

# Criação de Agentes BDI a partir de Modelos do UPPAAL

João Vicente Markovicz, Gleifer Vaz Alves, André Pinz Borges

Universidade Tecnológica Federal do Paraná - Campus Ponta Grossa

16 de agosto de 2024



- 1 Introdução
- 2 Modelos Éticos e Agentes BDI
- 3 Modelagem Formal das Abordagens Éticas
- 4 Tradução dos Modelos para Agentes BDI
- 5 Considerações Finais

- Geração de protótipos de agentes BDI (*Belief, Desire, Intention*) a partir de autômatos modelados no UPPAAL.
- Cenário envolvendo tomada de decisão ética.
- Extensão dos modelos construídos em:



MARKOVICZ, J. V., ALVES, G. V. (2023)

Modelagem formal de abordagens éticas para comportamento de agentes.

In *Anais do Workshop-Escola de Informática Teórica (WEIT)*, pages 134-138.



BENCH-CAPON, T. (2020)  
Ethical Approaches and Autonomous Systems.  
*Artificial Intelligence*, 281:103239.

## Ethical approaches and autonomous systems

T.J.M. Bench-Capon

*Department of Computer Science, University of Liverpool, Liverpool, UK*



### ARTICLE INFO

*Article history:*

Received 27 February 2019

Received in revised form 19 September 2019

Accepted 16 January 2020

Available online 21 January 2020

*Keywords:*

Consequentialism

Deontology

Virtue ethics

Reasoning about actions

Ethical agents

### ABSTRACT

In this paper we consider how the three main approaches to ethics – deontology, consequentialism and virtue ethics – relate to the implementation of ethical agents. We provide a description of each approach and how agents might be implemented by designers following the different approaches. Although there are numerous examples of agents implemented within the consequentialist and deontological approaches, this is not so for virtue ethics. We therefore propose a novel means of implementing agents within the virtue ethics approach. It is seen that each approach has its own particular strengths and weaknesses when considered as the basis for implementing ethical agents, and that the different approaches are appropriate to different kinds of system.

© 2020 Elsevier B.V. All rights reserved.

- Descreve abordagens éticas em agentes:
  - Consequencialista;
  - Deontológica;
  - Ética das Virtudes.
- Aplicação dos modelos em sistemas autônomos.

# Cenário

- Baseado na história “A Formiga e a Cigarra”.
- No verão a Formiga trabalha e a Cigarra joga;
- No inverno a Formiga tem comida e a Cigarra depende de sua ajuda;
- O autor propõe três abordagens para o problema.



- **UPPAAL**: ferramenta para modelagem e verificação formal de autômatos;



MARKOVICZ, J. V., ALVES, G. V. (2023)

Modelagem formal de abordagens éticas para comportamento de agentes.

In *Anais do Workshop-Escola de Informática Teórica (WEIT)*, pages 134-138.

- Modelagem dos autômatos na Ferramenta UPPAAL para as três abordagens.



- Estabelecer uma conexão entre os autômatos modelados e sua implementação usando agentes BDI;
- Arquitetura de agentes:
  - *Belief* - Crenças;
  - *Desire* - Objetivos/Desejos;
  - *Intention* - Intenções/Planos;
- Formato de um plano:

$TriggerEvent : \{Context\} \leftarrow Body$

# Modelagem Formal das Abordagens Éticas

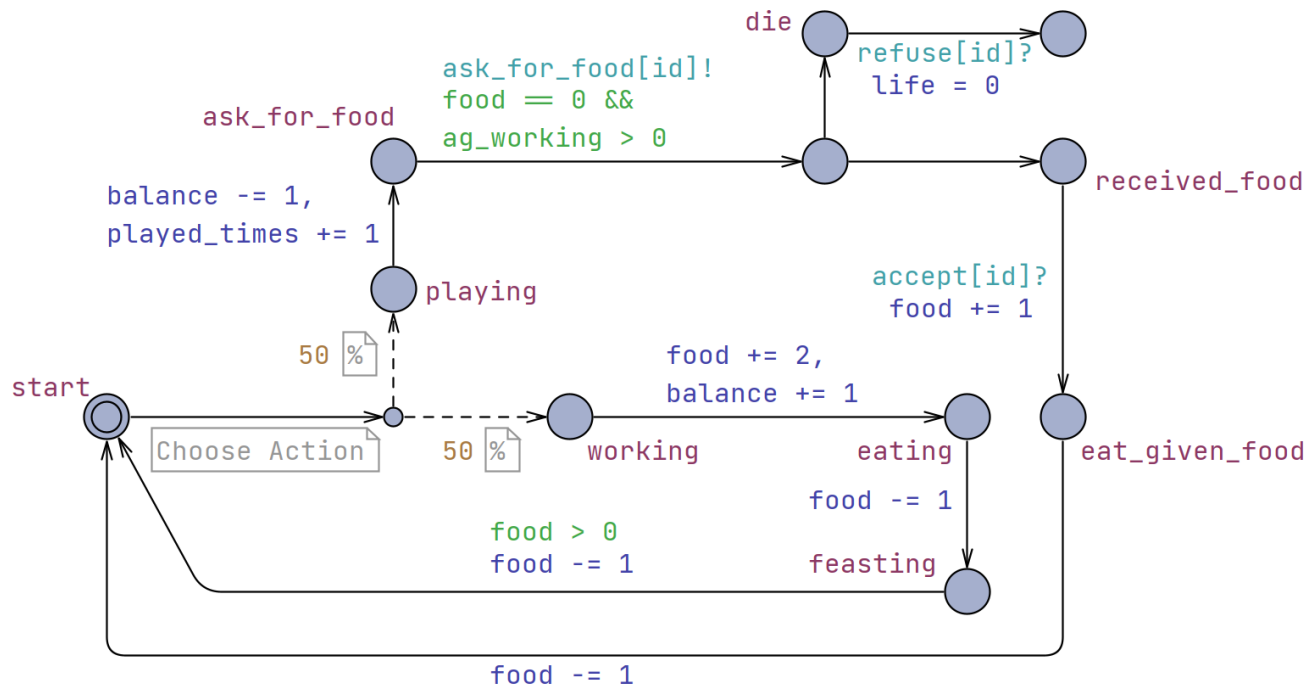
- A cada vez que a Cigarra joga ou trabalha variáveis são alteradas:
  - **Balance** - incrementa quando um agente trabalha, decrementa quando ele joga.
  - **Played\_times** - salva a quantidade de vezes o agente jogou.
- Modelo Deontológico
  - Utiliza a variável **played\_times** para decisão;
- Modelo Consequencialista
  - Utiliza a variável **balance** para decisão;
- Modelo da Ética das Virtudes
  - Utiliza ambas as variáveis para decisão;

# Modelagem Formal das Abordagens Éticas

- Em [MARKOVICZ (2023)], foi elaborado um agente que realizava o papel de ambos os personagens;
- Neste trabalho, separação dos papéis dos agentes;
- Uso de uma nova versão do modelo da Ética das Virtudes;
- Novas técnicas de abstração para facilitar o processo de tradução.



# Autômato Cigarra



# Tradução dos Modelos para Agentes BDI

- Estabelecer conexão:
  - Modelo Ético
  - Agentes BDI
- Nesse momento, não foi determinada uma linguagem alvo específica;
- Almeja-se que o protótipo possa ser aplicado em linguagens de Agentes BDI, que sigam o *AgentSpeak*.

# Tradução dos Modelos para Agentes BDI - Crenças e Objetivos

- Crenças → Variáveis declaradas no UPPAAL;
- Objetivos → Estados do autômato, nesse caso apenas o estado inicial.

```
1 Beliefs: life(1), food(0);  
2 Goals: start;
```

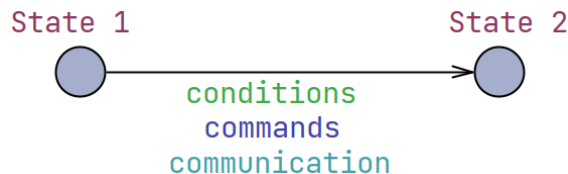
## Protótipo 1. Crenças e Objetivos - Agente Formiga

```
1 Beliefs: life(1), food(0), balance(0), played_times(0);  
2 Goals: start;
```

## Protótipo 2. Crenças e Objetivos - Agente Cigarra

# Tradução dos Modelos para Agentes BDI - Planos

- Para efetuar a tradução dos Planos do agente foi usado o seguinte mecanismo:



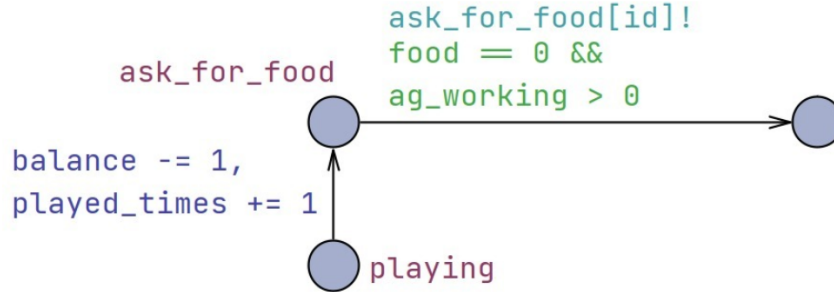
- Obtendo o seguinte plano:

$State1 : \{Conditions\} \leftarrow Commands, send(Agent, Communication), State2;$

- Obs.: nem sempre todos os componentes serão utilizados

# Exemplo da Tradução dos Planos de um agente

- A seguir um fragmento do autômato da Cigarra:



- Código resultante:

```
1 playing : {balance(X), played_times(Y)} <-  
2           balance(X-1), played_times(Y+1), ask_for_food;  
3 ask_for_food : {food(0) and ag_working()} <- send(Ant, negotiate(GH));
```

Legenda:

Ant → Formiga;

GH (Grasshopper) → Cigarra;

# Tradução dos Modelos para Agentes BDI - Planos Agente Formiga

```
1 start : {true} <- working;  
2 working : {food(X)} <- +food(X+2), eating;  
3 eating : {food(X)} <- food(X-1);  
4 negotiate(GH) : {true} <- analyse_negotiation(GH);  
5 give_food(GH) : {food(1)} <- send(GH, accept), start;  
6 refuse_food(GH) : {true} <- send(GH, refuse), feasting;  
7 feasting : {food(X) and X>0} <- food(X-1), start;
```

## Protótipo 3. Planos - Agente Formiga

# Tradução dos Modelos para Agentes BDI - Planos Agente Cigarra

```
1 start : {true} <- choose_action;
2 choose_action : {true} <- choose_action();
3 working : {food(X), balance(Y)} <- food(X+2), balance(Y+1), eating ;
4 playing : {balance(X), played_times(Y)} <-
5     balance(X-1), played_times(Y+1), ask_for_food;
6 ask_for_food : {food(0) and ag_working()} <- send(Ant, negotiate(GH));
7 received_food : {accept and food(X)} <- food(X+1), eat_given_food;
8 eat_given_food : {food(X)} <- food(X-1), start;
9 eating : {food(X)} <- food(X-1), feasting;
10 feasting : {food(X)} <- food(X-1), start;
11 die : {refuse} <- life(0);
```

## Protótipo 4. Planos - Agente Cigarra

# Considerações Finais

- A extensão dos modelos favoreceu a transição para um protótipo usando paradigma de Agentes BDI;
- Trabalhos Futuros:
  - Implementação dos protótipos em linguagem de programação para agentes, como Jason [Bordini et al. 2007], GWENDOLEN [Dennis and Müller] ou MASPY.
  - Alinhar com as atividades do Lab. de Pesquisa (LACA-IS), que tem como um dos tópicos Agentes e Veículos Autônomos (VAs);
  - Extensão para representar VAs em cenários críticos com suporte de modelos éticos.

# Referências



ALVES, G., DENNIS, L., FISHER, M. (2021a)

An Agent-based Architecture with Support to Ethical Decisions on a Road Traffic Scenario.

Publisher: Zenodo.



BENCH-CAPON, T. (2020)

Ethical Approaches and Autonomous Systems.

*Artificial Intelligence*, 281:103239.



BENGTSSON, J., LARSEN, K., LARSSON, F., PETTERSSON, P., YI, W. (1996)  
UPPAAL — a tool suite for automatic verification of real-time systems.

In Alur, R., Henzinger, T. A., and Sontag, E. D. (Eds.), *Hybrid Systems III*, number 1066 in Lecture Notes in Computer Science, pages 232–243. Springer Berlin Heidelberg.

# Referências



BORDINI, R. H., HÜBNER, J. F., and WOOLDRIDGE, M. (2007).  
Programming Multi-Agent Systems in AgentSpeak using Jason (Wiley Series in Agent  
Technology).  
John Wiley & Sons, Inc., Hoboken, NJ, USA.



BRATMAN, M (1987)  
Intentions, Plans and Practical Reason.  
Cambridge: Cambridge, MA: Harvard University Press.



DENNIS, L. A., MÜLLER, B. (2008)  
A BDI language for verifiable agents.



MARKOVICZ, J. V., ALVES, G. V. (2023)  
Modelagem formal de abordagens éticas para comportamento de agentes.  
In *Anais do Workshop-Escola de Informática Teórica (WEIT)*, pages 134-138.

# Contatos

- João Vicente Markovicz  
joao.080603@alunos.utfpr.edu.br
- Gleifer Vaz Alves  
gleifer@utfpr.edu.br
- André Pinz Borges  
apborges@utfpr.edu.br
- LaCA - Intelligent Systems  
<https://laca-is.github.io/>

