

Antropopatia em Entidades Virtuais

Cesar Analide e José Neves

Departamento de Informática
Universidade do Minho
Braga, PORTUGAL
{analide, jneves}@di.uminho.pt

Abstract. A associação de propriedades e comportamentos próprios dos seres humanos a entidades virtuais, sejam estas agentes, objectos ou outros tipos de entidades, tem passado, quase invariavelmente, pela avaliação desses pressupostos através de uma função de utilidade que mede essa qualidade, quantificando-a. Neste trabalho apresenta-se uma via alternativa para analisar o conhecimento de um sistema inteligente, no sentido de utilizar propriedades antropopáticas para definir o seu comportamento; o reconhecimento destas qualidades será efectuado com base numa análise qualitativa do seu corpo de conhecimento.

1 Introdução

No campo das Ciências da Computação, da Inteligência Artificial e dos Sistemas Multiagente, tem-se verificado uma aproximação crescente destas disciplinas ao espaço de trabalho das Ciências Sociais, desenvolvendo-se actividades cada vez mais relacionadas com as áreas da Antropologia, da Sociologia e da Psicologia.

Muitos têm sido os trabalhos que abordam temas relacionados com a personificação do comportamento de agentes virtuais, pela utilização de exercícios de categorização de sentimentos e emoções. Em [12] e [13] são apresentados estudos detalhados e propostas linhas de acção que consideram o tratamento da razão e a forma de atribuir emoções às máquinas. Atitudes como cooperação, competição e socialização entre agentes [5] são exploradas em diversas áreas como a Economia [4] e a Física [9], pela sua aplicação à resolução de problemas, como são os casos conhecidos por *El Farol Bar Problem*, *Minority Game* e *Iterated Prisoner's Dilemma*. Em [6] e [8] é reconhecida a importância da modelação de personalidades em agentes, no sentido de estes utilizarem características e qualidades próprias dos humanos nas suas tomadas de decisão.

Uma motivação importante para o desenvolvimento deste trabalho prende-se com a capacidade de representação de conhecimento e a sua flexibilidade na construção de soluções, permitidas pela Programação em Lógica Estendida (PLE) [1] [11] [14].

A utilização de valores nulos e a caracterização de situações de excepção relativas ao comportamento dos sistemas inteligentes são outra justificação para a adopção destes formalismos na representação do conhecimento. Sistemas inteligentes descritos deste modo fazem com que seja possível estabelecer os tipos de comportamento a es-

perar dos seus elementos constituintes (neste caso, tais entidades dão pelo nome e transportam toda a carga semântica de agente), o que vai no sentido de lhes associar propriedades e comportamentos próprios do ser humano; i.e., que apresentem valências antropopáticas. De acordo com [15]

1.1 Representação de Conhecimento

A representação de conhecimento, como forma de descrição do mundo real, com base em meios mecânicos, lógicos, ou outros, será sempre função da capacidade do sistema em descrever com maior ou menor exactidão o conhecimento que existe e aquele que se pretende representar. Na concepção de um sistema de representação de conhecimento é necessário considerar situações como as que se passam a descrever:

Informação existente – pode não ser sempre conhecida em toda a sua extensão. Uma situação real é caracterizada por um determinado conjunto de dados, que podem não ser todos conhecidos no instante em que se aborda o problema (uma observação em condições deficientes de luminosidade não produz uma descrição da realidade com toda a informação que a caracteriza).

Informação observada – que se adquire pela experiência; que se obtém por contacto ou observação. É a informação que a observação da realidade consegue reproduzir. Atente-se que a informação observada depende do observador, da mesma forma que uma medição é influenciada pelo instrumento de medida: uma descrição verbal de um acontecimento depende da formação do observador, do seu estado de espírito, dos seus preconceitos (só para mencionar alguns condicionalismos).

Informação representada – com respeito a um determinado objectivo, pode ser (ir)relevante representar certa informação. Apesar de todas as condicionantes, é possível que observações feitas por sujeitos diferentes, com formação e motivação distintas, apresentem o mesmo conjunto de dados fundamentais, função da utilidade da informação obtida. A diferença pode ter um papel secundário em comparação com a importância da informação comum (relevante) entre as observações. É esta informação que deve ser representada e entendida como característica do problema.

1.2 Informação Incompleta

Apesar de, numa teoria lógica clássica, o conhecimento ser equacionado em termos do que se prova ser *verdadeiro*, o que se prova ser *falso* e o que representa informação sobre a qual não se pode ser conclusivo, num programa em lógica, as respostas às questões são, apenas, de dois tipos: *verdadeiro* ou *falso*. Isto deve-se ao facto de um programa em lógica apresentar limitações em termos da representação de conhecimento (não permite representar explicitamente informação negativa), para além de que, em termos de uma semântica operacional, se aplica, automaticamente, o Pressuposto do Mundo Fechado (PMF) a todos os predicados.

A generalidade dos programas escritos em lógica representa implicitamente a informação negativa, assumindo a aplicação do raciocínio segundo o PMF. Uma extensão de um programa em lógica pode incluir informação negativa explicitamente [1]

[11], bem como explicitar directamente o PMF para alguns predicados. Consequentemente, torna-se possível distinguir três tipos de conclusões para uma questão: *verdadeira*, *falsa* ou, quando não exista informação que permita inferir uma ou outra, a resposta será *desconhecida*.

2 Preliminares

Este trabalho é suportado, em parte, por trabalho desenvolvido em [2] onde é abordada a temática da representação de informação incompleta e do raciocínio baseado em pressupostos parciais, para os quais se fez uso da representação de valores nulos [3] [10] na caracterização de situações anómalas ou de excepção.

2.1 Valores Nulos

Nesta secção é feito um estudo do tipo de valores mais comuns que podem surgir em situações de informação incompleta. Esta identificação de valores nulos surge como uma estratégia para a enumeração de casos, para os quais se pretende fazer a distinção entre situações em que as respostas a questões deverão ser concretizadas como *conhecidas* (verdadeiras ou falsas) ou *desconhecidas* [3] [14].

A representação de valores nulos será abordada no âmbito da PLE. Serão considerados, genericamente, dois tipos de valores nulos: o primeiro permitirá representar valores desconhecidos, não necessariamente de um conjunto determinado de valores, enquanto que o segundo representará valores desconhecidos, de um conjunto finito e determinado de valores.

Como caso prático, considere-se o horário de chegada das composições numa estação de caminhos-de-ferro. Para o efeito, utilizar-se-á o seguinte predicado:

ligação: Localidade \times Horário

em que o primeiro argumento identifica o local de partida e o segundo representa a hora de chegada (e.g., ligação(guimarães,17:00) denota que o comboio vindo de Guimarães tem chegada prevista para as 17 horas). O facto de este comboio não chegar nem às 17:01, nem às 16:59, nem em qualquer outro horário que não o especificado, representa-se tal como ilustra a segunda cláusula do Programa 1, apontando que deve ser interpretada como falsa qualquer situação para a qual não exista uma prova de que seja verdadeira.

```
ligação( guimarães,17:00 )
¬ligação( L,H ) ←
    não ligação( L,H )
```

Programa 1: Extensão do predicado que denota as chegadas à estação de caminho-de-ferro

Neste último programa, a conectiva \neg denota a negação forte, determinando o que deve ser interpretado, efectivamente, como *falso*, e o termo não designa a negação por falha na prova.

Desconhecido Indeterminado

Partindo do exemplo dado no Programa 1, admita-se que a ligação com origem no Porto ainda não chegou. A representação desta situação será feita à custa da identificação de um valor nulo, do tipo *desconhecido indeterminado*, que deverá permitir concluir que tal ligação existe, mas para a qual não se pode ser conclusivo relativamente à hora da chegada (Programa 2).

```
ligação( guimarães,17:00 )
ligação( porto,⊥ )
¬ligação( L,H ) ←
    não ligação( L,H ) ∧ não excepção( ligação( L,H ) )
excepção( ligação( L,H ) ) ←
    ligação( L,⊥ )
```

Programa 2: Informação relativa à ligação do Porto, com atraso não programado

Neste contexto, o símbolo \perp denota um valor nulo do tipo indefinido, ou seja, é uma representação que assume que qualquer valor corporiza uma eventual solução, mas que não permite que se seja conclusivo. Computacionalmente, não é possível determinar, a partir da informação positiva, a hora de chegada da ligação vinda do Porto; pela descrição da situação de excepção, através da quarta cláusula do Programa 2 (fecho do predicado *ligação*), fica vedada a possibilidade de se considerar falsa qualquer interpelação que questione sobre a hora específica de chegada dessa ligação.

Desconhecido Enumerado

Considere-se outro exemplo em que a ligação Lisboa com chegada prevista para as 18 horas, vem com um atraso até 15 minutos. Não é possível considerar como verdadeiro que o comboio chegue às 18:00; ou às 18:01; ou às 18:15. Contudo, pela informação disponível, pode-se considerar falso que o comboio chegue às 18:16 ou às 17:59. Este exemplo sugere que o desconhecimento relativamente a uma questão ocorre, apenas, para um conjunto enumerado de casos (Programa 3).

```
ligação( guimarães,17:00 )
ligação( porto,⊥ )
¬ligação( L,H ) ←
    não ligação( L,H ) ∧ não excepção( ligação( L,H ) )
excepção( ligação( L,H ) ) ←
    ligação( L,⊥ )
excepção( ligação( lisboa,H ) ) ←
    H ≥ 18:00 ∧ H ≤ 18:15
```

Programa 3: Representação da informação relativa à ligação com atraso de 15 minutos

A excepção ocorre, apenas, entre as 18:00 e as 18:15. É desconhecido se a ligação de Lisboa chega às 18:05 ou às 18:10; é falso que chegue às 17:55 ou às 18:20.

2.2 Interpretação de Valores Nulos

Para interpretar uma base de conhecimento descrita segundo os formalismos expostos, considere-se a construção de um procedimento representado pela extensão de um predicado dado em termos de uma linguagem de programação em lógica, `demo`, com a capacidade de analisar uma questão e calcular uma solução segundo os pressupostos descritos pela PLE, na qual se processa a representação de valores nulos. Este meta-predicado será definido na forma:

`demo: Questão × Resposta`

onde `Questão` identifica a interpelação a ser analisada em termos da PLE e `Resposta` identifica o valor de verdade a associar à sua resolução, representando uma de três possibilidades: Verdadeiro (\mathcal{V}), Falso (\mathcal{F}) ou Desconhecido (\mathcal{D}) (Programa 4).

`demo(Q, \mathcal{V}) ← Q`
`demo(Q, \mathcal{F}) ← $\neg Q$`
`demo(Q, \mathcal{D}) ← não Q \wedge não $\neg Q$`

Programa 4: Extensão do meta-predicado de interpretação do conhecimento descrito em termos da PLE, que contempla a representação de valores nulos

A primeira cláusula do Programa 4 determina que é verdadeira uma questão com solução ao nível da informação positiva; a segunda cláusula define ser falsa qualquer questão que admita prova em termos da informação negativa; a terceira cláusula considera a interpretação da questão desconhecida se não tiver sido possível encontrar uma resposta nem em termos positivos nem em termos negativos. A expressão:

`demo(ligação(guimarães,17:00), R) ?`

corresponde à colocação de uma questão através do meta-predicado `demo`, para que seja resolvida mediante os pressupostos enunciados pela PLE. A expressão denota uma interpelação ao sistema por forma a calcular a resposta, `R`, sobre uma ligação vinda de Guimarães com chegada prevista para as 17 horas. Em função do mecanismo de raciocínio implementado pelo Programa 4, a resolução desta questão passará por `R` tomar o valor de verdade \mathcal{V} . Contudo, a questão:

`demo(ligação(porto,20:00), R) ?`

não admite uma resposta conclusiva (\mathcal{V} ou \mathcal{F}) já que se está perante um caso que descreve um valor nulo do tipo indefinido; a resposta `R` será desconhecida, \mathcal{D} , para qualquer valor possível para a hora de chegada. Já no caso das questões:

`demo(ligação(lisboa,18:10), R) ?`
`demo(ligação(lisboa,18:20), R) ?`

a primeira terá resposta desconhecida (\mathcal{D}), porque a hora questionada para a chegada está dentro dos limites admitidos para o atraso, enquanto que a segunda questão resultará numa resposta falsa (\mathcal{F}), por ficar fora do intervalo definido para o atraso.

3 Reconhecimento de Qualidades Antropopáticas

Tendo por base, para a representação do conhecimento, os pressupostos enunciados anteriormente, é possível estabelecer mecanismos de análise da informação por forma a estudar o comportamento de entidades virtuais, em termos da sua personificação. Situações que envolvam esquecimento, lembrança, aprendizagem ou confiança poderão ser analisadas da forma que aqui se propõe.

A descrição de situações anómalas através da ocorrência de exceções na extensão de um predicado possibilita o estudo, em termos do conhecimento representado e da excelência dessa informação.

3.1 Caracterização de um Problema

Considere-se o seguinte exemplo, criado para demonstrar uma aplicação prática do que se pretende que seja a principal contribuição deste trabalho.

```
filho( joão, carlos )
¬filho( F,P ) ←
    não filho( F,P ) ∧ não exceção( filho( F,P ) )
```

Programa 5: Excerto de um programa em lógica estendida, representando o conhecimento num instante de tempo t_i

No Programa 5 é dado o axioma que denota que João é filho de Carlos. Assumindo que este é todo o conhecimento disponível no instante t_i , a segunda cláusula no Programa 5 impõe que deverão ser interpretadas como falsas todas as outras situações para as quais não se disponha de informação e não se verifique nenhuma exceção.

Suponha-se que, num instante posterior, t_j , o conhecimento evoluiu de tal forma que passou a ser representado pelo Programa 6.

```
¬filho( F,P ) ←
    não filho( F,P ) ∧ não exceção( filho( F,P ) )
exceção( filho( joão, carlos ) )
exceção( filho( joão, luís ) )
exceção( filho( joão, pedro ) )
```

Programa 6: Excerto da base de conhecimento para o instante de tempo t_j

Num terceiro instante de tempo, t_k , a informação detida passou a ser a que se representa no Programa 7.

```
filho( joão, ⊥ )
¬filho( F,P ) ←
    não filho( F,P ) ∧ não exceção( filho( F,P ) )
exceção( filho( F,P ) ) ←
    filho( F, ⊥ )
```

Programa 7: Excerto do programa que resultou da evolução do conhecimento entre o estado t_j e t_k , onde \perp identifica um valor nulo do tipo indefinido

Olhando para a evolução do conhecimento, desde o instante t_i até t_k , pode-se dizer que a informação representada foi perdendo especificidade, passando-se de uma situação em que se sabia ser João filho de Carlos (t_i), para outra em que já só se conheciam três hipóteses para determinar de quem o João era filho (Carlos, Luís ou Pedro) sem poder concretizar sobre nenhuma em particular (t_j), para, finalmente um terceiro instante (t_k), no qual já não se sabe, concretamente, de quem João será filho; i.e., admite-se qualquer hipótese para quem possa ser pai de João, não se podendo afirmar, contudo, que João não é filho de ninguém.

Neste sentido, em termos do eixo temporal $t_i \rightarrow t_j \rightarrow t_k$, pode-se afirmar que a evolução do conhecimento se deu de uma forma que permite diagnosticar a ocorrência de esvaziamento de informação (i.e., esquecimento) na base de conhecimento. Contudo, atendendo à evolução do conhecimento no sentido inverso, i.e., $t_k \rightarrow t_j \rightarrow t_i$, uma análise semelhante permite concluir que houve especialização no conhecimento representado, ou seja, pode-se falar em ter sido recordado ou ter-se aprendido algo em particular.

3.2 Procedimentos de Reconhecimento de Propriedades

No actual contexto, é possível chamar a atenção para o reconhecimento de qualidades próprias do ser humano, representadas num sistema inteligente que adoptou a PLE para a formalização do seu corpo de conhecimento. Por conseguinte, tem-se como propósito definir mecanismos que permitam, fazendo este reconhecimento, formalizar procedimentos que visam caracterizar qualquer agente de conhecimento em termos das suas qualidades antropopáticas, ou seja, reconhecer e conferir, a um sistema inteligente, atributos próprios do ser humano, permitindo uma certa personificação de agentes computacionais.

Para estudar tais procedimentos considere-se, novamente, o Programa 5 que descreve o estado do sistema no instante t_i , ao qual se coloca a questão para determinar de quem João será filho, $\text{filho}(\text{joão}, P)$?. Em termos do meta-predicado demo , tem-se:

1º Passo: $\forall_{(P)}: \text{demo}(\text{filho}(\text{joão}, P), \mathcal{V})?$
 \angle sucesso

$\forall_{(P)}: \text{demo}(\text{filho}(\text{joão}, P), \mathcal{F})?$
 \angle insucesso

$\forall_{(P)}: \text{demo}(\text{filho}(\text{joão}, P), \mathcal{D})?$
 \angle insucesso

A questão só tem solução quando interpretada em termos da informação positiva, quando é identificado Carlos como pai de João. Desta forma, num segundo passo em que se pretende determinar o conjunto de todas as soluções existentes para os casos em que se obteve sucesso na prova no 1º Passo, é possível ter uma noção da quantidade de informação que garante a solução da questão:

2º Passo: $\forall_{(P,S)}: \text{soluções}(P, \text{demo}(\text{filho}(\text{joão}, P), \mathcal{V}), S)?$
 $\angle S = [\text{carlos}]$

Considere-se o enquadramento dado pelo Programa 6, em termos do instante t_j . Aplicando o primeiro passo do mecanismo enunciado atrás, tem-se que:

$$\begin{aligned}
 1^\circ \text{ Passo: } & \forall_{(P)}: \text{demo}(\text{filho}(\text{joão}, P), \mathcal{V})? \\
 & \quad \angle \text{insucesso} \\
 & \forall_{(P)}: \text{demo}(\text{filho}(\text{joão}, P), \mathcal{F})? \\
 & \quad \angle \text{insucesso} \\
 & \forall_{(P)}: \text{demo}(\text{filho}(\text{joão}, P), \mathcal{D})? \\
 & \quad \angle \text{sucesso}
 \end{aligned}$$

ou seja, a questão só tem prova ao nível do desconhecimento das soluções concretas que a validam. Significa isto que a aplicação do segundo passo produzirá uma lista vazia de soluções se colocada em termos do meta-predicado demo , dado nos termos:

$$\begin{aligned}
 2^\circ \text{ Passo: } & \forall_{(P,S)}: \text{soluções}(P, \text{demo}(\text{filho}(\text{joão}, P), \mathcal{D}), S)? \\
 & \quad \angle S = []
 \end{aligned}$$

Esta situação denota que existem excepções para as quais não se pode ser conclusivo, permitindo, portanto, chegar a uma solução como sendo de resposta desconhecida, \mathcal{D} . Recorrendo às excepções para garantir uma resposta com uma valoração baseada nesta falta de conhecimento, tem-se:

$$\begin{aligned}
 3^\circ \text{ Passo: } & \forall_{(P,S)}: \text{soluções}(P, \text{excepção}(\text{filho}(\text{joão}, P)), S)? \\
 & \quad \angle S = [\text{carlos}, \text{lúis}, \text{pedro}] \\
 & \forall_{(S,N)}: \text{cardinalidade}(S, N)? \\
 & \quad \angle N = 3
 \end{aligned}$$

Este terceiro passo, que procura as soluções em termos das excepções colocadas à interpretação da questão dada, permite valorar a sua solução. Neste caso, o facto de existirem 3 excepções permite afirmar que, grosso modo, cada excepção contribui com $\frac{1}{3}$ para uma eventual certificação da solução.

Utilize-se, agora, o caso representado pelo Programa 7 referente ao instante t_k . Por aplicação dos mecanismos enunciados para o primeiro passo, tem-se que:

$$\begin{aligned}
 1^\circ \text{ Passo: } & \forall_{(P)}: \text{demo}(\text{filho}(\text{joão}, P), \mathcal{V})? \\
 & \quad \angle \text{insucesso} \\
 & \forall_{(P)}: \text{demo}(\text{filho}(\text{joão}, P), \mathcal{F})? \\
 & \quad \angle \text{insucesso} \\
 & \forall_{(P)}: \text{demo}(\text{filho}(\text{joão}, P), \mathcal{D})? \\
 & \quad \angle \text{sucesso}
 \end{aligned}$$

i.e., uma solução para a questão em equação só pode ser dada em termos de uma valoração de uma solução indefinida; neste caso, a aplicação do segundo passo produz um resultado particular, dado na forma:

$$\begin{aligned}
 2^\circ \text{ Passo: } & \forall_{(P,S)}: \text{soluções}(P, \text{demo}(\text{filho}(\text{joão}, P), \mathcal{D}), S)? \\
 & \quad \angle S = [\perp] \\
 & \forall_{(S,N)}: \text{cardinalidade}(S, N)? \\
 & \quad \angle N = \infty
 \end{aligned}$$

i.e., a quantificação do valor de verdade a associar à solução encontrada, recorre a um mecanismo que parte de um conjunto ilimitado de possíveis soluções, o que explica que a cardinalidade de tal conjunto seja interpretada como tendendo para o infinito.

3.3 Interpretação dos Resultados

No instante de tempo t_i , o conhecimento existente permite garantir uma solução para o problema no que respeita a determinar de quem João é filho. A questão tem solução em termos das extensões da parte positiva dos predicados que corporizam o universo de discurso. No segundo instante de tempo, t_j , não sendo possível determinar de forma conclusiva uma resposta para a questão dada, foi possível calcular, com um certo grau de confiança, a contribuição das extensões dos predicados, em termos de exceções, para uma eventual solução para o problema. Fazendo com que cada exceção contribua com uma parte igual, definiu-se o valor de cada contribuição parcelar como sendo o inverso da cardinalidade das exceções que contribuem para o ruído a associar à solução. No terceiro instante, t_k , apesar de a solução para o problema continuar a ser desconhecida, tal situação ocorre quando se encontra um valor nulo com a capacidade de representar um conjunto indeterminado de possíveis soluções. Desta forma o grau de confiança que se pode atribuir à solução para o problema tende a ser nulo, à medida que o conjunto de possíveis soluções tende para um conjunto infinito de termos, o que é dado na forma:

$$G_{\text{filho(joão,P)}} = \lim_{\perp \rightarrow \infty} \frac{1}{\perp} = 0$$

em que G mede o valor de verdade ou valoração atribuído à solução.

Assim, considerando que a evolução do conhecimento no sistema se deu no sentido $t_i \rightarrow t_j \rightarrow t_k$, o grau de confiança para a mesma questão em cada um dos instantes diminuiu, pelo que se considera ter havido um processo de esvaziamento de conhecimento, vulgo, esquecimento. Admitindo a evolução do conhecimento de forma inversa, $t_k \rightarrow t_j \rightarrow t_i$, dir-se-ia que tendo aumentado o grau de confiança entre cada instante, o conhecimento concreto para aquela questão foi sendo cada vez mais apurado, o que permite afirmar que o sistema aprendeu a recordar factos ou situações já passadas.

4 Conclusões

Neste trabalho, a ênfase vai para a análise de problemas associados à representação do conhecimento como tal, e não tanto a questões de quantificação da informação, situações que serão de considerar quando se tratar da representação de emoções ou outras propriedades e comportamentos próprios do ser humano.

Estes procedimentos são aplicados em bases de conhecimento que subscrevem a PLE como ferramenta de representação de conhecimento, o que permite, logo à partida, uma maior flexibilidade quer no que respeita à descrição de dados e conhecimento, quer nos mecanismos de raciocínio a utilizar. A aplicação destes mecanismos na área dos sistemas inteligentes resulta, sobretudo, em problemas em que se torne

evidente a necessidade de um agente tirar partido do conhecimento de um ou mais de seus pares, ou seja, quanto mais um agente for capaz de conhecer acerca do comportamento ou da personalidade de outro agente, maior será a sua capacidade de sobreviver nesse meio ambiente. Por exemplo, em sistemas de comércio electrónico [7] em que o comportamento dos agentes envolvidos numa negociação pode influenciar ou ser influenciado pelas acções de oponentes, é importante ter a possibilidade de recorrer a mecanismos que permitam antecipar ou influenciar tais comportamentos em proveito próprio.

Agradecimentos: O trabalho desenvolvido por José Neves foi apoiado pelo projecto POSI/EEL/13096/2000

Referências

1. Alferes, J., Pereira, L., Przymusiński, T.: Classical Negation in Nonmonotonic Reasoning and Logic Programming. *Journal of Automated Reasoning*, vol. 20, pp. 107-142 (1998)
2. Analide, C.: Representação de Conhecimento e Raciocínio em Estruturas Hierárquicas. Dissertação de Mestrado, Departamento de Informática, Universidade do Minho (1996)
3. Analide, C., Neves, J.: Representação de Informação Incompleta. 1ª CAPSI, 25-27 de Outubro, Universidade do Minho, Guimarães, Portugal (2000)
4. Arthur, W.: Inductive Reasoning and Bounded Rationality (The El Farol Bar Problem). *Proceedings of the American Economic Review*, 84:406 (1994)
5. Bazzan, A., Bordini, R.: Evolving agents with moral sentiments in an iterated prisoner's dilemma exercise. *Proceedings of the 2nd Workshop on Game Theoretic and Decision Theoretic Agents*, 4th ICMAS, 10-12 July, Boston, 13-25 (2000)
6. Bazzan, A., Bordini, R., Vicari, R., Wahle, J.: Evolving populations of agents with personalities in the minority game. *Proceedings of the International Joint Conference: 7th IBERAMIA and 15th SBIA*, 19-22 November, Atibaia-SP, Brazil, number 1952 in *Lecture Notes in Artificial Intelligence*, 166-175. Berlin: Springer-Verlag (2000)
7. Brito, L., Novais, P., Neves, J.: On the Logical Aspects of Argument-based Negotiation among Agents. 5th International Workshop on Cooperative Information Agents, Modena, Itália, 6-8 de Setembro (2001)
8. Castelfranchi, C., Rosis, F., Falcone, R.: Social Attitudes and Personalities. In *Agents, Socially Intelligent Agents*, AAI Fall Symposium Series 1997, MIT in Cambridge, Massachusetts, November 8-10 (1997)
9. Challet, D., Zhang, Y.: On the Minority Game: Analytical and Numerical Studies, *Physica A* 256, 514 (1998)
10. Gelfond, M.: Logic Programming and Reasoning with Incomplete Information, in *Annals of Mathematics and Artificial Intelligence*, 12, pp. 89-116 (1994)
11. Neves, J.: A Logic Interpreter to Handle Time and Negation in Logic Data Bases, in *Proceedings of ACM'84, The Fifth Generation Challenge*, pp. 50-54 (1984)
12. Ortony, A., Clore, G. L., and Collins, A.: *The Cognitive Structure of Emotions*. Cambridge, UK: Cambridge University Press (1988)
13. Picard, R.: *Affective Computing*, MIT Press, Cambridge (1997)
14. Traylor, B., Gelfond, M.: Representing Null Values in Logic Programming. *Proceedings of the ILPS'93 Workshop on Logic Programming with Incomplete Information*, pp. 35-47, Vancouver, Canadá (1993)
15. Costa, J., Melo, A.: *Dicionário da Língua Portuguesa – 7ª Edição*. Porto Editora L^{da}, Porto, Portugal (1997)