ON RESOURCE COMPLEMENTARITY IN ACTIVITY NETWORKS - FURTHER RESULTS

Helder C. Silva[1], <u>Anabela P. Tereso</u>[2], José A. Oliveira[3]

[1] IFAM – Instituto Federal de Educação Tecnológica do Amazonas, Manaus, Brazil, helder@ifam.edu.br
[2] University of Minho, Guimarães, Portugal, anabelat@dps.uminho.pt
[3] University of Minho, Guimarães, Portugal, zan@dps.uminho.pt

Abstract
We address the issue of optimal resource allocation, and more specifically, the analysis of complementarity of resources (primary resource or *P*-resource and supportive resource or *S*-resource) to activities in a project. We developed a mathematical model capable of determining the ideal mixture of resources allocated to the activities of a project, such that the project is completed with minimal cost. This problem has a circularity issue that greatly increases its complexity. We have developed a procedure which we illustrate by application to small instances of the problem, using complete enumeration over the decision space. The development of a more computationally efficient procedure awaits the second phase of this study.
Keywords: Project Management, Resource Allocation, Complementarity of Resources, Activity Network.

1. Introduction

This paper is concerned with the optimal resource allocation in activity networks under conditions of resource complementarity. The concept of complementarity which has been discussed from an economic point of view [1] can be incorporated into the engineering domain as an enhancement of the efficacy of a "primary" resource (*P*-resource) by adding to it other "supportive" resources (*S*-resources). Aspects related to performance improvement, short duration, quality improvement have been presented by Silva *et al.* [2] as well as the effect of the "supportive" resource for project cost.

The issue may be phrased as follows: how much of the *P*-resources and additional support to them in the form of *S*-resources should be allocated to project activities to achieve improved results most economically?

More will be said about the assumptions made in the appropriate sections of the paper, and all the concepts mentioned shall be made precise in our specification of the mathematical model of the problem. We illustrate the mathematical model using two small project networks.

2. Problem description

Consider a project network in the activity-on-arc (AoA) representation: $G = (N, A)$ with the set of nodes $|N| = n$ (representing the "events") and the set of arcs $|A| = m$ (representing the "activities"). In general each activity requires the simultaneous use of several resources [3][4][5].

There is a set of "primary" resources, denoted by P, with $|P| = \rho$. Typically, a primary resource has a capacity of several units (say workers, m/c's, processors; etc.) [6]. Additionally, there is a pool of "support" resources, denoted by S, with $|S| = \sigma$ (such as less-skilled labor, or computers and electronic devices; etc.) that may be utilized in conjunction with the primary resources to enhance their performance.

The number of support resources varies with the activity and the primary resources required for its execution. The relevance of each to the *P*-resources may best be represented in matrix format as shown in Table 1 ($\emptyset$ indicates inapplicability). An entry $v(r_p, s_q) \neq \emptyset$, measures the enhancement offered by *S*-resource s_q to *P*-resource r_p.

Table 1: Applicability and impact of support resources

$\downarrow P\text{-}Res/S\text{-}Res \rightarrow$	s_1	$\cdots$	s_q	$\cdots$	s_σ
r_1	$v(1,1)$	$\cdots$	$\emptyset$	$\cdots$	$v(1,\sigma)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
r_p	$\emptyset$	$\cdots$	$v(p,q)$	$\cdots$	$v(p,\sigma)$
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
r_ρ	$v(\rho,1)$	$\cdots$	$v(\rho,q)$	$\cdots$	$\emptyset$

The impact on the *P*-resource is evaluated in the following way: $0 < v(r_p, s_q) \leq 1$, indicates the fraction by which the *S*-resource s_q improves the performance of *P*-resource r_p. Typically, $v(r_p, s_q) \in [0.1, 0.5]$. If only one unit of *S*-resource is used, the performance of the allocation of *P*-resource r_p to activity *a*, which is denoted by $x_a(r_p)$, is augmented to,

$$x_a(r_p, s_p) = x_a(r_p) + v(r_p, s_p) \tag{1}$$

For the sake of simplicity, we make the following assumptions. First we assume that the impact of the *S*-resources is additive: if a subset $\{s_q\}_{q=1}^{\sigma}$ of the *S*-resources is used in support of *P*-resource r_p in activity *a*, and only one unit of each *S*-resource is used, then the performance of the former is enhanced to,

$$x_a\left(r_p, \{s_q\}_{q=1}^{\sigma}\right) = x_a(r_p) + \sum_{q=1}^{\sigma} v(r_p, s_q) \tag{2}$$

The primary resource $r_p \in P$ would accomplish activity *a* in time $y_a(r_p)$. If it is enhanced by the addition of one *S*-resource s_q then its processing time decreases to $y_a(r_p, s_q)$, with $y_a(r_p, s_q) < y_a(r_p)$. The issue now is to express the functional relationship between the resource allocation (both primary and support) and the activity duration.

Let $w_a(r_p)$ denote the work content of activity *a* for *P*-resource *r*. Let $x_a(r_p)$ denote, as suggested above, the amount of primary resource r_p allocated to activity *a*. The duration of activity *a* when using resource r_p is given by [7],

$$y_a(r_p) = \frac{w_a(r_p)}{x_a(r_p)} \tag{3}$$

If a support resource s_q is added to the primary resource r_p then the duration becomes,

$$y_a(r_p, s_q) = \frac{w_a(r_p)}{x_a(r_p, s_q)} \tag{4}$$

To illustrate these concepts, suppose an activity has work content $w_a(r_p) = 36$ man-days. Further, assume the *S*-resource s_q yields a rate $v(r_p, s_q) = 0.50$.

If $x_a(r_p) = 0.85$ then in the absence of the support resource the duration of the activity would be

$$y_a(r_p) = 36/0.85 = 42.35 \text{ days.}$$

But in the presence of the *S*-resource the duration would be only

$$y_a(r_p, s_q) = 36 / ((0.85 + 0.5) = 26.67 \text{ days,}$$

representing a saving of approximately 37%.
If $x_a(r_p) = 1.5$ then in the absence of the *S*-resource the duration of the activity would be

$$y_a(r_p) = 36/1.5 = 24 \text{ days.}$$

But in the presence of the *S*-resource the duration would be only

$$y_a(r_p, s_q) = {36}/{(1.5 + 0.5)} = 18 \text{ days,}$$

representing a saving of 25%.
An activity normally requires the simultaneous utilization of more than one P-resource for its execution. The problem then becomes:
"At what level should each resource be utilized and which supportive resource(s) should be added to it (if any) in order to optimize a given objective?"
Recall that the processing time of an activity is given by the maximum of the durations that would result from a specific allocation to each resource (see a previous discussion on the evaluation of the duration considering multiple resources in [3][4][5]).

$$y(a) = \underset{all\ r_p}{max} \{y_a(r_p)\} \tag{5}$$

To better understand this representation, consider the project on Figure 1 and Figure 2, AoN and AoA respectively. There, the reader will find a network, formed by three activities, 1, 2 and 3, for which we will assume that the project requires the utilization of four *P*-resources; not all resources are required by all the activities.
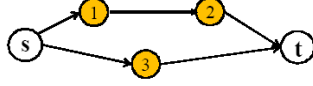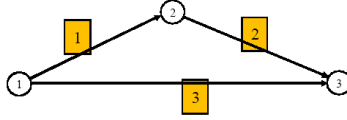

Figure 1: Project with 3 activities AoN.

Figure 2: AoA representation.

Table 2: Work content (in man-days) of the activities of project 1.

P-resource →	1	2	3	4
↓ Activity/Availability→	2	1	3	2
A1	16	0	12	12
A2	0	7	0	8
A3	20	22	0	0

Table 2 is to be read as follows. There are two units available of resources #1 & #4; one unit of resource #2 and 3 units of resource #3. Activity 1 requires 16 man-days of resource #1 and 12 man-days of each of resources #3 and #4. It does not require resource #2. Etc.

The relevance and impact of the support resources are represented in Table 3, which may be read as follows: *S*-resources 1 and 2 have availability of one unit each. *S*-resource 1 can support *P*-resources 1 & 3 and *S*-resource 2 can support *P*-resources 1 & 2; no support is available for *P*-resource 4.

Table 3: The P-S matrix: Impact of *S*-resources on *P*-resources.

	P-Resource →	*1*	*2*	*3*	*4*
↓S-Resource	*↓ Availability*				
1	*1*	0.25	ϕ	0.25	ϕ
2	*1*	0.15	0.35	ϕ	ϕ

With little additional data processing, the problem can be enriched with the inclusion of the cost of the resource utilization at each level. Then in each cell in both the primary and secondary resource tables there shall be added the marginal cost for the resource per unit time. If the project gains a bonus for early completion and incurs a penalty for late completion then one can easily include such costs in the criterion function.

At time 0 we may initiate both activities A1 and A3 because their required *P*-resources are available (A1 requires *P*-resources 1, 3 & 4 and A3 requires *P*-resources 1 & 2). Assume for the moment that no support resource is allocated to either activity. Further, suppose that each unit of the primary resource is devoted to its respective activity at level 1; i.e.,

$$x_1(r_1) = 1 = x_1(r_3) = x_1(r_4) \tag{6}$$
$$x_3(r_1) = 1 = x_3(r_2) \tag{7}$$

3

Observe that the *P*-resource availabilities have been respected: the two units of *P*-resource 1 have been equally divided between the two activities; *P*-resource 2 is not required by A1 and the unit available is allocated to A3, *P*-resources 3 & 4 are required only by A1. The *P*-resource allocation would look as shown in Table 4.

Table 4: The *P*-resources allocation at time 0.

Activity	P-Resource			
	1	2	3	4
A1	1	0	1	1
A3	1	1	0	0
Total Allocation	2	1	1	1

The durations of the two activities shall be:

$$A1: y(1) = max\left\{\frac{16}{1}, \frac{12}{1}, \frac{12}{1}\right\} = 16 \text{ days} \tag{8}$$

$$A3: y(3) = max\left\{\frac{20}{1}, \frac{22}{1}\right\} = 22 \text{ days} \tag{9}$$

At time t = 16 activity A1 completes processing and A2 becomes sequence feasible. Unfortunately it cannot be initiated because *P*-resource 2, of which there is only one unit, is committed to A3 which is still on-going. Therefore activity 2 must wait for the completion of A3, which occurs at t = 22. When initiated at resource levels

$$x_2(r_2) = 1 = x_2(r_4)$$

it will consume $y(2) = max\left\{\frac{7}{1}, \frac{8}{1}\right\} = 8$ days to complete.

The project duration (time of completion of node 3 in the AoA network) would be

$$t_3 = 22 + 8 = 30 \tag{10}$$

If the due date of the project were specified at $T_s = 24$, the project would be 6 days late.

Suppose that at the start of the project both support resources were allocated to activity 3 as follows $s_1 \rightarrow r_1$ and $s_2 \rightarrow r_2$; then

$$x_3(r_1, s_1) = 1 + 0.25 \tag{11}$$
$$x_3(r_2, s_2) = 1 + 0.35 \tag{12}$$

The duration of the A3 would change to

$$y(3) = max\left\{\frac{20}{1.25}, \frac{22}{1.35}\right\} \approx 16.30 \text{ days} \tag{13}$$

At t = 16.30 activity 2 can be initiated because primary resource 2 would be freed. If we continue with $x_2(r_2) = 1 = x_2(r_4)$ it will consume the same 8 days to complete and the project duration would be

$$t_3 = 16.30 + 8 = 24.30 \tag{14}$$

The project is almost on time!

Whether or not such allocation of the support resources is advisable shall depend on the relative costs of the *S*-resources and tardiness. In fact, again depending on the relative costs, it may be advisable to have allocated *S*-resource 1 to activity 1 when it is initiated at time 0 and, when completed, continue as above with activity 2, since the gain in the project completion time may secure some bonus payment that would more than offset the cost of the added support. It is also possible to allocate more than one *S*-resource to complement the *P*-resources in some activities. All these, and other, possibilities should be resolved by a formal mathematical model.

3. Mathematical model

We assume that all costs are linear or piece-wise linear in their argument.

Let:

C^k: the kth uniformly directed cutset *(udc)* of the project network that is traversed by the project progression; $k = 1, \dots, K$.

$x_a(r_p)$: level of allocation of (primary) resource r_p to activity a (assuming integer values from 1 to $Q_P(p)$ if the activity needs this resource).

$x_a^{r_p}(s_q)$: level of allocation of secondary resource s_q to primary resource r_p in activity a (assuming integer values from 0 to $Q_S(q)$).

$x_a\left(r_p, \{s_q\}_{q=1}^{\sigma}\right)$: total allocation of resource r_p (including complementary resources) to activity a.

$v(r_p, s_q)$: degree of enhancement of P-resource r_p by S-resource s_q.

$w_a(r_p)$: work content of activity a when P-resource r_p is used.

$y_a(r_p, \{s_q\}_{q=1}^{\sigma})$: duration of activity a imposed by primary resource r_p (including enhancement by complementary resources).

$y(a)$: duration of activity a (considering all resources).

ρ : number of primary resources, $\rho = |P|$.

σ : number of secondary resources, $\sigma = |S|$.

$Q_P(p)(Q_S(q))$: capacity of P-resource r_p (S-resource s_p) available.

γ_p : marginal cost of P-resource r_p.

γ_q : marginal cost of S-resource s_q.

γ_E : marginal gain from early completion of the project.

γ_L : marginal loss (penalty) from late completion of the project.

t_i : time of realization of node i (AoA representation), where node 1 is the "start node" of the project and node n its "end node".

T_s : target completion time of the project (due date).

$c_R(a, r_p)$: cost of resources for activity a resource r_P (including complementary resources).

$c_R(a)$: cost of resources for activity a (includes all resources).

e : earliness.

d : tardiness (delay).

c_E : cost of earliness.

c_T : cost of tardiness.

c_{ET}: cost of earliness and tardiness.

TC: total cost.

The constraints are enumerated next. To avoid confusion with node designation we refer to an activity as *"a"* and to a node as i or j. The notation $a \equiv (i, j)$ means that activity a is represented by arc (i, j).
Respect precedence among the activities:

$$t_j \geq t_i + y(a), \quad \forall a \equiv (i, j) \in A \tag{15}$$

Define total allocation of resource r_p (including complementary resources) in activity a,

$$x_a\left(r_p, \{s_q\}_{q=1}^{\sigma}\right) = x_a(r_p) + \sum_{q=1}^{\sigma} v(r_p, s_q) * x_a^{r_p}(s_q) \tag{16}$$

Define the duration of each activity when using each P-resource; then define the activity's duration as the maximum of individual resource durations:

$$y_a(r_p, \{s_q\}_{q=1}^{\sigma}) = \frac{w_a(r_p)}{x_a\left(r_p, \{s_q\}_{q=1}^{\sigma}\right)} \tag{17}$$

$$y(a) = \underset{all\ r_p}{max}\left\{y_a(r_p, \{s_q\}_{q=1}^{\sigma})\right\} \tag{18}$$

Respect the *P*-resource availability at each udc^1 traversed by the project in its execution,

$$\sum_{a \in C^k} x_a(r_p) \leq Q_P(p), \qquad \forall p \in P \tag{19}$$

in which $Q(p)$ is the capacity (i.e., availability) of *P*-resource r_p (in the three activities example given above, the vector $Q(P) = (2, 1, 3, 2)$).
Respect also the *S*-resources availability, considering again the current *udc*,

$$\sum_{a \in C^k} x_a^{r_p}(s_q) \leq Q_S(q) \quad \forall q \in S \tag{20}$$

in which $Q_S(q)$ is the capacity of *S*-resource s_q (in the three-activities example given above, the vector $Q_S(q) = (1, 1)$). Note that the requirement that an *S*-resource is applied only to its relevant *P*-resources is taken care of in the *P-S* matrix (see Table 3); what this constraint accomplishes is to limit its use to each resource's total availability.

The difficulty in implementing these constraints stems from the fact that we do not know a priori the identity of the *udc*'s that shall be traversed during the execution of the project, since that depends on the resource allocations (both the *P*- and *S*-resources). A circularity of logic is present here: the allocation of the resources is bounded by their availabilities at each *udc*, but these latter cannot be known except after the allocations have been determined. In this paper we propose a way to solve this circularity for small instance problems. We use complete enumeration of all decision variables, but we restrict these combinations to the possible states, considering all the precedence and resource constraints of the problem and for each combination, we go through all the states till the end of the project (see section 4 below).

Define earliness and tardiness by,

$$e \geq T_s - t_n \tag{21}$$

$$d \geq t_n - T_s \tag{22}$$

$$e, d \geq 0 \tag{23}$$

The criterion function is composed of two parts: the cost of use of the *P*- and *S*-resources, and the gain or loss due to earliness or tardiness, respectively, of the project completion time t_n relative to its due date.
For simplicity, we make the following two assumptions:
(i) The cost of resource utilization is quadratic in the resource allocation for the duration of the activity [7], which renders the cost linear in work content (recall that the work content is assumed a known constant),

$$c_R(a, r_p) = \left(\gamma_p * x_a(r_p) + \gamma_q * \sum_{q=1}^{\sigma} x_a^{r_p}(s_q) \right) * w(a, r_p) \tag{24}$$

$$c_R(a) = \sum_{\text{all } r_p} c_R(a, r_p) \tag{25}$$

(ii) The earliness-tardiness costs are linear in their respective marginal values, as shown in eq. (26)

$$c_{ET} = c_E + c_T = \gamma_E \cdot e + \gamma_L \cdot d \tag{26}$$

The desired objective function may be written simply as

$$\min TC = \sum_{a \in A} c_R(a) + c_{ET} \tag{27}$$

---

[1] The acronym *udc* stands for 'uniformly directed cutset', which is a cutset of the graph in which all arrows are directed from the subset of nodes H which contains the origin node, to the complementary subset $\overline{H} = N - H$ which contains the terminal node.

4. Description of the procedure adopted

The procedure we used to solve this problem begins by analyzing the network and the resource requirements and constructing the state space. To this end, we introduce some notation which is defined for a fixed allocation vector $X = \left\{ \{x_a(r_p)\}, \{x_a^{r_p}(s_q)\} \right\}$. We have $G = (N, A)$ as the project network. We assume that the project starts at time $t_0 = 0$ and ends at time t_n, a dummy node that signals the completion of all the original activities of the project. During the course of the project execution, each activity can be in one and only one of the following three states (see a similar approach in [8]):

   (i) *Active*: an activity a is active at time t if it is being executed at time t.

   (ii) *Dormant*: an activity a is dormant at time t if it has finished but there is at least one unfinished activity that ends at the same node as a.

   (iii) *Idle*: an activity a is called idle at time t if it is neither active nor dormant at time t: the activity is either completed or is yet to be started.

4.1 Application of the procedure to an activity network with 3 activities

Consider the small network represented in Figure 2 (it will be called network 1 from now on). If we did not had resource restrictions, the project would initiate with activities 1 and 3 active, and after activity 1 finished, activity 2 could be initiated and be active at the same time as activity 3, if activity 3 hadn't finished yet, by that time. In this case, we would have a state with activities 2 and 3 active. But, due to resource restrictions, this state can never happen. Observe that activities 2 and 3 both need *P*-resource 2 that has only 1 unit available. Activity 2 has to wait until activity 3 finishes, to begin executing. So we will have the state space represented in Figure 3.
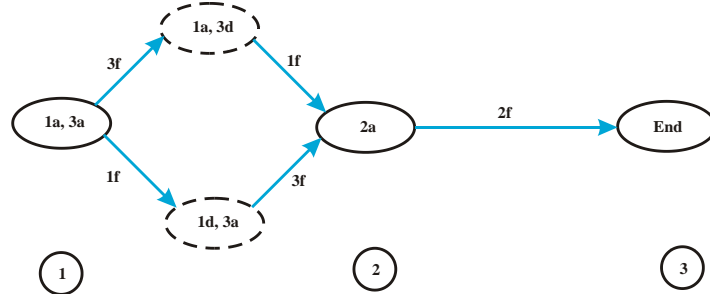

Figure 3: State Space diagram for the activity network 1.

The project begins in state 1 with both activities 1 and 3 active. There are enough *P*-resources to initiate these activities. We assume that we need at least 1 unit of each required *P*-resource to execute the activity.

Based on the precedence relationships, after activity 1 finishes, we could initiate activity 2. However, as explained before, due to resource restrictions, activities 2 and 3 cannot be initiated at the same time. So, activity 2 must wait until both activities 1 and 3 finish. We also have some intermediate states that do not need any decision. In state 1, if activity 1 finishes first, we would be in a state where activity 1 becomes dormant, but due to shortage of resource 2 we are waiting for the completion of activity 3. We represented these states in the diagram using dashed lines. Only after activity 3 finishes, will we reach state 2, where we have to decide how much resource to allocate to activity 2.

The next step is to analyze the possible values of each decision variable, considering the resources and the precedence constraints.

The decision variables for network 1 are represented in the Table 5 with the possible values they can take.

Table 5: Decision Variables and Possible Values

Decision Variable	Possible Values	Decision Variable	Possible Values
x_{11}	1	x_{222}	0..2
x_{111}	0..1	x_{24}	1..2
x_{112}	0..2	x_{31}	1
x_{13}	1..3	x_{311}	0..1
x_{131}	0..1	x_{312}	0..2
x_{14}	1..2	x_{32}	1
x_{22}	1	x_{322}	0..2

The meaning of the indexes is as follows:

x_{aps}
- a: activity.
- p: primary resource.
- s: supportive resource.

We consider that the primary resource is mandatory and the supportive resource is optional.

x_{11} and x_{31} can only have the value 1 because activities 1 and 3 will be executing at the same time, with each demanding at least one unit of the resource and we only have 2 units of resource 1.

The supportive resource 1 has one unit available, so we can use 0 or 1 units of this resource in each activity, considering that, at the same time, the total allocation, for all the activities, should not exceed 1. For supportive resource 2, we can use 0, 1 or 2 units to support an activity, with similar overall constraints.

We used the same reasoning to fill the rest of the table.

To reach a solution, we proceed in the following way:

For each of the possible combinations of the decision variables, restricted as before, we use a tridimensional structure to save the information as follows:

$$X[a, r, k]$$

Where,

- X: Structure that will contain the information for each combination of resources considered.
- a: Represents the activity;
- r: Represents the (primary) resource;
- k: Represents the kind of information stored.

If $k = 1 \Rightarrow X[a, r, 1]$ will have the quantities of the primary resource of a combination.

If $k = 2 \Rightarrow X[a, r, 2]$ will have the quantities of the S-resource 1.

If $k = 3 \Rightarrow X[a, r, 3]$ will have the quantities of the S-resource 2.

If $k = 4 \Rightarrow X[a, r, 4]$ will have the total quantity of resources (P- and S- resources) for each pair activity – P-resource.

If $k = 5 \Rightarrow X[a, r, 5]$ will have the corresponding resource cost.

If $k = 6 \Rightarrow X[a, r, 6]$ will have the duration of each pair activity – P-resource.

We will exemplify the evaluation of the values of X for one combination of the resources. Consider the following combination.

$X(k = 1)$:

↓Activity/P-Resource→	1	2	3	4
1	1	x	2	2
2	x	1	x	2
3	1	1	x	x

$X(k = 2)$:

↓Activity/P-Resource→	1	2	3	4
1	0	x	1	x
2	x	x	x	x
3	0	x	x	x

$X(k = 3)$:

↓Activity/P-Resource→	1	2	3	4
1	0	x	x	x
2	x	0	x	x
3	0	1	x	x

To evaluate the total resource ($k = 4$) we have to consider the contribution that a supportive resource can give to a primary resource. For the example under discussion, based on Table 3, we have:

$$X[a, r, 4] = X[a, r, 1] + X[a, r, 2] * v[1, r] + X[a, r, 3] * v[2, r] \tag{28}$$

So,

$X(k = 4)$:

↓Activity/P-Resource→	1	2	3	4
1	1	x	2.25	2
2	x	1	x	2
3	1	1.35	x	x

To evaluate the resource cost we have to multiply the quantity of the resource allocated with the unitary resource cost, which is different for the primary resource and supportive resources, and also, with the work content of each pair activity – P-resource.

The expression will be:

$$X[a,r,5] = \left(X[a,r,1] * \gamma_p + X[a,r,2] * \gamma_q + X[a,r,3] * \gamma_q \right) * w(a,r) \qquad (29)$$

Where:
γ_p: is the unitary cost of primary resource ($= 4$);
γ_q: is the unitary cost of supportive resource ($= 1$);
$w(a,r)$: is the work-content of activity a under P-resource r.

Considering the work content (see Table 2), the results for ($k = 5$) will be:

$X(k = 5)$:

↓Activity/P-Resource→	1	2	3	4
1	64	0	108	96
2	0	28	0	64
3	80	110	0	0

Finally, we have to evaluate the activities durations, using eq. (17).

$y_a(r_p, \{s_q\}_{q=1}^{\sigma})$ will be stored in $X[a,r,6]$.

In this case, we will have:

$$X[a,r,6] = \frac{w(a,r)}{X[a,r,4]} \qquad (30)$$

So,

$X(k = 6)$:

↓Activity/P-Resource→	1	2	3	4
1	16	x	5.3	6.0
2	x	7.0	x	4.0
3	20	16.3	x	x

After this, we can evaluate the duration of each activity, considering all the resources. The duration will be the maximum of the durations imposed by each of the resources separately. We represent the durations by $y(a)$.

Considering $X[a,r,6]$, we have:

$y(1) = \max(16.0, 5.3, 6.0) = 16.0$
$y(2) = \max(7.0, 4.0) = 7.0$
$y(3) = \max(20.0, 16.3) = 20.0$

Initially the state will be equal to 1. To proceed we have to evaluate if the combination is valid, considering the total availability of the resources. In state 1, for example, the combination is valid if the sum of the resources used in activities 1 and 3 does not exceed the total availability. The allocation of primary resource 1 is already valid, as it was decided on Table 5. We have to be sure that both supportive resources are correctly used. We have to ensure

that, for the supportive resource 1:

$$x_{111} + x_{131} + x_{311} \leq 1 \tag{31}$$

And for the supportive resource 2:

$$x_{112} + x_{312} + x_{322} \leq 2 \tag{32}$$

These constraints are respected, so the next state will be state 2, where there are no restrictions of this kind as we only have one activity active, so we can proceed to state 3, the end of the project.

If a combination reaches the end of the project, it means that it is valid, so we save that combination and evaluate the completion time of the project using CPM^2 and the information about the resources constraints. In the case of Network 1, we know that activity 2 only starts after 1 and 3 have finished. So, the project starts at $t_1 = 0$. t_2, the start of activity 2 will be the maximum between the duration of activity 1 and 3, being t_3 given by the result of t_2 plus the duration of activity 2. For the example combination, we will have:

$t_1 = 0$
$t_2 = \max(16, 20) = 20$
$t_3 = t_2 + y(2) = 20 + 7 = 27.$

The due date of the project is $T_s = 24$, so we will be 3 days late. The cost of tardiness will be,

$C_T = 3 * \gamma_L = 3 * 60 = 180$ as C_E=0, C_{ET}=180.

The resource cost in the sum of all resource costs for all pairs activity – P-resource. In this case, we have (using $X[a, r, 5]$):

$C_R = \ \ 64 + 108 + 96 + 28 + 64 + 80 + 110 = 550.$

And the total cost TC will be, in this case,

$TC = C_{ET} + C_R = 180 + 550 = 730.$

After repeating this procedure for all possible combinations, we came up with a solution of this network:

Table 6: The best combination of Network 1 with 3 activities.

x11	x111	x112	x13	x131	x14	x22	x222	x24	x31	x311	x312	x32	x322	Tn	C_E	C_T	C_{ET}	C_R	TC
1	0	0	1	0	1	1	2	2	1	1	0	1	1	20.41	-143.44	0	-143.44	476	332.56

4.2 Application results for an activity network with 5 activities.

We will apply the algorithm to a new activity network. We will consider an activity network with 5 activities, called network 2 from now on.

To better understand the solution for network 2, we will present on Figure 4 the precedence diagram and the state space as in the first case. The stages in red color indicate the states we must consider for the final analysis.

---

2 CPM – Critical Path Method

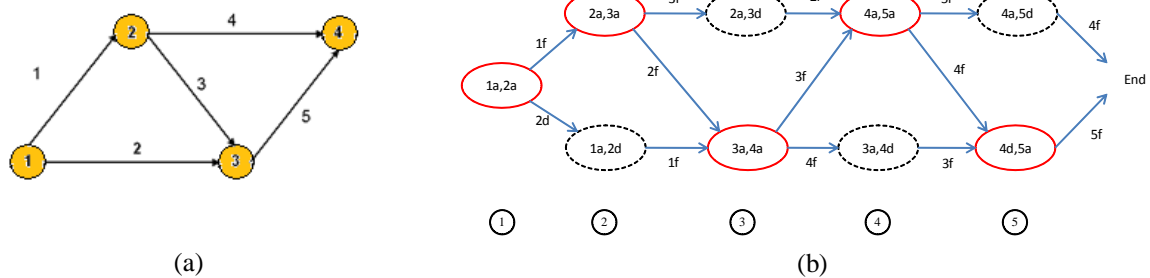(a)   (b)

Figure 4: Activity network 2. a) The precedence diagram AoA and b) the state space diagram.

For network 2, we considered that the work content is giving by Table 7.

Table 7: Work content (in man-days) of the activity network 2.

P-resource →	1	2	3	4
↓ Activity/Availability→	2	1	3	3
A1	16	0	12	12
A2	0	7	10	8
A3	20	0	22	0
A4	0	7	0	8
A5	20	0	16	0

Applying the same analysis as in network 1, we reach the following solution (see Table 8).

Table 8: The best combination for network 2 with 5 activities.

x11	x111	x112	x13	x131	x14	x22	x222	x23	x231	x24	x31	x311	x312	x33	x331	x42	x422	x44	x51	x511	x512	x53	x531
2	1	2	2	0	2	1	0	1	0	1	2	0	0	2	1	1	0	1	2	1	1	2	0

Tn	CE	CT	CET	CR	TC
24.61	0.00	36.47	36.47	1214.00	1250.47

4. Conclusions

The goal of this paper was to provide a formal model to some unresolved issues in the management of projects, especially as related to the utilization of supportive resources, and to its implementation. The relevance of the problem is the opportunity to shape a system that allows not only that we improve the allocation of often scarce resource(s), but also result in reduced uncertainties within the projects, combined with increased performance and lower project costs. The model was first presented in [2] but there remained its implementation and application to some project networks, to demonstrate its validity. In this paper we presented the procedure developed to solve the mathematical model and we applied it to two simple networks, obtaining the desired results, through an initial implementation in C. There still remains the implementation of the model in an easy-to-use computer code that renders it practically usable for networks of realistic size. This should be a general computer code, which will be capable calculating the solution for any activity network.

Considering the feasibility of the model proposed, we believe it can provide to user a new option to plan and to determine the best combination of resources and the lowest project cost, pushing the planning phase and increase the estimation ability of the companies.

References

[1]   M. Kremer, The O-Ring Theory of Economic Development, The Quartely Journal of Economics, Vol. 108, pp. 551-575, 1993.

[2]   H.C. Silva, A.P. Tereso and J. A. Oliveira, On Resource Complementarity in Activity Networks, 3rd International Conference on Information Systems, Logistics and Supply Chain, Casablanca - Marroco, April 14-16, 2010.

[3]   A. Tereso, M. Araújo, R. Moutinho and S. Elmaghraby, Project management: multiple resources allocation, International Conference on Engineering Optimization, Rio de Janeiro - Brazil, 2008.

[4]   A. Tereso, M. Araújo, R. Moutinho and S. Elmaghraby, Duration Oriented Resource Allocation Strategy on Multiple Resources Projects under Stochastic Conditions, International Conference on Industrial Engineering and Systems Management, Montreal - Canada, 2009.

[5]   A. Tereso, M. Araújo, R. Moutinho and S. Elmaghraby, Quantity Oriented Resource Allocation Strategy on Multiple Resources Projects under Stochastic Conditions, International Conference on Industrial Engineering and Systems Management, Montreal - Canada, 2009.

[6]   R. Mulcahy, PMP Exam Prep, Fifth Edition: Rita's Course in a Book for Passing the PMP Exam, RMC Publications, Inc., 2005.

[7]   A. Tereso, M. Araújo and S. Elmaghraby, Adaptive resource allocation in multimodal activity networks, International Journal of Production Economics, Vol. 92, pp. 1-10, 2004.

[8]   S. Elmaghraby and R. Girish, Optimal Resource Allocation in Activity Networks Under Stochastic Conditions, Internal Report, North Carolina State University, 2010.