 12] studied the effect of measurement errors by using simulation algorithms.

Sanayei *et al.* [163] studied the effect of measurement and modelling errors in parameter estimation. For these authors measurement error typically has zero mean and its magnitude depends on experimental equipment, test environment, and data processing procedure. Modeling error, however, is a bias error as it produces a shifted response.

The purpose of an assessment procedure is the comparison between measured and modeling data. Accordingly, this procedure is limited by the combination of these two sources of errors. Numerical models are then updated until a certain limit (threshold value), defined by the contribution of these two components, is attained. However, understanding errors is essential to compute this limit value. Therefore, it is further indicated how these two sources of errors may be computed and combined.

4.6.2.1. Measurement

Measurement error (u_{exp}) is the difference between real and measured quantities in a single measurement. It usually results from equipment and on-site installation faults. There are some authors which tried to quantify this error. Restivo and Sousa [154] evaluated the measurement uncertainty with an example of a material Young modulus test. They used the law of propagation of uncertainty, indicated at GUM (Guide to the expression of Uncertainty in Measurement) [37, 38], to combine all different contributions (e.g. sensor characteristics, equipment resolution and repeatability of used procedure).

An interesting work about measurement uncertainties is the work of Cabral [21]. Within this report it is indicated the concept of maximum admissible error of a transducer which is the product of its linearity by its full scale. This author differentiates type-A uncertainties which are determined from several and repeated measures of the same measurand (e.g. measurement stability) from type-B uncertainties which may be given in bibliography (e.g. equipment resolution). Different sources of uncertainties, indicated at Table 4.1, are respectively introduced.

Table 4.1. Measurement errors: causes and quantification (adapted from Cabral [21]).

Sources	Causes	Quantification method
Measurement equipment to calibrate	Resolution; measurement stability; etc.	Manufacturer (last certification)
Transducer equipment	Calibration; separation from “ideal pattern”; etc.	Operator (calibration history)
Measurement technique	Suitability of the method for the final purpose; approximation introduced in the model; etc.	Operator (computing error)
Operator	Experience; training; etc.	Average from multiple samples
Environmental conditions	Temperature; humidity; etc.	Manufacturer (last certification)

The measurement error may be also divided in the following contributions [67]: (1) sensor accuracy, value reported by the manufacturer; (2) stability (e.g. fatigue, ambient vibrations and its attenuation); (3) robustness (e.g. environmental effects due to temperature, humidity, etc.); (4) load positioning; and (5) load intensity. Table 4.2 presents an example of a general division of measurement errors and attributed values.

This example concerns a pinned-pinned supported reinforced concrete beam which is loaded up to failure in laboratory. The applied load is monitored through a load cell positioned inside the actuator. The beam deflection is monitored with a linear variable differential transformer (LVDT). The sensor accuracy value is recommended to be twice the manufacturer value

(0.05%) in order to consider cable and acquisition equipment losses [67]. An explanation of how other uncertainty sources are determined is given in Table 4.2.

Table 4.2. Measurement errors in a reinforced concrete beam example.

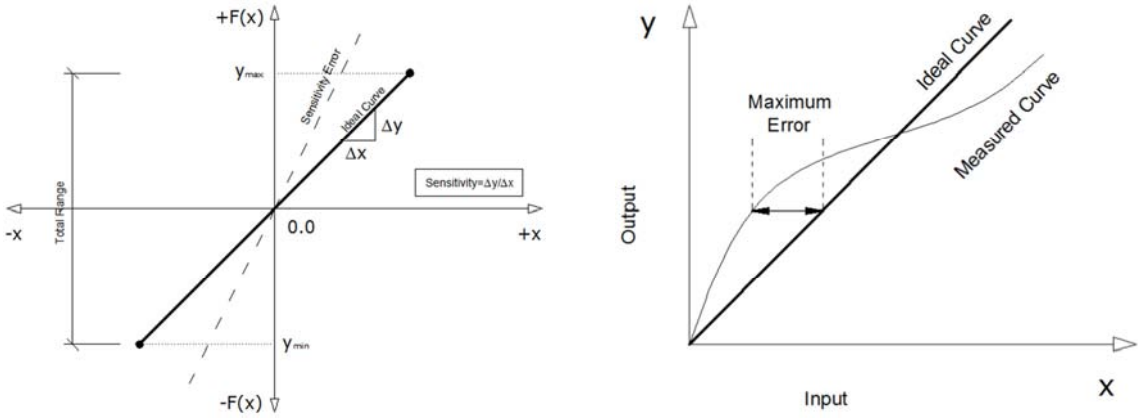
Sources	Quantification method	Value
		[%]
Sensor accuracy	Manufacturer (includes cable and acquisition equipment losses)	0.10 (LVDT)
Stability	Static load test (no fatigue or vibration problems detected)	→ 0.00
Robustness	Short term test (environmental effects neglected)	→ 0.00
Load positioning	Test assembly perfectly controlled	→ 0.00
Load intensity	Manufacturer (includes cable and acquisition equipment losses)	0.10 (load cell)

A sensor is a converter that measures a physical quantity (input, X) and converts it into a signal (output, Y) which can be read by an observer or by an electronic device. The sensor sensitivity is defined as the slope of the output characteristic curve ($\Delta Y/\Delta X$ in Figure 4.5a) or the minimum input of physical parameter that will create a detectable change in the output. In some sensors, the sensitivity is defined as the input parameter change required to produce a standardized output change. In others, it is defined as an output voltage change for a given change in input parameter. The sensitivity error (dotted curve in Figure 4.5a) is a departure from the ideal slope of the characteristic curve.

The linearity of the transducer is an expression of the extent to which the actual measured curve of a sensor departs from ideal curve. Figure 4.5b shows a somewhat exaggerated relationship between the ideal and the calibration line. Linearity is often specified in terms of percentage, which is defined as, (4.5),

$$Linearity = \frac{D_{in}(\max)}{IN_{f.s.}}[\%] \tag{4.5}$$

where linearity (%) is the percentage of linearity, D_{in} (max) is the maximum input deviation (absolute maximum error), $IN_{f.s.}$ is the maximum full-scale input. Transducers are usually classified by linearity classes. Both linearity and sensitivity values are important to characterize the sensor accuracy component of measurement error.

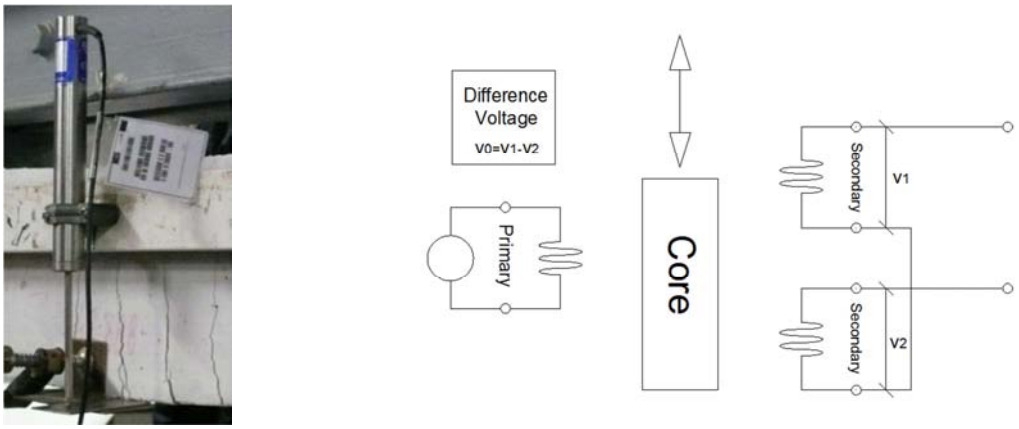


a) b)

Figure 4.5. Transducer properties: a) sensitivity; b) linearity.

A detailed study of such transducer properties is developed for LVDT, Figure 4.6a. These are widely used in civil engineer applications, mainly in load tests. LVDT is a linear displacement transducer that works on the principle of mutual inductance, producing an electrical signal that is proportional to the position of a separate moving core (Figure 4.6b). The transducer relates inductance to displacement by modifying the spatial distribution of an alternating magnetic field. A more detail description of this transducer is given in *fib* [55].

A rigorous study of different LVDT was developed. Chosen transducers vary according to their measuring field, manufacturer linearity and sensitivity. Table 4.3 indicates the characteristics of the studied equipment. It is important to note that the transducer sensitivity increases as the measuring field decreases and that linearity is an independent property (ranging around 0.10%). The following effects were studied: (1) cable length (10, 35 and 105 m); (2) wrapping cable, which introduce a magnetic field; and (2) calibration procedures. For each situation, both linearity and sensitivity values were determined.



a) b)

Figure 4.6. Linear variable differential transformer (LVDT): a) image; b) operating principle.

Table 4.3. Displacement transducers (characteristics).

Transducers						
LVDT	Type	Model / Reference	Series number	Measuring field	Linearity	Sensitivity
				[mm]	[%]	[mV/V/m]
1	RDP	GTX5000	40351	± 5.00	0.17	118.71
2	RDP	GTX5000	65050	±5.00	0.22	114.71
3	RDP	D6/05000A	83043	± 5.00	0.10	142.09
4	RDP	ACT500A	2987	±12.50	0.08	54.72
5	RDP	ACT500A	2990	±12.50	0.06	54.17
6	RDP	ACT500A	2934	±12.50	0.10	56.22
7	RDP	ACT1000A	3468	±25.00	0.08	33.84
8	RDP	ACT1000A	19906	±25.00	0.07	32.78
9	RDP	ACT1000A	82804	±25.00	0.10	34.00

Table 4.4 indicates the obtained results for LVDTs with a measurement field of ± 5.00 mm. The following conclusions are obtained: (1) the manufacturer linearity values are higher than those determined at laboratory for LVDT 1 and 2; (2) the manufacturer linearity value is close to that obtained at laboratory for LVDT 3; (3) the manufacturer sensitivity values are lower than those determined at laboratory; (4) it is verified an increase in linearity value with an increase in cable length; (5) the sensitivity value does not change with cable length; (6) the linearity and sensitivity values increases with wrapping for LVDT 1 and 3; (7) the linearity and sensitivity values remains unaffected with wrapping for LVDT 2; (8) the calibration procedure reduces the linearity values; (9) the sensitivity value decreases with the calibration procedure, especially for long cables; (10) the cable length effect in linearity value is reduced with the calibration procedure; (11) the wrapping effect in linearity value is minimized with the calibration procedure for LVDT 1 and 3.

A summary of obtained results for LVDTs with a measurement field of ± 5.00 mm is provided in Figure 4.7. Accordingly, it is verified an increase in linearity value with an increase in cable length. This variation is higher when calibration is not developed. This confirms the importance of calibrating the system transducer and cable before performing any test. Additionally the obtained linearity value with a cable length of 105 m is five times higher than the manufacturer value. Therefore, considering this value twice the recommended by

manufacturers, to take into account cable losses [67], may be not conservative when long cables are used.

Table 4.4. Results LVDT (± 5.00 mm).

LVDT	Cable length	Test type	Linearity		Sensitivity	
	[m]		Value	Mean	Value	Mean
			[%]		[mV/V/mm]	
1	5	calibrated	0.13	0.14	190.50	190.05
2			0.18		189.14	
3			0.12		190.50	
1	10	not calibrated	1.82	1.54	193.29	191.14
2			1.22		186.83	
3			1.60		193.29	
1	35	not calibrated	3.87	2.92	190.67	189.45
2			1.47		187.00	
3			3.42		190.67	
1	105	not calibrated and wrapped	13.41	9.82	202.46	199.07
2			4.21		192.30	
3			11.84		202.46	
1	105	not calibrated and stretched	4.81	6.63	195.09	195.44
2			4.25		193.42	
3			10.84		197.83	
1	10	calibrated	0.17	0.18	189.56	188.25
2			0.17		187.75	
3			0.19		187.43	
1	35	calibrated	0.17	0.31	189.50	187.93
2			0.17		187.10	
3			0.60		187.20	
1	105	calibrated and wrapped	0.18	0.59	194.11	190.33
2			0.19		187.75	
3			1.40		189.14	
1	105	calibrated and stretched	0.21	0.71	189.43	190.55
2			0.26		188.28	
3			1.67		193.94	

Table 4.5 indicates the obtained results for LVDTs with a measurement field of ± 12.50 mm. The following conclusions are obtained: (1) the manufacturer linearity value is lower than that determined at laboratory for LVDT 4; (2) the manufacturer linearity values are close to that obtained at laboratory for LVDT 5 and 6; (3) the manufacturer sensitivity values are lower than those determined at laboratory; (4) it is verified an increase in linearity value with an increase in cable length; (5) the sensitivity value does not change with cable length; (6) the linearity and sensitivity values remains unaffected with wrapping; (7) the calibration

procedure reduces the linearity values; (8) the sensitivity value remains unchanged with the calibration procedure; (9) the cable length effect in linearity value is reduced with the calibration procedure.

Table 4.5. Results LVDT (± 12.50 mm).

LVDT	Cable length	Test type	Linearity		Sensitivity	
	[m]		Value	Mean	Value	Mean
			[%]		[mV/V/mm]	
4	5	calibrated	0.18	0.12	78.87	78.65
5			0.08		78.47	
6			0.11		78.60	
4	10	not calibrated	0.22	0.42	77.86	77.20
5			0.41		76.75	
6			0.62		76.99	
4	35	not calibrated	1.75	1.89	76.47	76.28
5			1.77		76.22	
6			2.15		76.16	
4	105	not calibrated and wrapped	6.16	6.29	75.64	76.54
5			5.93		75.43	
6			6.78		78.54	
4	105	not calibrated and stretched	6.30	6.38	75.84	76.68
5			6.27		75.58	
6			6.58		78.64	
4	10	calibrated	0.09	0.10	78.80	77.53
5			0.09		76.85	
6			0.11		76.96	
4	35	calibrated	0.13	0.12	78.13	77.25
5			0.11		76.74	
6			0.12		76.86	
4	105	calibrated and wrapped	0.56	0.35	79.49	77.56
5			0.28		76.35	
6			0.21		76.83	
4	105	calibrated and stretched	0.30	0.26	77.10	76.82
5			0.27		76.49	
6			0.20		76.87	

Figure 4.8 provides a summary of obtained results for LVDTs with a measurement field of ± 12.50 mm. In this situation it is verified an increase in linearity value with an increase in cable length. This variation is higher when the system is not calibrated, confirming the importance of calibrating the system transducer and cable before performing any test. Moreover, a linearity value which is three times higher the manufacturer value is obtained when a cable length of 105 m is used. Therefore, considering this value twice the

recommended by manufacturers, to take into account cable losses [67], may be not conservative when long cables are used.

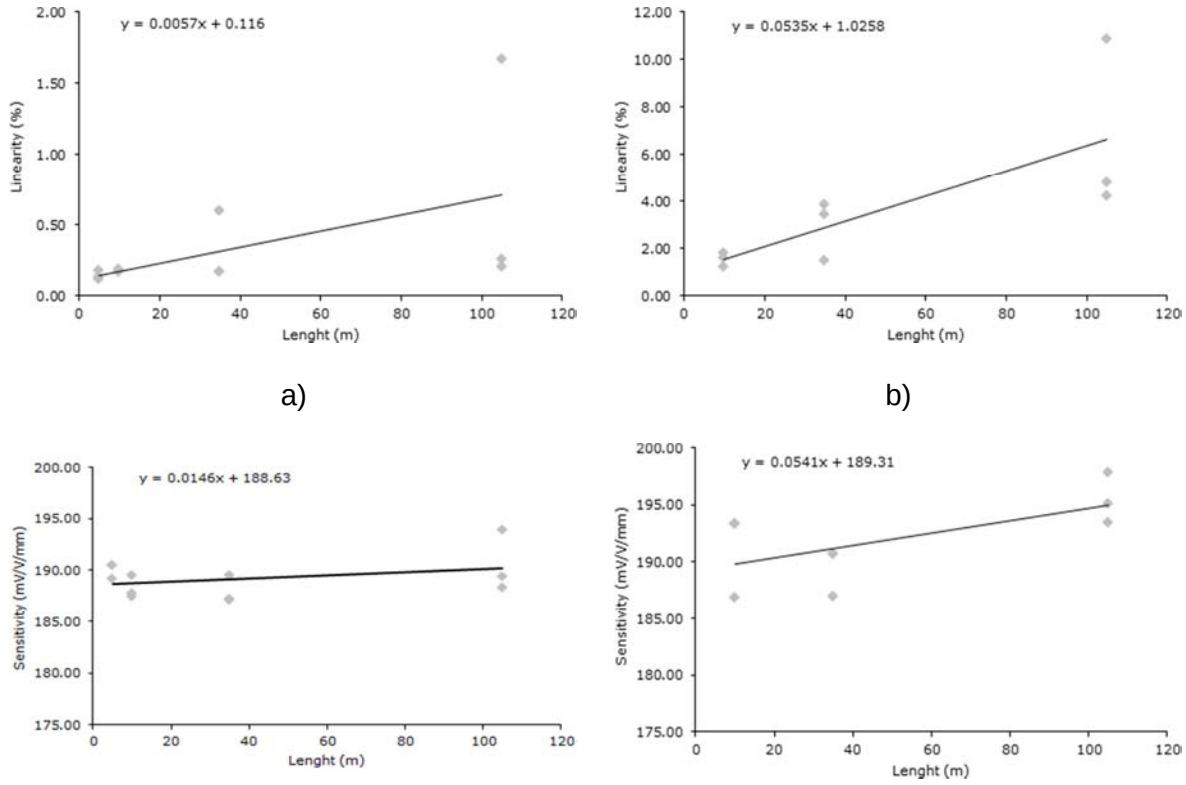


Figure 4.7. LVDT (± 5.00 mm): a) linearity calibrated; b) linearity not calibrated; c) sensitivity calibrated; d) sensitivity not calibrated.

Table 4.6 indicates the obtained results for LVDTs with a measurement field of ± 25.00 mm. The following conclusions are obtained: (1) the manufacturer linearity values are close to that obtained at laboratory for LVDT 7 and 8; (2) the manufacturer linearity value is lower than that determined at laboratory for LVDT 9; (3) the manufacturer sensitivity values are lower than those obtained at laboratory; (4) it is verified an increase in linearity value with an increase in cable length; (5) the sensitivity value does not change with cable length; (6) the linearity value remains unaffected with wrapping for LVDT 7 and 8; (7) the linearity value increases with wrapping for LVDT 9; (8) the sensitivity value remains unaffected with wrapping; (9) the calibration procedure reduces the linearity values; (10) the sensitivity value remains unchanged with the calibration procedure; (11) the cable length effect in linearity value is reduced with the calibration procedure; (12) the wrapping effect in linearity value is eliminated with the calibration procedure for LVDT 9.

A summary of obtained results for LVDTs with ± 25.00 mm of measurement field is given in Figure 4.9. An increase in linearity value is verified with an increase in cable length. This

variation is higher in the situation of no calibration. This confirms the necessity of developing the calibration of the system transducer and cable before performing any test. A linearity value which is three times higher the manufacturer value is obtained when a cable length of 105 m is used. Therefore, considering this value twice the recommended by manufacturers, to take into account cable losses [67], may be not conservative when long cables are used.

Table 4.6. Results LVDT (± 25.00 mm).

LVDT	Cable length [m]	Test type	Linearity		Sensitivity	
			Value [%]	Mean [%]	Value [mV/V/mm]	Mean [mV/V/mm]
7	5	calibrated	0.06	0.21	39.57	39.78
8			0.07		39.65	
9			0.49		40.12	
7	10	not calibrated	0.18	0.48	39.19	38.98
8			0.20		39.21	
9			1.05		38.55	
7	35	not calibrated	0.69	1.02	39.19	38.91
8			1.01		39.08	
9			1.35		38.45	
7	105	not calibrated and wrapped	3.20	3.84	39.23	39.97
8			3.96		40.18	
9			4.37		40.51	
7	105	not calibrated and stretched	2.99	3.05	39.18	39.67
8			3.42		39.34	
9			2.75		40.49	
7	10	calibrated	0.06	0.09	39.18	39.16
8			0.08		38.62	
9			0.13		39.69	
7	35	calibrated	0.24	0.21	39.23	39.14
8			0.20		38.59	
9			0.18		39.60	
7	105	calibrated and wrapped	0.54	0.49	39.20	39.38
8			0.46		39.36	
9			0.46		39.58	
7	105	calibrated and stretched	0.33	0.40	39.93	39.64
8			0.44		39.35	
9			0.43		39.63	

Commonly the values provided by manufacturers for sensitivity are lower than those determined at laboratory. This is explained by the fact that this property is determined by manufacturers in special site conditions that are practically impossible to reproduce at

laboratory. Additionally, some of these transducers already have years of use and this has an impact in determined sensitivity values.

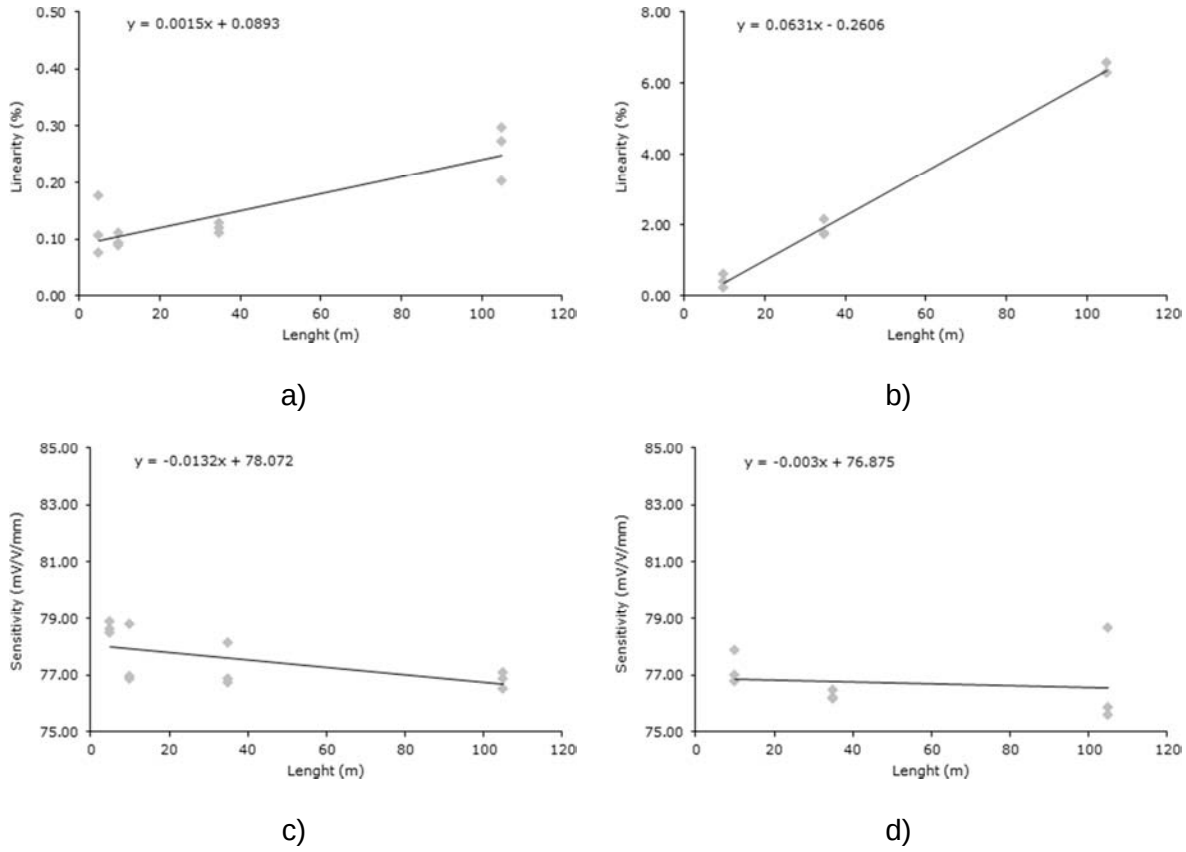


Figure 4.8. LVDT (± 12.50 mm): a) linearity calibrated; b) linearity not calibrated; c) sensitivity calibrated; d) sensitivity not calibrated.

Figure 4.10 indicate a summary of all results obtained with different LVDTs, respectively, for measurement fields of ± 5.00 mm (light), ± 12.50 mm (medium) and ± 25.00 mm (dark), which characteristics are provided in Table 4.3. The main conclusions are that: (1) linearity increases substantially with cable length being this value unaffordable when long cables are used; (2) the variation in linearity is higher when no calibration procedures are taken; (3) calibration reduces linearity values and so it is recommended to always calibrate the system transducer and cable previously to any application; (4) sensitivity is independent of both cable length and calibration procedures; (5) sensitivity is a property of used transducers as it only varies with measurement field.

Within this analysis it is verified that the sensor accuracy component of measurement error is only dependent on linearity, being sensitivity an intrinsic property of used sensor. However, it is also shown that the consideration of this component as twice the linearity value given by manufacturers, to take into account cable and acquisition equipment losses [67], may be not conservative when long cables are used. Therefore, the linearity value should be detailed

evaluated in these situations. Other measures are also suggested: (1) not to use long cables in monitoring systems; (2) always calibrate the system transducer with cable; and (3) avoid wrapping cables, as it may increase the linearity value.

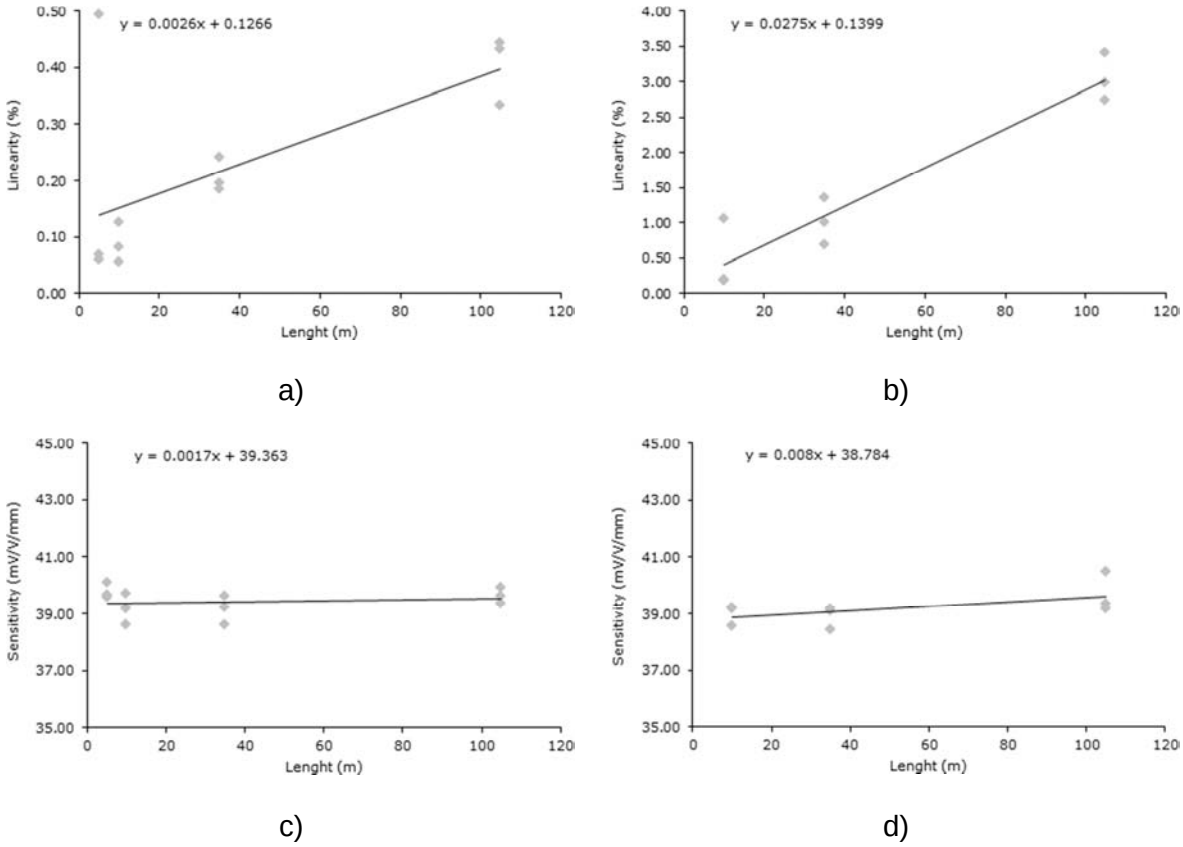


Figure 4.9. LVDT (± 25.00 mm): a) linearity calibrated; b) linearity not calibrated; c) sensitivity calibrated; d) sensitivity not calibrated.

According to Table 4.2, other sources of measurement error exist. In the examples of this thesis developed tests were performed in short term, which eliminates the environmental conditions effect (robustness), and loads were applied in a static way, which eliminates vibration and fatigue problems (stability) [67]. Therefore, these components are not detailed study in this thesis. In respect to applied load intensity and positioning, it is first important to identify if the test is developed at laboratory or not. In laboratory, the applied load is perfectly controlled and load positioning error may be neglected. In this situation the load intensity error is determined in a similar way as the sensor accuracy, but now considering a load cell. If in field, then load positioning is more difficult and some measures should be taken to minimize this error: (1) introduce lines in the pavement in specific coordinates for each load case; (2) measure the distance between vehicle axles. In this situation, load intensity may also be measured by weighting each vehicle. Once these different sources are identified it is

then necessary to compute the measurement error (u_{exp}) through the law of propagation of uncertainties [37, 38].

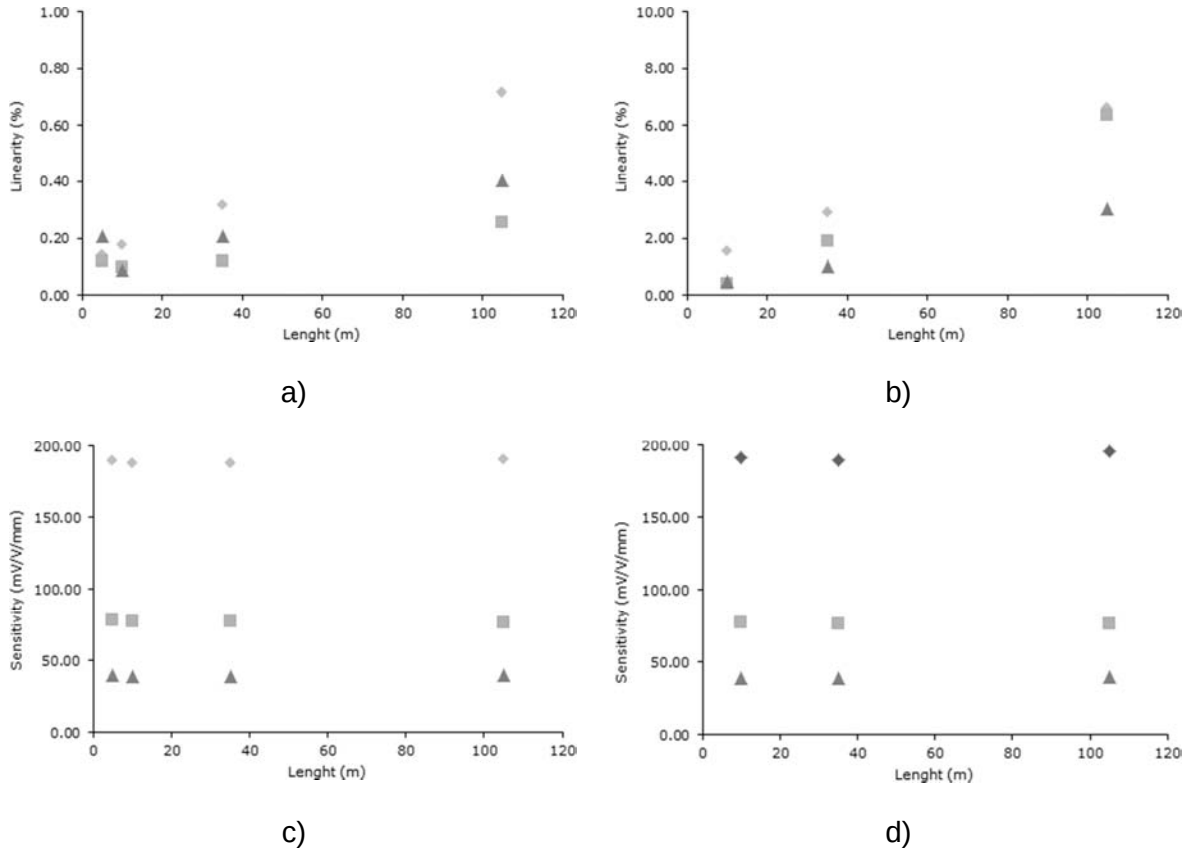


Figure 4.10. LVDT (light: ± 5.00 mm; medium: ± 12.50 mm; dark: ± 25.00 mm): a) linearity calibrated; b) linearity not calibrated; c) sensitivity calibrated; d) sensitivity not calibrated.

4.6.2.2. Modeling

Modeling error (u_{num}) is the difference between the response of a given model and that of an ideal model that accurately represents the structural behavior. Figure 4.11 presents the respective propagation. It is possible to divide this error in three main components [152]: (a) u_1 is due to discrepancy between the behavior of a mathematical model and that from the real structure; (b) u_2 is introduced during numerical computation of the solution of partial differential equations (e.g. finite element method, mesh quality, etc.); and (c) u_3 arises from inaccurate assumptions made during simulation (e.g. boundary condition such as support characteristics, applied load steps, etc.). Component u_1 is extremely difficult to quantify, it is problem dependent and it can be minimized using modeling expertise [12]. Ravindram *et al.* [152] proposes to consider this component as null when an ideal situation is assumed.

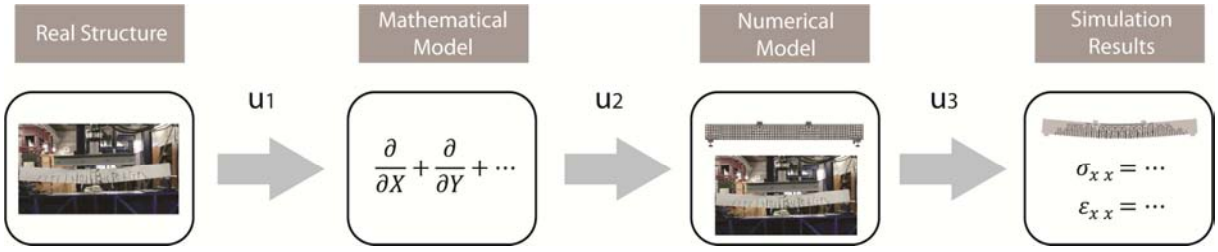


Figure 4.11. Modeling errors u_1, u_2 and u_3 (adapted from Ravindram *et al.* [152]).

On its preliminary study, Goulet *et al.* [69] present the estimation of modeling errors in model identification and its consequences. For these authors, modeling errors are classified in two classes, respectively: epistemic and aleatoric. The former arise from lack of knowledge about structural behavior, use of simplified hypothesis or modeling assumptions. Aleatoric errors describe uncertainties that are related to parameter values. These authors, on their analyses, only consider the epistemic uncertainties.

For Goulet *et al.* [67] the modeling uncertainties are divided in the following sources: (a) finite element method (u_1); (b) mesh discretization accuracy (u_2); (c) uncertainties in geometry of analyzed structure (u_3); (4) other assumptions (e.g. bearing devices assumed to be point loads) (u_3). These authors only consider the component u_2 and u_3, neglecting thus the component u_1 [152]. Table 4.7 summarizes these sources.

Table 4.7. Modelling errors: causes and quantification.

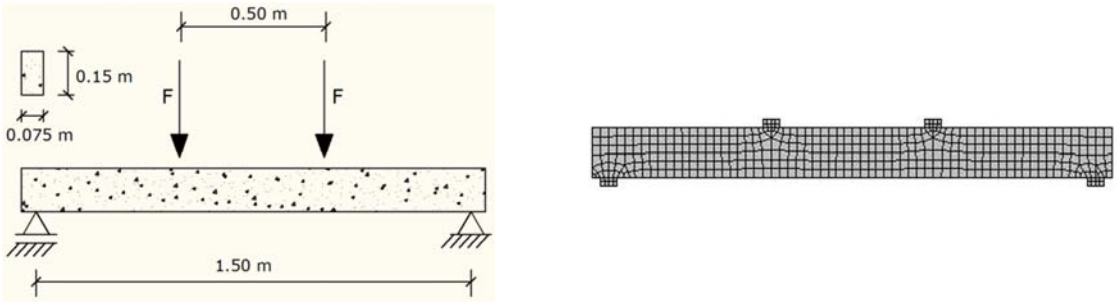
Sources	Causes	Quantification method
Finite element method	Modelling experience and judgment	Discrepancy between the real structure and the mathematical model (u_1)
Mesh discretization	Numerical analysis	Numerical computation of the solution of partial differential equations (u_2)
Model exactitude	Dimensions taken from the “as-built” structure	Inaccurate assumptions made during simulation (u_3)
Other assumptions (e.g. linear elastic behavior, bearing devices and truck wheels represented by point loads, etc.)	Linear superposition, other validating models	Inaccurate assumptions made during simulation (u_3)

The finite element method and mesh discretization component errors are computed by comparing the obtained results from the numerical model, used in model identification procedure, with a reference model. This comparison is established through the following expression (4.6),

$$u = \sum_{i=1}^{n} \left| y_i^{num1} - y_i^{num0} \right| \Big/ \max\left(y^{num0} \right) \cdot 1/n \, [\%] \tag{4.6}$$

in which y^{num0} indicate the registered value from the reference model, y^{num1} is the obtained value from the numerical model, used in model identification, and n the number of comparing points.

An example of a pinned-pinned supported reinforced concrete beam which was loaded up to failure in laboratory (Figure 4.12a), identical to that used to explain the measurement errors (Table 4.2), is presented to indicate typical modeling errors. In this situation the studied beam is modeled with an accurate finite element mesh (Figure 4.12b).



a) b)

Figure 4.12. Modelling errors in a reinforced concrete beam example: a) scheme; b) finite element mesh [116, 117].

In this case, the component u_1 of modeling error, due to the use of a finite element method, is neglected as it is supposed an ideal situation of modeling procedures [152]. However, other components, due to mesh discretization (u_2) and to inaccurate assumptions (u_3), are considered. Table 4.8 provides each component value. By comparing with results from Table 4.2 it is possible to conclude that modeling errors are much higher than measurement errors. This is common in laboratory controlled environments in which the measurement errors are very low.

In this situation model exactitude is neglected as it is assumed that the developed model represents geometry as accurate as possible. Other hypotheses, such as the representation of bearing devices as point loads, are also neglected in this case once a global structural analysis is developed [67, 69]. The mesh discretization and inaccurate assumptions are computed through expression (4.6). Once these different sources are identified it is then necessary to compute the modeling error (u_{num}) through the law of propagation of uncertainties [90, 91, 92].

Table 4.8. Modelling errors in a reinforced concrete beam example.

Sources	Quantification method	Value
		[%]
Finite element method	It is minimized through modelling experience	→ *0.00*
Mesh discretization	Based on preliminary study (by comparing to a refined mesh model)	*6.74*
Model exactitude	Model "as built"	→ *0.00*
Inaccurate assumptions	Based on preliminary study (by comparing to a model with a smaller load step length)	*3.79*
Considered hypotheses	Other hypothesis are negligible	→ *0.00*

4.6.3. Convergence criterion

There are several possibilities of applying the fitness function convergence criterion of model identification (Figure 4.3). A procedure is presented by Goulet *et al.* [67, 69], in which the numerical models that present a fitness function value lower than a pre-specified threshold value are considered as candidate models. Ravindram *et al.* [152] proposes a formulation for fitness function which helps to understand the importance of each component error on the convergence criterion proposed by Goulet *et al.* [67, 69]. Considering y as the real behavior of a quantity, y^{exp} the measured and y^{num} the numerical value, the following relationships are thus obtained through expression (4.7),

$$y = y^{\exp} + u_{\exp} \tag{4.7}$$

and (4.8),

$$y = y^{num} + u_{num} = y^{num} + \left(u_1 + u_2 + u_3\right) \tag{4.8}$$

Model identification procedure intends to minimize the absolute value of the difference between y^{num} and y^{exp} known as residual (q). Rearranging the terms in previous expressions (4.7) and (4.8), it follows the expression (4.9),

$$q = \left|y^{num} - y^{\exp}\right| = \left|u_{num} - u_{\exp}\right| = \left|\left(u_1 + u_2 + u_3\right) - u_{\exp}\right| \tag{4.9}$$

Considering that both u_{num} and u_{exp} might be positive or negative, it is possible to conclude that, (4.10),

$$q \le \left|y^{num}\right| + \left|y^{\exp}\right| \le \left|u_{num}\right| + \left|u_{\exp}\right| \le \left|u_1\right| + \left|u_2\right| + \left|u_3\right| + \left|u_{\exp}\right| \tag{4.10}$$

Being the fitness function directly dependent of computed residual *f(q)*. Considering expression (4.10), the proposed convergence criterion by Goulet *et al.* [67, 69] may be reduced to expression (4.11),

$$f\left(q\right) \le u \tag{4.11}$$

being *u* the global uncertainty. This value establishes a superior limit for the fitness function. In this situation, the convergence criterion defines that the improvement in fitness function value (*Δf*), from two models separated of a pre-specified gap (*n*), should be lower than or equal to a threshold value (ε), indicated at Figure 4.3. Accordingly, by considering expression (4.4), it is possible to conclude that, (4.12),

$$\Delta f \le \left|f_{i+n} - f_i\right| \le \left|f_{i+n}\right| + \left|f_i\right| \tag{4.12}$$

being f_i and f_{i+n}, respectively, the fitness function values for generation *i* and *i+n*. Therefore, and according to the methodology proposed by Ravindram *et al.* [152], the convergence criterion may be reduced to (4.13),

$$\Delta f \le \left|f_{i+n}\right| + \left|f_i\right| \le \left|u_{i+n}\right| + \left|u_i\right| \tag{4.13}$$

being u_i and u_{i+n} the global uncertainty, computed for generation *i* and *i+n*. This value establishes a superior limit for the improvement in fitness function.

In order to compute the global uncertainty (*u*) it will be necessary to combine both measurement and modeling errors through the law of propagation of uncertainty, indicated at JCGM [90, 91, 92], (4.14),

$$u^2 = \sum_{i=1}^{n} \left(\partial f / \partial x_i\right)^2 \cdot u(x_i)^2 + 2 \cdot \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left(\partial f / \partial x_i\right)^2 \cdot \left(\partial f / \partial x_j\right)^2 \cdot u(x_i) \cdot u(x_j) \cdot r\left(x_i, x_j\right) \tag{4.14}$$

where *u(x)* is the uncertainty related to each item *x*, *∂f/∂x* is the partial derivative of the fitness function in order to item *x*, and $r(x_i,x_j)$ the correlation coefficient. The following expression (4.15) is obtained when a null correlation coefficient is considered,

$$u^2 = \sum_{i=1}^{n} \left(\partial f / \partial x_i\right)^2 \cdot u(x_i)^2 \tag{4.15}$$

The fitness function, presented in expression (4.2), is composed by two terms, respectively, a numerical and an experimental. The partial derivative, in relation to each term, can be determined through a sensitivity analysis. This analysis is developed by sequentially

changing each term and, at same time, studying the fitness function response. The partial derivative values, obtained from this analysis, are given by expression (4.16),

$$\left(\partial f/\partial y^{num}\right) = 1/\max\left(y_i^{\exp}\right) \tag{4.16}$$

and (4.17),

$$\left(\partial f/\partial y^{\exp}\right) = 1/\max\left(y_i^{\exp}\right) \tag{4.17}$$

In order to apply expression (4.15) to determine the global uncertainty value it will be necessary to separately compute both measurement and modeling errors, and then to use the partial derivative values given by expression (4.16) and (4.17).

In some situations it is measured more than a structure property and/or in more than a location. Additionally, there are some cases in which there is more than a tested structure, in identical conditions, and/or more than a load case. The fitness function, in this situation, is given by expression (4.3). Accordingly, in this situation, the uncertainty related to each measured property in a specific location of the tested structure and for the studied load case is computed. Further, the expression (4.15) is applied to obtain the global uncertainty. In this case the partial derivatives $\partial f/\partial f_{i,j,k,l}$ are unitary.

According to expression (4.13), the threshold value (ε) is computed by taking into consideration the sum of global uncertainties from two generations (u_i and u_{i+n}) as a superior limit. These uncertainties are computed through expression (4.15). In this situation, the partial derivatives $\partial\Delta f/\partial u_{i+n}$ and $\partial\Delta f/\partial u_i$ are unitary. Obtained value is then introduced in the optimization algorithm of the model identification procedure.

4.6.4. Engineering judgment procedure

Global optimization algorithms, as the evolutionary strategies in its plus version [29], when incorporated in model identification, usually provide a population of models. A family of numerical models is thus extracted when an algorithm criteria is accomplished (Figure 4.3). At the same time, it is recommended to run each algorithm more than once with different initial points. Therefore, several models are obtained from model identification, being important to decide about the most adequate model.

The selection of these models may be based in experience [67] or eventually in more robust algorithms. However, even in this latter situation an engineer judgment criterion is always necessary. In this situation, an algorithm based in the probability of occurrence of each model is developed to select the most adequate model. This algorithm is based in the principle that the most suitable model is that which assessed parameter values are close to

initial mean values, unless some accidental situation is detected. Therefore, the following expression (4.18) might be used,

$$f_d = \prod_{i=1}^{n} f\left(x_{id}\right) \tag{4.18}$$

where x_{id} is the value of the assessed parameter i, and $f(x_{id})$ is the PDF value for this parameter, assuming a PDF from bibliography [93] or based in experience. Figure 4.13 illustrates how the PDF value is obtained.

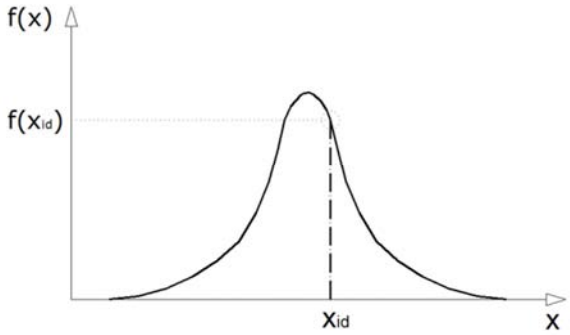


Figure 4.13. Engineering judgment procedure.

Then the product of all PDF values, for all assessed parameters, and for all extracted models, is computed (f_d). The updated model, from proposed model identification procedure, is that which presents the highest value. An engineer judgment criterion is intrinsic to the developed algorithm, when assuming the principle of the probability of occurrence of each model.

4.7. Probabilistic analysis

In a second step, and according to Figure 4.1, the proposed probabilistic assessment methodology performs the reliability analysis of the assessed structure in order to evaluate, from a probabilistic point of view, its behavior. Therefore, the previous updated numerical model is converted into a probabilistic model by introducing randomness in its model parameters.

This randomness represents the physic uncertainty source. The aleatoric variation in material, geometric and physic structure properties is therefore considered. Accordingly, each parameter is represented by a PDF. A correlation matrix, composed by correlation coefficients (ρ_{ij}), is also used. This coefficient is a measure of strength of linear dependence between two parameters. It is computed through the Pearson coefficient, which is defined as the covariance of two parameters divided by the product of their standard deviations [33,

123]. This value varies between -1 and 1. The closer the coefficient is to either -1 or 1, the stronger is the correlation between the parameters.

The statistical uncertainty is incorporated into the probabilistic model when establishing each parameter PDF (e.g. when defining the mean and the standard deviation value). This uncertainty source is reduced when more information regarding the parameter PDF is obtained (e.g. complementary data). A Bayesian inference algorithm is then used to incorporate such information into developed probabilistic model.

A sampling procedure is therefore implemented to randomly generate the model parameter values. For each set of generated parameter values, the updated numerical model is analyzed with nonlinear structural analysis software, being the obtained results statistically processed.

Two procedures may be developed to evaluate the structure behavior. Each procedure gives an appropriate structural performance index. In the first procedure the structure behavior is analyzed by a comparison with acquired experimental data. An evaluation index is introduced to interpret the reliability of obtained probabilistic model. In the second procedure the structure resistance is compared with loading. In this situation a reliability index, necessary to evaluate the structural safety, is computed.

Within the probabilistic analysis, some Matlab® routines [25, 129] were implemented. These routines are provided in Appendix B, respectively:

(1) *run.m*: run the probabilistic analysis. It defines the number of generated samples, each parameter PDF type and the correlation matrix. Additionally, it statistically processes the obtained numerical results and computes the performance indexes;

(2) *lhs_iman_n.m* [173]: generate random samples according to the Latin Hypercube sampling (LHS) algorithm, by guaranteeing the correlation coefficient values between these parameters through the application of the Iman and Conover algorithm [85];

(3) *mchol.m* [173]: induce data with correlation by performing the Cholesky decomposition;

(4) *latin_hs.m* [173]: generate data according to the LHS algorithm;

(5) *ranking.m* [173]: perform the ranking of a vector within the LHS algorithm;

(6) *parameter.m*: define each model parameter value;

(7) *data.m*: generate the data file which will be read and then processed in structural analysis software;

(8) *processment.m*: process the generated data file and convert the obtained results into Matlab® [25, 129];

(9) *matrix_num.m*: convert the obtained results from the structural analysis software into the same unities of experimental data, in order to be compared;

(10) *test_num.m*: divide the numerical results into specific steps in order to guarantee that the same points of experimental data are compared;

(11) *run_exp.m*: read experimental data and convert it into Matlab® [25, 129];

(12) *matrix_exp.m*: convert the experimental data into the same unities of numerical results, in order to be compared;

(13) *test_exp.m*: divide the experimental data into specific steps in order to guarantee that the same points of numerical results are compared;

(14) *plot.m*: program used to plot both experimental and numerical results.

Additionally the nonlinear structural analysis software ATENA® [24] is used, being incorporated the respective executable files: (1) *atenaconsole.exe*: run the developed nonlinear numerical model using the finite element methodology; (2) *gawk.exe* [3]: convert obtained results from ATENA® into readable data for Matlab® processing.

4.7.1. Randomness

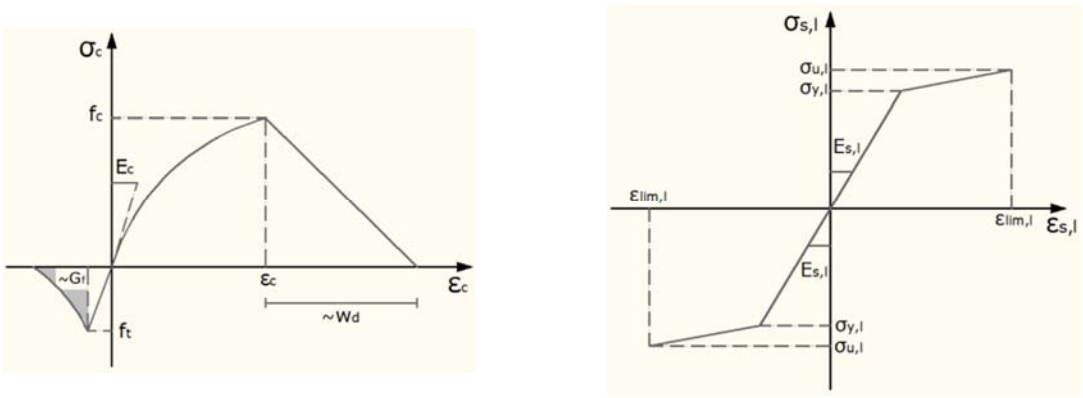
Randomness is introduced in numerical model by representing its parameters through appropriate PDF. The dependence between such parameters is given by adequate correlation coefficients. The majority of structural parameters present well defined PDF. However, there are some parameters for which the PDF is not defined in bibliography. In such situations an engineering judgment procedure is necessary. The correlation between structural parameters presents a high variability. In fact, these coefficients vary a lot with developed characterization tests being very difficult to quantify.

The PDF for current materials, such as concrete and steel, and for typical geometric elements (e.g. section dimensions and concrete cover) is defined in bibliography [77, 93, 134, 164, 196]. These PDFs may be adopted in both reinforced concrete and composite structures and they are important to determine the structure resistance capacity. Wisniewski [196] also indicates some reference values for pre-cast concrete structures. For these structures the standard deviation is lower than for cast-in-situ structures, as the quality control is higher.

4.7.1.1. Material

Concrete and steel are commonly used in civil engineer structures. Steel can be applied in reinforced concrete structures, as a reinforcing bar embedded in concrete, and in metallic or composite structures. There is a huge variety of concrete materials. A kind of concrete materials, whose application increased in the last years, is the lightweight concrete. Figure 4.14 permit to identify used parameters to describe each material behavior.

Concrete is a high heterogeneous material, defined by a strong nonlinear behavior (Figure 4.14a). The variability of its mechanical properties depends on the following factors: (1) component properties (e.g. cement type); (2) concrete composition (e.g. water-cement ratio); (3) execution (e.g. curing process); (4) testing procedure (e.g. velocity of applied load); (5) concrete being in the structure rather than in control specimens; (6) maintenance, material degradation, etc.



a) b)

Figure 4.14. Material behavior: (a) concrete; (b) steel (e.g. reinforcing steel).

The most investigated parameter is the compressive strength (f_c) as it serves to control the concrete quality during execution and is used to define the acceptance criteria. Other properties are also evaluated but due to a high correlation with concrete compressive strength they are usually defined via empirical relations.

According to JCSS [93] the concrete strength at any specific location *i*, in a given structure *j*, as a function of its standard strength f_{c0}, is given by the following expression (4.19),

$$f_{c,ij} = \alpha(t,\tau) f_{c0,ij}^{\lambda} Y_{1j} \tag{4.19}$$

The respective standard strength is given by (4.20),

$$f_{c0,ij} = \exp\left(U_{ij}\,\Sigma_j + M_j\right) \tag{4.20}$$

being $f_{c0,ij}$ a Lognormal variable, independent of Y_{1j}, with PDF mean (M_j) and standard deviation (Σ_j) parameters at job j, Y_{1j} a Lognormal variable representing additional variations due to placing, curing and hardening conditions of in-situ concrete at job j, U_{ij} a standard Normal variable representing variability within a structure, λ a Lognormal variable with mean 0.96 and CV 0.005 and $\alpha(t,\tau)$ a deterministic function which takes into account the concrete age at loading time t [days] and the duration of loading τ [days].

The remaining concrete properties as tensile strength (f_t), elasticity modulus (E_c) and ultimate strain in compression (ε_c) are given by (4.21),

$$f_{t,ij} = 0.3 f_{c,ij}^{2/3} Y_{2j} \tag{4.21}$$

and (4.22),

$$E_{c,ij} = 10.5 f_{c,ij}^{1/3} Y_{3j} \left(1 + \beta_d \phi(t,\tau)\right) \tag{4.22}$$

and (4.23),

$$\varepsilon_{c,ij} = 6 \cdot 10^{-3} f_{c,ij}^{-1/6} Y_{4j} \left(1 + \beta_d \phi(t,\tau)\right) \tag{4.23}$$

being ϕ the creep coefficient, β_d the total load which depends of the structure type and reflecting the variables Y_{2j}, Y_{3j} and Y_{4j}, variations due to factors not well accounted for by concrete compressive strength. Table 4.9 indicates the recommended PDF parameters for those variables.

Table 4.9. PDF for variables Y_{ij} [93].

Parameter	Variable	PDF	Mean value	CV
				[%]
Compression (f_c)	Y_{1j}	Lognormal	1.00	6.00
Tension (f_t)	Y_{2j}	Lognormal	1.00	30.00
Elasticity Modulus (E_c)	Y_{3j}	Lognormal	1.00	15.00
Ultimate Strain (ε_c)	Y_{4j}	Lognormal	1.00	15.00

The PDF indicated at JCSS [93] is a result of several studies, developed by different researchers. A summary of part of these research studies is presented in Wisniewski [196]. This author also compared precast and cast in-situ concrete material properties. Accordingly, the model of Bartlett and MacGregor [13] is recommended to define the concrete compressive strength. This model is given by equation (4.24),

$$f_{c,real} = F_1 F_2 f_{c,nominal} \tag{4.24}$$

being $f_{c,real}$ the real concrete compressive strength in the structure, $f_{c,nominal}$ the nominal concrete compressive strength, F_1 a random variable relating cylinder to nominal compressive strength, F_2 a random variable relating in-situ to cylinder strength, and by equation (4.25),

$$V_{c,real} = V_{F_1}^2 + V_{F_2}^2 \tag{4.25}$$

where $V_{c,real}$ is the real CV of concrete compressive strength, and V_{F1} and V_{F2} the CV of, respectively, F_1 and F_2.

For Wisniewski [196], a Normal or Lognormal PDF is used for these properties. The following statistical properties for concrete compressive strength are respectively indicated: (1) cast in-situ: $f_{c,real}$ = 1.00 * $f_{c,nominal}$, considering F_1 = 1.20 and F_2 = 0.85; $V_{c,real}$ = 12%, considering V_{F1} = 7% and V_{F2} = 10%; (2) precast: $f_{c,real}$ = 1.00 * $f_{c,nominal}$, considering F_1 = 1.10 and F_2 = 0.90; $V_{c,real}$ = 9%, considering V_{F1} = 5% and V_{F2} = 8%. Regarding probabilistic models for other concrete properties, Wisniewski [196] proposes to model the tensile strength (f_t) by a Normal or Lognormal PDF with CV of 20%, while the elasticity modulus (E_c) can be modeled by a Normal PDF with CV of 8%.

A parameter that is frequently used in numerical models but for which there is few information regarding its PDF is the fracture energy (G_f). This parameter may be defined by a Weibull PDF with a CV of 20% [174, 176]. In respect to correlation matrices for concrete properties, Strauss *et al.* [174, 176] propose the coefficients indicated at Table 4.10. These values are purely indicative as there are no normalized values for them due to its high variability.

Although, there are several probabilistic models for traditional concrete material, the information regarding the lightweight concrete material is still scarce, being thus very difficult to establish appropriate PDF. Some information about this material is given in EuroLightCon [51] and in Nowak *et al.* [135]. The former provides the main results of a European research project that focused a lightweight aggregate concrete. The second reference provide results of a research funded by the Portland Cement Association and presents PDF for several kind of concrete materials, including lightweight aggregate concrete. According to these authors it is recommended to use Normal PDF for all properties. A CV of 10% is proposed for compressive strength (f_c) and for elasticity modulus (E_c), while a value of 20% is indicated for tensile strength (f_t). By analyzing the obtained results by Valente [188] a value of 10% may be considered for all parameters, including the fracture energy (G_f).

Table 4.10. Correlation of concrete parameters (adapted from Strauss *et al.* [174, 176]).

	E_c	f_t	f_c	G_f
E_c	1.00	0.70	0.90	0.50
f_t	0.70	1.00	0.80	0.90
f_c	0.90	0.80	1.00	0.60
G_f	0.50	0.90	0.60	1.00

The variability of steel parameters is generally lower than the variability of concrete parameters (Figure 4.14b). This is due to the higher quality control implemented during its production. According to Sobriño [170] the main factors that might cause the variation of reinforcing steel bars are: (1) variability of material strength; (2) variability of bar geometry; (3) material degradation (e.g. corrosion); (4) load history (e.g. fatigue); (5) used method for strength parameters determination (e.g. velocity of applied load).

The probabilistic model code defines two independent models for reinforcing steel bars and laminated steel profiles. Therefore, the JCSS [93] proposes, for reinforcing steel, the definition of the steel bar yield strength (σ_y) as the sum of three independent Gaussian variables, (4.26),

$$X_1(d) = X_{11} + X_{12} + X_{13} \tag{4.26}$$

being $X_{11} = N(\mu_{11}(d),\sigma_{11})$ the variations in the global mean of different mills, $X_{12} = N(0,\sigma_{12})$ the batch to batch variation, $X_{13} = N(0,\sigma_{13})$ the variation within a single batch and d the nominal bar diameter. Accordingly, for a steel production of good quality the following standard deviation values can be used: (1) σ_{11} = 19 MPa; (2) σ_{12} = 22 MPa; and (3) σ_{13} = 8 MPa. This results in an overall standard deviation σ = 30 MPa. The yield strength mean value ($\mu = \mu_{11}$) is thus defined through equation (4.27),

$$\mu = S_{nom} + 2 \cdot \sigma \tag{4.27}$$

being S_{nom} the nominal value. In Table 4.11 it is indicated the PDF parameter values for reinforcing steel. For these variables Normal PDF can be adopted. Table 4.12 presents values for correlation coefficients between those parameters.

The JCSS [93] proposal is a conclusion of numerous research studies, developed worldwide by different authors. A summary of some of these works is indicated in Wisniewski [196]. This author proposes the following statistical parameters for reinforcing steel properties, in which λ is the bias factor or the ratio between the mean and the nominal value: (1) yield strength (σ_y): for older steels λ = 1.20 and CV = 10%, and for modern steels λ = 1.15 and CV

= 5%; (2) ultimate strength (σ_u): consider a value 15 - 20% higher than yield strength, and CV = 10% and 5%, respectively, for older and modern steels. These properties can be modeled by Normal or Lognormal PDF. Limit strain (ε_{lim}) can be modeled by Normal or Lognormal PDF with mean value of 10% and CV = 15%. Steel elasticity modulus (E_s) can be modeled by Normal PDF with mean value 202 GPa and CV = 4%. Reinforcing steel area (A_s) can be also modeled by a Normal PDF with CV = 2% and the mean value equal to its nominal value.

Table 4.11. PDF for reinforcing steel parameters [93].

Parameter	Mean value	Standard deviation	CV
			[%]
Bar area (A_s)	$A_{s,nom}$	-	2.0
Yield strength (σ_y)	$\sigma_{y,nom} + 2\sigma$	30 MPa	-
Ultimate strength (σ_u)	-	40 MPa	-
Ultimate strain (ε_{lim})	-	-	9.0

Table 4.12. Correlation between reinforcing steel parameters [93].

	A_s	σ_y	σ_u	ε_{lim}
A_s	1.00	0.50	0.35	0.00
σ_y	0.50	1.00	0.85	-0.50
σ_u	0.35	0.85	1.00	-0.55
ε_{lim}	0.00	-0.85	-0.55	1.00

The probabilistic model code indicates a different PDF for steel material in laminated steel profiles. Table 4.13 present the PDF parameter values for steel parameters according to JCSS [93]. These values are based on European studies developed from 1970 onwards. A Lognormal PDF is recommended for those parameters. Table 4.14 presents the correlation coefficient values between those variables.

Respectively, the suffix (sp) is used for the code nominal value, α is a spatial position factor (α = 1.05, for webs of hot rolled sections, and α = 1.00, otherwise), u is a factor related to the fractile of used PDF to describe the distance between the code nominal value and the mean value (u is found to be in the range of -1.5 to -2.0 for steel produced in accordance with relevant EN standards), C is a constant reducing the yield strength as obtained from usual mill tests to static yield strength (a value of 20 MPa is recommended) and parameter B = 1.5

for structural carbon steel, B = 1.4 for low alloy steel and B = 1.1 for quenched and tempered steel.

Table 4.13. PDF for steel profile parameters [93].

Parameter	Mean value	CV
		[%]
Yield strength (σ_y)	$\sigma_{y,sp}$ * α * exp(-u * CV) - C	0.07
Ultimate strength (σ_u)	B * E[σ_u]	0.04
Elasticity modulus (E_s)	$E_{s,sp}$	0.03
Ultimate strain (ε_{lim})	$\varepsilon_{lim,sp}$	0.06

Table 4.14. Correlation between steel profile parameters [93].

	σ_y	σ_u	E_s	ε_{lim}
σ_y	1.00	0.75	0.00	-0.45
σ_u	0.75	1.00	0.00	-0.60
E_s	0.00	0.00	1.00	0.00
ε_{lim}	-0.45	-0.60	0.00	1.00

Connectors, used in composite structures, to establish the connection between concrete slab and laminated steel profile, are traditionally in steel material. The Nelson headed studs [131] is a connector type, commonly used in composite structures. These connectors are executed in a similar manner to laminated steel profile. Accordingly, identical parameter PDF (e.g. for ultimate strength) are recommended.

4.7.1.2. Geometry

Geometric imperfections are due to deviations from specified values of the cross-sectional shape, the reinforcement position, the horizontality and verticality of concrete lines, and the alignment of columns, beams and surfaces of constructed structure [128]. Geometrical imperfections are related to quality control, during execution, and to permitted tolerances by codes. The following factors may affect the distribution of these imperfections: (1) structure type (e.g. slab bridge); (2) construction process and applied technology (e.g. precast); (3) execution quality (e.g. workmen).

According to the probabilistic model code [93] the deviations of a dimension X can be described by the statistical characteristics of its deviations Y from its nominal value X_{nom}, (4.28),

$$Y = X - X_{nom} \tag{4.28}$$

Accordingly, the deviations of the external dimensions of precast and cast in-situ concrete components, for nominal dimensions (X_{nom}) up to 1000 mm, can be modeled by a Normal PDF with mean and standard deviation value defined by (4.29),

$$0 \le \mu_y = 0.003 X_{nom} \le 3 [mm] \tag{4.29}$$

and (4.30),

$$\sigma_y = 4 + 0.006 X_{nom} \le 10 [mm] \tag{4.30}$$

Additionally, the deviations of concrete cover in reinforced concrete elements can be modeled by a Normal PDF with mean and standard deviation defined in Table 4.15.

The JCSS [93] proposal is a result of various studies, developed by several authors. Some of these studies are summarized in Wisniewski [196]. This author also compares precast and cast in-situ concrete geometric properties. In this situation, the absolute dimensions are studied, instead of deviation values. For Wisniewski [196], the precast concrete bridge girders and slabs height may be modeled by a Normal PDF with mean 0.5% higher than the nominal value and a standard deviation σ = 5 mm. The section width can be modeled by a Normal PDF with bias factor λ = 1.00 and standard deviation σ = 5 mm. Thickness may be assumed as normally distributed with bias factor λ = 1.05 and standard deviation σ = 10 mm. The cast in-situ reinforced concrete slabs height may be modeled by a Normal PDF with bias factor λ = 1.00 and standard deviation σ = 10 - 15 mm.

Table 4.15. PDF for concrete cover [93].

Parameter	Mean value	Standard deviation
	[mm]	[mm]
Column and wall	0 - 5	5 - 10
Slab bottom steel	0 - 10	5 - 10
Beam bottom steel	-10 – 0	5 - 10
Slab and beam top steel	0 - 10	10 - 15

In respect to steel section dimensions the probabilistic model given in JCSS [93] is based on the study of Fajkus *et al.* [54]. These authors presented some data on geometric deviations

of cross-sections of rolled products, according to expression (4.28). Preliminary results obtained for I profile (IPE 80 to 200) indicate that the mean and standard deviation of *Y* for basic dimensions (height, width and thickness) are less than 1 mm, (4.31),

$$-1.0mm \le \mu_y \le +1mm \tag{4.31}$$

and (4.32),

$$\sigma_y \le 1.0mm \tag{4.32}$$

For cross-section area it has been found that independently on the profile height, the mean value differs from the nominal value insignificantly. Accordingly, the obtained CV is about 3.20 %. The Normal PDF seems to be a fully satisfactory model for this property.

It has been found that external dimensions of concrete and steel sections are only dependent on production mode. No significant correlation has been established between vertical and horizontal dimensions. Furthermore, no data is available concerning the correlation of internal (e.g. concrete cover) and external dimensions.

Some of the geometric parameters, such as the steel profile top flange width and the connector geometry (e.g. stud shank diameter) are used to compute the interface parameters [50]. In this situation, it is recommended to use for the connector geometry a PDF identical to that indicated for the steel profile due to their similar execution process.

4.7.1.3. Physic

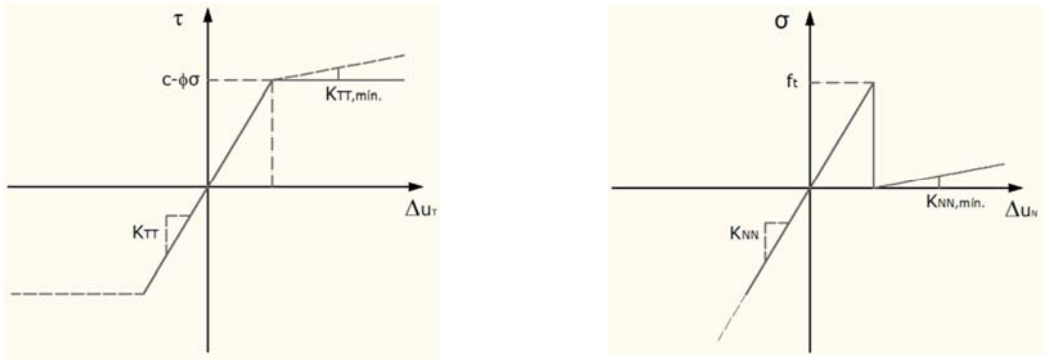
The physic properties incorporate all the model parameters which are not included in material and geometry properties. In this situation, the spring stiffness and the interface connection parameters are considered as physic properties. These parameters are more difficult to quantify as the existent information about them is scarce.

The spring stiffness is introduced to simulate the structural support behavior. Horizontal springs are used to represent the complex system bearing, column and foundation. In this situation, the PDF is defined by an engineering judgment procedure. However, once the information about this parameter is scarce, which makes the engineering judgment procedure very hard, another procedure is recommended. Accordingly, it is recommended to use in the probabilistic analysis the assessed value from model identification, instead of a PDF.

The interface connection parameters are introduced to simulate the steel–concrete connection in composite structures. Unfortunately, the information about these parameters is still scarce. An interesting work is that of Zona *et al.* [200] who proposed a methodology for

probabilistic nonlinear structural analysis of continuous steel-concrete composite girders. These authors presented PDF for various interface parameters.

Interface is characterized by normal and shear stresses. Figure 4.15 indicate these two models. The parameter PDF values for these models are computed through previous identified PDF for concrete and steel connector material, for steel profile and connector geometry and also for the interaction between these two materials [50]. However, while for material and geometric properties there are several sources of PDF values, for the interaction properties the existent information is scarce. This is especially due to the fact that these parameters are dependent on several factors, such as the connector type, used materials and geometry. A study which provided some information about this is that of Roik *et al.* [159]. This study presents the results of push-out tests with headed studs and different concrete materials. Accordingly, a Normal PDF with a CV of 30% is recommended for connection stiffness (*k*). The obtained results by Valente [188] within its study of headed stud connectors with light weight aggregate concrete indicate a Normal PDF with a CV of 20% for this property.



a) b)

Figure 4.15. Interface parameters: a) normal; b) shear stresses.

4.7.2. Bayesian inference

Bayesian inference is a framework that can be used to update the PDF when additional data is available. The objective is to reduce the statistical uncertainty. This procedure was detailed described at chapter three. Bayesian inference is incorporated in the probabilistic assessment algorithm. Therefore some Matlab® routines were developed, according to Figure 4.2b [25, 129]. These routines are provided in Appendix C, namely:

(1) *run.m*: run the Bayesian inference algorithm considering data.txt file, which contains measured values (likelihood), and winbugs_data.txt file, that provides some of WinBugs® model parameters [111] such as: (a) number of unknown parameters (mean, standard

deviation or both); (b) application of logarithmic conversion; (c) application of Weibull PDF; (d) number of samplings; (e) considered prior PDF (Jeffrey's or conjugate PDF); (f) prior PDF parameters (in case of an informative PDF) and respective weight (number of samplings);

(2) *read.m*: read data.txt file and winbugs_data.txt file and convert these values into legible Matlab® parameters [25, 129];

(3) *test.m*: validate the generated WinBugs® [111] model parameters;

(4) *winbugs.m*: run WinBugs® [111] for Normal or Lognormal PDF;

(5) *convert.m*: convert WinBugs® model parameters from Matlab® [25, 129] to WinBugs® [111];

(6) *calcbugs.m*: compute both likelihood mean and standard deviation values;

(7) *winbugsmu.m*, *winbugssigma.m*, *winbugsmusigma.m*: develop the Bayesian inference for the mean, the standard deviation or for both parameters. In each situation, the appropriate Bayesian model, written in BUGS language [111], is read;

(8) *matbugs.m* (by Kevin Murphy and Maryam Mahdaviani, August 2005: http://code.google.com/p/matbugs/): establish an interface between Matlab® [25, 129] and WinBugs® [111];

(9) *calcbugsmu.m*, *calcbugssigma.m*, *calcbugsmusigma.m*: process WinBugs® [111] results and compute the posterior PDF for each parameter and for the population;

(10) *plotbugsmu.m*, *plotbugssigma.m*, *plotbugsmusigma.m*: plot the prior, the likelihood and the posterior PDF for each parameter and for the population.

Additional Matlab® [25, 129] routines were developed to perform the Bayesian inference when a Weibull PDF is adopted. These routines, also indicated in Appendix C, are:

(1) *weibull.m*: run WinBugs® [111] when a Weibull PDF is chosen;

(2) *matrix_in.m*: read prior Weibull PDF parameter values;

(3) *wbl.m*: compute Weibull PDF parameters for measured values (likelihood);

(4) *matrix_up.m*: compute posterior PDF parameter values assuming a Weibull PDF;

(5) *plotbugswbl.m*: plot the prior, the likelihood and the posterior Weibull PDF.

Furthermore it is used the WinBugs® software [111], namely the executable file *winbugs14.exe*, which is a framework for constructing and analysing Bayesian full probability models through the Gibbs sampling (BUGS project). WinBugs® [111] processes the model specification and constructs an object-oriented representation of it. These models are

analysed by using Markov Chain Monte Carlo (MCMC) techniques. Developed models in WinBugs® [111], given at Appendix D, are: (1) unknown mean (μ) and known variance (σ^2) – *mu.txt*; (2) unknown mean (μ) and variance (σ^2) with Jeffrey's prior – *mu_tau_jeff.txt*; (3) unknown mean (μ) and variance (σ^2) with conjugate prior – *mu_tau_conj.txt*.

4.7.3. Simulation algorithms

Once the PDF is known and updated for all aleatoric variables of a structural problem, the use of simulation algorithms will permit to generate values for those variables, to compute the structural response and to evaluate each limit function when necessary. The failure probability (p_f) is the output of any limit function evaluation procedure. It may be described by the following integral (4.33),

$$p_f = \int_{g(x)\leq 0} I\left[g(X)\leq 0\right]\cdot f_x(X)\,dx \tag{4.33}$$

in which $f_x(X)$ is the PDF of variable X and I [g(X)≤0] is an indicator function, defined by (4.34),

$$I = \begin{cases} 1\ ;\ g(X)\leq 0\ (Failure) \\ 0\ ;\ g(X)>0\ (Safety) \end{cases} \tag{4.34}$$

Discrete integration techniques may be used within the sampling procedure to convert expression (4.33) into (4.35),

$$p_f \cong \tilde{p}_f = 1/N\cdot\sum_{i=1}^{n} I\left[g\left(X^{(i)}\right)\leq 0\right] \tag{4.35}$$

being N the number of samples and $X^{(i)}$ the vector of random variables for simulation i. Obtained results from sampling may be fitted to a cumulative distribution function, such as represented in Figure 4.16.

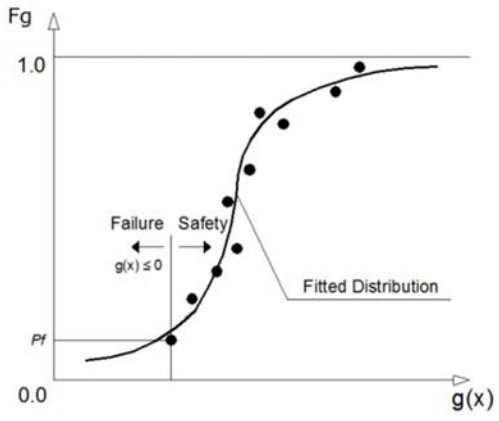


Figure 4.16. Curve fitting of cumulative distribution function.

The analysis of this figure permit to conclude that as the failure probability decreases, it also decreases the number of possible observations in interest region g(X)≤0, which means that the majority of observations will occur in safety region presenting thus no practical use. A way of increasing the number of simulations in such zone is by increasing the total number of samples. If the number of samples is infinite, obtained results from this method would be exact, (4.36),

$$p_f = P\left[g(X) \le 0\right] = \lim_{N \to \infty} \frac{N\left[g(X) \le 0\right]}{N} \tag{4.36}$$

Simulation algorithms have been widely used to assess the condition and safety of structures. They are especially important when complex analytic formulas are used. This is what happens in the majority of analyzed civil engineer structures. A technique that is worldwide used is the Monte Carlo (MC) simulation [33, 77, 134, 166]. The MC method is generally used to solve high complex problems in which computation is costly. This method involve the generation of samples according to a defined PDF and the use of obtained data to approach the interest function.

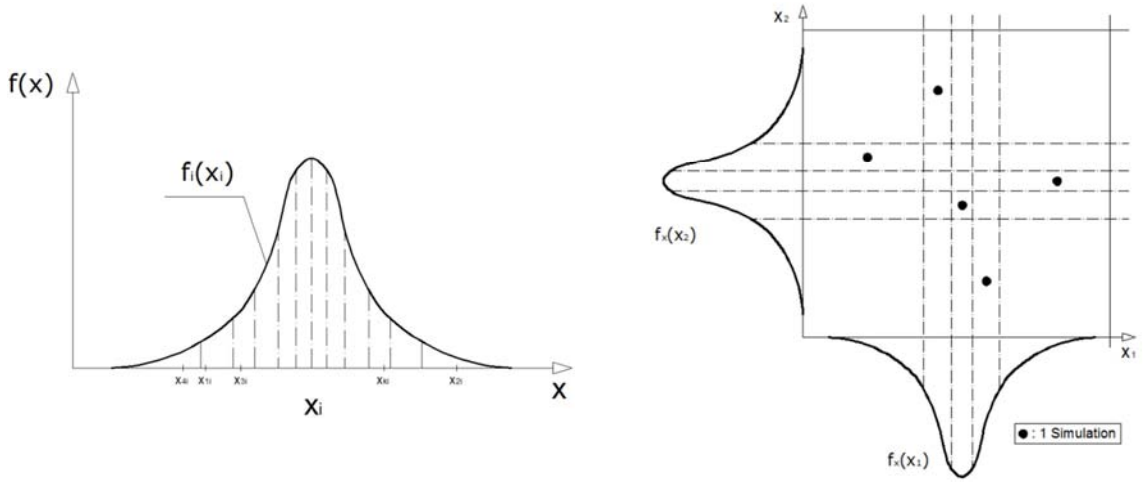
Accordingly, when applying MC techniques to determine the failure probability the following methodology should be used: (1) generate values for input parameters according to its PDF; (2) estimate the function g(X) with sampled values, through a deterministic approach; (3) verify if the limit state g(X)≤0 is attained; (4) count the number of simulations in which this limit is overpassed; (5) estimate the failure probability, through equation (4.35).

Simulation techniques as MC are computationally expensive due to the high number of simulations. Variance reduction techniques were thus introduced in order to diminish the number of samples. These techniques permit to significantly reduce the necessary simulations to compute a specific variance value or, in other words, they permit to reduce the variance values for a specific number of simulations.

One of these techniques is the importance sampling. Within this methodology the sampling values are concentrated in failure region. Therefore the sampling center is moved from its origin to the most probable failure point, which is located over the limit state region [42]. Enright and Frangopol [43] used an adaptive importance sampling algorithm to compute the failure probability.

The stratified sampling techniques are also variance reduction techniques. In such techniques, the integration region is divided into subregions in order to increase the number of simulations in subregions that present a high contribute to the failure probability. The most used stratified sampling algorithm is the LHS [77, 134, 144]. The main idea of this method is

to divide each input parameter domain X_i in equiprobable disjunctive intervals X_i^k, as presented in Figure 4.17a.



a) b)

Figure 4.17. Latin hypercube sampling (LHS): a) division in equal probability intervals; b) sampling scheme.

Each interval is characterized by a sample, defined according to the PDF. This method restricts the total number of samples, *N*, to the number of intervals which are considered in the division of the sampling region, *k*. These intervals are chosen in an aleatoric way, according to Figure 4.17b. The current practice is to directly choose samples by inverse transformation of the cumulative distribution function in the middle of k^{th} interval, through expression (4.37),

$$x_{i,k} = F_i^{-1}\left(\frac{k-0.5}{N}\right) \tag{4.37}$$

being $x_{i,k}$ the k^{th} sample of the i^{th} variable X_i and F_i^{-1} the inverse cumulative distribution function for variable X_i. Figure 4.18 illustrates the sampling procedure.

The LHS method is thus summarized in the following steps: (1) divide the domain of each aleatoric variable X_i into *k* equiprobable disjunctive regions; (2) generate *k* samples of each aleatoric variable, a sample per region; (3) generate *k* simulations with previous sampled values, being each value only used once and all the samples should be used; (4) evaluate the limit state function g(*X*), for each simulation, and register the number of times in which it is exceed; (5) Estimate the failure probability through expression (4.35).

The advantage of this method is that the number of simulations may be reduced maintaining the same numerical precision. It is verified that an acceptable precision is obtained if the

number of simulations is identical to the number of aleatoric variables, $N=X_i$, but very good results are obtained if $N=2*X_i$.

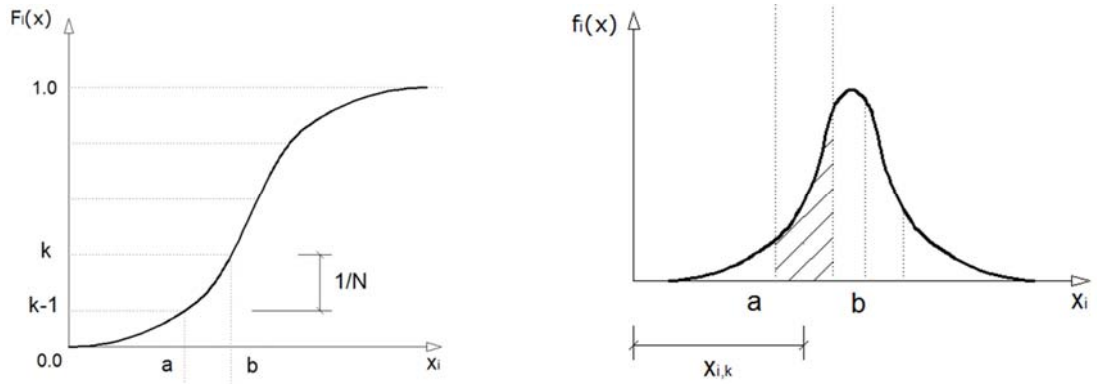


Figure 4.18. Illustration of sampling from marginal.

When sampling, the correlation structure should be maintained according to the target correlation matrix. There are generally two difficulties related to statistical correlation: (1) during sampling an undesired correlation may be introduced between random variables. It can happen, especially in the case of a very small number of simulations, where the number of combined intervals is limited; (2) introduce the prescribed statistical correlation between random variables, defined by the correlation matrix.

A technique for generation of correlated random variables has been proposed by Iman and Conover [85]. An application of this technique in a simple example is developed by Mildenhall [126]. This technique is simple to use, is distribution free, preserves the exact form of the marginal PDF in input variables, and may be used with any type of sampling scheme for which the correlation between input variables is a meaningful concept.

This technique is based on the iterative updating of the sampled matrix, through the Cholesky decomposition of the correlation matrix [85]. As a measure of statistical correlation, the Spearman correlation coefficient (T_{ij}) between the variables i and j, $T_{ij} \in [-1;1]$ [33, 123] is used, (4.38),

$$T_{ij} = 1 - \frac{6 \cdot \sum_k \left(R_{ki} - R_{kj}\right)^2}{N \cdot (N-1) \cdot (N+1)} \tag{4.38}$$

being R the matrix containing a permutation of the rank numbers in each column. The ranking numbers in each column of matrix R are then rearranged to have the same ordering as the numbers in each column of the target correlation matrix. The technique can be applied iteratively and can result in a very low correlation coefficient if uncorrelated random variables are generated. However, Huntington and Lyrintzis [84] found that the approach tends to

converge to an ordering which still gives significant correlation errors between some variables.

The described scheme is linked to the Spearman correlation measure which is very robust in cases of non-Gaussian PDF. Additionally, it only uses ranks, instead of sampled values. Its limitation is related to the fact of the necessary number of simulations being higher than the number of random variables to achieve a positive definite correlation matrix. It can be understood as a serious drawback in cases where a very high number of variables exist and a limited number of simulations are possible.

These obstacles stimulated the work of Huntington and Lyrintzis [84] who proposed the single-switch-optimized ordering scheme. This approach consists in the imposition of a target correlation structure by matrix manipulations only. However, this imposition may be understood as an optimization problem for which the difference between prescribed and generated correlation matrices should be as small as possible. Vorechovský and Novák [192, 193, 194] propose a new efficient technique to impose the statistical correlation when using LHS algorithms. This technique is based in a stochastic optimization method called simulated annealing. It is robust, efficient, fast and performs well with a large number of variables. An application of this technique is given in Bergmeister *et al.* [14].

Within this probabilistic assessment a LHS algorithm is used to sample numerical models according to updated model parameter PDF. This algorithm was implemented in Matlab® by Stein [173] and presented at Appendix B. In this situation, it will be generated 500 samples, which is higher than twice the number of aleatoric variables. The correlation coefficients, introduced by means of a correlation matrix, are kept during simulation through the application of the Iman and Conover [85] algorithm. This algorithm is also provided at Appendix B.

4.8. Structural performance indexes

During probabilistic assessment algorithm, values are generated for input parameters, and, accordingly, for each sample, the model is analyzed and obtained results are statistically processed, being obtained a numerical PDF. Performance indexes are then used to evaluate the structural behavior. In this situation, they are classified in evaluation and safety assessment indexes. While the former is related to the evaluation of the structural behavior, both in service and failure region, through a comparison with obtained experimental data, the latter is related to the failure limit state, through a comparison with loading PDF.

4.8.1. Evaluation assessment

The evaluation indexes are mostly used when it is necessary to verify if the structural behavior is, or not, within the expectable during a performance test. In this situation, it is introduced an index-*i*, used to evaluate the analyzed structure behavior within the whole test, and an index-*p* [118, 119, 120, 122], that permits to probabilistically evaluate such behavior in a time frame of the test.

The index-*i* consists in evaluating if the measured value during a test falls into the computed 95% confidence interval for an output parameter, and according to Figure 4.19a, (4.39),

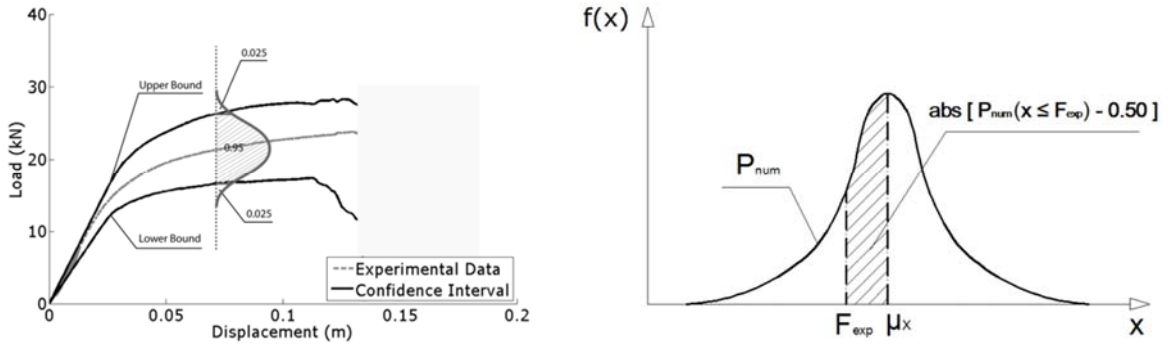
$$\text{index-i} = n_i / n_t \,[\%] \tag{4.39}$$

being n_i the number of measured values that falls into it and n_t the total number of registered values. This value increases with the number of experimental registers that fall into predicted numerical confidence interval.

The index-*p* evaluates the approximation of experimental data to computed numerical PDF, through equation (4.40),

$$\text{index-p} = \left[1 - \left(\text{abs}\left[P_{num}\left(X \le F_{\text{exp}}\right) - 0.5\right] / 0.5\right) \cdot CV_{num}\right][\%] \tag{4.40}$$

being F_{exp} the measured data, P_{num} the obtained numerical PDF, $X \le F_{exp}$ indicates the probability of X being lower than or equal to F_{exp} considering the numerical PDF, 0.5 is the obtained probability when F_{exp} is equal to the mean value of the numerical PDF, and CV_{num} is the coefficient of variation of numerical PDF. These parameters are given at Figure 4.19b. The highest value of index-*p* corresponds to the most reliable numerical PDF.



a) b)

Figure 4.19. Evaluation assessment: a) index-*i*; b) index-*p*.

4.8.2. Safety assessment

The reliability index is a performance index, used to quantify the structure safety. It is more objective and explicitly than traditional visual-inspection based indexes. Therefore, it is a more impartial mean to assess the structure performance, and thus it is recommended to be used in maintenance and management procedures [59].

In safety assessment, a comparison between resistance (R) and loading (S) PDF is performed [33, 134]. According to Figure 4.20a, the failure probability (p_f) is proportional to the intersection area of these PDF, which corresponds to the case in which the structural resistance is lower than the applied load, (4.41),

$$p_f = P\left(R \leq S\right) \tag{4.41}$$

Or in other words, (4.42),

$$p_f = P\left(R - S \leq 0\right) \tag{4.42}$$

In this situation the indicator function g(X) is the limit function ($Z(R,S) = R - S$). Therefore, it is possible to express (4.41) through (4.43),

$$p_f = P\left(Z\left(R,S\right) \leq 0\right) \tag{4.43}$$

In this situation, the failure probability is computed, and according to expression (4.35), through (4.44),

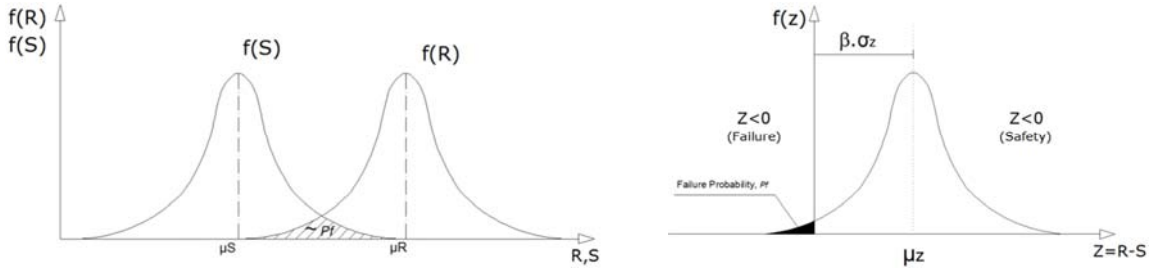
$$P_f = \frac{N\left[Z\left(R,S\right) \leq 0\right]}{N} = \frac{n_f}{n_t}[\%] \tag{4.44}$$

being n_f the number of sampled models which fall into failure region and n_t the total number of sampled models. The reliability index (β) is then given by expression (4.45),

$$\beta = -\Phi^{-1}\left(P_f\right) \tag{4.45}$$

being Φ^{-1} the inverse cumulative distribution function for a standard Normal PDF. Figure 4.20b presents the limit function PDF and its relation to the reliability index.

The structural safety assessment [93, 134, 166] consists in computing the obtained reliability index and comparing it to a target reliability index (β_{target}). The target reliability values have to be established based on the reliability analysis of many structures. Therefore, the nominal target values have to take into account several factors as the failure type, associated risk and costs, etc.



a) b)

Figure 4.20. Safety assessment: a) resistance and loading; b) failure probability.

It is important to divide these indexes into those that are used in a structural element analysis and those that are used in a global structure analysis. As target reliabilities for member level it is proposed the values of AASHTO LRFD [1] and LRFR [2], of EN-1990 [46], of ISO 13822 [86], of JCSS [93], and of Schneider [166].

In respect to ISO 13822 [86], the target reliability level used for verification of an existing structure can be determined based on calibration with current code, the concept of minimum total expected cost, and/or the comparison with other social risks. The requirements should also reflect the type and importance of the structure, possible failure consequences and socio-economic criteria.

Although the performance requirements on safety and serviceability for the assessment of existing structures are in principle the same as for the design of new structures, there are some fundamental differences between these two activities affecting the differentiation in structural reliability, namely: (1) economic considerations: the incremental cost between acceptance and upgrading the existing structure can be very large, whereas the incremental cost of increasing the safety of a structural design is generally very small; (2) social considerations: this include disruption of occupants and activities, considerations that do not affect the structural design of new structures; (3) sustainability considerations: reduction of waste and recycling, which are less important considerations in the design of new structures.

The remaining working life determined at the assessment is considered as a reference period for an existing structure for serviceability, while the design working life is often considered as a reference period for a new structure. The target reliability indexes may be chosen in accordance with current codes, if provided, otherwise the values given in Table 4.16 are intended as illustrations for the assessment of existing structures.

Since Eurocodes are legal codes in most of European countries, the reliability levels stated there, especially in EN-1990 [46], could give some idea of required safety of bridges in Europe. Two sets of indexes are presented for different reference periods (1 and 50 years).

The reliability classes used for ultimate limit state are RC-1, RC-2 and RC-3 corresponding to low, medium and high failure consequences, related to loss of human life and economic, social and environmental costs. These values might be over conservative for existing structures as they are mainly proposed for bridge design. Table 4.17 indicates those values.

Table 4.16. Target reliability indexes (β_{target}) for structures [86].

Limit states	Target reliability index	Reference period
Serviceability		
Reversible	0.00	Intended remaining working life
Irreversible	1.50	Intended remaining working life
Ultimate		
Very low consequences of failure	2.30	L_S years *
Low consequences of failure	3.10	L_S years *
Medium consequences of failure	3.80	L_S years *
High consequences of failure	4.30	L_S years *

* L_S is the minimum safety standard period (e.g. 50 years).

Table 4.17. Target reliability indexes (β_{target}) for structures [46].

Limit state	Reference period 1 year			Reference period 50 years		
	RC-1	RC-2	RC-3	RC-1	RC-2	RC-3
Ultimate	4.20	4.70	5.20	3.30	3.80	4.30
Serviceability	-	2.90	-	-	1.50	-

In respect to JCSS [93] the target reliability values for ultimate limit states are based on cost benefit analysis and are compatible with calibration studies and statistical observations. The values proposed by JCSS [93] are indicated at Table 4.18.

The consequence class classification is based in the ratio ρ between total costs (i.e. construction costs and failure costs) and construction costs. Accordingly, there are: (1) class 1: minor consequences, ρ is less than 2 (risk to life, given a failure, is small to negligible and economic consequences are small or negligible; e.g. agricultural structures, silos, masts, etc.); (2) class 2: moderate consequences, ρ is between 2 and 5 (risk to life, given a failure, is medium or economic consequences are considerable; e.g. office buildings, industrial buildings, apartment buildings, etc.); (3) class 3: large consequences, ρ is between 5 and 10 (risk to life, given a failure, is high or economic consequences are significant;

e.g. main bridges, theaters, hospitals, high rise buildings, etc.). Failure consequences also depend on failure type, which can be classified according to: (1) ductile failure with reserve strength capacity, resulting from strain hardening; (2) ductile failure with no reserve capacity; (3) brittle failure.

Table 4.18. Target reliability indexes (β_{target}) for structures [93].

Relative cost of safety measure	Ultimate limit state			Serviceability limit state
	Minor consequences of failure	Moderate consequences of failure	Large consequences of failure	
Large	3.10 ($p_f \approx 10^{-3}$)	3.30 ($p_f \approx 5*10^{-4}$)	3.70 ($p_f \approx 10^{-4}$)	1.30 ($p_f \approx 10^{-1}$)
Normal	3.70 ($p_f \approx 10^{-4}$)	4.20 ($p_f \approx 10^{-5}$)	4.40 ($p_f \approx 5 * 10^{-6}$)	1.70 ($p_f \approx 5 * 10^{-2}$)
Small	4.20 ($p_f \approx 10^{-5}$)	4.40 ($p_f \approx 5 * 10^{-6}$)	4.70 ($p_f \approx 10^{-6}$)	2.30 ($p_f \approx 10^{-2}$)

The minimum reliability index, $\beta_{target} = 3.50$, required for the design of bridges in USA, defined at AASHTO LRFD [1], is based on average reliability index values computed from a sample of past design. This value corresponds to the reliability of an individual member and strength limit states. It was computed considering specific probabilistic models for loads, geometry and mechanical properties of materials. Therefore, if different probabilistic models were used, different target reliability would be obtained for the same sample of past design. The code AASHTO LRFR [2] proposes a value of target reliability index for the strength assessment of bridge members, $\beta_{target} = 2.50$. This value is lower than the previous due to the fact that evaluation is performed for a much shorter exposure period.

Schneider [166] proposes the following Table 4.19 as a rough indication of the order of magnitude of reliability indexes per year. Reliability indexes concerning lifetime are smaller because acceptable failure probabilities are roughly annual failure probabilities times the expected lifetime of the structure. Reliability requirements regarding the structure serviceability are less severe than those related to structural safety.

The consequences of a possible failure as well as the type of failure must be considered. Therefore, the following consequences are provided: (1) class 1: almost no consequences, small transient disturbances for users, serviceability affected during short periods; (2) class 2: minor consequences, no hazards to life and limb, economic consequences of failure are low; (3) class 3: moderate consequences, hazards to life and limb are low, economic consequences of failure are considerable; (4) class 4: large consequences, hazards to life and limb are medium, economic consequences of failure are considerable; (5) class 5: extreme consequences, hazards to life and limb are high, economic consequences

of failure are very high. Moreover, the following failure types are given: (1) type A: serviceability failure, structure almost in elastic domain; (2) type B: ductile failure of redundant systems with reserve strength; (3) type C: ductile failure, but almost no reserve strength; (3) type D: brittle failure of non-redundant systems.

Table 4.19. Target reliability indexes (β_{target}) for structures [166].

		Type of failure			
		Type A	Type B	Type C	Type D
Consequences	Class 1	1.00	1.50	2.00	2.50
	Class 2	1.50	2.00	2.50	3.00
	Class 3	2.00	2.50	3.50	4.00
	Class 4	2.50	3.00	4.50	5.00
	Class 5	3.00	4.00	5.00	6.00

In respect to target reliabilities for system level it is proposed the models of *fib* [55] and of Tabsh and Nowak [180]. According to JCSS [93] the previous values, given in Table 4.18, relate to the dominant failure mode of the structural system or to the structural component dominating system failure. Therefore, a structure with multiple and equally important failure modes should be designed for a higher level of reliability.

Therefore, the reliability of a bridge, when considered as a structural system, is usually higher than the reliability of its members. According to Tabsh and Nowak [180] the difference between common values for those indexes in USA bridges is about 2, i.e., instead of using β=3-4, for bridge member, for the whole system a value of β=5-6 is recommended. Traditional design or assessment methods did not consider this additional structural capacity in a quantitative manner, but it is known that this additional over-strength exists and that member requirements can be chosen in a less conservative manner.

Nowadays the assessment can be performed at both member and system level. Therefore, and in order to introduce consistency between these two levels, target reliability indexes for system level should be higher than those for member level. So far, and unfortunately, there is no code or guideline that clearly defines target reliability indexes for assessment of structures at system level. Many codes are member orientated and the effort in last decades is to develop target safety for design and evaluation of structural members without accounting for redundancy and system effect. Two important works in this respect are from Ghosn and Moses [64] and from Liu *et al.* [108]. Other interesting work is that indicated at *fib* [55]. These

authors proposed five bridge reliability states. Table 4.20 indicate these states and related reliability indexes.

Table 4.20. Bridge reliability states [55].

Reliability states				
5	4	3	2	1
Reliability index				
$\beta \geq 9.00$	$9.00 > \beta \geq 8.00$	$8.00 > \beta \geq 6.00$	$6.00 > \beta \geq 4.60$	$4.60 > \beta$
Attribute for reliability				
Excellent	Very good	Good	Fair	Unacceptable

4.9. Conclusions

A methodology for probabilistic assessment of structures is presented in this chapter. This methodology permits to evaluate the structural safety and condition. It is composed by a two main steps. In a first step the numerical model is updated through a model identification procedure. Further, the updated deterministic model is transformed into a probabilistic model and a probabilistic analysis is developed. Finally, each parameter PDF may be updated with complementary data, through a Bayesian inference algorithm.

This methodology contemplates all sources of uncertainty. Modeling and measurement errors are introduced when computing the model identification tolerance criterion. Physical uncertainties are incorporated into model parameter PDF. Statistical uncertainty may be reduced with the application of the Bayesian inference algorithm. Human errors may be identified during the probabilistic assessment procedure.

The developed algorithm presents a high computational cost. In order to minimize it, a sensitivity analysis, in which the most important parameters are selected, should be applied. The implemented procedures of such analysis are introduced. This procedure is highly dependent on the defined target importance measure value. If any parameter presents a value that is higher than this value, then it should be considered as critical.

A study is developed to evaluate measurement errors, specifically in LVDT. Accordingly, the linearity and sensitivity of different transducers, with different cable lengths, is evaluated. This study permits to conclude that the system transducer and cable should be always calibrated before being implemented in a monitoring system, as it reduces in large scale the linearity value and so the measurement error. Additionally, it is verified that long cables are

not recommended as its linearity value increases in a way that makes the assessment procedure inaccurate.

Modeling errors, and the way they are computed, are also indicated. Additionally it is shown how these errors should be combined with measurement errors, in order to obtain the model identification fitness function tolerance criterion. Once a pool of models is obtained from model identification, then an engineering judgment procedure, based in the probability of occurrence of each model, is performed to select the most suitable model.

Important PDF for material, geometry and physic properties are further defined. Some of these parameters are well defined in bibliography, but others, as those related to lightweight concrete or to steel-concrete interface, are not. Therefore, in this situation, the PDF definition is based in experience. When defining the PDF parameters (e.g. mean and standard deviation), a statistical uncertainty is introduced. Accordingly, this uncertainty can be reduced with the acquisition of new data. Bayesian methods are thus applied to include information from complementary data into these models.

It is also presented some simulation algorithms, necessary to sample the numerical models which will be analyzed within the probabilistic assessment. A special emphasis is given to the LHS method and to Iman and Conover [85] algorithm, necessary to implement the correlation matrix. Performance indexes that provide information about the structure behavior are also provided. Two of these indexes are related to the evaluation of the structural behavior, both in service and failure region, while the other, the reliability index, gives information on structural safety. Some target values for this latter index are also given.

5 Reinforced Concrete Beams

5.1. Introduction

This chapter presents the probabilistic study on the behavior of two sets of reinforced concrete beams, a pinned-pinned and other pinned-fixed supported, which were tested in laboratory up to failure. The purpose of such analysis is to probabilistically assess the behavior of reinforced concrete structures, by taking into consideration all sources of uncertainty.

In a first step, a nonlinear deterministic numerical model was developed and simplified, by reducing both the finite elements and load step increment. Further, a sensitivity analysis was performed, in order to study the influence of each variable on the structural behavior. The most relevant parameters were then selected for further analysis. These procedures are important as they reduce the computational cost of a further probabilistic analysis.

In a second step, model identification was executed. The deterministic numerical model is therefore updated with obtained experimental data by using a robust optimization technique and taking into account both measurement and modeling errors. Parameters as material, geometry and physic ones were respectively assessed. For pinned-fixed beams, the spring constant, used to simulate the fixed support, was also assessed. This procedure is important as it permit to obtain an updated deterministic numerical model of evaluated structure, to be used in a further probabilistic analysis.

In a third step, a nonlinear probabilistic analysis was developed. Therefore, it was necessary to define, according to bibliography, each input parameter probability density function (PDF). For pinned-fixed beams those PDF were later updated through a Bayesian inference approach, by considering the results from laboratory characterization tests. Such tests were developed to characterize both material and geometry properties. An updated resistance PDF is then obtained, necessary for a safety assessment procedure.

Different resistance PDF of reinforced concrete beams were then computed for different levels of updating, specifically, considering or not model identification and Bayesian inference procedures. Obtained results were then compared to measured data. The influence of both model identification and Bayesian inference procedures is pointed out on a safety assessment example.

5.2. Experimental tests

5.2.1. Pinned-pinned beams

The first set of tested structures is composed by two pinned-pinned supported beams. Such beams, with a rectangular section of 75 x 150 mm^2 and 1.50 m span length, were concreted at once. The scheme of these tests is presented in Figure 5.1a. Loads were applied by an actuator with 150 kN capacity and were positioned at 1/3 and 2/3 of the span length. Each beam was supported in two elements, placed symmetrically in relation to their symmetric axle. While one of those elements restricts only the vertical displacement, the other restricts both vertical and horizontal one [114, 115, 116, 117].

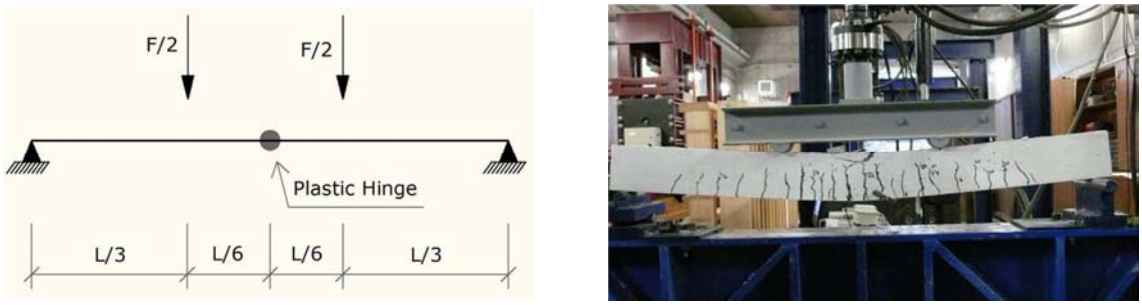
Tested beams present the same concrete class (C25/30) and longitudinal steel reinforcement type (S500B), according to EN 1992-1-1 [48]. A longitudinal reinforcement of 3ϕ6 (A_s = 0.85 cm^2) and a transversal reinforcement of ϕ4@0.10 (A_{sw}/s = 2.51 cm^2/m) was used. The concrete cover was considered to be 10 mm in all sides.

The laboratory test was developed with displacement control, by using a displacement transducer positioned inside the actuator. The applied load was registered through a load cell positioned inside the actuator. The middle span displacement was monitored through a displacement transducer (LVDT), with a measurement field of ± 25.00 mm, a sensitivity of 34.00 mV/V/mm and an accuracy of 0.10%. The repeatability was reached by using the same materials and providing identical laboratory conditions during the developed tests. Figure 5.1b presents an image of those tests.

a) b)

Figure 5.1. Test: a) scheme; b) laboratory (top right: actuator; bottom right: LVDT).

The collapse mechanism, presented in Figure 5.2a, is characterized by a plastic hinge located at middle span. Figure 5.2b shows the collapse mechanism. Obtained experimental results are given in Figure 5.3. The structural behavior is divided in three steps: (1) Pre-cracking phase, before concrete cracking, in which steel did not reach the yield strength, and concrete did not reach both tensile and compressive strength; (2) Post-cracking phase in which the concrete, submitted to tensile stresses, already reached the tensile strength, but both steel and concrete did not reach their yield and compressive strength; (3) A final phase in which the concrete, in compression, and the steel, in tension, already reached the compressive and yield strength. A bending failure mode with concrete crushing and yielding of longitudinal steel reinforcement was obtained. The failure load (F_R) is of 24.61 kN (beam 1) and of 24.97 kN (beam 2).



a) b)

Figure 5.2. Collapse mechanism: a) scheme b) image.

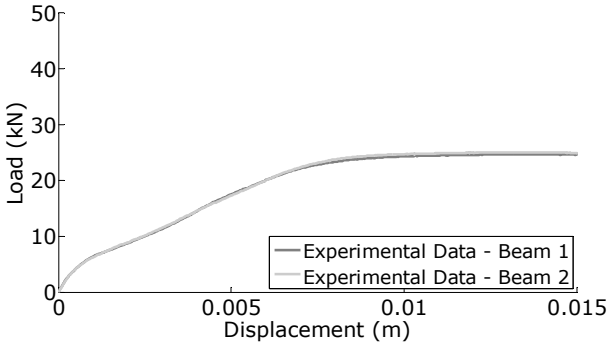


Figure 5.3. Experimental data.

5.2.2. Pinned-fixed beams

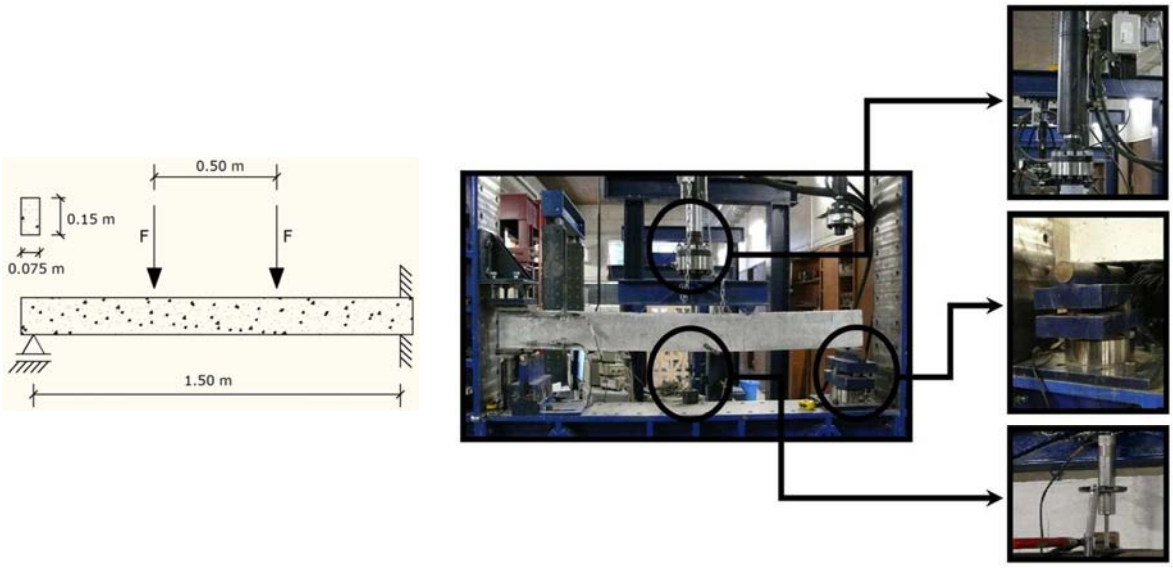
The second set comprises two tested pinned-fixed supported beams. Those beams, with a rectangular section of 75 x 150 mm^2 and a free span of 1.50 m, were concreted at once. Figure 5.4a presents a scheme of developed laboratory tests. Loads were applied by an actuator with 150 kN capacity, and were positioned at 1/3 and 2/3 of the span length. Those beams were supported in two elements, a pinned and a fixed support. While one of those elements restricts only the vertical displacement, the other restricts both the vertical and the horizontal one and, also, the bending moment. In this case there is no symmetry and the structure is one degree hyperstatic [115, 116, 117, 119, 121, 122].

Tested beams present the same concrete class (C25/30) and longitudinal steel reinforcement type (S500B), according to EN 1992-1-1 [48]. It was respectively considered a superior and an inferior longitudinal reinforcement of 2ϕ8 (A_s = 1.01 cm^2) and of 3ϕ6 (A_s = 0.85 cm^2). It was adopted a transversal reinforcement of ϕ4@0.03 (A_{sw}/s = 8.38 cm^2/m), in supports region, and of ϕ4@0.08 (A_{sw}/s = 3.14 cm^2/m), at middle span. The concrete cover was considered to be 20 mm in all sides.

The laboratory test was developed with displacement control, using a displacement transducer positioned inside the actuator. The applied load was registered through a load cell positioned inside the actuator. The middle span displacement was controlled by a displacement transducer (LVDT), with a measurement field of ± 25.00 mm, a sensitivity of 33.84 mV/V/mm and an accuracy of 0.10%. The pinned support reaction was registered through a load cell with 200 kN of capacity and 0.10% of accuracy. The repeatability was achieved, by using the same materials and providing the same laboratory conditions during the developed tests. Figure 5.4b shows those tests.

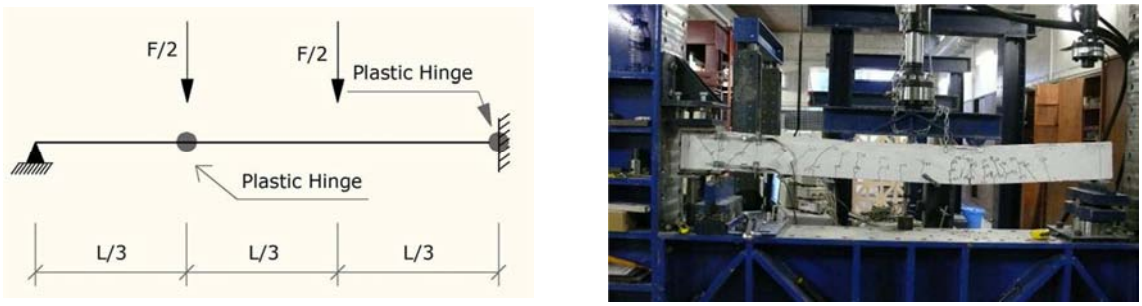
Due to the fact of being one degree hyperstatic the collapse mechanism is characterized by two plastic hinges, Figure 5.5a, located at fixed support and beside the point load that is

close to the pinned support. Figure 5.5b shows the collapse mechanism. As the applied load and the pinned support reaction were measured, it was possible to obtain, in an indirect way, the bending moment at fixed support, by using the static equilibrium equations.



a) b)

Figure 5.4. Test: a) scheme; b) laboratory (top right: actuator; center right: load cell; bottom right: LVDT).

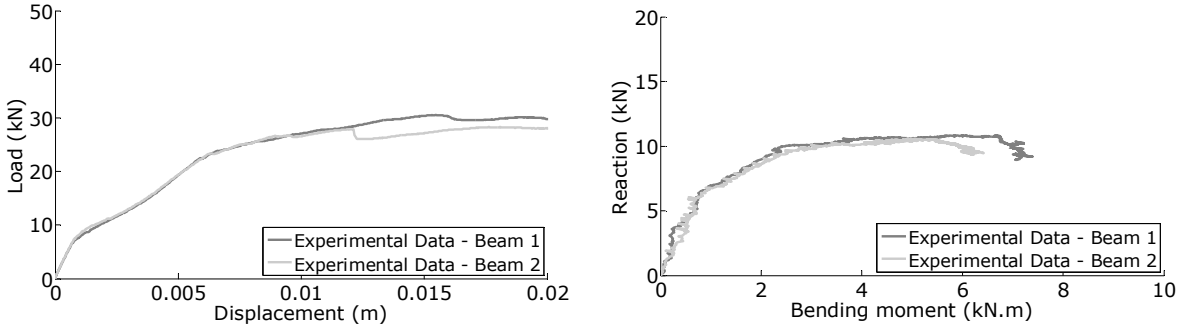


a) b)

Figure 5.5. Collapse mechanism: a) scheme b) image.

Obtained experimental results are given in Figure 5.6. The structural behavior is divided into five steps: (1) Pre-cracking phase, before concrete cracking, in which steel did not reach the yield strength, and concrete did not reach both tensile and compressive strength and in which a partial restraint is verified at fixed support; (2) Post-cracking phase in which the concrete, submitted to tensile stresses, already reached the tensile strength; (3) Post-cracking phase in which a full restraint is observed at fixed support; (4) First hinge phase in which the concrete, in compression, and the steel, in tension, respectively reached the compressive and yield strength at fixed support; (5) Second hinge phase in which the

concrete, in compression, and the steel, in tension, respectively reached the compressive and yield strength in the section beside the point load that is close to the pinned support. Obtained failure mode was of bending with concrete crushing and yielding of longitudinal steel reinforcement. It is also verified that the fixed support does not work as a full clamp since the beginning of loading, due to concrete accommodation. The failure load (F_R) and maximum bending moment (M_{R1}^*) is of 30.52 kN and 7.38 kN.m (beam 1) and of 28.26 kN and 6.43 kN.m (beam 2).



a) b)

Figure 5.6. Experimental data.

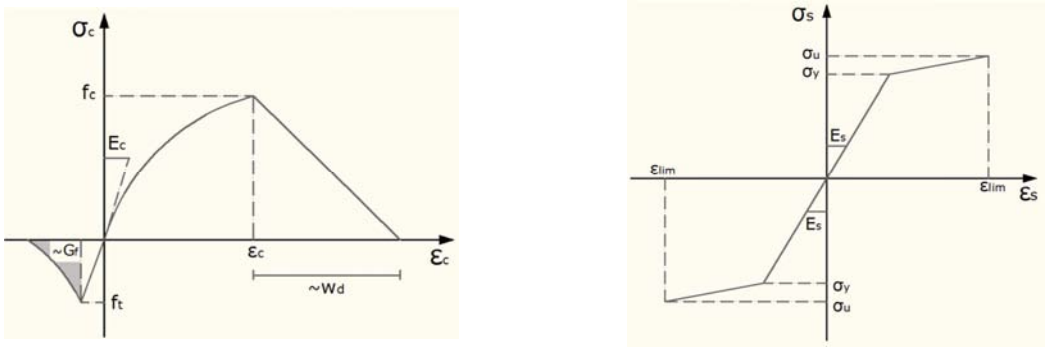
5.3. Numerical analysis

Tested reinforced concrete beams were numerically studied through a nonlinear finite element model developed with software ATENA® [23, 24]. Used materials, concrete and steel, were defined by a set of parameters that define the respective constitutive law.

The stress-strain law for concrete is presented in Figure 5.7a. This law is defined by some parameters as: (1) elasticity modulus (E_c); (2) compressive strength (f_c); (3) compressive strain at compressive strength (ε_c); and (4) tensile strength (f_t). This material presents a different behavior when submitted to compressive and tensile stresses. In compression it is characterized by an initial parabolic phase until compressive strength and then by a linear phase (softening). The critical displacement parameter (w_d) is used to define the softening phase. When submitted to tension stresses, the material behavior is characterized by an initial linear phase until tensile strength and then by an exponential phase. The fracture energy (G_f) is proportional to the area of this diagram.

The uniaxial stress-strain law for steel is presented in Figure 5.7b. This law is defined by some parameters as: (1) elasticity modulus (E_s); (2) yield strength (σ_y); (3) limit strain (ε_{lim}); and (4) limit strength (σ_u). This material presents a bi-linear behavior, characterized by an initial phase in which the material presents a typical elastic behavior and a second phase

from material yielding until failure (hardening). A similar behavior is obtained in compression and in tension.



a) b)

Figure 5.7. Material stress-strain law: a) concrete; b) reinforcement steel.

Those materials, concrete and steel, were modeled by using, respectively, an SBETA material and a bilinear with hardening Von Mises model, which are constitutive models of the ATENA® library [23, 24]. A biaxial stress failure criterion and a Von Mises yield criterion are respectively established. The parameter values were taken from EN 1992-1-1 [48]. Table 5.1 and 5.2 indicate those values.

Table 5.1. Material properties (concrete).

Parameter		Value
Elasticity modulus (E_c)	[GPa]	31.00
Tensile strength (f_t)	[MPa]	2.60
Compressive strength (f_c)	[MPa]	33.00
Fracture energy (G_f) [24]	[N/m]	65.00
Compressive strain at compressive strength (ε_c)	[‰]	2.00
Critical compressive displacement (w_d) [24]	[m]	$5.00 * 10^{-4}$

Table 5.2. Material properties (steel).

Parameter		Value
Elasticity modulus (Es)	[GPa]	200.00
Yield strength (σ_y)	[MPa]	500.00
Limit strength (σ_u)	[MPa]	540.00 (k = 1.08)
Limit strain (ε_{lim})	[‰]	50.00

5.3.1. Pinned-pinned beams

The analyzed beams present a rectangular section of 75 x 150 mm^2 and a span length of 1.50m. Such beams are simply supported. While one of the supports restricts only the vertical displacement, the other restricts both the vertical and the horizontal one. A uniform mesh of quadrilateral elements was considered. Reinforcing steel bar elements were considered to be embedded in concrete.

In order to correctly simulate the developed laboratory test a displacement control numerical test was used. Two different load cases were adopted, respectively, one representing the real supports and other the applied displacement. One of the used supports restraint both horizontal and vertical displacements, while the other only restraint the vertical ones.

Two point loads were introduced at 1/3 and at 2/3 of the span length. An identical increment was considered for the applied displacement in each point load, equal to $1.00 * 10^{-4}$ m (downward). In order to avoid high local stresses in both support and load points, a steel plate was placed in such positions. A Newton-Raphson nonlinear search algorithm was used. Considered parameters are given in Table 5.3. The middle span displacement and applied load were monitored during the analysis [114, 116, 117].

Table 5.3. Solution parameters (Newton-Raphson).

Solution method	Newton-Raphson
Stiffness / Update	Tangent / Each iteration
Iterations number limit	100
Error tolerance	$1.00 * 10^{-2}$
Line search	With iterations

When performing several analysis of the same numerical model, as in model identification or within a probabilistic analysis, the issue of computational cost becomes very important. In order to overcome it, the developed numerical model was simplified. Therefore, both finite element and load step numbers were minimized. Accordingly, three mesh types (with 708, 432 and 325 elements) and two different load steps (175 steps of factor 1 and 30 steps of factor 5) were studied. Figure 5.8a presents a finite element mesh with 432 elements.

In a further analysis, the performance of each model was evaluated. In order to assure identical computational conditions, the same computer was used. The computational time and the related error were determined for each analysis. The applied load error was computed through equation (5.1),

$$\Delta^i = \left(F_1^i - F_0^i\right) / F_0^i \,[\%] \tag{5.1}$$

in which F_1^i is the applied load for a specific step *i* of evaluated numerical model and F_0^i is the applied load for the same step *i* of the reference numerical model. The reference model is the one that presents the most refined mesh and higher number of load steps. Afterwards, the maximum and minimum Δ-values are determined and the sum of their absolute values is computed. Finally, the applied load error (θ) is computed by dividing this value per two. Table 5.4 shows the obtained results. It is possible to conclude that the most suitable model, for a further analysis, is the number 3. In fact, by comparing with other models, it is possible to conclude that this model presents a lower error (≈5%) and, at same time, by comparing with reference model (number 0) it reduces the computational cost in almost 90%.

Table 5.4. Simplification results.

Numerical model	Finite element number	Step number	Computational time [s]	Applied load error - θ [%]
0	708	175	151.07	-
1	708	30	40.16	2.49
2	432	175	126.49	5.05
3	432	30	18.71	5.06
4	325	175	94.46	8.36
5	325	30	15.88	10.03

Figure 5.8b presents the deformation, crack pattern and horizontal strain of the analyzed beam for chosen numerical model. In this case the collapse mechanism is characterized by the presence of a plastic hinge at the beam middle span. A bending failure mode, with concrete crushing and yielding of longitudinal steel reinforcement, is obtained. The numerical behavior was similar to the experimental one, validating the developed model.

a) b)

Figure 5.8. Numerical model: a) finite element mesh; b) failure mechanism.

5.3.2. Pinned-fixed beams

These beams present a rectangular section of 75 x 150 mm^2 and a span length of 1.50 m. They are supported in two points, a pinned and a fixed one. While one of those elements

restricts only the vertical displacement, the other restricts both the vertical and the horizontal one along a line. Vertical spring elements were introduced along the support line, in this latter case, to study the related restraint to this degree of freedom. Such springs would only work for vertical displacements and when in compression. In order to avoid high local stresses, a steel plate was placed between the structure and both structural supports. A uniform mesh of quadrilateral elements was considered. Reinforcing steel bar elements were considered to be embedded in concrete.

In order to correctly simulate the laboratory test, some considerations were taken. A displacement control numerical test was adopted. Three load cases were considered, one representing the real supports with spring elements placed at fixed support, other representing real supports with a vertical displacement restraint at fixed support and other representing the applied displacement. The load is applied at middle span of a steel profile which will load the beam in two points, respectively, at 1/3 and at 2/3 of the span length. An identical increment was considered for the applied displacement in each point load, equal to $1.00 * 10^{-4}$ m (downward). A Newton-Raphson nonlinear search algorithm was used. Considered parameters are given in Table 5.3. The middle span displacement, the applied load and the pinned support reaction were monitored during the analysis. The bending moment at fixed support was computed through static equilibrium equations [116, 117, 119, 121, 122].

When performing several analysis of the same numerical model, as in model identification or within a probabilistic analysis, the issue of computational cost becomes very important. In order to overcome it, the developed numerical model was simplified. Therefore, both finite element and load step numbers were minimized. Accordingly, three mesh types (with 3983, 1333 and 427 elements) and two different load steps (210 steps of factor 0.50 and 60 steps of factor 2) were studied. Figure 5.9a presents a finite element mesh with 1333 elements.

In a further analysis, the performance of each model was evaluated. In order to assure identical computational conditions, the same computer was used. The computational time and the related error were determined for each analysis. The applied load error was computed through equation (5.2),

$$\Delta_F^i = \left(F_1^i - F_0^i\right) \Big/ F_0^i \,[\%] \tag{5.2}$$

in which F_1^i is the applied load for a specific step *i* of evaluated numerical model and F_0^i is the applied load for the same step *i* of the reference numerical model. The reference model is the one that presents the most refined mesh and higher number of load steps. Then, the maximum and minimum Δ_F-values are determined and the sum of their absolute values is

computed. Finally, the applied load error (θ_F) is computed by dividing this value per two. The pinned support reaction error was computed through equation (5.3),

$$\Delta_R^i = \left(R_1^i - R_0^i\right) / R_0^i \,[\%] \tag{5.3}$$

in which R_1^i is the obtained reaction value for a specific step *i* of evaluated numerical model and R_0^i is the obtained reaction value for the same step *i* of the reference numerical model. The reference model is the same as the one used to determine the applied load error. Then, the maximum and minimum Δ_R-values are determined and the sum of their absolute values is computed. The reaction error (θ_R) is then computed by dividing this value per two. Table 5.5 gives the obtained results. It is possible to conclude that the most suitable model, for a further analysis, is the number 3. In fact, by comparing with other models, it is possible to conclude that this model presents a lower error (≈6%) and, at same time, by comparing with reference model (number 0) it reduces the computational cost in almost 90%.

Table 5.5. Simplification results.

Numerical model	Finite element number	Step number	Computational time [*s*]	Applied load error - θ_F [%]	Reaction error - θ_R [%]
0	3983	210	966.26	-	-
1	3983	60	756.41	2.14	3.80
2	1333	210	211.38	4.80	4.58
3	1333	60	141.01	6.00	5.82
4	427	210	79.15	9.16	9.32
5	427	60	41.56	9.23	12.57

A first step calibration procedure was developed, taking into consideration the chosen numerical model, to determine the most suitable spring stiffness value (k = 149.13 kN/m). Such procedure consisted in identifying the value that optimizes the distance between numerical and experimental data. In order to develop this analysis, an evolutionary strategies optimization algorithm in its plus version [29] was used. During this procedure the other parameter values were fixed. It is verified that the fixed support is not working as a full clamp since the beginning of loading, as the obtained stiffness value is low. This parameter intends to represent the concrete accommodation in fixed support during the initial phase.

The vertical displacement restraint is only assured in a more advanced phase of the test, in which the spring elements are replaced by pinned supports. During the same optimization procedure it was identified this instant, which occurs for load step 30. Accordingly, the load

cases were divided into 30 steps with a factor of 2 in the presence of spring supports, and 30 steps with a factor of 2 with pinned supports. During this last phase the fixed support works as a full clamp, totally restraining the beam rotation.

The collapse mechanism is characterized by two plastic hinges, the first at fixed support and the second beside the point load, which is close to the pinned support. A bending failure mode, with concrete crushing and yielding of longitudinal steel reinforcement, is obtained. The numerical behavior of analyzed reinforced concrete beams was similar to the one obtained in experimental tests. Obtained results validate the numerical model. Figure 5.9b presents the deformation, crack pattern and horizontal strain of tested beam.

a) b)

Figure 5.9. Numerical model: a) finite element mesh; b) failure mechanism.

5.4. Model identification

The basis of a model identification methodology is to rearrange a set of numerical parameters in such a way that the numerical response best fits the existent experimental data. This fact converts this kind of analysis into a typical optimization problem. In this case, the optimization function is based in an approximation between numerical and experimental data, and the objective is to obtain the curve that best adapts to existent experimental data. A detailed description of this function is given in chapter four.

The optimization algorithm that was used in this analysis was the evolutionary strategy in its plus version [29]. It begins with an initial population of critical parameter values, generated randomly, and then, using the evolutionary operators, new populations are generated. A final population is extracted for each run. A detailed description of this algorithm is given at chapter two. This algorithm is processed with different starting points. An engineer judgment procedure, based in the probability of occurrence of each individual, is developed to determine the most suitable individual, from those previously extracted. This procedure is detailed described in chapter four.

When using this procedure, multiple runs of the same numerical model are necessary. In each run, the fitness function value, which characterizes the approximation between experimental and numerical curves, is computed. The identification stops when one of the algorithm stopping criteria is attained. One of these criteria consists in establishing that the improvement on minimum fitness function value, obtained from two generations separated of

a pre-specified gap, should be less than or equal to a threshold value. This value is computed through the law of propagation of uncertainty [90, 91, 92], detailed described at chapter four. It may be interpreted as the methodology precision, once obtained results become more accurate with its decrease.

It is known that one of the main disadvantages of model identification, the computational cost, increases with the number of variables to be optimized. Therefore, it is first important to select those which are critical. This can be done by performing a sensitivity analysis. This analysis consists in studying the fitness function variation with each input parameter. An importance measure (b_k) was then obtained for each evaluated parameter. A detailed description of this measure is given in chapter four. In this case, if this value is equal or higher than 10% (b_{lim}) then the parameter will be considered as critical.

In this case, model identification was performed for both service and failure region. The analysis in service phase identified different combinations of values for critical parameters that lead to very good results. However, the majority of these combinations lead to bad results in an analysis until failure load. This is important to highlight as model identification is typically performed with data from load test, and the majority of load tests are developed in service phase. Therefore, although a good approximation of registered data is obtained, this does not mean that the structure behavior until failure load is perfectly characterized. In order to overcome this, other complementary tests such as nondestructive tests (NDT), laboratory characterization tests and visual inspections should be executed.

5.4.1. Pinned-pinned beams

Two sensitivity analyses were developed in this case, one for service phase and other until failure load. The evaluated parameters are those related to materials, concrete and steel, and to geometry. It was respectively varied one standard deviation (σ) from each parameter mean value. In order to compute each standard deviation (σ), the following coefficient of variations (CV) were established [93]: (1) concrete elasticity modulus (E_c): 10%; (2) concrete tensile strength (f_t): 20%; (3) concrete compressive strength (f_c): 10%; (4) concrete fracture energy (G_f): 10%; (5) concrete compressive strain at compressive strength (ε_c): 10%; (6) concrete critical displacement (w_d): 10%; (7) reinforcing steel elasticity modulus (E_s): 5%; (8) reinforcing steel yield strength (σ_y): 5%; (9) reinforcing steel limit strength (σ_u): 5%; (10) reinforcing steel limit strain (ε_{lim}): 15%; (11) reinforcing steel area (A_s): 2%; (12) inferior concrete cover (c_{inf}): 20%; (13) superior concrete cover (c_{sup}): 20%; (14) beam width (b): 10%; and (15) beam height (h): 10%. Figure 5.10 and 5.11 gives the obtained results.

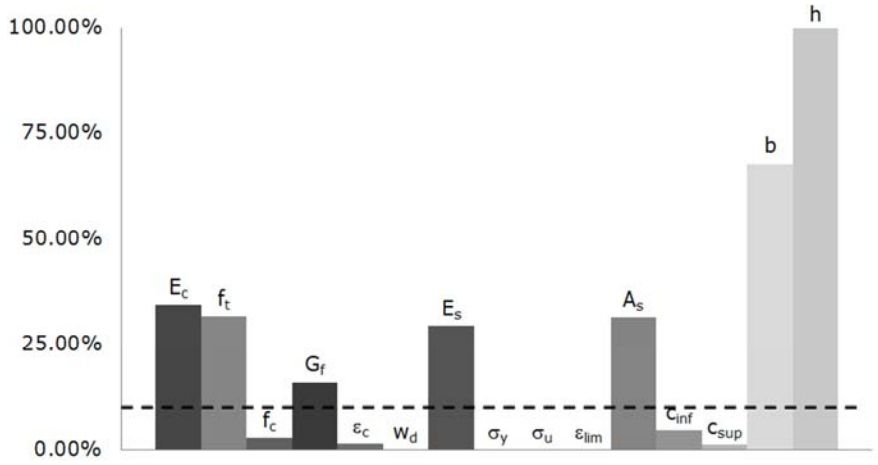


Figure 5.10. Importance measure (service).

The analysis developed in service phase pointed out for the importance of concrete elasticity modulus (E_c), tensile strength (f_t) and fracture energy (G_f), of longitudinal steel reinforcement elasticity modulus (E_s) and area (A_s), and of section width (b) and height (h). In fact, it is reasonable to admit that when submitted to low stresses the beam response is only dependent of concrete elasticity modulus and fracture energy, and of steel elasticity modulus. In this situation the concrete, in tension, reached their tensile strength and started to crack. Therefore, tensile strength is also an important parameter. By developing this analysis, from 15 initial possible parameters, only 7 of them were considered critical, reducing so the computational cost.

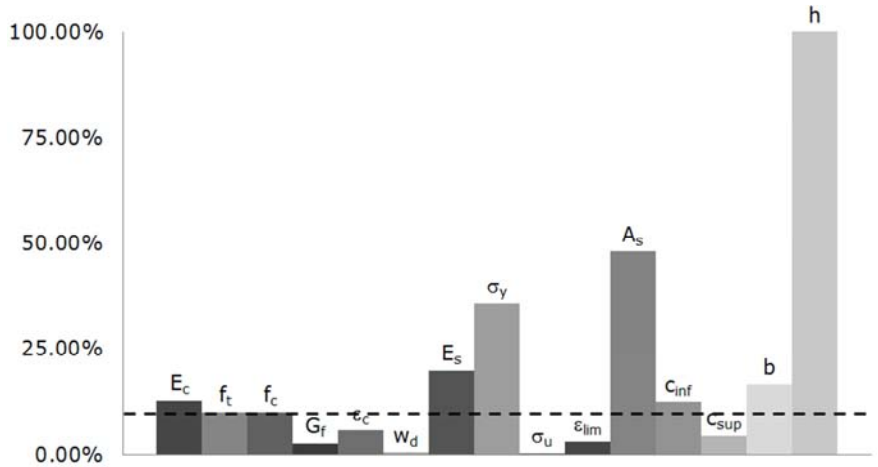


Figure 5.11. Importance measure (failure).

The analysis performed until failure load revealed a decrease on the importance of concrete parameters, with exception of compressive strength (f_c) that increases. In respect to longitudinal steel reinforcement it is important to point out an increase on the importance of the yield strength (σ_y). In fact, it is verified for higher loads that steel reached this strength and started the hardening phase. In respect to geometry parameters, it is important to refer

the increase on the importance measure of inferior concrete cover (c_{inf}). Other critical parameters, previously identified during the sensitivity analysis in service phase, are still considered. Therefore, from 15 initial parameters, only 9 of them were considered, reducing so the computational cost.

Once the numerical model and critical parameters, to be optimized, are identified, the further step is the application of proposed model identification methodology. In this case, the middle span displacement was measured during the laboratory test. This measurement is expressed by the graphic that plots the applied load (F) against the measured displacement (δ). In this situation the fitness function characterizes the approximation between numerical and experimental values for applied loads. One of used tolerance criteria is related to the convergence in the space of fitness value, defined by the threshold value (ε). In order to obtain this value, a division between uncertainty types, respectively, experimental and numerical, is developed.

From experimental uncertainties, it is possible to select: (1) Sensor accuracy (0.10%), which includes not only the displacement transducer precision but also the cable and acquisition equipment losses [68, 154]; (2) Load positioning, which, in this case is considered to be zero as it was perfectly controlled within the developed test; (3) Load intensity (0.10%), that includes not only the load cell resolution but also the cable and the acquisition equipment losses [68, 154]; (4) Environmental effects, which can be neglected due to the fact of being a short term test and so the variations in temperature and humidity are very small to be considered; (5) Vibration noise, that can also be ignored as the test is performed in a static way.

In this case, only the load intensity component will be considered when computing the experimental uncertainty. This component presents a uniform PDF (Type B) and so, according to JCGM [90, 91, 92], it should be divided by $\sqrt{3}$, obtaining then the result of $5.77*10^{-2}$ %. In order to compute the experimental uncertainty it will be necessary to determine the experimental data derivative in respect to this component ($\partial y^{exp}/\partial x$ = 1.00 kN). This uncertainty is obtained through equation (5.4) [90, 91, 92],

$$u_{\exp}^2 = 1.00^2 \cdot \left(5.77 \cdot 10^{-2}/100\right)^2 = 3.33 \cdot 10^{-7} \rightarrow u_{\exp} = 5.77 \cdot 10^{-4} \text{ kN} \tag{5.4}$$

From numerical uncertainties, it is possible to select: (1) Finite element method accuracy (3.79%), determined by comparing the developed numerical model with other which presents a higher number of load steps [69]; (2) Mesh refinement (6.74%), determined by comparing the developed numerical model with other which presents a more refined mesh [69]; (3) Model exactitude, that can be neglected as the numerical model is developed

according to the experimental test; (4) Considered hypothesis, that are also neglected as all model simplifications (e.g. consideration of supports as point loads) are validated within a global structural analysis.

When computing the numerical uncertainty, both finite element method and mesh refinement components will be considered. These components are represented by a uniform PDF (Type B) and so, according to JCGM [90, 91, 92], they should be divided by √3, obtaining then the result of 2.19 % and of 3.89 %, respectively. In order to determine the numerical uncertainty, the partial derivative of the numerical results in respect to these two components should be computed ($\partial y^{num}/\partial x$ = 1.00 kN). This uncertainty is obtained through equation (5.5) [90, 91, 92],

$$u_{num}^2 = 1.00^2 \cdot \left(3.89/100\right)^2 + 1.00^2 \cdot \left(2.19/100\right)^2 = 1.99 \cdot 10^{-3} \rightarrow u_{num} = 4.46 \cdot 10^{-2} \text{ kN} \quad (5.5)$$

Once the experimental and numerical uncertainties are computed, it will be possible to determine the fitness function uncertainty. In order to obtain this value, it is necessary to compute the partial derivative of the fitness function in respect to both experimental and numerical data. These values vary with tested beam as they are proportional to maximum applied load ($\partial f_1/\partial y^{num} = \partial f_1/\partial y^{exp}$ = 1/max(y_1^{exp}) = 4.10 * 10^{-2} kN^{-1}, for beam 1, and $\partial f_2/\partial y^{num} = \partial f_2/\partial y^{exp}$ = 1/max(y_2^{exp}) = 4.00 * 10^{-2} kN^{-1}, for beam 2). The fitness function uncertainty is respectively computed, for each tested beam, through equations (5.6) and (5.7) [90, 91, 92],

$$u_{f_1}^2 = \left(4.10 \cdot 10^{-2}\right)^2 \cdot \left(5.77 \cdot 10^{-4}\right)^2 + \left(4.10 \cdot 10^{-2}\right)^2 \cdot \left(4.46 \cdot 10^{-2}\right)^2 = 3.29 \cdot 10^{-6} \rightarrow u_{f_1} = 1.82 \cdot 10^{-3} \quad (5.6)$$

$$u_{f_2}^2 = \left(4.00 \cdot 10^{-2}\right)^2 \cdot \left(5.77 \cdot 10^{-4}\right)^2 + \left(4.00 \cdot 10^{-2}\right)^2 \cdot \left(4.46 \cdot 10^{-2}\right)^2 = 3.20 \cdot 10^{-6} \rightarrow u_{f_2} = 1.79 \cdot 10^{-3} \quad (5.7)$$

The global fitness function value is obtained through the square root of the sum of the square of these components. In order to determine the global uncertainty, the partial derivative of the fitness function in respect to each component should be computed ($\partial f/\partial f_1 = \partial f/\partial f_2$ = 1.00). This uncertainty is obtained through equation (5.8) [90, 91, 92],

$$u_f^2 = 1.00^2 \cdot \left(1.82 \cdot 10^{-3}\right)^2 + 1.00^2 \cdot \left(1.79 \cdot 10^{-3}\right)^2 = 6.49 \cdot 10^{-6} \rightarrow u_f = 2.55 \cdot 10^{-3} \quad (5.8)$$

The improvement on global fitness value (Δf) from two generations, separated of a specified gap (n), is given in chapter four. In order to determine its uncertainty, the partial derivative of the improvement in respect to each component needs to be computed ($\partial \Delta f/\partial f_{i+n} = \partial \Delta f/\partial f_i$ = 1.00). This uncertainty is obtained through equation (5.9) [90, 91, 92],

$$u_{\Delta f}{}^2 = 1.00^2 \cdot \left(2.55 \cdot 10^{-3}\right)^2 + 1.00^2 \cdot \left(2.55 \cdot 10^{-3}\right)^2 = 1.30 \cdot 10^{-5} \rightarrow u_{\Delta f} = 3.60 \cdot 10^{-3} \quad (5.9)$$

As all uncertainty sources are of Type B, a coverage factor (*k*) of 2 should be adopted [90, 91, 92]. The fitness value criterion establishes that the respective improvement (*Δf*) should be less than or equal to the threshold value (*ε*). This value is obtained by multiplying the value from expression (5.9) by factor *k*. The obtained threshold value for the analysis in service phase is determined in a similar way. These values are further indicated, (5.10),

$$\begin{cases} Service \rightarrow \varepsilon = 7.15 \cdot 10^{-3} = 0.72\% \\ Failure \rightarrow \varepsilon = 7.21 \cdot 10^{-3} = 0.72\% \end{cases} \quad (5.10)$$

This means that, for instance, for model identification until failure load, if the improvement in minimum fitness value of a population from two generations separated of a specified gap (*n*) is, respectively, less than or equal to 0.72%, the algorithm stops, as the fitness function tolerance criterion is reached. This shows that it is not meaningful to improve the fitness function of a value that is less than or equal to the precision itself.

The evolutionary strategy algorithm in its plus version [29] is further executed. In this case, a parent population (*μ*) and a parent for recombination (*ρ*) of 10 individuals, and an offspring population (*λ*) of 50 individuals were defined. The algorithm will run until one of the established criteria is reached. Other stopping criteria, as the maximum generation's number (1000), were considered. The generation gap (*n*), used for the fitness function tolerance criterion, is proportional to this number. It was established that this value is 10% of the specified maximum generation's number. Therefore, the improvement on minimum fitness value is evaluated from a gap of 100 generations. Once the algorithm stops, a population, constituted by different individuals, is obtained.

The respective algorithm is processed with different starting points. An engineer judgment procedure is developed to determine the most suitable individual, from those previously extracted. This individual is constituted by a set of values, a value for each critical parameter. Table 5.6 presents the nominal values, and individuals obtained from model identification in service phase and until failure load. In the same table, between brackets, the bias factor, which represents the ratio between the identified and the nominal value for each variable, is also presented. When applying this methodology in service phase, not only the critical parameters, as already identified during the sensitivity analysis, but also their optimal values, may differ from the application until failure load.

From a first analysis of Table 5.6., it is possible to realize that: (1) Obtained value of some parameters, as the concrete tensile (f_t) and compressive strength (f_c), the section width (*b*)

and height (h) and the inferior concrete cover (c_{inf}), is lower than the nominal one; (2) The longitudinal steel reinforcement elasticity modulus (E_s) value, obtained from this methodology, when applied until failure load, is lower than the nominal one and, both these values are lower than the one from the application of the methodology in service phase; (3) The concrete elasticity modulus (E_c) value, obtained from this methodology, when applied until failure load, is higher than the nominal one and, both these values are higher than the one obtained from the application of the methodology in service phase; (4) Obtained value of some parameters, as the concrete fracture energy (G_f) and the longitudinal steel reinforcement yield strength (σ_y) and area (A_s), is higher than the nominal one.

Table 5.6. Model identification results.

Numerical model					Nominal value	Model identification	
						Service *	Failure *
Parameter	Material	Concrete	E_c	[GPa]	31.00	24.80 (0.80)	31.44 (1.01)
			f_t	[MPa]	2.60	2.09 (0.80)	2.54 (0.98)
			f_c	[MPa]	33.00	33.00 (-)	29.87 (0.91)
			G_f	[N/m]	65.00	74.50 (1.15)	65.00 (-)
		Longitudinal steel reinforcement	E_s	[GPa]	200.00	233.27 (1.17)	186.73 (0.93)
			σ_y	[MPa]	500.00	500.00 (-)	535.83 (1.07)
			A_s	[cm^2]	0.85	1.02 (1.20)	0.91 (1.07)
	Geometry		c_{inf}	[cm]	1.00	1.00 (-)	0.99 (0.99)
			b	[cm]	7.50	6.53 (0.87)	7.27 (0.97)
			h	[cm]	15.00	14.52 (0.97)	14.28 (0.95)

* Bias factor is presented between brackets.

Both analyses indicated that concrete material presents a worse quality than the expected. The main reasons for that are the difficulties related to the concreting process of small structural elements. However, the analysis performed until failure load indicated a higher elasticity modulus than the predicted. When evaluating the steel reinforcement, obtained results indicated a better quality material. However, the analysis performed until failure load indicated a lower elasticity modulus than the predicted. Both analyses indicated a higher steel area. Obtained value from the analysis in service phase is far from the others. In respect to geometry parameters, obtained values for inferior concrete cover are close to nominal ones. Additionally, obtained beam dimensions are smaller than the predicted. Obtained results from the analysis until failure load are the closest to nominal values.

It is also important to mention that the same collapse mechanism and failure mode was obtained in experimental and numerical tested beams. Figure 5.12 presents the applied load (kN) plotted against the middle span displacement (m) for measured experimental data and for numerical results, obtained by considering the nominal values, and those from model identification in service phase and until failure load. From the analysis it is possible to conclude that the results from model identification until failure load are those that best fit the experimental curve. The numerical results considering nominal values present a higher cracking load and post-cracking stiffness and a lower failure load, while the results obtained by using model identification in service phase, are very similar to experimental data, within the service region, presenting then a higher post-cracking stiffness and failure load.

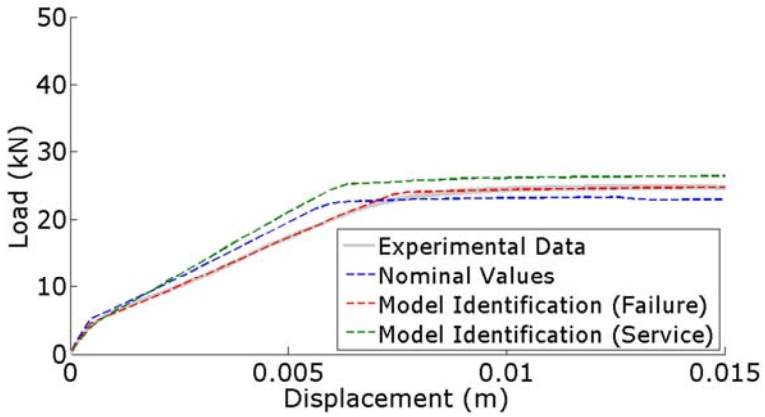


Figure 5.12. Numerical results.

Table 5.7 presents the minimum fitness function values obtained by considering the nominal values and those from model identification in service phase and until failure load. A first analysis permit to conclude that the fitness function value obtained in service phase is lower than that determined until failure load. This is due to the fact that in service region the numerical results are close to experimental data than in failure region. It is also verified, for both cases, an improvement of this value with model identification. However, it is also verified that this improvement is higher in service region (92.97%) than in failure region (75.90%). The structural behavior in failure region presents a higher nonlinearity. Therefore, for this region, optimization becomes harder and obtained results are not as good.

Table 5.8 presents the obtained failure load (F_R) by considering the nominal values and those from model identification in service phase and until failure load. Obtained error from model identification until failure load is considerably lower than that given by nominal values and by model identification in service phase. In fact, during this later analysis the model identification is only realized in service phase. Therefore, it becomes difficult to predict the failure load. Hence, model identification in service phase, itself, does not give good results.

Additional complementary testes are thus recommended. In respect to model identification until failure load, obtained error is less than 1% which is very good.

Table 5.7. Minimum fitness function values.

Numerical model	Fitness function			
	Service		Failure	
	Value [%]	Improvement [%]	Value [%]	Improvement [%]
Nominal values	3.38	-	7.70	-
Model identification	0.24	92.97	1.86	75.90

Table 5.8. Failure load (F_R).

Numerical model		Failure load	
		Value [kN]	Error [%] *
Nominal values		23.28	6.13
Model identification	Service	26.42	6.53
	Failure	24.84	0.16

* Comparing with the real failure load.

5.4.2. Pinned-fixed beams

Two sensitivity analyses were developed, one for service phase and other until failure load [122]. It were evaluated the same parameters as for pinned-pinned beams. Figure 5.13 and 5.14 presents, respectively, the results from the former and from the latter analysis.

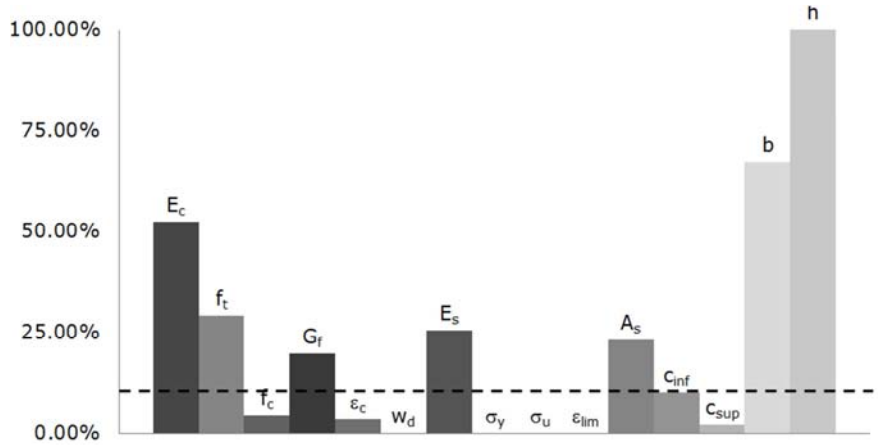


Figure 5.13. Importance measure (service).

The analysis developed in service phase pointed out for the importance of concrete elasticity modulus (E_c), tensile strength (f_t) and fracture energy (G_f), of inferior longitudinal reinforcement steel elasticity modulus (E_s) and area (A_s) and of section width (b) and height (h). Therefore, from 15 possible parameters, only 7 of them were considered critical, reducing thus the computational cost.

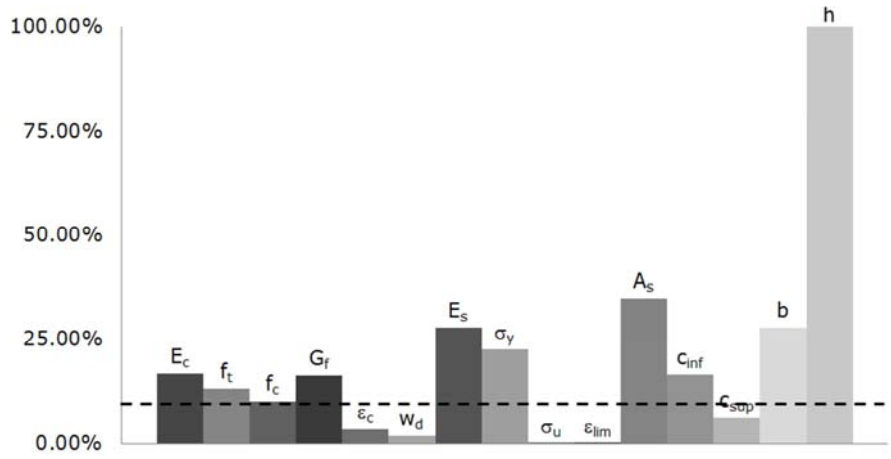


Figure 5.14. Importance measure (failure).

The analysis performed until failure load revealed that the critical parameters, previously identified with the analysis in service phase, still present an important influence in the structural behavior. On the other hand the influence of concrete compressive strength (f_c), of inferior longitudinal reinforcement yield strength (σ_y), and of inferior concrete cover (c_{inf}), increases, as expected, and should be also considered. Hence, from 15 possible initial parameters, only 10 of them were considered, reducing then the computational cost.

Two additional physic parameters that have a large influence on the overall structural behavior, namely, the spring stiffness and the instant in which the fixed support starts to work as a full clamp, were also considered during the analysis. Accordingly, 9 and 12 parameters will be, respectively, evaluated in the analysis for service phase and until failure load.

Once the numerical model is validated and the critical parameters selected, the further step consists in applying the model identification methodology. In this case, during the laboratory test, both the middle span displacement and the pinned support reaction were measured. These measurements are expressed by two graphics, respectively, one that plots the applied load (F) against the measured displacement (δ) and, other that plots the measured reaction (R) against the bending moment at fixed support (M). In this situation the fitness function characterizes the approximation between numerical and experimental values for both applied load and measured reaction.

One of used tolerance criteria is related to the convergence in the space of fitness value, defined by the threshold value (ε). In order to obtain this value, the uncertainty of each fitness function component will be studied in separate. The global uncertainty is then computed through their combination. Therefore, and for each component, it is necessary to identify both experimental and numerical uncertainty sources.

From experimental uncertainties, it is possible to select: (1) Sensor accuracy (0.10% - displacement transducer; 0.10% - load cell), which includes also the cable and acquisition equipment losses [68, 154]; (2) Load intensity (0.10%) that includes not only the load cell resolution but also the cable and acquisition equipment losses [68, 154]; (3) Load positioning, environmental effects and vibration noise which effects can be neglected in this case.

From numerical uncertainties, it is possible to select: (1) Finite element method accuracy (5.91%, δ-F; 6.16%, M-R), determined by comparing the developed numerical model with other which presents a higher number of load steps [69]; (2) Mesh refinement (11.42%, δ-F; 11.75%, M-R), determined by comparing the developed numerical model with other which presents a more refined mesh [69]; (3) Model exactitude and considered hypothesis (e.g. consideration of supports as point loads) that can be neglected in this situation.

The following step consists in determining the uncertainty of the fitness function applied load component. In this case, the experimental uncertainty is only dependent on the load intensity component. This component presents a uniform PDF (Type B) and so, according to JCGM [90, 91, 92], it should be divided by $\sqrt{3}$, obtaining then the result of $5.77*10^{-2}$ %. The experimental data is linearly dependent of this component ($\partial y^{exp}/\partial x$ = 1.00 kN). The experimental uncertainty is computed through equation (5.11) [90, 91, 92],

$$u^2_{\exp_{\delta-F}} = 1.00^2 \cdot \left(5.77 \cdot 10^{-2}/100\right)^2 = 3.33 \cdot 10^{-7} \rightarrow u_{\exp_{\delta-F}} = 5.77 \cdot 10^{-4} \text{ kN} \quad (5.11)$$

The same procedure is used to determine the numerical uncertainty. For this case, both finite element method and mesh refinement effects will be considered. These errors are represented by a uniform PDF (Type B) and so, according to JCGM [90, 91, 92], they should be divided by $\sqrt{3}$, obtaining then the result of 3.41% and 6.59%, respectively. The numerical results are linearly dependent of these components ($\partial y^{num}/\partial x$ = 1.00 kN). The numerical uncertainty is then computed through equation (5.12) [90, 91, 92],

$$u^2_{num_{\delta-F}} = 1.00^2 \cdot \left(3.41/100\right)^2 + 1.00^2 \cdot \left(6.59/100\right)^2 \rightarrow u_{num_{\delta-F}} = 7.42 \cdot 10^{-2} \text{ kN} \quad (5.12)$$

Once the experimental and numerical uncertainties are computed, it will be possible to determine the uncertainty of the fitness function applied load component. In order to obtain

this value, it is necessary to compute the partial derivative of this component in respect to both experimental and numerical data. These values vary with tested beam as they are proportional to maximum applied load ($\partial f_{\delta\text{-}F1}/\partial y^{num} = \partial f_{\delta\text{-}F1}/\partial y^{exp} = 1/\max(y_1^{exp}) = 3.30*10^{-2}kN^{-1}$, for beam 1, and $\partial f_{\delta\text{-}F2}/\partial y^{num} = \partial f_{\delta\text{-}F2}/\partial y^{exp} = 1/\max(y_2^{exp}) = 3.60 * 10^{-2}$ kN^{-1}, for beam 2). The uncertainty of the fitness function applied load component is then obtained, for each beam, through equations (5.13) and (5.14) [90, 91, 92],

$$u_{\delta-F_1}^2 = \left(3.30\cdot10^{-2}\right)^2\cdot\left(5.77\cdot10^{-4}\right)^2 + \left(3.30\cdot10^{-2}\right)^2\cdot\left(7.42\cdot10^{-2}\right)^2 = 5.92\cdot10^{-6} \rightarrow u_{\delta-F_1} = 2.43\cdot10^{-3} \quad (5.13)$$

$$u_{\delta-F_2}^2 = \left(3.60\cdot10^{-2}\right)^2\cdot\left(5.77\cdot10^{-4}\right)^2 + \left(3.60\cdot10^{-2}\right)^2\cdot\left(7.42\cdot10^{-2}\right)^2 = 6.90\cdot10^{-6} \rightarrow u_{\delta-F_2} = 2.63\cdot10^{-3} \quad (5.14)$$

The fitness function applied load component is obtained through the square root of the sum of the square of each beam component. In order to determine its uncertainty, the partial derivative of the fitness function applied load component in respect to each beam component should be computed ($\partial f_{\delta\text{-}F}/\partial f_{\delta\text{-}F1} = \partial f_{\delta\text{-}F}/\partial f_{\delta\text{-}F2} = 1.00$). This uncertainty is obtained through equation (5.15) [90, 91, 92],

$$u_{\delta-F}^2 = 1.00^2\cdot\left(2.43\cdot10^{-3}\right)^2 + 1.00^2\cdot\left(2.63\cdot10^{-3}\right)^2 = 1.28\cdot10^{-5} \rightarrow u_{\delta-F} = 3.58\cdot10^{-3} \quad (5.15)$$

In a second step, the uncertainty of the fitness function measured reaction component is determined. In this case, the experimental uncertainty is only dependent on the sensor accuracy component. As this component presents a uniform PDF (Type B), and according to JCGM [90, 91, 92], it should be divided by √3, obtaining then the result of 5.77*10^{-2} %. The experimental data is linearly dependent of this component ($\partial y^{exp}/\partial x$ = 1.00 kN). The experimental uncertainty is given by equation (5.16) [90, 91, 92],

$$u_{\exp_{M-R}}^2 = 1.00^2\cdot\left(5.77\cdot10^{-2}/100\right)^2 = 3.33\cdot10^{-7} \rightarrow u_{\exp_{M-R}} = 5.77\cdot10^{-4} \text{ kN} \quad (5.16)$$

An identical procedure is used to determine the numerical uncertainty. For this case, both finite element method and mesh refinement effects are considered. These errors are represented by a uniform PDF (Type B) and so, according to JCGM [90, 91, 92], they should be divided by √3, obtaining then the result of 3.56% and 6.78%, respectively. The numerical results are linearly dependent of these components ($\partial y^{num}/\partial x$ = 1.00 kN). The numerical uncertainty is then computed through equation (5.17) [90, 91, 92],

$$u_{num_{M-R}}^2 = 1.00^2\cdot\left(3.56/100\right)^2 + 1.00^2\cdot\left(6.78/100\right)^2 \rightarrow u_{num_{M-R}} = 7.66\cdot10^{-2}\,\text{kN} \quad (5.17)$$

Then, it is necessary to determine the uncertainty of the fitness function measured reaction component. The partial derivatives of such component, in respect to both experimental and numerical data, differ from beam to beam as they are proportional to maximum measured

reaction ($\partial f_{M\text{-}R1}/\partial y^{num} = \partial f_{M\text{-}R1}/\partial y^{exp} = 1/\max(y_1^{exp}) = 1.36 * 10^{-1}$ kN^{-1}, for beam 1, and $\partial f_{M\,R2}/\partial y^{num} = \partial f_{M\text{-}R2}/\partial y^{exp} = 1/\max(y_2^{exp}) = 1.56 * 10^{-1}$ kN^{-1}, for beam 2). The uncertainty of the fitness function measured reaction component is then computed, for each beam, through equations (5.18) and (5.19) [90, 91, 92],

$$u_{M-R_1}^2 = \left(1.36 \cdot 10^{-1}\right)^2 \cdot \left(5.77 \cdot 10^{-4}\right)^2 + \left(1.36 \cdot 10^{-1}\right)^2 \cdot \left(7.66 \cdot 10^{-2}\right)^2 = 1.08 \cdot 10^{-4} \rightarrow u_{M-R_1} = 1.04 \cdot 10^{-2} \quad (5.18)$$

$$u_{M-R_2}^2 = \left(1.56 \cdot 10^{-1}\right)^2 \cdot \left(5.77 \cdot 10^{-4}\right)^2 + \left(1.56 \cdot 10^{-1}\right)^2 \cdot \left(7.66 \cdot 10^{-2}\right)^2 = 1.42 \cdot 10^{-4} \rightarrow u_{M-R_2} = 1.19 \cdot 10^{-2} \quad (5.19)$$

The fitness function measured reaction component is obtained through the square root of the sum of the square of each beam component. In order to determine its uncertainty, the partial derivative of the fitness function measured reaction component in respect to each beam component should be computed ($\partial f_{M\text{-}R}/\partial f_{M\text{-}R1} = \partial f_{M\text{-}R}/\partial f_{M\text{-}R2} = 1.00$). This uncertainty is obtained through equation (5.20) [90, 91, 92],

$$u_{M-R}^2 = 1.00^2 \cdot \left(1.04 \cdot 10^{-2}\right)^2 + 1.00^2 \cdot \left(1.19 \cdot 10^{-2}\right)^2 = 2.50 \cdot 10^{-4} \rightarrow u_{M-R} = 1.58 \cdot 10^{-2} \quad (5.20)$$

The global fitness value is the square root of the sum of the square of each component, respectively, for applied load and measured reaction. In this situation the global fitness function is linearly dependent of these components ($\partial f/\partial f_{\delta\text{-}F} = \partial f/\partial f_{M\text{-}R} = 1.00$). The global uncertainty is computed through equation (5.21) [90, 91, 92],

$$u_f^2 = 1.00^2 \cdot \left(3.58 \cdot 10^{-3}\right)^2 + 1.00^2 \cdot \left(1.58 \cdot 10^{-2}\right)^2 = 2.62 \cdot 10^{-4} \rightarrow u_f = 1.62 \cdot 10^{-2} \quad (5.21)$$

The improvement on global fitness value (Δf) from two generations, separated of a specified gap (n), is given in chapter four. In order to determine its uncertainty, the partial derivative of the improvement in respect to each component needs to be computed ($\partial \Delta f/\partial f_{i+n} = \partial \Delta f/\partial f_i = 1.00$). This uncertainty is obtained through equation (5.22) [90, 91, 92],

$$u_{\Delta f}{}^2 = 1.00^2 \cdot \left(1.62 \cdot 10^{-2}\right)^2 + 1.00^2 \cdot \left(1.62 \cdot 10^{-2}\right)^2 = 5.25 \cdot 10^{-4} \rightarrow u_{\Delta f} = 2.29 \cdot 10^{-2} \quad (5.22)$$

As all uncertainty sources are of Type B, a coverage factor (k) of 2 should be adopted [90, 91, 92]. The fitness function criterion establishes that the respective improvement (Δf) should be less than or equal to the threshold value (ε). This value is obtained by multiplying the value from expression (5.22) by factor k. The obtained threshold value for the analysis in service phase is determined in a similar way. These values are further indicated, (5.23),

$$\begin{cases} Service \rightarrow \varepsilon = 4.17 \cdot 10^{-2} = 4.17\% \\ Failure \rightarrow \varepsilon = 4.58 \cdot 10^{-2} = 4.58\% \end{cases} \quad (5.23)$$

This means that, for instance, for model identification until failure load, if the improvement in minimum fitness function value of a population from two generations separated of a specified gap (n) is, respectively, less than or equal to 4.58% then the algorithm stops, as the fitness function convergence criteria is attained. This shows that it is not meaningful to improve the fitness function of a value less than or equal to the precision itself. These values are higher than those obtained for pinned-pinned beams. In fact, the threshold value increases with the number of necessary points to compute the fitness function.

The evolutionary strategy algorithm in its plus version [29] is further executed. In this case, it was considered a parent population (μ) and a parent for recombination (ρ) of 10 individuals, and an offspring population (λ) of 50 individuals. The algorithm is executed until one of the established criteria is attained. Other stopping criteria, as the maximum number of generations (1000), were considered. The generation gap (n), used for the fitness function tolerance criterion, is proportional to this number. It was established that this value is 10% of the specified maximum number of generations. Therefore, the improvement on minimum fitness function value is evaluated from a gap of 100 generations. Once the algorithm stops, a population, constituted by different individuals, is obtained.

The respective algorithm is executed with different starting points. An engineer judgment procedure is developed to determine the most suitable individual, from those previously extracted. This individual is constituted by a set of values, a value for each critical parameter. Table 5.9 presents the nominal values, and individuals obtained from model identification in service phase and until failure load [119, 121]. In the same table, between brackets, the bias factor, which represents the ratio between the identified and the nominal value for each variable, is also presented.

From a first analysis of Table 5.9., it is possible to realize that: (1) Obtained value of some parameters, as the concrete elasticity modulus (E_c) and compressive strength (f_c), the section width (b) and height (h) and the load step, is lower than nominal one; (2) The inferior longitudinal steel reinforcement elasticity modulus (E_s) and spring stiffness (k) values, obtained from this methodology, when applied until failure load, are lower than the nominal ones, being both these values lower than the ones obtained from the application of the methodology in service phase; (3) The concrete tensile strength (f_t) and fracture energy (G_f) values, obtained from this methodology, when applied until failure load, are higher than the nominal ones and, both these values are higher than the ones obtained from the application of the methodology in service phase; (4) Obtained value of some parameters, like the inferior longitudinal steel reinforcement yield strength (σ_y) and area (A_s) and the inferior concrete cover (c_{inf}), is higher than nominal one.

Table 5.9. Model identification results.

Numerical model					Nominal value	Model identification	
						Service *	Failure *
Parameter	Material	Concrete	E_c	[GPa]	31.00	30.34 (0.98)	29.07 (0.94)
			f_t	[MPa]	2.60	2.45 (0.94)	2.63 (1.01)
			f_c	[MPa]	33.00	33.00 (-)	30.74 (0.93)
			G_f	[N/m]	65.00	63.40 (0.98)	67.00 (1.03)
		Inferior longitudinal steel reinforcement	E_s	[GPa]	200.00	244.58 (1.22)	180.96 (0.90)
			σ_y	[MPa]	500.00	500.00 (-)	548.28 (1.10)
			A_s	[cm^2]	0.85	1.02 (1.20)	0.89 (1.05)
	Geometry		c_{inf}	[cm]	2.00	2.00 (-)	2.04 (1.02)
			b	[cm]	7.50	7.04 (0.94)	7.15 (0.95)
			h	[cm]	15.00	12.16 (0.81)	13.59 (0.91)
	Physic		k	[kN/m]	149.13	164.21	112.75
			step	[-]	30	26	25

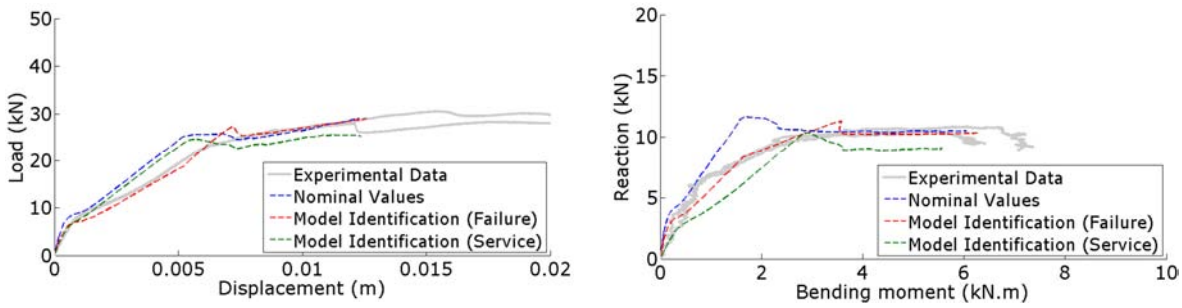
* Bias factor is presented between brackets.

Both analyses indicated that concrete material presents a worse quality than the expected. When evaluating the steel reinforcement, obtained results indicated a better quality material. Both analyses indicated a higher steel area. Obtained value from the analyses in service phase is far from the others. In respect to geometry parameters, obtained values for inferior concrete cover are close to nominal ones. Additionally, obtained beam dimensions are smaller than the predicted. Considering physic parameters, a lower spring stiffness value is given by the analysis until failure load. Obtained results from the analysis until failure load are the closest to nominal values.

Comparing these results with those obtained from pinned-pinned beams, and taking into consideration that the used steel is of same quality, that concrete material is of same class, and that geometry and production procedures are identical, the following conclusions are attained: (1) Obtained results for pinned-fixed beams are closer to nominal values than for pinned-pinned beams; (2) The concrete quality is slightly inferior than the predicted; (3) The steel reinforcement quality is higher than the predicted; (4) The steel reinforcement elasticity modulus, obtained from the analysis until failure load, is lower than the nominal value; (5) For both analysis, the steel area is higher than the nominal value; (6) Beam dimensions are, for

both analysis, smaller than the predicted; (7) Obtained inferior concrete cover, for both analysis, is close to the nominal value.

It is also important to mention that the same collapse mechanism and failure mode was obtained in experimental and numerical tested beams. In Figure 5.15a the applied load (kN) is plotted against the middle span displacement (m), and in Figure 5.15b the reaction at pinned support (kN) is plotted against the bending moment at fixed support (kN.m), for measured data and for numerical results, obtained by considering the nominal values and those from model identification in service phase and until failure load. From the analysis it is possible to conclude that the results from model identification until failure load are those that best fit the experimental curve. The numerical results considering nominal values present a higher cracking load and post-cracking stiffness. Obtained results from model identification in service phase are similar to experimental data, in service region, presenting then a higher post-cracking stiffness and a lower failure load and bending moment at fixed support.



a) b)

Figure 5.15. Numerical results.

Table 5.10 presents the minimum fitness function values obtained by considering the nominal values and those from model identification in service phase and until failure load. It is verified that the obtained fitness function value in service phase is lower than that determined until failure load. This is due to the fact that in service region the numerical results are close to experimental data than in failure region. Even so, for both situations, the developed methodology revealed a considerable improvement on this value. This updating procedure is more efficient in the service region (55.26%) than in failure region (28.69%). The fitness function value obtained in this situation is higher, and the respective improvement lower, than that for the pinned-pinned beams. This is due to an increase in the number of critical parameters and of necessary points to compute the fitness function, from a situation to another.

Table 5.11 indicates the failure load (F_R) and maximum bending moment at fixed support values by considering the nominal values and those from model identification in service

phase and until failure load. Obtained error from model identification until failure load is lower than that given by nominal values and by model identification in service phase. In fact, it is possible to verify that, when applying the methodology in service phase, the model identification is performed for this region, being not possible to guarantee the curve fitting for the failure region. Obtained error for the situation of model identification until failure load is less than 10% which is reasonable. Comparing the results from this analysis with those obtained for pinned-pinned beams it is possible to conclude that, again, this analysis revealed to be less precise.

Table 5.10. Minimum fitness function values.

Numerical model	Fitness function			
	Service		Failure	
	Value [%]	Improvement [%]	Value [%]	Improvement [%]
Nominal values	15.09	-	21.73	-
Model identification	6.75	55.26	15.50	28.69

Table 5.11. Failure load (F_R) and maximum bending moment (M_{R1}^*).

Numerical model		Failure load		Maximum bending moment	
		Value [kN]	Error [%] *	Value [kN.m]	Error [%] *
Nominal values		29.01	1.29	6.04	12.46
Model identification	Service	25.51	13.20	5.56	19.42
	Failure	29.17	0.75	6.26	9.28

* Comparing with the real failure load and maximum bending moment.

5.5. Characterization tests

5.5.1. Concrete material

Used concrete on pinned-fixed beams was of class C25/30 [48]. The stress-strain law parameters were determined by uniaxial compression tests in six cylindrical specimens [136] and by fracture energy tests in six beam specimens [155]. Each cylinder was tested at 28 days, presenting 300 mm of height and 150 mm of diameter, Figure 5.16a. Each beam was tested at 28 days, and presents 850 mm length by 100 mm height and 100 mm width. A notch with 25 mm depth and 5 mm thickness was executed in each specimen. These beams were submitted to a point load at middle span until failure, Figure 5.16b.

a) b)

Figure 5.16. Laboratory tests: a) uniaxial compression test; b) fracture energy test.

From the uniaxial compression tests it was possible to determine the concrete elasticity modulus (E_c), the compressive strain at compressive strength (ε_c), and the compressive strength (f_c). The fracture energy tests gave the concrete tensile strength (f_t) and the fracture energy (G_f). A statistical analysis was developed for each parameter. Both mean and standard deviation were determined. Table 5.12 gives those values. A bias value, which represents the ratio between the experimental and the nominal value, is also presented for each variable.

Table 5.12. Concrete parameters.

Parameter		Nominal value	Mean value (μ) *	Standard deviation (σ)
Elasticity modulus (E_c)	[GPa]	31.00	28.01 (0.90)	1.73
Tensile strength (f_t)	[MPa]	2.60	2.67 (1.03)	0.22
Compressive strength (f_c)	[MPa]	33.00	30.77 (0.93)	0.78
Fracture energy (G_f)	[N/m]	65.00	103.91 (1.60)	9.93
Compressive strain at compressive strength (ε_c)	[‰]	2.00	2.79 (1.40)	0.13

* Bias factor is presented between brackets.

Some bias values are close to one, which means that obtained data is close to the nominal value. However, this is not verified for some parameters as the compressive strain at compressive strength (ε_c) and the fracture energy (G_f). An important conclusion is that used concrete presents a lower quality than what was initially expected as bias values of more significant parameters, specifically elasticity modulus (E_c) and compressive strength (f_c), are lower than one. This confirms the obtained model identification results. Coefficients of

variation are all less than 10% which indicates that the variability of such parameters is small. Obtained correlation coefficient (ρ_{ij}) values for measured parameters are presented in Table 5.13.

Table 5.13. Correlation coefficients (ρ_{ij}) in concrete.

	E_c	f_t	f_c	G_f	ε_c
E_c	1.00	*	0.67	*	0.99
f_t	*	1.00	*	0.75	*
f_c	0.67		1.00	*	0.19
G_f	*	0.75	*	1.00	*
ε_c	0.99	*	0.19	*	1.00

* These parameters present no correlation as they were determined by different laboratory tests.

5.5.2. Steel material

Used steel on pinned-fixed beams was classified as S500B, according to EN 1992-1-1 [48]. Longitudinal steel reinforcement bars of ϕ6 and ϕ8 were used. In respect to transversal reinforcement the used diameter is ϕ4. Such steel was not classified before and so this analysis constitutes it first characterization. Six specimens were considered for each reinforcement diameter. The uniaxial tensile tests were executed according to the norm NP ENV 10002-1 [139] (Figure 5.17a). Each specimen presents a length of 500 mm (Figure 5.17b).

After collecting data from tested specimens, a statistical analysis was developed for each parameter and the respective mean and standard deviation value were obtained. A bias value, which represents the ratio between the experimental and the nominal value, is also presented for each variable. Such value was not determined for ϕ4 reinforcement as there was no information from the producer about this steel. Results are presented on Table 5.14 and 5.15.

a) b)

Figure 5.17. Laboratory tests: a) uniaxial tension test; b) reinforcing steel specimen.

Table 5.14. Steel parameters (transversal reinforcement).

Parameter		Mean value (μ)	Standard deviation (σ)
Elasticity modulus (E_s)	[GPa]	195.39	38.67
Yield strength (σ_y)	[MPa]	356.56	16.47
Limit strength (σ_u)	[MPa]	483.86	6.21
Limit strain (ε_{lim})	[‰]	143.62	6.38

Table 5.15. Steel parameters (longitudinal reinforcement).

Parameter		Nominal value	Mean value (μ) *	Standard deviation (σ)
Elasticity modulus (E_s)	[GPa]	200.00	205.31 (1.03)	20.22
Yield strength (σ_y)	[MPa]	500.00	582.94 (1.16)	22.46
Limit strength (σ_u)	[MPa]	540.00	692.61 (1.28)	13.54
Limit strain (ε_{lim})	[‰]	50.00	134.99 (2.70)	24.52

* Bias factor is presented between brackets.

The transversal reinforcement was identified to be of lower quality than S400B steel. For this situation the CV of elasticity modulus (E_s) presents a higher value. This is essentially due to the fact of being difficult to tie these bars during the test due to their small diameters.

For longitudinal reinforcement obtained bias values are all higher than one, which indicates that the steel quality is higher than what was initially expected. This confirms the obtained results from model identification. Coefficients of variation are, in a general way, less than 10% which indicates that the variability of such parameters is small. This is not verified

for limit strain (ε_{lim}) as obtained standard deviations indicates a high dispersion. The correlation coefficients (ρ_{ij}) between measured parameters were also determined. Obtained results are presented on Table 5.16 and 5.17.

Table 5.16. Correlation coefficients (ρ_{ij}) in steel (transversal reinforcement).

	E_s	σ_y	σ_u	ε_{lim}
E_s	1.00	0.10	0.43	0.19
σ_y	0.10	1.00	0.83	0.43
σ_u	0.43	0.83	1.00	0.58
ε_{lim}	0.19	0.43	0.58	1.00

Table 5.17. Correlation coefficients (ρ_{ij}) in steel (longitudinal reinforcement).

	E_s	σ_y	σ_u	ε_{lim}
E_s	1.00	0.23	0.33	0.18
σ_y	0.23	1.00	0.49	0.59
σ_u	0.33	0.49	1.00	0.48
ε_{lim}	0.18	0.59	0.48	1.00

5.5.3. Concrete cover

The concrete cover was evaluated for pinned-fixed beams, by cutting in two parts the tested beams, after each test, and measuring the distance from the longitudinal reinforcement to the top (c_{sup}) and bottom (c_{inf}) beam surfaces. Cuts were developed in the location of plastic hinges.

A statistical analysis was developed for both superior and inferior concrete cover. The mean and standard deviation values were determined through this analysis. Obtained results are presented in Table 5.18. A bias value, which represents the ratio between the experimental and the nominal value, is also presented.

Table 5.18. Concrete cover.

Parameter		Nominal value	Mean value (μ) *	Standard deviation (σ)
Inferior concrete cover (c_{inf})	[mm]	20.00	23.00 (1.15)	2.24
Superior concrete cover (c_{sup})	[mm]	20.00	24.94 (1.25)	4.17

* Bias factor is presented between brackets.

Bias values indicate that obtained concrete cover values are close to the nominal ones. This confirms the results obtained from model identification. The superior longitudinal reinforcement presents a higher concrete cover (c_{sup}). Moreover, a higher standard deviation is obtained in this situation. This is due to the fact that it is more difficult to control this cover during concreting. The correlation coefficient (ρ_{ij}) between these two parameters is 0.50.

5.6. Probabilistic analysis

Once the numerical model is developed and calibrated according to obtained experimental data, the next step consists in determining a reliable PDF for resistance. Such curve can be used in a further safety analysis, in which it will be compared to a specific loading PDF [33, 123]. This analysis will be important to help engineers in any decision regarding structural safety.

The resistance of reinforced concrete beams is dependent on material (concrete and steel), geometry (section dimensions and concrete cover) and physic parameters. Physic properties, as spring stiffness, previously determined by an optimization procedure, are considered to be deterministic due to the fact of being inherent to the studied structure. In order to obtain the resistance PDF it is important to consider the randomness in such parameters.

Model parameters are thus characterized by a PDF. The most used PDF are the Normal ones. In some situations, for which the parameters cannot assume negative values, a Lognormal PDF should be used instead. Those curves are defined by a mean (μ) and standard deviation (σ) value. In this case, these values are obtained from bibliography [26, 93, 148, 181, 182].

In some situations, when there is complementary data due to visual inspection, non-destructive tests (NDT), or even from installed monitoring systems, a Bayesian inference [15] approach can be used. This approach is detailed described in chapter three. In the situation of pinned-fixed beams, the updating procedure is based in collected data from material (concrete and steel) and geometry (concrete cover) tests. The Bayesian inference was developed by considering both informative and non-informative (Jeffrey's) prior. Considered posterior PDF is the one that presents the lowest standard deviation.

Once each critical parameter PDF is defined, the next step consists in random generation of these parameter values to be used in a further probabilistic numerical analysis. This procedure is based in a random sampling technique designated by Latin Hypercube sampling (LHS) [144]. This procedure is detailed described at chapter four.

A set of values are respectively obtained from the probabilistic analysis. These values are then statistically processed and fitted to a Normal PDF. Two indexes, index-*i* and index-*p*, presented in chapter four, are introduced. While the former is used to evaluate each beam behavior within the whole test, the latter is used to evaluate the accuracy of obtained resistance PDF.

5.6.1. Pinned-pinned beams

In this case, Normal and Lognormal PDF were considered for material and geometry parameters [114, 116, 117]. It was considered as mean value, the nominal values and those from model identification in service phase and until failure load. The adopted coefficients of variation were the same of previous sensitivity analysis. These values are indicated at Table 5.19. Table 5.20 presents the considered correlations between these parameters.

Table 5.19. Parameter values.

Parameter		Nominal value		Model identification (service)		Model identification (failure)	
		μ	σ	μ	σ	μ	σ
E_c	[GPa]	31.00	3.10	24.80	2.48	31.44	3.14
f_t	[MPa]	2.60	0.52	2.09	0.42	2.54	0.51
f_c	[MPa]	33.00	3.30	33.00	3.30	29.87	2.99
G_f	[N/m]	65.00	6.50	74.50	7.45	65.00	6.50
E_s	[GPa]	200.00	10.00	233.27	11.66	186.73	9.34
σ_y	[MPa]	500.00	25.00	500.00	25.00	535.83	26.79
A_s	[cm^2]	0.85	0.02	1.02	0.02	0.91	0.02
c_{inf}	[cm]	1.00	0.20	1.00	0.20	0.99	0.20
b	[cm]	7.50	0.75	6.53	0.65	7.27	0.73
h	[cm]	15.00	1.50	14.52	1.45	14.28	1.43

The applied load is obtained during the probabilistic analysis for each measured and computed displacement. Figure 5.18 shows the obtained results for the situation of nominal values. It is verified that the experimental data is within the 95% confidence interval for the majority of evaluated points. Obtained index-*i* presents a value of 94.25% for beam 1 and of 93.73% for beam 2.

Table 5.20. Correlation coefficients (ρ_{ij}).

	E_c	f_t	f_c	G_f	E_s	σ_y	A_s	c_{inf}	b	h
E_c	1.00	0.70	0.90	0.50	0.00	0.00	0.00	0.00	0.00	0.00
f_t	0.70	1.00	0.80	0.90	0.00	0.00	0.00	0.00	0.00	0.00
f_c	0.90	0.80	1.00	0.60	0.00	0.00	0.00	0.00	0.00	0.00
G_f	0.50	0.90	0.60	1.00	0.00	0.00	0.00	0.00	0.00	0.00
E_s	0.00	0.00	0.00	0.00	1.00	0.80	0.50	0.00	0.00	0.00
σ_y	0.00	0.00	0.00	0.00	0.80	1.00	0.50	0.00	0.00	0.00
A_s	0.00	0.00	0.00	0.00	0.50	0.50	1.00	0.00	0.00	0.00
c_{inf}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.10	0.60
b	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10	1.00	0.10
h	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.60	0.10	1.00

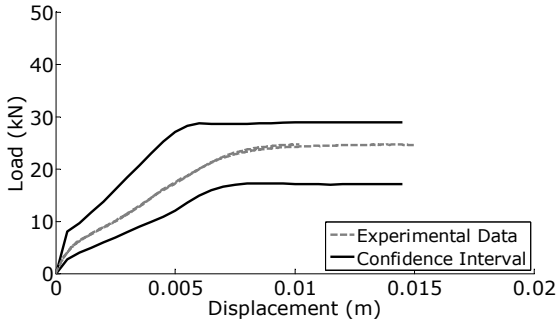


Figure 5.18. Nominal values.

Figure 5.19a displays the results for the situation of model identification in service phase. For this situation the experimental data is within the bounds defined by the 95% confidence interval for the majority of analysed points. The index-*i* is of 92.94% for beam 1 and of 93.52% for beam 2.

Figure 5.19b presents the results obtained by model identification until failure load. For this situation the experimental data is within the bounds defined by the 95% confidence interval for almost evaluated points. It is important to verify that, in this case, the experimental data curve is in the middle of these two bounds. Obtained index-*i* is of 98.85% for beam 1 and of 98.68% for beam 2.

In this case, a set of failure load (F_R) values is obtained. A Normal PDF, which represents the structural resistance, is then adjusted to this set. Obtained resistance PDF parameter values

are indicated in Table 5.21. On the same table it is also given the obtained index-*p* for each tested beam and for each numerical model.

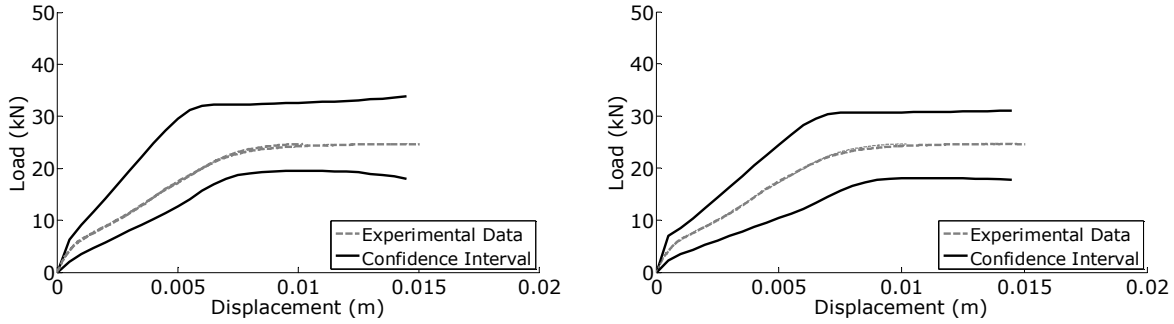


Figure 5.19. Model identification: a) service; b) failure.

From the analysis of these results it is important to conclude that: (1) Obtained mean and standard deviation with nominal values are lower than the ones obtained with values from model identification; (2) A high index-*p* is obtained for all situations. Therefore, an accurate resistance PDF is obtained for all cases; (3) Obtained index-*p* with nominal values is closer to the one obtained with values from model identification in service phase. This means that model identification in service phase does not increase the accuracy of resistance PDF; (4) Obtained index-*p* with values from model identification until failure load is the highest. This means that the use of model identification until failure load increases the accuracy of resistance PDF.

Figure 5.20 presents the resistance PDF that result from the parameter values (standard deviation and mean) indicated in Table 5.21. It is possible to identify that the obtained resistance PDF with values from model identification until failure load is located among the others. The obtained resistance PDF with values from model identification in service phase gives the highest mean. This will lead to a non-conservative safety analysis. The obtained resistance PDF with nominal values is close to the one obtained with values from model identification until failure load.

Table 5.21. Failure load (F_R).

Numerical model	μ [kN]	σ [kN]	Index-*p* [%]	
			Beam 1	Beam 2
Nominal values	23.27	2.97	95.56	94.47
Model identification (service)	26.37	3.34	94.91	95.89
Model identification (failure)	24.67	3.19	99.80	99.02

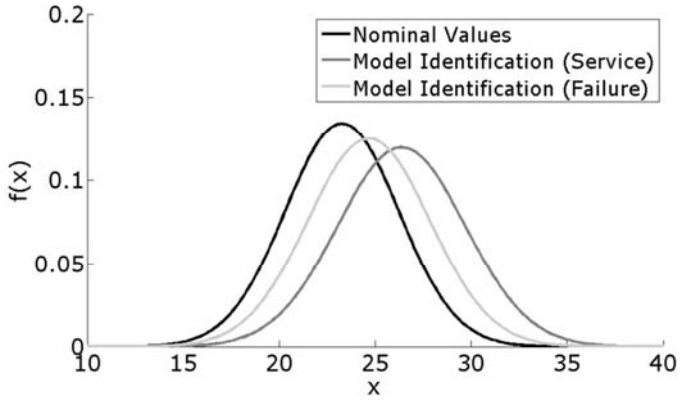


Figure 5.20. Failure load (F_R).

5.6.2. Pinned-fixed beams

Normal and Lognormal PDF were considered for material and geometry properties of pinned-fixed beams [116, 117, 119, 121, 122]. The PDF mean values were considered to be the nominal values and those from model identification in service phase and until failure load. The adopted coefficients of variation were the same of previous sensitivity analysis. These values are indicated at Table 5.22.

Table 5.22 present, between brackets, the results from a Bayesian inference analysis [119, 121, 122]. In this situation, the materials (concrete and steel) and geometry (concrete cover) parameters were updated. In respect to concrete material, obtained mean values are close to previous ones. However, for concrete fracture energy (G_f) this is not verified. The Bayesian inference reduced the standard deviation value for concrete compressive strength (f_c) and tensile strength (f_t). In respect to reinforcing steel elasticity modulus (E_s), obtained mean values are close to previous ones. However, for the reinforcing steel yield strength (σ_s) this is not verified. In respect to inferior concrete cover (c_{inf}), obtained mean value increased with Bayesian inference.

The Jeffrey's prior is used, during this procedure, to update the concrete compressive strength (f_c), except for data values due to model identification until failure load, and fracture energy (G_f). It was also used to update the reinforcing steel yield strength (σ_y) and the inferior concrete cover (c_{inf}), except for the situation of model identification until failure load. In the other situations an informative prior was used as it provides a lower standard deviation value.

Table 5.23 indicates the used correlation values. Inside brackets it is indicated the real values from complementary tests. These values are used when considering the updated PDF. It is verified, in a general way, that real correlations are not as strong as predicted.

Table 5.22. Parameter values.

Parameter		Nominal value *		Model identification (service) *		Model identification (failure) *	
		μ	σ	μ	σ	μ	σ
E_c	[GPa]	31.00 (30.10)	3.10 (4.79)	30.34 (29.68)	3.03 (4.51)	29.07 (28.76)	2.91 (3.36)
f_t	[MPa]	2.60 (2.62)	0.52 (0.31)	2.45 (2.68)	0.49 (0.34)	2.63 (2.64)	0.53 (0.31)
f_c	[MPa]	33.00 (30.79)	3.30 (1.38)	33.00 (30.79)	3.30 (1.38)	30.74 (30.69)	3.07 (1.16)
G_f	[N/m]	65.00 (104.61)	6.50 (15.83)	63.41 (104.61)	6.34 (15.83)	66.95 (104.61)	6.70 (15.83)
E_s	[GPa]	200.00 (202.48)	10.00 (11.69)	244.58 (223.98)	12.23 (14.20)	180.96 (192.84)	9.05 (12.09)
σ_y	[MPa]	500.00 (579.59)	25.00 (27.85)	500.00 (579.59)	25.00 (27.85)	548.28 (579.59)	27.41 (27.85)
A_s	[cm^2]	0.85	0.02	1.02	0.02	0.89	0.02
c_{inf}	[cm]	2.00 (2.14)	0.40 (0.48)	2.00 (2.14)	0.40 (0.48)	2.04 (2.16)	0.41 (0.46)
b	[cm]	7.50	0.75	7.04	0.70	7.15	0.72
h	[cm]	15.00	1.50	12.16	1.22	13.59	1.36

* Bayesian inference values are presented between brackets.

Table 5.23. Correlation coefficients (ρ_{ij}).

	E_c	f_t	f_c	G_f	E_s	σ_y	A_s	c_{inf}	b	h
E_c	1.00	0.70	0.90 (0.67) *	0.50	0.00	0.00	0.00	0.00	0.00	0.00
f_t	0.70	1.00	0.80	0.90 (0.75) *	0.00	0.00	0.00	0.00	0.00	0.00
f_c	0.90 (0.67) *	0.80	1.00	0.60	0.00	0.00	0.00	0.00	0.00	0.00
G_f	0.50	0.90 (0.75) *	0.60	1.00	0.00	0.00	0.00	0.00	0.00	0.00
E_s	0.00	0.00	0.00	0.00	1.00	0.80 (0.23) *	0.50	0.00	0.00	0.00
σ_y	0.00	0.00	0.00	0.00	0.80 (0.23) *	1.00	0.50	0.00	0.00	0.00
A_s	0.00	0.00	0.00	0.00	0.50	0.50	1.00	0.00	0.00	0.00
c_{inf}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00	0.10	0.60
b	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.10	1.00	0.10
h	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.60	0.10	1.00

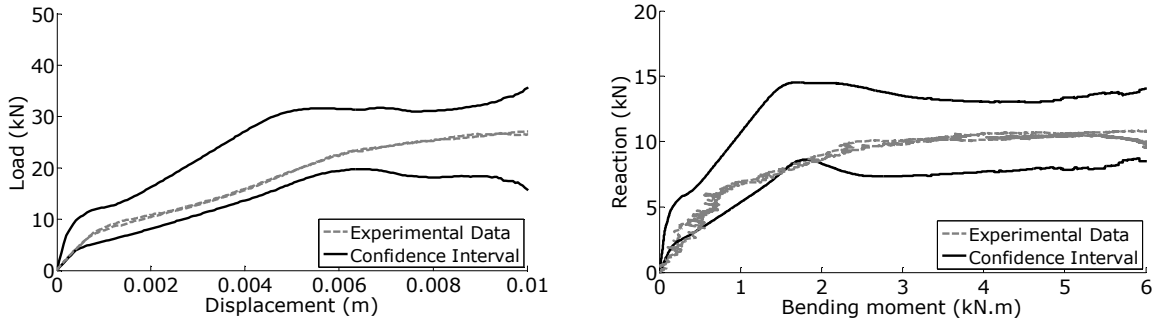
* Real values are presented between brackets.

During the probabilistic analysis it was evaluated the applied load and the bending moment at fixed support, respectively, for each measured and computed displacement and pinned support reaction. Figure 5.21 shows the obtained results for the situation of nominal values. It

is verified that the experimental data is within the 95% confidence interval for the majority of evaluated points. Obtained index-*i* presents a value of 91.34% (beam 1) and 92.03% (beam 2), for applied load, and of 93.90% (beam 1) and 92.30% (beam 2), for bending moment at fixed support [122]. The most accurate model is, in this case, the bending moment at fixed support one.

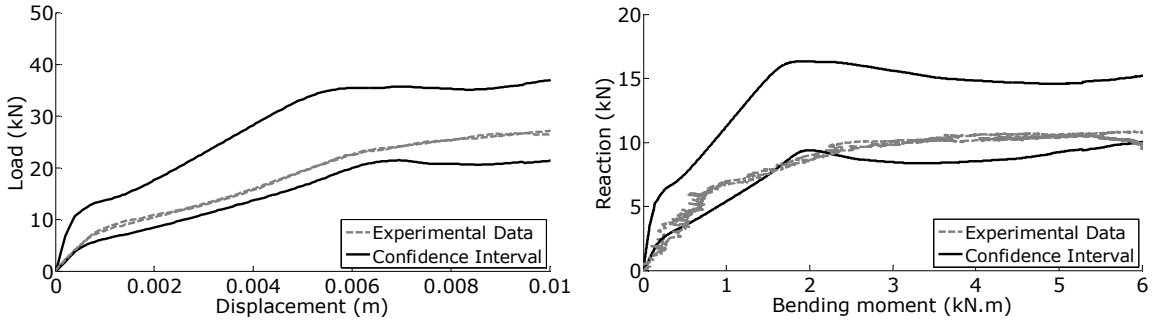
When applying the Bayesian inference to nominal values it is obtained the results given by Figure 5.22. In this case the index-*i* is of 89.54% (beam 1) and 89.81% (beam 2), for applied load, and of 88.99% (beam 1) and 87.89% (beam 2), for bending moment at fixed support [122]. This indicates that the Bayesian inference did not improve the accuracy of both numerical models. The most accurate model is, in this case, the applied load one.

Figure 5.23 gives the obtained results with values from model identification in service phase. In this case the index-*i* is of 91.99% (beam 1) and 91.98% (beam 2) for applied load, and of 92.55% (beam 1) and 92.62% (beam 2), for bending moment at fixed support. By comparing with models obtained for nominal values, it is possible to conclude that there is not any increase on the accuracy of both numerical models. The most accurate model is still the bending moment at fixed support one.



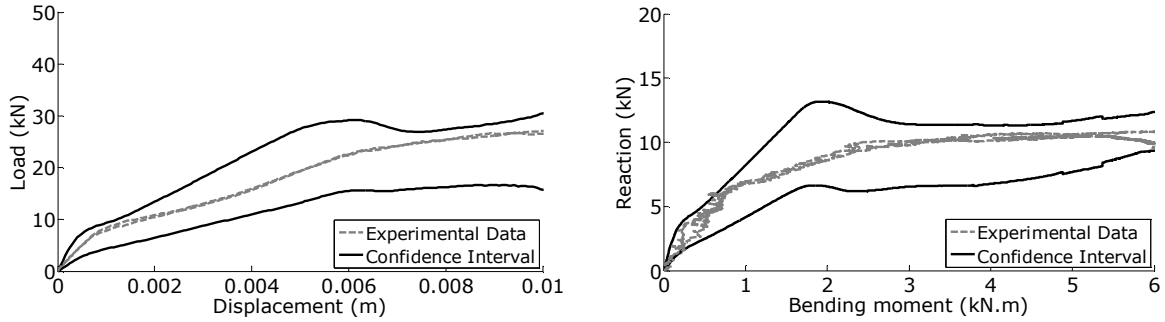
a) b)

Figure 5.21. Nominal values without Bayesian Inference.



a) b)

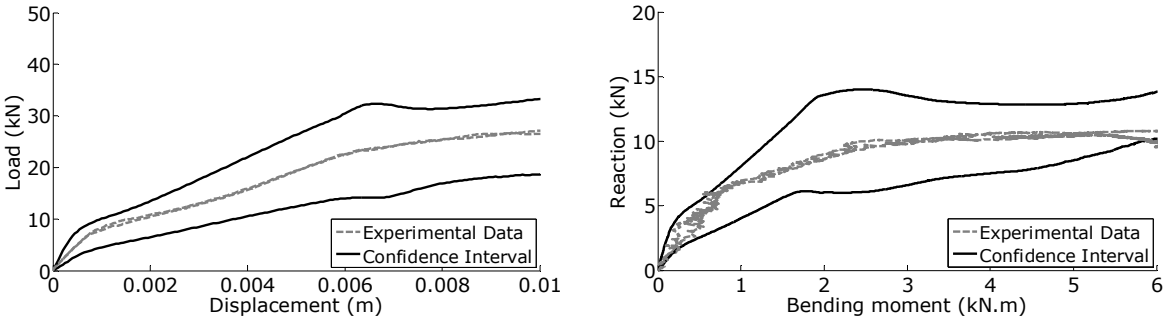
Figure 5.22. Nominal values with Bayesian Inference.



a) b)

Figure 5.23. Model identification (service) without Bayesian Inference.

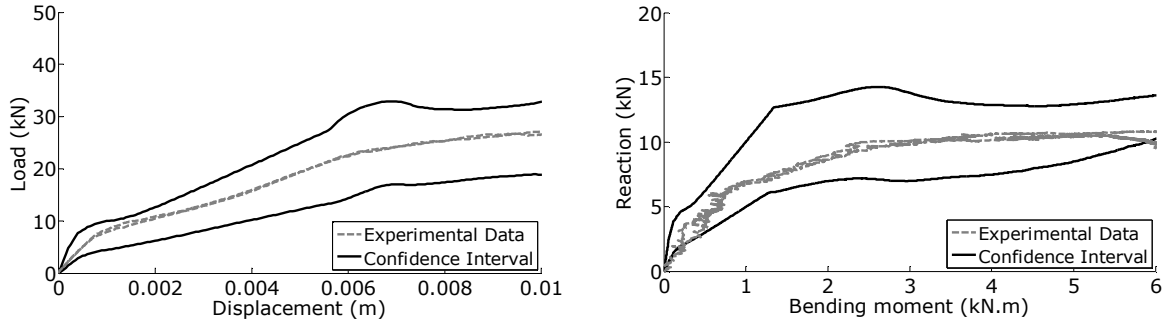
When applying the Bayesian inference to obtained values from model identification in service phase it is obtained the results given by Figure 5.24. Obtained index-*i* presents a value of 96.11% (beam 1) and 96.25% (beam 2), for applied load, and of 95.46% (beam 1) and 94.52% (beam 2), for bending moment at fixed support. The Bayesian inference improved, in this situation, the accuracy of both numerical models. The applied load model becomes the most accurate one.



a) b)

Figure 5.24. Model identification (service) with Bayesian Inference.

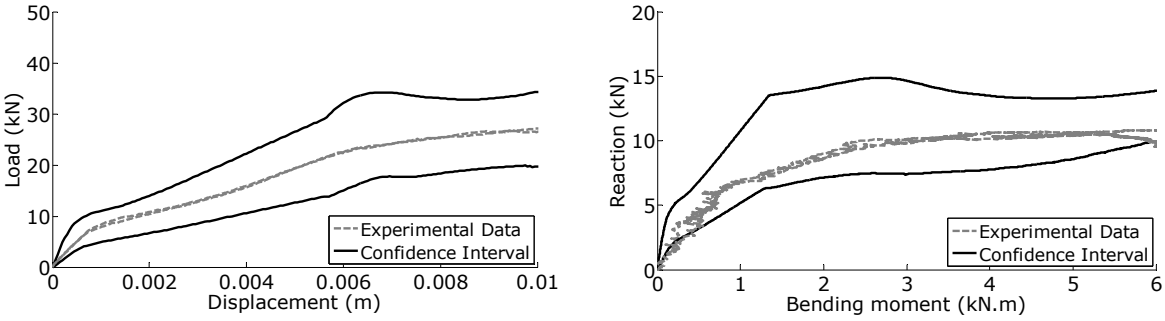
Figure 5.25 indicates the obtained results with values from model identification until failure load. In this case, the index-*i* is of 95.19% (beam 1) and 94.88% (beam 2), for applied load, and of 94.49% (beam 1) and 93.11% (beam 2), for bending moment at fixed support. It was verified that both models gave, in this situation, excellent results. There was an improvement on both the applied load and bending moment model, comparing with nominal values and those from model identification in service phase. The most accurate model is, in this case, the applied load one.



a) b)

Figure 5.25. Model identification (failure) without Bayesian Inference.

When applying the Bayesian inference to obtained values from model identification until failure load, it is obtained the results given by Figure 5.26. Obtained index-*i* presents a value of 96.04% (beam 1) and 95.91% (beam 2), for applied load, and of 92.91% (beam 1) and 90.94% (beam 2), for bending moment at fixed support. The Bayesian inference only improved the applied load model. The most accurate model is still the applied load one.



a) b)

Figure 5.26. Model identification (failure) with Bayesian Inference.

In this case, a set of failure load (F_R) values is obtained. A Normal PDF, which represents the structural resistance, is then adjusted to this set. Obtained resistance PDF parameter values are indicated in Table 5.24. On the same table it is also given the obtained index-*p* for each tested beam and for each numerical model [119, 122].

From the analysis of these results, it is possible to conclude that: (1) Obtained mean and standard deviation with nominal values and with values from model identification until failure load are higher than the ones obtained with values from model identification in service phase; (2) The Bayesian inference approach increased the mean and standard deviation of obtained resistance PDF; (3) Obtained index-*p* value for beam 2 is, in most cases, higher than the one obtained for beam 1; (4) In majority of situations, the Bayesian inference approach increased the index-*p* value. Obtained index-*p* values are very good, being, several

times, higher than 90%, which indicates accurate resistance PDF curves. Obtained resistance PDF with values from model identification until failure load, considering the Bayesian inference, presents the highest index-*p* values, considering both tested beams. Results from model identification in service phase are the poorer ones.

Table 5.24. Failure load (F_R).

Numerical model	μ [kN]	σ [kN]	Index-*p* [%]	
			Beam 1	Beam 2
Nominal values	28.49	3.79	94.58	99.35
Nominal values + Bayesian inference	31.69	4.19	97.09	92.24
Model identification (service)	25.12	3.37	88.06	91.31
Model identification (service) + Bayesian inference	27.82	3.66	92.90	98.73
Model identification (failure)	27.79	3.71	92.82	98.66
Model identification (failure) + Bayesian inference	29.07	3.84	96.12	97.79

Figure 5.27 indicates the resistance PDF, whose parameter values (mean and standard deviation) are presented in Table 5.24. Obtained resistance curves with nominal values and with values from model identification until failure load data are almost identical. The resistance PDF due to model identification in service phase presents a lower mean and standard deviation. An important conclusion is that the resistance PDF mean and standard deviation increased with the use of Bayesian inference.

In this case, a set of maximum bending moment at fixed support (M_{R1}^{*}) values is obtained. A Normal PDF, which represents the structural resistance, is then adjusted to this set. Obtained resistance PDF parameter values are indicated in Table 5.25. On the same table it is also given the obtained index-*p* for each tested beam and for each numerical model [119, 122].

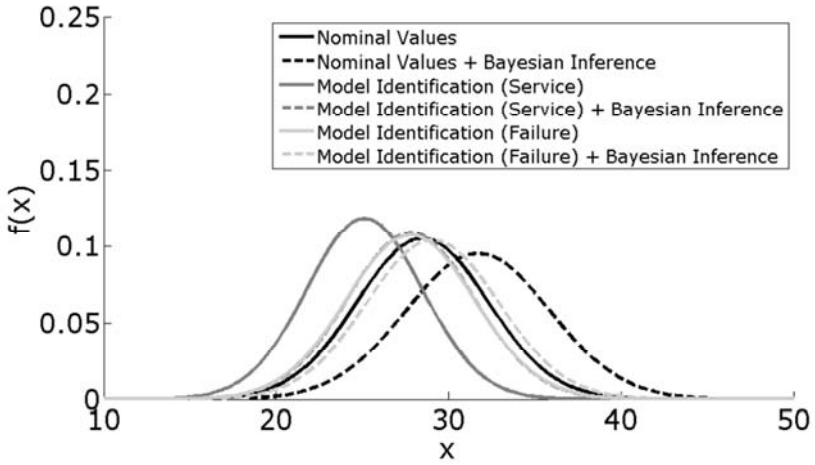


Figure 5.27. Failure load (F_R).

Table 5.25. Maximum bending moment (M_{R1}^*).

Numerical model	μ [kN.m]	σ [kN.m]	Index-p [%]	
			Beam 1	Beam 2
Nominal values	5.81	0.90	85.76	92.11
Nominal values + Bayesian inference	6.16	0.94	87.69	96.54
Model identification (service)	5.23	0.71	86.47	87.69
Model identification (service) + Bayesian inference	5.49	0.66	87.97	89.81
Model identification (failure)	5.49	0.73	86.82	89.34
Model identification (failure) + Bayesian inference	5.80	0.77	87.23	92.24

From the analysis of these results it is important to conclude that: (1) Obtained mean and standard deviation with nominal values are higher than the ones obtained with values from model identification; (2) The Bayesian inference increased the mean of obtained resistance PDF; (3) In majority of situations, the Bayesian inference increased the standard deviation of resistance PDF; (4) Obtained index-p value for beam 2 is higher than the one obtained for beam 1; (5) The Bayesian inference approach increased the index-p value. Obtained index-p values are good, being, in some situations, higher than 90%, which indicates accurate resistance PDF. Obtained resistance PDF with nominal values,

considering the Bayesian inference, presents the highest index-*p* values, considering both tested beams. Results from model identification in service phase are the poorer ones.

Figure 5.28 presents the resistance PDF, whose parameter values (mean and standard deviation), are indicated at Table 5.25. It is possible to identify that the obtained resistance PDF with values from model identification until failure load is located among the others. Obtained resistance PDF with nominal values, considering or not the Bayesian inference, are the ones that give the highest mean. The Bayesian inference approach increased the mean and standard deviation of maximum bending moment at fixed support, except for model identification in service phase.

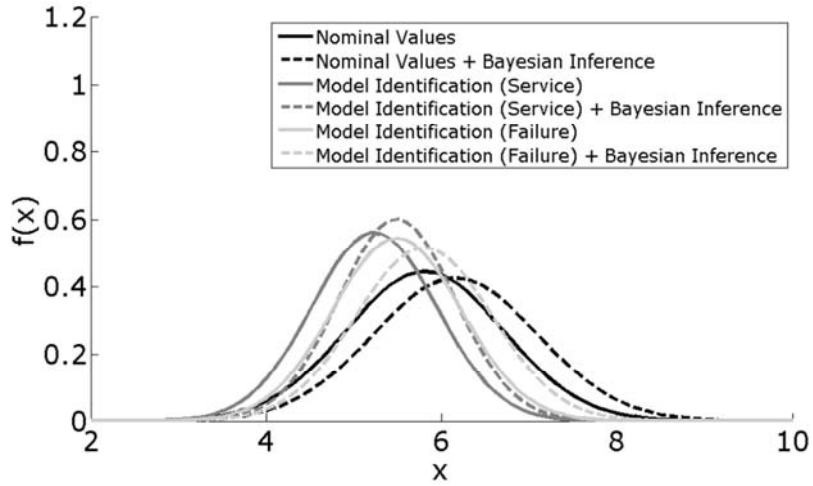


Figure 5.28. Maximum bending moment (M_{R1}^{*}).

5.7. Safety assessment

Obtained resistance model is then used in a simple example of safety assessment. This example consists in assessing the studied beam safety in a residential building. Two analyses were developed, respectively one for pinned-pinned (Figure 5.29a) and other for pinned-fixed beams (Figure 5.29b).

In this situation, loading is divided into self-weight and live load. All the other permanent loads were neglected. Self-weigh (w) is computed through equation (5.24),

$$w = \gamma_{conc} \cdot A_{conc} + \gamma_{steel} \cdot A_{steel} \tag{5.24}$$

being γ_{conc} and γ_{steel} the concrete and reinforcing steel specific weight, and A_{conc} and A_{steel} the concrete and reinforcing steel area. These parameters present a Normal PDF, according to JCSS [93]. Therefore, self-weight presents a Normal PDF.

Live loads are divided into: (1) sustained or long-term (q_{lt}), defined by a Gamma PDF [93]; and (2) intermittent or short term (q_{st}), defined by an exponential PDF [93]. The applied load (p) is the sum of self-weight with both long and short-term live load component multiplied by the influence length of the beam (L_{inf}), which is, in this situation, 6.0 m. This value is given by equation (5.25). Table 5.26 provides the mean and standard deviation of each PDF.

$$p = w + q_{lt} \cdot L_{\text{inf}} + q_{st} \cdot L_{\text{inf}} \tag{5.25}$$

Table 5.26. Probabilistic models.

Parameter		PDF	μ	σ
γ_{conc}	[kN/m^3]	Normal	24.00	0.96
γ_{steel}	[kN/m^3]	Normal	77.00	0.77
q_{lt}	[kN/m^3]	Gamma	0.30	0.45
q_{st}	[kN/m^3]	Exponential	0.30	0.57

5.7.1. Pinned-pinned beams

In this case, the resistance model is given by the failure load model (F_R), whose parameters are provided at Table 5.21. A model is obtained for each analysis, respectively, considering the nominal values and those from model identification in service phase and until failure load. In order to compare resistance and loading curves it is necessary to transform this model into a model for maximum bending moment at middle span (M_R), through equation (5.26),

$$M_R = \left(F_R \cdot 2\right)/2 \cdot L/2 \tag{5.26}$$

This model depends on the beam span (L) which is, in this situation, 1.50 m. A Normal PDF is obtained for resistance. The further step consists in computing the maximum bending moment (M_S), through equation (5.27),

$$M_S = \left(p \cdot L^2\right)/8 \tag{5.27}$$

A Lognormal PDF is then adjusted to obtained data. A limit state function (Z), which compares resistance and loading curves, is then defined through equation (5.28),

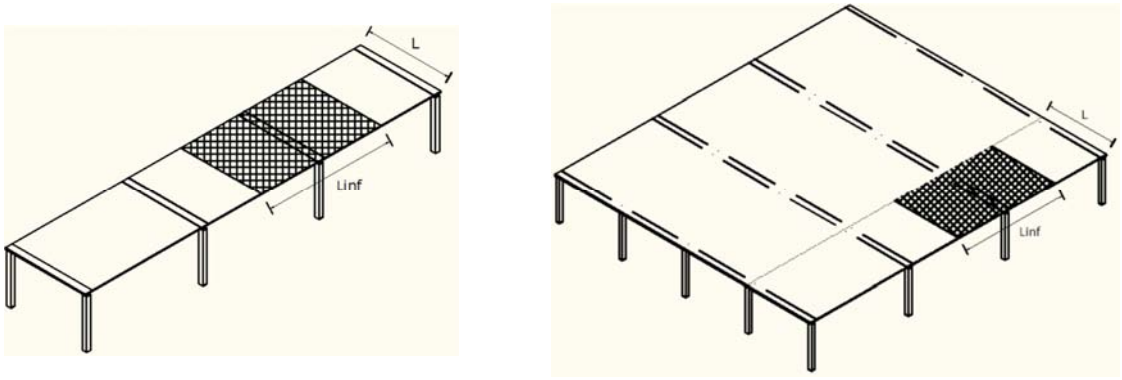
$$Z = M_R - M_S \tag{5.28}$$

The limit state is exceeded when loading is higher than resistance. The further step consists in generating values for each curve, according to each PDF parameters, and to register the

number of values in which this limit state is exceeded in relation to the total number of evaluated points. The failure probability (P_f) is determined through equation (5.29),

$$P_f = P(Z \leq 0) \tag{5.29}$$

The reliability index (β) is then obtained, considering this value. A detailed description of how this index is computed is given in chapter four. Table 5.27 presents both failure probabilities and reliability indexes for all models.



a) b)

Figure 5.29. Residential building: a) pinned-pinned beams; b) pinned-fixed beams [121].

Through the analysis of this table, it is possible to conclude that obtained β-value considering the values from model identification until failure load is identical to the one considering nominal values. An increase on β-value is verified when considering the values from model identification in service phase. However, and according to Table 5.21, this model presents a low reliability. Accordingly, the most accurate result is the one considering the values from model identification until failure load.

In this example, the building is of class 2 (apartment building – risk to life, given a failure, is medium or economic consequences are considerable) and of class B (normal cost of safety measure), according to JCSS [93]. Therefore, a target reliability index (β_{target}) of 3.3 is recommended. This will permit to conclude that the assessed beam is safe.

Table 5.27. Safety assessment.

Numerical model	P_f	β
Nominal values	$2.55 * 10^{-4}$	3.48
Model identification (service)	$1.00 * 10^{-4}$	3.72
Model identification (failure)	$1.72 * 10^{-4}$	3.51

5.7.2. Pinned-fixed beams

In this situation, due to the fact of being one degree hyperstatic, the collapse mechanism is characterized by two plastic hinges, located at fixed support and beside the point load that is close to the pinned support. Therefore, the limit state function (Z) is composed by two equations, one for each plastic hinge. In this case, the resistance and loading model for maximum bending moment are respectively compared in each equation.

In this case, the resistance model is given by the failure load model (F_R), whose parameters are provided at Table 5.24, and by the maximum bending moment at fixed support model (M_{R1}^{*}), whose parameters are given at Table 5.25. A model is obtained for each analysis, respectively, considering the nominal values and those from model identification in service phase and until failure load, considering or not the Bayesian inference.

Therefore, it is necessary to transform these models into a model for maximum bending moment at fixed support (M_{R1}) and beside the point load that is close to the pinned support (M_{R2}). Consequently, the maximum bending moment at fixed support (M_{R1}^{**}) is computed for each generated value of failure load model (F_R), according to the static equilibrium equations, (5.30),

$$M_{R1}^{**} = 0.125 \cdot F_R \cdot L \tag{5.30}$$

If M_{R1}^{**} is lower than M_{R1}^{*}, then $M_{R1} = M_{R1}^{**}$ and M_{R2} is computed through the static equilibrium equations. If M_{R1}^{**} is higher than M_{R1}^{*}, then $M_{R1} = M_{R1}^{*}$. In this case, the load intensity (F_{R1}), necessary to obtain M_{R1}^{*}, is computed through equation (5.31),

$$M_{R1}^{*} = 0.125 \cdot F_{R1} \cdot L \rightarrow F_{R1} = 8.0 \cdot M_{R1}^{*} / L \tag{5.31}$$

In this case, M_{R2} is obtained in other way, (5.32),

$$\begin{aligned} &M_{R2} = 0.250 \cdot \left(F_R - F_{R1}\right) \cdot L + 0.1875 \cdot F_{R1} \cdot L \Leftrightarrow \\ &M_{R2} = 0.250 \cdot \left(F_R - 8.0 \cdot M_{R1}^{*} / L\right) \cdot L + 0.1875 \cdot 8.0 \cdot M_{R1}^{*} \Leftrightarrow \\ &M_{R2} = 0.250 \cdot F_R \cdot L - 0.5 M_{R1}^{*} \end{aligned} \tag{5.32}$$

Both M_{R1} and M_{R2} models depend on the beam span (L) which is, in this situation, 1.50 m. These models are represented by a Normal PDF. The further step consists in computing the maximum bending moment at fixed support (M_{S1}) through the static equilibrium equations, (5.33),

$$M_{S1} = \left(p \cdot L^2\right) / 8 \tag{5.33}$$

In this situation the maximum bending moment beside the point load that is close to the pinned support (M_{S2}) is computed through the static equilibrium equations. However, if M_{S1} is higher than M_{R1}, M_{S2} needs to be computed in other way. In this case, the load intensity (p_1), necessary to obtain M_{R1}, is computed through equation (5.34),

$$M_{R1} = \left(p_1 \cdot L^2\right)/8 \rightarrow p_1 = 8 \cdot M_{R1} / L^2 \tag{5.34}$$

In this case, M_{S2} is obtained through equation (5.35),

$$\begin{aligned} &M_{S2} = \left((p - p_1) \cdot L^2\right)/8 + 0.0625 \cdot p_1 \cdot L^2 \Leftrightarrow \\ &M_{S2} = \left(\left(p - 8 \cdot M_{R1} / L^2\right) \cdot L^2\right)/8 + 0.0625 \cdot 8 \cdot M_{R1} / L^2 \cdot L^2 \Leftrightarrow \\ &M_{S2} = p \cdot L^2 / 8 - 0.5 \cdot M_{R1} \end{aligned} \tag{5.35}$$

A Lognormal PDF is then adjusted to obtained data. A limit state function (Z), which compares resistance and loading curves, is then defined through equation (5.36),

$$Z = \begin{cases} M_{R1} - M_{S1} \\ M_{R2} - M_{S2} \end{cases} \tag{5.36}$$

The limit state is exceeded when loading is higher than resistance. The further step consists in generating values for each curve, according to each PDF parameters, and to register the number of values in which this limit state is exceeded in relation to the total number of evaluated points. The failure probability (P_f) is determined through equation (5.37),

$$P_f = P(Z \leq 0) \tag{5.37}$$

The reliability index (β) is then obtained, considering this value. A detailed description of how this index is computed is given in chapter four. Table 5.28 presents both failure probabilities and reliability indexes for all models [121].

Table 5.28. Safety assessment.

Numerical model	P_f	β
Nominal values	$1.40 * 10^{-5}$	4.20
Nominal values + Bayesian inference	$5.00 * 10^{-6}$	4.42
Model identification (service)	$4.22 * 10^{-5}$	3.93
Model identification (service) + Bayesian inference	$1.65 * 10^{-5}$	4.15
Model identification (failure)	$1.97 * 10^{-5}$	4.11
Model identification (failure) + Bayesian inference	$1.30 * 10^{-5}$	4.21

Through the analysis of this table, it is possible to conclude that obtained β-value considering the values from model identification until failure load is lower than the one considering nominal values. A decrease on β-value is verified when considering the values from model identification in service phase. An increase on β-value is verified with Bayesian inference. In this case, and according to Table 5.24 and 5.25, the most accurate result is the one considering nominal values and Bayesian inference.

In this example, the building is of class 2 (apartment building – risk to life, given a failure, is medium or economic consequences are considerable) and of class B (normal cost of safety measure), according to JCSS [93]. Therefore, a target reliability index (β_{target}) of 3.30 is recommended. This will permit to conclude that assessed beam is safe.

5.8. Conclusions

This chapter describes the probabilistic assessment of two different sets of reinforced concrete beams, a pinned-pinned and other pinned-fixed supported, which were tested in laboratory up to failure. In order to do so, a nonlinear numerical model was developed and further simplified, without changing too much its accuracy. A sensitivity analysis was further developed in order to identify the critical parameters, from material, geometry or physic source. Some of these parameters were detailed characterized at laboratory.

A model identification procedure was then executed to update the numerical model with measured data. To perform that an optimization technique, based in the evolutionary strategies algorithm in its plus version, was used. Both modelling and measurement errors were considered in an optimization algorithm stopping criteria. This process was developed for both service and failure region.

A nonlinear probabilistic analysis was then executed. In order to do so a PDF was defined for each critical parameter. Some of these PDF were then updated with results from laboratory characterization tests, through a Bayesian inference approach. In some cases, the statistical uncertainty was reduced with this process. A probabilistic analysis, based in a LHS procedure, was further executed.

From the probabilistic analysis it was obtained an updated resistance PDF for applied load, in the situation of pinned-pinned beams, and also for maximum bending moment at fixed support, for the case of pinned-fixed beams. It was then possible to probabilistically evaluate the experimental behavior of each tested beam by comparing those curves with obtained experimental data. These resistance PDF models are then used in a safety assessment example.

Some conclusions were obtained from probabilistic assessment: (1) model identification until failure load gives very good results (errors less than 10%); (2) model identification in service phase gives good results only for service region. Obtained results for failure region are bad. Complementary tests are thus recommended in this situation; (3) the most accurate models from a probabilistic analysis are those with values from model identification until failure load. Therefore the application of model identification before any probabilistic analysis is recommended; (4) Bayesian inference also increases the accuracy of probabilistic models. Therefore it is recommended the use of this procedure when complementary data is available.

6 Composite Beams

6.1. Introduction

The probabilistic assessment of two composite beams that were tested at laboratory, up to failure, is presented in this chapter. The purpose of such analysis is to probabilistically assess the behavior of composite structures, by taking into consideration all sources of uncertainty. In order to perform this, it was necessary to develop first a nonlinear numerical model.

This model was then simplified by reducing both the finite element mesh size and load step increment. Simplification is an important step as computational cost is a relevant issue on the use of such methodology. With the same purpose, a sensitivity analysis was executed too. The objective of such study is to evaluate the influence of each parameter on the overall structural behavior. The most important parameters were then identified.

Afterwards, model identification was developed. During this procedure, material (concrete and steel), geometry and physic (steel-concrete interaction) parameters are adjusted in an automatic way, so that numerical results best fit obtained experimental data. A robust optimization technique is then incorporated. Both modeling and measurement errors were introduced in this analysis. This procedure is important as it permit to obtain an updated deterministic numerical model of evaluated structure.

A nonlinear probabilistic analysis was developed. Therefore, it was necessary to define, according to bibliography, each input parameter probability density function (PDF). Those

PDF were then updated through a Bayesian inference procedure and taking into consideration results from laboratory characterization tests. Such tests were executed to determine some properties of used materials and of steel-concrete interface. An updated resistance PDF is then obtained, necessary for a safety assessment procedure.

Different resistance PDF of composite beams was then computed for different levels of updating, specifically, considering or not model identification and Bayesian inference procedures. Obtained results were then compared to measured data. The influence of both model identification and Bayesian inference procedures is pointed out on a safety assessment example.

6.2. Experimental tests

Tested composite beams are made up of a laminated steel profile connected to a solid lightweight concrete slab through headed stud steel connectors [118, 120, 121, 188, 189], according to Figure 6.1a. These beams, with a span length of 4.50 m (L), are pinned-pinned supported, according to Figure 6.1b.

The lightweight concrete slab is designed to resist to compressive stresses when the beam is submitted to bending moments. A lightweight concrete, classified as LC 50/55 according to EN 1992-1-1 [48], was used. The concrete slab presents a width of 350 mm (b_{slab}) and a thickness of 60 mm (h_{slab}).

Two layers of reinforcing steel wires spaced of 10 cm (4ϕ3.80), classified as S500B according to EN 1992-1-1 [48], were used in lightweight concrete slab, according to Figure 6.1a. This corresponds to a total steel area ($A_{s,l}$) of 0.91 cm^2. The concrete inferior (c_{inf}) and superior (c_{sup}) cover is of 10 mm.

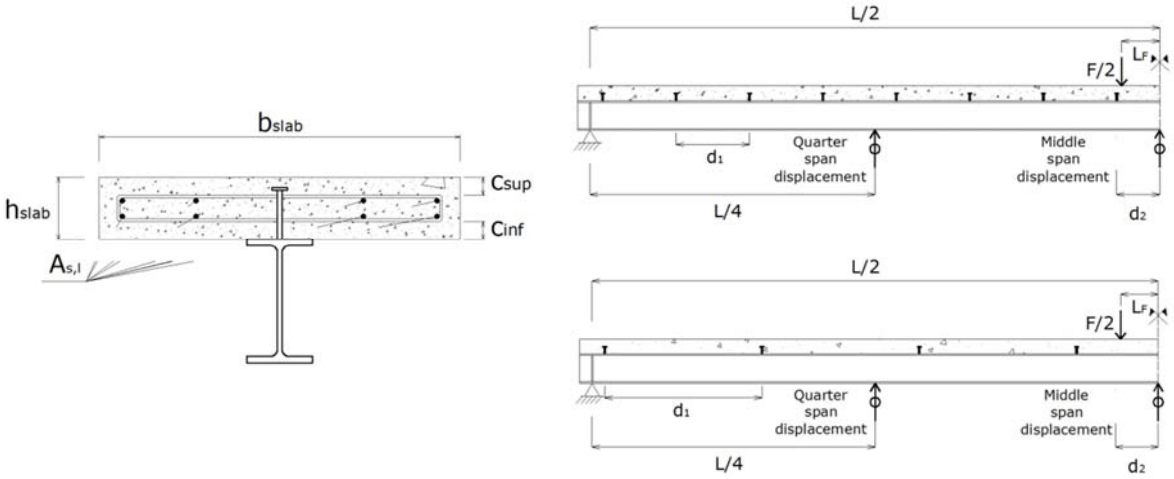
An IPE 120 laminated steel profile, in S275 steel according to EN 1993-1-1 [49], was chosen to guarantee that the composite cross section is of class 1, according to EN 1994-1-1 [50], and that the neutral axis is positioned at concrete slab when the beam is submitted to bending moments. The top surface of steel beam is greased with concrete mould releasing agent in order to eliminate adherence between steel and concrete.

The steel-concrete connection is provided by headed stud steel connectors. They are fabricated with steel type St-37-3-K according to regulation DIN 17 100 [159], which corresponds to S235J2G3+C450 according to EN 10025 [44, 131]. These connectors are welded to the steel beam and then concreted inside the lightweight concrete slab in order to guarantee that it works as a unique element.

One of the tested beams is designed for total connection, designated by beam 1, as the cross section ultimate strength does not depend on connection resistance [118, 120, 121, 188, 189]. In order to do that, shear studs are uniformly disposed along the beam span, guaranteeing a full shear connection. The adopted distribution is of 8 studs in half beam span. The considered distance between studs, and according to Figure 6.1b (top), is of 0.29 m (d_1) and 0.17 m (d_2). This means that failure occurs on a composite element, before the connection failure happens. This beam was tested with 31 days of concrete age.

The other tested beam, labeled as beam 2, presents a distribution of 4 studs in half beam span [188, 189]. The shear connection, in this situation, is only partial. The distance between studs is, and in agreement with Figure 6.1b (bottom), of 6.25 * 10^{-1} m (d_1) and 3.25 * 10^{-1} m (d_2). For this situation, a connection failure is expected. This beam was tested with 35 days of concrete age.

These beams were submitted to a short-term static load with two closely spaced concentrated loads (F) on beam middle span region, according to Figure 6.1b (L_F = 0.15 m), approaching to a single concentrate load [118, 120, 121, 188, 189]. Loads were applied by a hydraulic actuator positioned at beam middle span. This actuator is connected to a load cell with a resolution of 0.05%. A steel plate was used to divide the load cell into two equal loads. This division intends to avoid stress concentration on beam middle span and the possibility of concrete crushing. The supports, placed symmetrically, are materialized with two hinges which allow the support rotation. A Teflon sheet is placed in a hinge to permit the sliding in horizontal direction. The vertical displacement is restricted in both hinges.



a) b)

Figure 6.1. Scheme of tested beam, adapted from Valente [188]: a) transversal; b) longitudinal (half span).

Developed test is divided in four steps [188]: (1) load control: cyclic loading varying 25 times between 2 and 15 kN, with a rate of 0.50 kN/s; (2) load control: constant increasing load from 2 to 20 kN; (3) displacement control: linear increase of displacement at beam middle span, with a rate of 0.02 mm/s, until a total displacement of 70 mm; (4) displacement control: taking the beam up to failure, with a rate of 0.05 mm/s.

During laboratory test it was measured the applied load, by a load cell located inside the actuator, and the quarter and middle span vertical displacement, by different displacement transducers (LVDT) [118, 120, 121, 188, 189]. Used displacement transducers present a measurement field of ± 25.00 mm (quarter span) and of ± 50.00 mm (middle span), a sensitivity of 33.00 mV/V/mm (quarter span) and of 28.00 mV/V/mm (middle span) and an accuracy of 0.10 %. Figure 6.2 presents an image of the developed test.

Both beams present an identical collapse mechanism, characterized by a plastic hinge located in the region between applied loads (Figure 6.3a). In respect to obtained failure modes, beam 1 suffered a bending failure. Concrete crushes near the point load, with a longitudinal crack at middle height of concrete section, growing towards the beam middle span. The steel reinforcement near the crushing zone shows some local buckling. Beam 2 suffered a bending failure associated to a shear connection failure. Concrete crushes near the point load and, at final stages of test, stud failure takes place. For this beam, tensile cracks appear at bottom face of concrete slab. Additionally, a horizontal slip between steel profile and concrete slab is visible. Figure 6.3b and 6.3c, respectively, presents an image of obtained failure mode for beam 1 and 2.

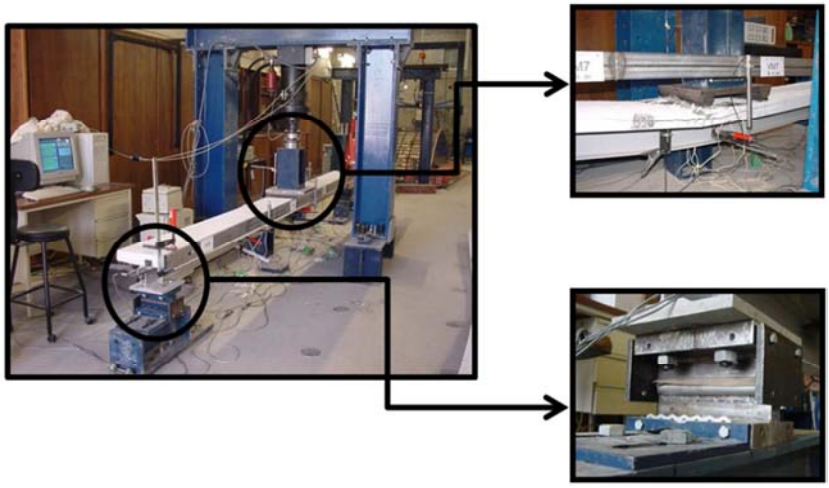
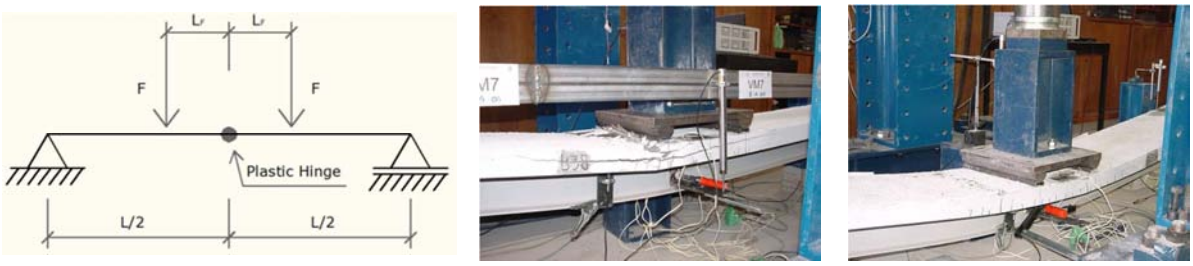


Figure 6.2. Experimental test [188].

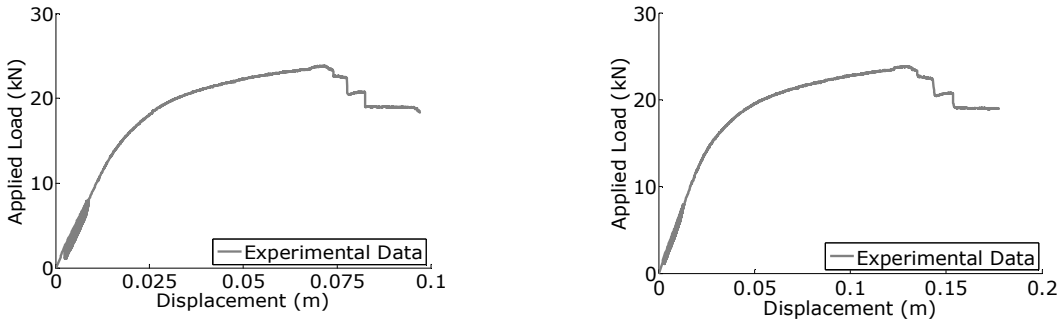


a) b) c)

Figure 6.3. Collapse mechanism [188]: a) scheme; b) beam 1; c) beam 2.

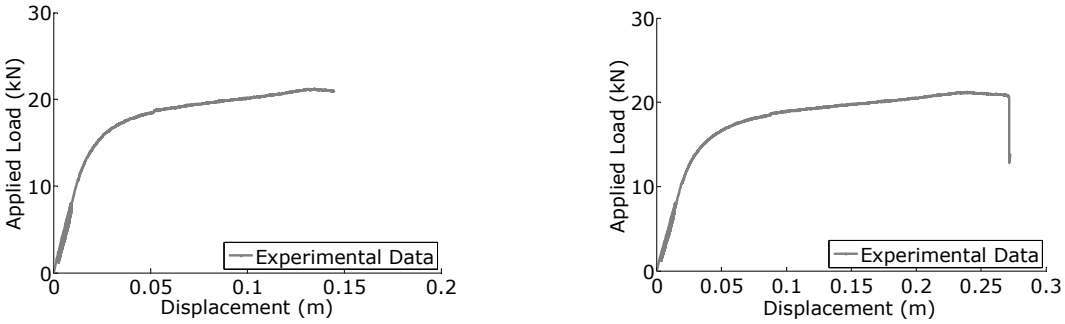
Obtained experimental results are presented in Figure 6.4 and 6.5, respectively, for beam 1 and 2. From the analysis of these results, an elastic behavior is observable at initial phase of the test, as there is an approximate linear relation between applied load and deflection value. This is expected as all materials present an initial elastic behavior too and a total compatibility between steel and concrete materials is verified for lower loads. A loss of stiffness is then verified as the increase in applied load is smaller while deformation keeps the same growing rate. Both beams present a ductile behavior due to a significant vertical deformation while the maximum applied load is kept almost constant.

For beam 1, with total connection design, failure is conditioned by concrete. In this situation it is verified that: (1) bending failure occurs before shear connection failure; (2) the connection behavior is rigid; and (3) the connection presents an ability to develop higher slip deformations when failure happens.



a) b)

Figure 6.4. Experimental data (beam 1): a) quarter span displacement; b) middle span displacement.



a) b)

Figure 6.5. Experimental data (beam 2): a) quarter span displacement; b) middle span displacement.

Beam 2 shows a lower stiffness as it presents larger deformation for the same load level. This aspect is more or less observable since the beginning of the test and was expected because less shear studs has the consequence of more load being applied to each connector, which results in higher deformation at the steel to concrete interface. In this situation, the beam load capacity is conditioned by the connection load capacity. It was also verified that shear connector failure occurs almost simultaneously with concrete slab crushing.

In experimental tests it was verified that slip values for partial connection (beam 2) are significantly higher than those for total connection (beam 1) [188]. In fact, partial connection design keeps bending failure from occurring previously to shear failure and, therefore, high values of slip develop before failure. Additionally, vertical deflection is influenced by connection deformability as a high vertical deflection is attained for partial connection. Ductile behavior of composite beams results not only from ductile behavior of steel acting together with lightweight concrete but also of shear connection ductile behavior.

The connection deformability also changes the longitudinal shear flow. The connectors positioned near supports become more loaded than those at beam middle span. As only small friction forces are developed between concrete slab and steel beam, due to applied mould releasing agent, all shear forces are transmitted through connectors. For partial connection, the connection deformability has an important influence on reducing the shear flow value. When connection behavior is no longer elastic, shear flow is reduced and an important loss of composite action takes place. Consequently, the composite beams flexural stiffness is affected by the loss of composite action between steel profile and concrete slab.

For both beams, the yielding of steel section lower fibbers should occur before concrete cracking on slab lower fibbers. The connection deformability and the loss of composite action

induce a redistribution of stress within the cross section, which results in successive changes of the neutral axis position. The yield of steel section results in a higher neutral axis. As the neutral axis initial position is supposed to be close to the concrete slab, it is probable that if it changes its position, tensile stresses will appear at concrete slab lower fibbers, leading it to crack. Table 6.1 presents, for each tested beam, the failure load (F_R) and the corresponding vertical displacement (δ_R).

Table 6.1. Failure load (F_R) and corresponding vertical displacement (δ_R) [188].

Beam	Failure load [kN]	Displacement [mm]
1	23.86	129.80
2	21.20	239.50

6.3. Numerical analysis

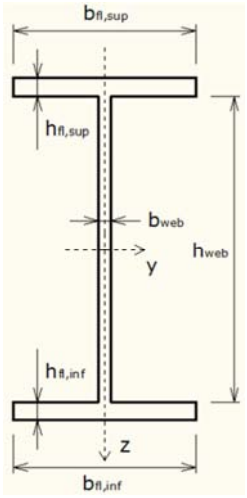
A nonlinear finite element model developed with software ATENA® [23, 24], that considers both shear connection and materials nonlinear behavior, was used to analyze the behavior of tested composite beams [118, 120, 121, 188, 189]. The defined geometry is previously presented in Figure 6.1, in accordance with experimental specimens. Used materials, concrete and steel, and also the interface law were defined by a set of parameters.

The existence of a curvature ratio between both flanges and the web of the IPE 120 laminated steel profile makes the numerical representation more difficult. In order to overpass it, a simplification is developed in order to take into account such curvature. Such simplification is made by equating the area and inertia of the real steel profile to an equivalent one. Therefore, the web height (h_{web}) and flanges thickness ($h_{fl,inf}$ and $h_{fl,sup}$) are determined, maintaining the flange width ($b_{fl,inf}$ and $b_{fl,sup}$) and the total height (h = 120.00 mm) of the real steel profile. Figure 6.6 represents the equivalent steel profile with the computed dimensions.

Figure 6.7a presents the concrete material stress-strain law. In the tensile region, the diagram is described by a linear initial step until tensile strength (f_{lt}) is reached and then by a decreasing exponential part. The fracture energy (G_{lf}) is proportional to the area of this region. In the compressive region, the behavior is characterized by an initial parabolic phase, until compressive strength (f_{lc}) is reached, and then by a decreasing linear part, designated by softening. The concrete elasticity modulus (E_{lc}), the compressive strain at compressive strength (ε_{lc}) and the critical displacement (w_{ld}) are parameters that define this region.

The same stress-strain law is used to simulate both reinforcement bars and laminated steel profile, as indicated in Figure 6.7b and 6.7c. This material presents an identical behavior

when in compression and tensile region. It is characterized by an initial linear phase in which the material presents a typical elastic behavior, following a hardening region, from material yielding until failure. The initial phase is characterized by an elasticity modulus ($E_{s,l}$ and $E_{s,p}$) until the yield strength is reached ($\sigma_{y,l}$ and $\sigma_{y,p}$). The second part is characterized in a different way for reinforcement and for steel profile. In the former case, it is defined by a limit strain ($\varepsilon_{lim,l}$) and strength ($\sigma_{u,l}$) while, for the latter, it is only described by a hardening modulus (H_p), presenting no failure criteria.



$b_{fl,inf}$ = 64.00 mm

b_{web} = 4.40 mm

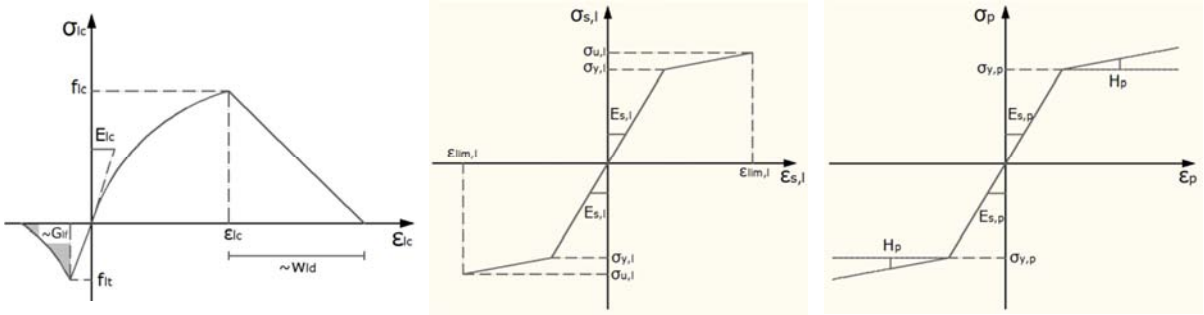
$b_{fl,sup}$ = 64.00 mm

$h_{fl,inf}$ = 6.60 mm

h_{web} = 106.80 mm

$h_{fl,sup}$ = 6.60 mm

Figure 6.6. Equivalent steel profile (nominal values).



a) b) c)

Figure 6.7. Stress-strain law: a) concrete; b) reinforcing steel; c) steel profile.

Used materials, concrete and steel, were modeled by an SBETA and a bilinear with hardening Von Mises material model, which are constitutive models of the ATENA® library [23, 24]. A biaxial stress failure criterion and a Von Mises yield criterion are respectively established. The nominal values, considered in numerical model, were those indicated at EN 1992-1-1 [48], for concrete and steel reinforcement, and in EN 1993-1-1 [49] for steel profile. Table 6.2, 6.3 and 6.4 present those values.

In respect to lightweight concrete material, its density (ρ) was estimated to be 1811.50 kg/m^3 according to bibliography [188]. For this material, the elasticity modulus (E_{lc}), the tensile strength (f_{lt}) and the compressive strain at compressive strength (ε_{lc}) can be determined through the following equations (6.1), (6.2) and (6.3) [48],

$$E_{lc} = E_c \cdot \eta_E = 37.0 \cdot 0.68 = 25.09 GPa \tag{6.1}$$

$$f_{lt} = f_t \cdot \eta_1 = 4.10 \cdot 0.89 = 3.67 GPa \tag{6.2}$$

$$\varepsilon_{lc} = k \cdot f_{lc} / \left(1.05 \cdot E_c \cdot \eta_E\right) = 1.00 \cdot 58.00 / \left(1.05 \cdot 37.00 \cdot 0.68\right) = 2.20‰ \tag{6.3}$$

being $\eta_E = (\rho/2200)^2 = 0.68$, $\eta_1 = 0.40 + 0.60 * \rho/2200 = 0.89$ and $k = 1.00$ (lightweight aggregate).

The connection between concrete slab and steel profile, provided by headed stud steel connectors, is modeled with an interface material model [23, 24]. This model is based on a Mohr-Coulomb criterion with tension cut-off. This law is given in terms of shear (τ) and normal stresses (σ). According to Figure 6.8a, the initial failure corresponds to the moment when cohesion (c) value is reached. After stress violates this condition, the relation between these stresses is given by the dry friction (ϕ) that is considered to be very low.

For shear stresses, and for positive slip (Δu_T), this law is characterized by an initial shear stiffness (K_{TT}), until the Mohr-Coulomb criterion is reached, and then it presents a minimum shear stiffness ($K_{TT,min}$) that is 1% of K_{TT} (Figure 6.8b). This behavior tries to replicate the relation between shear stress and slip at steel to concrete interface, measured in push-out tests. For normal stresses, and for positive uplifts (Δu_N), it is defined by an initial normal stiffness (K_{NN}) until the tensile strength (f_t) is reached. Once attained, the normal stress is reduced to 0, being this law defined by a minimum normal stiffness ($K_{NN,min}$) that is 1% of K_{NN} (Figure 6.8c).

Table 6.2. Material properties (concrete).

Parameter		Value
Elasticity modulus (E_{lc})	[GPa]	25.09
Tensile strength (f_{lt})	[MPa]	3.67
Compressive strength (f_{lc})	[MPa]	58.00
Fracture energy (G_{lf}) [23, 24]	[N/m]	91.75
Compressive strain at compressive strength (ε_{lc})	[‰]	2.20
Critical compressive displacement (w_{ld}) [23, 24]	[m]	$1.50 * 10^{-3}$

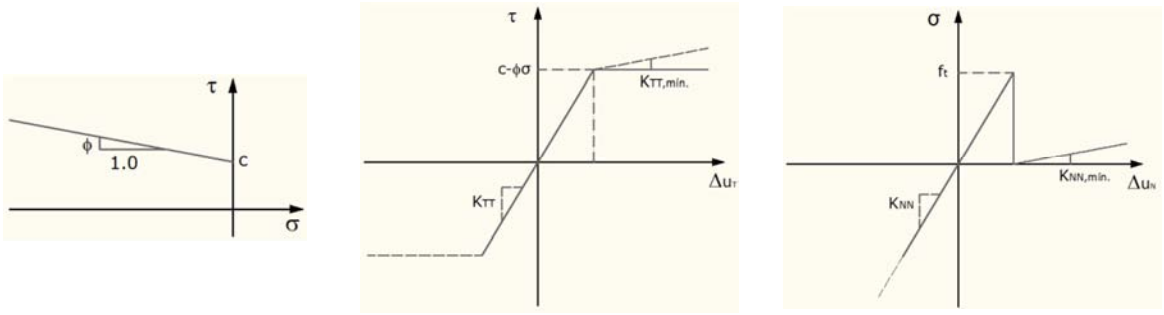
Table 6.3. Material properties (reinforcing steel).

Parameter		Value
Elasticity modulus ($E_{s,l}$)	[GPa]	200.00
Yield strength ($\sigma_{y,l}$)	[MPa]	500.00
Limit strength ($\sigma_{u,l}$)	[MPa]	540.00 (k = 1.08)
Limit strain ($\varepsilon_{lim,l}$)	[‰]	50.00

Table 6.4. Material properties (steel profile).

Parameter		Value
Elasticity modulus ($E_{s,p}$)	[GPa]	210.00
Yield strength ($\sigma_{y,p}$)	[MPa]	275.00
Hardening modulus (H_p)	[GPa]	1.04

The parameter values considered on this model are dependent on shear stud and concrete material. Normal stiffness (K_{NN}) and tensile strength (f_t) values are obtained from stud behavior in tension. Both these parameters are assumed to present high values in order to guarantee that the connection is working when submitted to normal stresses.



a) b) c)

Figure 6.8. Interface law: a) normal and shear stress; b) shear stress and slip c) normal stress and uplift.

The cohesion (c) or maximum stress at interface element, and shear stiffness (K_{TT}) values are respectively determined through equations (6.7) and (6.10) [118, 120, 121, 188, 189]. These values are different from beam 1 to beam 2 as the number of studs change. The cohesion value depends on the stud maximum load capacity (P_{Rm}). This value is determined according to EN 1994-1-1 [50] by using the following equations (6.4), (6.5) and (6.6),

$$P_{Rk} = 0.8 \cdot f_u \cdot \pi d^2 / 4 = 47.78\,kN \tag{6.4}$$

$$P_{Rk} = 0.29 \cdot \alpha \cdot d^2 \cdot \sqrt{f_{lck} \cdot E_{lcm}} = 53.19\,kN \tag{6.5}$$

$$\alpha = 0.2 \cdot \left(h_{sc} / d + 1 \right) = 0.97 \tag{6.6}$$

in which, d is the diameter of the stud shank equal to 13 mm, h_{sc} is the overall stud height equal to 50 mm, f_u is the ultimate tensile strength of stud material equal to 450 MPa, α is a parameter that depends from the ratio h_{sc} / d = 3.84, E_{lcm} is the concrete elasticity modulus (Table 6.2), f_{lck} the characteristic concrete compressive strength equal to 50 MPa. The characteristic value of stud maximum load capacity (P_{Rk} = 47.78 kN) is the minimum value of expression (6.4) and (6.5). The mean value (P_{Rm} = 53.09 kN) is then obtained by dividing this value per 0.90, according to EN 1994-1-1 [50]. By considering this value it is then possible to determine the cohesion nominal values for both situations of total (8 studs) and partial connection (4 studs) through equations (6.7), (6.8) and (6.9),

$$c = \frac{\left(stud\ \max imum\ load\ capacity \cdot number\ of\ studs\ in\ half\ span \right)}{\left(width\ of\ the\ \text{int}erface \cdot half\ span\ length \right)} \tag{6.7}$$

$$c = \frac{\left(53.09 \cdot 8 \right)}{\left(6.40 \cdot 10^{-2} \cdot 2.25 \right)} \cong 2.95\,MPa \tag{6.8}$$

$$c = \frac{\left(53.09 \cdot 4 \right)}{\left(6.40 \cdot 10^{-2} \cdot 2.25 \right)} \cong 1.47\,MPa \tag{6.9}$$

A similar procedure is considered to determine the shear stiffness (K_{TT}). In order to obtain such parameter it is necessary to compute the stud stiffness value. However, this value is extremely difficult to quantify as it depends from several factors. Accordingly, it was defined a value of 220 kN/mm based on bibliography [188, 189]. By taking this value into consideration it is then possible to determine the shear stiffness for both situations of total (8 studs) and partial connection (4 studs), through equations (6.10), (6.11) and (6.12). Table 6.5 presents the considered values to define the interface model.

$$K_{TT} = \frac{\left(stud\ stiffness \cdot number\ of\ studs\ in\ half\ span \right)}{\left(width\ of\ the\ \text{int}erface \cdot half\ span\ length \right)} \tag{6.10}$$

$$K_{TT} = \frac{\left(220 \cdot 8 \right)}{\left(6.40 \cdot 10^{-2} \cdot 2.25 \right)} \cong 12222.2(2)\ kN/m^2/mm \tag{6.11}$$

$$K_{TT} = \frac{\left(220 \cdot 4 \right)}{\left(6.40 \cdot 10^{-2} \cdot 2.25 \right)} \cong 6111.1(1)\,kN/m^2/mm \tag{6.12}$$

Table 6.5. Interface properties.

Parameter		Value	
		Beam 1	Beam 2
Normal stiffness (K_{NN})	[MPa]	$1.00 * 10^{7}$	$1.00 * 10^{7}$
Shear stiffness (K_{TT})	[MPa] (*per mm*)	12.2(2)	6.1(1)
Tensile strength (f_t)	[MPa]	$1.00 * 10^{2}$	$1.00 * 10^{2}$
Cohesion (c)	[MPa]	2.95	1.47

With the purpose of reducing the computational cost, only half of the beam was modeled, taking advantage of existent symmetry (Figure 6.1b). In order to do that it was necessary to introduce horizontal supports along the symmetry line. Additionally, a vertical support was included in the model to simulate real supports. A uniform finite element mesh composed by quadrilateral elements, for concrete slab and steel profile, by truss elements, embedded in concrete slab, for reinforcing steel, and by interface elements, for steel-concrete connection, was adopted [23, 24].

Additional considerations were taken when developing the numerical model, in order to adequately simulate the laboratory test. A displacement control numerical test was used in whole numerical analysis. Two different load cases were considered, respectively, one representing the real supports and other the applied displacement. A downward increment of $1.00 * 10^{-4}$ m was considered for the applied displacement. In order to avoid high local stresses in both support and load point, a steel plate was placed in such positions. A Newton-Raphson nonlinear search algorithm was used. Adopted parameters are present on Table 6.6. During the analysis, the quarter and the middle span displacements were monitored, according to Figure 6.1b, and also the applied load.

Table 6.6. Solution parameters (Newton-Raphson).

Solution method	Newton-Raphson
Stiffness / Update	Tangent / Each iteration
Iterations number limit	50
Error tolerance	$1.00 * 10^{-2}$
Line search	With iterations

The computational cost issue becomes relevant in both model identification and probabilistic analysis procedures, in which numerous analyses of the same numerical model are realized.

In order to overcome this problem, the developed numerical model is simplified by reducing the mesh size and increasing the step length. Therefore, three different meshes (18620, 4934 and 318 elements) and two load steps (270 steps with different factors and 700 steps of factor 2.0) are considered. The 270 steps are respectively divided in: (1) 15 steps of factor 20; (2) 10 steps of factor 15; (3) 10 steps of factor 10; (4) 45 steps of factor 5; and (5) 190 steps of factor 3. Figure 6.9 present a mesh constituted by 4934 elements. The performance of each model was studied with the same computer, in order to assure identical computational conditions. In this situation, the applied load error was computed through equation (6.13),

$$\Delta^i = \left(F_1^i - F_0^i\right) \Big/ F_0^i \,[\%] \tag{6.13}$$

where F_1^i indicates the applied load in step i and F_0^i is the applied load for the same step in the reference numerical model. This model is considered to be that with a more refined mesh and with a lower step length. The maximum and minimum Δ-values are then obtained and the sum of their absolute values is computed. Finally, the applied load error (θ) is determined by dividing this value per two. Obtained results are indicated in Table 6.7.

The chosen model for subsequent analysis, and according to obtained results, is the number 3. In fact, and by comparing with reference model number 0, models number 4 and 5 present a higher error. Moreover, model number 1 and 2 present a higher computational cost. The numerical model for beam 2 is the same as for beam 1. The only difference is that for beam 2 more load steps, respectively, 270 steps of factor 5, are added to those from model 3 as the vertical deflection increases, rising the computational cost too.

Table 6.7. Simplification results.

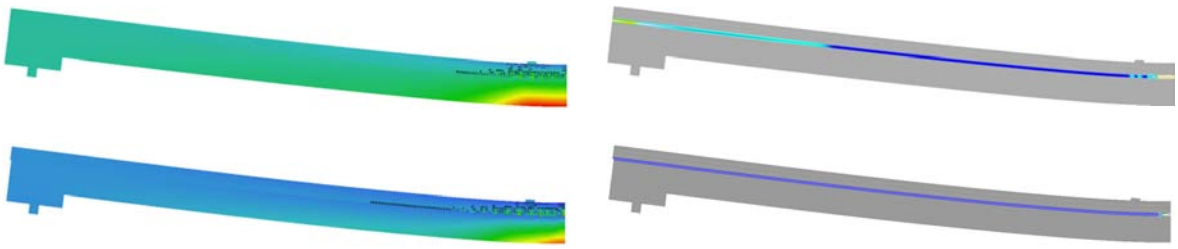
Numerical model	Finite element number	Step number	Computational time [s]	Applied load error - θ_F [%]
0	18620	700	39334.94	-
1	18620	270	28096.33	0.20
2	4934	700	5444.82	0.30
3	4934	270	2493.79	0.44
4	318	700	952.38	4.14
5	318	270	440.24	4.38

Figure 6.10a (top) indicates the deformation, crack pattern in concrete slab and horizontal strains of analyzed beam 1, for chosen numerical model and considering the nominal values.

In this case the collapse mechanism is characterized by the presence of a plastic hinge in the middle span region. A bending failure mode, with concrete crushing and yielding of steel profile is obtained. Figure 6.10b (top) indicates interface stresses between concrete slab and steel profile. It is possible to observe that cohesion value (blue) is only reached in a small region of the interface. Obtained behavior is similar to experimental one, validating the developed numerical model.

Figure 6.10a (bottom) provides the deformation, crack pattern in concrete slab and horizontal strains of analyzed beam 2, for chosen numerical model and considering the nominal values. The collapse mechanism is described by a plastic hinge which appears in middle span region. Figure 6.10b (bottom) indicates the interface stresses between concrete slab and steel profile. The cohesion value (blue) is reached along the whole interface. Obtained failure mode is bending with concrete crushing and yielding of steel profile, together with a lack of capacity to redistribute more shear stress along the steel-concrete interface. Obtained behavior is similar to experimental one, validating the developed numerical model.

Figure 6.9. Numerical model (finite element mesh).



a) b)

Figure 6.10. Numerical model (top: beam 1; bottom: beam 2): a) failure mechanism; b) interface stresses.

Both numerical models indicate that due to the existence of slip at steel to concrete interface a redistribution of stresses is verified at interface [188, 189]. A higher failure load is obtained for beam 1 (total connection). In this case, a total redistribution of stresses is verified. The shear stresses redistribution along the beam is possible due to the high deformability of connectors. Although, if such deformation capacity is not guaranteed, then connectors positioned close to supports suffer failure right after the maximum stress is installed. This is verified for beam 2 (partial connection).

6.4. Model identification

The aim of a model identification methodology is to rearrange a set of numerical parameters in such a way that the numerical response best fits the existent experimental data [118, 120, 121]. This fact converts this kind of analysis into a typical optimization problem. In this case, the optimization function is based in an approximation between numerical and experimental data, and the objective is to obtain the curve which best adapts to existent experimental data. A detailed description of this function is given in chapter four.

The optimization algorithm that was used during this analysis was the evolutionary strategies in its plus version [29]. It begins with an initial population of critical parameter values, generated randomly, and then, using the evolutionary operators, new populations are generated. A final population is extracted for each run. A detailed description of this algorithm is given at chapter two. This algorithm is processed with different starting points. An engineer judgment procedure, based in the probability of occurrence of each individual, is developed to determine the most suitable individual, from those previously extracted. This procedure is detailed described in chapter four.

When using this procedure, multiple runs of the same numerical model are necessary. In each run, the fitness function value, which characterizes the approximation between experimental and numerical curves, is computed. The identification stops when one of the previously defined algorithm stopping criteria is achieved. One of these criteria consists in establishing that the improvement on minimum fitness function value, obtained from two generations separated of a pre-specified gap, should be less than or equal to a threshold value. This value is computed through the law of propagation of uncertainty [90, 91, 92], detailed described at chapter four. It may be interpreted as the methodology precision, once obtained results become more accurate with its decrease.

Model identification computational cost depends on the number of parameters to be optimized. A sensitivity analysis is performed to identify the critical parameters, or, in other words, those that present a higher influence on the structural behavior [120]. This analysis consists in studying the fitness function variation with each input parameter. An importance measure (b_k) was then obtained for each evaluated parameter. A detailed description of this measure is given in chapter four. If this value is equal or higher than 10% (b_{lim}), the parameter will be considered as critical.

Two sensitivity analyses were developed, one for service phase and other until failure load. The analysis in service phase identified different combinations of values for critical parameters that lead to very good results. However, the majority of these combinations lead

to bad results in an analysis until failure load. This is important to highlight as model identification is typically performed with data from load test, and the majority of load tests are developed in service phase. Therefore, although a good approximation of registered data is obtained, this does not mean that the structure behavior until failure load is perfectly characterized. In order to overcome this, other complementary tests such as nondestructive tests (NDT), laboratory characterization tests and visual inspections should be executed.

Studied variables are related to material, geometry and physic parameters. It was respectively varied a standard deviation (σ) from each parameter mean value. In order to compute each standard deviation (σ), the following coefficient of variations (CV) were established [51, 93, 102, 135, 159, 190]: (1) concrete elasticity modulus (E_{lc}): 10%; (2) concrete tensile strength (f_{lt}): 20%; (3) concrete compressive strength (f_{lc}): 10%; (4) concrete fracture energy (G_{lf}): 10%; (5) concrete compressive strain at compressive strength (ε_{lc}): 10%; (6) concrete critical displacement (w_{ld}): 10%; (7) Steel profile elasticity modulus ($E_{s,p}$): 5%; (8) steel profile yield strength ($\sigma_{y,p}$): 5%; (9) steel profile hardening modulus (H_p): 20%; (10) reinforcing steel elasticity modulus ($E_{s,l}$): 5%; (11) reinforcing steel yield strength ($\sigma_{y,l}$): 5%; (12) reinforcing steel limit strength ($\sigma_{u,l}$): 5%; (13) reinforcing steel limit strain ($\varepsilon_{lim,l}$): 15%; (14) interface shear stiffness (K_{TT}): 10%; (15) interface cohesion (c): 12.5%; (16) steel profile bottom flange width ($b_{fl,inf}$): 2%; (17) steel profile web thickness (b_{web}): 2%; (18) steel profile top flange width ($b_{fl,sup}$): 2%; (19) concrete slab width (b_{slab}): 10%; (20) reinforcing steel area ($A_{s,l}$): 2%; (21) steel profile bottom flange thickness ($h_{fl,inf}$): 2%; (22) steel profile web height (h_{web}): 2%; (23) steel profile top flange thickness ($h_{fl,sup}$): 2%; (24) concrete slab height (h_{slab}): 10%; (25) inferior concrete cover (c_{inf}): 20%; (26) superior concrete cover (c_{sup}): 20%. Figure 6.11 and 6.12 gives the obtained results for beam 1.

From the analysis of Figure 6.11, it is possible to identify as critical parameters, respectively, those related to used materials, namely, concrete elasticity modulus (E_{lc}) and steel profile elasticity modulus ($E_{s,p}$), and geometric ones, as slab width (b_{slab}) and height (h_{slab}), the steel profile web thickness (b_{web}) and height (h_{web}), the superior flange width ($b_{fl,sup}$) and the inferior flange thickness ($h_{fl,inf}$). In fact, for lower intensity loadings, used materials, according to Figure 6.7a, 6.7b and 6.7c, are working in elastic region, being their elastic properties the most important ones. Accordingly, from 26 possible parameters, only 8 of them were considered, reducing so the computational cost in the model identification procedure.

The analyzes of Figure 6.12 shows that critical parameters identified during the analysis for service phase still present a significant influence in structural behavior until failure load. In this evaluation all concrete parameters become critical. From all these parameters it is

important to indicate those that describe its behavior in compression. These parameters present a higher impact as a bending failure with concrete crushing is identified. The same way, steel profile parameters, with exception of hardening modulus (H_p), are critical as steel material yields before concrete crushing. In respect to interface parameters, for higher loads, the maximum stress at interface element is reached in some regions. Consequently, the cohesion parameter (c) becomes an important parameter too. In a general way, the geometric parameters related to concrete slab and laminated steel profile dimensions, with exception of inferior flange width ($b_{fl,inf}$), present a high influence on structural behavior. In this situation, from 26 possible initial parameters, only 16 were considered in the study, reducing then the computational cost.

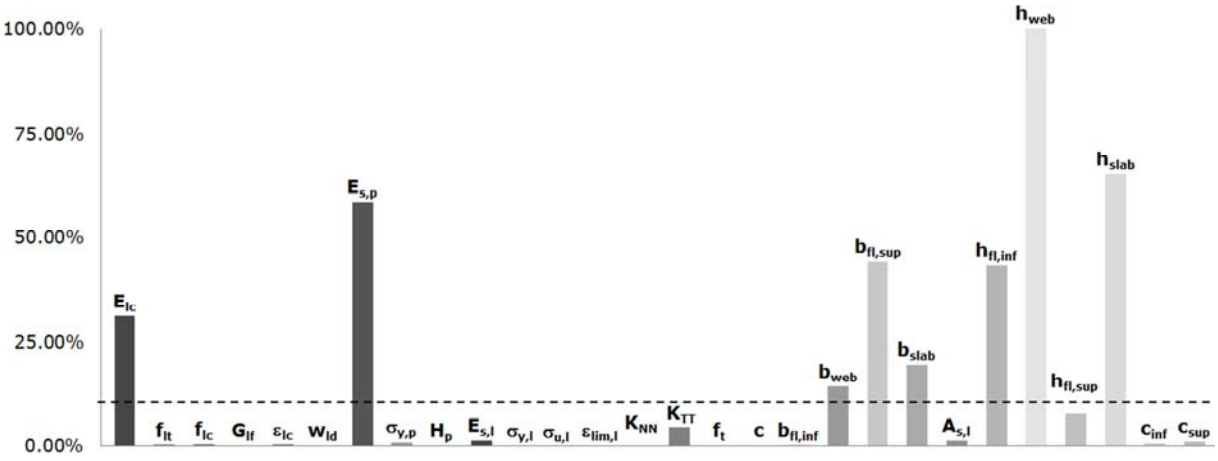


Figure 6.11. Importance factor (service).

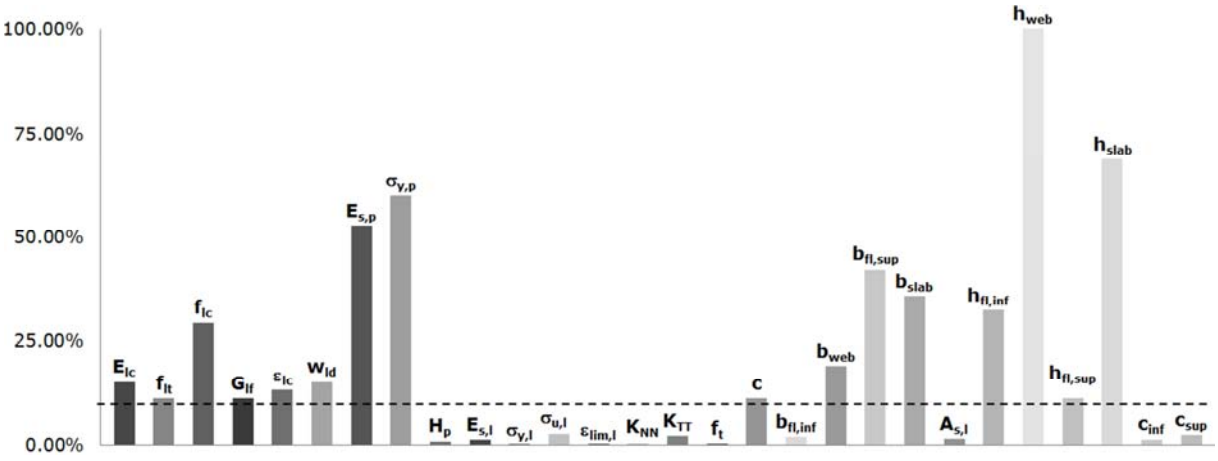


Figure 6.12. Importance factor (failure).

The sensitivity analysis performed for beam 2 is identical to that developed for beam 1. In fact, both numerical model and evaluated parameters are the same for these beams. The difference consists in some of the interface parameters, namely, shear stiffness (K_{TT}) and cohesion (c), for which the nominal values are different. The importance measures of these

two parameters are then determined for beam 2. A value of 0.00 % and of 9.99 %, for the analysis for service phase, and, of 13.62 % and of 4.04 %, for the analysis until failure load, respectively, for cohesion and shear stiffness parameters were obtained. As expected, there is an increase in shear stiffness importance measure in service region but not sufficient to be considered as critical.

Once the numerical model and critical parameters, to be optimized, are identified, the further step is the application of proposed model identification methodology. In this case, both quarter and middle span displacements were measured during the laboratory test (Figure 6.1b). The results from these two measurements are presented in two independent graphics, namely, the applied load (F) plotted against the quarter span displacement ($\delta_{1/4}$) and against the middle span displacement ($\delta_{1/2}$). The fitness function is, in this situation, a combination of two different measurements. It is characterized by the approximation between numerical and experimental values for applied loads. Before running the model identification methodology, it is necessary to determine the threshold value (ε) that defines the fitness function convergence criteria. In order to compute it, a division of uncertainty types, in experimental and numerical ones is necessary.

The sources of experimental uncertainty are, in this situation: (1) Sensor accuracy (0.20%) that includes the displacement transducer precision, and the cable and acquisition equipment losses [68, 154]; (2) Load intensity (0.10%) which includes the load cell resolution and the cable and acquisition equipment losses [68, 154]; (3) Load positioning, environmental effects and vibration noise which effects may be neglected in this case.

In this case, only the load intensity component will be considered when computing the experimental uncertainty for quarter span displacement. This component presents a uniform PDF (Type B) and so, according to JCGM [90, 91, 92], it should be divided by $\sqrt{3}$, obtaining then the result of $5.77*10^{-2}$ %. The experimental data is linearly dependent on this component ($\partial y^{exp}/\partial x$ = 1.00 kN). The experimental uncertainty is then computed through equation (6.14) [90, 91, 92],

$$u^2_{\exp_{\delta 1/4-F}} = 1.00^2 \cdot \left(5.77 \cdot 10^{-2}/100\right)^2 = 3.33 \cdot 10^{-7}\ \text{kN}^2 \rightarrow u_{\exp_{\delta 1/4-F}} = 5.77 \cdot 10^{-4}\ \text{kN} \tag{6.14}$$

An identical procedure is used for computing the experimental uncertainty for middle span displacement. This value is given by equation (6.15) [90, 91, 92],

$$u^2_{\exp_{\delta_{1/2}-F}} = 1.00^2 \cdot \left(5.77 \cdot 10^{-2}/100\right)^2 = 3.33 \cdot 10^{-7}\ \text{kN}^2 \rightarrow u_{\exp_{\delta_{1/2}-F}} = 5.77 \cdot 10^{-4}\ \text{kN} \tag{6.15}$$

In respect to numerical uncertainty, it is indicated the following sources: (1) Finite element method accuracy (0.34%, $\delta_{1/4}$ – F; 0.74%, $\delta_{1/2}$ - F), determined by comparing the previous

numerical model with other with a higher number of load steps [69]; (2) Mesh refinement (0.70%, $\delta_{1/4}$ - F; 1.42%, $\delta_{1/2}$ - F), obtained by comparing the previous numerical model with other with a more refined mesh [69]; (3) Model exactitude and considered hypothesis (e.g. consideration of supports as point loads) may be neglected in this situation.

In this case, both finite element method and mesh refinement effects will be considered when computing the numerical uncertainty for quarter span displacement. These components are represented by a uniform PDF (Type B) and so, according to JCGM [90, 91, 92], they should be divided by √3, obtaining then the result of 0.20% and of 0.40%, respectively. The effects of both components on numerical results are linear ($\partial y^{num}/\partial x$ = 1.00 kN). The numerical uncertainty is then computed through equation (6.16) [90, 91, 92],

$$u^2_{num_{\delta_{1/4}-F}} = 1.00^2 \cdot (0.20/100)^2 + 1.00^2 \cdot (0.40/100)^2 = 2.02 \cdot 10^{-5}\ \text{kN}^2 \rightarrow u_{num_{\delta_{1/4}-F}} = 4.49 \cdot 10^{-3}\ \text{kN} \quad (6.16)$$

An identical procedure is used to determine the numerical uncertainty for middle span displacement. This value is given by equation (6.17) [90, 91, 92],

$$u^2_{num_{\delta_{1/2}-F}} = 1.00^2 \cdot (0.43/100)^2 + 1.00^2 \cdot (0.82/100)^2 = 8.55 \cdot 10^{-5}\ \text{kN}^2 \rightarrow u_{num_{\delta_{1/2}-F}} = 9.25 \cdot 10^{-3}\ \text{kN} \quad (6.17)$$

Once the experimental and numerical uncertainties are computed, it will be possible to determine the fitness function uncertainty. In order to obtain this value, it is necessary to compute the partial derivative of the fitness function in respect to both experimental and numerical data. These values vary with tested beam as they are proportional to maximum applied load ($\partial f/\partial y^{num} = \partial f/\partial y^{exp} = 1/\max(y_i^{exp}) = 4.20*10^{-2}\text{kN}^{-1}$). The fitness function uncertainty in respect to quarter span displacement is given by equation (6.18) [90, 91, 92],

$$u^2_{\delta_{1/4}-F} = \left(4.20 \cdot 10^{-2}\right)^2 \cdot \left(5.77 \cdot 10^{-4}\right)^2 + \left(4.20 \cdot 10^{-2}\right)^2 \cdot \left(4.49 \cdot 10^{-3}\right)^2 = 3.60 \cdot 10^{-8} \rightarrow u_{\delta_{1/4}-F} = 1.90 \cdot 10^{-4} \quad (6.18)$$

The fitness function uncertainty in respect to middle span displacement is obtained through an identical procedure. This value is given by equation (6.19) [90, 91, 92],

$$u^2_{\delta_{1/2}-F} = \left(4.20 \cdot 10^{-2}\right)^2 \cdot \left(5.77 \cdot 10^{-4}\right)^2 + \left(4.20 \cdot 10^{-2}\right)^2 \cdot \left(9.26 \cdot 10^{-3}\right)^2 = 1.51 \cdot 10^{-7} \rightarrow u_{\delta_{1/2}-F} = 3.88 \cdot 10^{-4} \quad (6.19)$$

The global fitness function value is obtained through the square root of the sum of the square of these components. In order to determine the global uncertainty, the partial derivative of the fitness function in respect to each component is computed ($\partial f/\partial f_{\delta 1/2\text{-}F} = \partial f/\partial f_{\delta 1/4\text{-}F}$ = 1.00). This value is then computed through equation (6.20) [90, 91, 92],

$$u_f{}^2 = 1.00^2 \cdot \left(3.88 \cdot 10^{-4}\right)^2 + 1.00^2 \cdot \left(1.90 \cdot 10^{-4}\right)^2 = 1.87 \cdot 10^{-7} \rightarrow u_f = 4.32 \cdot 10^{-4} \quad (6.20)$$

The improvement on global fitness function minimum value (Δf) from two generations, separated of a specified gap (n), is given in chapter four. Accordingly, and assuming that the partial derivative of the improvement in respect to each component is unitary ($\partial \Delta f / \partial f_{i+n} = \partial \Delta f / \partial f_i$ = 1.00), it is possible to obtain the respective uncertainty value through equation (6.21) [90, 91, 92],

$$u_{\Delta f}{}^2 = 1.00^2 \cdot \left(4.32 \cdot 10^{-4}\right)^2 + 1.00^2 \cdot \left(4.32 \cdot 10^{-4}\right)^2 = 3.73 \cdot 10^{-7} \rightarrow u_{\Delta f} = 6.11 \cdot 10^{-4} \tag{6.21}$$

As all uncertainty sources are from Type B a coverage factor (k) of 2 should be adopted [90, 91, 92]. The fitness value criterion establishes that the respective improvement (Δf) should be less than or equal to the threshold value (ε). Correspondingly, this value is obtained by multiplying that determined from expression (6.21) by factor k. The obtained value from the analysis in service phase is determined in a similar way.

For beam 2, the same laboratory equipment and numerical model that was adopted for beam 1 is used. Consequently, the only difference on threshold values computation remains on the derivative of fitness function in respect to experimental and numerical results, as the maximum experimental load is different in this situation ($\partial f / \partial y^{num} = \partial f / \partial y^{exp}$ = 1/max(y_i^{exp}) = 4.70 * 10^{-1} kN^{-1}). Obtained threshold values, for beam 1 and 2, are indicated in equation (6.22),

$$\begin{aligned} &beam\ 1: \begin{cases} Service \rightarrow \varepsilon = 8.00 \cdot 10^{-4} = 0.08\% \\ Failure \rightarrow \varepsilon = 1.20 \cdot 10^{-3} = 0.12\% \end{cases} \\ &beam\ 2: \begin{cases} Service \rightarrow \varepsilon = 9.00 \cdot 10^{-4} = 0.09\% \\ Failure \rightarrow \varepsilon = 2.50 \cdot 10^{-3} = 0.25\% \end{cases} \end{aligned} \tag{6.22}$$

This means that, for instance, for beam 1, if the improvement in minimum fitness function value, until failure load, of a population from two generations separated of a specified gap (n) is, respectively, less than or equal to 0.12% than the algorithm stops, as the fitness function convergence criteria is achieved. This means that it is not meaningful to improve the fitness function of a value that is less than or equal to the precision itself.

The evolutionary strategies algorithm in its plus version [29] is further executed. In this case, it was defined a parent population (μ) and parent for recombination (ρ) of 10 and 20 individuals, and an offspring population (λ) of 50 and 75 individuals, respectively, for service and failure analysis. Other stopping criteria, like the maximum number of generations (1000), were also considered. The generation gap (n) used for the fitness function criteria is proportional to this number. It was established that this value is 10% of the specified maximum generation's number. Therefore, the improvement on minimum fitness value is

evaluated from a gap of 100 generations. Once the algorithm stops, a population, constituted by different individuals, is obtained.

The respective algorithm is processed with different starting points. An engineer judgment procedure is developed to determine the most suitable individual, from those previously extracted. This individual is constituted by a set of values, a value for each critical parameter. Tables 6.9 and 6.10 present the nominal values and individuals obtained from model identification in service phase and until failure load, respectively, for beam 1 and 2. In the same table, between brackets, the bias factor, which represents the ratio between the identified and the nominal value for each variable, is also presented. When applying this methodology in service phase, not only the critical parameters, as already identified during the sensitivity analysis, but also their optimal values, may differ from the application until failure load.

By analyzing the results from Table 6.8 it is possible to conclude that: (1) Obtained value for concrete parameters from model identification until failure load is close to the nominal one, with exception of the compressive strain at compressive strength (ε_{lc}). A higher value is obtained for compressive strength (f_{lc}) and for critical displacement (w_{ld}). A lower value is obtained for elasticity modulus (E_{lc}), for tensile strength (f_{lt}) and for fracture energy (G_{lf}). For model identification in service phase the elasticity modulus (E_{lc}) presents a higher value, far from the nominal one; (2) Obtained value for reinforcing steel parameter from model identification is slightly higher than the nominal one; (3) In respect to cohesion parameter, obtained value from model identification until failure load is close, although slightly higher, to the nominal one, (4) In respect to geometry parameters, obtained values from model identification are close to nominal ones. Exceptions are found for the inferior flange thickness ($h_{fl,inf}$), in situation of model identification in service phase, and for the web thickness (b_{web}), in the case of model identification until failure load.

While obtained values from model identification in service phase indicate a higher quality concrete material, those obtained from model identification until failure load indicate that its quality is close to the expected. In respect to steel material, the results revealed a higher quality than expected. Initial prediction of steel-concrete interface model is confirmed by model identification until failure load. Steel profile dimensions are close to those expected in design for both model identification procedures and for the majority of assessed parameters. The slab geometry is close to that predicted in design. Obtained values from model identification until failure load are the closest to nominal ones.

Table 6.8. Model identification results (beam 1).

Numerical model					Nominal value	Model identification	
						Service *	Failure *
Parameter	Material	Concrete	E_{lc}	[GPa]	25.09	30.00 (1.20)	23.71 (0.94)
			f_{lt}	[MPa]	3.67	3.67 (-)	3.56 (0.97)
			f_{lc}	[MPa]	58.00	58.00 (-)	59.19 (1.02)
			G_{lf}	[N/m]	91.75	91.75 (-)	91.18 (0.99)
			ε_{lc}	[‰]	2.20	2.20 (-)	2.69 (1.22)
			w_{ld}	[m]	$1.50 * 10^{-3}$	$1.50 * 10^{-3}$ (-)	$1.51 * 10^{-3}$ (1.01)
		Steel profile	$E_{s,p}$	[GPa]	210.00	230.00 (1.10)	215.65 (1.03)
			$\sigma_{y,p}$	[MPa]	275.00	275.00 (-)	297.98 (1.08)
	Physic		c	[MPa]	2.95	2.95 (-)	3.00 (1.02)
	Geometry		b_{web}	[mm]	4.40	4.20 (0.95)	5.22 (1.19)
			$b_{fl,sup}$	[mm]	64.00	63.00 (0.98)	63.95 (1.00)
			b_{slab}	[mm]	350.00	348.63 (1.00)	353.83 (1.01)
			$h_{fl,inf}$	[mm]	6.60	7.60 (1.15)	6.64 (1.01)
			h_{web}	[mm]	106.80	106.04 (0.99)	106.89 (1.00)
			$h_{fl,sup}$	[mm]	6.60	6.60 (-)	7.21 (1.09)
			h_{slab}	[mm]	60.00	61.26 (1.02)	62.14 (1.04)

* Bias factor is presented between brackets.

By analyzing the results from Table 6.9 it is possible to conclude that: (1) Obtained value for concrete parameters from model identification until failure load is close to the nominal one, with exception of the tensile strength (f_{lt}), compressive strain at compressive strength (ε_{lc}) and critical displacement (w_{ld}). A higher value is obtained, with exception of the tensile strength (f_{lt}). For model identification in service phase the elasticity modulus (E_{lc}) also presents a higher value, far from the nominal one; (2) Obtained value for steel profile elasticity modulus ($E_{s,p}$) from model identification is close to the nominal one. Obtained steel profile yield strength ($\sigma_{y,p}$) from model identification until failure load is higher than the nominal one; (3) In respect to cohesion parameter, obtained value from model identification until failure load is close, although slightly higher, to the nominal one, (4) In respect to geometry parameters, obtained values from model identification are close to the nominal

ones. An exception is verified for the inferior flange thickness ($h_{fl,inf}$) when developing the model identification in service phase.

Table 6.9. Model identification results (beam 2).

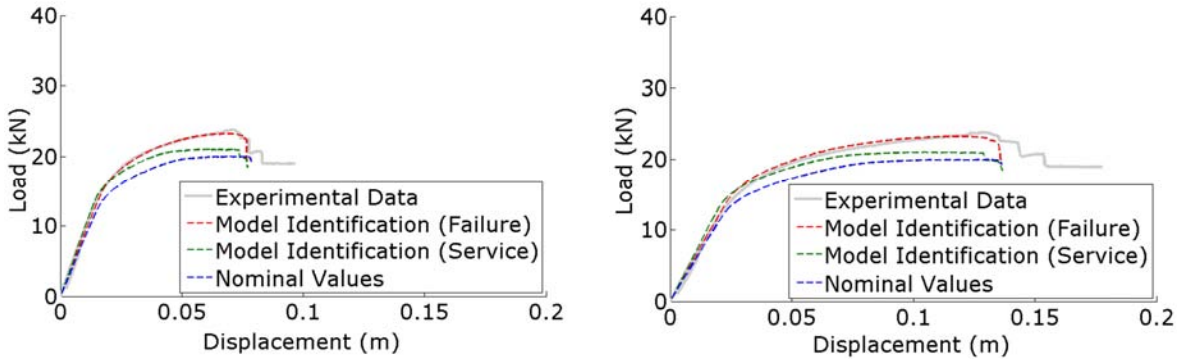
Numerical model					Nominal value	Model identification	
						Service *	Failure *
Parameter	Material	Concrete	E_{lc}	[GPa]	25.09	30.00 (1.20)	26.73 (1.07)
			f_{lt}	[MPa]	3.67	3.67 (-)	3.16 (0.86)
			f_{lc}	[MPa]	58.00	58.00 (-)	58.93 (1.02)
			G_{lf}	[N/m]	91.75	91.75 (-)	91.67 (1.00)
			ε_{lc}	[‰]	2.20	2.20 (-)	2.80 (1.27)
			w_{ld}	[m]	$1.50 * 10^{-3}$	$1.50 * 10^{-3}$ (-)	$1.71 * 10^{-3}$ (1.14)
		Steel profile	$E_{s,p}$	[GPa]	210.00	216.51 (1.03)	199.75 (0.95)
			$\sigma_{y,p}$	[MPa]	275.00	275.00 (-)	350.00 (1.27)
	Physic		c	[MPa]	1.47	1.47 (-)	1.55 (1.05)
	Geometry		b_{web}	[mm]	4.40	4.80 (1.09)	4.79 (1.09)
			$b_{fl,sup}$	[mm]	64.00	63.74 (1.00)	63.81 (1.00)
			b_{slab}	[mm]	350.00	354.91 (1.01)	349.56 (1.00)
			$h_{fl,inf}$	[mm]	6.60	7.60 (1.15)	6.50 (0.98)
			h_{web}	[mm]	106.80	106.92 (1.00)	106.86 (1.00)
			$h_{fl,sup}$	[mm]	6.60	6.60 (-)	6.60 (1.00)
			h_{slab}	[mm]	60.00	59.49 (0.99)	59.85 (1.00)

* Bias factor is presented between brackets.

Obtained values from model identification in service phase indicate a higher quality concrete material. However, those obtained from model identification until failure load indicate that its quality is close to the expected. Obtained values from model identification revealed a higher quality steel material than expected. Initial estimate for steel-concrete interface is confirmed by model identification until failure load. Steel profile dimensions are close to those expected in design for both model identification procedures and for most of assessed parameters. The slab geometry is close to that predicted in design. Obtained values from model identification until failure load are the closest to nominal ones.

Comparing both results, obtained for beam 1 and 2, it is possible to conclude the following: (1) Obtained values from model identification for beam 1 are the closest to nominal ones; (2) The concrete quality is close to initially predicted; (3) Reinforcing steel material presents a higher quality than initially expected; (4) Initial estimate of steel-concrete interface is confirmed; (5) Concrete slab dimensions are close to what was predicted in design; (6) Steel profile geometry is close to what was expected in design.

In Figure 6.13 and 6.14 the applied load (kN) is plotted against the quarter and the middle span displacement (m), respectively, for beam 1 and 2, for measured data and for numerical results, considering the nominal values, and those from model identification in service phase and until failure load. By studying both figures it is possible to conclude that model identification until failure load presents the numerical curve that best fits the experimental data.



a) b)

Figure 6.13. Numerical results (beam 1).

For beam 1, the obtained failure mode was identical in both experimental and numerical tests corresponding to a bending failure with concrete crushing and yielding of the steel profile. This beam presents an initial elastic behavior. In this situation a good agreement between experimental data and numerical results is obtained for all situations. A loss of stiffness is then verified for load values higher than 0.50 F_R. This fact is visible as the increase in load values is smaller while deformation keeps the same growing rate. For these ratios of load it is verified in numerical model that the steel section low fibbers are close to yielding strain which means that the first nonlinear behavior is conditioned by steel section. In respect to numerical results, this is the point where they start to differ from each other. Considering the experimental data to be the reference, the model which results first deviate from that ones, is the one with nominal values, for a load equivalent to 0.50 F_R. For a load equivalent to 0.70 F_R the first cracks appear at concrete slab. For this ratio, the obtained model from model identification in service phase starts to diverge. Finally, the obtained model from model

identification until failure load follows the experimental data until F_R. Consequently, the numerical model that presents a failure load close to F_R is the one due to model identification until failure load. The other two models present so lower loads, being the lowest failure load the one due to numerical model considering the nominal values.

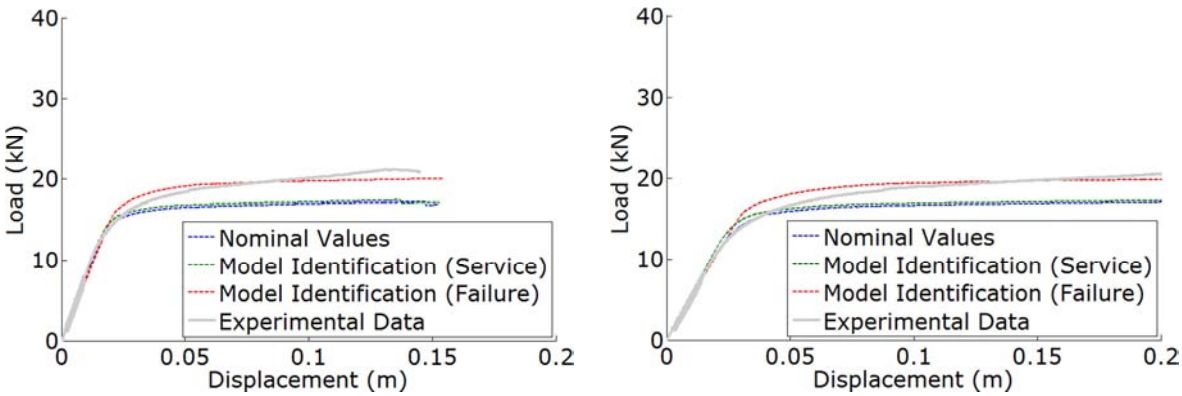


Figure 6.14. Numerical results (beam 2).

For beam 2, in both experimental and numerical tests the obtained failure mode was identical, respectively, a combination of a bending failure with concrete crushing and yielding of the steel profile with a lack of capacity of redistributing more shear stresses along the steel to concrete interface. When studying, in detail, its behavior, it is possible to conclude that it is similar to the one of beam 1. However, a loss of stiffness is verified for a lower ratio. In fact, the nonlinearity appears at ratios of around 0.45 F_{max}. For this situation, both the steel beam yielding and the concrete cracking are not conditioning the nonlinear behavior of composite beam, as the strain values for both materials are far from those limits. Therefore the nonlinear behavior is due to steel to concrete connection. The rest of its behavior until failure is identical to that described for beam 1. Nevertheless, this beam presents a lower failure load which results from the fact that the connectors shear capacity is accomplished at the whole interface, meaning that the beam has no more capacity to redistribute loads. Additionally, this beam presents a higher ductility as all the connection capacity is achieved.

Table 6.10 indicates the minimum fitness function values obtained by considering the nominal values and those from model identification in service phase and until failure load for beam 1. It is verified that obtained value from model identification in service phase is lower than that determined until failure. In fact, in service region, experimental and numerical results are closer than for higher applied loads. Nevertheless, for these two situations the applied methodology revealed an important improvement of this value. In fact, this improvement is, for both situations, higher than 80% which is really impressive.

Table 6.10. Minimum fitness function values (beam 1).

Numerical model	Fitness function			
	Service		Failure	
	Value [%]	Improvement [%]	Value [%]	Improvement [%]
Nominal values	4.65	-	19.35	-
Model identification	0.80	82.80	2.13	88.99

In Table 6.11 It is indicated the minimum fitness function values obtained by considering the nominal values and those from model identification in service phase and until failure load for beam 2. The improvement on fitness function value, obtained in this situation, is slightly lower than that determined for beam 1. In this situation, as the number of parameters to optimize is the same, this is due to an increase in the number of necessary points to compute the fitness function.

Table 6.11. Minimum fitness function values (beam 2).

Numerical model	Fitness function			
	Service		Failure	
	Value [%]	Improvement [%]	Value [%]	Improvement [%]
Nominal values	3.85	-	20.43	-
Model identification	0.95	75.33	5.82	71.51

Table 6.12 indicates failure load (F_R) and the corresponding displacement (δ_R), measured at beam middle span for beam 1. Obtained values from model identification until failure load are those that present a lower error. In fact, it is possible to verify that, when applying the methodology in service phase, the model identification is performed for this region, being not possible to guarantee the curve fitting for the failure region. Obtained error for the situation of model identification until failure load is less than 10%, which is considered to be very good.

Table 6.13 indicates the values obtained for beam 2. By analyzing these results, it is possible to conclude that model identification until failure load provides the most accurate failure load (F_R). Moreover, a higher improvement is verified for displacement at failure load (δ_R) when developing model identification in service phase. However, when considering this model, a smaller improvement on failure load is obtained. Therefore, the most accurate model is that given by model identification until failure load. This confirms the obtained results for beam 1.

Table 6.12. Failure load (F_R) and corresponding vertical displacement (δ_R) (beam 1).

Numerical model		Failure load		Displacement	
		Value [kN]	Error [%] *	Value [mm]	Error [%] *
Nominal values		19.99	16.22	122.66	5.50
Model identification	Service	21.06	11.74	105.03	19.08
	Failure	23.26	2.51	119.79	7.71

* Comparing with the real failure load and correspondent displacement.

Table 6.13. Failure load (F_R) and corresponding vertical displacement (δ_R) (beam 2).

Numerical model		Failure load		Displacement	
		Value [kN]	Error [%] *	Value [mm]	Error [%] *
Nominal values		17.18	18.97	262.76	9.71
Model identification	Service	17.41	17.88	244.06	1.90
	Failure	20.04	5.47	273.70	14.28

* Comparing with the real failure load and correspondent displacement.

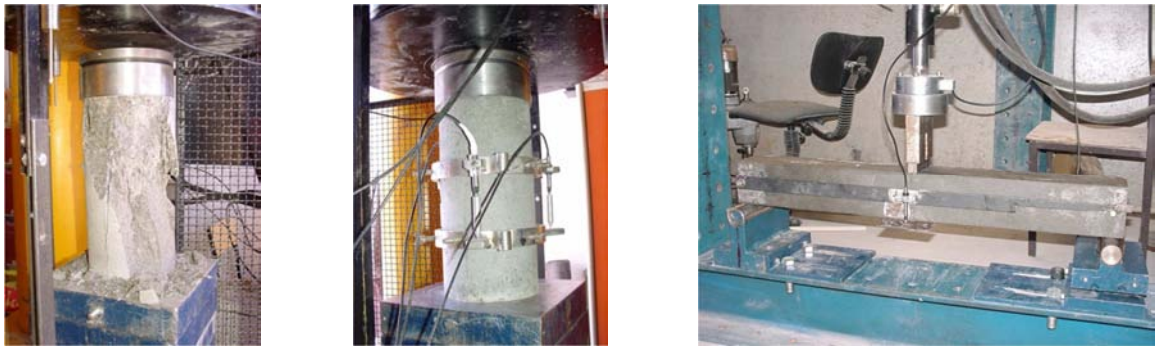
6.5. Characterization tests

6.5.1. Concrete material

The concrete in composite beams is a lightweight LC50/55 [48]. Some parameters that represent the stress-strain law curve were determined through characterization tests. Respectively, the compressive strength (f_{lc}) was obtained by uniaxial compression tests [136], the elasticity modulus (E_{lc}) by specific compression tests [109], and both the concrete tensile strength (f_{lt}) and fracture energy (G_{lf}) by fracture energy tests [155]. Additional tests were developed to determine the concrete over-dry density. It was obtained a mean value of 1811.5 kg/m^3 which confirmed the initial estimation.

The concrete specimens were produced at same time of corresponding beam and later tested at same date. The same mixture was used for every specimen but as casting was developed in different dates, slight variations on concrete properties were expected. Each cylinder from uniaxial compressive tests presents 300 mm of height and 150 mm of diameter (Figure 6.15a). Identical dimensions were adopted for tested cylinders to determine the elasticity modulus (Figure 6.15b). Each beam from fracture energy test present, as dimensions, 850 mm of length by 100 mm of height and 100 mm of width. A notch with

25 mm of depth and 5 mm of thickness was executed in each proof. These beams, with a span of 800 mm were submitted to a middle span point load until failure (Figure 6.15c).



a) b) c)

Figure 6.15. Laboratory tests [188]: a) uniaxial compression test; b) modulus of elasticity test; c) fracture energy test.

Compressive tests on cylinders proved that the compressive strength evolution in time is not significant, especially from 28 days. Two reasons explain this fact, respectively, the used cement CEM I 52.5 R, which confers high initial strength to concrete, and a lightweight aggregate characteristic referred by many authors [17]. Accordingly, it was considered on this study all values obtained from concrete cylinders with 28 days or more. It were used for such analysis the results from 203 different tests for compressive strength.

For elasticity modulus a similar conclusion was obtained. In fact, for the first days of concrete age this value tends to grow faster but, after 28 days of age, it tends to stabilize, growing slower. Accordingly, it was considered on this analysis all tests developed in concrete cylinders with 28 days or more. However, in this situation, the number of developed tests is a little lower. It was respectively considered the results from 186 tests.

Both tensile strength and fracture energy present a reduced evolution in time. This is specially verified for concrete specimens with 28 days or more. Consequently, all tests performed in specimens with an age equal or higher than 28 days were considered. Accordingly, the results from 5 fracture energy tests were adopted on this analysis.

A statistical analysis was then developed for each parameter. Both mean and standard deviation value were determined (Table 6.14). A bias value, which represents the ratio between the experimental and the nominal value, is also presented for each variable. Obtained values are close to one, which means that obtained data is close to nominal value. However, this is not verified for fracture energy (G_{lf}) as obtained values are lower than the nominal ones. Coefficients of variation (CV) are all less than 10% which indicates that the

variability of such parameters is small. The correlation coefficients (ρ_{ij}) between measured parameters were also determined. Such values are presented in Table 6.15.

Table 6.14. Concrete parameters.

Parameter		Nominal value	Mean value (μ) *	Standard deviation (σ)
Elasticity modulus (E_{lc})	[GPa]	25.09	24.81 (0.99)	2.23
Tensile strength (f_{lt})	[MPa]	3.67	3.78 (1.03)	0.16
Compressive strength (f_{lc})	[MPa]	58.00	58.32 (1.01)	4.97
Fracture energy (G_{lf})	[N/m]	91.75	78.42 (0.86)	1.83

* Bias factor is presented between brackets.

Table 6.15. Correlation coefficients (ρ_{ij}) in concrete.

	E_{lc}	f_{lt}	f_{lc}	G_{lf}
E_{lc}	1.00	*	0.70	*
f_{lt}	*	1.00	*	0.79
f_{lc}	0.70	*	1.00	*
G_{lf}	*	0.79	*	1.00

* These parameters present no correlation as they were determined by different laboratory tests.

6.5.2. Steel material

In order to characterize used steel material in stud connectors, reinforcement bars and laminated profile, uniaxial tensile tests were performed [139]. Tested specimens were sampled from the same stud connectors, reinforcement bars and laminated steel profile, used in push-out tests and in composite beams.

Due to the small size of headed studs (13 mm diameter), no experimental testing was done for steel properties of this connecting devices. However, it was considered the obtained values from tested studs with diameters of 19, 22 and 25 mm. The same occurred with reinforcement bars. In fact the diameter of ϕ3.80 was too small to be tested and, tests were so developed with diameter of ϕ5. 8 specimens of studs (Figure 6.16a), 3 of reinforcement bars (Figure 6.16b), and 3 of laminated steel profile (Figure 6.16c) were tested.

Some parameters that represent the stress-strain law curve were determined through these characterization tests. In respect to stud connectors, it is only present the results concerning the ultimate strength. These values will characterize the tensile strength (f_t) parameter used in interface model. A statistical analysis was developed for each parameter and the

respective mean and standard deviation were obtained. Results are presented on Table 6.16, 6.18 and 6.20. A bias value, which represents the ratio between the experimental and the nominal value, is also presented. The correlation coefficients (ρ_{ij}) between measured parameters were also determined. Results are given on Table 6.17 and 6.19.

a) b) c)

Figure 6.16. Laboratory tests [188]: a) stud; b) reinforcement; c) steel profile.

Obtained values indicate that bias factors are higher than one. This confirms that used material is of better quality than expected in design. Obtained CV values are, in general, lower than 5% which indicates a small variation of these properties. A higher CV of 12.50% is obtained for steel profile hardening modulus.

Table 6.16. Steel parameters (laminated profile).

Parameter		Nominal value	Mean value (μ) *	Standard deviation (σ)
Yield strength ($\sigma_{y,p}$)	[MPa]	275.00	335.67 (1.22)	9.10
Hardening modulus (H_p)	[GPa]	1.04	0.72 (0.69)	0.09

* Bias factor is presented between brackets.

Table 6.17. Correlation coefficients (ρ_{ij}) in steel (laminated profile).

	$\sigma_{y,p}$	H_p
$\sigma_{y,p}$	1.00	0.63
H_p	0.63	1.00

Table 6.18. Steel parameters (reinforcement).

Parameter		Nominal value	Mean value (μ) *	Standard deviation (σ)
Yield strength ($\sigma_{y,l}$)	[MPa]	500.00	583.41 (1.17)	8.02
Ultimate strength ($\sigma_{u,l}$)	[MPa]	540.00	606.06 (1.12)	8.32

* Bias factor is presented between brackets.

Table 6.19. Correlation coefficients (ρ_{ij}) in steel (reinforcement).

	$\sigma_{y,l}$	$\sigma_{u,l}$
$\sigma_{y,l}$	1.00	0.60
$\sigma_{u,l}$	0.60	1.00

Table 6.20. Steel parameters (stud connectors).

Parameter		Nominal value	Mean value (μ) *	Standard deviation (σ)
Tensile strength (f_t)	[MPa]	450.00	567.57 (1.26)	18.95

* Bias factor is presented between brackets.

6.5.3. Push-out tests

Several authors developed push-out tests with steel headed stud connectors in the past few years [8, 60, 61]. These tests allows for a rigorous analysis on the shear connection behavior by assessing the load-slip relation until failure and the respective failure mechanisms. Obtained failure mode for stud connectors are by shear failure of used connector, by local concrete crushing or by pull-out of a concrete cone.

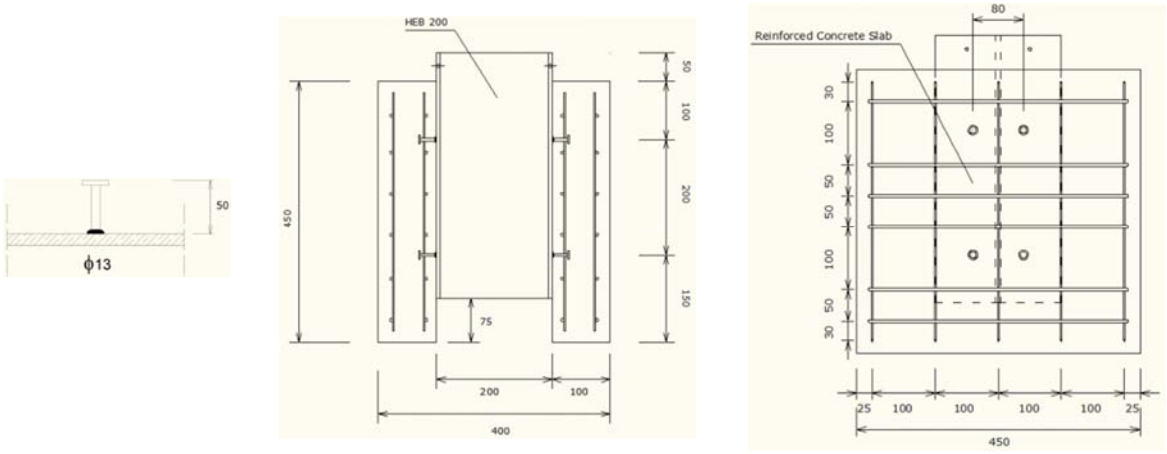
Valente [188] developed push-out tests to study different connection types. This test tries to simulate the transfer of shear forces at the steel to concrete interface of composite beams. At this point it is only described and presented the results from standard push-out tests (POST) on headed studs [50].

The geometry of tested specimens was the same. It was only considered specimens with 13 mm stud diameter as this is the diameter which is used in tested composite beams (Figure 6.17a). The specimens, presented at Figure 6.18a, consists of two lightweight concrete slabs held in vertical position, and of a steel HEB260 profile positioned between them, with welded studs concreted inside the slab. The slabs present as dimensions 450 mm x 450 mm x 100 mm. Two layers of ϕ5 reinforcement bars were embedded in concrete slab. Figure 6.17b and c gives a scheme of tested specimens.

Shear failure on studs is identified on all tested specimens. The behavior until failure between tested specimens is identical. It is characterized by an initial stiffer behavior, followed by a plastic behavior, with a constant or slow increasing load capacity. The maximum load values are very similar between developed tests. The lightweight concrete slabs were observed for identification of crack pattern. There is spread cracking around the stud position (Figure 6.18b). There is also some horizontal cracking between the two studs

positioned at same horizontal level. Studs achieve important deformation before failure. No cracking is observed at exterior face of concrete slab.

The first important result measured in these tests is the maximum applied load. This value, divided by the number of similar connectors in each specimen, corresponds to the maximum applied load (per connector) (P_{max}). Five different tests were respectively developed. Obtained mean is of 56.74 kN and standard deviation of 1.73 kN. Then, different values of this parameter are generated randomly, considering a Normal PDF and, by taking into account equation (6.7), different cohesion values are also obtained.



a) b) c)

Figure 6.17. Push-out tests (mm) [188]: a) headed stud configuration; b) and c) specimen geometry for POST tests.

a) b)

Figure 6.18. Laboratory tests [188]: a) testing frame; b) failure pattern on concrete slab.

The second important result is the connection stiffness (k) which is defined as a relation between the percentage of maximum applied load (per connector) and the correspondent slip value (s). These values are determined assuming an elastic behavior of connector until a

percentage of maximum applied load (per connector) is reached (*X* is variable), through equation (6.23) [188],

$$k\left(X\%P_{\max}\right)=P_{X\%P_{\max}}\Big/s_{X\%P_{\max}} \tag{6.23}$$

It is verified that all specimens present an elastic behavior until around 0.60 P_{max}. In this situation it was used 12 results for a % of maximum applied load between 0.40 and 0.65. A mean of 196.01 kN/m and a standard deviation of 22.80 kN/m were obtained. Then, different values of this parameter are generated randomly, considering a Normal PDF and, by taking into account expression (6.10), different tangent stiffness values are also obtained.

Table 6.21 and 6.22 presents the results from this analysis for both beams, respectively, with total (beam 1) and partial connection (beam 2). As the considered value of maximum applied load (per connector) (P_{max}) and of connection stiffness (k) is the same, the main difference between both beams consists in the number of studs in half span.

Table 6.21. Interface parameters (beam 1).

Parameter		Nominal value	Mean value (μ) *	Standard deviation (σ)
Shear stiffness (K_{TT})	[MPa] (*per mm*)	12.2(2)	10.89 (0.89)	1.27
Cohesion (c)	[MPa]	2.95	3.15 (1.07)	0.09

* Bias factor is presented between brackets.

Table 6.22. Interface parameters (beam 2).

Parameter		Nominal value	Mean value (μ) *	Standard deviation (σ)
Shear stiffness (K_{TT})	[MPa] (*per mm*)	6.1(1)	5.44 (0.89)	0.63
Cohesion (c)	[MPa]	1.47	1.58 (1.07)	0.05

* Bias factor is presented between brackets.

6.6. Probabilistic analysis

At this point a deterministic numerical model was developed and calibrated according to obtained experimental data by using a model identification procedure. The next step of this methodology consists in determining a reliable probabilistic numerical model that provides a PDF for resistance [118, 120, 121]. This curve may be further used for structural safety analysis, in which it is compared to a loading PDF [33, 123]. This analysis will be important to help engineers in any decision regarding structural safety.

The resistance curve of composite beams is obtained by considering the randomness of some parameters, as those related to concrete slab (material and geometry), steel profile (material and geometry) and interface between them. Used parameters were those considered to be critical from previous sensitivity analysis. This uncertainty is introduced in the probabilistic analysis by using a specific PDF for each parameter. The most used PDF are the Normal ones. In some situations, for which the parameters cannot assume negative values, a Lognormal PDF should be used instead. Those curves are defined by mean (μ) and standard deviation (σ) value. In this case, these values are based either in experience, when the information about the studied parameter is few, or in bibliography.

In respect to lightweight concrete parameters the existent information is few, since this material is recently used in civil engineer structures. The work of Leming [102], of Nowak *et al.* [135], of Valum and Nilsskog [190] and the results from the European Research Project EuroLightCon [51] are important sources of the needed information. For steel material, at laminated profile, and for slab and steel profile geometry the Probabilistic Model Code [93] indicates the needed values. Finally, for the stud maximum load capacity the present information is rare due to the fact of this capacity being dependent from used concrete, from connector material and from developed welding procedures. Therefore, a decision, based in experience, presents a very important role. Nevertheless, the work of Roik *et al.* [159] is considered.

In some situations, when there is complementary data, the values of these parameters may be updated through a Bayesian inference approach [15, 118, 120, 121]. This approach is detailed described in chapter three. In this case, the updating procedure is based in collected data from material (concrete and steel) and steel to concrete interface tests. The Bayesian inference was developed by considering both informative and non-informative (Jeffrey's) prior. Considered posterior PDF is the one that presents the lowest standard deviation.

Once each critical parameter PDF is defined, the next step consists in random generation of these parameter values to be used in a further probabilistic numerical analysis. This procedure is based in a random sampling technique designated by Latin Hypercube sampling (LHS) [144]. This procedure is detailed described at chapter four. Obtained results are a set of computed values that can be statistically processed and adjusted to a Normal PDF.

The beam behavior is strictly monitored during the whole analysis. Such procedure consists in evaluating if each measured load for a specific displacement is within the numerical 95% confidence interval. An index-*i* is then used to study this. Another index, indicated as

index-*p*, is used to evaluate the accuracy of obtained resistance PDF [118, 120]. A description of these indexes is provided at chapter four.

A Normal and Lognormal PDF were considered for material, geometric and physic properties of these beams. The mean values are the nominal ones or those obtained from model identification in service phase and until failure load. The adopted coefficients of variation were the same of previous sensitivity analysis. Those values are indicated on Table 6.23.

Table 6.23. Parameter values (beam 1).

Parameter		Nominal value *		Model identification (service) *		Model identification (failure) *	
		μ	σ	μ	σ	μ	σ
E_{lc}	[GPa]	25.09 (24.81)	2.51 (2.20)	30.00 (24.81)	3.00 (2.20)	23.71 (24.81)	2.37 (2.20)
f_{lt}	[MPa]	3.67 (3.78)	0.73 (0.28)	3.67 (3.78)	0.73 (0.28)	3.56 (3.78)	0.71 (0.28)
f_{lc}	[MPa]	58.00 (58.31)	5.80 (5.02)	58.00 (58.31)	5.80 (5.02)	59.19 (58.31)	5.92 (5.02)
G_{lf}	[N/m]	91.75 (78.33)	9.18 (7.13)	91.75 (78.33)	9.18 (7.13)	91.18 (78.33)	9.12 (7.13)
ε_{lc}	[‰]	2.20	0.22	2.20	0.22	2.69	0.27
w_{ld}	[m]	$1.50*10^{-3}$	$0.15*10^{-3}$	$1.50*10^{-3}$	$0.15*10^{-3}$	$1.51*10^{-3}$	$0.15*10^{-3}$
$E_{s,p}$	[GPa]	210.00	10.50	230.00	11.50	215.65	10.78
$\sigma_{y,p}$	[MPa]	275.00 (337.61)	13.75 (37.75)	275.00 (337.61)	13.75 (37.75)	297.98 (337.61)	14.90 (37.75)
c	[MPa]	2.95 (3.12)	0.37 (0.10)	2.95 (3.12)	0.37 (0.10)	3.00 (3.08)	0.38 (0.10)
b_{web}	[mm]	4.40	0.09	4.20	0.08	5.22	0.10
$b_{fl,sup}$	[mm]	64.00	1.28	63.00	1.26	63.95	1.28
b_{slab}	[mm]	350.00	35.00	348.63	34.86	353.83	35.38
$h_{fl,inf}$	[mm]	6.60	0.13	7.60	0.15	6.64	0.13
h_{web}	[mm]	106.80	2.14	106.04	2.12	106.89	2.14
$h_{fl,sup}$	[mm]	6.60	0.13	6.60	0.13	7.21	0.14
h_{slab}	[mm]	60.00	6.00	61.26	6.13	62.14	6.21

* Bayesian inference values are presented between brackets.

In such Table, and between brackets, it is presented the results from a Bayesian inference procedure. In this situation, both materials (concrete and steel) and the interface parameter cohesion were updated. In a general way, the Bayesian updating provided mean results close to the nominal values and those from model identification. Exceptions are the concrete fracture energy (G_{lf}) and steel profile yield strength ($\sigma_{y,p}$). Moreover, and with exception of steel profile yield strength ($\sigma_{y,p}$), for which obtained experimental data is far from numerical

results, the Bayesian inference procedure reduced the standard deviation values. The Jeffrey's prior is used in almost all situations, except for interface cohesion (*c*) in model identification until failure.

Table 6.24 indicates the parameter values for beam 2, respectively, for material, geometric and physic properties. A Normal and Lognormal PDF were considered for those parameters. The mean values are the nominal ones or those from model identification in service phase and until failure load. Considered coefficients of variation are the same as those used in previous sensitivity analysis. In the same table, and between brackets, are indicated the Bayesian inference values. In a general way, obtained results are similar to those for beam 1. In this situation, the steel profile yield strength ($\sigma_{y,p}$), from model identification until failure load, is reduced with Bayesian inference. In this situation, the conjugate informative prior is used. In the other cases the Jeffrey's prior is used.

Table 6.24. Parameter values (beam 2).

Parameter		Nominal value *		Model identification (service) *		Model identification (failure) *	
		μ	σ	μ	σ	μ	σ
E_{lc}	[GPa]	25.09 (24.81)	2.51 (2.21)	30.00 (24.81)	3.00 (2.21)	26.73 (24.81)	2.67 (2.21)
f_{lt}	[MPa]	3.67 (3.78)	0.73 (0.28)	3.67 (3.78)	0.73 (0.28)	3.16 (3.78)	0.63 (0.28)
f_{lc}	[MPa]	58.00 (58.31)	5.80 (5.02)	58.00 (58.31)	5.80 (5.02)	58.93 (58.31)	5.89 (5.02)
G_{lf}	[N/m]	91.75 (78.33)	9.18 (7.13)	91.75 (78.33)	9.18 (7.13)	91.67 (78.33)	9.17 (7.13)
ε_{lc}	[‰]	2.20	0.22	2.20	0.22	2.80	0.28
w_{ld}	[m]	$1.50*10^{-3}$	$0.15*10^{-3}$	$1.50*10^{-3}$	$0.15*10^{-3}$	$1.71*10^{-3}$	$0.17*10^{-3}$
$E_{s,p}$	[GPa]	210.00	10.50	216.51	10.83	199.75	9.99
$\sigma_{y,p}$	[MPa]	275.00 (337.61)	13.75 (37.75)	275.00 (337.61)	13.75 (37.75)	350.00 (342.81)	17.50 (24.63)
c	[MPa]	1.47 (1.56)	0.18 (0.07)	1.47 (1.56)	0.18 (0.07)	1.55 (1.56)	0.19 (0.07)
b_{web}	[mm]	4.40	0.09	4.80	0.10	4.79	0.10
$b_{fl,sup}$	[mm]	64.00	1.28	63.74	1.27	63.81	1.28
b_{slab}	[mm]	350.00	35.00	354.91	35.49	349.56	34.96
$h_{fl,inf}$	[mm]	6.60	0.13	7.60	0.15	6.50	0.13
h_{web}	[mm]	106.80	2.14	106.92	2.14	106.86	2.14
$h_{fl,sup}$	[mm]	6.60	0.13	6.60	0.13	6.60	0.13
h_{slab}	[mm]	60.00	6.00	59.49	5.95	59.85	5.99

* Bayesian inference values are presented between brackets.

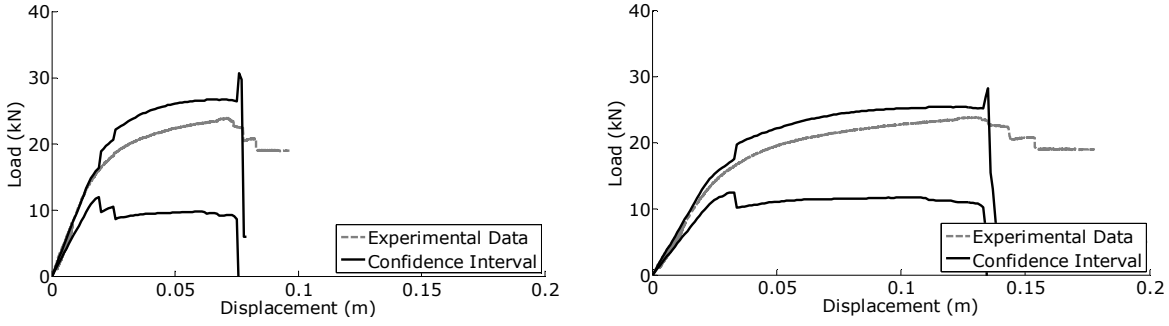
Table 6.25 indicates the used correlation values. Inside brackets are presented the real values, obtained from characterization tests. These values are used in the analysis considering the Bayesian inference. It is verified, in a general way, that real correlations are not as strong as it was predicted in theory.

Table 6.25. Correlation coefficients (ρ_{ij}).

	E_{lc}	f_{lt}	f_{lc}	G_{lf}	$E_{s,p}$	$\sigma_{y,p}$
E_{lc}	1.00	0.70	0.90 (0.70) *	0.50	0.00	0.00
f_{lt}	0.70	1.00	0.80	0.90 (0.79) *	0.00	0.00
f_{lc}	0.90 (0.70) *	0.80	1.00	0.60	0.00	0.00
G_{lf}	0.50	0.90 (0.79) *	0.60	1.00	0.00	0.00
$E_{s,p}$	0.00	0.00	0.00	0.00	1.00	0.80
$\sigma_{y,p}$	0.00	0.00	0.00	0.00	0.80	1.00

* Real values are presented between brackets.

During the probabilistic analysis it was evaluated the applied load for each measured and computed quarter and middle span displacement. In Figure 6.19 and 6.20 are presented the results that correspond to a situation of nominal parameter values, respectively, for beam 1 and 2 models. Obtained index-*i* were, respectively, of 89.10% (beam 1) and 91.59% (beam 2), for quarter span, and of 89.75% (beam 1) and 92.65% (beam 2), for middle span displacement. Obtained values are high which indicate that most of measured points fall into computed confidence interval.



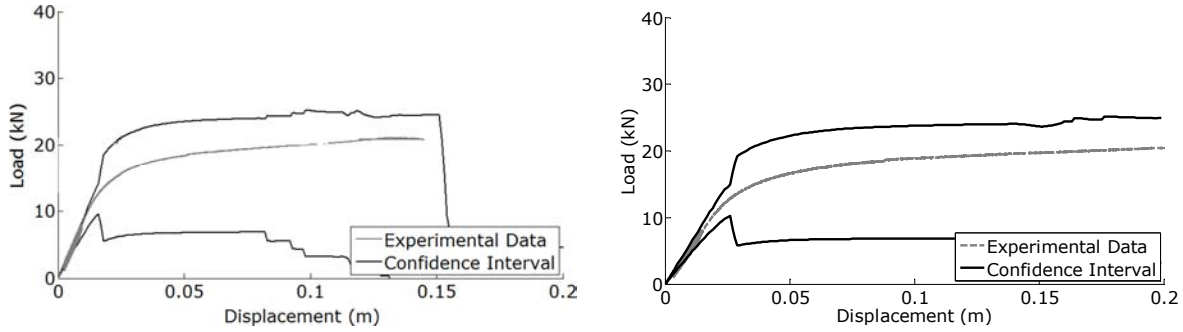
a) b)

Figure 6.19. Nominal values (beam 1) without Bayesian inference: a) quarter span; b) middle span.

The application of a Bayesian inference procedure to nominal data values lead to the results presented in Figure 6.21 and 6.22. In this situation obtained index-*i* were of 94.18% (beam 1) and 96.07% (beam 2), for quarter span and, of 95.73% (beam 1) and 96.73% (beam 2), for

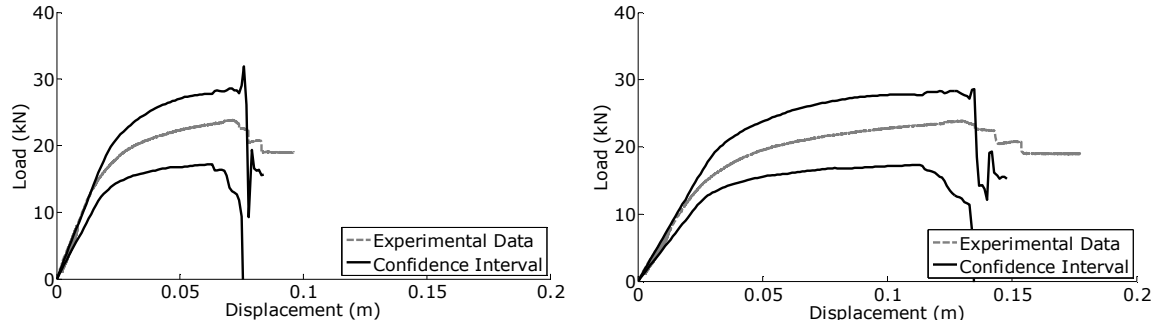
middle span displacement. These values are higher than those presented before, which indicates an increase on the accuracy of both models due to Bayesian inference.

In Figure 6.23 and 6.24 are presented the results with values from model identification in service phase. It is important to mention that, for this situation, excellent results were obtained for index-*i*, respectively, 94.31% (beam 1) and 94.74% (beam 2) for quarter span, and of 94.56% (beam 1) and 94.28% (beam 2) for middle span displacement. By comparing with obtained results with nominal parameter values it is possible to conclude that there is an improvement on the accuracy of numerical model with model identification.



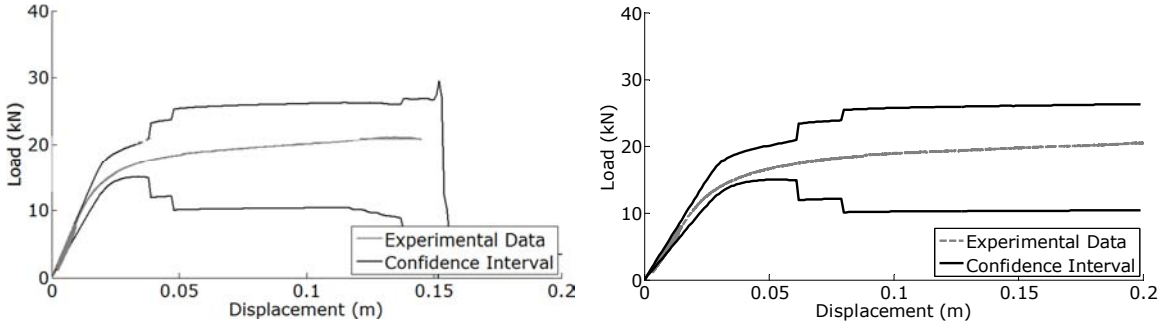
a) b)

Figure 6.20. Nominal values (beam 2) without Bayesian inference: a) quarter span; b) middle span.



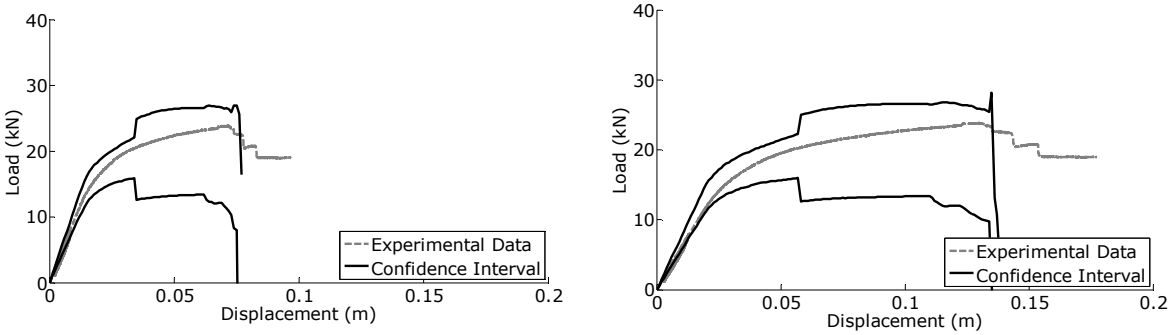
a) b)

Figure 6.21. Nominal values (beam 1) with Bayesian inference: a) quarter span; b) middle span.



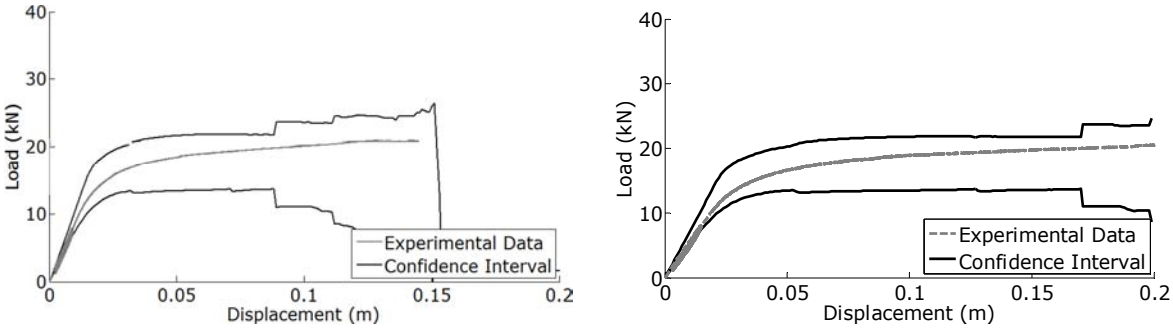
a) b)

Figure 6.22. Nominal values (beam 2) with Bayesian inference: a) quarter span; b) middle span.



a) b)

Figure 6.23. Model identification (service) (beam 1) without Bayesian inference: a) quarter span; b) middle span.



a) b)

Figure 6.24. Model identification (service) (beam 2) without Bayesian inference: a) quarter span; b) middle span.

Figure 6.25 and 6.26 indicate obtained results for the situation of data values obtained from modal identification in service phase, but taking into consideration the Bayesian inference procedure. Obtained index-*i* were of 95.73% (beam 1) and 94.76% (beam 2) for quarter

span, and of 95.95% (beam 1) and 95.28% (beam 2) for middle span displacement. In this situation the inference increased the accuracy of obtained models.

Figure 6.27 and 6.28 indicate the results obtained by considering data values from model identification until failure load. The index-*i* were of 94.56% (beam 1) and 95.55% (beam 2), for quarter span, and of 93.76% (beam 1) and 96.62% (beam 2), for middle span displacement. It was verified that both models provided, in this situation, excellent results, and, with exception of beam 1 for middle span displacement, an increase on its accuracy is obtained with model identification.

The application of a Bayesian inference procedure to data values determined by model identification until failure load provided the results presented in Figure 6.29 and 6.30. For this situation the index-*i* were of 96.02% (beam 1) and 97.10% (beam 2), for quarter span, and of 96.74% (beam 1) and 97.59% (beam 2), for middle span displacement. An increase on its accuracy is obtained with inference, for all analysed beams. Obtained models, and by comparing with the others, are the most accurate ones.

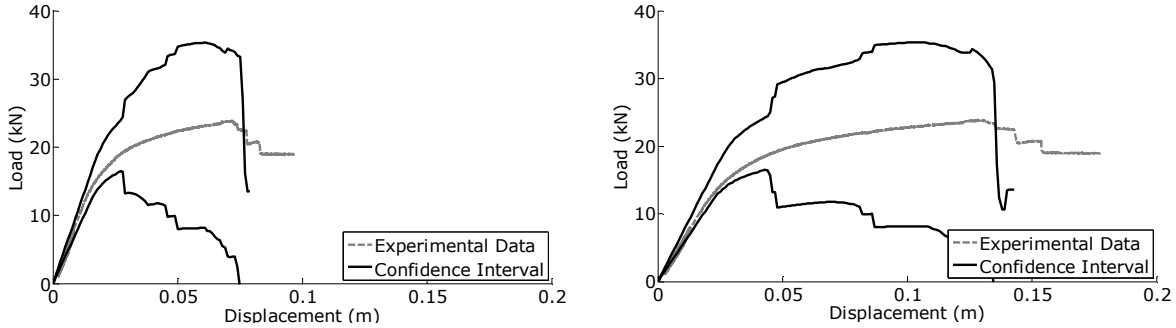


Figure 6.25. Model identification (service) (beam 1) with Bayesian inference: a) quarter span; b) middle span.

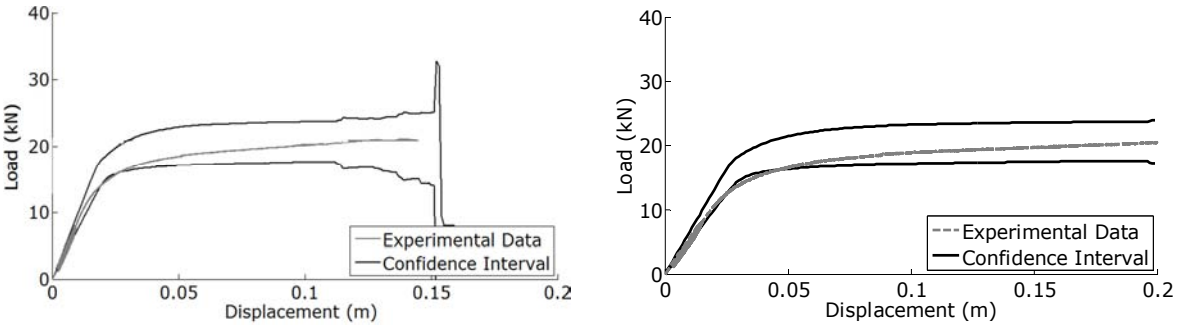
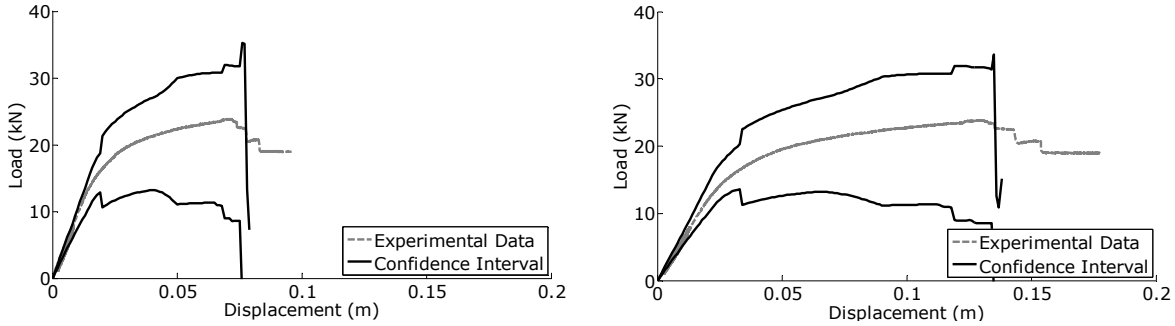
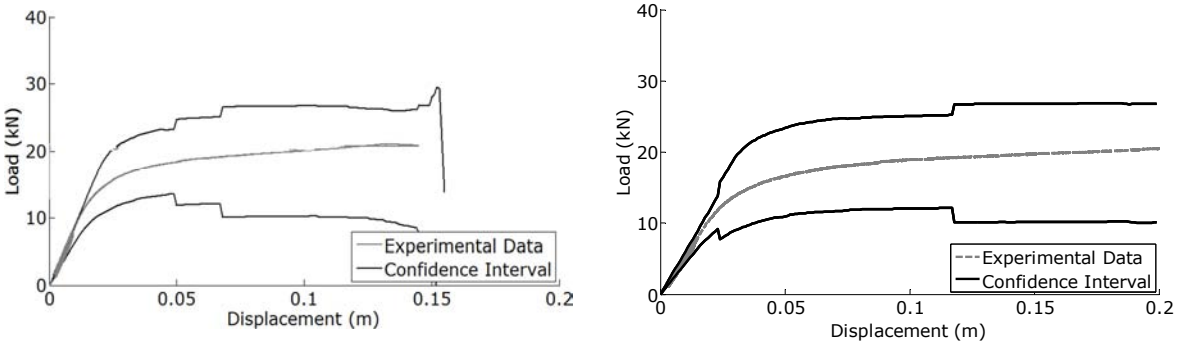


Figure 6.26. Model identification (service (beam 2) with Bayesian inference: a) quarter span; b) middle span.



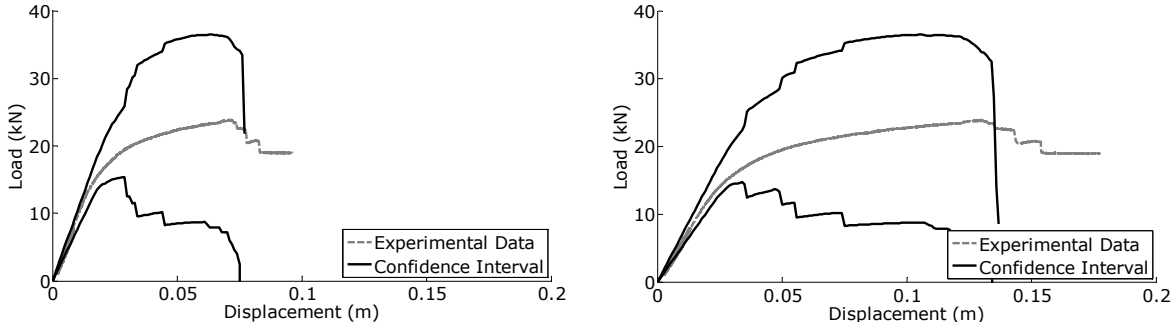
a) b)

Figure 6.27. Model identification (failure) (beam 1) without Bayesian inference: a) quarter span; b) middle span.



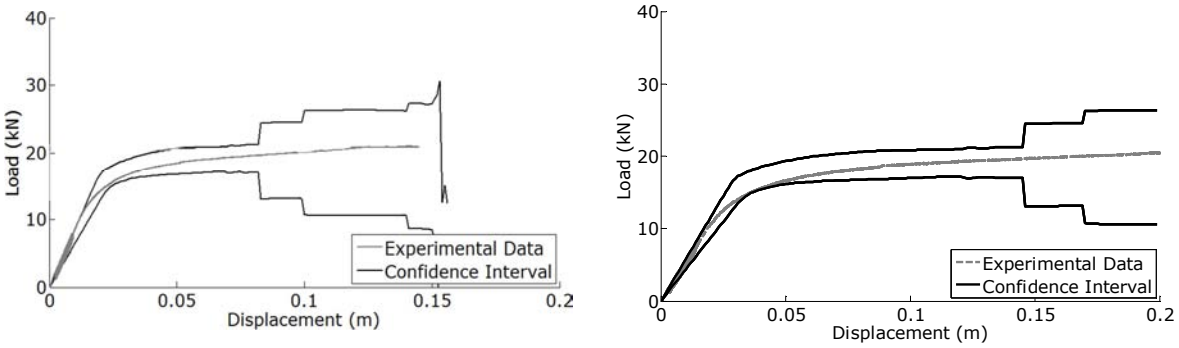
a) b)

Figure 6.28. Model identification (failure) (beam 2) without Bayesian inference: a) quarter span; b) middle span.



a) b)

Figure 6.29. Model identification (failure) (beam 1) with Bayesian inference: a) quarter span; b) middle span.



a) b)

Figure 6.30. Model identification (failure) (beam 2) with Bayesian inference: a) quarter span; b) middle span.

In this case, a set of failure load (F_R) values is obtained. A Normal PDF, which represents the structural resistance, is then adjusted to this set. Obtained resistance PDF parameter values for beam 1 are indicated in Table 6.26. On same table, the index-*p* values are also presented.

Table 6.26. Failure load (F_R) (beam 1).

Numerical model	μ [kN]	σ [kN]	Index-*p* [%]
Nominal values	19.00	2.21	89.60
Nominal values + Bayesian inference	22.76	2.50	96.63
Model identification (service)	20.47	1.76	92.04
Model identification (service) + Bayesian inference	23.87	2.28	97.84
Model identification (failure)	21.89	2.56	93.76
Model identification (failure) + Bayesian inference	24.42	2.49	99.60

From the analysis of obtained results, it is possible to conclude that: (1) The mean value increases as model identification procedures are applied; (2) The Bayesian inference approach increases the mean and standard deviation values of obtained resistance PDF, with exception of model identification until failure load for which the standard deviation value is decreased; (3) The index-*p* value is increased with the application of model identification techniques; (4) The index-*p* value also increases with the application of Bayesian inference procedure. It is possible to state that index-*p* presents very good results, being, several times, higher than 90%, which indicates accurate resistance PDF. The resistance PDF

obtained from model identification until failure load with Bayesian inference presents the highest index-*p* values. Results from nominal data values are the poorer ones.

Table 6.27 indicates the obtained resistance PDF parameter values for beam 2. On same table, the index-*p* values are also presented. From the analysis of obtained results, it is possible to conclude that: (1) The mean value is increased with the application of model identification; (2) The Bayesian inference approach increases the mean value, and decreases the standard deviation; (3) The index-*p* value is increased with the application of model identification; (4) Also, this parameter value is increased with the application of Bayesian inference procedure. It is possible to say that index-*p* presents very good results, being, several times, higher than 90%. This indicates that obtained resistance PDF are accurate. The resistance PDF obtained from model identification until failure load with Bayesian inference presents the highest index-*p* values. Results from nominal data values are the poorer ones.

Table 6.27. Failure load (F_R) (beam 2).

Numerical model	μ [kN]	σ [kN]	Index-*p* [%]
Nominal values	16.49	2.02	88.54
Nominal values + Bayesian inference	19.14	1.58	95.00
Model identification (service)	18.16	1.96	91.75
Model identification (service) + Bayesian inference	20.86	1.55	96.80
Model identification (failure)	19.35	2.54	95.49
Model identification (failure) + Bayesian inference	19.56	0.92	98.56

Figure 6.31 represents the resistance PDF (F_R), for beam 1, whose parameter values (mean and standard deviation) are presented in Table 6.28. An important concern is that the use of a Bayesian procedure increases the resistance PDF mean and standard deviation value for nominal values and those from model identification until service. For model identification until failure, only an increase on mean value is verified. The same way, model identification revealed an increase on mean value.

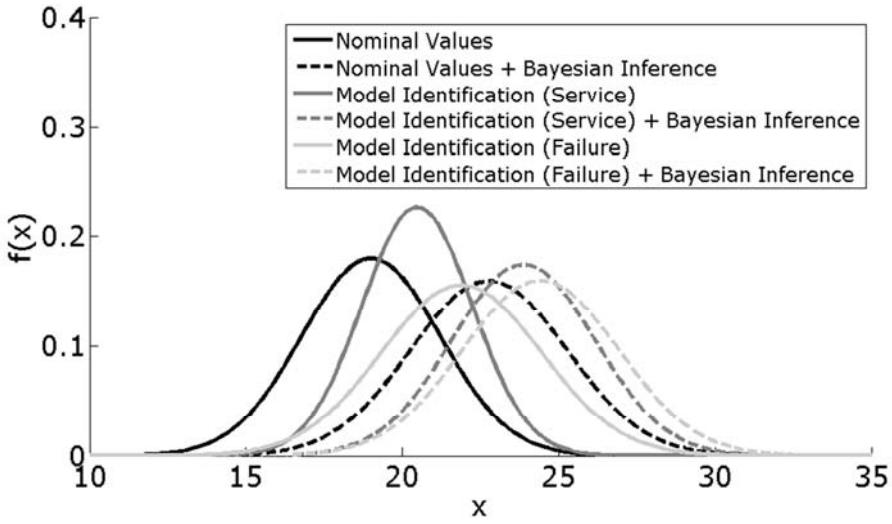


Figure 6.31. Failure load (F_R) (beam 1).

Figure 6.32 indicates resistance PDF (F_R), for beam 2, whose parameter values (mean and standard deviation) are presented in Table 6.28. An important concern is that the use of a Bayesian procedure increases the resistance PDF mean and diminishes the standard deviation value for both nominal values and those from model identification. The same way, model identification revealed an increase on mean value.

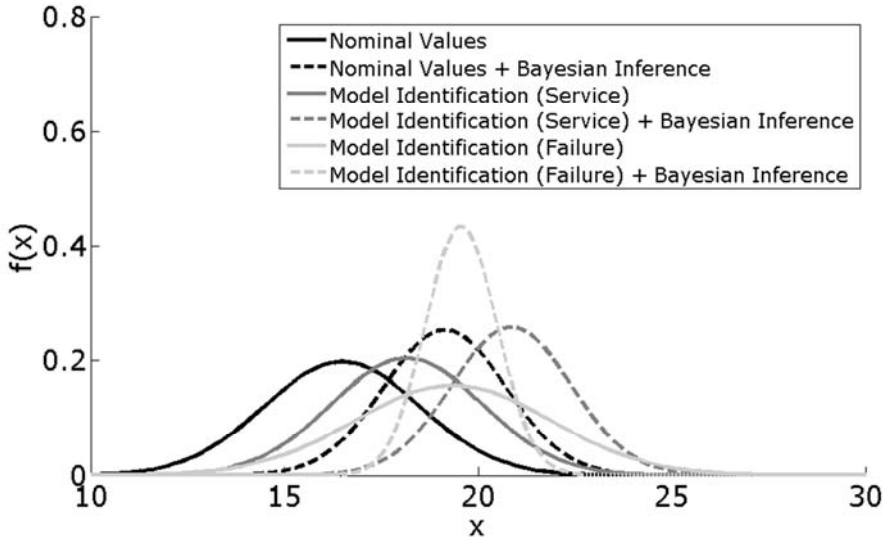


Figure 6.32. Failure load (F_R) (beam 2).

6.7. Safety assessment

Obtained resistance model is then used in a simple example of safety assessment [121]. This example consists in assessing the studied beam safety in a residential building, Figure 6.33. In this case, the resistance model is given by the failure load model (F_R), whose parameters are provided at Table 6.26 and 6.27, respectively, for beam 1 and 2. A model is

obtained for each analysis, respectively, considering the nominal values and those from model identification in service phase and until failure load.

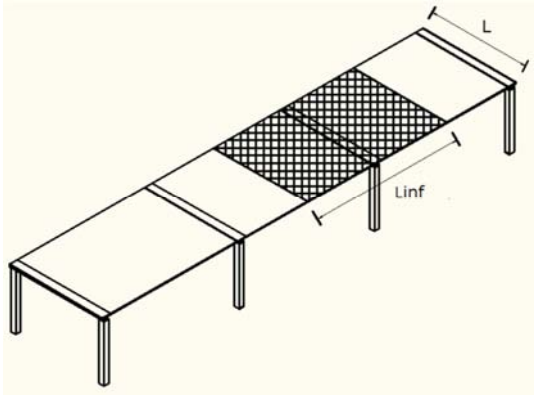


Figure 6.33. Residential building.

In order to compare resistance and loading curves it is necessary to transform the obtained resistance model into a model for maximum bending moment at middle span (M_R), through equation (6.24),

$$M_R = (F_R \cdot 2)/2 \cdot L/2 \tag{6.24}$$

This curve depends from the beam span (L) which is, in this situation, of 4.50 m. A Normal PDF is obtained for resistance.

Then, it is necessary to determine the loading curve. In order to do that it is important to determine the slab span. Such span will define the influence length of analysed beam (L_{inf}) which is, in this example, of 4.00 m. In this situation, loading is divided in self-weight and live load. All the other permanent loads were neglected. The self-weigh (w) is then computed through equation (6.25),

$$w = \gamma_{conc} \cdot A_{conc} + \gamma_{steel} \cdot A_{steel} \tag{6.25}$$

being γ_{conc} and γ_{steel} the concrete and laminated steel profile specific weight, and A_{conc} and A_{steel} the concrete and laminated profile area. These parameters present a Normal PDF, according to JCSS [93]. Therefore, self-weight presents a Normal PDF.

Live loads are divided into sustained, or long-term (q_{lt}), and intermittent, or short term (q_{st}), according to JCSS [93]. These PDF are respectively defined by a Gamma and an exponential PDF [93]. Table 6.28 provides the mean and standard deviation of each PDF. The applied load (p) is the sum of self-weight with both long and short-term live load component multiplied by the influence length of the beam, through equation (6.26),

$$p = w + q_{lt} \cdot L_{\inf} + q_{st} \cdot L_{\inf} \tag{6.26}$$

Table 6.28. Probabilistic models.

Parameter		PDF	μ	σ
γ_{conc}	[kN/m^3]	Normal	17.17	0.69
γ_{steel}	[kN/m^3]	Normal	77.00	0.77
q_{lt}	[kN/m^2]	Gamma	0.30	0.45
q_{st}	[kN/m^2]	Exponential	0.30	0.57

The further step consists in computing the maximum bending moment (M_S), through equation (6.27),

$$M_S = \left(p \cdot L^2\right) \Big/ 8 \tag{6.27}$$

Obtained values are then adjusted to a Lognormal PDF. In order to compare resistance and loading curves a limit state function (Z) is defined, through equation (6.28),

$$Z = M_R - M_S \tag{6.28}$$

This limit state is exceeded when loading is higher than resistance. The further steps consist in generating several values for each curve, according to each PDF parameters, and to register the number of values in which this limit state is exceeded in relation to the total number of evaluated points. The failure probability (P_f) is determined through equation (6.29),

$$P_f = P(Z \leq 0) \tag{6.29}$$

The reliability index (β) is then obtained, considering this value. A detailed description of how this index is computed is given in chapter four. On Table 6.29 and 6.30 are represented both failure probabilities and reliability indexes for all models, considering beam 1 and 2.

In this example, the building is of class 2 (apartment building – risk to life, given a failure, is medium or economic consequences are considerable) and of class B (normal cost of safety measure), according to JCSS [93]. Therefore, a target reliability index (β_{target}) of 3.30 is recommended.

The following conclusions are then obtained for beam 1: (1) when considering the nominal values or those from model identification in service phase the beam is classified as unsafe; (2) when the values from model identification until failure are taken into account, the beam is considered to be safe; (3) obtained results for all models, considering a Bayesian inference

approach, indicate that the beam is classified as safe. This means that the probabilistic assessment revealed a capacity reserve which was not accounted in design.

Table 6.29. Safety assessment (beam 1).

Numerical model	p_f	β
Nominal values	$1.05 * 10^{-3}$	3.07
Nominal values + Bayesian inference	$2.90 * 10^{-4}$	3.44
Model identification (service)	$5.44 * 10^{-4}$	3.27
Model identification (service) + Bayesian inference	$1.82 * 10^{-4}$	3.57
Model identification (failure)	$4.06 * 10^{-4}$	3.34
Model identification (failure) + Bayesian inference	$1.57 * 10^{-4}$	3.60

Table 6.30. Safety assessment (beam 2).

Numerical model	p_f	β
Nominal values	$2.53 * 10^{-3}$	2.81
Nominal values + Bayesian inference	$8.32 * 10^{-4}$	3.15
Model identification (service)	$1.31 * 10^{-3}$	3.00
Model identification (service) + Bayesian inference	$4.51 * 10^{-4}$	3.31
Model identification (failure)	$1.04 * 10^{-3}$	3.08
Model identification (failure) + Bayesian inference	$6.46 * 10^{-4}$	3.22

In respect to beam 2 the following conclusions are pointed out: (1) all the evaluated models are considered to be unsafe; (2) the β-value increases with model identification; (3) this value also increases with Bayesian inference; (4) the highest β-value is obtained when considering the values from model identification in service phase with Bayesian inference. Both beams present the same tendency. Two differences are respectively identified: (1) beam 2 presents lower β-values; (2) the highest β-value is obtained for different models.

6.8. Conclusions

This chapter provides the probabilistic assessment of two composite beams which were loaded at laboratory up to failure. The first beam presents a total connection while the other is partially connected. All other properties are maintained. Consequently, while the former presents a typical failure mode of bending with concrete crushing and yielding of steel profile, the latter presents a combined failure mode of bending and connection.

In this analysis, a nonlinear numerical model was developed and then simplified, without changing too much its accuracy. A sensitivity analysis is further executed to identify the most important properties, which can be due to material, geometric and physic source. Some of these parameters were characterized with detail at laboratory.

The developed numerical model is then adjusted to experimental data, through a model identification procedure. To perform that an optimization technique, based in the evolutionary strategies algorithm in its plus version, was used. Both modelling and measurement errors were considered in the algorithm stopping criteria. This procedure was developed for both service and failure region.

Further, a nonlinear probabilistic analysis was executed. In order to do so it was defined a PDF for each critical parameter. Such parameters are then updated with complementary data from laboratory characterization tests, by using a Bayesian inference approach. In some cases, the statistical uncertainty was reduced with this process. A probabilistic analysis, based in a LHS procedure, was further executed.

From the probabilistic analysis it was obtained an updated resistance PDF for applied load, for both analysed beams. It was then possible to probabilistically evaluate the experimental behavior of each tested beam by comparing those curves with obtained experimental data. These resistance PDF models are then used in a safety assessment example.

Some conclusions were obtained from probabilistic assessment: (1) model identification until failure load gives very good results (errors less than 10% for failure load); (2) model identification in service phase gives good results only for service region. Obtained results for failure region are bad. Complementary tests are thus recommended in this situation; (3) the most accurate models from a probabilistic analysis are those with values from model identification until failure load. Therefore the application of model identification before any probabilistic analysis is recommended; (4) Bayesian inference also increases the accuracy of probabilistic models. Therefore it is recommended the use of this procedure when complementary data is available.

7 Sousa River Bridge

7.1. Introduction

The developed probabilistic assessment framework is now tested with a case study that consists on a composite bridge submitted to a performance load test. Accordingly, within this chapter some aspects of Sousa River Bridge, the chosen structure, are described. The developed load test is also presented with reference to some important issues concerning applied loads, introduced load cases, monitoring schemes, etc.

A numerical model is developed in order to evaluate the structural behavior, by comparing obtained results with those from load test. This model is, in a first phase, calibrated with measured data. The main concerns regarding the numerical model development, such as used mesh, geometry definition, etc., are provided. A sensitivity analysis is then developed, both in service and failure region, to identify the most important parameters.

Once the numerical model is calibrated, the next step consists in applying the probabilistic assessment algorithm. This methodology, detailed described in chapter four, is divided in two modules, the model identification and the probabilistic analysis. During the identification procedure the numerical parameters are automatically adjusted in order that obtained results best fit the experimental data. This updating procedure is performed until a certain limit, given by the tolerance criterion, is attained. This value results from a weighted combination of measurement and modeling errors.

Once updated, a probabilistic numerical model is obtained by introducing randomness in input parameters, from material, geometry and physic sources. The statistical analysis of obtained results permits to determine the resistance probabilistic density function (PDF), necessary for safety assessment purposes. A description of used loading PDF is also provided.

Moreover, complementary tests are developed to increase the reliability of some input parameters. A description of these tests and obtained results are also given. An updated resistance PDF is computed through the use of a Bayesian inference procedure and considering obtained data from these tests. Obtained results from this case study will also enhance the advantages of using the probabilistic assessment framework.

7.2. Load test

7.2.1. Description

The Sousa river bridge was designed by Lisconcebe, S.A. [104, 105, 106] and constructed by Teixeira Duarte, S.A. [183]. This bridge belongs to the highway A43, Gondomar to Aguiar de Sousa (IC24), which is operated by BRISA, S.A. (Figure 7.1).

Figure 7.1. Sousa river bridge, overview [99].

This bridge presents a total length, between abutments, of 202.00 m – from 6+722.50 km to 6+924.50 km - corresponding to four spans of 44.00 m and an extreme span, near abutment A2, of 26.00 m, as indicated in Figure 7.2.

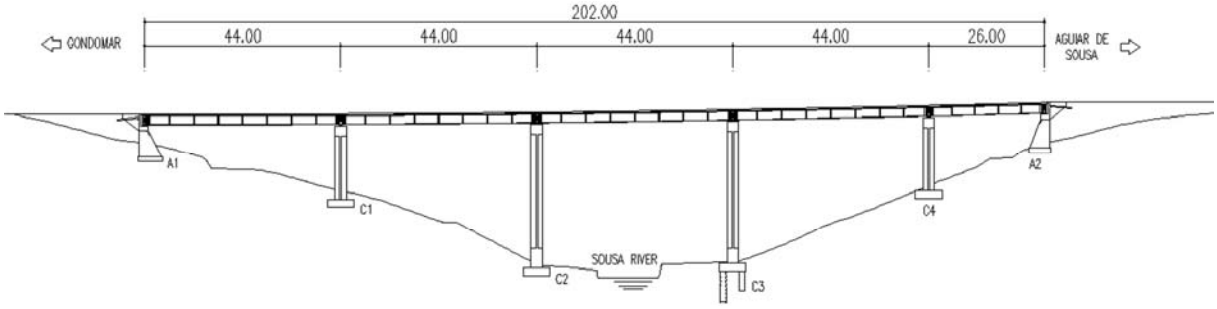


Figure 7.2. Sousa river bridge, side view (m) [105].

This bridge consists in two adjacent and independent structures, with identical typology, spaced of 3.00 m, each with a traffic flow, according to Figure 7.3. This figure also shows that this bridge is located in a straight zone of the highway.

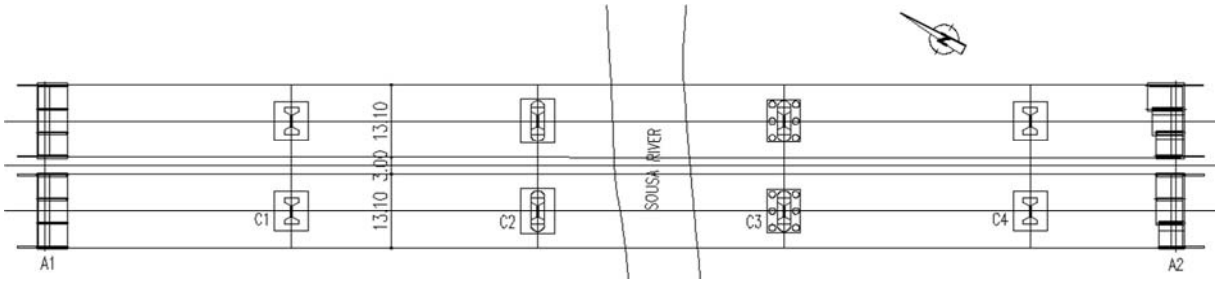


Figure 7.3. Sousa river bridge, horizontal plan (m) [105].

The continuous deck is composed by a precast reinforced concrete slab supported in two longitudinal metallic I-beams, as represented in Figure 7.4. The carriageway presents a transversal dimension of 7.50 m, an interior shoulder of 1.00 m, an external shoulder of 3.00 m, sidewalks (including cornice) and safety guards of 1.60 m, which result in a total of 13.10 m.

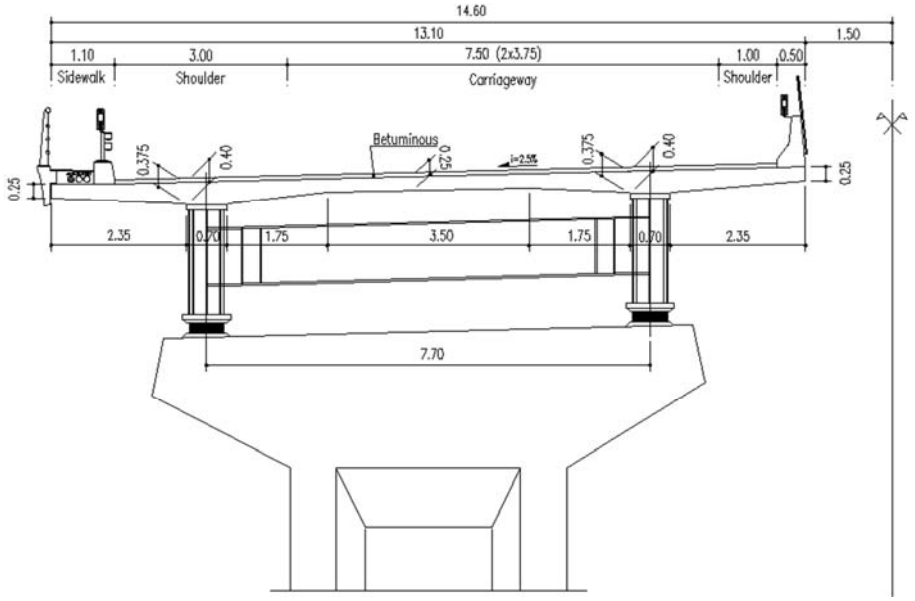


Figure 7.4. Sousa river bridge, transversal profile (m) [105].

The longitudinal girders present a constant height of 2.00 m, with exception to the extreme span in which it varies with deck inclination. These beams are made of welded steel profiles, constituted by individual plates. Top and bottom flange widths are constant within the whole bridge with, respectively, 700 mm and 840 mm. The web and flange thickness varies longitudinally in order to accomplish the relevant design criteria [105]. Therefore, the top flange thickness varies between 20 mm and 100 mm, the bottom flange thickness varies

between 25 mm and 110 mm, and the web thickness varies between 14 mm and 20 mm. Figure 7.5 shows the fabricated metallic girders already assembled on site.

Figure 7.5. Metallic girders, overview (provided by Teixeira Duarte, S.A.).

Transversally, these girders are fixed by stringers, separated of 5.50 m in 44.00 m span, and of 5.20 m in 26.00 m extreme span (Figure 7.6 and 7.7). The stringers are formed by laminated steel profiles welded to half steel profiles of the same type (IPE600). These half profiles are thus welded to the longitudinal girder flanges. The reinforced concrete slab, the metallic girders and the stringers constitute a transversal and rigid framework.

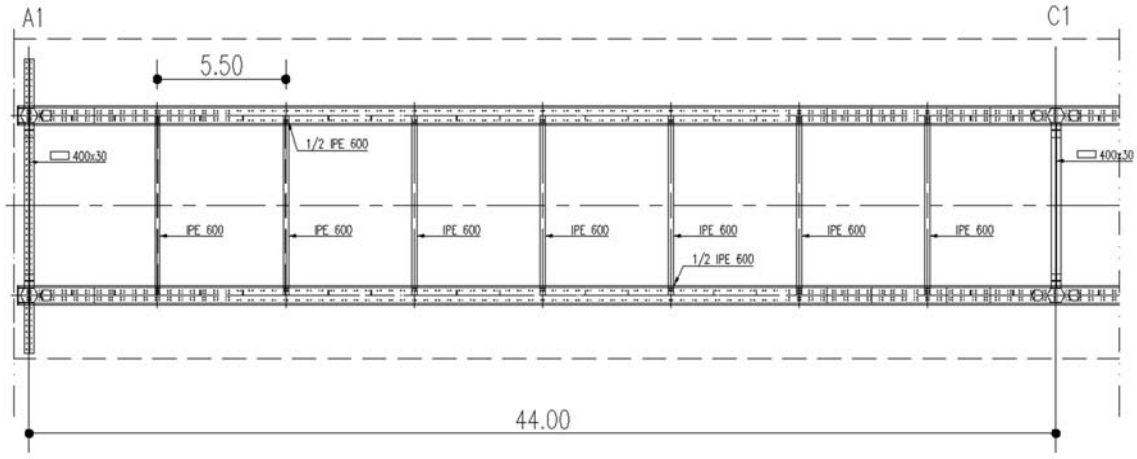


Figure 7.6. Metallic girders, horizontal plan (m) [105].

Additionally, and for web stability reasons, transversal reinforcements consisting in vertical plates, separated of 1/3 of the distance between stringers, are introduced. At support region, in both columns and abutments, the web reinforcements are interior and exterior, in order to resist to concentrated reactions. The stringers at support region are welded profiles from individual plates with different dispositions, whether it is an interior or an extreme support. These dispositions can be identified in both Figure 7.6 and 7.7.

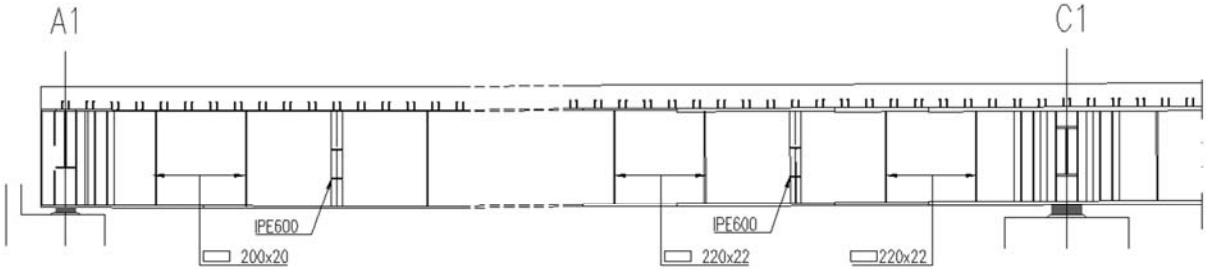


Figure 7.7. Metallic girders, side view (m) [105].

The precast reinforced concrete slabs are placed along the deck. Nelson headed studs [131], welded to the top flange of metallic girders, are used to establish the connection between these concrete slabs and the steel girders. There are three distinct zones of higher, medium and lower density of studs, as it can be verified in Figure 7.6. In order to guarantee waterproofing, neoprene joints are introduced in the slab / girder interface, eliminating thus the existent friction between these two elements.

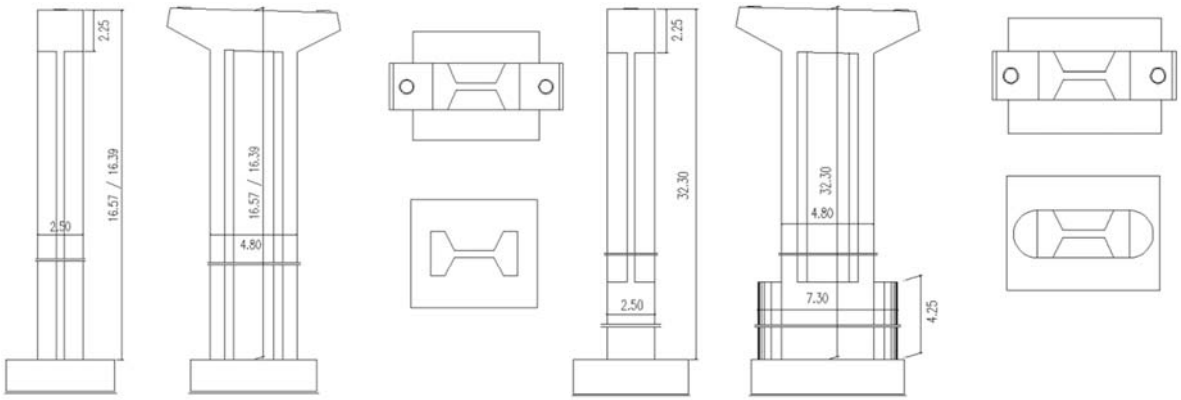
The deck is supported in columns and abutments through pot bearings (Figure 7.4). Longitudinally, it is fixed in all columns and moves over the abutments. Transversally, both columns and abutments give support to the deck. Expansion joints, that allow the bridge to move under temperature and other environmental effects, are introduced in the bridge extremities.

Columns are executed in reinforced concrete, presenting a maximum height of 35.00 m (average of 25.00 m). A constant I-section is adopted, with maximum dimensions 2.50 m x 4.80 m, being the highest value in transversal direction. Figure 7.8, indicates the dimensions of C1 and C4, and C2, while Figure 7.9 gives the dimensions of C3 (identified in Figure 7.2). A capital is introduced on the top of each column, in order to increase the section dimensions. The foundations are rigid, except in C3 for which, six short piles of 1.20 m diameter, connected to a pile cap block, are used.

The abutments are independent from the rest of the structure (Figure 7.2 and 7.3). They are constituted by a shear beam that is responsible of supporting the deck. This beam is continuously connected to parallel walls of constant thickness and variable width that increase with depth. A rigid foundation is used for the abutments. A transition slab is placed between the bridge and the rest of the highway. A view of the inferior side of the bridge deck (C4 and A2) is shown in Figure 7.10.

The pavement is betuminous with 20 mm thickness of regularization layer and 30 mm thickness of abrasion layer. It is also considered when in construction other equipment's

such as rain sewage, cable infrastructure, safety guards, etc. This equipment's are represented at Figure 7.4.



a) b)

Figure 7.8. Columns (m) [105]: a) C1 and C4; b) C2.

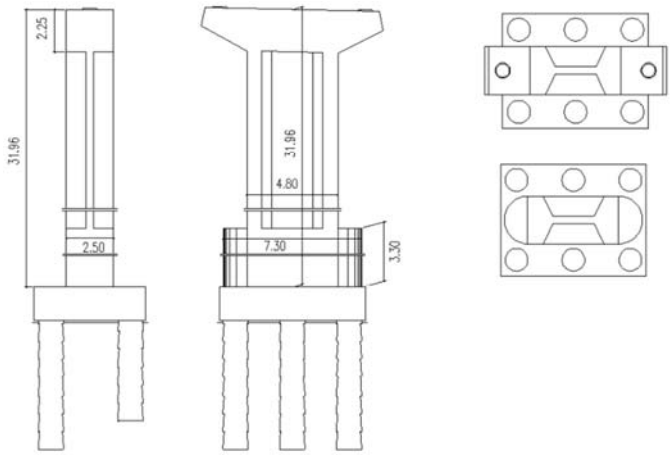


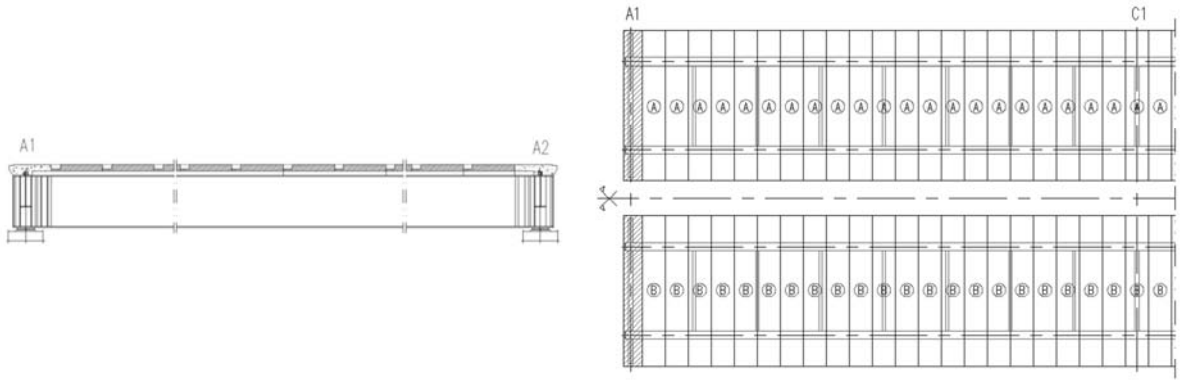
Figure 7.9. Column C3 (m) [105].

Figure 7.10. Sousa River Bridge, inferior side view (provided by Teixeira Duarte, S.A.).

The foundations, abutments and columns are erected following traditional procedures. The steel deck is manufactured by Socometal, S.A. [171] and incrementally launched through the

abutment A1, as it can be seen in Figure 7.5. Afterwards, the precast concrete slabs, developed by Mota Engil, S.A. [183], are placed over the metallic girders, according to Figure 7.11. These panels are connected through reinforcement bars and cast in-situ concrete.

A negative is left during construction between precast slabs, according to Figure 7.12. The precast slabs will be placed on the steel beams and the position of the negatives corresponds to the position of the studs in the steel beam. After placing, concrete is cast on the negatives, in order to establish the connection between concrete panels and the steel beams.



a) b)

Figure 7.11. Precast concrete slabs [105]: a) side view; b) horizontal plan.

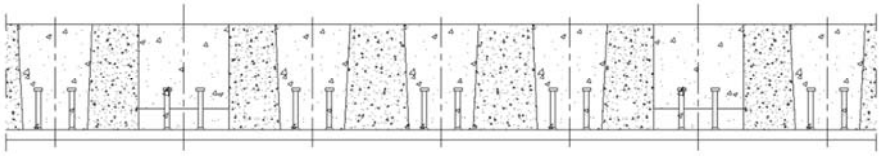


Figure 7.12. Connection between precast and cast in-situ concrete [105].

Concrete of class C30/37 [48] was used for bridge foundations, abutments and columns, and concrete of class C40/50 [48] was used for precast slabs and cast in-situ concrete. A 500 NR SD [48] steel is adopted for the reinforcement bars, which are perfectly embedded in the precast concrete slab. S355 [49] steel is considered for metallic girder. Used headed studs are produced in S235 J2 G3 + C450 [131] steel.

7.2.2. Obtained results

In order to evaluate the bridge behavior before its exploitation phase, a load test was performed by LABEST [99]. The vertical displacement and the temperature were measured through an automatic data acquisition system, during the test. The temperature was

measured at both inferior and superior side of the deck. All transducers are electric based and were tested and calibrated in laboratory, before load test.

The vertical displacement measurements were made with reference to the ground level. Linear variable differential transformers (LVDTs) were used for that. Such transducers present a precision of 0.05 mm (maximum value) for a measurement field of ± 25 mm (full scale), corresponding to a linearity of 0.10%. Six transducers were installed, two in the upward and four in the downward bridge. In this study, only this latter structure is considered. Accordingly, two transducers were installed at span A1 - C1, designated by VD1 and VD2, and other two at span C1 - C2, indicated as VD3 and VD4. Figure 7.13 shows a horizontal plan of used instrumentation.

A data acquisition system was used to register the signal in a continuous way from LVDTs and from environmental temperature sensors. A frequency of 10 Hz was designed for registering the vertical displacement data. The observation post is implemented closer to section A1 (Figure 7.13). This post is directly connected to a notebook in order to minimize any human error. A cable was used for each displacement transducer. Therefore two transducers, present a cable length of 17 m (VD1 and VD2), while the other two, present 66 m of cable length (VD3 and VD4. A description of installed LVDTs is presented at Table 7.1.

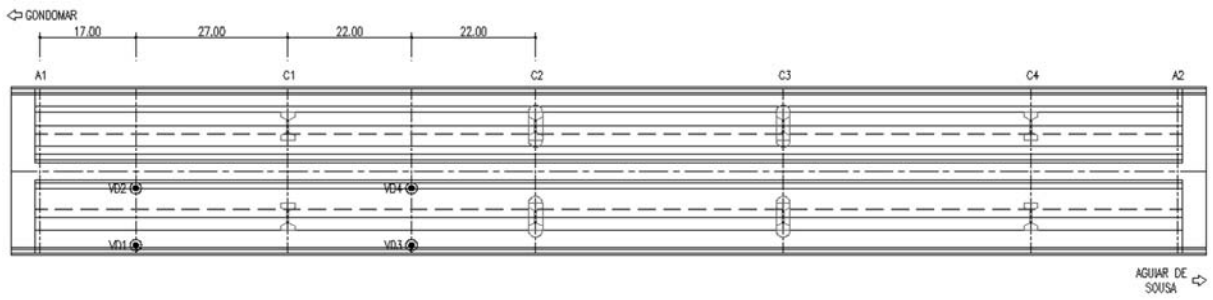


Figure 7.13. Instrumentation, horizontal plan [99].

Table 7.1. Installed LVDTs [99].

Bridge span A1-C1	Position	[m]	17	Downward bridge	Electric LVDT	VD1
						VD2
Bridge span C1-C2			66			VD3
						VD4

The load test was developed with four typologically identical vehicles (four axles), as represented at Figure 7.14. These vehicles are loaded with sand in order to obtain a total weight that is close to 32 ton.

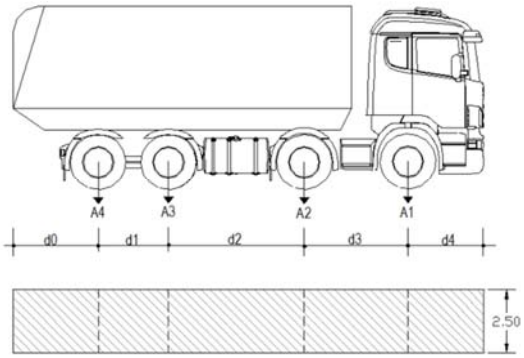


Figure 7.14. Four axles vehicle used in the load test [99].

Accordingly, while the two front axles support 40% (20% each) of the load, the two rear axles support 60% (30% each) of the total weight. A detailed description of each vehicle weight and distance between axles is indicated at Table 7.2.

Table 7.2. Identification of used vehicles [99].

		Vehicles			
		A	B	C	D
d_1	[m]	1.35	1.40	1.40	1.35
d_2	[m]	2.25	2.20	2.30	2.50
d_3	[m]	2.10	2.10	2.00	1.85
Total weight	[ton]	32.82	32.36	32.68	32.92

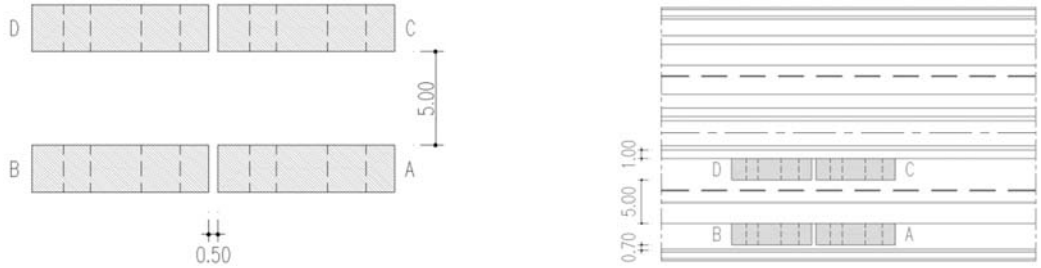
In this situation, three different load cases (LC1 - LC3) will be studied. In each situation the vehicles are immobilized in the bridge deck a time period of around 5 minutes. This period, between load cases, should be large enough to eliminate any vibration but, at same time, not too high, in order to minimize environmental effects. The exact position where the vehicles will stop is marked in the pavement. These load cases were defined in order to obtain the maximum value of each measured parameter. Table 7.3 indicates the main results to be expected from each load case.

In this situation, the considered load cases correspond to four vehicles in order to take the advantage of transversal symmetry. Figure 7.15a indicates a horizontal plan of how these vehicles are separated from each other, and Figure 7.15b shows these four vehicles positioned in the bridge deck.

Table 7.3. Main results to be expected [99].

Structure	Load case	Number of vehicles	Objective
Downward bridge	LC1	4	Maximum displacement at A1 - C1
	LC2	4	Maximum displacement at C1 - C2 and rotation at C1
	LC3	4	Maximum rotation at C2

The three load cases are schematically represented in Figure 7.16 to 7.18. In LC1 (Figure 7.16) the front axles of the two front vehicles are positioned at 26.00 m distance from section A1, at span A1 - C1.



a) b)

Figure 7.15. Distance gap (m): a) between vehicles; b) in the bridge deck [99].

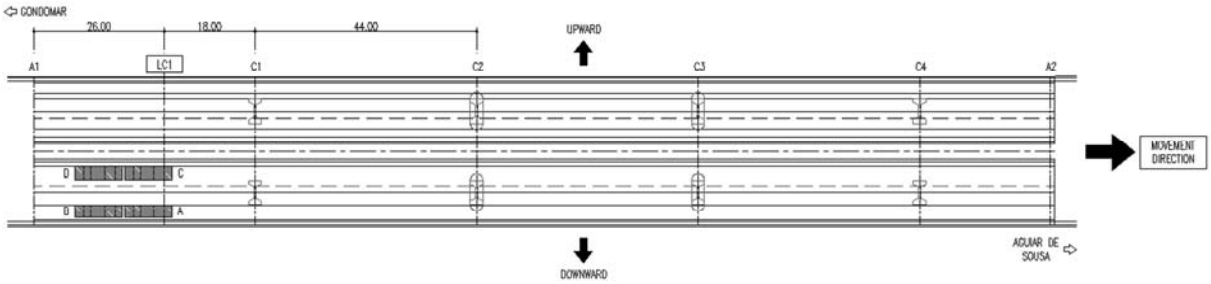


Figure 7.16. Load case 1 [99].

In LC2 (Figure 7.17) the front axles of the two front vehicles are positioned at 31.00 m distance from section C1, at span C1 - C2.

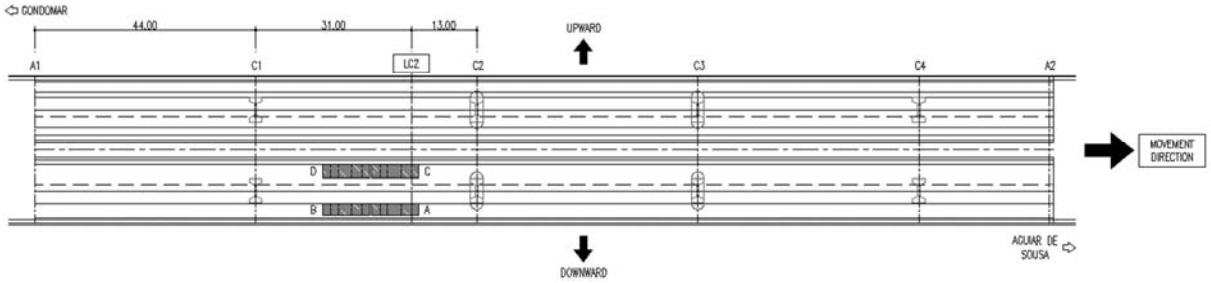


Figure 7.17. Load case 2 [99].

In LC3 (Figure 7.18) the front axles of the two front vehicles are positioned at 31.00 m distance from section C2, at span C2 – C3.

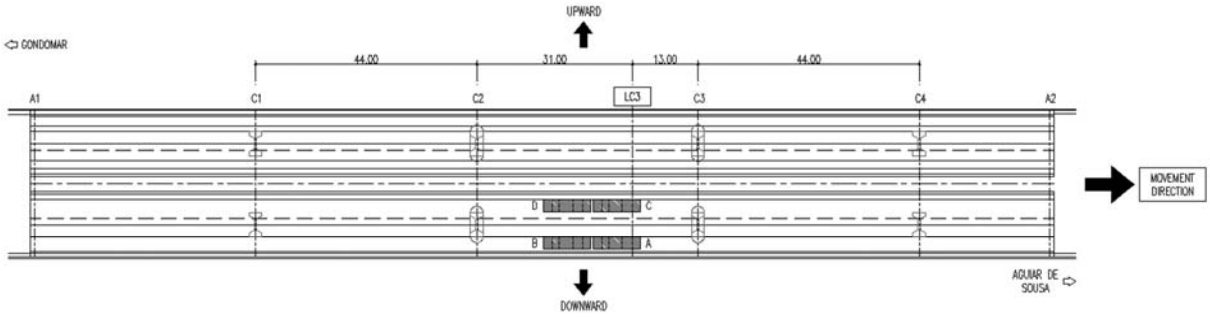


Figure 7.18. Load case 3 [99].

The three load cases were developed in a continuous basis on the way Gondomar to Aguiar de Sousa. Some images of the developed test are shown in Figure 7.19. The zero reference value was measured when no vehicle was applied to the bridge. This measurement is developed at the beginning of the test, in the defined interval between each load case and in the final phase of the test.

The data acquisition for unload situation permits to observe and control the temperature effect on the structure and in the monitoring system along the test. It was verified that the temperature was kept almost constant. This information permit to conclude that the temperature effect in registered values may be neglected [99].

a) b)

Figure 7.19. Load test: a) transversal profile; b) overview [99].

The obtained values for the four displacement transducers and for the three load cases are indicated in Table 7.4. The maximum vertical displacement is registered for the extreme span A1 - C1. As the span dimension and the applied load are identical for all load cases, this fact might be due to an inferior restriction of this span rotation.

Additionally, it is verified a variation between pairs of transducers positioned at same section of the span (lower than 5%). These differences might be due to load disposition along the carriageway, deck geometry, or variations in metallic girder connections. Nevertheless an

identical structural response is observed for all transducers. The results from the load test report indicate that the structural behavior is admissible and that the constructed bridge is in agreement with the design project [99].

Table 7.4. Registered vertical displacements [99].

Load Case	VD1	VD2	VD3	VD4
	[mm]	[mm]	[mm]	[mm]
LC1	16.01	14.48	-4.11 *	-3.51 *
LC2	-4.10 *	-3.84 *	14.00	13.40
LC3	1.86	1.84	-3.47 *	-2.92 *

* Negative value corresponds to a displacement in upward direction.

7.3. Numerical analysis

In order to evaluate the bridge behavior, a numerical model is developed with ATENA® [23, 24] nonlinear structural analysis software. This model is executed according to the design project of Lisconcebe, S.A. [104, 105, 106], and calibrated with obtained results from load test. A comparison is thus established with experimental data and each simplification is evaluated from a modeling error perspective.

7.3.1. Numerical model

Before developing the nonlinear structural model, a test with an elastic structural analysis software is performed in order to validate some simplifications. In this situation, and for the three considered load cases, a comparison is made with the structural behavior of a bridge with five spans of 44.00 m, which corresponds to a total length of 220.00 m, and with a constant metallic girder height of 2.00 m. Obtained results provide an error that is lower than 1%, which validates this simplification.

A 2D plane stress model is then developed in ATENA® [23, 24]. This model intends to represent the tested bridge. The bridge transversal section, constituted by two metallic girders and a precast reinforced concrete slab, is symmetric (Figure 7.4). In this situation, the three load cases are also symmetric, as can be checked from Figure 7.16 to 7.18. Therefore, when developing the structural model, it is possible to take advantage of symmetry. Accordingly, it is considered a width of 6.55 m for the precast slab (half of the full width, 13.10 m). An important consideration is the variation of the effective width along the bridge due to the shear lag effect [50]. Therefore, in the first and final 11.00 m of the bridge length, a deck width of 6.08 m is considered.

The precast slab presents a non-uniform geometry. In fact, its height varies from 0.25 m, at internal shoulder and slab middle span, to 0.40 m above the metallic girder (Figure 7.4). The width is constant, in the first 0.25 m, and then it progressively diminishes, in the subsequent 0.15 m, to 0.70 m. This non-uniformity is considered in the numerical model by introducing several layers. As the number of layers increase, the model gets close to reality, but it also gets more complex. Therefore, two rectangular layers are proposed and considered to be a good compromise between accuracy and efficiency. Each layer presents the same height (h_{slab}) as before, varying its width (b_{slab}) in a way that area and inertia remain unchanged. This approximation does not guarantee the same gravity center (Y_G). An error lower than 7% is computed, which is not considered relevant to the global bridge behavior. The obtained results are indicated in Table 7.5.

Table 7.5. Precast concrete slab equivalent section.

Length	b_{slab}	h_{slab}	Area	Inertia	Y_G *		
					Real	Equivalent	Error
[m]	[m]	[m]	[m^2]	[m^4]	[m]	[m]	[%]
0 – 11	2.76	0.15	1.99	$2.21 * 10^{-2}$	$2.49 * 10^{-1}$	$2.33 * 10^{-1}$	6.38
	6.30	0.25					
11 – 209	2.95	0.15	2.11	$2.35 * 10^{-2}$	$2.50 * 10^{-1}$	$2.33 * 10^{-1}$	6.91
	6.68	0.25					
209 – 220	2.76	0.15	1.99	$2.21 * 10^{-2}$	$2.49 * 10^{-1}$	$2.33 * 10^{-1}$	6.38
	6.30	0.25					

* This value is determined considering the interface section as reference.

The reinforcement bars were considered to be completely embedded in the precast concrete slab. Therefore, the number of bars was counted and the corresponding area was divided in layers, in the numerical model. Two layers were respectively considered at the upper side of the concrete slab and three layers were defined in the lower side. Obtained reinforcing steel area ($A_{s,l}$) per layer is given at Table 7.6.

The position of each layer in the numerical model is also given at Table 7.6. In order to obtain such position it becomes necessary to compute the superior concrete cover ($c_{sup} = 6.33 * 10^{-2}$ m). It is assumed the same cover for superior and inferior layer of the upper side of concrete slab. For lower side of concrete slab, the total area of each layer is distributed in its gravity center.

Table 7.6. Precast concrete slab reinforcement.

Layer	Length	$A_{s,l}$	Position
	[m]	[cm^2]	[m]
1	0.00 – 11.00; 209.00 – 220.00	97.77 (34ϕ16+26ϕ12)	2.34
	11.00 – 209.00	104.10 (36ϕ16+28ϕ12)	2.34
2	0.00 – 11.00; 209.00 – 220.00	49.13 (16ϕ16+15ϕ12)	2.21
	11.00 – 209.00	55.42 (18ϕ16+17ϕ12)	2.21
3	0.00 – 220.00	25.14 (8ϕ16+8ϕ12)	2.12
4	0.00 – 220.00	12.57 (4ϕ16+4ϕ12)	2.08
5	0.00 – 220.00	12.06 (6ϕ16)	2.02

The metallic I-beam is constituted by welded plates. It presents a constant height of 2.00 m, along the bridge, except for the extreme span in which it presents a slightly variation. However, and according to previous simplification, this value is considered to be constant in the whole bridge. The bottom and top flange presents, respectively, widths of 700 and 840 mm. The flanges and web thickness varies along the bridge. These values are given at Table 7.7.

All these sections were classified according to EN-1993-1-1 [49]. For negative bending moment they all present a web of class 2 or 3. For these classes, the full length of the web is considered to be effective. For positive bending moment, all the cross sections present a web of class 4. Thus it is necessary to evaluate the neutral axis position. This analysis provided the information that its location is always in the concrete slab, which means that the web is in tension and thus there is no risk of buckling. Accordingly, the web effective length may be considered as total within the whole bridge. In respect to flanges, all sections are of class 2 or 3.

Once a plane stress model is adopted, it becomes costly to compute all variations in flanges thickness, according to Table 7.7. This would imply a horizontal line along the bridge for each thickness. Therefore, the adopted mesh size should be very small in order to maintain an appropriate edge length ratio. Additionally, the number of macro elements would increase considerably. This would imply a high computational cost which is not appropriate for a probabilistic assessment algorithm.

Therefore, and in order to overcome this problem, it was used only a macro element, to simulate the girder web, with a total height of 2.00 m, and with a thickness variation (b_{web})

according to Table 7.7. Moreover, horizontal steel bars, embedded in this macro element, were introduced to simulate the girder flanges. The diameter of each bar is computed in a way that the real and equivalent flange area ($A_{fl,sup}$ and $A_{fl,inf}$) and section gravity center remain unchanged. In this situation, obtained maximum error in moment of inertia is of 10.52%. Table 7.8 present the corresponding results.

Table 7.7. Thickness variation along the bridge.

Section	Length	Thickness [mm]			Section	Length	Thickness [mm]		
		Top Flange	Bottom Flange	Web			Top Flange	Bottom Flange	Web
	[m]	[mm]	[mm]	[mm]		[m]	[mm]	[mm]	[mm]
1	0.00 – 33.90	30	40	16	1	121.90 – 123.90	30	40	16
2	33.90 – 35.90	40	60	16	6	123.90 – 125.90	45	55	16
3	35.90 – 37.90	65	75	18	7	125.90 – 128.95	65	75	16
4	37.90 – 40.95	75	85	18	8	128.95 – 135.05	85	90	18
5	40.95 – 47.05	100	110	20	7	135.05 – 138.10	65	75	16
4	47.05 – 50.10	75	85	18	6	138.10 – 140.10	45	55	16
3	50.10 – 52.10	65	75	18	1	140.10 – 165.90	30	40	16
2	52.10 – 54.10	40	60	16	2	165.90 – 167.90	40	60	16
1	54.10 – 79.90	30	40	16	3	167.90 – 169.90	65	75	18
6	79.90 – 81.90	45	55	16	4	169.90 – 172.95	75	85	18
7	81.90 – 84.95	65	75	16	5	172.95 – 179.05	100	110	20
8	84.95 – 91.05	85	90	18	4	179.05 – 182.10	75	85	18
7	91.05 – 94.10	65	75	16	3	182.10 – 184.10	65	75	18
6	94.10 – 96.10	45	55	16	2	184.10 – 186.10	40	60	16
1	96.10 – 98.10	30	40	16	1	186.10 – 220.00	30	40	16
9	98.10 – 121.90	25	30	14	-	-	-	-	-

The web reinforcement at both abutments and columns, necessary to concentrate reactions (Figure 7.6 and 7.7), is simulated by considering an equivalent thickness at each web macro element, in a length and height that corresponds to reality. This value is computed in a way that the reinforcement area is kept unchanged, being then added to the real web thickness.

Accordingly, at A1 and A2 the thickness value increases to 53.00 mm, at C1 and C4 to 48.20 mm and at C2 and C3 to 46.20 mm.

Moreover, a vertical steel bar, embedded in web macro element, is introduced in each support section to simulate an existent steel plate (Figure 7.6 and 7.7). The diameter of this bar is computed in a way that the total area remains unaltered. Therefore, a diameter of 114.30 mm and of 66.10 mm is obtained for the steel bar, respectively, at abutment and at column support section.

In order to avoid any local instability, the web is reinforced at bridge span with vertical stringers, respectively, a half IPE600 spaced of 5.50 m and a steel plate spaced of 1/3 of 5.50 m (Figure 7.6 and 7.7). In order to model these elements, a vertical steel bar, embedded at web macro element, is considered. The diameter of this bar is computed in a way that the total area remains unchanged. Therefore, a diameter of 99.65 mm and of 71.36 mm is respectively considered for the half IPE600 and for the steel plate.

Table 7.8. Metallic girder equivalent section.

Section	Equivalent diameter		Area		Y_G			Inertia		
	Top Flange	Bottom Flange	Top Flange	Bottom Flange	Top Flange	Bottom Flange	Total *	Real	Equivalent *	Error
	[mm]	[mm]	[m^2]	[m^2]	[m]	[m]	[m]	[m^4]	[m^4]	[%]
1	161.64	204.86	$2.10 * 10^{-2}$	$3.36 * 10^{-2}$	1.99	$2.00 * 10^{-2}$	$8.59 * 10^{-1}$	$6.05 * 10^{-2}$	$6.24 * 10^{-2}$	3.16
2	186.64	250.90	$2.80 * 10^{-2}$	$5.04 * 10^{-2}$	1.98	$3.00 * 10^{-2}$	$8.06 * 10^{-1}$	$7.94 * 10^{-2}$	$8.34 * 10^{-2}$	5.10
3	237.58	280.17	$4.55 * 10^{-2}$	$6.30 * 10^{-2}$	1.97	$3.75 * 10^{-2}$	$8.84 * 10^{-1}$	$1.09 * 10^{-1}$	$1.16 * 10^{-1}$	6.72
4	255.20	298.26	$5.25 * 10^{-2}$	$7.14 * 10^{-2}$	1.96	$4.25 * 10^{-2}$	$8.88 * 10^{-1}$	$1.22 * 10^{-1}$	$1.31 * 10^{-1}$	7.81
5	294.25	338.89	$7.00 * 10^{-2}$	$9.24 * 10^{-2}$	1.95	$5.50 * 10^{-2}$	$8.96 * 10^{-1}$	$1.53 * 10^{-1}$	$1.69 * 10^{-1}$	10.52
6	197.97	240.22	$3.15 * 10^{-2}$	$4.62 * 10^{-2}$	1.98	$2.75 * 10^{-2}$	$8.71 * 10^{-1}$	$8.11 * 10^{-2}$	$8.49 * 10^{-2}$	4.69
7	237.93	280.51	$4.55 * 10^{-2}$	$6.30 * 10^{-2}$	1.97	$3.75 * 10^{-2}$	$8.81 * 10^{-1}$	$1.08 * 10^{-1}$	$1.15 * 10^{-1}$	6.82
8	271.68	306.91	$5.95 * 10^{-2}$	$7.56 * 10^{-2}$	1.96	$4.50 * 10^{-2}$	$9.10 * 10^{-1}$	$1.31 * 10^{-1}$	$1.43 * 10^{-1}$	8.51
9	147.77	177.63	$1.75 * 10^{-2}$	$2.52 * 10^{-2}$	1.99	$1.50 * 10^{-2}$	$8.93 * 10^{-1}$	$4.93 * 10^{-2}$	$5.05 * 10^{-2}$	2.34

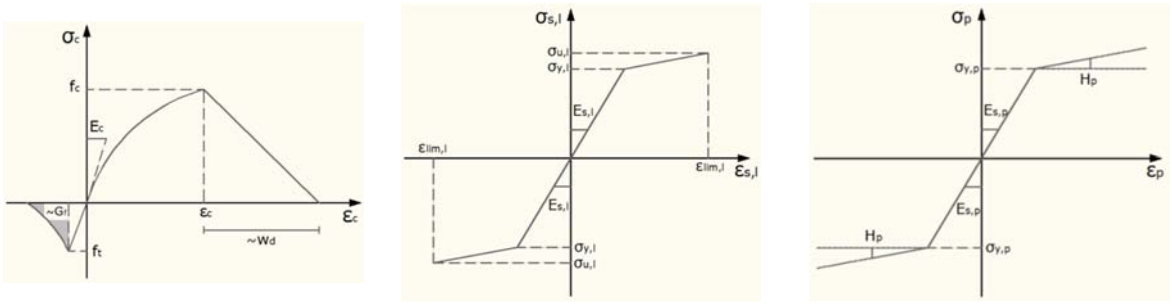
* This value is computed considering the girder web.

The structure is supported in two abutments and four columns (Figure 7.2). The vertical displacement is restricted in these positions. While at abutments the horizontal displacement is free, at columns it is supposed to be fixed. However, this restriction is not total due to the deformability of the column, the pot bearing and the foundation.

In order to study this effect, these columns are grouped in two sets according to their height and configuration (Figure 7.8 and 7.9), respectively, C1 and C4 (set 1) and C2 and C3 (set 2). A model is thus developed for each set, in an elastic structural analysis software. Each vertical bar is divided in two or three segments, according to each set configuration. The geometry (area and inertia) and the material properties (C30/37, E_c=33GPa) are the real values. These vertical bars are restricted at their bottom in all directions, considering that foundation is perfect. A unitary load is horizontally applied at their top, being then computed the correspondent horizontal displacement.

Therefore, in ATENA® [23, 24] numerical model, vertical restrictions are considered in all supports (including abutments). A horizontal spring is also considered at each column. The horizontal spring constant is the inverse of the computed horizontal displacement, with elastic structural analysis software. Accordingly, a value of k_1 = 56.69 kN/m and of k_2 = 9.93 kN/m is respectively obtained for the first and second set. In order to avoid stress concentration a steel plate, with a width equal to the metallic girder bottom flange (840 mm), is introduced at each support.

In respect to materials, C40/50 [48] was adopted on both precast slabs and cast in-situ concrete. Figure 7.20a presents the stress-strain law for concrete. Both compressive strain at compressive strength (ε_c) and critical compressive displacement (w_d) are considered to be deterministic within the analysis.



a) b) c)

Figure 7.20. Stress-strain law: a) concrete; b) reinforcing steel; c) steel profile.

Used reinforcement, embedded in precast concrete slab, is of class A 500 NR SD [48]. The stress-strain law for this material is shown at Figure 7.20b. The elasticity modulus ($E_{s,l}$) is considered to be deterministic within the analysis. S355 [49] material was considered for metallic girder. Figure 7.20c indicates the stress-strain law for this material. In this situation, for plates with thickness lower than 50 mm it is used a S355J0, between 50 and 75 mm a S355J2, between 75 and 90 mm a S355K2, and between 90 and 110 mm a S355ML. For laminated steel profiles, S355J0 is used [44]. An elasticity modulus of 210 GPa ($E_{s,p}$) was

considered for steel material. Such parameter is considered to be deterministic within the analysis.

Used materials, concrete and steel, were modeled by an SBETA and a bilinear with hardening Von Mises material model, which are constitutive models of the ATENA® library [23, 24]. A biaxial stress failure criterion and a Von Mises yield criterion are respectively established. The nominal values, considered in numerical model, were those indicated at EN 1992-1-1 [48], for concrete and steel reinforcement, and in EN 1993-1-1 [49] for steel profile. Table 7.9, 7.10 and 7.11 present those values.

Table 7.9. Material properties (precast slab and cast in-situ concrete).

Parameter		Mean value
Elasticity modulus (E_c)	[GPa]	35.00
Tensile strength (f_t)	[MPa]	3.50
Compressive strength (f_c)	[MPa]	48.00
Fracture energy (G_f) [23, 24]	[N/m]	87.50
Compressive strain at compressive strength (ε_c)	[‰]	2.00
Critical compressive displacement (w_d) [23, 24]	[m]	$5.00 * 10^{-4}$

Table 7.10. Material properties (reinforcing steel).

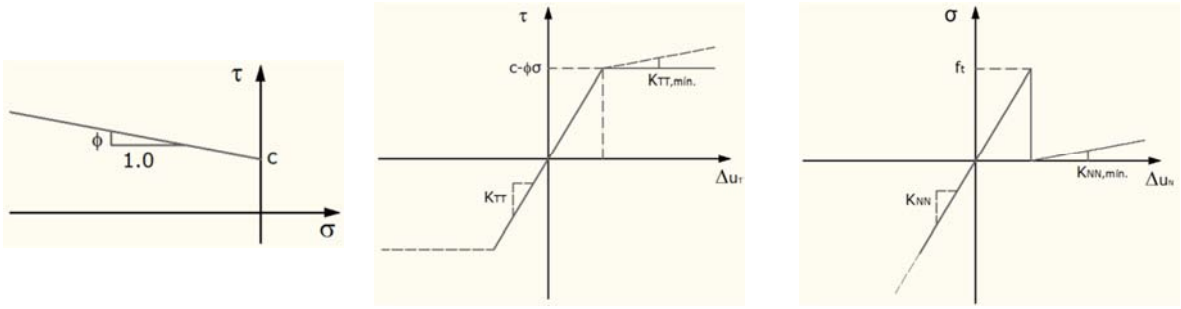
Parameter		Mean value
Elasticity modulus ($E_{s,l}$)	[GPa]	205.00
Yield strength ($\sigma_{y,l}$)	[MPa]	560.00
Limit strength ($\sigma_{u,l}$) [48]	[MPa]	644.00 (k = 1.15)
Limit strain ($\varepsilon_{lim,l}$) [48]	[‰]	80.00

The connection between precast concrete slab and metallic girder is guaranteed by headed studs, produced in S235 J2 G3 + C450 steel [131], with yield strength (σ_y) of 350 MPa, limit strength (σ_u) of 450 MPa and limit strain (ε_{lim}) of 18 %, and by cast in-situ concrete. An interface material model, based in a Mohr-Coulomb failure criterion with tension cut-off, is used to describe such behavior. This law is given in terms of shear (τ) and normal stresses (σ). According to Figure 7.21a, the initial failure corresponds to the moment when cohesion (c) value is reached. After stress violates this condition, the relation between these stresses is given by the dry friction (ϕ) which is considered to be very low.

Table 7.11. Laminated steel profile material [44].

No.	Thickness	Yield strength ($\sigma_{y,p}$)	Limit strength ($\sigma_{u,p}$)	Limit strain ($\varepsilon_{lim,p}$)	Hardening Modulus (H_p)
	[mm]	[MPa]	[MPa]	[%]	[MPa]
1	≤ 16	355	470-53	20-22	696.10
2	≤ 40	345	470-53	20-22	743.90
3	≤ 60	335	470-53	19-21	831.60
4	≤ 80	325	470-53	18-20	928.60
5	≤100	315	470-530	18-20	981.40
6	≤ 110	295	450-600	18-18	1287.80

For shear stresses, and for positive slip (Δu_T), this law is characterized by an initial shear stiffness (K_{TT}), until the Mohr-Coulomb criterion is reached, and then it presents a minimum shear stiffness ($K_{TT,min}$) that is 1% of the initial value (Figure 7.21b). For normal stresses, and for positive uplifts (Δu_N), it is defined by an initial normal stiffness (K_{NN}) until the tensile strength (f_t) is reached. Once attained, the normal stress is reduced to 0, being this law defined by a minimum normal stiffness ($K_{NN,min}$) that is 1% of K_{NN} (Figure 7.21c).



a) b) c)

Figure 7.21. Interface law: a) normal and shear stress; b) shear stress and slip; c) normal stress and uplift.

There are three distributions of headed studs along the bridge, respectively, low (6 studs), medium (8 studs) and high (10 studs) density (Figure 7.6). The space between each layer of studs, 0.50 m, is the same. The length of each region is defined according to the shear stress diagram [105]. The medium density is used to avoid an abrupt transition from low to high density region. As this region is very short, it will not be considered in this model. Accordingly, the definition of the length of each region is presented at Table 7.12.

In respect to normal behavior, the interface is defined by normal stiffness (K_{NN}) and tensile strength (f_t) parameters. Both these parameters are assumed to present high values in order to guarantee the composite action. These values are considered to be deterministic within the whole analysis.

Considering the shear behavior, the interface is characterized by shear stiffness (K_{TT}) and cohesion (c) parameters. The cohesion value depends from the stud maximum load capacity (P_{Rm}). This value is determined according to EN 1994-1-1 [50] by using equation (7.1) and (7.2),

$$P_{Rk} = 0.80 \cdot f_u \cdot \pi \cdot d^2/4 = 102.07\,kN \tag{7.1}$$

$$P_{Rk} = 0.29 \cdot \alpha \cdot d^2 \cdot \sqrt{f_{ck} \cdot E_{cm}} = 123.87\,kN \tag{7.2}$$

being d the stud shank diameter (19 mm), h_{sc} the stud height (150 mm), f_u the limit strength of stud material (450 MPa) [131], α a parameter that depends from the ratio h_{sc} / d = 7.89, E_{cm} the concrete elasticity modulus (Table 7.9) and f_{ck} the characteristic concrete compressive strength (40 MPa). The characteristic stud maximum load capacity (P_{Rk} = 102.07 kN) is the minimum value of (7.1) and (7.2). The mean value (P_{Rm} = 113.41 kN) is then obtained by dividing the previous value per 0.90 [50]. By considering this value it is then possible to determine the cohesion value, through equations (7.3), (7.4) and (7.5),

$$c = \frac{(stud\ \max imum\ load\ capacity \cdot number\ of\ studs)}{(width\ of\ the\ \text{int}erface \cdot length)} \tag{7.3}$$

$$c = \frac{(113.41 \cdot 10)}{(0.70 \cdot 0.50)} \cong 3.24\,MPa \tag{7.4}$$

$$c = \frac{(113.41 \cdot 6)}{(0.70 \cdot 0.50)} \cong 1.94\,MPa \tag{7.5}$$

A similar procedure is used to determine the shear stiffness. In order to obtain such parameter it is necessary to compute the stud stiffness value. However, this is extremely difficult to quantify as it depends from several factors. Accordingly, it was defined a value of 325 kN/mm, based in experience. By taking this value into consideration it is then possible to determine the shear stiffness through equations (7.6), (7.7) and (7.8),

$$K_{TT} = \frac{(stud\ stiffness \cdot number\ of\ studs)}{(width\ of\ the\ \text{int}erface \cdot length)} \tag{7.6}$$

$$K_{TT} = \frac{(325.00 \cdot 10)}{(0.70 \cdot 0.50)} \cong 9285.71\,kN/m^2/mm \tag{7.7}$$

$$K_{TT} = \frac{(325.00 \cdot 6)}{(0.70 \cdot 0.50)} \cong 5571.43 \; kN/m^2/mm \tag{7.8}$$

Table 7.13 summarizes the obtained values for interface properties, considering both situations of high and low density of headed studs.

Table 7.12. Interface density region.

Density	Length [m]	Density	Length [m]
High	0.00 – 8.50	High	120.75 – 123.50
Low	8.50 – 32.75	Low	123.50 – 126.25
High	32.75 – 44.00	High	126.25 – 132.00
Low	44.00 – 49.75	Low	132.00 – 137.25
High	49.75 – 52.50	High	137.25 – 140.00
Low	52.50 – 55.25	Low	140.00 – 142.75
High	55.25 – 77.25	High	142.75 – 164.75
Low	77.25 – 80.00	Low	164.75 – 167.50
High	80.00 – 82.75	High	167.50 – 170.25
Low	82.75 – 88.00	Low	170.25 – 176.00
High	88.00 – 93.75	High	176.00 – 187.25
Low	93.75 – 96.50	Low	187.25 – 211.50
High	96.50 – 99.25	High	211.50 – 220.00
Low	99.25 – 120.75	-	-

In respect to mesh, quadrilateral elements with 0.21 m dimension size are used. Additionally, interface elements are used to simulate the steel to concrete connection and spring elements are chosen to simulate the horizontal support conditions. In order to simulate the reinforcement bars, nonlinear truss elements, embedded at concrete slab, are considered. These elements are also introduced to simulate the web reinforcements and both top and bottom flanges of metallic girder. A frame of obtained mesh is shown at Figure 7.22.

The pavement was not considered as part of the structure but as a distributed load instead (p_{pav}). In fact, it presents 0.05 m thickness, which means that in order to obtain an appropriate edge-length ratio a small mesh size should be used, requiring a high computational cost. Accordingly, two distributed loads of 6.50 kN/m and of 6.81 kN/m were,

respectively, computed considering the betuminous specific-weight (γ_{pav} = 22.50 kN/m^3) and the pavement width for the initial and final 11.00 m (5.77 m) and for the rest of the bridge length (6.05 m), Figure 7.4. Concerning the bridge self-weight, it is adopted a concrete (γ_{conc}) and steel (γ_{steel}) specific weight, respectively, of 24.00 kN/m^3 and 77.00 kN/m^3. An additional load of 7.85 kN/m, corresponding to the self-weight of secondary elements (i.e. guardrail and central reservation, Figure 7.4), is incorporated.

Table 7.13. Interface properties.

Parameter		Value	
		High density	Low density
Normal stiffness (K_{NN})	[GPa]	1.00 * 10^4	1.00 * 10^4
Shear stiffness (K_{TT})	[MPa] (*per mm*)	9.29 * 10^3	5.57 * 10^3
Tensile strength (f_t)	[MPa]	1.00 * 10^2	1.00 * 10^2
Cohesion (c)	[MPa]	3.24	1.94

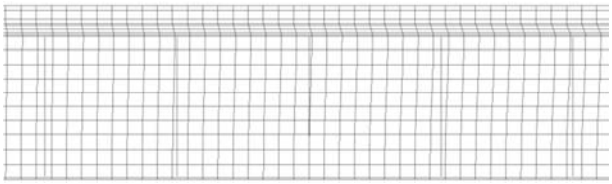


Figure 7.22. Finite element mesh.

Additionally, single loads are introduced, a load per each axis, to simulate the load test vehicles (Figure 7.14). The load position varies according to load case. Three load cases are considered, respectively, with the front axis of the first vehicle positioned at a distance of 26.00 m from A1, of 31.00 m from C1 and of 31.00 m from C2 (Figures 7.16 to 7.18). The load weight is distributed in a proportion of 40% - 60%, respectively, for front and rear axles. The total weight of each vehicle (to be converted into kN) and the distance between axles is indicated at Table 7.2. A total weight of 321.28 kN and of 320.20 kN is, respectively, considered to simulate the vehicles (A+C)/2 and (B+D)/2, Figure 7.15a. These vehicles are loading half of the bridge section considering the respective transversal symmetry.

For each load step it is monitored the support reaction and the vertical displacement at same points of the monitoring plans (17 and 66 m, Figure 7.13). The structure is first loaded with ten steps, with a factor of 0.10, with self-weight only (including pavement), and then it is introduced the vehicle loading with ten steps, with a factor of 0.10. The difference between obtained displacements for step 20 (with self-weight and vehicles) and for step 10 (with self-

weight only) is then computed. Accordingly, obtained vertical displacements may be, respectively, compared to experimental values, from Table 7.4. In order to develop this analysis, a Newton-Raphson algorithm is used. The main parameter values of this algorithm are indicated at Table 7.14.

Then, this structure is loaded up to failure by adding more steps with a factor of 0.10, respectively, by increasing the vehicle loading. The number of necessary steps to carry the structure up to failure will depend on the considered load case and on previously defined structural parameter values. In this situation an arc-length algorithm is used in order to better evaluate the peak behavior. A description of main parameters of this algorithm is presented at Table 7.15.

Table 7.14. Solution parameters (Newton-Raphson).

Solution method	Newton-Raphson
Stiffness / Update	Tangent / Each iteration
Limit number of iterations	40
Tolerance error	$1.00 * 10^{-2}$
Line search	With iterations

Table 7.15. Solution parameters (Arc-Length).

Solution method		Arc Length
Stiffness / Update		Tangent / Each iteration
Limit number of iterations		40
Tolerance error		$1.00 * 10^{-2}$
Arc length	Method	Consistently linearized
	Adjustment method	Constant
	Load / Displacement ratio	0.20
Line search		With iterations

Figure 7.23 presents the bridge vertical deformation for step 10 (self-weight). It is possible to detect a symmetric behavior, as expected, being the most critical sections those located at extreme spans. Those sections are positioned at interior support and at 40% of the span length, respectively, for negative and positive bending moment. These sections are considered to be critical as they present the highest stress values.

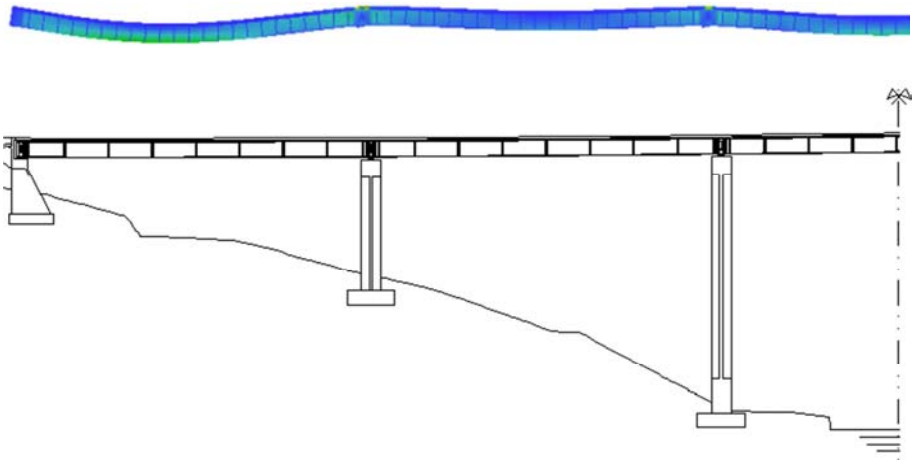
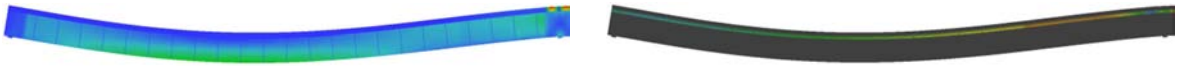


Figure 7.23. Bridge deformation (step 10).

In Figure 7.24a it is provided, for the first span, the normal strain values. Localized cracking is detected at negative bending moment region. A maximum tensile strain of 1.03 ‰ is verified in this region, at top fibbers. Concrete slab and part of the metallic girder are in tension. The maximum compressive strain value, in metallic girder, is of 0.39 ‰. At positive bending moment region the concrete slab is in compression. A maximum tensile strain of 0.64 ‰ is attained at metallic girder bottom fibers. Figure 7.24b indicates obtained values for interface tangential stresses. The analysis permit to verify that for positive bending moment region this value is of 0.66 MPa (low density) while for negative bending moment region this value is of 1.44 MPa (high density). Accordingly, these values are far from interface cohesion values (Table 7.13).



a) b)

Figure 7.24. Obtained results (step 10): a) normal strain; b) interface tangential stress.

Figure 7.25 indicates the bridge deformation for step 20 and for load case 1, in which the vehicles are applied at first span (Figure 7.16). By comparing with Figure 7.23 it is possible to verify that bridge deformation is no longer symmetric, due to the asymmetry of applied load. Therefore, it is verified an increase in horizontal strain for the first span of the bridge. Critical sections are at interior support and at 40% of the span length.

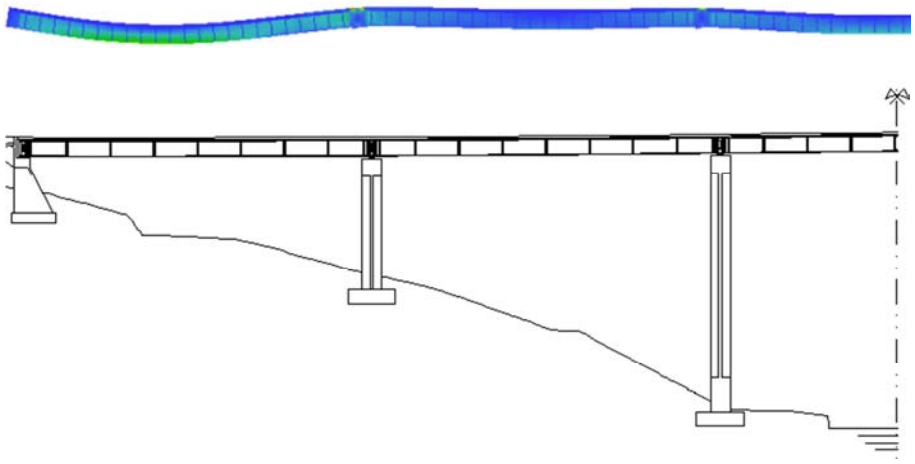
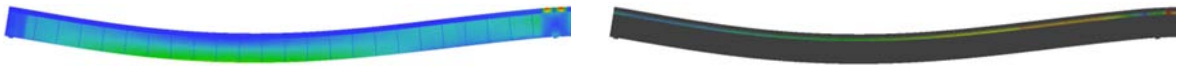


Figure 7.25. Bridge deformation for load case 1 (step 20).

At Figure 7.26a it is given the normal strains for critical span. It is verified an increase in cracking at negative bending moment region, above the support. For this section the maximum tensile strain at top fibers reaches 1.21 ‰. The concrete slab and part of the metallic girder are in tension. Compressive strain is only attained in the metallic girder, being its maximum value 0.45 ‰. For positive bending moment region, part of concrete slab is in compression and part is in tension, which means that neutral axis moved in upward direction. The maximum tensile strain at girder bottom fibers is of 0.76 ‰. Figure 7.26b presents obtained interface tangential stresses. It is verified a maximum value for positive bending moment region of 0.93 MPa (low density) and for negative bending moment region of 1.47 MPa (high density), which are still lower than the interface cohesion values (Table 7.13).



a) b)

Figure 7.26. Obtained results for load case 1 (step 20): a) normal strain; b) interface tangential stress.

At Figure 7.27 is represented the beam deformation for load step 70 and for load case 1. By comparing with Figure 7.25 it is possible to verify an increased deformation on the first bridge span, while all the other spans are released. Critical sections are those identified in previous step 20.

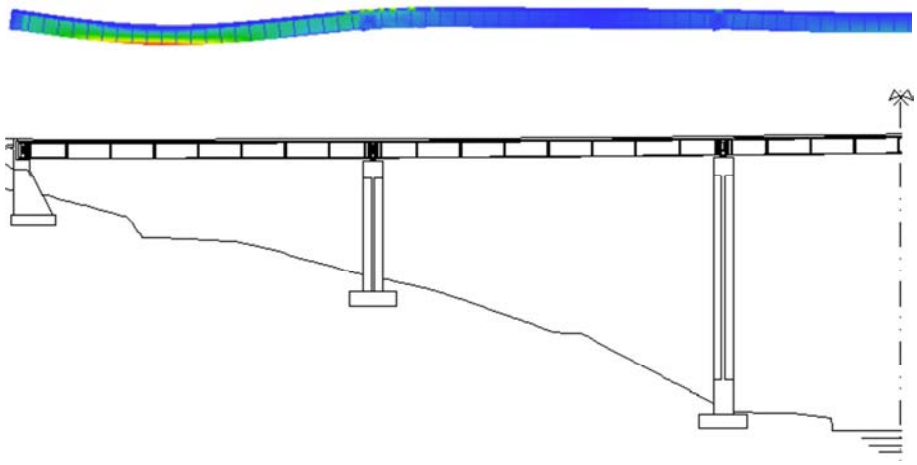
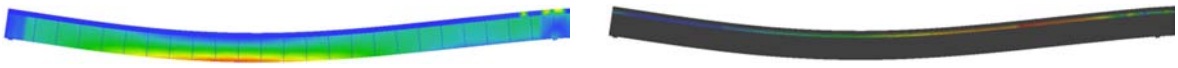


Figure 7.27. Bridge deformation for load case 1 (step 70).

Figure 7.28a presents the normal strain values for critical span. In the support section the tensile strain in the concrete slab is of 1.41 ‰. The metallic girder is partly in tension and partly in compression, presenting a maximum compressive strain of 0.47 ‰. The maximum tensile strain is verified at positive bending moment region, at the metallic girder bottom fibbers. A value of 1.88 ‰ is obtained. The metallic girder is, in this region, completely in tension and the concrete slab is partly in compression and partly in tension. In this situation, localized cracking, due to concrete crushing, is detected at positive bending moment region. Figure 7.28b presents the interface tangential stresses. In this situation it is obtained a value for positive bending moment region of 1.37 MPa (lowest density) and for negative bending moment region of 2.20 MPa (highest density), which are closer to defined interface cohesion values (Table 7.13), than previously.



a) b)

Figure 7.28. Obtained results for load case 1 (step 70): a) normal strain; b) interface tangential stress.

Figure 7.29 present the bridge deformation for step 186 (bridge collapse) and for load case 1. By comparing this figure with Figure 7.27, it is possible to verify a higher deformation at first span while all the others are being progressively released. Critical sections are the same as those identified at step 20 and 70.

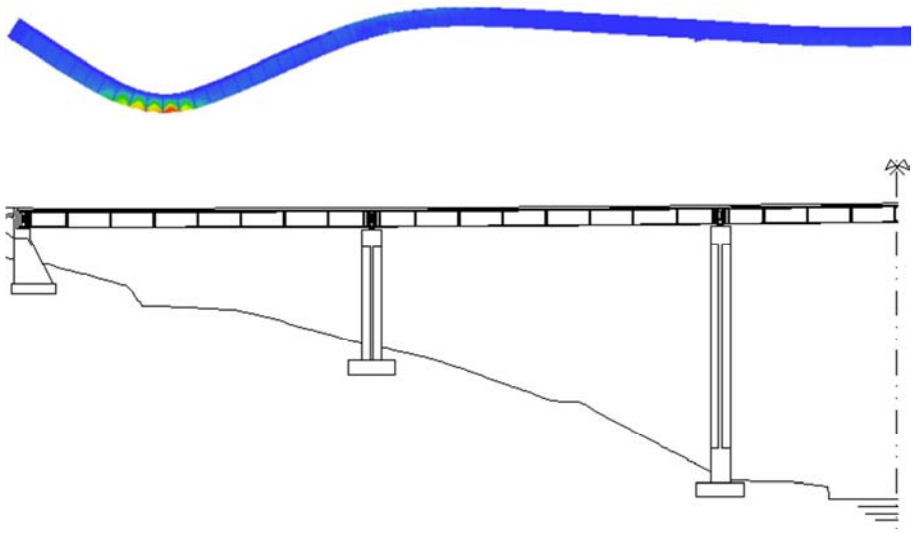


Figure 7.29. Bridge deformation for load case 1 (step 186).

Figure 7.30a presents the critical span normal strain values. Cracking in concrete slab is verified for both negative and positive bending moment region. In support section, the tensile strain is of 2.97 ‰, being the concrete slab completely in tension. The metallic girder is partly in tension and partly in compression, presenting a maximum compressive strain of 1.48 ‰. A maximum tensile strain of 11.87 ‰ is obtained in positive bending moment region. The metallic girder and a portion of the concrete slab are in tension, which indicates that neutral axis is positioned in top fibers. Figure 7.30b presents interface tangential stresses. It is verified a maximum value for positive bending moment region of 1.62 MPa (low density) and for negative bending moment region of 3.24 MPa (high density). This means that the cohesion value for high density region is attained (Table 7.13) and a redistribution of tangential stresses is produced.

a) b)

Figure 7.30. Obtained results for load case 1 (step 186): a) normal strain; b) interface tangential stress.

The collapse mechanism is characterized by two plastic hinges, respectively, at column C1 (step 70) and at 40% of the span length (step 186). Applied loads are redistributed from the first to the second hinge. It is important to notice that the numerical analysis stops, in this situation, for load step 186. A detailed analysis shows that reinforcement at negative bending moment region reaches, in this moment, its limit strain. Therefore, the redistribution capacity is over and the bridge failure is attained. A bending failure mode with concrete crushing, and yielding of both reinforcement bars and steel profile, is obtained.

Figure 7.31 presents obtained results for load case 1. The applied load, equal to the sum of all support reactions, is plotted against the obtained vertical displacements VD1* and VD2*, respectively, at 17 m and 66 m of the bridge length. These values correspond to VD1/VD2 and to VD3/VD4 transducers, indicated at Figure 7.13. It is possible to verify an inversion in VD2* values as the vehicle load increases, which reverses previous displacement due to self-weight.

Accordingly, the bridge presents an initial behavior, close to elastic, until the first hinge is formed. Then, applied stresses are redistributed to middle span region. The bridge stiffness is reduced but it still presents a high resistance capacity. This second phase ends with the materialization of the second hinge. During the third phase, the bridge presents a small increase in resistance capacity that is due to steel profile hardening. However, the increase in maximum load is less than 5%. This stage finishes with the attainment of the limit strain at reinforcement.

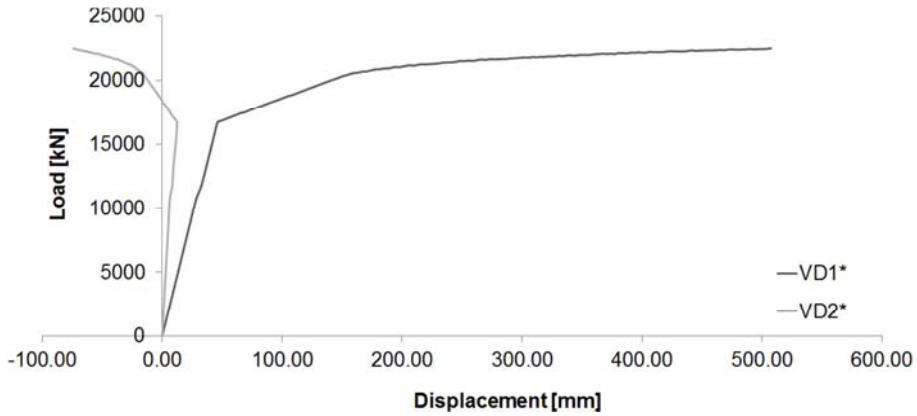


Figure 7.31. Obtained results for load case 1.

Figure 7.32 shows the obtained bridge deformation with load case 2, in which the vehicles are positioned in the second bridge span (Figure 7.17). In this situation the collapse mechanism is defined by three hinges, respectively, at column C1 and C2 (step 70) and at middle span (step 104). Applied loads are redistributed from negative to positive bending moment region. The numerical analysis stops, in this situation, for load step 104. A detailed analysis shows that reinforcement at negative bending moment region reaches, in this moment, its limit strain. Therefore, the redistribution capacity is over and the bridge failure is attained. A bending failure mode with concrete crushing, and yielding of both reinforcement bars and steel profile, is obtained.

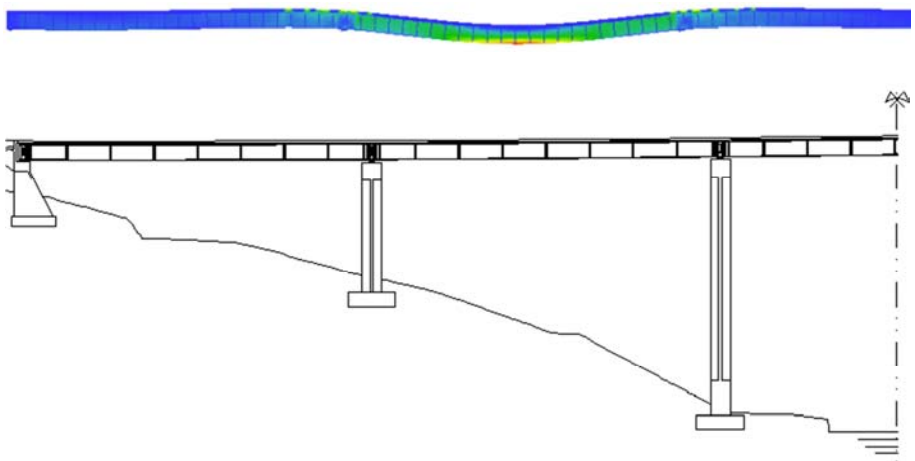


Figure 7.32. Bridge deformation for load case 2 (step 104).

Figure 7.33 presents obtained results, for load case 2. An inversion is verified in VD1* values as the vehicle load increases, which reverses previous displacement due to self-weight. Accordingly, the bridge presents an initial behavior, close to elastic, until the first hinges are formed, respectively, at column C1 and at column C2. These two hinges result in the diminishment of the bridge stiffness. Then, applied stresses are redistributed to middle span region. A third hinge appears at this region. Then the bridge presents a residual resistance capacity until its collapse, due to steel profile hardening. It is verified that the increase in maximum load is less than 5%.

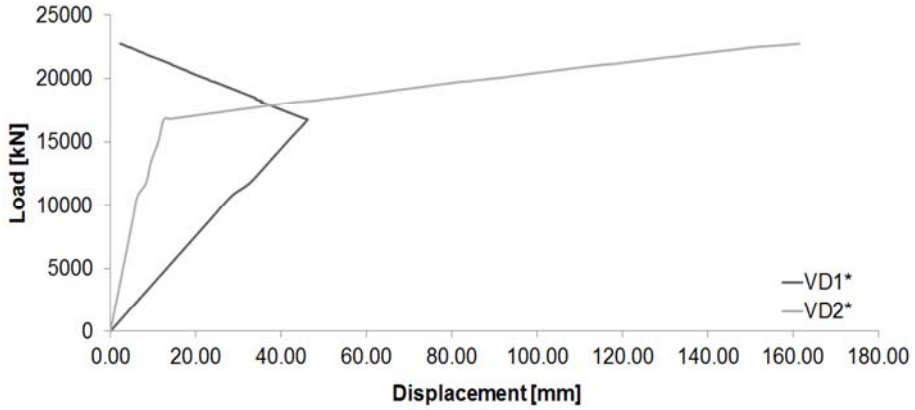


Figure 7.33. Obtained results for load case 2.

Figure 7.34 shows the obtained bridge deformation with load case 3, in which the vehicles are positioned in the third bridge span (Figure 7.18). In this case the collapse mechanism is defined by three hinges, respectively, at column C2 and C3 (step 70) and at middle span (step 210). Applied loads are redistributed from negative to positive bending moment region. The numerical analysis stops for load step 210. A detailed analysis shows that reinforcement at negative bending moment region reaches its limit strain. Therefore, the redistribution

capacity is over and the bridge failure is attained. A bending failure mode with concrete crushing, and yielding of both reinforcement bars and steel profile, is obtained.

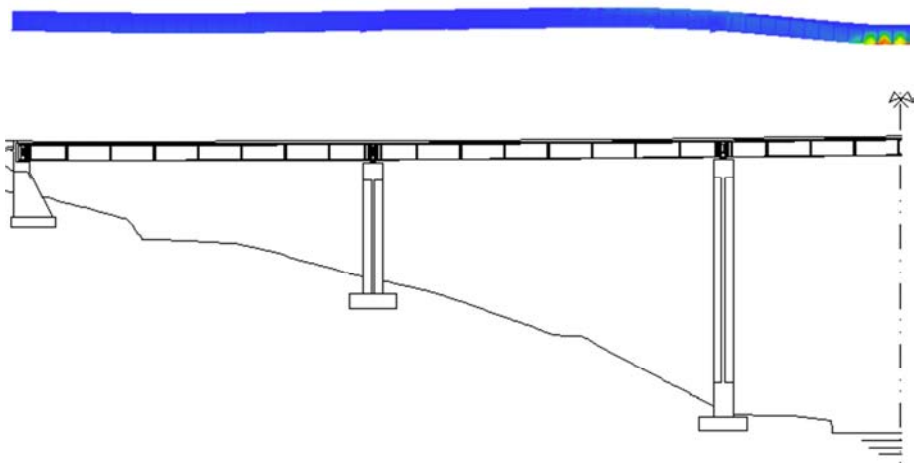


Figure 7.34. Bridge deformation for load case 3 (step 210).

Figure 7.35 presents obtained results, for load case 3. An inversion is verified in VD2* values with the increase in vehicle load, which reverses previous displacement due to self-weight. Accordingly, the bridge presents an initial behavior, close to elastic, until the first hinges are formed, respectively, at column C2 and at column C3. The bridge stiffness is then reduced. The applied stresses are redistributed to middle span region. A third hinge appears at middle of the bridge span. The bridge still presents a residual resistance capacity until its collapse, due to steel profile hardening. It is verified that the increase in maximum load due to hardening is less than 5%.

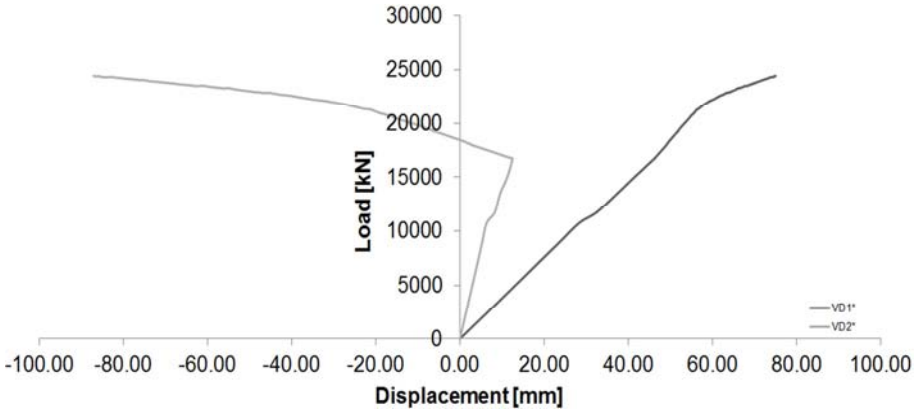


Figure 7.35. Obtained results for load case 3.

The analysis stops, for the three load cases, when the reinforcement limit strain at negative bending moment region is attained. Moreover, It is verified that the moment it stops vary with considered load case. It also varies with defined parameter values. In this situation, the developed model will be applied in a probabilistic analysis, considering different load cases,

for which the parameter values are randomly generated. Therefore, a maximum number of 300 steps are established in order to account for all possibilities.

Additionally, it is important to notice that within the probabilistic analysis it might appear a combination of parameter values that may lead to concrete or steel profile material failure. While for concrete material failure the nonlinear analysis software automatically stops, for steel profile this is not verified. In fact, it is important to notice that used model for metallic girder material (Figure 7.20c) presents an infinite hardening. Therefore, in this situation, the only way of determining the exact failure load is in post processing. However, a detailed study revealed that the possibility of this failure type is extremely low and thus, from a probabilistic point of view, the results from these models may be neglected.

Table 7.16 presents VD1* and VD2* displacement values for calibrated model and for the three considered load cases. The error between numerical and experimental data, given at Table 7.4, is also computed for each situation. Obtained error for LC1, measurement VD1*, is of 16.86%, and for LC2, measurement VD2*, is of 4.58%. These results validate the developed model, as they are the most significant ones. It is important to notice that the developed numerical model is less stiff than the real structure.

Other analyses were developed in order to justify some model simplifications. Therefore, and for each situation, both VD1* and VD2* displacement values and its difference from the reference model are respectively computed. Obtained results, errors and justifications are provided in Table 7.17. In this case the reference model is that previous described, whose results are indicated in Table 7.16.

Table 7.16. Obtained numerical results.

Load case	VD1*	Error	VD2*	Error
	[mm]	[%]	[mm]	[%]
LC1	17.77	16.86	-5.09	34.43
LC2	-4.90	23.56	14.32	4.58
LC3	1.40	24.32	-4.82	51.99

* Negative value corresponds to a displacement in upward direction.

The analysis presented in Table 7.17 permits to conclude that: (1) introducing more layers in reinforced concrete slab will decrease the edge-length ratio and will lead to a stiffer model. In order to overcome it, a small element size may be introduced but this will lead to a higher number of elements, increasing the computational cost. Therefore, a compromise between cost and accuracy is guaranteed by considering only two layers; (2) the error due to the

simulation of both metallic beam flanges with truss elements is negligible. As previous indicated the error in steel profile inertia presents a maximum of 10.52 % (Table 7.8), which is of minor importance when an overall analysis is developed. This simplification is therefore validated and obtained model becomes lighter; (3) when considering the medium density region at interface the model gets stiffer. However, this interface is located in a very short length (Figure 7.6), conducing thus to a higher computational cost. In order to keep the cost and accuracy compromise, the use of two interfaces is recommended; (4) the consideration of a pavement macro element slightly increases the bridge stiffness. However, the number of finite elements increases a lot which results in a higher computational cost. Therefore it is recommended not to consider this layer in order to establish a compromise between cost and accuracy; (5) removing the reinforcements at support sections will conduct to a less stiff model. In fact, these elements rigidify this region, increasing the rotation restriction. This effect becomes more relevant when evaluating the structural behavior up to failure. Therefore, these reinforcements should be considered in the numerical model; (6) removing the reinforcements at bridge span, slightly reduces the bridge stiffness. These elements are also important in an analysis up to failure load as they avoid any web instability. Therefore, it is recommended to include them on the numerical model; (7) results are practically the same either considering a quadrilateral or mixed finite element mesh, which means that the adopted element size is reasonable and that the finite element mesh, only composed by quadrilateral elements, is adequate. This validates the used finite element mesh.

Table 7.17. Simplifications and modeling errors.

Simplification	Justification	Error
		[%]
(1) Introducing five reinforced concrete slab layers	More stiff model	6.30
(2) Introducing both top and bottom flanges macro elements	No changes	≈ 0.00
(3) Introducing medium density region at interface	More stiff model	9.96
(4) Introducing pavement macro element	More stiff model	3.15
(5) Removing web reinforcements at supports	Less stiff model	5.23
(6) Removing web reinforcements at bridge span	Less stiff model	1.07
(7) Introducing mixed mesh	No changes	≈ 0.00

With this model a good compromise between cost and accuracy is guaranteed. For one side it is possible to run the model in ATENA® console in less than five minutes, in service phase, and less than half an hour, until failure load, which is a reasonable time for the probabilistic

assessment algorithm. For the other side, the developed numerical model is validated as obtained results are close to load test data (Table 7.16).

7.3.2. Sensitivity analysis

A sensitivity analysis is performed to identify the critical parameters, or, in other words, those that present a high influence on the structural behavior. This analysis is developed for the three load cases (Table 7.3) and for both service and failure region. In this situation, model identification is only performed in service phase as obtained data from load test is in this region. However, the probabilistic analysis is developed up to failure. Therefore, the fitness function and failure load variation with each input parameter are respectively evaluated in the sensitivity analysis in service phase and until failure load.

Studied parameters are related to material, geometry and physic sources. In order to perform the sensitivity analysis a standard deviation (σ) is respectively varied from each parameter mean value. An importance measure (b_k) is then computed for each evaluated parameter. A detailed description of this measure is given in chapter four. If this value is equal or higher than 10% (b_{lim}), the parameter will be considered as critical.

The analyzed parameters are 58, which are then grouped into 20, corresponding to: (1) concrete material - elasticity modulus (E_c), tensile strength (f_t), compressive strength (f_c) and facture energy (G_f); (2) reinforcing steel material - yield strength ($\sigma_{y,l}$), limit strength ($\sigma_{u,l}$) and strain ($\varepsilon_{lim,l}$); (3) laminated steel profile material - yield strength ($\sigma_{y,p}$) and hardening (H_p), which corresponds to six different profiles (Table 7.11); (4) steel to concrete interface - shear stiffness (K_{TT}) and cohesion (c), corresponding to two different stud distributions (Table 7.13); (5) reinforced concrete slab dimensions - width (b_{slab}) and height (h_{slab}), which corresponds to two concrete layers (Table 7.5); (6) laminated steel profile dimensions - web thickness (b_{web}), corresponding to four different cases (Table 7.7), and both top ($A_{fl,sup}$) and bottom ($A_{fl,inf}$) flanges area, corresponding to nine different cases (Table 7.8); (7) reinforcing steel area ($A_{s,l}$), corresponding to five different cases (Table 7.6); (8) superior reinforcement concrete cover (c_{sup}) at top reinforced concrete slab layer; (9) concrete specific weight (γ_{conc}); (10) pavement weight (p_{pav}), corresponding to two different values. All the other parameters are considered to be irrelevant for the sensitivity analysis.

Table 7.18 and 7.19 indicate the evaluated parameters and, respectively, their coefficient of variations (CV) and standard deviations (σ), used to compute the importance measures. For some of these parameters, such values are provided in bibliography [93, 196]. However, for others, they are obtained in an alternative way: (1) the laminated steel profile hardening modulus (H_p), which is computed through the CV of yield strength, limit strength and limit

strain, see Table 7.11; (2) steel to concrete interface parameters (K_{TT} and c), which are computed through the CV of concrete [48], headed stud material and geometry [131] parameters, through equations (7.1) to (7.8); (3) pavement self-weight (p_{pav}), which is a combination of the CV of the pavement thickness and of betuminous specific weight.

Table 7.18. Parameter variation in sensitivity analysis.

Parameter	CV [%]	Parameter	CV [%]	Parameter	CV [%]	Parameter	CV [%]
E_c	10.00	$\sigma_{y,l}$	5.00	H_p	20.00	$A_{fl,inf}$	2.00
f_t	20.00	$\sigma_{u,l}$	5.00	K_{TT}	10.00	$A_{s,l}$	2.00
f_c	10.00	$\varepsilon_{lim,l}$	15.00	c	12.50	γ_{conc}	3.00
G_f	10.00	$\sigma_{y,p}$	5.00	$A_{fl,sup}$	2.00	p_{pav}	10.00

Obtained results, for each parameter, are then grouped by specific sets and their mean value is computed. These results, from sensitivity analysis in service phase, are given at Figure 7.36. In this situation, the critical parameters are: (1) concrete elasticity modulus (E_c); (2) concrete tensile strength (f_t); (3) reinforced concrete slab height (h_{slab}); (4) concrete specific weight (γ_{conc}); and (5) pavement weight (p_{pav}). Accordingly, from 58 parameters, 7 of them are considered to be critical. Therefore, the computational cost of model identification is reduced.

Table 7.19. Parameter variation in sensitivity analysis.

Parameter	σ [mm]	Parameter	σ [mm]
b_{slab}	5.00	b_{web}	1.00
h_{slab}	10.00	c_{sup}	1.50

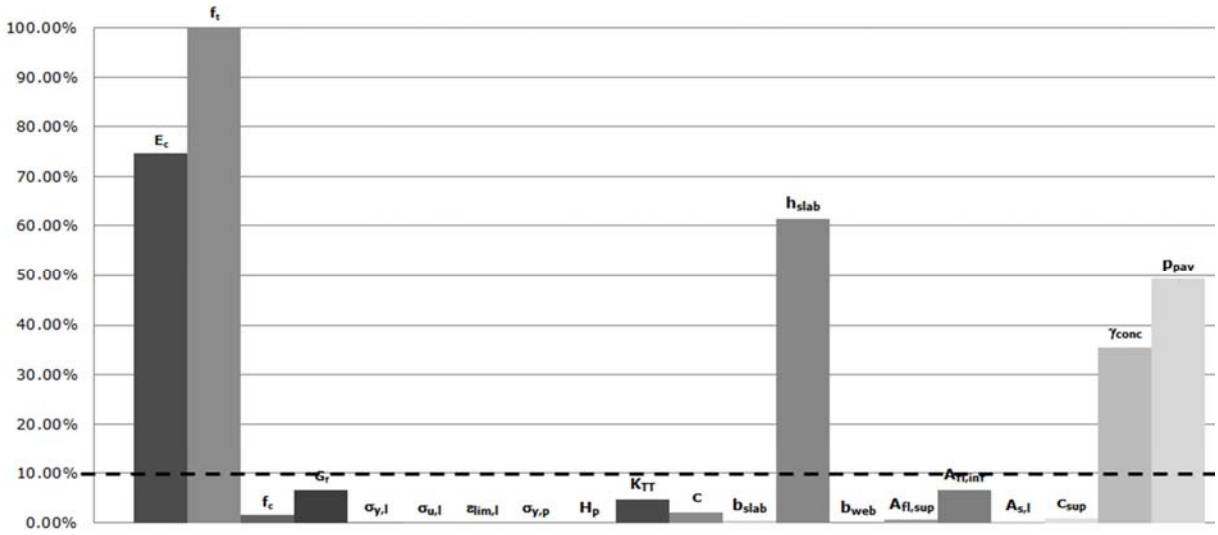


Figure 7.36. Sensitivity analysis (service).

The concrete elasticity modulus (E_c) and tensile strength (f_t) are extremely important for the overall structure behavior. Fracture energy (G_f) presents some importance, as it is proportional to concrete behavior area in tension, but not enough to be considered critical. The compressive strength (f_c) does not influence the structural behavior. In fact, for lower intensity loadings, concrete material, according to Figure 7.20a, is working in elastic region, being their elastic properties the most important ones. The influence of both reinforcing steel ($\sigma_{y,l}$, $\sigma_{u,l}$ and $\varepsilon_{lim,l}$) and laminated steel profile ($\sigma_{y,p}$ and H_p) material is very low. This is confirmed by the analysis of Figure 7.20b and 7.20c, for which these parameters are irrelevant for lower intensity loads. Both reinforcing steel area ($A_{s,l}$) and superior concrete cover (c_{sup}) do not influence the bridge behavior.

In respect to interface parameters it is important to notice that both shear stiffness (K_{TT}) and cohesion (c) parameters present some influence on overall structure behavior. Nevertheless, the influence of these parameters is not sufficient to be considered critical when analyzing the bridge behavior.

In respect to concrete slab, it is verified that its width (b_{slab}) does not influence the bridge behavior. In other way, the slab height (h_{slab}) presents a high influence on its behavior. This is explained by the inertia formula in which the slab inertia is cubically proportional to its height. Moreover, the variation in its height is higher than in its width (Table 7.19). This is due to the fact of its height being computed as a combination of precast and cast in-situ components, and, in this latter situation, a higher dispersion is obtained.

The influence of some laminated steel profile geometry parameters (b_{web} and $A_{fl,sup}$) is null. This is essentially due to the fact that very low variations are obtained for these parameters (Table 7.18 and 7.19). The main reason for that is the high quality control in fabrication

process of these girders. The bottom flange area ($A_{fl,inf}$) presents a high influence, but not enough to be considered critical. In fact, the composite section inertia is highly dependent of this parameter, due to its distance to the cross section gravity center, and according to Steiner's theorem.

Both concrete (γ_{conc}) specific weight and pavement (p_{pav}) weight present a high influence in the structural behavior. For one side these parameters influence the overall bridge behavior since the beginning of the test. For other side, the influence of pavement weight is higher than expected due to a high uncertainty in its value (Table 7.18), as it is a combination of betuminous specific weight with its thickness.

A sensitivity analysis, developed up to failure load, is further executed. The results from such analysis are provided in Figure 7.37. The critical parameters may be different from those previously identified. In Figure 7.37 it is not provided the results for the metallic girder yield strength ($\sigma_{y,p}$). In fact, it is verified, in this situation, a large variability in importance measure values for all evaluated plates (Table 7.11). Therefore, each yield strength importance measure is considered as a single value. Figure 7.38 presents obtained results.

Obtained results indicate as critical parameters: (1) concrete elasticity modulus (E_c); (2) concrete tensile strength (f_t); (3) concrete compressive strength (f_c); (4) reinforcing steel yield strength ($\sigma_{y,l}$); (5) metallic girder yield strength for plate 1 ($\sigma_{y,p1}$) and for plate 2 ($\sigma_{y,p2}$). Therefore, from 57 evaluated parameters, 6 of them are considered to be critical. This will make the probabilistic analysis more efficient from a computational cost point of view.

In respect to concrete parameters, both tensile (f_t) and compressive (f_c) strength are important. In fact, and according to Figure 7.20a, these parameters define the concrete material maximum load capacity in tension and in compression. The importance of the elasticity modulus (E_c) decreases in this situation, but still presents a high influence in the overall structural behavior. Fracture energy (G_f) is important, due to its relation to the concrete tensile area, but not enough to be considered as a critical parameter. The reinforcing steel yield strength ($\sigma_{y,l}$) is relevant for the structural behavior. This material is considered to be embedded in concrete slab and thus the composite section resistance will depend of it. The influence of reinforcing steel limit strength ($\sigma_{u,l}$) and strain ($\varepsilon_{lim,l}$) is very low. However, as indicated before, this latter parameter is important as it establishes the moment the bridge collapse.

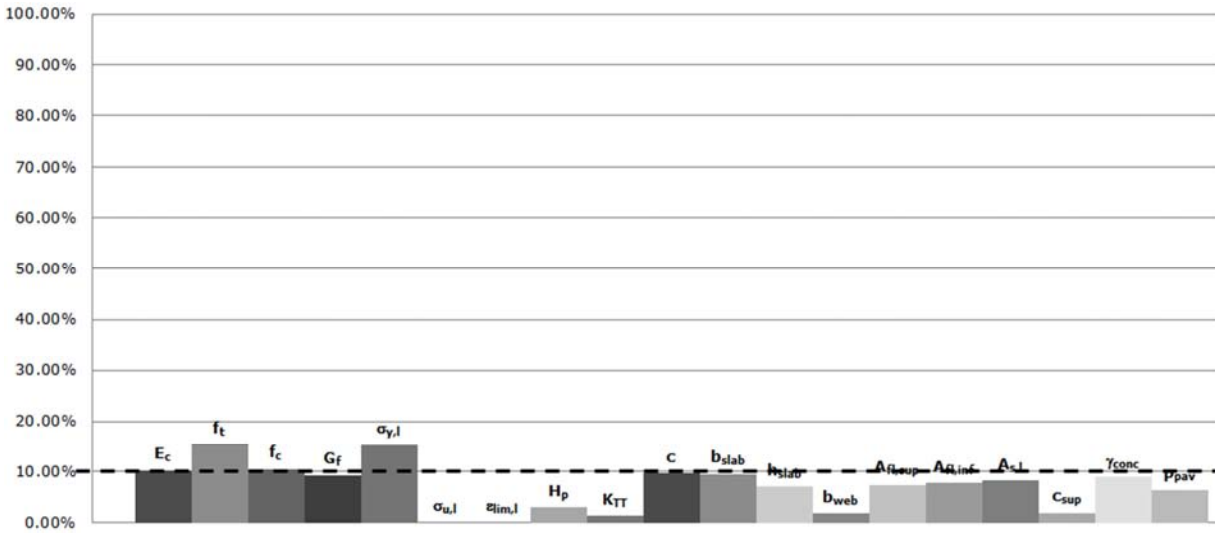


Figure 7.37. Sensitivity analysis (failure).

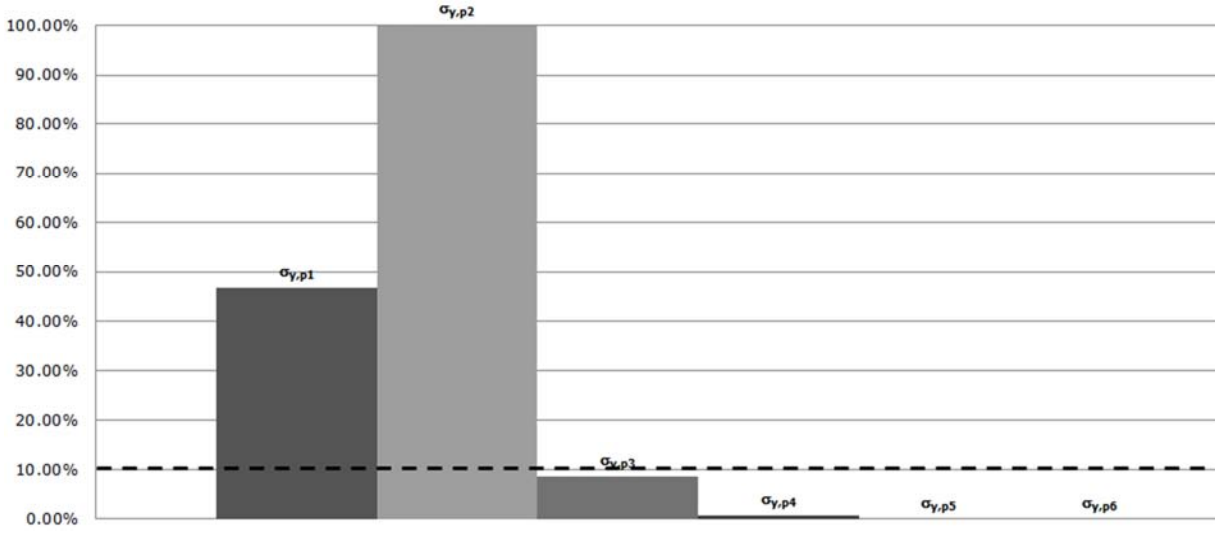


Figure 7.38. Sensitivity analysis of steel profile yield strength (failure).

The reinforcing steel area ($A_{s,l}$) influences the structure behavior, but not enough to be considered critical. In fact, once this material is completely embedded in concrete slab, the reinforcement area is necessary to determine the composite section resistance. The importance of the superior concrete cover (c_{sup}) in the structural behavior is low. In fact, this parameter presents a very low influence when computing the cross section resistance. Additionally, these two parameters present a low variation (Table 7.18 and 7.19).

According to Figure 7.38 the laminated steel profile yield strength ($\sigma_{y,p}$) presents a high importance on the structure behavior. The bridge resistance is highly influenced by the metallic girder, and, according to Figure 7.20c, this parameter is very important as it defines the transition between elastic and plastic regions of the stress-strain law. However, such effect is stronger in plate 1 ($\sigma_{y,p1}$) and plate 2 ($\sigma_{y,p2}$). All the other plates present a lower

influence on the structural behavior. The most important plates are, in fact, positioned at the first and second span, for the considered load cases (Table 7.3). The hardening parameter (H_p) practically does not influence the structure behavior.

In respect to interface, the influence of cohesion (c) parameter is high, but not sufficient to be considered critical. In fact, as load increases, the cohesion value is attained in critical sections being then progressively mobilized to closer sections. Additionally, it is verified a lower importance of shear stiffness parameter (K_{TT}). This parameter, in fact, defines the redistribution rate at interface but does not influence the failure load.

Regarding the section dimensions it is verified that both parameters, concrete slab width (b_{slab}) and height (h_{slab}), influence the structural behavior. The metallic girder geometry also influences such behavior. However, these parameters are not enough to be considered critical. This might be due to their low variability (Table 7.18). From all geometric parameters, the bottom flange area ($A_{fl,inf}$) is that which presents a high influence in the structure behavior. In fact, when computing the section inertia, the influence of this parameter is higher than of both web thickness (b_{web}) and top flange area ($A_{fl,sup}$), due to its distance to the cross section gravity center, and according to Steiner's theorem.

The importance of concrete self-weight (γ_{conc}) is high, but not enough to be considered critical. When evaluating the failure load variation, it is studied the carrying capacity of a structure already loaded with its self-weight. In this situation, the numerical analysis showed that localized cracking appears at negative bending region of the bridge (Figure 7.24a). In fact, this parameter may influence the cracking distribution along the bridge and, therefore, its carrying capacity. The pavement self-weight (p_{pav}) influences the structural behavior, but not sufficient to be considered critical. In fact, it is known that this value presents a little contribution in the structure self-weight.

7.4. Model identification

This methodology is applied to identify the critical parameter values, previously selected through a sensitivity analysis developed in service phase (Figure 7.36). Additionally, two other parameters corresponding to the horizontal spring stiffness at supports (k_1 and k_2) will be assessed within this procedure. These values were computed in numerical analysis but they are, in fact, unknown as they result of a combination of column stiffness with local pot bearing effects and foundation restrictions. Accordingly, they are determined within this process being considered as deterministic values in a further probabilistic analysis. Therefore, 9 parameters are adopted.

The basis of this methodology is to reorganize a set of numerical parameters in such a way that the numerical output best fits the measured data. This fact transforms this analysis into an optimization problem for which there is a function to be minimized. In this case, the fitness function is based in an approximation between numerical and experimental data. A detailed description of this function is given in chapter four. In order to process it, an optimization algorithm becomes necessary.

In this situation, evolutionary strategies algorithm in its plus version are used [29]. A detailed description of this algorithm is provided at chapter two. It begins with an initial population of critical parameter values, generated randomly, and then, new populations are generated through the use of evolutionary operators. In this case, it was defined a parent population (μ) and a parent for recombination (ρ) of 10 individuals, and an offspring population (λ) of 50 individuals. The algorithm is processed with five different starting points. A final population is extracted in each run. An engineer judgment procedure, based in the probability of occurrence of each individual, is developed to determine the most suitable individual, from those previously extracted. This procedure is detailed described in chapter four.

When using this procedure, several runs of the same numerical model are necessary. The fitness function value is computed in each run. The identification stops when a previously defined algorithm stopping criteria is reached. One of these criteria is related to the improvement on minimum fitness function value. It defines that this value, obtained from two generations separated of a pre-specified gap, should be less than or equal to a threshold value. This value is computed through the law of propagation of uncertainty [90, 91, 92], detailed described at chapter four. It may be interpreted as the methodology precision, once obtained results become more accurate with its decrease.

Other stopping criteria, like the maximum number of generations (1000), were also considered. The generation gap (n) used for the fitness function criteria is proportional to this number. In this situation, it was established that this value is 2% of the specified maximum generation's number. Therefore, the improvement on minimum fitness value is evaluated from a gap of 20 generations.

7.4.1. Tolerance criterion

In this situation the fitness function relates numerical and experimental vertical displacements, at 17 m and 66 m of the bridge length (Table 7.1), for the three load cases (Table 7.3). In order to perform model identification, it is necessary to determine the threshold value (ε) that defines the fitness function convergence criteria. In order to compute

this value, a division of uncertainty types, respectively, in measurement and modeling errors, is necessary.

In respect to measurement errors, only the sensor accuracy component is considered. In fact, all the other components may be neglected, such as [68]: (1) vibration noise caused by vehicles, not detected during the load test [99]; (2) environmental effects, that did not influence the obtained results [99]; (3) load positioning, perfectly controlled during the load test by introducing visible marks in the pavement; and (4) load intensity, also controlled as each vehicle weight was precisely measured beforehand (Table 7.2).

In this situation it were used LVDTs, with ± 25 mm of measurement field, to monitor the vertical displacement. The linearity value of these transducers, given by the manufacturer, is of 0.10%. Other effects, such as the cable length, should be considered when computing the sensor accuracy component. In this situation, the system transducer and cable is calibrated beforehand. In chapter four it is given different curves that relate the LVDT error with cable length for different cases. Accordingly, the following equation (7.9) is used,

$$u_{\exp_i} = 2.60 \cdot 10^{-3} \cdot L_i + 1.27 \cdot 10^{-1} \tag{7.9}$$

being L_i the cable length (m) for transducer *i*. In this situation, and according to Table 7.1, there are two different cable lengths, respectively, 17 m (VD1 and VD2) and 66 m (VD3 and VD4). Accordingly, two linearity values are obtained: (1) VD1 and VD2 = 1.71 * 10^{-1} %; (2) VD3 and VD4 = 2.98 * 10^{-1} %.

The measurement error presents a uniform PDF (type B) and so, according to JCGM [90, 91, 92], it should be divided by √3. Therefore, the following values are obtained: (1) VD1 and VD2 = 9.86 * 10^{-2} %; (2) VD3 and VD4 = 1.72 * 10^{-1} %. In order to compute the experimental uncertainty, through equation (7.10) and (7.11) [90, 91, 92], it is necessary to determine the experimental data derivative in respect to each component ($\partial y^{exp}/\partial x$ = 1.00 mm),

$$u^2_{\exp\ VD1,VD2} = 1.00^2 \cdot \left(9.86 \cdot 10^{-2}/100\right)^2 = 9.72 \cdot 10^{-7} \rightarrow u_{\exp\ VD1,VD2} = 9.86 \cdot 10^{-4} \text{ mm} \tag{7.10}$$

$$u^2_{\exp\ VD3,VD4} = 1.00^2 \cdot \left(1.72 \cdot 10^{-1}/100\right)^2 = 2.96 \cdot 10^{-6} \rightarrow u_{\exp\ VD3,VD4} = 1.72 \cdot 10^{-3} \text{ mm} \tag{7.11}$$

The following components are considered when computing the modeling error: (1) finite element method accuracy (VD1* = 3.53 * 10^{-1} %; VD2* = 2.81 %), obtained by comparing the previous numerical model with other with a higher number of load steps [69]; (2) mesh refinement (VD1* = 1.80 %; VD2* = 9.77 %), obtained by comparing the previous numerical model with other with a more refined mesh [69]; and (3) considered hypothesis (Table 7.17). The model exactitude component may be neglected [69].

The modeling error presents a uniform PDF (type B) and so, according to JCGM [90, 91, 92], it should be divided by √3, being obtained the following values: (1) finite element method accuracy - VD1* = 0.20 %, VD2* = 1.62 %; (2) mesh refinement - VD1* = 1.04 %, VD2* = 5.64 %; (3) considered hypothesis (Table 7.17): (a) introducing five reinforced concrete slab layers - VD1* and VD2* = 3.64 %, (b) introducing medium density region at interface - VD1* and VD2* = 5.75 %, (c) introducing pavement macro element - VD1* and VD2* = 1.82 %. In order to compute the numerical uncertainty, through equation (7.12) and (7.13) [90, 91, 92], it is necessary to determine the numerical results derivative in respect to each component ($\partial y^{num}/\partial x$ = 1.00 mm),

$$u^2_{num\ VD1^*} = 1.00^2 \cdot \left((1.04/100)^2 + (0.20/100)^2 + (3.64/100)^2 + (5.75/100)^2 + (1.82/100)^2 \right) = \\ = 5.08 \cdot 10^{-3} \rightarrow u_{num\ VD1^*} = 7.12 \cdot 10^{-2} \text{ mm} \quad (7.12)$$

$$u^2_{num\ VD2^*} = 1.00^2 \cdot \left((5.64/100)^2 + (1.62/100)^2 + (3.64/100)^2 + (5.75/100)^2 + (1.82/100)^2 \right) = \\ = 8.41 \cdot 10^{-3} \rightarrow u_{num\ VD2^*} = 9.17 \cdot 10^{-2} \text{ mm} \quad (7.13)$$

Once the experimental and numerical uncertainties are computed, it will be possible to determine the fitness function uncertainty. In order to obtain this value, it is necessary to compute the partial derivative of the fitness function in respect to both experimental and numerical data ($\partial f/\partial y^{num} = \partial f/\partial y^{exp} = 1/\max(y_i^{exp})$), for each monitoring point and for each load case. These values are provided at Table 7.20. They are computed based on the values of Table 7.4 (absolute values).

Table 7.20. Partial derivative values (mm^{-1}).

Load Case	VD1	VD2	VD3	VD4
LC1	0.06	0.07	0.24	0.28
LC2	0.24	0.26	0.07	0.08
LC3	0.54	0.54	0.29	0.34

In this situation, there are four comparing points (VD1 and VD2 - VD1*; VD3 and VD4 - VD2*) and three load cases (Table 7.3), which results in twelve components. The following Table 7.21 provides the computed fitness function uncertainty value for each component [90, 91, 92]. The global fitness function value is the square root of the sum of the square of these components. In order to determine the global uncertainty, the partial derivative of the global fitness function in respect to each component should be computed ($\partial f/\partial f_{VD1\text{-}VD1^*} = \partial f/\partial f_{VD2\text{-}VD1^*} = \partial f/\partial f_{VD3\text{-}VD2^*} = \partial f/\partial f_{VD4\text{-}VD2^*}$ = 1.00). The global uncertainty value (u_f = 9.14 %) is then obtained through the square root of the sum of all the values indicated at Table 7.22.

Table 7.21. Fitness function uncertainty values (%).

Load Case	VD1 - VD1*	VD2 - VD1*	VD3 - VD2*	VD4 - VD2*
LC1	0.45	0.45	2.23	2.23
LC2	2.24	2.24	0.51	0.51
LC3	4.93	4.93	2.64	2.64

Table 7.22. Square of fitness function uncertainty values (Table 7.21).

Load Case	VD1 - VD1*	VD2 - VD1*	VD3 - VD2*	VD4 - VD2*
LC1	0.20	0.20	4.98	4.98
LC2	5.00	5.00	0.26	0.26
LC3	24.30	24.30	6.98	6.98

The improvement on global fitness function minimum value (Δf) from two generations, separated of a specified gap (n), is given in chapter four. Accordingly, and assuming that the partial derivative of the improvement in respect to each component is unitary ($\partial \Delta f / \partial f_{i+n} = \partial \Delta f / \partial f_i = 1.00$), it is possible to obtain the respective uncertainty value through equation (7.14) [90, 91, 92],

$$u_{\Delta f}{}^2 = 1.00^2 \cdot (9.14/100)^2 + 1.00^2 \cdot (9.14/100)^2 = 16.71 \cdot 10^{-3} \rightarrow u_{\Delta f} = 12.92 \cdot 10^{-2} \quad (7.14)$$

Due to the fact of all uncertainty sources being from type B, a coverage factor (k) of 2 must be adopted [90, 91, 92]. The fitness value criterion establishes that its improvement (Δf) should be less than or equal to the threshold value (ε). Correspondingly, this value is obtained by multiplying the value from expression (7.14) by factor k, resulting in 25.84 %.

7.4.2. Obtained results

Each analysis provides a final population of 10 models. As five analyses were developed, 50 models are identified. Figure 7.39 shows the obtained fitness function value for the selected models. It is possible to verify a small dispersion in fitness function value between models belonging to the same population. In this case, the minimum value is obtained for the first analysis (model 1) while its maximum is obtained on the second analysis (model 20).

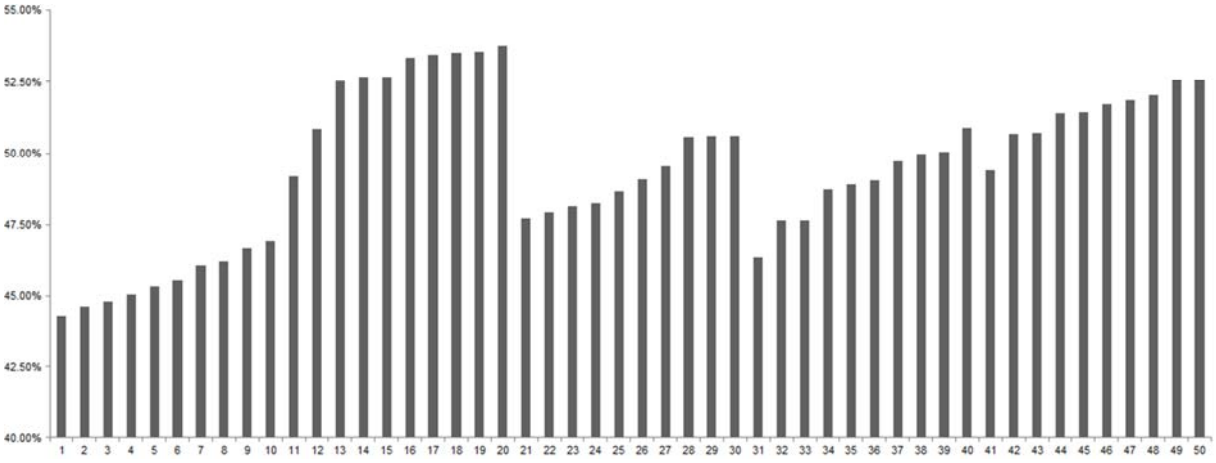


Figure 7.39. Model identification, fitness function value.

An engineering judgment evaluation, based in the probability of occurrence of each individual, is further developed in order to identify the most appropriate model. A detailed description of this procedure is given at chapter four. Figure 7.40 presents obtained probability values for all selected models. In this situation, model 20, from the second analysis, presents the highest value being thus selected.

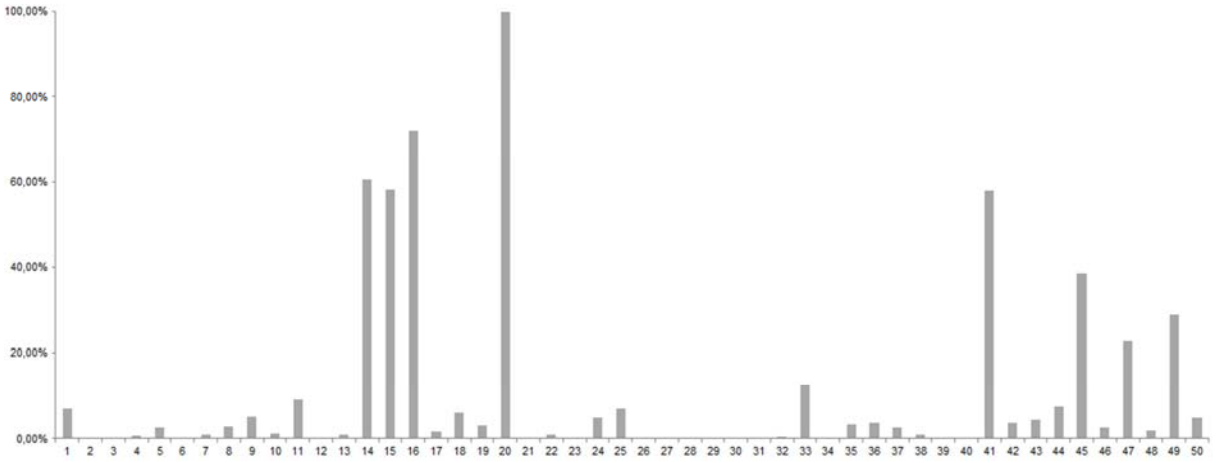


Figure 7.40. Model identification, engineering judgment evaluation.

Table 7.23 indicates the nominal and identified value (model 20) for critical parameters, previously selected through a sensitivity analysis. In the same table, between brackets, the bias factor, which represents the ratio between the identified and the nominal value, is also presented.

Obtained results permit to conclude that used concrete presents a higher quality than initially expected. In fact, both elasticity modulus (E_c) and tensile strength (f_t) values are greater than nominal ones, respectively, 3% and 15% higher. In respect to horizontal spring stiffness it is important to conclude that identified k_1 and k_2 values are respectively lower and higher than

the predicted. Both biases are far from one, which confirms the high uncertainty of this parameter. This value increases with the system column, pot bearing and foundation stiffness. Therefore it is possible to verify a higher stiffness in C2 and C3 (set 2) and a lower stiffness in C1 and C4 (set 1) support sections. The slab height (h_{slab}) is slightly higher, around 3%, than the design value. The concrete self-weight (γ_{conc}) is practically unchanged. However, the obtained pavement load (p_{pav}) is 15% higher than the design value. This might be due to the irregularity in betuminous thickness.

Table 7.23. Parameter values.

Numerical model			Nominal value	Model identification *
Parameter	E_c	[GPa]	35.00	35.98 (1.03)
	f_t	[MPa]	3.50	4.03 (1.15)
	k_1	[kN/m]	56.69	36.98 (0.65)
	k_2	[kN/m]	9.93	12.90 (1.30)
	h_{slab}	[m]	0.15	0.16 (1.05)
			0.25	0.25 (1.00)
	γ_{conc}	[kN/m^3]	24.00	24.34 (1.01)
	p_{pav}	[kN/m]	6.50	7.38 (1.14)
			6.81	7.73 (1.14)

* Bias factor is presented between brackets.

Table 7.24 indicates the fitness function value, considering the nominal and identified value for critical parameters (Table 7.23), and its improvement. In this situation it is possible to guarantee an improvement on this value of around 20% which is considered to be good. Table 7.25 presents VD1* and VD2* displacement values considering model identification. The error between numerical and experimental data, given at Table 7.4, is also computed for each situation.

Table 7.24. Fitness function values.

Numerical model	Value	Improvement
	[%]	[%]
Nominal values	67.33	-
Model identification	53.79	20.11

The following conclusions are obtained, through a comparison with Table 7.16: (1) a high decrease in both errors for LC1; (2) a decrease in VD1* and a slight increase in VD2* error for LC2; and (3) an increase in VD1* and a decrease in VD2* error for LC3. Therefore, this analyses permit to identify a model, in service region, that is more reliable than that used in design.

Table 7.25. Obtained numerical results.

Load case	VD1*	Error	VD2*	Error
	[mm]	[%]	[mm]	[%]
LC1	15.43	5.02	-3.57	7.87
LC2	-3.64	8.31	12.06	11.97
LC3	1.04	43.78	-3.48	8.92

* Negative value corresponds to a displacement in upward direction.

7.5. Complementary tests

7.5.1. Developed tests

During construction, other complementary tests were developed. These tests were important to control the precast slab quality and to classify used materials (concrete, reinforcing steel and laminated steel profile). These tests were developed by the manufacturer and they were requested by the constructor, Teixeira Duarte S.A.

The precast slab dimensions and concrete cover were detailed controlled by the manufacturer [183]. The obtained variation around these parameters was very small. In order to assess the concrete material quality, C40/50 [48], it was developed uniaxial compressive tests in cubic specimens according to NP EN 13747 [137], for precast concrete slab, and to NP 206-1 [138], for cast in-situ concrete. In this situation, it was only obtained the concrete compressive strength (f_c) [48].

The reinforcing steel quality, A 500 NR SD [48], was controlled through uniaxial tensile tests, according to LNEC E 456 specification [110]. Within this analysis, both yield ($\sigma_{y,l}$) and limit strength ($\sigma_{u,l}$), and also its limit strain ($\varepsilon_{lim,l}$) were determined. The metallic girder steel quality, S355 [49], was evaluated through uniaxial tensile tests, according to EN 10025-2 [45]. In this situation, the steel yield ($\sigma_{y,p}$), limit strength ($\sigma_{u,p}$) and also limit strain ($\varepsilon_{lim,p}$) were determined.

7.5.2. Obtained results

Obtained results for precast slab height (h_{slab}), from 52 proofs, are provided at Table 7.26. A bias value, which represents the ratio between the experimental and the nominal value, is also presented. Obtained results permit to conclude that geometric variations may be neglected within the whole analysis. This confirms the results from model identification procedure, see Table 7.23.

Table 7.26. Precast slab geometry.

Parameter		Length [m]	Nominal Value	Mean value (μ) *	Standard deviation (σ)
Precast slab height (h_{slab})	[m]	0 – 11	$1.50 * 10^{-1}$	$1.50 * 10^{-1}$ (1.00)	$4.84 * 10^{-4}$
			$2.50 * 10^{-1}$	$2.51 * 10^{-1}$ (1.00)	$8.06 * 10^{-4}$
		11 – 209	$1.50 * 10^{-1}$	$1.50 * 10^{-1}$ (1.00)	$4.84 * 10^{-4}$
			$2.50 * 10^{-1}$	$2.51 * 10^{-1}$ (1.00)	$8.06 * 10^{-4}$
		209 – 220	$1.50 * 10^{-1}$	$1.50 * 10^{-1}$ (1.00)	$4.84 * 10^{-4}$
			$2.50 * 10^{-1}$	$2.51 * 10^{-1}$ (1.00)	$8.06 * 10^{-4}$

* Bias factor is presented between brackets.

Obtained results from uniaxial concrete compressive tests for both in-situ and precast concrete are provided at Table 7.27. A bias value, which represents the experimental to nominal value ratio, is also given. These results are grouped as a low variation is verified from batch to batch. Therefore, 203 proofs are considered. Both elasticity modulus (E_c) and tensile strength (f_t) parameters are computed through obtained compressive strength (f_c) values [48]. Those results indicate that used concrete presents a slightly superior quality than the predicted. In fact, a higher value than the nominal one is obtained for all studied parameters. This confirms the results from model identification, see Table 7.23.

Table 7.28 provides the results, from 70 proofs, obtained from the uniaxial tensile tests on reinforcing steel. A bias value, which represents the ratio between the experimental and the nominal value, is also presented. Obtained values for yield ($\sigma_{y,l}$) and limit ($\sigma_{u,l}$) strength confirm the steel quality predicted in design. The only exception is verified with the limit strain ($\varepsilon_{lim,l}$) for which the obtained value is higher than the nominal one. The correlation coefficients (ρ_{ij}) between these parameters are provided at Table 7.29.

Table 7.30 provides the results, from 31 proofs, obtained from uniaxial tensile tests on laminated steel profile. A bias value, which represents the experimental to nominal value

ratio, is also presented. It is verified that steel material presents a slightly superior quality than that expected in design. In fact, a higher value is obtained for yield ($\sigma_{y,p}$) and limit ($\sigma_{u,p}$) strength. For the limit strain ($\varepsilon_{lim,p}$) the obtained value is far from the nominal one. The correlation coefficients (ρ_{ij}) between these parameters are provided at Table 7.31.

Table 7.27. Concrete material.

Parameter		Nominal Value	Mean value (μ) *	Standard deviation (σ)
Elasticity modulus (E_c)	[GPa]	35.00	37.04 (1.06)	0.63
Tensile strength (f_t)	[MPa]	3.50	3.98 (1.14)	0.14
Compressive strength (f_c)	[MPa]	48.00	56.86 (1.18)	3.21

* Bias factor is presented between brackets.

Table 7.28. Reinforcing steel material.

Parameter		Nominal Value	Mean value (μ) *	Standard deviation (σ)
Yield strength ($\sigma_{y,l}$)	[MPa]	560.00	562.94 (1.01)	21.42
Limit strength ($\sigma_{u,l}$)	[MPa]	644.00	645.49 (1.00)	20.36
Limit strain ($\varepsilon_{lim,l}$)	[‰]	80.00	96.39 (1.20)	35.78

* Bias factor is presented between brackets.

Table 7.29. Correlation coefficients (ρ_{ij}) in reinforcing steel material.

	$\sigma_{y,l}$	$\sigma_{u,l}$	$\varepsilon_{lim,l}$
$\sigma_{y,l}$	1.00	0.87	0.79
$\sigma_{u,l}$	0.87	1.00	0.92
$\varepsilon_{lim,l}$	0.79	0.92	1.00

The number of proofs is appropriate for quality control procedures. However, a lower number may be used in probabilistic assessment. The developed model identification algorithm and obtained results, given at Table 7.23, are validated with data obtained from complementary tests. This data is further used in the probabilistic assessment algorithm.

Table 7.30. Metallic girder material.

No.	Thickness	Yield strength ($\sigma_{y,p}$)			Limit strength ($\sigma_{u,p}$)			Limit strain ($\varepsilon_{lim,p}$)		
		[MPa]			[MPa]			[%]		
	[mm]	Nominal Value	Mean value (μ) *	Standard deviation (σ)	Nominal Value	Mean value (μ) *	Standard deviation (σ)	Nominal Value	Mean value (μ) *	Standard deviation (σ)
1	≤ 16	355	388.32 (1.09)	17.52	470-530	540.39 (1.08)	18.18	20-22	28.71	2.33 (1.37)
2	≤ 40	345	377.38 (1.09)	17.02	470-530	540.39 (1.08)	18.18	20-22	28.71	2.33 (1.37)
3	≤ 60	335	366.44 (1.09)	16.53	470-530	540.39 (1.08)	18.18	19-21	27.35	2.22 (1.37)
4	≤ 80	325	355.51 (1.09)	16.04	470-530	540.39 (1.08)	18.18	18-20	25.98	2.11 (1.37)
5	≤100	315	344.57 (1.09)	15.54	470-530	540.39 (1.08)	18.18	18-20	25.98	2.11 (1.37)
6	≤ 110	295	322.69 (1.09)	14.56	450-600	567.41 (1.08)	19.09	18-18	24.61	2.00 (1.37)

* Bias factor is presented between brackets.

Table 7.31. Correlation coefficients (ρ_{ij}) in metallic girder material.

	$\sigma_{y,p}$	$\sigma_{u,p}$	$\varepsilon_{lim,p}$
$\sigma_{y,p}$	1.00	0.98	0.94
$\sigma_{u,p}$	0.98	1.00	0.95
$\varepsilon_{lim,p}$	0.94	0.95	1.00

7.6. Probabilistic analysis

The previous developed numerical model is then converted into a probabilistic model by introducing randomness in its critical parameters, previous identified through a sensitivity analysis up to failure. A random number generation algorithm, based in a Latin Hypercube sampling (LHS) procedure [144], is introduced to generate different parameter values. This procedure is detailed described at chapter four. In this situation, 100 models are generated. These models are analyzed up to failure with a nonlinear structural analysis software [23, 24]. The failure load PDF is statistically computed for each load case. This PDF represents the bridge resistance curve. This curve is then compared to a loading one in order to assess safety.

7.6.1. Bayesian inference

The probabilistic numerical model is obtained by considering a PDF for each critical parameter. All the other parameters are considered to be deterministic. A Normal PDF is usually considered for these parameters. In some situations, for which the parameters cannot assume negative values, a Lognormal PDF should be used instead. Those curves are defined by mean (μ) and standard deviation (σ) value.

Obtained results from complementary tests are used to update the critical parameters, through a Bayesian inference algorithm [15]. This procedure is detailed described at chapter three. Therefore, four different probabilistic models are respectively considered: (1) nominal values; (2) model identification values; (3) nominal values + Bayesian inference; (4) model identification values + Bayesian inference. The mean values (μ) from the first model are those from design. The standard deviation (σ) is obtained by the application of the CV, from Table 7.18. The mean values (μ) from the second model are those obtained with model identification (Table 7.23). The standard deviation (σ) is obtained in a similar way by using the coefficients of variation from Table 7.18.

The other models consider the updated PDF. The Bayesian inference procedure was developed by considering both informative and non-informative (Jeffrey's) prior. Considered posterior PDF is that with the lowest standard deviation. Table 7.32 gives the critical parameter PDF for the considered probabilistic models. The correlation coefficients (ρ_{ij}) between those variables are provided at Table 7.33.

Table 7.32. Parameter values.

Parameters		Nominal values *		Model identification *	
		μ	σ	μ	σ
E_c	[GPa]	35.00 (37.04)	3.50 (0.63)	35.98 (36.51)	3.60 (0.52)
f_t	[MPa]	3.50 (3.99)	0.70 (0.15)	4.03 (3.99)	0.81 (0.15)
f_c	[MPa]	48.00 (56.86)	4.80 (3.24)	48.00 (56.86)	4.80 (3.24)
$\sigma_{y,l}$	[MPa]	560.00 (562.92)	28.00 (21.61)	560.00 (562.92)	28.00 (21.61)
$\sigma_{y,p1}$	[MPa]	355.00 (387.93)	17.75 (18.35)	355.00 (387.93)	17.75 (18.35)
$\sigma_{y,p2}$	[MPa]	345.00 (387.93)	17.25 (18.35)	345.00 (387.93)	17.25 (18.35)

* Bayesian inference values are presented between brackets.

Table 7.33. Correlation coefficients (ρ_{ij}).

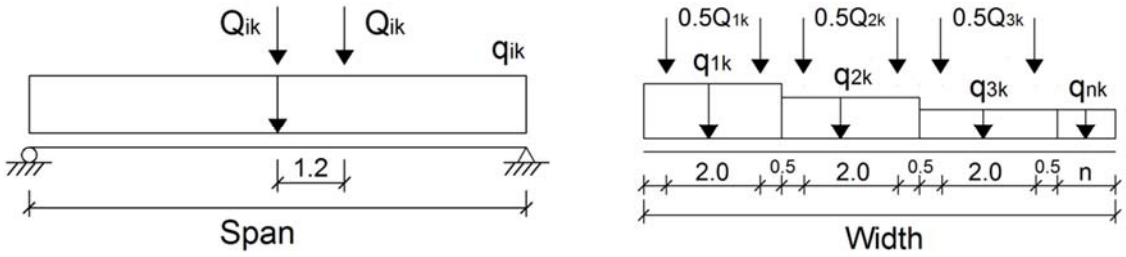
	E_c	f_t	f_c	$\sigma_{y,l}$	$\sigma_{y,p1}$	$\sigma_{y,p2}$
E_c	1.00	0.70	0.90	0.00	0.00	0.00
f_t	0.70	1.00	0.80	0.00	0.00	0.00
f_c	0.90	0.80	1.00	0.00	0.00	0.00
$\sigma_{y,l}$	0.00	0.00	0.00	1.00	0.00	0.00
$\sigma_{y,p1}$	0.00	0.00	0.00	0.00	1.00	0.90
$\sigma_{y,p2}$	0.00	0.00	0.00	0.00	0.90	1.00

The analysis of these results permit to conclude that: (1) concrete material presents a higher quality than that initially predicted. This was previously confirmed by model identification and complementary tests; (2) reinforcing steel material quality corresponds to that predicted in design. This was previously confirmed by complementary tests; (3) metallic girder material presents a higher quality than that initially predicted. This was previously confirmed by complementary tests; (4) the inference procedure reduces the standard deviation values, the only exception is verified with metallic girder yield strength ($\sigma_{y,p}$) for which obtained results from complementary tests are far from the nominal values.

7.6.2. Loading curve

In order to assess the bridge safety, the computed resistance PDF should be compared with the loading one. Usually this curve is obtained with the information of the real histogram of vehicle loading. However, in this situation such information does not exist and a standardized curve should be used instead. Accordingly, one of the most important models is the highway load model LM1 from EN 1991-2 [47], schematically represented in Figure 7.41. This model can be used on both local and global verification of bridge elements. The characteristic values of load intensity, defined as a 95^{th} percentile for a return period of 50 years, are provided at Table 7.34.

This model defines that applied load should be positioned at the most unfavorable position for the structural component and load effect in question. According to this model, a carriageway is divided into an integer number of 3.00 meters wide lanes (Figure 7.41b). Among these lanes, that causing the most unfavorable effect is labeled lane 1, with the second most unfavorable lane 2, etc. Space not occupied by these lanes is named remaining area.



a) b)

Figure 7.41. Highway traffic load model LM1 from EN 1991-2 [47].

Correspondingly, and for the analyzed composite bridge deck (Figure 7.4), the total applied load, per meter, is then computed according to expressions (7.15), (7.16) and (7.17),

$$Q_{ik} = 300 + 200 + 100 = 600\ kN \tag{7.15}$$

$$q_{ik}^{1} = 9.00\,kN/m^2 \cdot 3.00 + 2.50\,kN/m^2 \cdot (13.10 - 3.00) = 52.25\,kN/m \tag{7.16}$$

$$q_{ik}^{2} = 9.00\,kN/m^2 \cdot 3.00 + 2.50\,kN/m^2 \cdot (6.08 \cdot 2 - 3.00) = 49.92\,kN/m \tag{7.17}$$

being Q_{ik} the concentrated load, q_{ik}^{1} the distributed load for 11.00 m to 209.00 m of the bridge length (L_1) and q_{ik}^{2} for the first and final 11.00 m of the bridge length (L_2). This division is necessary to take into consideration the effective width variation along the bridge. As only half section is considered in numerical model, due to its symmetry, the obtained values are then divided per two. Therefore, a concentrated load of 300 kN (Q_{ik}), and a distributed load of 26.12 kN/m (q_{ik}^{1}) and of 24.96 kN/m (q_{ik}^{2}) are obtained.

Table 7.34. Intensity of loads in highway traffic load model LM1 from EN 1991-2 [47].

Lanes	Concentrated loads (Q_{ik})	Distributed loads (q_{ik})
	[kN]	[kN/m^2]
Lane 1	300	9.00
Lane 2	200	2.50
Lane 3	100	2.50
Other lanes *n*	0	2.50
Remaining area	0	2.50

In this situation, the considered load cases will correspond to those defined at load test for which the vehicles are positioned at critical position (Table 7.3) [99]. Therefore the concentrated load position will correspond to the gravity center of each vehicle. According to Figure 7.41a two concentrated loads (Q_{ik}), spaced of 1.20 m, are considered.

Correspondingly, for LC1 it is considered that applied loads are positioned at 18.52 m and at 17.32 m of bridge length (Figure 7.16), for LC2 at 67.52 m and at 66.32 m of bridge length (Figure 7.17), and for LC3, at 111.52 m and at 110.32 m of bridge length (Figure 7.18).

The highway traffic load model LM1 [47] is then transformed into a loading PDF, in order to assess the bridge safety. A Gumbel PDF is therefore considered [196]. Accordingly, the total applied load (F_{95}), corresponding to 95th percentile for a return period of 50 years, is given by equation (7.18),

$$F_{95} = q_{ik}^{1} \cdot L_1 + q_{ik}^{2} \cdot L_2 + Q_{ik} \cdot n = 26.12 \cdot (209.00 - 11.00) + 24.96 \cdot (11.00 \cdot 2) + 300 \cdot 2 = 6321.82 \; kN \quad (7.18)$$

being n the number of concentrated loads (Q_{ik}). Additionally, and according to Wisniewski [196], a CV of 15% is considered for loading curve. Therefore, and through an iterative analysis, the following parameter values, indicated at Table 7.35 are obtained for loading PDF.

Table 7.35. Loading PDF (S).

Parameter	Value
	[kN]
Location parameter (μ)	4606.00
Shape parameter (β)	577.68
Mean value	4939.40
Standard deviation value	740.91
95th percentile value (F_{95})	6321.82

The loading values are then randomly generated, according to a LHS algorithm [144], and taking into consideration the obtained parameters (Table 7.35). The generated values are then compared with those from the bridge resistance curve in order to assess its safety.

7.7. Safety assessment

The safety assessment consists in a comparison between resistance (R) and loading (S) curves for the bridge in analysis, according to the limit state function ($Z=R–S$). In this situation, the highway traffic load model LM1 [47] is used. This model is different from that used in load test as it presents a combination of single and distributed loads. This results in a different bridge behavior and failure load value. In order to minimize this effect the simplified load model is adjusted in a way that single loads are positioned in agreement with those from load test.

The load model is therefore introduced in previously developed numerical model [23, 24] and then, the bridge is loaded up to failure. Accordingly, 10 load steps of factor 0.10 are initially considered in order to take into account the structure self-weight and then 10 more steps of factor 0.10 are added in order to consider the load model (service region). Finally, the structure is carried up to failure by adding more 300 steps of factor 0.10. This value will guarantee that failure is always attained.

As previously indicated, in this situation four probabilistic models are analysed, being obtained a set of failure load (R) values for each analysis. A Normal PDF is therefore adjusted to these results. Obtained mean (μ) and standard deviation (σ) value are respectively indicated in Table 7.36.

Table 7.36. Resistance PDF (R).

Numerical model	LC1		LC2		LC3	
	μ	σ	μ	σ	μ	σ
	[kN]	[kN]	[kN]	[kN]	[kN]	[kN]
Nominal values	24796.00	902.40	39294.00	700.82	35991.00	659.80
Model identification	24749.00	936.94	39550.00	780.70	35990.00	665.32
Nominal values + Bayesian inference	26770.00	904.49	41210.00	751.58	37148.00	648.64
Model identification + Bayesian inference	26769.00	911.05	41188.00	742.84	37270.00	649.74

The analysis of this results permit to conclude that, for considered load cases, and for developed numerical model, the overall bridge resistance is substantially higher than the applied load model (Table 7.35). Additionally, it is identified that the critical load case corresponds to the first one as it presents a lower mean value. The other two load cases present similar results. The CV for the first load case is higher than that obtained for the other two situations.

By comparing the obtained resistance PDF for the four probabilistic models, it is possible to conclude that model identification practically did not change the obtained results. This is due to the fact that the majority of assessed parameters in model identification, in service phase, do not influence the bridge behavior up to failure. The application of a Bayesian inference procedure, considering both nominal and identified parameter values, increases the failure load. This is due to an identified increment in the quality of concrete and laminated steel

profile materials (Table 7.32). This confirms an additional structural resistance capacity reserve which was not initially identified.

When evaluating the CV it is possible to conclude that both initial and model identification models provide similar results. A slightly decrease of this value is verified with Bayesian inference procedure. This is due to a decrease on standard deviation value of some updated parameters (Table 7.32), confirming an increment in the degree of belief of obtained resistance curves.

The safety assessment of this bridge consists in the evaluation of the limit state function (Z) which compares the resistance curves (R), whose parameters are indicated at Table 7.36, with loading curve (S), whose parameters are indicated at Table 7.35. Therefore it is necessary to sample these curves and to compute the respective failure probability (p_f) and reliability index (β). Obtained values from this safety assessment procedure are indicated at Table 7.37.

Table 7.37. Safety assessment.

Numerical model	p_f *	β *
Nominal values	$3.59 * 10^{-16}$	8.32
Model identification	$4.26 * 10^{-16}$	8.30
Nominal values + Bayesian inference	$1.17 * 10^{-17}$	8.73
Model identification + Bayesian inference	$1.19 * 10^{-17}$	8.72

* Considered the most critical value from all load cases.

The analysis of obtained results confirms what was previously specified, namely: (1) obtained failure probability (p_f) and reliability index (β) values are high, which indicates an excellent structural performance; (2) obtained results from probabilistic numerical model, considering nominal and identified parameter values, are close; (3) the application of a Bayesian inference procedure increases the safety index (β) and decreases the probability of failure (p_f).

By comparing these results with proposed target reliability indexes (β_{target}), and considering that an overall analysis of the structure is developed, it is possible to conclude that the assessed bridge is in very good situation ($9 > \beta \geq 8$) [55]. This is in agreement with Tabsh and Nowak [180] guidelines, which indicate that a β-value higher than 5-6 corresponds to a structure with a very good performance. Once the assessed bridge is new, obtained β-value is within the expected.

This increment in β values when evaluating the overall system is due to: (1) the use of a nonlinear structural analysis, which considers the structure geometry and the materials behavior as they are in reality; (2) the consideration of a whole structure analysis, and not a section by section evaluation procedure, which takes into account the stress redistribution within the structure. In this situation there are other reasons that explain this increment, namely: (1) the consideration of structural elements which were not considered in design, as the web and support reinforcements; (2) the use of transversal symmetry which takes out of this evaluation load cases which imply torsional effects that might be critical.

7.8. Conclusions

This chapter presents the application of the developed probabilistic assessment methodology to a composite bridge (Sousa river bridge). It begins with a description of the structure, the executed load test and obtained results. It is also presented the developed numerical model and sensitivity analysis, realized both in service and failure region. The analysis permits to conclude that the bridge failure mode is of bending with concrete crushing and steel profile yielding. The bridge collapse mechanism is constituted by two or three hinges, depending on the load case.

In a second step, the model identification algorithm is applied with the results obtained during load test in service phase. Within this analysis, the tolerance value that is a combination of modelling and measurement errors is computed. Such analysis permit to conclude that concrete material quality is superior than expected, that spring stiffness is a highly uncertain variable, that concrete slab height is well controlled during construction, as well as the concrete weight, and that the betuminous pavement thickness is higher than the predicted in design.

A description of complementary tests and of obtained results is also provided. Uniaxial compressive tests with concrete and uniaxial tensile tests with both reinforcement and steel profile material are executed. The slab dimensions are strictly controlled and they practically do not change within the whole bridge. Concrete quality is superior to what was initially expected. Reinforcement presents the same quality as predicted in design. Steel profile material presents a slightly superior quality.

A probabilistic analysis is further developed. In order to do so, the developed deterministic model is converted into a probabilistic model by introducing randomness in its input variables. A LHS algorithm is introduced to generate the parameter values. Four probabilistic models are obtained, respectively, with nominal and identified parameter values, and considering, or not, a Bayesian inference procedure. This procedure will be developed with

results from complementary tests, confirming a higher quality of concrete and metallic girder steel material.

The developed model is then carried up to failure by introducing a standard load model in a location which corresponds to the load test one. Obtained resistance curve is then compared with loading curve. A reliability index is therefore obtained from safety assessment. Such index is then compared with a target one. This comparison indicated an excellent structural performance. The application of this framework, within this example, points out the advantages of the developed algorithm and, especially, of both model identification and Bayesian inference procedures.

8 Conclusions

8.1. Concluding remarks

According to OECD [140, 141, 142] there is a need for novel frameworks that can help countries in prioritizing their budget expenditures in asset management. One of these frameworks is the structural assessment algorithms, as they give an accurate representation on the current structural condition of existing structures. Such algorithms are applied to civil engineer structures.

The present thesis has, as its main objective, the development and validation of an advanced probabilistic assessment framework. This framework is implemented with commercially available nonlinear structural analysis software and tested on representative structural examples, respectively, a reinforced concrete beam and a steel-concrete composite beam, both tested in laboratory environment, and a steel-concrete composite bridge, with an in situ field test.

This framework is divided in two main steps, respectively: (1) step 1: model identification; (2) step 2: probabilistic analysis. In model identification the numerical results are best fitted to experimental data by adjusting model parameter values. This algorithm is important as it permits to determine the quality of used materials, the definition of structural geometry, and other physical parameters related to support conditions or interface connection.

Different optimization algorithms are tested and their efficiency is evaluated when implemented in the model identification framework. From those algorithms, the evolutionary

strategies in its plus version [29], is selected. Obtained results reveal this algorithm to be more efficient than the others, as the improvement in fitness function is higher and, at same time, the computational cost is acceptable.

Both modeling and measuring errors are combined and incorporated in an indirect way in the model identification procedure. A tolerance value is computed, based in these errors, through the law of propagation of uncertainties [90]. This threshold value defines the fitness function stop criterion. Table 8.1 provides the obtained tolerance values for all analyzed structures.

The analysis of this table permits to conclude that obtained tolerance values in laboratory controlled conditions are lower than that obtained in-field (composite bridge). The tolerance value increases with the number of comparing points in fitness function, which varies from service to failure region, and with the nonlinearity of the structure behavior. This nonlinear behavior increases with the simulation of fixed supports (reinforced concrete beams) and with partial connection (composite beams). The composite bridge behavior is highly nonlinear.

A detailed description of both measurement and modeling errors is also presented. The analysis of measurement error sources permit to conclude that: (1) it is not reasonable to use long cable lengths; (2) the calibration of the system transducer, cable and related equipment's, before monitoring, is recommended; and (3) wrapping cables should be avoided in any monitoring system.

Table 8.1.Tolerance values.

<table>
<tr><th colspan="3">Structure type</th><th>Tolerance value [%]</th></tr>
<tr><td rowspan="4">Reinforced concrete beam (chapter five)</td><td rowspan="2">Pinned-pinned beam</td><td>Service</td><td>0.72</td></tr>
<tr><td>Failure</td><td>0.72</td></tr>
<tr><td rowspan="2">Pinned-fixed beam</td><td>Service</td><td>4.17</td></tr>
<tr><td>Failure</td><td>4.58</td></tr>
<tr><td rowspan="4">Composite beam (chapter six)</td><td rowspan="2">Full connection beam</td><td>Service</td><td>0.08</td></tr>
<tr><td>Failure</td><td>0.12</td></tr>
<tr><td rowspan="2">Partial connection beam</td><td>Service</td><td>0.09</td></tr>
<tr><td>Failure</td><td>0.25</td></tr>
<tr><td colspan="2">Composite bridge (chapter seven)</td><td>Service</td><td>25.84</td></tr>
</table>

A population of models is respectively obtained from model identification. All these models are a possible solution for the problem in analysis. Therefore, an engineering judgment

criterion, based in the probability of occurrence of each model, is used to identify the most likely model from the population. The improvement in fitness function value for this model indicates a close approximation between experimental and numerical data. This improvement varies with the structure type and the intensity of the applied load, which varies from service to failure region, according to Table 8.2.

The analysis of this table permits to conclude that: (1) the improvement in service region is, generally, higher than in failure; (2) the improvement typically decreases with the number of assessed parameters; (3) higher improvements are obtained for laboratory tested structures; (4) the improvement decreases with the nonlinearity of the structure behavior.

Table 8.2. Improvement in fitness function value.

Structure type			Improvement [%]	Number of parameters
Reinforced concrete beam (chapter five)	Pinned-pinned beam	Service	92.97	7
		Failure	75.90	9
	Pinned-fixed beam	Service	55.26	9
		Failure	28.69	12
Composite beam (chapter six)	Full connection beam	Service	82.80	8
		Failure	88.99	16
	Partial connection beam	Service	75.33	8
		Failure	71.51	16
Composite bridge (chapter seven)		Service	20.11	9

A sensitivity analysis is developed before model identification to determine the critical parameters. The identified parameters in service region may not coincide with those in failure. Moreover, the assessed values for these parameters are different in service and in failure region. This results in a complex extrapolation of results from service to failure region.

In real situations, as the analyzed composite bridge, existing structures are evaluated through a load test that is developed in service phase. Therefore, it is important to notice that model identification, supported in load test data, provides an incomplete representation of this structure behavior in failure region. Complementary characterization tests are thus recommended in this case.

In order to develop the probabilistic analysis, uncertainty is incorporated in the updated deterministic numerical model through the introduction of randomness in model parameters. The input values are randomly generated through a Latin Hypercube sampling (LHS)

algorithm [144]. While for some of these parameters it is possible to find a probability density function (PDF) and correlation coefficients in bibliography [93], for others they are defined according to experience (e.g. interface connection parameters).

The statistical uncertainty in model parameters PDF may be reduced with complementary data, obtained in laboratory characterization tests, nondestructive tests or permanent monitoring systems. In order to do so, a Bayesian inference algorithm [15] is used. This procedure is important to improve the developed probabilistic numerical model.

Two updating schemes are thus incorporated in the developed probabilistic assessment framework, respectively: (1) model identification; (2) Bayesian inference. An important decision deals with the time in which each procedure should be used. Model identification is applied with deterministic models, it requires a higher computational cost, and should be used when a high gap is verified between numerical and experimental data. Bayesian inference is used with probabilistic models. It is of easy computation and may be applied whenever there is complementary data from existing structure.

Accordingly, and in order to validate the developed framework, laboratory tested structures are analyzed with the following probabilistic models: (1) initial model; (2) model identification; (3) initial model + Bayesian inference; (4) model identification + Bayesian inference. Model identification results, both in service and in failure region, are considered for those structures. For these structures it is introduced an index-*p* that will permit to evaluate the reliability of the obtained probabilistic model. Obtained results are provided at Table 8.3. Generally, it is verified an improvement in the reliability of probabilistic model with the application of model identification and Bayesian inference. Obtained results validate the application of this framework in laboratory tested structures.

The developed probabilistic assessment framework will provide an updated resistance curve of evaluated structure that might be used in safety assessment. This algorithm will permit to identify, in some situations, an additional structure capacity reserve. This is important as some structures are initially considered to be unsafe and with the application of this algorithm it is determined that they are, in fact, in safe region.

The algorithm is also implemented with a real case of a composite bridge submitted to a load test [99]. In this situation, it is not possible to compute the reliability of probabilistic model (index-*p*) due to the lack of experimental data in failure region. Nevertheless, the model identification is developed in service region, being the Bayesian inference used to update some of input parameters. A safety assessment procedure is then developed with the

updated resistance curve. Obtained results validate the application of this framework in real structures.

The developed probabilistic assessment algorithm incorporates different sources of uncertainty at different stages, namely: (1) measurement and modeling errors in model identification; (2) physic uncertainty in input parameters of probabilistic numerical model. The statistical uncertainty may be reduced with Bayesian inference, if complementary data is available. Human errors might be identified in model identification.

Table 8.3.Index-*p* value.

Structure type			Liability index [%]			
			Initial	Model identification	Initial + Bayesian inference	Model identification + Bayesian inference
Reinforced concrete beam (chapter five) *	Pinned-pinned beam	Service	95.02	95.40	-	-
		Failure		99.41		-
	Pinned-fixed beam	Service	96.97	89.69	94.67	95.82
		Failure		95.74		96.96
Composite beam (chapter six)	Full connection beam	Service	89.60	92.04	96.63	97.84
		Failure		93.76		99.60
	Partial connection beam	Service	88.54	91.75	95.00	96.80
		Failure		95.49		98.56

* Obtained mean value for the analyzed beams.

The developed probabilistic assessment algorithm is combined with nonlinear structural analysis software. This will permit to process the algorithm both in service and failure region. An advantage of this framework is that it works well with any commercial available software. However, the accuracy of obtained results depends on the chosen software. In this situation, the software ATENA® [23, 24] is chosen, since it provides robust and adequate models for the used materials (concrete and steel) and for material interfaces (in composite structures).

Therefore, the main advantages of this algorithm are: (1) it incorporates the majority of the uncertainties sources that might be identified in a structural assessment procedure; (2) it incorporates experimental data obtained from different sources; (3) it may be executed with any commercial available software. However, it presents the following drawbacks: (1) its use still requires a permanent accompaniment from the user; (2) for more complex structures the computational cost is still very high.

With this framework it will be possible to help operators in better characterizing the current state of assessed structures. This is important as it provides accurate information that might be used to support the maintenance strategies. Therefore, it might be a major step forward in the process of decision making of investment in maintenance and replacement strategies of civil engineer structures.

8.2. Suggestions for future work

A framework for the probabilistic assessment of civil engineer structures is developed and validated within this thesis. However, it is recognized that this research is not finished and different work fields were opened for future developments.

The cable length and its wrapping effects in displacement transducer measurement error are detailed studied in this thesis. Other effects which are common in monitoring systems, such as external magnetic fields, data acquisition equipment errors, etc., should be also analyzed in the future. The measurement errors of other sensors, such as inclinometers, strain gages, etc., may be also evaluated.

The developed framework was only tested with static data, both in service and in failure region. However, in several situations, the structure´s dynamic behavior is conditioning and dynamic data is obtained from monitoring systems. Therefore, it would be important to test the algorithm with a finite element model that includes dynamic analysis.

The developed framework does not include the component time. It is known that concrete parameters vary with time due to shrinkage, creep and hardening effects. Other phenomena's, as relaxation of pre-stress cables, etc., result in a time variant behavior of assessed structures. Therefore, it would be worthwhile to incorporate a time variant stress-strain law in the developed numerical model. The obtained experimental data would be then used to update such law parameters.

A drawback of this algorithm is still the computational cost, especially when applied to more complex structures. Therefore the use of artificial neural networks (ANN) may be useful to diminish this cost [195]. In order to use ANN it is important to run the numerical model a few times. An ANN can be thus calibrated with obtained results. The reliability of the ANN increases with the number of training data. Once trained, the ANN may be used within this framework, instead of the numerical model.

In the application of this framework to composite structures it was verified that the available information regarding the steel to concrete interface is scarce. Some of the parameters were thus estimated based in experience. This becomes relevant when randomness is considered

and the PDF parameters need to be estimated. Therefore, it is important to develop more laboratory tests to evaluate the interface parameters variation with geometry, connector type and used materials.

As previously identified, the developed framework requires a permanent accompaniment from the user. Therefore, it is important to make it more user friendly by converting the developed routines into an executable file. The objective is to introduce each parameter value in a friendly way and to run the executable file accordingly, obtaining the requested data in an informative way.

This framework will provide information regarding the assessed structure. This information is useful to support any answer on whether an intervention is, or not, required. However, as different kinds of maintenance actions exist, it is necessary to develop one other framework that supports operators in a decision regarding the best maintenance strategy. Such framework would be based in a time variant cost-benefit model, that incorporates the effect of each maintenance action in the structure behavior and also the maintenance actions and the doing nothing costs. An optimization algorithm may be used to identify the most suitable maintenance strategy.

References

[1] AASHTO LRFD (1994) Load and resistance factor bridge design specification, American association of state highways transportation officials, DC, Washington.

[2] AASHTO LRFR (2003) Manual for condition evaluation and load and resistance factor rating of highway bridges, American association of state highways transportation officials, DC, Washington.

[3] Aho, A. V., Kernigan, B. K.; Weinberger, P. J. (1988) The AWK Programming Language Reading, Addison-Wesley, 210 pp., ISBN 020107981X.

[4] Akaike, H. (1974) A new look at the statistical model identification, IEEE Transactions on Automatic Control, Vol.19 (6): 716–723, doi: 10.1109/TAC.1974.1100705.

[5] Akgul, F., Frangopol, D. M. (2004) Computational Platform for Predicting Lifetime System Reliability Profiles for Different Structure Types, Journal of computing in civil engineering, ASCE, Vol. 18 (2): pp. 92 – 104.

[6] Aktan, A. E., Farhey, D. N., Helmicki, A. J., Brown, D. L., Hunt, V. J., Lee, K. –L., Levi, A. (1998) Structural identification for condition assessment: Experimental arts, Journal of structural engineering, Vol. 123 (12): pp. 1674-1684.

[7] Aktan, A. E., Çatbas, N., Turer, A., Zhang, Z. (1998) Structural identification: Analytical aspects. Journal of structural engineering, Vol. 124 (7): pp. 817-829.

[8] An, L., Cederwall, K. (1996) Push-out tests on studs in high strength and normal strength concrete, Journal of Construction Steel Research, January, Vol. 36, N. 1, pp. 15-29.

[9] Bäck, T. (1996) Evolutionary Algorithms in Theory and Practice, Oxford University Press, New York.

[10] Baker, J. E. (1985) Adaptive selection methods for genetic algorithms, Proceedings of the 1st International Conference on Genetic Algorithms and their Applications, Lawrence Erlbaum Associates, Hillsdale, NJ: pp. 101-111.

[11] Banan, Mo. R., Banan Ma. R., Hjelmstad, K. D. (1994) Parameter estimation of structures from static response I: Computational aspects, Journal of structural engineering, ASCE, Vol. 120 (11): pp. 3243 – 3258.

[12] Banan, M. R., Hjelmstad, K. D. (1994) Parameter estimation of structures from static response, II: Numerical simulation studies, Journal of structural engineering, ASCE, Vol. 120 (11): pp. 3259 – 3283.

[13] Bartlett, F. M., MacGregor, J. G. (1996) Statistical analysis of compressive strength of concrete structures, ACI materials Journal, Vol. 93 (2): pp. 158 - 168.

[14] Bergmeister, K., Novák, D., Pukl, R., Cervenka, V. (2009) Structural assessment and reliability analysis for existing engineering structures, theoretical background, Structure and Infrastructure Engineering, Vol. 5 (4): pp. 267 – 275.

[15] Bernardo, J., Smith, A. (2004) Bayesian theory, Wiley, 611 pp., ISBN 0471924164.

[16] Beyer, H. -G., Schwefel, H. -P. (2002) Evolution strategies – A comprehensive introduction, Natural Computing, Vol. 1 (1): pp. 3-52.

[17] BE96-3942/R2 (1998) EurolightCon – Economic Design and Construction with Light Weight Aggregate Concrete, Project BE96-3942, Report BE96-3942/R2: LWAC Material Properties – State of the Art.

[18] Biggs, M. C. (1975) Constrained minimization using recursive quadratic programming, Towards global optimization, North-Holland: pp. 341-349.

[19] Bolstad, W. M. (2004) Introduction to Bayesian Statistics, John Wiley & Sons, 362 pp., ISBN 0471270202.

[20] Brennan, A., Kharroubi, S. A. (2007) Expected value of sample information for Weibull survival data, Health Economics, Vol. 16 (11): pp. 1205-1225.

[21] Cabral, P. (2004) Erros e incertezas nas medições, Relatório Técnico, IEP – Instituto Eletrónico Português, Laboratório de Metrologia e Ensaios e ISEP – Instituto Superior de Engenharia do Porto, Departamento de Física, 116 pp.

[22] Červenka, V., Červenka, J., Pukl, R. (2002) ATENA® – A tool for engineering analysis of fracture in concrete, Sädhanä, Vol. 27 (4): pp. 485–492.

[23] Červenka, V. (2002) Computer simulation of failure of concrete structures for practice, Proceedings of the 1st fib Congress Concrete Structures in 21st Century, pp.: 289-304, Osaka, Japan, ISBN: 90-5809-530-4.

[24] Červenka, V., Jendele, L., Červenka, J. (2009) ATENA® Program Documentation, Part 1: Theory, Prague, Czech Republic, May. http://www.cervenka.cz.

[25] Chapman, S. J. (2002) Matlab® Programming for Engineers, Brooks/Cole Thomson Learning Ed., 478 pp., ISBN 0534390560.

[26] Choi, B. -S. Scanlon, A. and Johnson, P. A. (2004) Monte Carlo Simulation of Immediate and Time-Dependent Deflections of Reinforced Concrete Beams and Slabs, ACI –Structural Journal, Vol. 101, No. 5: pp. 633 – 641.

[27] Chou, J. –H., Ghaboussi, J. (2001) Genetic algorithm in structural damage detection, Computers and structures, Vol. 79: pp. 1335-1353.

[28] Congdon, P. (2003) Applied Bayesian Modelling, John Wiley & Sons, 478 pp., ISBN 0471486957.

[29] Costa, L., Oliveira, P. (2001) Evolutionary Algorithms Approach to the Solution of Mixed Integer Non-Linear Programming Problems, Computers and Chemical Engineering, Vol. 25: pp. 257-266.

[30] Costa, L., Oliveira, P. (2003) An Adaptive Sharing Elitist Evolution Strategy for Multiobjective Optimization, Evolutionary Computation, MIT Press, Vol. 11 (4): pp. 417-438.

[31] Davis, L. (1991) Handbook of Genetic Algorithms, Van Nostrand Reinhold, New York.

[32] Dellacroce, R., Schieb, P. A., Stevens, B. (2011) Pension funds investment in infrastructure - A survey, International Futures Programme, OECD.

[33] Ditlevsen, O., Madsen, H. O. (1996) Structural Reliability Methods, John Wiley & Sons, 372 pp., ISBN 0471960861.

[34] Dorigo, M., Maniezzo, V., Colorni, A. (1991) Positive Feedback as a Search Strategy, Technical Report No. 91-016, Dipartimento di Elettronica, Politecnico di Milano, Italy.

[35] Dorigo, M. (1992) Optimization, Learning and Natural Algorithms (in Italian), PhD thesis, Dipartimento di Elettronica, Politecnico di Milano, Italy.

[36] Dorigo, M., Di Caro, G. (1999) The Ant Colony Optimization Meta-Heuristic, New Methods in Optimization, Corne. D., Dorigo, M. and Glover, F. Eds., McGraw-Hill.

[37] EA-4/02 (1999) Expression of the uncertainty of measurement in calibration, EA - Technical Report, 79 pp.

[38] EA-4/16 (2003) EA guidelines on the expression of uncertainty in quantitative testing, EA - Technical Report, 27 pp.

[39] Eberhart, R. C., Kennedy, J. (1995) A New Optimizer Using Particles Swarm Theory, Sixth International Symposium on Micro Machine and Human Science, Nagoya, Japan: pp. 39–43.

[40] Ellingwood, B. (2005) Risk-informed condition assessment of civil infrastructure: state of practice and research issues, Structure and Infrastructure Engineering, Vol. 1 (1): pp. 7 – 18.

[41] Enevoldsen, I. (2001) Experience with probabilistic-based assessment of bridges, Structural Engineering International (SEI), Vol. 11 (4): pp. 251 – 260.

[42] Engelund, S., Rackwitz, R. (1993) A benchmark study on importance sampling techniques in structural reliability, Structural Safety, Vol. 12 (4): pp. 255 - 276.

[43] Enright, M. P., Frangopol, D. M. (1999) Reliability-based condition assessment of deteriorating concrete bridges considering load redistribution, Structural Safety, Vol. 21 (2), pp. 159 – 195.

[44] EN 10025-1 (2004) Hot rolled products of structural steels. Part 1: General technical delivery conditions. European Committee for Standardisation (CEN).

[45] EN 10025-2 (2004) European structural steel standard: Grade designations, properties and nearest equivalents. European Committee for Standardisation (CEN).

[46] EN 1990 (2002) Eurocode 0: Basis of structural design, CEN, 87 pp., ISBN 0580401863.

[47] EN 1991-2 (2002) Eurocode 1: Actions on structures – Part 2: Traffic loads on bridges, July.

[48] EN 1992-1-1 (2004) Eurocode 2: Design of concrete structures – Part 1-1: general rules and rules for buildings.

[49] EN 1993-1-1 (2010) Eurocode 3: Design of steel structures, General rules and rules for buildings, Mars.

[50] EN 1994-1-1 (2004) Eurocode 4: Design of composite steel and concrete structures. General rules and rules for buildings. December.

[51] EuroLightCon (2000) Mechanical Properties of LWAC compared with both NWC and HSC, EuroLightCon - Economic Design and Construction with Lightweight Aggregate Concrete, Technical Report, 194 pp., ISBN 903760238X.

[52] Faber, M. H. (2000) Reliability based assessment of existing Structures, Progress in Structural Engineering and Materials, Vol. 2 (2): pp. 247 – 253.

[53] Faber, M. H. (2008) Risk and Safety in Civil, Surveying and Environmental Engineering, Lecture Notes at Swiss Federal Institute of Technology, 336 pp.

[54] Fajkus, M., Holický, M., Rozlívka L., Vorlíček M. (1999) Random Properties of Steel Elements Produced In Czech Republic, Proceedings of Eurosteel'99, Prague, Czech Republic.

[55] fib – federation international du béton (2003) Bulletin 22: Monitoring and Safety Evaluation of Existing Concrete Structures, State-of-Art Report, 297 pp., ISBN 2883940622.

[56] Fletcher, R. (1987) Practical Methods of Optimization, John Wiley and Sons.

[57] Franco, G. (2003) Application of evolutionary strategies to structural identification and damage detection, Doctoral of philosophy dissertation, Columbia University, Department of Civil Engineering and Engineering Mechanics.

[58] Franco, G., Betti, R. and Lus, H. (2004) Identification of structural systems using an evolutionary strategy, Journal of Engineering Mechanics, Vol. 130 (10): pp. 1125-1139.

[59] Frangopol, D. M., Liu, M. (2007) Maintenance and management of civil infrastructure based on condition, safety, optimization, and life-cycle cost, Structure and Infrastructure Engineering, Vol. 3 (1): pp. 29 – 41.

[60] Galjaraard, J. C., Walraven, J. C. (2000) Behaviour of shear connector devices for lightweight steel-concrete composite structures – results, observations and comparisons of static tests, Second International Symposium on Structural Lightweight Aggregate Concrete, Kristiansand, Norway.

[61] Galjaraard, J. C., Walraven, J. C. (2001) Static tests on various types of shear connectors for composite structures, International Symposium on Connections between Steel and Concrete, University of Stuttgard, Vol. 2, pp. 1313-1322.

[62] Gelfand, A. E., Hills, S. E., Racine–Poon, A., Smith, A. F. M. (1990) Illustration of Bayesian inference in normal data models using Gibbs sampling, Journal of the American Statistical Association, Vol. 85 (412): pp. 972 – 985.

[63] Gentile, C. (2006) Modal and structural identification of a R.C. arch bridge, Structural engineering and mechanics, Vol. 22 (1): pp. 53-70.

[64] Ghosn, M., Moses, F. (1998) Redundancy in highway bridge superstructures, NCHRP Report No. 406, Transportation research board, DC, Washington, 44 pp.

[65] Gill, P. E., Murray, W., Wright, M. H. (1981) Practical Optimization, London, Academic Press.

[66] Giuntini, R. E., Giuntini, M. E. (1993) Simulating a Weibull posterior using Bayes inference, Reliability and maintainability Symposium, Atlanta, USA.

[67] Goulet, J. - A., Kripakaran, P., Smith, I. F. C. (2009) Langesand Bridge in Lucerne, Results from Phase-I Static-Load Tests, Technical Report, EPFL – École Polytechnique et Fédérale de Lausanne, 37 pp.

[68] Goulet, J. - A., Kripakaran, P., Smith, I. F. C. (2009) Considering sensor characteristics during measurement-system design for structural system identification, ASCE computing conference, Austin.

[69] Goulet, J. - A., Kripakaran, P., Smith, I. F. C. (2009) Estimation of Modeling Errors in Structural System Identification, 4th International Conference on Structural Health Monitoring on Intelligent Infrastructure (SHMII-4), Zurich, Switzerland.

[70] Goulet, J. - A., Kripakaran, P. Smith, I. F. C. (2009) Structural identification to improve bridge management, IABSE Symposium, Bangkok, Thailand.

[71] Goulet, J. –A., Kripakaran, P., Smith, I. F. C. (2010) Multimodel structural performance monitoring, Journal of structural engineering, Vol. 136 (10): pp. 1309-1318.

[72] Goulet, J. -A., Smith, I. F. C. (2010) CMS4SI Structural Identification Approach for Interpreting Measurements, IABSE Symposium, Venice: pp. 224-225.

[73] Hajek, B. (1988) Cooling schedules for optimal annealing, Mathematics of Operations Research, Vol. 13 (4): pp. 563–571.

[74] Hajela, P., Soeiro, F. J. (1990) Recent developments in damage detection based on system identification methods, Structural optimization, Vol. 2: pp. 1-10.

[75] Han, S. P. (1977) A globally convergent method for nonlinear programming, Journal Optimization Theory and Applications, Vol. 22: pp. 297.

[76] Hastings, W. (1970) Monte Carlo sampling methods using Markov chains and their applications, Biometrika, Vol. 57 (1): pp. 97–109.

[77] Henriques, A. A. R. (1998) Aplicação de novos conceitos de segurança no dimensionamento do betão estrutural, Dissertação para Doutoramento em Engenharia Civil, Faculdade de Engenharia da Universidade do Porto, Fevereiro 1998, 522 pp.

[78] Hjelmstad, K. D., Shin, S. (1997) Damage detection and assessment of structures from static response, Journal of engineering mechanics, Vol. 123 (6): pp. 568-576.

[79] Holland, J. (1975) Adaptation in natural and artificial systems, University of Michigan Press, Michigan.

[80] Hosser, D., Klinzmann, C., Schnetgoke, R. (2008) A framework for reliability-based system assessment based on structural health monitoring, Structure and Infrastructure Engineering, Vol. 4 (4): pp. 271 – 285.

[81] Hu, X. (2002) Structural damage identification based on static dead load strain measurements, Doctoral of philosophy dissertation, University of Delaware, USA.

[82] Hu, X., Shenton, H. W. (2002) Structural damage identification using static dead load strain measurements, 15th ASCE engineering mechanics conference, Columbia University, New York, USA, 2-5 June.

[83] Hu, X., Shenton, H. W. (2006) Damage identification based on dead load redistribution: effect of measurement error, Journal of structural engineering, Vol. 132 (8): pp. 1264-1273.

[84] Huntington, D. E., Lyrintzis, C. S. (1998) Improvements to and limitations of Latin hypercube sampling, Probabilistic Engineering Mechanics, Vol. 13 (4): pp. 245-253.

[85] Iman, R. L., Conover, W. J. (1982) A distribution-free approach to inducing rank correlation among input variables, Communications in Statistics Simulation and Computation, Vol. 11 (3): pp. 311 - 334.

[86] ISO 13822 (2001) Bases for design of structures – Assessment of existing structures, 43 pp.

[87] Jacinto, L. A. C. (2011) Avaliação de segurança de pontes existentes: Abordagem probabilística Bayesiana, Dissertação para Doutoramento em Engenharia Civil, Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, 293 pp.

[88] Jaishi, B., Ren, W. (2005) Structural finite element model updating using ambient vibration test results. Journal of structural engineering, Vol. 131 (4): pp. 617-628.

[89] Jaishi, B., Kim, H. –J., Kim, M. K., Ren, W. –X., Lee, S. –H. (2007) Finite element model updating of concrete-filled steel tubular arch bridge under operational condition using modal flexibility, Mechanical Systems and Signal Processing, Vol. 21: pp. 2406-2426.

[90] JCGM 100:2008 (2008) Evaluation of measurement data — Guide to the expression of uncertainty in measurement, JCGM (GUM 1995 with minor corrections), Technical Report, 132 pp.

[91] JCGM 101:2008 (2008) Evaluation of measurement data — Supplement 1 to the “Guide to the expression of uncertainty in measurement” — Propagation of distributions using a Monte Carlo method, JCGM, Technical Report, 90 pp.

[92] JCGM 104:2009 (2009) Evaluation of measurement data — An introduction to the “Guide to the expression of uncertainty in measurement” and related documents, JCGM, Technical Report, 28 pp.

[93] JCSS - Joint Committee on Structural Safety (2001) Probabilistic Model Code, 12th Draft, Available at URL: http://www.jcss.ethz.ch, ISBN 978-3-909386-79-6.

[94] Kaminskiy, M. P., Krivtsov, V. V. (2005) A Simple Procedure for Bayesian Estimation of the Weibull Distribution, IEEE transaction on reliability, Vol. 54 (4): pp. 612 - 616.

[95] Kennedy, J., Eberhart, R. C. (1995) Particle Swarm Optimization, Proceedings of the IEEE International Conference on Neural Networks, Perth, Australia: pp. 1942–1948.

[96] Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P. (1983) Optimization by simulated annealing, Science, Vol. 220: pp. 671–679.

[97] Kiureghian, A., Ditlevsen, O. (2009) Aleatoric or Epistemic? Does it matter? Structural Safety, Vol. 31 (2): pp. 105 - 112.

[98] Koh, C. G., Chen, Y. F., Liaw, C. –Y. (2003) A hybrid computational strategy for identification of structural parameters, Computers and Structures, Vol. 81: pp. 107-117.

[99] LABEST/FEUP – NewMensus (2011) Instrumentação e Observação do Comportamento da Ponte sobre Rio Sousa II durante o Ensaio de Carga, Technical Reeport, 35 pp.

[100] Lagaros, N. D., Papadrakakis, M., Kokossalakis, G. (2002) Structural optimization using evolutionary algorithms, Computers and structures, Vol. 80 (7-8): pp. 571–589.

[101] Laurdisen, J., Jensen, J. S., Enevoldsen, I. B. (2007) Bridge owner's benefits from probabilistic approaches, Structure and Infrastructure Engineering, Vol. 3 (4): pp. 281 – 302.

[102] Leming, M. L. (1988) Properties of High Strength Concrete, An Investigation of High Strength Concrete Characteristics Using Materials in North Carolina, North Carolina State University, July 1988, Raleigh, North Carolina.

[103] Levin, R. I., Lieven, N. A. J. (1998) Dynamic finite element model updating using simulated annealing and genetic algorithms, Mechanical Systems and Signal Processing, Vol. 12 (1): pp. 91–120.

[104] LisConcebe (2009) A43 – Gondomar / Aguiar de Sousa (IC24), Projeto de Execução, Obras de Arte Especiais, Ponte sobre Rio Sousa II (Alteração da Metodologia Construtiva), Memória Descritiva e Justificativa, Relatório Técnico, 13 pp.

[105] LisConcebe (2009) A43 – Gondomar / Aguiar de Sousa (IC24), Projeto de Execução, Obras de Arte Especiais, Ponte sobre Rio Sousa II (Alteração da Metodologia Construtiva), Projeto de Execução, Memória de Cálculo, Relatório Técnico, 89 pp.

[106] LisConcebe (2009) A43 – Gondomar / Aguiar de Sousa (IC24), Projeto de Execução, Obras de Arte Especiais, Ponte sobre Rio Sousa II (Alteração da Metodologia Construtiva), Projeto de Execução, Clausulas Técnicas Complementares, Relatório Técnico, 39 pp.

[107] Liu, S. C., Yao, J. T. P. (1978) Structural identification concept. Journal of structural division, Vol. 104 (12): pp. 1845 – 1858.

[108] Liu, W., Neuenhoffer, A., Ghosn, M., Moses, F. (2001) Redundancy in highway bridge substructures, NCHRP Report No. 458, Transportation research board, DC, Washington, 44 pp.

[109] LNEC E 397 (1993) Betão – Ensaio de módulo de elasticidade.

[110] LNEC E 456 (2008) Varões de aço A500 ER para armaduras de betão armado. Características, ensaios e marcação.

[111] Lunn, D. J., Thomas, A., Best, N. (2000) WinBugs® – A Bayesian modeling framework: Concepts, structure, and extensibility, Statistics and Computing, Vol. 10 (4): pp. 325–337.

[112] Maeck, J., Peeters, B., De Roeck, G. (2001) Damage identification on the z24 bridge using vibration monitoring, Smart materials and structures, Vol. 10: pp. 512-517.

[113] Maljaars, J., Steenbergen, R., Abspoel, L., Kolstein, H. (2012) Safety assessment of existing highway bridges and viaducts, Structural Engineering International, Vol. 22 (1): pp. 112 – 120.

[114] Matos, J. C., Valente, M. I. B., Cruz, P. J. S. (2009) Avaliação de incertezas no comportamento até à rotura de vigas de betão armado, ASCP09, 1º Congresso de Segurança e Conservação de Pontes, ISBN 978-989-20-1559-0, pp.: 5-12, Lisbon, Portugal, 1-3 July.

[115] Matos, J. C., Valente, M. I. B., Cruz, P. J. S. (2010) Análise experimental do comportamento até à rotura de vigas de betão armado, 8º Congresso Nacional de Mecânica Experimental, Guimarães, Portugal, 21 – 23 April.

[116] Matos, J. C., Valente, M. I. B., Cruz, P. J. S. (2010) Uncertainty evaluation of reinforced concrete structures behavior, The fifth international conference on bridge maintenance, safety and management – IABMAS 2010, Philadelphia, USA, 11–15 July.

[117] Matos, J. C., Valente, M. I. B., Cruz, P. J. S. (2010) Nonlinear probabilistic analysis of reinforced concrete structures, 34th IABSE Symposium – Large structures and infrastructures for environmentally constrained and urbanized areas, Venice, Italy, 22–24 September.

[118] Matos, J. C., Valente, M. I. B., Neves, L. C., Cruz, P. J. S. (2011) Avaliação probabilística do comportamento até à rotura de estruturas: aplicação a vigas mistas, VIII Congresso de Construção Metálica e Mista, Guimarães – Portugal, 24 – 25 November.

[119] Matos, J. C., Cruz, P. J. S., Valente, M. I. B., Neves, L. C. (2012) An advanced reliability procedure for lifetime assessment of Structures: Application to reinforced concrete beams, fib 2012 Symposium, Stockholm – Sweden, 11-14 June.

[120] Matos, J. C., Valente, M. I. B., Cruz, P. J. S., Neves, L. C. (2012) Uncertainty evaluation of the behavior of a composite beam, IABMAS 6th International Conference on Bridge Maintenance, Safety and Management, Lake Maggiore – Italy, 8 – 12 July.

[121] Matos, J. C., Valente, M. I. B., Cruz, P. J. S., Neves, L. C. (2012) A probabilistic assessment methodology for life cycle analysis of Structures, 18th IABSE Congress on Innovative Infrastructures, Seoul – South Korea, 19-21 September.

[122] Matos, J. C., Valente, M. I. B., Cruz, P. J. S., Neves, L. C. (2012) An advanced probabilistic updating algorithm for life cycle analysis of civil structures, IALCCE - Third international Symposium on Life-Cycle Civil Engineering, Wien – Austria, 3-6 October.

[123] Melchers, R. E. (1999) Structural reliability analysis and prediction, John Wiley & Sons, 437 pp., ISBN 0471987719.

[124] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., Teller, E. (1953) Equations of state calculations by fast computing machines, Journal of Chemical Physics, Vol. 21: pp. 1087-1091.

[125] Michaeliwcz, Z. (1992) Genetic Algorithms + Data Structures = Evolution Programs, Springer Verlag, New-York.

[126] Mildenhall, S. J. (2005) Correlation and Aggregate Loss Distributions With An Emphasis On The Iman-Conover Method, CAS Working Party on Correlation, Casualty Actuarial Society Forum, 100 pp.

[127] Miranda, T. (2007) Geomechanical parameters evaluation in underground structures: artificial intelligence, Bayesian probabilities and inverse methods, PhD Thesis, Minho University, 319 pp.

[128] Mirza, S. A., MacGregor, J. G. (1979) Variation in dimensions of reinforced concrete members, ASCE Journal of the structural division, Vol. 105 (4): pp. 751 - 766.

[129] Morais, V., Vieira, C. (2006) Matlab® 7 & 6 Curso Completo, FCA Ed., 644 pp., ISBN 9789727223541.

[130] Mori, Y., Nonaka, M. (2001) LRFD for assessment of deteriorating existing structures, Structural Safety, Vol. 23 (4): pp. 297 – 313.

[131] Nelson Bolzenschweiß-Technik (2003) European Technical Approval for NELSON-headed studs, Deutsches Institut für Bautechnik, ETA 03/0041 and ETA 03/0042 13.

[132] Nocedal, J., Wright, S. J. (1999) Numerical Optimization. Springer, New York, USA.

[133] Novák, D., Pukl, R., Strauss, A. (2011) Reliability/risk assessment of concrete structures: Methodology, software and case study, First Middle East Conference on Smart Monitoring, Assessment and Rehabilitation of Civil Structures, Dubai, UAE.

[134] Nowak, A. S., Collins, K. R. (2000) Reliability of Structures, McGraw-Hill, 338 pp., ISBN 0071163549.

[135] Nowak, A. S., Szersen, M. M., Szeliga E. K., Szwed, A., Podhorecki, P. J. (2008) Reliability-based calibration for structural concrete, Phase 3, Portland Cement Association, PCA R&D Serial No. 2849, Technical Report, 115 pp.

[136] NP EN 12390-3 (2003) Ensaios do betão endurecido, Parte 3: Resistência à compressão dos provetes de ensaio.

[137] NP EN 13747 (2005) Produtos prefabricados de betão: Pré-lajes para pavimentos.

[138] NP EN 206-1 (2007) Betão. Parte 1: Especificação, desempenho, produção e conformidade.

[139] NP ENV 10002-1 (2001) Ensaio de tração de materiais metálicos.

[140] OECD (2007) Infrastructure to 2030, Volume 2: Mapping policy for electricity, water and transport. ISBN 978-92-64-03131-9.

[141] OECD (2008) Transport Infrastructure Investment: Options for Efficiency. ISBN 978-92-821-0155-1.

[142] OECD (2008) Policy brief - infrastructure to 2030. OECD Observer.

[143] Oh, B. H., Jung, B. S., Cha, S. W. (1999) Damage assessment of structures using static and dynamic test data, Transactions of the 15th International Conference on Structural Mechanics in Reactor Technology (SMiRT-15), Seoul, Korea, 15-20 August: pp. 179-186.

[144] Olsson, A., Sandberg, G., Dahlblom, O. (2003) On Latin hypercube sampling for structural reliability analysis, Structural Safety, Vol. 25 (1): pp. 47 – 68.

[145] Paulino, C. D., Turkman, M. A. A., Murteira, B. (2003) Estatística Bayesiana, Fundação Calouste Gulbenkian Ed., 446 pp., ISBN 9723110431.

[146] Powell, M. J. D. (1970) A Fortran subroutine for solving systems of nonlinear algebraic equations, Numerical methods for Nonlinear Algebraic equations.

[147] Powell, M. J. D. (1978) The convergence of variable metric methods for nonlinearly constrained optimization calculations, Nonlinear programming, Vol. 3, Academic Press.

[148] Pukl, R., Jansta, M., Červenka, J., Vorechovsky, M., Novak, D. and Rusina, R. (2006) Spatial variability of material properties in advanced nonlinear computer simulation, Proceedings of the EURO-C Conference, Mayrhofen, Austria: pp. 891 – 896.

[149] Rafiq, M. I., Onoufriou, T., Chryssanthopoulos, M. (2006) Sensitivity of uncertainties in performance prediction of deteriorating concrete structures, Structure and Infrastructure Engineering, Vol. 2 (2): pp. 117 – 130.

[150] Rao, S. S. (1996) Engineering optimization – Theory and practice, John Wiley and Sons, New York.

[151] Raphael, B., Smith, I. F. C. (2003) A direct stochastic algorithm for global search, Journal of Applied Mathematics and Computation, Vol. 146 (2-3): pp. 729-758.

[152] Ravindram, S., Kripakaran, P., Smith, I. F. C. (2007) Evaluating reliability of multiple-model system identification, 14th EG-ICE workshop, Maribor, Slovenia.

[153] Rechenberg, I. (1994) Evolutionsstrategie, Frommann-Holzboog, Stuttgart.

[154] Restivo, M. T., Sousa, C. (2008) Measurement Uncertainties in the Experimental Field, Sensors and Transducers Journal, Vol. 95 (8): pp. 1-12.

[155] RILEM TC50-FMC (1985) Determination of fracture energy of mortar and concrete by means of three point bend tests on notched beams, Materials and Structures, Vol. 18, N. 106, July-August, pp. 285-290.

[156] Robert-Nicoud, Y., Raphael, B., Smith, I. (2000) Decision support through multiple models and probabilistic search, Proceedings of construction information technology, Iceland building research institute, Reykjavic. Iceland: pp. 765-779.

[157] Robert-Nicoud, Y., Raphael, B., Burdet, O., Smith, I. F. C. (2005) System identification through model composition and stochastic search, Journal of computing in civil engineering, Vol. 19 (3): pp. 239-247.

[158] Robert-Nicoud, Y., Raphael, B., Burdet, O., Smith, I. F. C. (2005) Model identification of bridges using measurement data, Computer-aided civil and infrastructure engineering, Vol. 20: pp. 118-131.

[159] Roik, E. H. K., Hanswille, G., Lanna, A. C. - O. (1989) Report on Eurocode 4. Clause 6.3.2.: Stud Connectors, Harmonisation of the European Construction Codes, Eurocode 3, 4 and 8 / part 3, Report EC4/8/88, 92 pp.

[160] Ryall, M. J. (2001) Bridge management, Butterworth-Heinemann, 450 pp., ISBN 075065077X.

[161] Sanayei, M., Imbaro, G. R., McClain, J. A. S., Brown, L. C. (1997) Structural model updating using experimental static measurements, Journal of structural engineering, Vol. 123 (6): pp. 792-798.

[162] Sanayei, M., McClain, J. A. S., Wadia-Fascetti, S., Santini, E. M. (1999) Parameter estimation incorporating modal data and boundary conditions, Journal of structural engineering, Vol. 125 (9): pp. 1048-1055.

[163] Sanayei, M., Fascetti, S. W., Arya, B., Santini, E. M. (2001) Significance of Modeling Error in Structural Parameter Estimation, Computer-Aided Civil and Infrastructure Engineering, Vol. 16 (1): pp. 12–27.

[164] Santa, U., Bergmeister, K., Strauss, A. (2004) Discussion of stochastic models in structural engineering, fib Symposium 2004, Avignon, France.

[165] Schlune, H., Plos, M. (2008) Bridge Assessment and Maintenance based on Finite Element Structural Models and Field Measurements: State of the art review, Report, Chalmers University of Technology, Goteborg, Sweden.

[166] Schneider, J. (1997) Vol. 5: Introduction to safety and reliability of structures, Structural Engineering Documents (SED), International Association for Bridge and Structural Engineering (IABSE), ISBN 9783857480935.

[167] Schwefel, H. -P. (1995) Evolution and Optimum Seeking, New York, Wiley.

[168] Shenton, H. W., Hu, X. (2006) Damage identification based on dead load redistribution: methodology, Journal of structural engineering, Vol. 132 (8): pp. 1254-1263.

[169] Smith, I. F. C., Saitta, S. (2008) Improving knowledge of structural system behavior through multiple models, Journal of structural engineering, Vol. 134: pp. 553-561.

[170] Sobriño, J. A. (1993) Evaluación del comportamiento funcional y de la seguridad estructural de puentes existentes de hormigón armado y pretensado, PhD thesis, Technical University of Catalonia, Department of Construction Engineering, 140 pp.

[171] SocoMetal (2010) Douro Litoral – Vigas Metálicas, Relatório de Qualidade.

[172] Spellucci, P. (1998) A new technique for inconsistent QP problems in the SQP method, Journal of Mathematical Methods of Operations Research, Vol. 47 (3): pp. 355–400.

[173] Stein, M. (1987) Large sample properties of simulations using Latin Hypercube Sampling, Technometrics, Vol. 29 (2): pp. 143 - 151.

[174] Strauss, A., Bergmeister, K., Santa, U. (2004) Reliability based assessment procedure for concrete structures, fib Symposium 2004, Avignon, France.

[175] Strauss, A., Frangopol, S. M., Kim, S. (2008) Use of monitoring extreme data for the performance prediction of structures: Bayesian updating, Engineering Structures, Vol. 30 (12): pp. 3654 – 3666.

[176] Strauss, A., Bergmeister, K., Hoffmann, S., Pukl, R., Novák, D. (2008) Advanced Life-Cycle Analysis of Existing Concrete Bridges, Journal of materials in civil engineering, ASCE; Vol. 20 (1), pp. 9 – 19.

[177] Strauss, A., Hoffmann, S., Wendner, R., Bergmeister, K., (2009) Structural assessment and reliability analysis for existing engineering structures, applications for real structures, Structure and Infrastructure Engineering, Vol. 5 (4): pp. 277 – 286.

[178] Suykens, J. A. K., Vandewalle, J., De Moor, B. (2001) Intelligence and cooperative search by Coupled Local Minimizers, International Journal of Bifurcation and Chaos, Vol. 11 (8): pp. 2133–2144.

[179] Suykens, J. A. K., Vandewalle, J. (2002) Coupled Local Minimizers: Alternative formulations and extensions, 2002 World Congress on Computational Intelligence - International Joint Conference on Neural Networks IJCNN 2002, Honolulu, USA: pp. 2039–2043.

[180] Tabsh, S. W., Nowak, A. S. (1991) Reliability of highway girder bridges, ASCE Journal of structural engineering, Vol. 117 (8): pp. 2372 - 2388.

[181] Teigen, J. G., Frangopol, D. M., Sture, S. and Felippa, C. A. (1991) Probabilistic FEM for Nonlinear Concrete Structures I: Theory, ASCE – Journal of Structural Engineering, Vol. 117, No. 9: pp. 2674 – 2689.

[182] Teigen, J. G., Frangopol, D. M., Sture, S. and Felippa, C. A. (1991) Probabilistic FEM for Nonlinear Concrete Structures II: Applications, ASCE – Journal of Structural Engineering, Vol. 117, No. 9: pp. 2690 – 2707.

[183] Teixeira Duarte (2010) Douro Litoral – Lajes Tabuleiro Rio Sousa, Relatório de Qualidade.

[184] Teughels, A. (2003) Inverse modeling of civil engineering structures based on operational modal data, Doctoral of philosophy dissertation, Leuven Catholic University.

[185] Tierney, L. (1994) Markov chains for exploring posterior distributions, Annals of Statistics, Vol. 22 (4): pp. 1701 – 1728.

[186] Tone, K. (1983) Revisions of constraint approximations in the successive QP method for nonlinear programming problems, Journal of Mathematical Programming, Vol. 26 (2): pp. 144–152.

[187] Törn, A., Zilinskas, A. (1989) Global Optimization, Lecture Notes in Computer Science, Nº. 350, Springer-Verlag, Berlin.

[188] Valente, M. I. B. (2007) Experimental studies on shear connection systems in steel and lightweight concrete composite bridges, PhD Thesis, Minho University, 411 pp.

[189] Valente, M. I. B., Lage, R., Matos, J. C. (2011) Modelação não linear de vigas mistas em aço e betão, VIII Congresso de Construção Metálica e Mista, Guimarães – Portugal, 24–25 November 2011.

[190] Valum, R., Nilsskog, J. E. (1999) Production and Quality Control of High Performance Lightweight Concrete for the Raftsundet Bridge, 5th International Symposium on Utilization of High Strength/High Performance Concrete, Sandefjord, Norway, 20-24 June, 1999.

[191] Van Laarhoven, P. J. M., Aarts, E. H. L. (1987) Simulated Annealing: Theory and Applications, Kluwer Academic Publishers, Dordrecht, The Netherlands.

[192] Vorechovský, M., Novák, D. (2002) Correlated random variables in probabilistic simulation, 4th PhD Symposium in Civil Engineering, Munich, Germany.

[193] Vorechovský, M., Novák, D. (2003) Statistical correlation in stratified sampling, Proceedings of ICASP 9, San Francisco, USA.

[194] Vorechovský, M., Novák, D. (2009) Correlation controls in small-sample Monte Carlo type simulations I: A simulated annealing approach, Probabilistic Engineering Mechanics, Vol. 24 (3): pp. 452 – 462.

[195] Whitely, D., Hanson, T. (1989) Optimizing neural networks using faster, more accurate genetic search, Proceedings of the 3rd International Conference on Genetic Algorithms and their Applications, George Mason University: pp. 370-374.

[196] Wisniewski, D. F. (2007) Safety format for the assessment of concrete bridges with special focus on precast concrete, Doctoral Thesis, Department of Civil Engineering, School of Engineering, University of Minho, 376 pp.

[197] Wisniewski, D. F., Casas, J. R., Ghosn, M. (2009) Simplified probabilistic non-linear assessment of existing railway bridges, Structure and Infrastructure Engineering, Vol. 5 (6): pp. 439 – 453.

[198] Xia, Y., Hao, H. (2001) A genetic algorithm for structural damage detection based on vibration data, Proceedings of IMAC XIX: A conference on structural dynamics, Kissimmee, Florida: pp. 1381–1387.

[199] Zhao, L., Shenton, H. W. (2002) Direct identification of damage in beam structures using dead load measurements, 15th ASCE Engineering Mechanics Conference, Columbia University, New York, USA, 2-5 June.

[200] Zona, A., Barbato, M., Dall'Asta, A., Dezi, L. (2010) Probabilistic analysis for design assessment of continuous steel-concrete composite girders, Journal of Constructional Steel Research, Vol. 66 (7): pp. 897 – 905.

Appendix A: Model Identification Routines

According to what was previously indicated at chapter four, some Matlab® routines [25, 129] were developed within the model identification procedure. These routines are linked according to Figure A.1, organization chart. The routine *es.m* refers to the evolutionary strategies algorithm [29]. This algorithm was already implemented by Costa and Oliveira [29]. Some of the developed routines are provided further. Other routines, such as the *parameter.m* and the *constraint_value.m*, should be adapted to each situation.

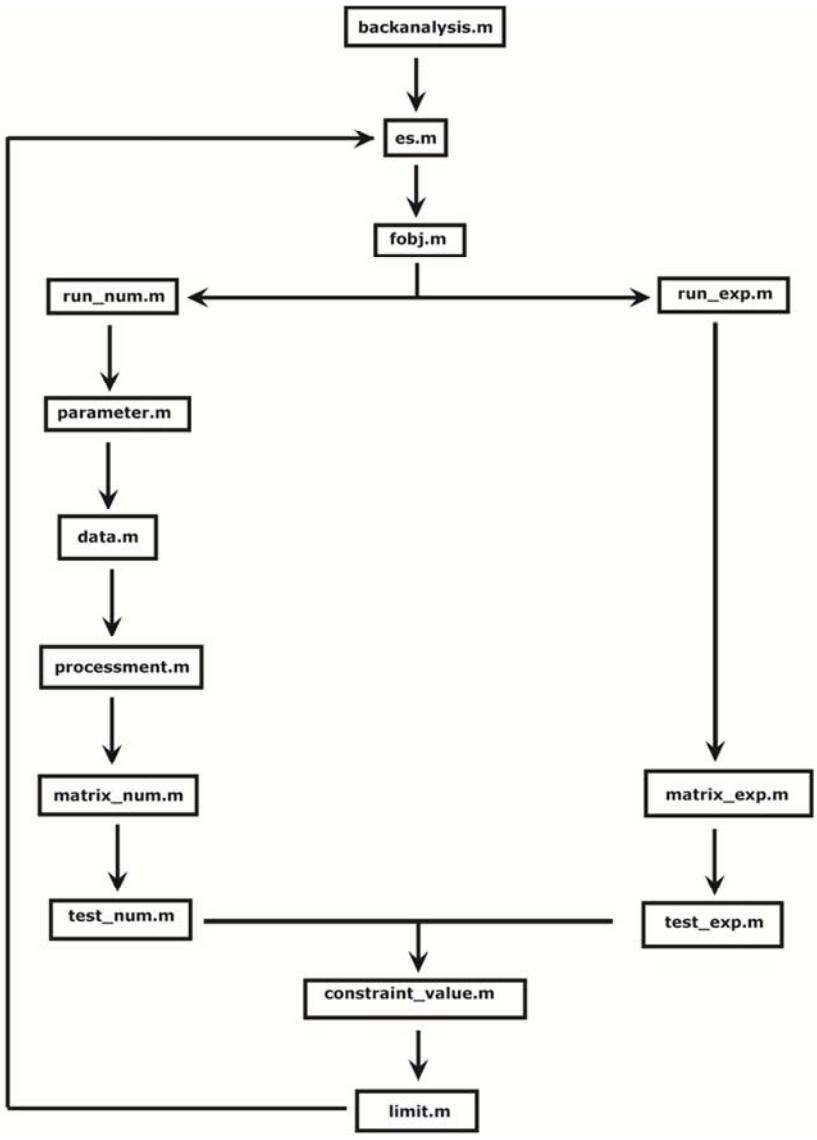


Figure A.1. Organization chart.

(1) *backanalysis.m*

```
k=(…); % 1 - ES (Evolutionary Strategies) {,}; 2 - ES (Evolutionary Strategies) {+}
E=(…); % Tolerance Criterion
LB=[…]; % Lower Bound (Parameters)
UB=[…]; % Upper Bound (Parameters)
x_0=[…]; % Initial Value (Parameters)
B=[…]; % Other parameters (fixed values)
nvars=size(…); % Number of parameters
if k==1
Problem.ObjFunction='fobj';
Problem.LB=LB;
Problem.UB=UB;
Options.MaxObj=(…);
Options.MaxGen=(…);
Options.Mu=(…);
Options.Rho=(…);
Options.Lambda=(…);
Options.Selection=',';
Options.CPTolerance=E;
Options.CPGenTest=(…);
Options.Verbosity=(…);
InitialPopulation(1).x=x_0;
elseif k==2
Problem.ObjFunction='fobj';
Problem.LB=LB;
Problem.UB=UB;
Options.MaxObj=(…);
Options.MaxGen=(…);
Options.Mu=(…);
Options.Rho=(…);
Options.Lambda=(…);
Options.Selection='+';
Options.CPTolerance=E;
Options.CPGenTest=(…);
Options.Verbosity=(…);
InitialPopulation(1).x=x_0;
end
[x,fval,RunData]=es(Problem,[…],Options,B,nvars);
```

(2) *fobj.m*

```
function f = fobj(x,B,nvars)
A = convert(x,B,nvars); % Compute A from x (variable) and B (fixed)
[K,mexp,limd,lim,pd,k,C,R] = run_num(A);
```

```
P = run_exp(mexp);
BD = test_num(K,pd);
BE = test_exp(P,mexp,pd);
v = limit(BD,BE,mexp,k);
for u=1:mexp
% Compute the fitness function value
difd=0;
s=size(BD);
s=s(1);
m=size(BE{u});
m=m(1);
ad1=min(s,m);
ad=0;
% Verify if it is the same comparing point
if BD(1,1)==BE{u}(1,1)
% Verify limit displacement
if BD(1,1)<=limd || BE{u}(1,1)<=limd
% Check if i tis lower than maximum load
if BD (1,2)<=v(u) && BE{u}(1,2)<=v(u)
ad=ad+1;
difd=difd+sqrt((BD (1,2)-BE{u}(1,2))^2);
end
end
end
for i=1:ad1-1
% Verify if it is the same comparing point
if BD(i+1,1)==BE{u}(i+1,1)
% Verify limit displacement
if BD(i+1,1)<=limd || BE{u}(i+1,1)<=limd
% Check if i tis lower than maximum load
if BD(i+1,2)<=v(u) && BE{u}(i+1,2)<=v(u)
ad=ad+1;
difd=difd+sqrt((BD(i+1,2)-BE{u}(i+1,2))^2);
end
end
end
end
bd(u,1)=max(BE{u}(:,2));
end
sum=0;
for i=1:mexp
f(i,1)=difd(i,1)/(bd(i,1)*ad(i,1))*R;
sum=sum+f(i,1)^2;
end
f=sqrt(sum)*C;
```

(3) *es.m* [29]

```
function [x, fx, RunData]=es(Problem, InitialPopulation, Options, varargin)
% Check parameters consistence
if nargin < 1
error('ES:AtLeastOneInput','ES requests at least one input (Problem definition).');
end
% If parameters are missing just define them as empty
if nargin < 3, Options=[];
if nargin < 2, InitialPopulation = [];
end
end
% Problem should be a structure
if ~isstruct(Problem)
error('ES:StructProblem', 'First parameter must be a struct.');
end
% Do some check-up on the user provided parameters and data
if ~isfield(Problem,'ObjFunction') || isempty(Problem.ObjFunction)
error('ES:ObjMissing', 'Objective function name is missing.');
end
% and simple bound for evolution strategies
if ~isnumeric(Problem.LB) || ~isnumeric(Problem.UB) || isempty(Problem.LB) || isempty(Problem.UB)
error('ES:Bounds', 'Population relies on finite bounds on all variables.');
end
% Bound arrays must have the same size
if length(Problem.LB)~=length(Problem.UB)
error('ES:BoundsSize', 'Lower bound and upper bound arrays length mismatch.');
end
% Compute the number of variables
if (~isfield(Problem, 'Variables') || isempty(Problem.Variables))
Problem.Variables=length(Problem.LB);
end
% Check for computed number of variables
if Problem.Variables<0 || Problem.Variables>length(Problem.LB)
error('ES:VariablesNumber', 'Number of variables do not agree with bound constraints');
end
% Initialize options – Local options
MaxGenerations=GetOption('MaxGen',Options,DefaultOpt);
MaxEvals=GetOption('MaxObj',Options,DefaultOpt);
Mu=GetOption('Mu', Options, DefaultOpt);
Rho=GetOption('Rho',Options,DefaultOpt);
Lambda=GetOption('Lambda',Options,DefaultOpt);
SelectType=GetOption('SelectType',Options,DefaultOpt);
RecombIndType=GetOption('RecombIndType',Options,DefaultOpt);
RecombParType=GetOption('RecombParType',Options,DefaultOpt);
```

```
RecombScope=GetOption('RecombScope',Options,DefaultOpt);
SAdaptType=GetOption('SAdaptType',Options,DefaultOpt);
% Initialize learning coefficients for self-adaptation rules
Eta1=1/sqrt(2*Problem.Variables);
Eta2=1/sqrt(2*sqrt(Problem.Variables));
% Initialize coefficient for rotation in self-adaptation rules
Beta=5/180*pi;
% Inferior bound for step sizes
Tol=1e-10;
% Initialize options – Global options
Problem.Verbose=GetOption('Verbosity', Options, DefaultOpt);
Problem.Tolerance=GetOption('CPTolerance', Options, DefaultOpt);
Problem.GenTest=GetOption('CPGenTest', Options, DefaultOpt);
% Number of objective function calls
Problem.Stats.ObjFunCounter=0;
% Generate initial population
[Problem,ParentPop]=InitPopulation(Problem, InitialPopulation, Mu, varargin{:});
% Sort population
temp=[ParentPop.x ParentPop.f ParentPop.std ParentPop.rot];
temp=sortrows(temp,Problem.Variables+1);
ParentPop.x=temp(:,1:Problem.Variables);
ParentPop.f=temp(:,Problem.Variables+1);
ParentPop.std=temp(:,Problem.Variables+2:2*Problem.Variables+1);
ParentPop.rot=temp(:,2*Problem.Variables+2:2*Problem.Variables+1+Problem.Variables*(Problem.Variables-1)/2);
if Problem.Verbose
disp('ES is alive... ');
% Initialize counters
Problem.Stats.GenCounter=0;
% Initialize statistics structures
Problem.Stats.Best(Problem.Stats.GenCounter+1)=ParentPop.f(1);
Problem.Stats.Worst(Problem.Stats.GenCounter+1)=ParentPop.f(Mu);
Problem.Stats.Mean(Problem.Stats.GenCounter+1)=mean(ParentPop.f);
Problem.Stats.Std(Problem.Stats.GenCounter+1)=std(ParentPop.f);
% Main cycle of the evolution strategy
% Stop if the maximum number of iterations or objective function evaluations is reached.
while(Problem.Stats.GenCounter<MaxGenerations && Problem.Stats.ObjFunCounter<MaxEvals)
% Stop if the improvement is inferior to the Tolerance in the last generations
if Problem.Stats.GenCounter>0 && ~mod(Problem.Stats.GenCounter,ceil(Problem.GenTest*MaxGenerations)) &&
abs(Problem.Stats.Best(Problem.Stats.GenCounter+1)-Problem.Stats.Best(Problem.Stats.GenCounter+1-
ceil(Problem.GenTest*MaxGenerations))) < Problem.Tolerance
disp('Stopping due to objective function improvement inferior to CPTolerance in the last CPGenTest generations');
break;
end
% Increment generation counter.
Problem.Stats.GenCounter=Problem.Stats.GenCounter+1;
```

```
% Define the recombination scope
switch RecombScope
case 'H'
list=create_group(Mu,Rho,0);
case 'N'
list=create_group(Mu,Rho,1);
otherwise
error('ES:Invalid recombination scope option', 'Define: H for Hemaphrodite; N for Not hemaphrodite.');
end
% Recombination
switch RecombParType
case 'D'
% Discrete Recombination for search parameters
for i=1:Mu
for j=1:Problem.Variables
i1=randint(1,1,[1 Rho]);
RecPop.std(i,j)=ParentPop.std(list(i1),j);
end
if Problem.Variables*(Problem.Variables-1)/2>0
for j=1:Problem.Variables*(Problem.Variables-1)/2
i2=randint(1,1,[1 Rho]);
RecPop.rot(i,j)=ParentPop.rot(list(i2),j);
end
end
end
case 'I'
% Intermediate Recombination for search parameters
for i=1:Mu
for j=1:Problem.Variables
for k=1:Rho
RecPop.std(i,j)=sum(ParentPop.std(list(k),j),1);
end
RecPop.std(i,j)=RecPop.std(i,j)/Rho;
end
for j=1:Problem.Variables*(Problem.Variables-1)/2
for k=1:Rho
RecPop.rot(i,j)=sum(ParentPop.rot(list(k),j),1);
end
RecPop.rot(i,j)=RecPop.rot(i,j)/Rho;
end
end
otherwise
error('ES:Invalid parameters recombination option', 'Define: D for Discrete recombination; I for Intermediate recombination.');
end
switch RecombIndType
```

```
case 'D'
% Discrete Recombination for individuals
for i=1:Mu
for j=1:Problem.Variables
i1=randint(1,1,[1 Rho]);
RecPop.x(i,j)=ParentPop.x(list(i1),j);
end
end
case 'I'
% Intermediate Recombination for individuals
for i=1:Mu
for j=1:Problem.Variables
for k=1:Rho
RecPop.x(i,j)=sum(ParentPop.x(list(k),j),1);
end
RecPop.x(i,j)=RecPop.x(i,j)/Rho;
end
end
otherwise
error('ES:Invalid individual recombination option', 'Define: D for Discrete recombination; I for Intermediate recombination.');
end
% Gaussian mutation for parameters
for i=1:Mu
lognormal=Eta1*randn(1);
for j=1:Problem.Variables
prev_std=RecPop.std(i,j);
switch SAdaptType
case 'I'
% Isotropic rule
RecPop.std(i,j)=RecPop.std(i,j)*exp(lognormal);
case {'N','R'}
% Nonisotropic rule
RecPop.std(i,j)=RecPop.std(i,j)*exp(lognormal + Eta2*randn(1));
otherwise
error('ES:Invalid auto-adaptation option', 'Define: I for Isotropic; N for Nonisotropic; R for Nonisotropic with rotation.');
end
% Test if inferior bound for step sizes is reached
if (RecPop.std(i,j)<Tol || (max(abs(RecPop.x(i,:)))~=0 && RecPop.std(i,j)/max(abs(RecPop.x(i,:)))<Tol))
RectPop.std(i,j)=prev_std;
end
end
% Rotations mutation
for j=1:Problem.Variables*(Problem.Variables-1)/2
RecPop.rot(i,j)=RecPop.rot(i,j)+Beta*randn(1);
% Reflect rotations for -pi to pi
```

```
if abs(RecPop.rot(i,j))>pi
RecPop.rot(i,j)=RecPop.rot(i,j)-2*pi*sign(RecPop.rot(i,j));
end
end
end
% Gaussian mutation of individuals
for i=1:Lambda
if SAdaptType~='R'
% Isotropic or nonisotropic
OffPop.x(i,:)=RecPop.x(mod(i-1,Mu)+1,:)+RecPop.std(mod(i-1,Mu)+1,:).*randn(1,Problem.Variables);
OffPop.std(i,:)=RecPop.std(mod(i-1,Mu)+1,:);
else
% Nonisotropic with rotation
pos=1;
RR=eye(Problem.Variables);
for (k1=1:Problem.Variables-1)
for (k2=k1+1:Problem.Variables)
R=eye(Problem.Variables);
R(k1,k1)=cos(RecPop.rot(mod(i-1,Mu)+1,pos));
R(k2,k2)=cos(RecPop.rot(mod(i-1,Mu)+1,pos));
R(k1,k2)=-sin(RecPop.rot(mod(i-1,Mu)+1,pos));
R(k2,k1)=-R(k1,k2);
pos=pos+1;
RR=R*RR;
end
end
% Compute step sizes with rotations
OffPop.x(i,:)=RecPop.x(mod(i-1,Mu)+1,:)+RecPop.std(mod(i-1,Mu)+1,:).*randn(1,Problem.Variables)*RR;
OffPop.std(i,:)=RecPop.std(mod(i-1,Mu)+1,:);
end
if Problem.Variables*(Problem.Variables-1)/2>0
OffPop.rot(i,:)=RecPop.rot(mod(i-1,Mu)+1,:);
end
% Project points into feasible region
OffPop.x(i,:)=Bounds(OffPop.x(i,:),Problem.LB(1:Problem.Variables),Problem.UB(1:Problem.Variables));
[Problem,OffPop.f(i,:)]=ObjEval(Problem, OffPop.x(i,:), varargin{:});
end
% Select the best from Mu+Lambda individuals or Lambda individuals
switch SelectType
case '+'
if Problem.Variables*(Problem.Variables-1)/2>0
temp=[ParentPop.x ParentPop.f ParentPop.std ParentPop.rot;OffPop.x OffPop.f OffPop.std OffPop.rot];
else
temp=[ParentPop.x ParentPop.f ParentPop.std;OffPop.x OffPop.f OffPop.std];
end
```

```
case ','
if Problem.Variables*(Problem.Variables-1)/2>0
temp=[OffPop.x OffPop.f OffPop.std OffPop.rot];
else
temp=[OffPop.x OffPop.f OffPop.std];
end
otherwise
error('ES:Invalid selection option', 'Define: + for plus selection; , for comma selection.');
end
if Problem.Verbose
% The new population is sorted again
temp=sortrows(temp,Problem.Variables+1);
ParentPop.x=temp(1:Mu,1:Problem.Variables);
ParentPop.f=temp(1:Mu,Problem.Variables+1);
ParentPop.std=temp(1:Mu,Problem.Variables+2:2*Problem.Variables+1);
ParentPop.rot=temp(1:Mu,2*Problem.Variables+2:2*Problem.Variables+1+Problem.Variables*(Problem.Variables-1)/2);
% Statistics
Problem.Stats.Best(Problem.Stats.GenCounter+1)=ParentPop.f(1);
Problem.Stats.Worst(Problem.Stats.GenCounter+1)=ParentPop.f(Mu);
Problem.Stats.Mean(Problem.Stats.GenCounter+1)=mean(ParentPop.f);
Problem.Stats.Std(Problem.Stats.GenCounter+1)=std(ParentPop.f);
end
% End of main cycle
% Print if it was stopped due to the maximum of iterations or objective function evaluations
if Problem.Stats.GenCounter>=MaxGenerations || Problem.Stats.ObjFunCounter>=MaxEvals
disp('Maximum number of iterations or objective function evaluations reached');
end
% return leader position and objective function value
x=ParentPop.x(1,:);
fx=ParentPop.f(1);
RunData=Problem.Stats;
return;
function list=create_group(m,r,t)
if t==1
listall=1:m;
for k=1:r
element=randint(1,1,[1 length(listall)]);
list(k)=listall(element);
listall=setdiff(listall,listall(element));
end
else
list=randint(1,r,[1 m]);
end
return
function out=randint(m,n,range)
```

```
range = sort(range);
% Calculate the range the distance for the random number generator
distance = range(2) - range(1);
% Generate the random numbers.
r = floor(rand(m, n) * (distance+1));
% Offset the numbers to the specified value.
out = ones(m,n)*range(1);
out = out + r;
return
function [Problem,Population]=InitPopulation(Problem, InitialPopulation, Size, varargin)
% Check if user provides a valid initial population
if ~isempty(InitialPopulation) && ~isstruct(InitialPopulation)
error('ES:InitPopulation:InitialPopulation', 'Initial population must be defined in a structure.');
else
% Check for size
if length(InitialPopulation)>Size
% User provided an initial population greater than the parent population size
error('ES:InitPopulation:InitialPopulationSize', 'Initial population size must be inferir to Mu.');
end
% Copy the initial population for the population and initialize them
for i=1:length(InitialPopulation)
Population.x(i,:)=Bounds(InitialPopulation(i).x,Problem.LB(1:Problem.Variables),Problem.UB(1:Problem.Variables));
[Problem,Population.f(i)]=ObjEval(Problem, Population.x(i,:), varargin{:});
end
end
% Define initial step sizes based on independent uniform distributions
Population.std=ones(Size,1)*sqrt((Problem.UB-Problem.LB).^2/(12*Problem.Variables));
% Initial rotations are zero
Population.rot=zeros(Size,Problem.Variables*(Problem.Variables-1)/2);
% Compute the centroid for initial population creation
if ~isempty(InitialPopulation)
centroid=sum(InitialPopulation(:).x,1)/length(InitialPopulation)
else
centroid=sum(Problem.UB+Problem.LB,1)/2;
end
% Randomly generate the remaining population
for i=length(InitialPopulation)+1:Size
Population.x(i,:)=Problem.LB(1:Problem.Variables)+(Problem.UB(1:Problem.Variables)-
Problem.LB(1:Problem.Variables)).*rand(1,Problem.Variables);
for j=1:Problem.Variables
Population.x(i,j)=centroid(j)+Population.std(i,j)*randn(1);
end
% Project into feasible region
Population.x(i,:)=Bounds(Population.x(i,:),Problem.LB(1:Problem.Variables),Problem.UB(1:Problem.Variables));
[Problem,Population.f(i)]=ObjEval(Problem, Population.x(i,:), varargin{:});
```

```
end
Population.f=Population.f';
return;
function X=Bounds(X, L, U)
for i=1:length(X)
if X(i)<L(i)
X(i)=L(i);
end
if X(i)>U(i)
X(i)=U(i);
end
end
return
function [Problem,ObjValue] = ObjEval(Problem, x, varargin)
try
ObjValue=feval(Problem.ObjFunction, x, varargin{:});
% update counter
Problem.Stats.ObjFunCounter=Problem.Stats.ObjFunCounter+1;
catch
error('ES:ObjectiveError',['Cannot continue because user supplied objective function', ' failed with the following error:\n%s'], lasterr)
end
return;
function [Value]=GetOption(Option, Options, DefaultOpt)
% Check for user provided options
if isempty(Options) || ~isstruct(Options)
% User does not provides options
Value=DefaultOpt.(Option);
return;
end
% Try the option provided by user
try
Value=Options.(Option);
catch
Value=[];
end
% Option not provided by user
if isempty(Value)
Value=DefaultOpt.(Option);
end
return
```

(4) *run_num.m*

```
function [K,mexp,limd,lim,pd,k,C,R] = run_num(A)
```

```
[…]=parameter(A); % generate model parameters
data(…); % generate ATENA input file
M=processment;
K=matrix_num(M);
end
```

(5) *data.m*

```
function Data(…)
dadosf…=fopen('file_….txt','w'); % open file.txt to write
fprintf(dadosf…, …); % write on file.txt
fclose(dados…); % close file.txt
system('type file_01.txt > file.txt'); % construct ATENA input file.txt
system('type file_….txt >> file.txt');
delete file_….txt; % delete generated file.txt
end
```

(6) *processment.m*

```
function M=processment
system(""…" file.txt results.txt'); % run ATENA
system(""…" -f numeric_processment.awk.txt results.txt > processed.txt'); % run GAWK
M=csvread('processed.txt'); % read obtained results in MATLAB
delete ….txt; % delete generated files
end
```

(7) *matrix_num.m*

```
function K=matrix_num(M)
[k,q]=size(M);
K(1,…)=0;
for u=1:k
K(u+1,…)=M(u,…); % generate matrix K with results
end
end
```

(8) *test_num.m*

```
function BD = test_num(K,pd)
% Number of lines from matrix BD
k=size(K(:,:));
k=k(1);
q=max(K(:,1));
v=round(q/pd);
if mod(v,2)==0
```

```
if v*pd<q
o=v+1;
else
o=v;
end
else
o=v;
end
% Computing BD from K using one discretization by steps (interpolation)
BD(1,1)=0;
BD(2,1)=pd;
for t=2:o-1
for j=1:k-1
if((K(j+1,1)==BD(t,1)))
BD(t,2)=K(j+1,2);
end
if((K(j+1,1)-BD(t,1))*(K(j,1)-BD(t,1))<0)
BD(t,2)=(K(j+1,2)-K(j,2))/(K(j+1,1)-K(j,1))*(BD(t,1)-K(j,1))+K(j,2);
break;
end
end
if(t<o-1)
BD(t+1,1)=BD(t,1)+pd;
end
end
end
```

(9) *run_exp.m*

```
function P = run_exp(mexp)
P=cell(mexp,1);
for u=1:mexp
% read each experimental data
q1=(['experimental',int2str(u),'.txt']);
q2=(['experimental_result',int2str(u),'.txt']);
t1=(['system(""…" -f introduction_comma.awk.txt ', q1, ' > ', q2, ' ")']); % run GAWK
t2=(['system(""…" -f experimental_processement.awk.txt ', q1, ' > ', q2, ' ")']); % run GAWK
K=(['csvread("', q2, '")']); % read experimental data in MatLab
P{u}=eval(K); % read each registered data
t3=(['delete ',q2]); % delete each register
end
end
```

(10) *matrix_exp.m*

```
function Q=matrix_exp(P,u,,limd,flim)
e=size(P{u,1});
e1=e(1);
w=1;
flag=0;
Q{u,1}(1,1)=0;
Q{u,1}(1,2)=0;
while w+1<=e1 && flag==0 && P{u,1}(w,1)<=limd && P{u,1}(w,3)<=flim
Q{u,1}(w+1,1)=P{u,1}(w+1,1);
Q{u,1}(w+1,2)=P{u,1}(w+1,3);
w=w+1;
end
end
```

(11) *test_exp.m*

```
function BE = test_exp(P,mexp,pd)
BE=cell(mexp,1);
for u=1:mexp
% Number of lines from matrix BE
k=size(P{u});
k=k(1);
q=max(P{u}(:,1));
v=round(q/pd);
if mod(v,2)==0
if v*pd<q
o=v+1;
else
o=v;
end
else
o=v;
end
% Computing BE from P using one discretization by steps (interpolation)
BE{u}(1,1)=0;
BE{u}(2,1)=pd;
for t=2:o-1
for j=1:k-1
if((P{u}(j+1,1)==BE{u}(t,1)))
BE{u}(t,2)=P{u}(j+1,2);
end
if((P{u}(j+1,1)-BE{u}(t,1))*(P{u}(j,1)-BE{u}(t,1))<0)
BE{u}(t,2)=(P{u}(j+1,2)-P{u}(j,2))/(P{u}(j+1,1)-P{u}(j,1))*(BE{u}(t,1)-P{u}(j,1))+P{u}(j,2);
```

```
break;
end
end
if(t<o-1)
BE{u}(t+1,1)=BE{u}(t,1)+pd;
end
end
end
end
```

(12) *limit.m*

```
function v = limit(BD,BE,mexp,k)
v=zeros(mexp,1);
for i=1:mexp
v1=max(BD(:,2));
v2=max(BE{i}(:,2));
v3=max(v1,v2);
v(i)=k*v3;
end
end
```

Appendix B: Probabilistic Analysis Routines

According to what was previously indicated at chapter four, some Matlab® routines [25, 129] were developed within the probabilistic analysis of structures. These routines are linked according to Figure B.1, organization chart. The routines *lhs_iman_n.m*, *mchol.m, latin_hs.m* and *ranking.m* refer to the Latin Hypercube sampling algorithm [173]. This algorithm was already implemented by Stein [173]. Some of the developed routines are provided further. Other routines, such as the *parameter.m* and the *plot.m*, should be adapted to each situation.

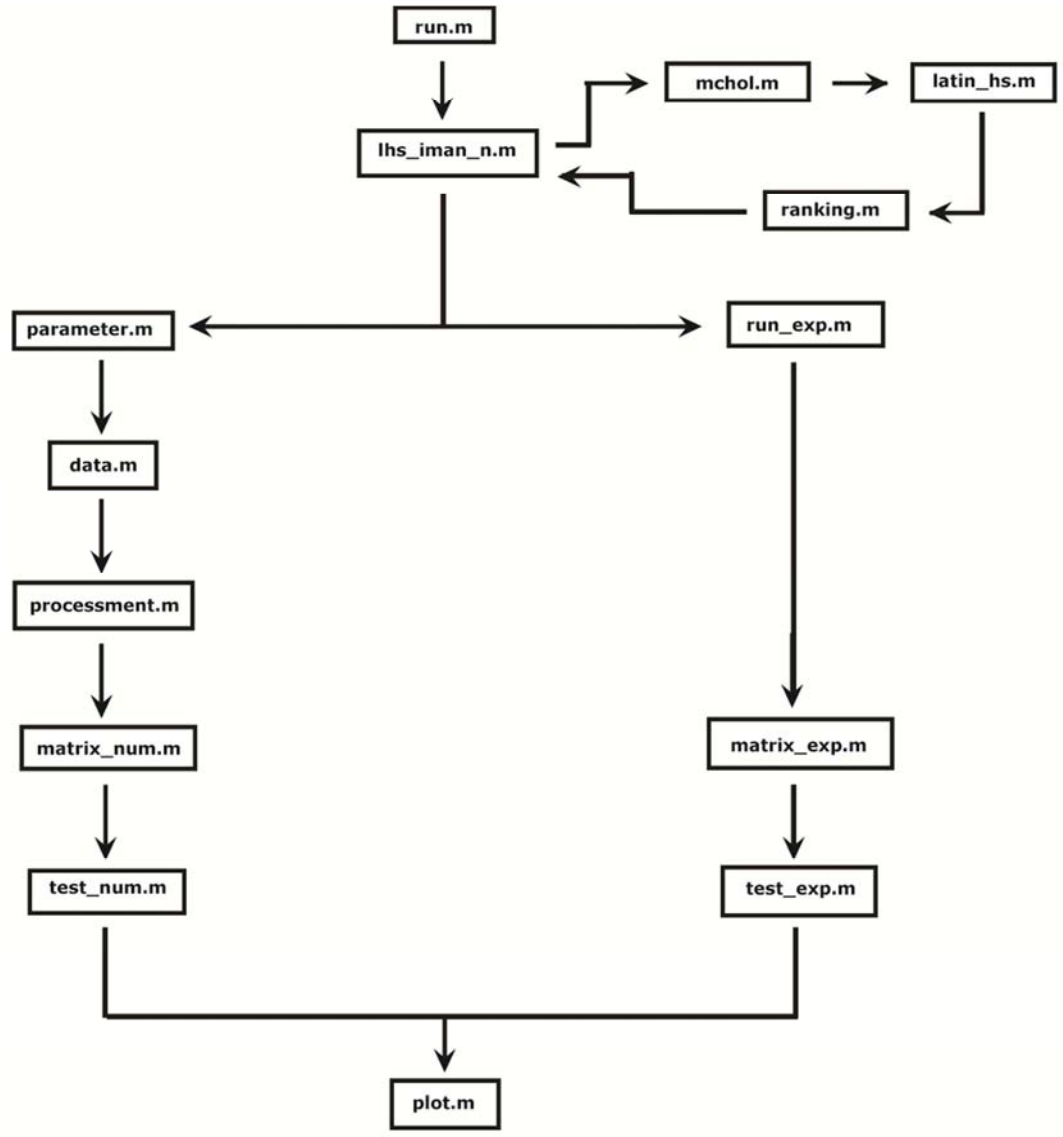


Figure B.1. Organization chart.

(1) *run.m*

```
nsample = (…); % Number of parameters to be generated
xmean=[…]; % Parameter mean value
xsd=[…]; % Parameter standard deviation
corr=[…]; % Correlation matrix
B=[…]; % Deterministic parameters
presults=run(xmean,xsd,corr,nsample,B);
```

```
function presults=run(xmean,xsd,corr,nsample,B)
z=lhs_iman_n(xmean,xsd,corr,nsample); % Random parameter generation (z)
ANM=cell(nsample,1);
for i=1:nsample
A=Convert(z,B,i); % Compute A from z (variable) and B (fixed)
[…]=parameter(A); % generate model parameters
data(…); % generate ATENA input file
M=processment;
AN=matrix_num(M);
ANM{i}(:,:)=AN(:,:);
end
% Computing maximum load
FMAX=max(ANM{1}(:,1));
for i=1:nsample
FM(i)=max(ANM{i}(:,1));
FMAX1=max(ANM{i}(:,1));
if FMAX1>=FMAX
FMAX=FMAX1;
end
end
[muhat,sigmahat]=normfit(FM);
mean=muhat;
stdev=sigmahat;
interval=norminv([0.025,0.975],mean,stdev);
flim=k*FMAX;
% Compute the confidence intervals
for i=1:nsample
BD= test_num(ANM,i,pd);
end
p=size(BD{1}(:,:));
p1=p(1);
for i=1:nsample
p=size(BD{i}(:,:));
p=p(1);
if p>=p1
p1=p;
```

```
end
end
for i=1:nsample
FT{1,1}(i)=0.0;
end
for j=2:p1
for i=1:nsample
p=size(BD{i}(:,:));
p=p(1);
if p>=j && BD{i}(j,1)<=limd && BD{i}(j,2)<=flim && BD{i}(j,2)~=0
FT{j,1}(i)=BD{i}(j,2);
end
end
end
q=size(FT);
q1=q(1);
for i=1:nsample
q=size(FT{i}(:));
q=q(1);
if q>=q1
q1=q;
end
if q<q1
q1=q;
end
end
for j=2:q1
[muhat,sigmahat]=normfit(FT{j}(:));
mn(j)=muhat;
st(j)=sigmahat;
int(j,:)=norminv([0.025,0.975],muhat,sigmahat);
end
for i=1:q1
LB(i,1)=pd*(i-1);
UB(i,1)=pd*(i-1);
LB(i,2)=int(i,1);
UB(i,2)=int(i,2);
end
% Plot the confidence intervals
Q=run_exp(mexp,limd,flim,LB,UB);
BE=test_exp(Q,mexp,pd,limd,flim);
for i=1:mexp
BEF(i)=max(BE{i}(:,2));
end
r=size(BE{1}(:,:));
```

```
r1=r (1);
for i=1:mexp
r=size(BE{i}(:,:));
r=r(1);
if r>=r1
r1=r;
end
if r<r1
r11=r;
end
end
% Compute the gap between numeric and experimental data
% Index 0
for i=1:mexp
s1=min(q1,r1);
countn1=0;
countn0(i,1)=s1;
for j=1:s1
if BE{i,1}(j,2)>=LB(j,2) && BE{i,1}(j,2)<=UB(j,2)
countn1=countn1+1;
end
end
index0(i,1)=countn1/countn0(i,1)*100;
end
% Index 1
for i=1:mexp
s1=min(q1,r1);
countn1=0;
index1(i,1)=0.0;
if s1>=1
countn0(i,1)=s1;
for j=2:s1
P=normcdf(BE{i,1}(j,2),mn(j,1),st(j,1));
p(i,j)=(abs(P-0.5)/0.5)*stdev/mean;
p1(i,j)=(1-p(i,j))*100;
countn1=countn1+p1(i,j);
end
index1(i,1)=countn1/countn0(i,1);
else
index1(i,1)=0.0;
end
end
% Maximum applied load
% Index 0
index0=zeros(mexp,1);
```

```
for i=1:mexp
if BEF(i,1)>=interval(1) && BEF(i,1)<=interval(2)
index0(i,1)=100;
else
index0(i,1)=0;
end
end
% Index 1
index1=zeros(mexp1,1);
for i=1:mexp
P=normcdf(BEF(i,1),mean,stdev);
p(i,1)=(abs(P-0.5)/0.5)*stdev/mean;
index1(i,1)=(1-p(i,1))*100;
end
% Obtained results
presults=struct(…);
end
```

(2) *lhs_iman_n.m* [173]

```
function z=lhs_iman_n(xmean,xsd,corr,nsample,ntry)
% LHS with correlation, normal distribution - Method of Iman & Conover
nvar=length(xmean);
if(nargin==4), ntry=1; end;
% induce data with correlation
[L,D,E]=mchol(corr);
%P = chol(corr+E)';
P=L*sqrt(D);
xm=zeros(1,nvar);
xs=ones(1,nvar);
R=latin_hs(xm,xs,nsample,nvar);
T = corrcoef(R);
[L,D,E]=mchol(T);
%Q=chol(T+E)';
Q=L*sqrt(D);
S = P * inv(Q);
RB= R*S';
amin=realmax;
for il=1:ntry
for j=1:nvar
[r,id]=ranking(RB(:,j));
[RS,id]=sort(R(:,j));
z(:,j) = RS(r).*xsd(j)+xmean(j);
end
ae=sum(sum(abs(corrcoef(z)-corr)));
```

```
if(ae<amin),
zb=z;
amin=ae;
end;
end
z=zb;
```

(3) *mchol.m* [173]

```
function [L,D,E,pneg]=mchol(G)
n=size(G,1);
gamma=max(diag(G));
zi=max(max(G-diag(diag(G))));
nu=max([1,sqrt(n^2-1)]);
beta2=max([gamma, zi/nu, 1.0E-15]);
C=diag(diag(G));
L=zeros(n);
D=zeros(n);
E=zeros(n);
for j=1:n,
bb=[1:j-1];
ee=[j+1:n];
if (j > 1),
L(j,bb)=C(j,bb)./diag(D(bb,bb))';
end;
if (j >= 2),
if (j < n),
C(ee,j)=G(ee,j)-(L(j,bb)*C(ee,bb)')';
end;
else
C(ee,j)=G(ee,j);
end;
if (j == n)
theta(j)=0;
else
theta(j)=max(abs(C(ee,j)));
end;
D(j,j)=max([eps,abs(C(j,j)),theta(j)^2/beta2]');
E(j,j)=D(j,j)-C(j,j);
ind=[j*(n+1)+1 : n+1 : n*n]';
C(ind)=C(ind)-(1/D(j,j))*C(ee,j).^2;
end;
ind=[1 : n+1 : n*n]';
L(ind)=1;
if ((nargout == 4) & (min(diag(C)) < 0.0))
```

```
[m,col]=min(diag(C));
rhs=zeros(n,1);
rhs(col)=1;
pneg=L'\rhs;
else
pneg=[];
end;
return
```

(4) *latin_hs.m* [173]

```
function s=latin_hs(xmean,xsd,nsample,nvar)
ran=rand(nsample,nvar);
s=zeros(nsample,nvar);
for j=1: nvar
idx=randperm(nsample);
P=(idx'-ran(:,j))/nsample; % probability of the cdf
s(:,j) = xmean(j) + ltqnorm(P).* xsd(j); % this can be replaced by any inverse distribution function
end
```

(5) *ranking.m* [173]

```
function [r,i]=ranking(x)
n=length(x);
[s,i]=sort(x);
r(i,1)=[1:n]';
```

(6) *data.m*

```
function data(…)
dadosf…=fopen('file_….txt','w'); % open file.txt to write
fprintf(dadosf…, …); % write on file.txt
fclose(dados…); % close file.txt
system('type file_01.txt > file.txt'); % construct ATENA input file.txt
system('type file_….txt >> file.txt');
delete file_….txt; % delete generated file.txt
end
```

(7) *processment.m*

```
function M=processment
system('"…" file.txt results.txt'); % run ATENA
system('"…" -f numeric_processment.awk.txt results.txt > processed.txt'); % run GAWK
M=csvread('processed.txt'); % read obtained results in MATLAB
delete ….txt; % delete generated files
```

```
end
```

(8) *matrix_num.m*

```
function AN=matrix_num(M)
[k,q]=size(M);
AN(1,…)=0;
for u=1:k
AN(u+1,…)=M(u,…); % generate matrix K with results
end
end
```

(9) *test_num.m*

```
function BD=test_num(ANM,i,pd)
% Number of lines from matrix BD
q=max(ANM{i}(:,1));
v=round(q/pd);
if mod(v,2)==0
if v*pd<q
o=v+1;
else
o=v;
end
else
o=v;
end
% Computing BD from ANM using one discretization by steps (interpolation)
[k,l]=size(ANM{i}(:,:));
BD{i}(1,1)=0;
BD{i}(2,1)=pd;
for t=2:o-1
for j=1:k-1
if((ANM{i}(j+1,1)==BD{i}(t,1)))
BD{i}(t,2)=ANM{i}(j+1,2);
end
if((ANM{i}(j+1,1)-BD{i}(t,1))*(ANM{i}(j,1)-BD{i}(t,1))<0)
BD{i}(t,2)=(ANM{i}(j+1,2)-ANM{i}(j,2))/(ANM{i}(j+1,1)-ANM{i}(j,1))*(BD{i}(t,1)-ANM{i}(j,1))+ANM{i}(j,2);
break;
end
end
if(t<o-1)
BD{i}(t+1,1)=BD{i}(t,1)+pd;
end
end
```

```
end
```

(10) *run_exp.m*

```
function Q=run_exp(mexp,limd,flim,LB,UB)
P=cell(mexp,1);
L1=cell(mexp,1);
% read each experimental data
for u=1:mexp
q1=(['experimental',int2str(u),'.txt']);
q2=(['resultado_experimental',int2str(u),'.txt']);
t2=(['system(""…" -f introduction_comma.awk.txt ', q1, ' > ', q2, ' ")']); % run GAWK
t3=(['system(""…" -f experimental_processment.awk.txt ', q1, ' > ', q2, ' ")']); % run GAWK
K=(['csvread("", q2, ")']); % read experimental data in MatLab
P{u,1}=eval(K); % read each registered data
Q=matrix_exp(P,u,limd,flim);
t4=('Plot(Q,u,LB,UB)'); % Plot each register
eval(t4);
t5=(['delete ',q2]); % delete each register
end
end
```

(11) *matrix_exp.m*

```
function Q=matrix_exp(P,u,,limd,flim)
e=size(P{u,1});
e1=e(1);
w=1;
flag=0;
Q{u,1}(1,1)=0;
Q{u,1}(1,2)=0;
while w+1<=e1 && flag==0 && P{u,1}(w,1)<=limd && P{u,1}(w,3)<=flim
Q{u,1}(w+1,1)=P{u,1}(w+1,1);
Q{u,1}(w+1,2)=P{u,1}(w+1,3);
w=w+1;
end
end
```

(12) *test_exp.m*

```
function BE=test_exp(Q,mexp,pd,limd,flim);
BE=cell(mexp,1);
for u=1:mexp
% Number of lines from matrix BE
k=size(P{u,1});
```

```
k=k(1);
q=max(P{u,1}(:,1));
v=round(q/pd);
if mod(v,2)==0
if v*pd<q
o=v+1;
else
o=v;
end
else
o=v;
end
% Computing BE from P using one discretization by steps (interpolation)
BE{u,1}(1,1)=0;
BE{u,1}(2,1)=pd;
for t=2:o-1
for j=1:k-1
if((P{u,1}(j+1,1)==BE{u,1}(t,1)))
BE{u,1}(t,2)=P{u,1}(j+1,2);
end
if((P{u,1}(j+1,1)-BE{u,1}(t,1))*(P{u,1}(j,1)-BE{u,1}(t,1))<0)
BE{u,1}(t,2)=(P{u,1}(j+1,2)-P{u,1}(j,2))/(P{u,1}(j+1,1)-P{u,1}(j,1))*(BE{u,1}(t,1)-P{u,1}(j,1))+P{u,1}(j,2);
break;
end
end
if(t<o-1)
BE1{u,1}(t+1,1)=BE1{u,1}(t,1)+pd;
end
end
end
BE=cell(mexp,1);
for u=1:mexp
k=size(BE1{u,1}(:,:));
k=k(1);
for i=1:k
if BE1{u,1}(i,1)<=limd && BE1{u,1}(i,2)<=flim
BE{u,1}(i,1)=BE1{u,1}(i,1);
BE{u,1}(i,2)=BE1{u,1}(i,2);
end
end
end
```

Appendix C: Bayesian Inference Routines

According to what was previously indicated at chapter four, some Matlab® routines [25, 129] were developed within the Bayesian inference procedure [15]. These routines are linked according to Figure C.1, organization chart. The routine *matbugs.m* establishes the interface between Matlab® [25, 129] and WinBugs® [111] software. Some of the developed routines are provided further. Other routines, such as the *plotbugsmu.m*, *plotbugssigma.m*, *plotbugsmusigma.m* and the *plotbugswbl.m*, should be adapted to each situation.

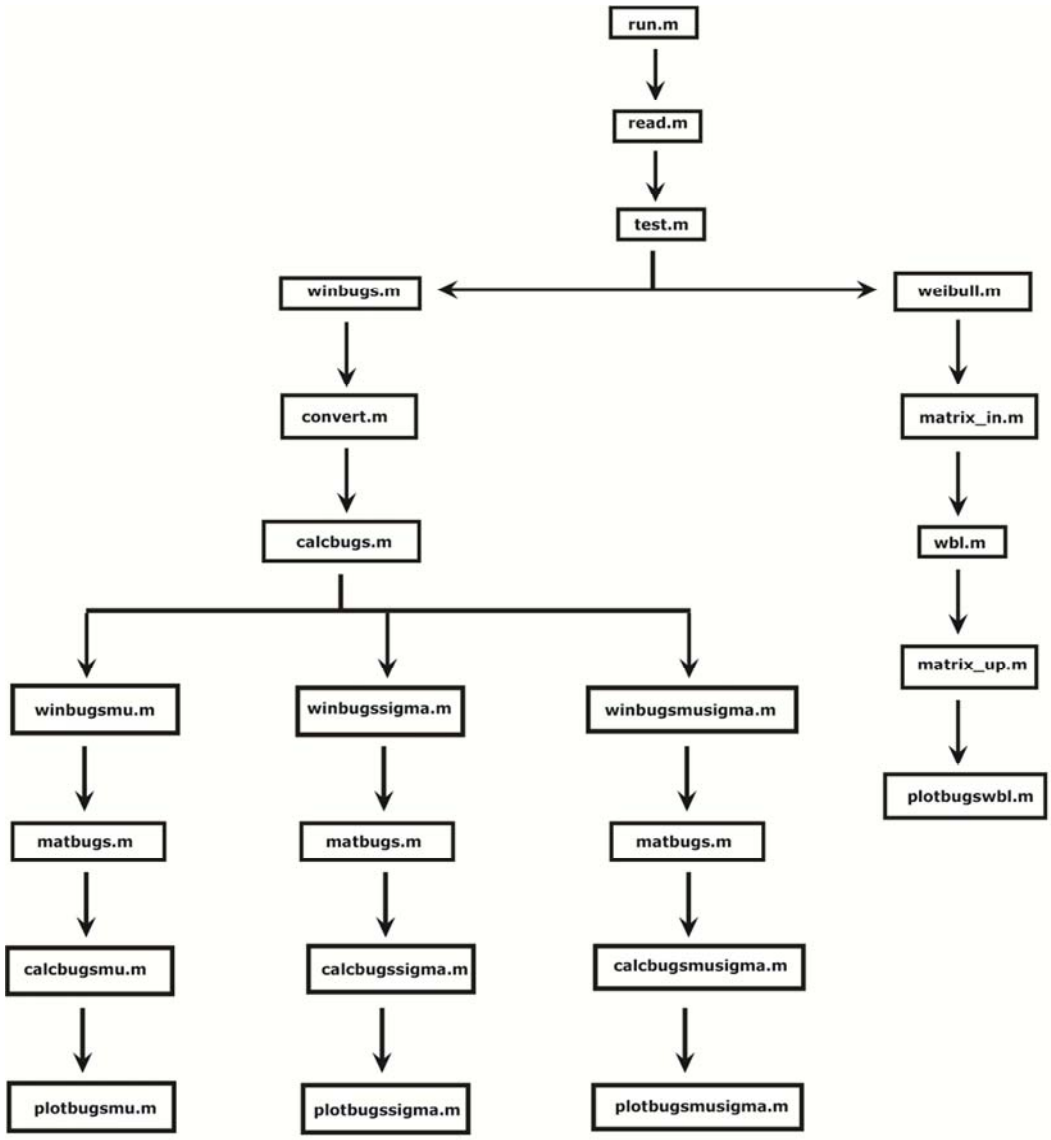


Figure C.1. Organization chart.

(1) *run.m*

```
function res=run
[data,winbugs_model,flq]=read; % Read data
if flq==0
test(winbugs_model); % Perform the check-up of Winbugs model parameters
res=winbugs(data,winbugs_model,flq); % Run Winbugs
elseif flq==1
test(winbugs_model); % Perform the check-up of Winbugs model parameters
res=weibull(winbugs_model,data,flq); % Run Winbugs with Weibull distribution
end
end
```

(2) *read.m*

```
function [data,winbugs_model,flq]=read
% Read data file
fid = fopen('data.txt');
dt = textscan(fid, ' %f');
fclose(fid);
% Convert data
data(:,:)=dt{1}(:,:);
data=data';
% Read Winbugs parameters
fid = fopen('winbugs_parameters.txt');
wb = textscan(fid, '%s %f');
fclose(fid);
% Convert Winbugs parameters (structure)
winbugs_model.(…)=wb{2}(…);
% Weibull analysis
flq=winbugs_model.wbl;
end
```

(3) *test.m*

```
function test(winbugs_model)
if (…) % Test Winbugs parameters
% Display on Matlab
error('Error in WinBugs_Model Parameters');
end
end
```

(4) *winbugs.m*

```
function res=winbugs(data,winbugs_model,flq)
```

```
[P,flp,flc,flk,flu]=convert(data,winbugs_model); % Convert Winbugs parameters
if flp==0
[N,PL,mean2,std2,x]=calcbugs(P,flc);
% Run WinBugs
stats=winbugsmu(N,PL,std2,winbugs_model);
% Compute stats and random distribution functions
[res,pdfNormal1,pdfNormal2,pdfNormal3,pdfNormal4]=calcbugsmu(stats,N,mean2,std2,x,winbugs_model,flk,flc);
if flq==0
% Plot random distribution functions
plotbugsmu(pdfNormal1,pdfNormal2,pdfNormal3,pdfNormal4,x);
end
elseif flp==1
[N,PL,mean2,std2,x]=calcbugs(P,flc);
% Run Winbugs
stats=winbugssigma(N,PL,mean2,winbugs_model);
% Compute stats and random distribution functions
[res,pdfNormal1,pdfNormal2,pdfNormal3,f_x_4]=calcbugssigma(stats,N,mean2,std2,x,winbugs_model,flk,flc);
if flq==0
% Plot random distribution functions
plotbugssigma(pdfNormal1,pdfNormal2,pdfNormal3,f_x_4,x,flu);
end
elseif flp==2
[N,PL,mean2,std2,x]=calcbugs(P,flc);
% Run WinBugs
stats=winbugsmusigma(N,PL,flk,winbugs_model);
% Compute stats and random distribution functions
[res,pdfNormal11,pdfNormal12,pdfNormal2,pdfNormal4,f_x_3,f_x_5]=
calcbugsmusigma(stats,N,mean2,std2,x,winbugs_model,flk,flc);
if flq==0
% Plot random distribution functions
plotbugsmusigma(pdfNormal11,pdfNormal12,pdfNormal2,pdfNormal4,f_x_3,f_x_5,x,flu);
end
end
end
```

(5) *convert.m*

```
function [P,flp,flc,flk,flu]=convert(data,winbugs_model)
P=data;
flp=winbugs_model.par;
flc=winbugs_model.log;
flk=winbugs_model.prior;
flu=winbugs_model.graph;
end
```

(6) *calcbugs.m*

```
function [N,PL,mean2,std2,x]=calcbugs(P,flc)
p=size(P);
q=p(2);
N=q;
x=0:0.1:1000;
if flc==0
PL=P;
mean2=mean(PL);
std2=std(PL,1);
elseif flc==1
PL=log(P);
mean2=mean(PL);
std2=std(PL,1);
end
end
```

(7) *winbugsmu.m*, *winbugssigma.m*, *winbugsmusigma.m*

```
function stats=winbugsmu(N,PL,std2,winbugs_model)
mean1=winbugs_model.mean;
tau1=winbugs_model.std^(-2);
dataStruct = struct('N', N, income', PL, 'sigma', std2, 'mean', mean1, 'taum', tau1);
S.mu = 0;
initStructs = S;
[~,stats] = matbugs(dataStruct, fullfile(pwd, 'normal_mu.txt'), 'init', initStructs, 'view', winbugs_model.view, 'nburnin', winbugs_model.nburnin, 'nsamples', winbugs_model.nsamples, 'thin', winbugs_model.thin, 'monitorParams', {'mu'}, 'Bugdir', '…', 'DICstatus',winbugs_model.dic);
end
```

```
function stats=winbugssigma(N,PL,mean2,winbugs_model)
alfa1=winbugs_model.alfa;
beta1=winbugs_model.beta;
dataStruct = struct('N', N, 'income', PL, 'mu', mean2, 'alfa', alfa1, 'beta', beta1);
S.tau = 1;
initStructs = S;
[~, stats] = matbugs(dataStruct, fullfile(pwd, 'normal_tau.txt'), 'init', initStructs, 'view', winbugs_model.view, 'nburnin', winbugs_model.nburnin, 'nsamples', winbugs_model.nsamples, 'thin', winbugs_model.thin, 'monitorParams', {'tau'}, 'Bugdir', '…', 'DICstatus',winbugs_model.dic);
end
```

```
function stats=winbugsmusigma(N,PL,flk,winbugs_model)
mean1=winbugs_model.mean;
tau1=winbugs_model.std^(-2);
alfa1=winbugs_model.alfa;
beta1=winbugs_model.beta;
```

```
if flk==0
dataStruct = struct('N', N, 'income', PL, 'mean', mean1, 'taum', tau1, 'alfa', alfa1, 'beta', beta1);
S.mu = 0;
S.tau = 1;
initStructs = S;
[~,stats] = matbugs(dataStruct, fullfile(pwd, 'normal_mu_tau.txt'), init', initStructs, 'view', winbugs_model.view, 'nburnin', winbugs_model.nburnin, 'nsamples', winbugs_model.nsamples, 'thin', winbugs_model.thin, 'monitorParams', {'mu','tau'}, 'Bugdir', '…', 'DICstatus',winbugs_model.dic);
elseif flk==1
N0=winbugs_model.n0;
dataStruct = struct('N', N, 'income', PL, 'mean', mean1, 'alfa', alfa1, 'beta', beta1, 'N0', N0);
S.mu = 0;
S.tau = 1;
initStructs = S;
[~, stats] = matbugs(dataStruct, fullfile(pwd, 'normal_mu_tau_conj.txt'), 'init', initStructs, 'view', winbugs_model.view, 'nburnin', winbugs_model.nburnin, 'nsamples', winbugs_model.nsamples, 'thin', winbugs_model.thin, 'monitorParams', {'mu','tau'}, 'Bugdir', '…', 'DICstatus',winbugs_model.dic);
end
end
```

(8) *matbugs.m* (by Kevin Murphy and Maryam Mahdaviani, August 2005: http://code.google.com/p/matbugs/)

```
function [samples, stats, structArray] = matbugs(dataStruct, bugsModel, varargin)
% MATBUGS a Matlab interface for WinBugs
[openBUGS, junk] =  process_options(varargin, 'openBUGS', 0);
if openBUGS
Bugdir = '…';
else
Bugdir = '…';
end
if isunix
wine=1;
pathPrefix='z:';
else
wine=0;
pathPrefix='';
end
[initStructs, Bugdir, nChains, view, workingDir, nBurnin, nSamples, monitorParams, thin, blocking, refreshrate, DICstatus, openBUGS, junk] = process_options(varargin, 'init', {}, 'Bugdir', Bugdir, 'nChains', 3, 'view', 0, 'workingDir', fullfile(pwd,'tmp'), 'nBurnin', 1000, 'nSamples', 5000, 'monitorParams', {}, 'thin', 1, 'blocking', 1, 'refreshrate',100, 'DICstatus',0, 'openBUGS', 0);
if 0 % length(initStructs) ~= nChains
error(['init structure does not match number of chains ', sprintf('(%d)', nChains)]);
end
if ~exist(workingDir, 'dir')
mkdir(pwd, 'tmp');
end
```

```
log_filename = fullfileKPM(workingDir, 'log.txt');
his_filename = fullfileKPM(workingDir, 'history.txt');
if wine
scriptFile = [Bugdir,filesep,'script.txt'];
else
scriptFile = [Bugdir,'\','script.txt'];
end
bugsModel = strrep(bugsModel, '\', '/');
codaFile = fullfileKPM(workingDir, 'coda');
fid = fopen(scriptFile,'w');
if (fid == -1)
error(['Cannot open ', scriptFile]);
end
fprintf(fid, 'display("log") \n');
if openBUGS
fprintf(fid, 'modelCheck("%s")\n',[pathPrefix,bugsModel]);
else
fprintf(fid, 'check("%s")\n',[pathPrefix,bugsModel]);
end
if ~isempty(dataStruct)
dataFile = fullfileKPM(workingDir, 'data.txt');
dataGen(dataStruct, dataFile);
if openBUGS
fprintf(fid, 'modelData("%s")\n', [pathPrefix,dataFile]);
else
fprintf(fid, 'data("%s")\n', [pathPrefix,dataFile]);
end
end
if openBUGS
fprintf(fid, 'modelCompile(%u) \n', nChains);
else
fprintf(fid, 'compile(%u) \n', nChains);
end
initfileN = size(initStructs,2);
for i=1:initfileN
initFileName = fullfileKPM(workingDir, ['init_', num2str(i) '.txt']);
dataGen(initStructs(i), initFileName)
if openBUGS,
fprintf(fid, 'modelInits("%s", %u)\n', [pathPrefix,initFileName], i);
else
fprintf(fid, 'inits (%u, "%s")\n', i, [pathPrefix,initFileName]);
end
end
if 0
fprintf(fid, 'blockfe(1)\n');
```

```
end
fprintf(fid, 'refresh(%u) \n', refreshrate);
if openBUGS
fprintf(fid, 'modelGenInits() \n');
fprintf(fid, 'modelUpdate(%u, TRUE)\n', nBurnin);
else
fprintf(fid, 'gen.inits() \n');
fprintf(fid, 'update(%u)\n', nBurnin);
end
if isempty(monitorParams)
if openBUGS
fprintf(fid, 'samplesSet ("*")\n');
else
fprintf(fid, 'set (*)\n');
end
else
for i=1:length(monitorParams)
if openBUGS
fprintf(fid, 'samplesSet (%s)\n', strrep(monitorParams{i}, '_', '.'));
else
fprintf(fid, 'set (%s)\n', strrep(monitorParams{i}, '_', '.'));
end
end
end
if DICstatus; fprintf(fid, 'dic.set()\n');
if openBUGS
fprintf(fid, 'samplesThin(%u)\n', thin);
fprintf(fid, 'modelUpdate(%u)\n', nSamples);
fprintf(fid, 'samplesCoda("*", "%s")\n',  [pathPrefix,codaFile]);
fprintf(fid, 'samplesStats("*")\n');
fprintf(fid, 'samplesDensity("*")\n');
fprintf(fid, 'samplesHistory("*")\n');
else
fprintf(fid, 'thin.updater(%u)\n', thin);
fprintf(fid, 'update(%u)\n', nSamples);
fprintf(fid, 'coda(*, "%s")\n',  codaFile);
fprintf(fid, 'stats(*)\n');
end
if DICstatus; fprintf(fid, 'dic.stats()\n #endDIC');
if openBUGS
fprintf(fid, 'samplesHistory("*", "%s")\n', his_filename);
fprintf (fid, 'modelSaveLog("%s")\n',  log_filename);
else
fprintf(fid, 'history(*, "%s")\n', his_filename);
fprintf (fid, 'save ("%s")\n',  log_filename);
```

```
end
if (view == 0)
if openBUGS
fprintf(fid, 'modelQuit("y")\n');
else
fprintf(fid, 'quit() \n');
end
end
fclose(fid);
if openBUGS
f = fullfile(Bugdir, 'winbugs.exe');
else
if wine
f = fullfile(Bugdir, 'WinBUGS14.exe');
else
f = fullfile(Bugdir, 'Winbugs14.exe');
end
end
str = ['"',f,'" /PAR script.txt'];
dos(str);
if openBUGS
codaIndex = [codaFile, 'CODAindex.txt'];
else
codaIndex = [codaFile, 'Index.txt'];
end
for i=1:nChains
if openBUGS
codaF = [codaFile, 'CODAchain', num2str(i), '.txt'];
else
codaF = [codaFile, num2str(i), '.txt'];
end
S = bugs2mat(codaIndex, codaF);
structArray(i) = S;
end
samples = structsToArrays(structArray);
stats = computeStats(samples);
if DICstatus;
DICstats = getDICstats(workingDir);
stats.DIC = DICstats;
end
if nChains == 1
disp('EPSR not calculated (only one chain)');
end
end
function dataGen(dataStruct, fileName)
```

```
if nargin<2
error(['This function needs two arguments']);
end
fieldNames = fieldnames(dataStruct);
Nparam = size(fieldNames, 1);
fid = fopen(fileName, 'w');
if fid == -1
error(['Cannot open ', fileName ]);
end
fprintf(fid,'list(');
for i=1:Nparam
fn = fieldNames(i);
fval = fn{1};
val = getfield(dataStruct, fval);
[sfield1, sfield2]= size(val);
msfield = max(sfield1, sfield2);
newfval = strrep(fval, '_', '.');
if ((sfield1 == 1) && (sfield2 == 1))
fprintf(fid, '%s=%G',newfval, val);
elseif ((length(size(val)) == 2) && ((sfield1 == 1) || (sfield2 == 1)))
fprintf(fid, '%s=c(',newfval);
for j=1:msfield
if (isnan(val(j)))
fprintf(fid,'NA');
else
fprintf(fid,wb_strval(val(j)));
end
if (j<msfield)
fprintf(fid, ', ');
else
fprintf(fid, ')');
end
end
else
valsize = size(val);
alldatalen = prod(valsize);
if length(valsize)<3
alldata = reshape(val', [1, alldatalen]);
elseif length(valsize)==3
clear valTransp
for j=1:valsize(3)
valTransp(j,:,:)=val(:,:,j)';
end
alldata=valTransp(:)';
else
```

```
['Error: 4D and higher dimensional arrays not accepted']
return
end
fprintf(fid, '%s=structure(.Data=c(', newfval);
for j=1:alldatalen
if (isnan(alldata(j)))
fprintf(fid,'NA');
else
fprintf(fid,wb_strval(alldata(j)));
end
if (j < alldatalen)
fprintf(fid,',');
else
fprintf(fid,'), .Dim=c(', alldata(j));
end
end
for j=1:length(valsize)
if (j < length(valsize))
fprintf(fid, '%G,', valsize(j));
else
fprintf(fid, '%G))', valsize(j));
end
end
end
if (i<Nparam)
fprintf(fid, ', ');
else
fprintf(fid, ')\n');
end
end
fclose(fid);
end
function s = wb_strval(v)
s = sprintf('%G', v);
if strfind(s, 'E')
if length(strfind(s, '.')) == 0
s = strrep(s, 'E', '.0E');
end
s = strrep(s, 'E+0', 'E+');
s = strrep(s, 'E-0', 'E-');
end
end
function f = fullfileKPM(varargin)
f = fullfile(varargin{:});
f = strrep(f, '\', '/');
```

```
end
function A = structsToArrays(S)
C = length(S);
fld = fieldnames(S);
A = [];
for fi=1:length(fld)
fname = fld{fi};
tmp = getfield(S(1), fname);
sz = size(tmp);
psz = prod(sz);
data = zeros(C, psz);
for c=1:C
tmp = getfield(S(c), fname);
%data = cat(1, data, tmp);
data(c,:) = tmp(:)';
end
if sz(2) > 1 % vector or matrix variable
data = reshape(data, [C sz]);
end
A = setfield(A, fname, data);
end
end
function [Rhat, m, s] = EPSR(samples)
[n m] = size(samples);
meanPerChain = mean(samples,1);
meanOverall = mean(meanPerChain);
if m > 1
B = (n/(m-1))*sum( (meanPerChain-meanOverall).^2);
varPerChain = var(samples);
W = (1/m)*sum(varPerChain);
vhat = ((n-1)/n)*W + (1/n)*B;
Rhat = sqrt(vhat/(W+eps));
else
Rhat = nan;
end
m = meanOverall;
s = std(samples(:));
end
function stats = computeStats(A)
fld = fieldnames(A);
N = length(fld);
stats = struct('Rhat',[], 'mean', [], 'std', []);
for fi=1:length(fld)
fname = fld{fi};
samples = getfield(A, fname);
```

```
sz = size(samples);
clear R m s
Nchains = sz(1);
Nsamples = sz(2);
st_mean_per_chain = mean(samples, 2);
st_mean_overall   = mean(st_mean_per_chain, 1);
if Nchains > 1
B = (Nsamples/Nchains-1) * sum((st_mean_per_chain - repmat(st_mean_overall, [Nchains,1])).^2);
varPerChain = var(samples, 0, 2);
W = (1/Nchains) * sum(varPerChain);
vhat = ((Nsamples-1)/Nsamples) * W + (1/Nsamples) * B;
Rhat = sqrt(vhat./(W+eps));
else
Rhat = nan;
end
samp_shape = size(squeeze(st_mean_overall));
reshape_target = [Nchains * Nsamples, samp_shape];
reshaped_samples = reshape(samples, reshape_target);
st_std_overall = std(reshaped_samples);
if ~isnan(Rhat)
stats.Rhat = setfield(stats.Rhat, fname, squeeze(Rhat));
end
squ_mean_overall = squeeze(st_mean_overall);
st_mean_size = size(squ_mean_overall);
if (length(st_mean_size) == 2) && (st_mean_size(2) == 1)
stats.mean = setfield(stats.mean, fname, squ_mean_overall');
else
stats.mean = setfield(stats.mean, fname, squ_mean_overall);
end
stats.std = setfield(stats.std, fname, squeeze(st_std_overall));
end
end
function DICstats = getDICstats(workingDir)
DICstats = [];
FIDlog = fopen([workingDir '\log.txt'],'r');
ct = 0;
test = 0;
endloop = 0;
while 1
tline = fgets(FIDlog);
if tline == -1; break; end
if endloop; break; end
if strfind(tline,'dic.set cannot be executed');
DICstats.error = 'DIC monitor could not be set by WinBUGS';
end
```

```
if size(tline,2)>6
if strcmp(tline(1:5),'total'); endloop = 1; end;
end
if size(tline,2)>2
if strcmp(tline(1:3),'DIC'); test = 1; end
end
if test
ct=ct+1;
if ct >= 4
A = sscanf(tline,'%*s %f %f %f %f');
S = sscanf(tline, '%s %*f %*f %*f %*f');
DICstats.S.(…) = A(…);
end
end
end
fclose(FIDlog)
end
function S=bugs2mat(file_ind,file_out,dir)
if nargin>2,
file_ind=[dir '/' file_ind];
file_out=[dir '/' file_out];
end
ind=readfile(file_ind);
data=load(file_out);
Nvars=size(ind,1);
S=[];
for k=1:Nvars
[varname,indexstr]=strtok(ind(k,:));
varname=strrep(varname,'.','_');
indices=str2num(indexstr);
if size(indices)~=[1 2]
error(['Cannot read line: [' ind(k,:) ']']);
end
sdata = size(data);
samples=data(indices(1):indices(2),2);
varname(varname=='[')='(';
varname(varname==']')=')';
leftparen=find(varname=='(');
outstruct=varname;
if ~isempty(leftparen)
outstruct=sprintf('%s(:,%s',varname(1:leftparen-1),varname(leftparen+1:end));
end
eval(['S.' outstruct '=samples;']);
end
end
```

```
function T=readfile(filename)
f=fopen(filename,'r');
if f==-1, fclose(f); error(filename); end
i=1;
while 1
clear line;
line=fgetl(f);
if ~isstr(line), break, end
n=length(line);
T(i,1:n)=line(1:n);
i=i+1;
end
fclose(f);
end
function [varargout] = process_options(args, varargin)
n = length(varargin);
if (mod(n, 2))
error('Each option must be a string/value pair.');
end
if (nargout < (n / 2))
error('Insufficient number of output arguments given');
elseif (nargout == (n / 2))
warn = 1;
nout = n / 2;
else
warn = 0;
nout = n / 2 + 1;
end
varargout = cell(1, nout);
for i=2:2:n
varargout{i/2} = varargin{i};
end
nunused = 0;
for i=1:2:length(args)
found = 0;
for j=1:2:n
if strcmpi(args{i}, varargin{j})
varargout{(j + 1)/2} = args{i + 1};
found = 1;
break;
end
end
if (~found)
if (warn)
warning(sprintf('Option "%s" not used.', args{i}));
```

```
args{i}
else
nunused = nunused + 1;
unused{2 * nunused - 1} = args{i};
unused{2 * nunused} = args{i + 1};
end
end
end
if (~warn)
if (nunused)
varargout{nout} = unused;
else
varargout{nout} = cell(0);
end
end
end
```

(9) *calcbugsmu.m*, *calcbugssigma.m*, *calcbugsmusigma.m*

```
function [res,pdfNormal1,pdfNormal2,pdfNormal3,pdfNormal4]=calcbugsmu(stats,N,mean2,std2,x,winbugs_model,flk,flc)
mean1l=winbugs_model.mean;
std1l=winbugs_model.std;
mean2l=mean2;
std2l=std2;
mean3l=stats.mean.mu;
std3l=stats.std.mu;
mean4l=mean3l;
if flk==0
if (N+1)<0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
std4l=std3l*sqrt(N+1);
end
elseif flk==1
if std1l==0 || std3l==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
lambda=1/std2l^2;
lambdan=1/std3l^2;
end
if  (lambdan+lambda)==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
lambdau=lambda*lambdan/(lambdan+lambda);
end
```

```
if (lambdau^-1)<0 || lambdau==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
std4l=sqrt(lambdau^-1);
end
end
if flc==0
mean(…)=mean(…);
std(…)=std(…);
elseif flc==1
mean(…)=exp(mean(…)+std(…)^2/2);
if (exp(2*mean(…)+std(…)^2)*(exp(std(…)^2)-1))<0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
std(…)=sqrt(exp(2*mean(…)+std(…)^2)*(exp(std(…)^2)-1));
end
end
pdfNormal(…)=normpdf(x,mean(…),std(…));
interval(…)=norminv([0.025,0.975],mean(…),std(…));
res.prior.par_mu.mean=mean1;
res.prior.par_mu.std=std1;
res.prior.par_mu.int=interval1;
res.likelihood.mean=mean2;
res.likelihood.std=std2;
res.likelihood.int=interval2;
res.posterior.par_mu.mean=mean3;
res.posterior.par_mu.std=std3;
res.posterior.par_mu.int=interval3;
res.posterior.pop.mean=mean4;
res.posterior.pop.std=std4;
res.posterior.pop.int=interval4;
end
```

```
function [res,pdfNormal1,pdfNormal2,pdfNormal3,f_x_4]=calcbugssigma(stats,N,mean2,std2,x,winbugs_model,flk,flc)
alfa1=winbugs_model.alfa;
beta1=winbugs_model.beta;
if winbugs_model.beta==0 || (winbugs_model.alfa/winbugs_model.beta^2)<0 || beta1==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
mean1l=winbugs_model.alfa/winbugs_model.beta;
std1l=sqrt(winbugs_model.alfa/winbugs_model.beta^2);
b1=1/beta1;
end
mean2l=mean2;
std2l=std2;
```

```
mean3l=stats.mean.tau;
std3l=stats.std.tau;
if std3l==0 || mean3l==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
alfa3=(mean3l/std3l)^2;
b3=std3l^2/mean3l;
end
if flk==0
mean4l=mean2l;
if std3l==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
alfa4=2*(mean3l/std3l)^2;
end
if ((mean3l/std3l^2)/((mean3l/std3l)^2))<0 || (alfa4/(alfa4-2))<0 || std3l==0 || (alfa4-2)==0 || ((mean3l/std3l)^2)==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
lambda4=(mean3l/std3l^2)/((mean3l/std3l)^2);
std4l=sqrt(lambda4)*sqrt(alfa4/(alfa4-2));
end
elseif flk==1
mean4l=mean2l;
alfa4=2*alfa1+N;
if 1/(alfa3*b3)<0 || (alfa3*b3)==0 || (alfa4/(alfa4-2))<0 || (alfa4-2)==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
lambda4=1/(alfa3*b3);
std4l=sqrt(lambda4)*sqrt(alfa4/(alfa4-2));
end
end
Y=gamrnd(alfa3,b3,1e6,1);
z=size(Y);
z=z(1);
for i=1:z
Y1(i,1)=1/sqrt(Y(i,1));
end
mean3l=mean(Y1);
std3l=std(Y1);
phat=gamfit(Y1);
alfa3=phat(1,1);
b3=phat(1,2);
if flc==0
mean(…)=mean(…);
std(…)=std(…);
```

```
elseif flc==1
mean(…)=exp(mean(…)+std(…)^2/2);
if (exp(2*mean(…)+std(…)^2)*(exp(std(…)^2)-1))<0
error('Error in WinBugs_Model Parameters'); % display on the matlab
else
std(…)=sqrt(exp(2*mean(…)+std(…)^2)*(exp(std(…)^2)-1));
end
end
pdfNormal1=gampdf(x,alfa1,b1);
pdfNormal2=normpdf(x,mean2,std2);
pdfNormal3=gampdf(x,alfa3,b3);
interval1=gaminv([0.025,0.975],alfa1,b1);
interval2=norminv([0.025,0.975],mean2,std2);
interval3=gaminv([0.025,0.975],alfa3,b3);
dist=ProbDistUnivParam('tlocationscale',[mean4,std4,alfa4]);
f_x_4=random(dist,5000,1);
flo=mean4-tpdf(0.025,alfa4)*std4;
fup=mean4+tpdf(0.025,alfa4)*std4;
interval4=[flo,fup];
res.prior.par_sigma.mean=mean1;
res.prior.par_sigma.std=std1;
res.prior.par_sigma.int=interval1;
res.likelihood.mean=mean2;
res.likelihood.std=std2;
res.likelihood.int=interval2;
res.posterior.par_sigma.mean=mean3;
res.posterior.par_sigma.std=std3;
res.posterior.par_sigma.int=interval3;
res.posterior.pop.mean=mean4;
res.posterior.pop.std=std4;
res.posterior.pop.int=interval4;
end
```

```
function [res,pdfNormal11,pdfNormal12,pdfNormal2,pdfNormal4,f_x_3,f_x_5]
=calcbugsmusigma(stats,N,mean2,std2,x,winbugs_model,flk,flc)
alfa11=winbugs_model.alfa;
beta11=winbugs_model.beta;
if winbugs_model.beta==0 || (winbugs_model.alfa/winbugs_model.beta^2)<0 || beta11==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
mean11l=winbugs_model.alfa/winbugs_model.beta;
std11l=sqrt(winbugs_model.alfa/winbugs_model.beta^2);
b11=1/beta11;
end
mean12l=winbugs_model.mean;
```

```
std12l=winbugs_model.std;
mean2l=mean2;
std2l=std2;
mean3l=stats.mean.mu;
std3l=stats.std.mu;
mean4l=stats.mean.tau;
std4l=stats.std.tau;
if std4l==0 || mean4l==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
alfa4=(mean4l/std4l)^2;
b4=std4l^2/mean4l;
end
mean5l=mean3l;
if flk==0
if std4l==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
alfa3=2*(mean4l/std4l)^2;
alfa5=2*(mean4l/std4l)^2;
lambda3=std3l^(-2)*alfa3/(alfa3-2);
end
if (alfa5+2)==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
lambda5=lambda3/(alfa5+2);
end
if (lambda5^(-1)*(alfa5/(alfa5-2)))<0 || lambda5==0 || (alfa5-2)==0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
std5l=sqrt(lambda5^(-1)*(alfa5/(alfa5-2)));
end
elseif flk==1
N0=winbugs_model.n0;
alfa3=2*alfa11+N;
alfa5=alfa3;
if (std3l^2*(N+N0+1))<0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
std5l=sqrt(std3l^2*(N+N0+1));
end
end
Y=gamrnd(alfa4,b4,1e6,1);
z=size(Y);
z=z(1);
```

```
Y1=zeros(z,1);
for i=1:z
Y1(i,1)=1/sqrt(Y(i,1));
end
mean4l=mean(Y1);
std4l=std(Y1);
phat=gamfit(Y1);
alfa4=phat(1,1);
b4=phat(1,2);
if flc==0
mean(…)=mean(…);
std(…)=std(…);
elseif flc==1
mean(…)=exp(mean(…)+std(…)^2/2);
if (exp(2*mean(…)+std(…)^2)*(exp(std(…)^2)-1))<0
error('Error in WinBugs_Model Parameters'); % display on Matlab
else
std(…)=sqrt(exp(2*mean(…)+std(…)^2)*(exp(std(…)^2)-1));
end
end
pdfNormal11=gampdf(x,alfa11,b11);
pdfNormal12=normpdf(x,mean12,std12);
pdfNormal2=normpdf(x,mean2,std2);
dist=ProbDistUnivParam('tlocationscale',[mean3,std3,alfa3]);
f_x_3=random(dist,5000,1);
flo_3=mean3-tpdf(0.025,alfa3)*std3;
fup_3=mean3+tpdf(0.025,alfa3)*std3;
pdfNormal4=gampdf(x,alfa4,b4);
dist=ProbDistUnivParam('tlocationscale',[mean5,std5,alfa5]);
f_x_5=random(dist,5000,1);
flo_5=mean5-tpdf(0.025,alfa5)*std5;
fup_5=mean5+tpdf(0.025,alfa5)*std5;
interval11=gaminv([0.025,0.975],alfa11,b11);
interval12=norminv([0.025,0.975],mean12,std12);
interval2=norminv([0.025,0.975],mean2,std2);
interval3=[flo_3,fup_3];
interval4=gaminv([0.025,0.975],alfa4,b4);
interval5=[flo_5,fup_5];
res.prior.par_sigma.mean=mean11;
res.prior.par_sigma.std=std11;
res.prior.par_sigma.int=interval11;
res.prior.par_mu.mean=mean12;
res.prior.par_mu.std=std12;
res.prior.par_mu.int=interval12;
res.likelihood.mean=mean2;
```

```
res.likelihood.std=std2;
res.likelihood.int=interval2;
res.posterior.par_mu.mean=mean3;
res.posterior.par_mu.std=std3;
res.posterior.par_mu.int=interval3;
res.posterior.par_sigma.mean=mean4;
res.posterior.par_sigma.std=std4;
res.posterior.par_sigma.int=interval4;
res.posterior.pop.mean=mean5;
res.posterior.pop.std=std5;
res.posterior.pop.int=interval5;
end
```

(10) *weibull.m*

```
function res=weibull(winbugs_model,data,flq)
datain=data; % Convert initial data
par=matrix_in(winbugs_model); % Read values from prior distribution
for i=1:2
[data,winbugs_model,parmean,parstd]=wbl(datain,winbugs_model,par,i); % Weibull conversion of registered data
parmean_upp(i)=parmean(i);
parstd_upp(i)=parstd(i);
res=winbugs(data,winbugs_model,flq); % Run Winbugs analysis
[data_up,p]=matrix_up(winbugs_model,res,i); % Generation of normal data for posterior population
p_upd(:,i)=p(:,i);
data_upp{i}(:,:)=data_up{i}(:,:);
end
n1=0;
for i=1:2
p=size(data_upp{i}(:,:)); % Determine size of posterior parameter data
n=p(2);
if n>=n1
n1=n;
end
end
sumalfa=0;
sumbeta=0;
for j = 1:n1
NS=winbugs_model.n; % Determine number of computed values
Q{j}(:,:)=wblrnd(data_upp{…}(:,i),data_upp{…}(:,i),1,NS); % Generation of posterior population data
mn{j}(:,:)=wblfit(Q{j}(:,:)); % Curve fitting with Weibull distribution of posterior population
sumalfa=sumalfa+mn{j}(…);
sumbeta=sumbeta+mn{j}(…);
end
% Determine Weibull parameters for prior, likelihood and posterior
```

```
par(…)=winbugs_model.meana;
par(…)=winbugs_model.meanb;
res.prior.pop.alfa=par(…);
res.prior.pop.beta=par(…);
res.likelihood.pop.alfa=parmean_upp(…);
res.likelihood.pop.beta=parmean_upp(…);
res.posterior.pop.alfa=sumalfa/n1;
res.posterior.pop.beta=sumbeta/n1;
x=0:0.1:1000;
% Determine distribution for prior, likelihood and posterior
pdfNormal1=wblpdf(x,res.prior.pop.alfa,res.prior.pop.beta);
[a1,b1]=wblstat(res.prior.pop.alfa,res.prior.pop.beta);
res.prior.pop.mean=a1;
res.prior.pop.std=sqrt(b1);
interval1=wblinv([0.025,0.975],res.prior.pop.alfa,res.prior.pop.beta);
res.prior.pop.int=interval1;
% Likelihood
% Posterior
% Plot Weibull distribution
plotbugswbl(pdfNormal1,pdfNormal2,pdfNormal3,x);
% Results for mean and std of each parameter
res.prior.alfa.(…)=winbugs_model.(…);
% Zeros Restriction
X=wblrnd(res.prior.pop.alfa,res.prior.pop.beta,1e6,1);
censored=(X<0);
p=wblfit(X,censored);
p(…)=p(…);
[a,b]=wblstat(…);
res.prior.pop.truncated.(…)=p(…);
X=wblrnd(res.likelihood.pop.alfa,res.likelihood.pop.beta,1e6,1);
censored=(X<0);
p=wblfit(X,censored);
p(…)=p(…);
[a,b]=wblstat(…);
res.likelihood.pop.truncated.(…)=p(…);
X=wblrnd(res.posterior.pop.alfa,res.posterior.pop.beta,1e6,1);
censored=(X<0);
p=wblfit(X,censored);
p(…)=p(…);
[a,b]=wblstat(…);
res.likelihood.pop.truncated.(…)=p(…);
% Reliability Value
n=((…)-0.005)/0.005+1;
X_Prior=cell(2);
for i=1:n
```

```
P=0.005*i;
a=res.prior.pop.alfa;
b=res.prior.pop.beta;
X_Prior{1}(i,1)=P;
X_Prior{1}(i,2)=wblinv(P,a,b);
a=res.prior.pop.truncated.alfa;
b=res.prior.pop.truncated.beta;
X_Prior{2}(i,1)=P;
X_Prior{2}(i,2)=wblinv(P,a,b);
end
% Likelihood
% Posterior
end
end
```

(11) *matrix_in.m*

```
function par=matrix_in(winbugs_model)
% Read values from prior distribution
par(…,...)=winbugs_model.(…);
end
```

(12) *wbl.m*

```
function [data,winbugs_model,parmean,parstd]=wbl(datain,winbugs_model,par,i)
% Compute prior values for mean of distribution alfa and beta
winbugs_model.mean=par(…);
% Compute prior values for standard deviation of distribution alfa and beta
winbugs_model.std=par(…);
% Determine n0 values
winbugs_model.n0=par(…);
% Curve fitting with Weibull distribution of likelihood
[meanh,inth]=wblfit(datain);
parmean(i)=meanh(…);
x1=abs(inth(…)-meanh(…));
x2=abs(inth(…)-meanh(…));
parstd(i)=(x1+x2)/2;
% Data generation according to Normal distribution
NS=winbugs_model.n;
data=normrnd(parmean(i),parstd(i),1,NS);
end
```

(13) *matrix_up.m*

```
function [data_up,p]=matrix_up(winbugs_model,res,i)
```

```
% Compute population distribution for each parameter
p(…)=res.posterior.pop.(…);
% Generation of Normal data values for parameter
NS=winbugs_model.n;
data_up{i}(:,:)=normrnd(p(…),p(…),1,NS);
end
```

Appendix D: WinBugs Models

According to what was previously indicated at chapter four, it is presented the three developed models to run in WinBugs® software [111], within the Bayesian inference procedure [15]. The following situations were, respectively, considered:

(1) unknown mean (μ) and known variance (σ^2) – *mu.txt*

```
model
{
for(i in 1:N){
income[i]~dnorm(mu,tau)}
mu~dnorm(mean,taum)
tau<-pow(sigma,-2)
}
```

(2) unknown mean (μ) and variance (σ^2) with Jeffrey's prior – *mu_tau_jeff.txt*

```
model
{
for(i in 1:N){
income[i]~dnorm(mu,tau)}
mu~dnorm(mean,taum)
tau~dgamma(alfa,beta)
}
```

(3) unknown mean (μ) and variance (σ^2) with conjugate prior – *mu_tau_conj.txt*

```
model
{
for(i in 1:N){
income[i]~dnorm(mu,tau)}
mu~dnorm(mean,tau.mu)
tau~dgamma(alfa,beta)
tau.mu<-tau*N0
}
```