

# Composição de Agentes EBDI: Integração WASABI-Jason

## Composing EBDI Agents through WASABI-Jason Intregation

Diego Couto <sup>1</sup>  
Jerusa Marchi <sup>2</sup>  
Thiago Â. Gelaim <sup>2</sup>

*Data de submissão: 30/07/2017, Data de aceite: 25/10/2017*

**Resumo:** Emoções assumem um papel muito importante no processo de tomada de decisão de seres humanos e é natural que a sua utilização torne-se um campo bastante fértil na área de Inteligência Artificial, seja com o intuito de aproximar o comportamento de agentes computacionais ao de humanos, ou simplesmente para aprimorar a relação humano-computador. Este trabalho apresenta a integração do simulador de emoções WASABI com o arcabouço para agentes BDI JASON, para prover agentes emocionais. Os resultados obtidos sugerem que o desempenho dos agentes no ambiente, assim como em sociedades humanas, é suscetível não só ao seu comportamento programado, mas à sua personalidade emocional e às condições do ambiente.

**Abstract:** Emotions play a very important role in the decision-making process of human beings and it's natural that the use of emotions become a very fertile field in the Artificial Intelligence area, whether in order to approximate the behavior of computational agents to humans, or simply to improve the human-computer interaction. This work presents the integration of WASABI emotions simulator and JASON BDI framework in order to provide emotional agents. The results suggest that the performance of the agents in the environment is susceptible not only to their behavior, but to their emotional character and environmental conditions.

**Palavras-chave:** Agentes, emoções, modelo EBDI.

**Keywords:** Agents, emotions, EBDI model.

## 1 Introdução

Reconhecidas como fundamentais para o processo de tomada de decisão em seres humanos, as emoções também fomentam uma linha de pesquisa quando a inteligência é artificial. Usualmente, emoções eram incorporadas a agentes inteligentes como uma tentativa de simular o comportamento humano, fazendo uso de um ambiente de testes controlado para reduzir as variáveis disponíveis a fim de melhor

---

<sup>1</sup>Departamento de Informática e Estatística - INE  
Universidade Federal de Santa Catarina - UFSC  
Florianópolis - SC - Brasil  
[diego.gcouto@gmail.com](mailto:diego.gcouto@gmail.com)

<sup>2</sup>Programa de Pós-Graduação em Ciência da Computação  
Universidade Federal de Santa Catarina - UFSC  
Florianópolis - SC - Brasil  
[jerusa.marchi@ufsc.br](mailto:jerusa.marchi@ufsc.br), [t.gelaim@posgrad.ufsc.br](mailto:t.gelaim@posgrad.ufsc.br)

compreender determinadas características da atuação de seres humanos. São exemplos dessas abordagens os trabalhos de [1], que busca dimensionar a influência que um agente com emoções pode oferecer durante sua interação com um ser humano e [2] que busca compreender o comportamento de pessoas em situações de emergência através de uma simulação computacional.

A importância de se investigar a plausibilidade do uso de emoções no ambiente computacional, entretanto, não se restringe ao desenvolvimento de modelos cognitivos. Emoções também figuram como uma das etapas necessárias para que se possa atingir a consciência como determinada no campo da Inteligência Artificial (IA) forte e como uma maneira de aprimorar a interação entre agentes e seres humanos, através da composição de uma camada social [3].

Ainda que sejam qualidades inesperadas, ao passar a se considerar emoções durante o ciclo de raciocínio, pode-se obter não somente um agente com melhores habilidades cooperativas mas também com melhor desempenho – pela redução na quantidade de ciclos necessários – para tomada de decisões [4]. Esses resultados são particularmente importantes pois demonstram que as emoções são um componente complexo indispensável na construção de sistemas que buscam simular a mente humana. Isso não significa, por outro lado, que o melhor desempenho no ciclo de raciocínio seja uma característica intrínseca das emoções. Em determinadas arquiteturas, em que sua concepção é voltada ao estudo da influência de emoções em ambientes de incerteza, por exemplo, essa não é uma característica assegurada, uma vez que emoções primárias – comumente associadas a um comportamento reativo – não são consideradas [5]. Além disso, é preciso considerar que agentes emocionais podem ser concebidos com personalidades distintas, tornando-os mais ou menos suscetíveis a estímulos emocionais, o que pode impactar tanto positivamente como negativamente nos resultados obtidos através de suas operações no ambiente.

Este trabalho apresenta a integração do simulador de emoções WASABI [6] com o arcabouço Jason [7], onde estados emocionais passam a compor crenças do agente e são utilizados para a tomada de decisão. Dois cenários são utilizados para verificação da proposta. O artigo está estruturado da seguinte forma: a seção 2 apresenta alguns conceitos básicos sobre emoções em agentes, abordando em mais detalhes o modelo WASABI. A seção 3 apresenta detalhes do desenvolvimento do *middleware* que operacionaliza a inserção e manutenção dos estados emocionais de agentes concebidos em Jason, bem como a forma encontrada para modelar a "personalidade" dos agentes. Na seção 4 são descritos os cenários utilizados na verificação e as simulações realizadas, visando verificar a diferença de comportamento entre agentes que fazem uso de emoções em seus processos de raciocínio dos demais. Por fim, a seção 5 apresenta algumas considerações e apontamentos para trabalhos futuros.

## 2 Emoções em Agentes Computacionais

A integração de emoções em sistemas computacionais objetiva melhorar a interação humano-máquina; testar, refinar e desenvolver hipóteses emocionais; e colaborar no refinamento de técnicas de Inteligência Artificial (IA) [8]. Neste sentido, em seu livro *A Sociedade da Mente*, Minsky apresenta que 'a questão não é se máquinas inteligentes podem ter emoções, mas se máquinas podem ser inteligentes sem emoções' [9, p.163] <sup>3</sup>. As duas principais vertentes de modelos computacionais de emoções são: teorias apreciativas (do inglês *appraisal theories*) [10] e teorias dimensionais.

Em teorias apreciativas, a elicitación e diferenciação de emoções são feitas por meio da avaliação, ou apreciação subjetiva do indivíduo considerando uma situação, objeto ou evento [10]. O modelo OCC, proposto por Ortony, Clore e Collins [11], é um dos modelos mais utilizados em sistemas computacionais. Este modelo especifica 22 emoções como baseadas em eventos, baseadas em agentes ou baseadas em

<sup>3</sup>The question is not whether intelligent machines can have any emotions, but whether machines can be intelligent without any emotions.

aspectos de objetos [11].

As teorias dimensionais visualizam emoções e outros fenômenos afetivos, como humor, como pontos em um espaço dimensional contínuo [8]. Este espaço pode ser formado por duas dimensões (prazer e ativação) ou três (prazer, ativação e dominância) [12, 13]. Embora existam outras, como anatômicas, racionais e comunicativas [8], as teorias apreciativas e dimensionais são as mais comumente utilizadas para alicerçar modelos computacionais de emoções, como por exemplo, WASABI [6], FLAME [14], ALMA[15] e EMA [16].

Outro campo de estudo sobre emoções e IA é a integração de emoções na tomada de decisão de agentes cognitivos. Tomando como base o modelo de agentes BDI (*Belief-desire-intention*), emoções são implementadas em dois níveis: num primeiro nível, emoções são usualmente associadas a um comportamento instintivo [17] e por isso, diretamente relacionadas às percepções do agente, já em um segundo nível, emoções são utilizadas como um componente que influencia o mecanismo cognitivo e deliberativo do agente, atualizando seu estado emocional [18].

Quanto a sua duração, percebe-se intuitivamente que é necessário um determinado período para que emoções sejam ativadas e desativadas. No trabalho de Van Dyke Parunak et al [19] este fenômeno é modelado com auxílio do conceito de feromônios, fazendo com que emoções sejam dissipadas com o passar do tempo. Além disso, existem feromônios externos, que representam uma maneira de externalizar emoções de modo que outros agentes possam percebê-las. Baseando-se na estrutura do modelo emocional OCC para calcular a intensidade de uma emoção, da Silva Bitencourt [20] faz uso de uma *função limiar*, considerando também a potencialidade da emoção e o tempo corrente. A função limiar determina qual é o valor mínimo necessário para que a emoção passe a influenciar o comportamento do agente, enquanto a função de potencialidade dita as chances de uma determinada emoção ocorrer. Quando essas duas funções são associadas ao tempo corrente, pode-se gerar um efeito similar ao uso de feromônios, em que o nível de uma emoção varia gradativamente ao longo do tempo.

A arquitetura proposta por Jiang et al. [4] fornece resultados que são bastante importantes no contexto deste trabalho: em um sistema multiagente, agentes com comportamento egoísta podem apresentar melhorias em suas habilidades de cooperação e agentes BDI com emoção podem atingir seus objetivos com um menor número de ciclos se comparado a implementações sem o uso de emoções.

## 2.1 WASABI

A proposta do modelo WASABI [6] tem origem no aprimoramento da interação humano-computador. Por conta de limitações existentes em um modelo de emoções estrutural – como o OCC – no que tange o comportamento crível de agentes emocionais, a ideia que suporta o WASABI envolve a utilização de um espaço tridimensional denominado *Pleasure-Arousal-Dominance Space* ou Espaço PAD [21]. Cada eixo é independente e bipolar representando prazer (*pleasure*), ativação (*arousal*) e dominância (*dominance*), respectivamente. O eixo *x* (prazeroso / não-prazeroso), corresponde aos julgamentos avaliativos, quanto maior a avaliação, mais prazeroso. O eixo *y* (ativação / desativação), representa o nível de atividade. O eixo *z* (dominância / submissão), é uma medida de controle, ou poder [12]. As emoções são definidas no espaço PAD através de polígonos e diz-se que um agente percebe determinada emoção quando seu ponto de referência pertence a um dos pontos delimitados pelo círculo externo de uma dada emoção, enquanto os círculos internos indicam a área em que as emoções possuem valor máximo, como pode ser verificado na Figura 1.

Determina-se, com isso, o estado emocional do agente através da intensidade das emoções definidas no espaço PAD em que o ponto de referência se encontra. Impulsos, positivos ou negativos, fazem com que o ponto de referência mova-se no espaço e, como num sistema de massa-mola, ou seja, o ponto

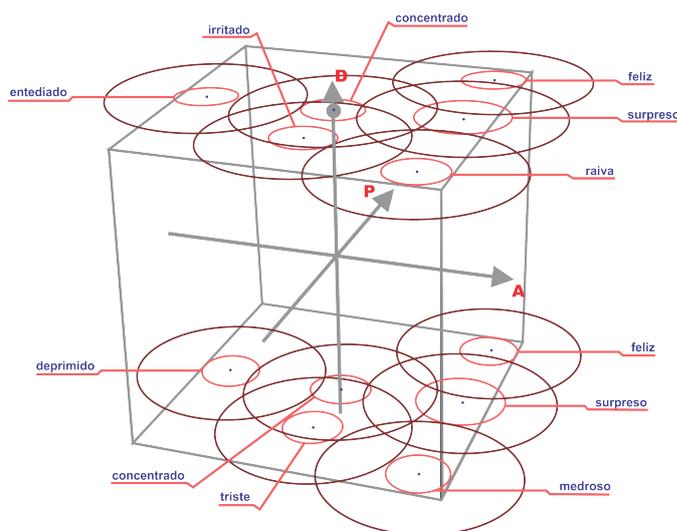


Figura 1: Representação gráfica do espaço PAD.

de referência possui um valor de massa e dada a ocorrência de um impulso, esta massa é deslocada no eixos, promovendo a variação dos estados emocionais. O ponto tende a voltar gradativamente para o seu local de origem (chamado de *estado homeostático*). Esse movimento acontece de forma branda, quanto o valor da massa é maior, ou de forma mais abrupta, quando o valor da massa é menor.

Ao alterar a massa de um ponto de referência ou os coeficientes das forças que agem sobre os eixos, pode-se fazer com que pontos de referência diversos possuam comportamentos distintos no espaço PAD para um mesmo estímulo e a partir de um mesmo estado atual [6], o que será explorado neste trabalho como um artifício para a definição de "personalidades" dos agentes.

Por ter sido desenvolvido para ser executado em um sistema distribuído, o WASABI tem suporte nativo a execução das principais ações da aplicação através de duas portas de comunicação UDP, uma para leitura, em que podem ser enviados comandos predefinidos, e outra para escrita, em que serão impressos os estados afetivos correntes dos pontos de referência existentes.

Código 1: Geração de um impulso com intensidade 100.

```
senderID&IMPULSE&agentId&100
```

O Código 1 apresenta a estrutura básica para a realização de qualquer ação sobre o WASABI. Os diferentes componentes da mensagem são sempre separados pelo símbolo & e todas as mensagens enviadas precisam, obrigatoriamente, começar com um identificador do agente que está enviando a mensagem, representado no código por *senderID*, embora, na implementação atual do WASABI, este valor não seja utilizado na avaliação da mensagem. Como uma consequência de ações como a geração de novos impulsos, o estado afetivo de agentes em simulação pelo WASABI deve ser alterado.

Código 2: Exemplo de mensagem de retorno.

```
(10:51:02) Transmit (success) [
  agents : timeStamp=244344992
  total=2
  ID1=ag1 (   happy=0.5625
```

```

concentrated = 0.75
P = 15
A = -100
D = 0
ID2 = ag2 ( bored = 0.72932
P = 15
A = -100
D = 100 ) ]
    
```

O WASABI envia mensagens, como a exemplificada no Código 2, em que são listadas todas as emoções elicitadas com suas respectivas valências para cada um dos agentes em execução.

### 3 Integração WASABI-Jason

O objetivo deste trabalho é o desenvolvimento de uma interface entre o arcabouço Jason e o WASABI, ou seja, um *middleware*, para que o agente, descrito em Jason, possa ter *crenças emocionais*, e possa fazer uso destas crenças no processo de raciocínio BDI. De modo a garantir que a integração ocorresse sem alterações estruturais tanto no Jason quanto no WASABI, o *middleware* opera basicamente como uma interface entre o WASABI e o Jason, obtendo e estruturando os estados afetivos atuais dos agentes para que serem utilizados durante o ciclo de raciocínio e abstraído e encaminhando comandos para o WASABI, como o envio de impulsos ou alterações de dominância. A Figura 2 ilustra a proposta.

Para que os estados afetivos calculados pelo WASABI possam ser atribuídos a agentes em execução pelo Jason e que impulsos possam ser disparados pelos agentes para o sistema de simulação afetiva, foram definidas as seguintes especificações:

- Impulsos são disparados sobre agentes específicos, assim como alterações de dominância, e devem ser propagados tão logo quanto possível;
- Durante a sua etapa de revisão de crenças, cada agente precisa ter acesso ao seu estado afetivo mais recente, de modo a permitir a atualização das crenças relacionadas ao comportamento emocional;
- A operação do *middleware* precisa ser iniciada e finalizada em conjunto com a execução do sistema multiagente.

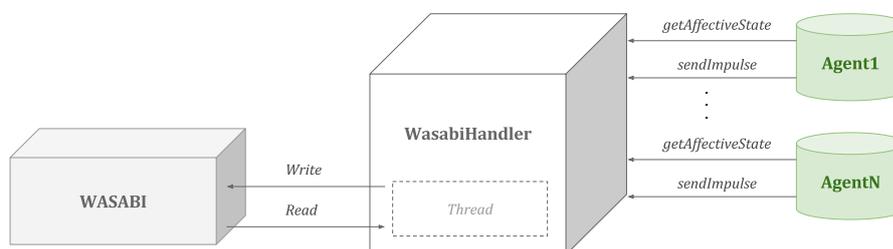


Figura 2: *WasabiHandler*.

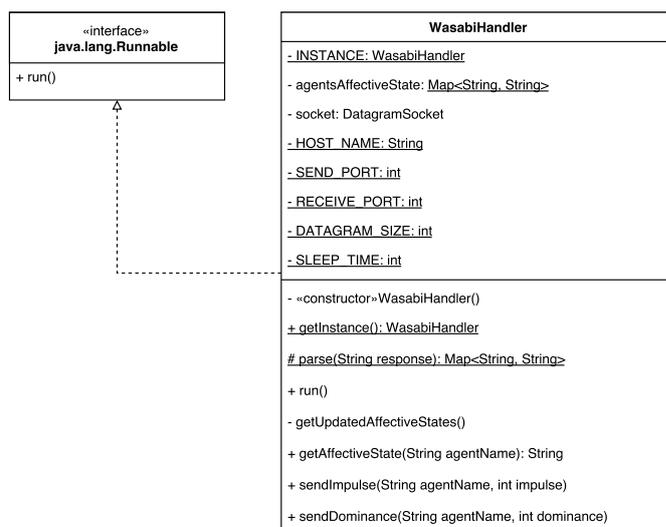


Figura 3: Diagrama de classe UML referente à classe *WasabiHandler*.

Para que os agentes tenham acesso as representações dos seus estados afetivos, durante o processo de revisão de crenças, foi preciso monitorar a escrita de novas mensagens do WASABI com os valores atualizados e, posteriormente, efetuar o *parsing* desse conteúdo em busca dos valores específicos para o agente requerente. Esta única característica influenciou boa parte do projeto do *middleware*, como detalhado no diagrama de classes apresentado na Figura 3, e impulsionou a adoção dos seguintes requisitos básicos:

- Considerando a existência de diversos agentes em execução em um mesmo instante de tempo, a obtenção dos estados afetivos não pode ser executada sob demanda, a fim de evitar *overhead* de comunicação e também impactos negativos no tempo de execução da função de revisão de crenças, aumentando assim, a duração do ciclo de tomada de decisão.
- Fez-se necessário o desenvolvimento de um *buffer*, que contém os valores mais recentes dos estados afetivos de todos os agentes até que uma nova atualização seja indicada pelo WASABI.
- As informações armazenadas no *buffer* estão disponíveis para todos os agentes em simulação.

Visando satisfazer tais requisitos, foi criada uma *thread* específica para a comunicação com o WASABI, a fim de obter os valores dos estados afetivos e evitar o *overhead* das requisições individuais geradas pelos agentes. A operação desta *thread* restringe-se apenas à obtenção dos estados afetivos calculados pelo WASABI e à sua estruturação de modo a permitir acesso rápido ao ser requisitada pelos agentes. Como a busca e a inserção em tabelas *hash* possuem complexidade  $O(1)$ , esta foi a estrutura utilizada para armazenar as emoções dos agentes, em que o nome do agente serve como chave e o valor representa a emoção mais expressiva da verificação mais recente. Todos os agentes em operação possuem acesso aos valores desta lista pois, antes de operarem sobre o *middleware*, a instância da classe é obtida.

É importante perceber que, como apenas a emoção de maior intensidade é armazenada, a utilização desta informação pelos agentes é simplificada, uma vez que se faz necessário verificar apenas se o agente está feliz, e não se está feliz com uma intensidade determinada. Por outro lado, perde-se a possibilidade de se construir planos mais subjetivos, como em: o agente estar incomodado com intensidade 0.54 e, ao mesmo tempo, irritado com intensidade 0.21, por exemplo.

### 3.1 Preparação do WASABI

A posição dos pontos de referência no espaço PAD é atualizada com uma frequência de 50Hz, podendo ser customizada para qualquer valor no intervalo [0, 50]. Esta taxa de atualização, no entanto, não influencia a frequência de envio das informações sobre os estados afetivos através da porta de comunicação UDP, que é definida separadamente.

Ainda que a frequência de execução seja mais pertinente como uma característica do problema em questão do que da própria integração, é importante notar que os tempos de execução do WASABI e do Jason não são sincronizados por definição, o que compromete o *debug*. Já quanto a taxa de envio, ou *sendrate*, recomenda-se que a frequência seja alta o suficiente para que as estruturas que representam os estados afetivos dos agentes sejam atualizadas consistentemente. Como referência, neste trabalho as frequências de execução e envio foram definidas em 25Hz.

Um detalhe importante para o sucesso da implementação consiste na correta criação de novos pontos de referência no espaço PAD, que precisam ser definidos com o mesmo identificador utilizado no Jason durante a execução de uma simulação.

### 3.2 Integração com Jason

Considerando as especificações e requisitos necessários à proposta, o escopo da integração com Jason foi restrito às seguintes etapas:

1. Controle de execução (inicialização e finalização) do *middleware*;
2. Sincronização entre crenças e os estados afetivos correntes;
3. Suporte a geração de novos impulsos e atualizações de dominância em agentes.

Atrelar a inicialização do *middleware* à execução do sistema multiagente (MAS) foi bastante simples, uma vez que a estrutura que define o ambiente faz uso de métodos *init* e *stop* (apresentados abaixo) para estabelecer o que ocorre quando uma simulação é iniciada ou finalizada. O desenvolvimento de um novo cenário de execução poderá levar à extensão desta classe que define o ambiente, permitindo a sobrecarga dos métodos descritos como demonstrado no Código 3 que, ainda que trivial, explicita a criação da thread a partir de uma instância global da classe, respeitando o padrão *Singleton*.

Código 3: Inicialização da thread de atualização do middleware.

```
@Override
public void init(String [] args) {
    initWorld ();

    wasabiHandler = new Thread (WasabiHandler . getInstance ());
    wasabiHandler . start ();
}
@Override
public void stop () {
    super . stop ();
    wasabiHandler . stop ();
}
```

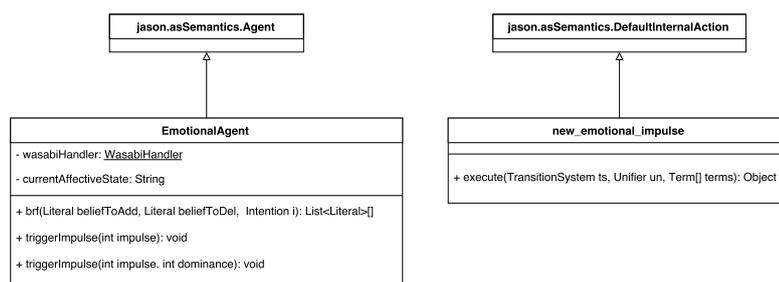


Figura 4: Diagrama das classes *EmotionalAgent* e *newEmotionalImpulse*.

A sincronização de crenças com os estados afetivos, por sua vez, ocorre na etapa de revisão de crenças do ciclo de tomada de decisões dos agentes BDI. Seguindo a mesma estratégia anterior, que também será reproduzida na próxima etapa, a criação de uma nova classe para a definição de agentes, denominada *EmotionalAgent* (Figura 4), estende a definição original do Jason e é suficiente para que se possa sobrescrever a definição das operações que ocorrem no método *brf* (*belief revision function*).

Durante esta etapa, o estado afetivo atualizado é requerido através de uma instância do *WasabiHandler*. Se o valor resultante for igual ao valor armazenado pelo agente (e já repassado como uma crença), nada ocorre, no entanto, se houver variação, o valor armazenado é utilizado para indicar a crença que precisa ser removida e o novo estado afetivo determinará a crença a ser adicionada.

O envio de impulsos, por fim, ocorre através de ações internas que podem ser disparadas utilizando a linguagem Jason durante a própria definição dos planos do agente. Esta facilidade torna-se possível ao definir uma nova classe que represente a ação desejada e estenda a classe *DefaultInternalAction*. De funcionamento bastante simples, a única tarefa desta ação é a de, a partir de uma instância do agente, requisitar o método responsável por enviar um determinado impulso para o WASABI através do *middleware* e, opcionalmente, informar um novo valor de dominância para solicitar sua alteração.

### 3.3 WASABI e Personalidade

Atribuir personalidades aos agentes através do WASABI consiste em manipular a massa e as forças  $F_x$  e  $F_y$  que agem sobre o ponto de referência. Na tese que o originou [6], discute-se um parâmetro  $\alpha$  que, dependendo da magnitude dos valores utilizados para defini-lo, poderia culminar na concepção de agentes letárgicos ou temperamentais. Ainda que a utilização deste parâmetro esteja completamente alinhada aos interesses deste trabalho, não foi encontrada uma interface para manipulação desse valor, seja ela visual ou por código. Por conta deste contratempo, a massa do ponto de referência foi utilizada para prover a diferenciação emocional entre os agentes pois influencia na ação das forças  $F_x$  e  $F_y$  sem que estas precisem ser alteradas.

Para a definição dos pesos a serem aplicados, manteve-se o valor padrão utilizado pelo WASABI como um valor intermediário e outros dois novos valores foram empregados para que fosse possível verificar o comportamento dos agentes nos pontos extremos. Com isso, os pesos utilizados compõem o conjunto:  $\{500, 5000, 9900\}$ . Coerente com os resultados esperados de um sistema massa-mola, com uma massa menor (valor 500), rapidamente o ponto de referência retorna ao seu estado homeostático, enquanto com uma massa maior (valor 9900), este retorno é mais gradual. Na prática, o que deve ocorrer é que, após o recebimento de um impulso positivo, por exemplo, um agente que possui seu comportamento emocional atrelado a um ponto de referência com um peso pequeno para sua massa deve retornar rapidamente para o estado *concentrated*, enquanto o comportamento oposto é esperado de um agente representado por um ponto de referência com peso consideravelmente superior.

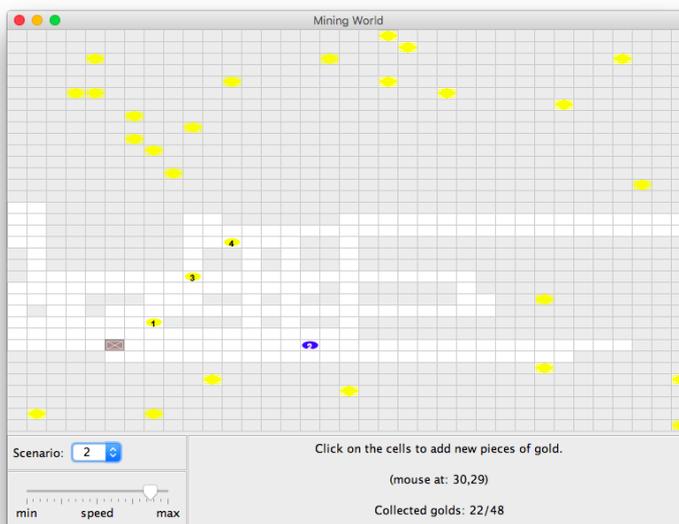


Figura 5: Captura de tela do cenário dos mineiros.

## 4 Cenários de teste e Experimentos

Para verificar a atuação do WASABI como simulador de emoções em um sistema de agentes, foram escolhidos dois cenários, brevemente descritos a seguir.

### 4.1 Cenário A: Mineiros

O primeiro cenário, proposto em [22], é composto por um grupo de agentes encarregados de coletarem pepitas de ouro disponíveis em um ambiente (Figura 5). Embora tenha sido desenvolvida uma versão em que existam equipes distintas que competem entre si, a versão pertinente para este estudo supõe a existência de um ambiente orientado à colaboração que, em decorrência da sua extensão para contemplar o uso de emoções, apresentará alguns componentes que sugerem um funcionamento competitivo entre os agentes.

Para otimizar os resultados, a solução foi desenvolvida considerando a divisão do ambiente em diferentes segmentos, proporcionais a quantidade de agentes disponíveis, de modo com que cada agente seja responsável por percorrer seu respectivo quadrante. Esta divisão, entretanto, não serve como um limite e sim como um guia de navegação para agentes ociosos.

Naturalmente, em ambientes em que se assume a existência de colaboração, é esperado que algum tipo de comunicação ocorra entre os agentes. Neste caso em particular, também para aprimorar os resultados obtidos, existe não apenas a comunicação entre agentes mas pequenas negociações para determinar qual agente será responsável por coletar o recurso. Esta negociação ocorre quando um determinado agente já está comprometido a coletar uma pepita de ouro e, com isso, move-se em direção a este objeto para coletá-lo, por exemplo, mas durante seu trajeto, depara-se com uma outra pepita. Neste momento, como a nova pepita está mais próxima que a anterior, o agente decide por coletar a mais próxima, abandonando o interesse pela anterior, e comunicando aos demais a localização da pepita que desejava obter inicialmente. A figura de um líder é definida como um artifício para simplificar as negociações entre

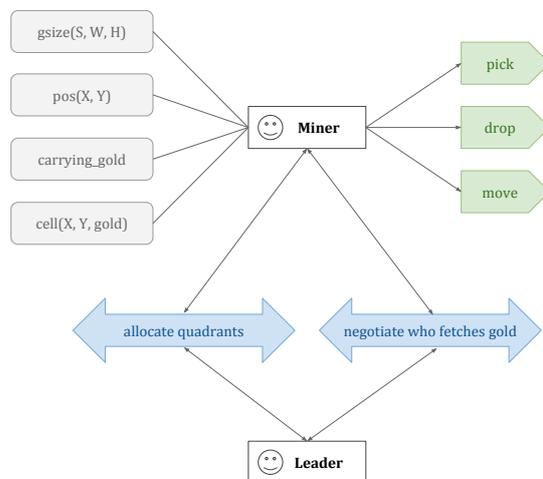


Figura 6: Visão geral da operação de um mineiro no ambiente e de sua interação com o líder na implementação de referência [22].

os agentes. Assim que um agente informa a localização de uma pepita de ouro disponível aos demais, cada agente interessado em coletá-la irá comunicar ao líder sua posição corrente no ambiente. O líder designará como responsável pela coleta aquele que estiver mais próximo do objetivo. Uma visão geral da interação entre um mineiro e o líder e da operação dos mineiros pode ser vista na Figura 6.

Todas as características deste cenário discutidas até aqui, foram definidas como tais pelos seus próprios autores. Não foram realizadas modificações no comportamento padrão dos agentes, apenas extensões para contemplar o uso de emoções, como será discutido na seção a seguir.

**4.1.1 Integração do Componente Emocional** O cenário dos mineiros foi estendido de modo que, ao coletar uma pepita de ouro, um estímulo positivo seja gerado para o agente coletor. Ao receber uma determinada quantidade de estímulos positivos, e neste caso a quantidade necessária varia de acordo com a personalidade do agente, um agente eventualmente deve encontrar-se em um estado afetivo que denote *felicidade*. Quando está feliz, entretanto, este agente tornar-se ganancioso e, no lugar de informar para os demais agentes a localização de uma pepita que ele não pode carregar, ele mente, informando uma localização aleatória. Deste modo, a pepita de ouro avistada tende a ficar reservada apenas para si, como demonstrado esquematicamente no Código 4.

Código 4: Influência de crenças emocionais no comportamento do agente.

```

+cell(X, Y, gold) : not committed(gold(X, Y)) & happy
  <- +gold(X, Y);
     . broadcast(tell, gold(RandomX, RandomY)).

+cell(X, Y, gold) : not committed(gold(X, Y))
  <- +gold(X, Y);
     . broadcast(tell, gold(X, Y)).
    
```

No entanto, como uma penalização indireta, ao perceber que na localização informada não existe nenhuma pepita de ouro disponível, o agente que recebeu a informação passa a tratar o emissor como mentiroso e deixa de informar para este agente a localização de pepitas disponíveis enquanto estiver ocupado. Aqui é o líder quem concentra a informação, pois de fato é ele quem recebe a mensagem do agente enganado e exclui o agente mentiroso da relação de agentes com os quais o agente enganado compartilha informações sobre suas pepitas não coletadas.

**4.1.2 Experimentos do cenário A** - As simulações foram realizadas tomando como medida de qualidade do desempenho do agente a quantidade de pepitas de ouro obtidas por ele. Foram realizadas 10 simulações e o valor considerado é a média de objetos coletados individualmente. O desvio padrão também foi calculado para determinar a consistência dos resultados.

Todos os experimentos foram executados utilizando a mesma frequência de execução, até que um limite total de 100 objetos fosse coletado. Uma nova simulação parte de uma nova instância do WASABI, garantindo que não haja influência de simulações anteriores. Em ambos os casos, o desempenho de quatro agentes distintos é verificado, um BDI e três agentes EBDI com valores de massa iguais a 500, 5000 e 9900, conforme discutido na seção 3.3.

Os resultados obtidos a partir da simulação e apresentados na Tabela 1, foram bastante consistentes e sugerem que a utilização de agentes emocionais pode efetivamente promover benefícios.

A maior consistência entre as simulações é observada no agente BDI, visto que apresenta o menor desvio entre os agentes avaliados. No entanto, este também apresenta o pior desempenho. Para que se possa compreender as razões que fundamentam os resultados obtidos é preciso considerar que:

- O tamanho do mapa utilizado (36x36) favorece a acumulação de pepitas de ouro próximas umas das outras, o que faz com que um agente emocional gere impulsos positivos com bastante frequência, o que o deixa feliz e consequentemente ganancioso;
- Como os agentes emocionais tendem a estar felizes com frequência, mesmo que os impulsos negativos ao descobrirem que foram enganados tenha valência máxima, isso não deve deixá-los irritados diretamente pois a influência que os impulsos oferecem varia de acordo com o estado corrente do agente e um agente feliz é mais sensível a impulsos positivos e menos sensível a impulsos negativos.
- Por não considerar emoções, o agente BDI não torna-se ganancioso, o que faz com que continue sempre divulgando, ao estar ocupado, a localização de pepitas de ouro encontradas para todos os demais agentes. Não há a distinção entre agentes que mentiram ou não, uma vez que a identificação de uma mentira está atrelada ao agente estar irritado.

Quanto maior a massa do ponto de referência no WASABI, maior será o tempo em que um agente permanecerá em um mesmo estado afetivo. Sabendo que, por conta das características do cenário os agentes tendem a manter-se felizes, o que leva diretamente à ganância, percebe-se que o melhor desempenho decorreu justamente dos agentes que mais mentiram e guardaram a localização das pepitas de ouro encontradas para si.

Este comportamento fica ainda mais claro ao se analisar a coleta de pepitas de ouro no decorrer do tempo. A partir do exemplo apresentado na Figura 7, é possível perceber que o agente EBDI representado no WASABI pelo ponto de referência de maior peso, acumulou consideravelmente mais ouro que o agente BDI durante todo o tempo de simulação, o que sugere a coleta de ouro em sequência, compatível com o comportamento de guardar a localização apenas para si, no lugar de compartilhar com os demais.

Tabela 1: Resultados obtidos a partir do cenário dos mineiros.

Total de Pepitas de Ouro Obtidas				
Experimento	BDI	EBDI-500	EBDI-5000	EBDI-9900
<b>1</b>	20	23	26	31
<b>2</b>	24	22	26	28
<b>3</b>	22	26	26	26
<b>4</b>	20	29	23	28
<b>5</b>	21	23	25	31
<b>6</b>	22	27	26	25
<b>7</b>	23	27	22	28
<b>8</b>	22	24	27	27
<b>9</b>	22	24	27	27
<b>10</b>	24	26	22	28
<b>Média</b>	<b>22</b>	<b>25</b>	<b>26</b>	<b>28</b>
Variância	2,000	4,989	3,778	3,656
<b>Desvio Padrão</b>	<b>1,414</b>	<b>2,234</b>	<b>1,944</b>	<b>1,912</b>
Variância Populacional	7,6			

Visando uma melhor avaliação do uso de personalidade, o experimento foi refeito considerando 4 agentes emocionais com o mesmo peso de massa, ou seja, com personalidades similares.

Correspondendo aos extremos avaliados, os valores de massa utilizados foram iguais a 500 e 9900. Na prática, um valor de massa consideravelmente baixo nesta modelagem favorecerá um ambiente de colaboração, uma vez que menos agentes devem mentir para os demais, enquanto um valor de massa alto deve propiciar o oposto.

Os resultados obtidos com este enfoque, resumidos na Tabela 2, mostraram-se bastante homogêneos, independentemente do valor de massa utilizado. Com isso, pode-se perceber que qualquer possível vantagem de um comportamento individualista, como verificado anteriormente, deixa de existir quando todos os agentes compartilham-se do mesmo modo.

A pequena variação no número de pepitas de ouro coletadas, para uma diferença substancial no valor de massa da representação emocional do agente, é explicada pelo próprio ambiente, que não impõem desafios significativos que possam ser superados apenas através da colaboração. Além disso, pela definição da arquitetura WASABI, ao receber um estímulo, não há uma transição gradual para o novo estado homeostático, o que torna o comportamento emocional menos flexível e, naturalmente, mais previsível.

#### 4.2 Cenário B: Coletores de Lixo

Compartilhando o mesmo objetivo geral do cenário anterior, que é o de coletar determinados objetos presentes no mundo, este cenário difere por ter sido concebido com o propósito de trabalhar com emoções. Originalmente proposto – sem que tenha sido implementado – em [17], a problemática proposta consiste na coleta de lixos que surgem pela cidade por agentes que precisam também manter-se atentos a ataques de cachorros errantes que vagam livremente pelo ambiente.

Os coletores de lixo possuem um campo de visão limitado, como pode ser visto na Figura 8, e são representados por uma área contígua aos agentes (pequeno triângulo em frente ao agente). Os lixos

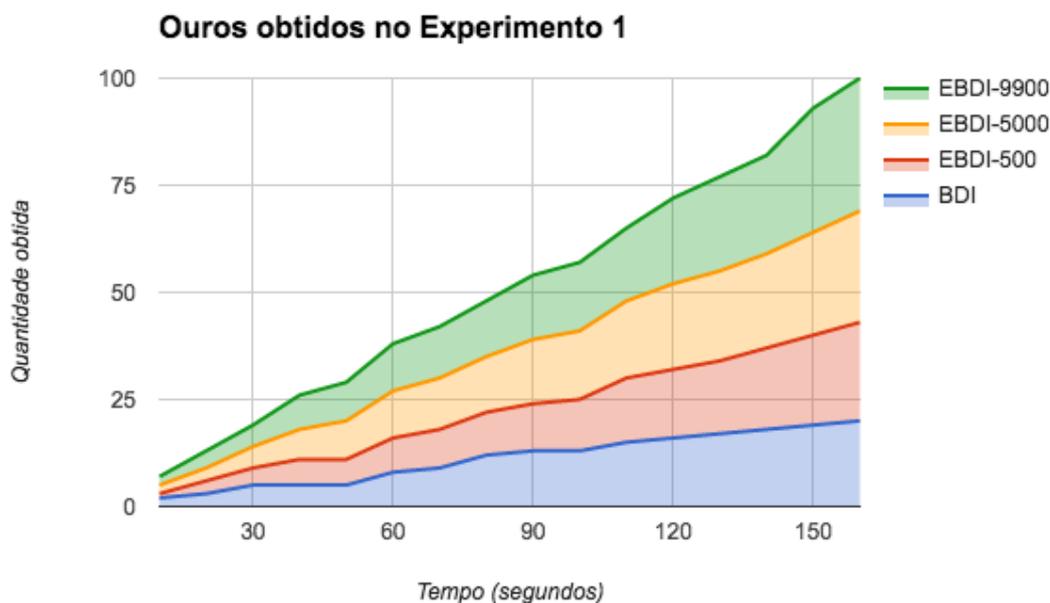


Figura 7: Quantidade de pepitas de ouro coletadas por agente no primeiro experimento.

Tabela 2: Resultados da utilização de agentes emocionais de personalidades similares.

Média de Pepitas de Ouro Obtidas		
Agente	EBDI-500	EBDI-9900
1	25,5	25
2	23	25
3	26	24,5
4	26	25,5
<b>Desvio Padrão Médio</b>	<b>1,484</b>	<b>1,959</b>

disponíveis pela cidade estão retratados pelos círculos maiores (em verde), enquanto os círculos menores (em vermelho) simbolizam os cachorros raivosos. Coletores possuem uma quantidade finita de energia que se esvai ao percorrer o ambiente ou mais rapidamente ao ser mordido por um cachorro. Quando um coletor está cansado, é preciso sentar para que sua disposição se restabeleça e ele possa voltar a percorrer o seu caminho.

Estar amedrontado pode fazer com que um agente abandone a intenção de coletar um lixo em vista, e este estado, pode ser desencadeado ao avistar um cachorro ou, com ainda mais intensidade, ao ser mordido por um, ou seja, ocupar um espaço adjacente a um cachorro no ambiente.

Ainda que a implementação deste cenário (resumida visualmente através da Figura 9) tenha sido realizada com a intenção de ser o mais próximo possível ao que havia sido proposto por [17], existem determinados aspectos que ou não puderam ser implementados como especificado ou foram simplificados por não representarem ganhos ao presente trabalho, como:

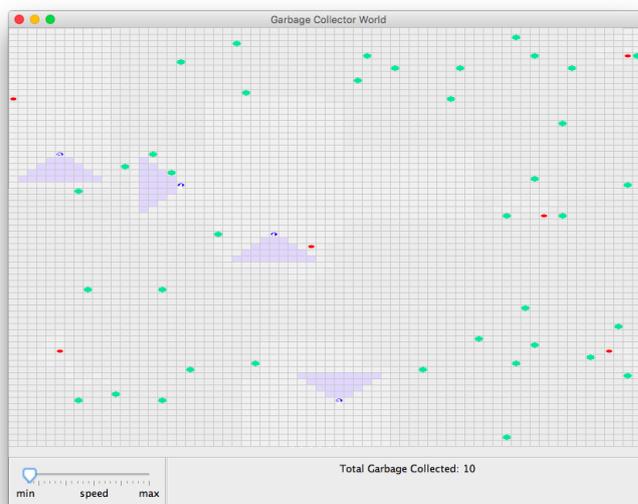


Figura 8: Captura de tela do cenário dos coletores de lixo.

- **Movimentos possíveis:** Como idealizado inicialmente, deveriam existir três categorias distintas de movimentos, compreendendo de movimentos para frente em uma célula e rotações de 90° até a pulos para os lados e pulos sobre duas células. O nível de desgaste do agente após executar algum dos movimentos era definido dependendo da categoria à qual pertencesse. Como essa modelagem não tende a afetar consideravelmente o estado do agente e nem mesmo o ambiente, apenas as movimentações mais importantes foram definidas sem que houvesse distinção entre categorias.
- **Ataque de cachorros:** Ser mordido por um cachorro, na versão proposta inicialmente, significava que um agente ocuparia, no mesmo instante, um espaço já ocupado por um cachorro. Fazendo uso da estrutura padrão do Jason para modelagem do mundo, um agente, por definição, não pode ocupar o mesmo espaço que outro, uma vez que a estrutura de dados utilizada para armazenar os objetos no mundo faz uso de uma máscara binária para denotar o tipo de objeto que está disposto em uma determinada posição. A fim de respeitar as leis da física e evitar a extensão das definições do Jason, escolheu-se alterar as regras para permitir um ataque sempre que um agente e um cachorro ocupassem células contíguas.

**4.2.1 Experimentos do cenário B** Para este cenário, novamente a medida de qualidade foi dada pela quantidade de lixos coletados e os experimentos foram reproduzidos dez vezes, sendo os resultados apresentados a média destes experimentos. A frequência de execução também foi mantida, usando como limite a coleta de 100 objetos. Novamente são considerados quatro agentes distintos, um BDI e três agentes EBDI com valores de massa iguais a 500, 5000 e 9900.

Diferentemente do que ocorreu no cenário anterior, os resultados obtidos a partir da simulação do problema dos coletores de lixo não demonstra uma evolução progressiva entre os tipos de agentes utilizados, tampouco a mesma consistência em relação ao desvio padrão, como pode ser verificado na Tabela 2.

Além da geração aleatória de lixos e do tamanho do ambiente consideravelmente maior, este cenário conta também com cachorros que movem-se de maneira imprevisível no ambiente, afetando diretamente a movimentação dos agentes. Era esperado, com isso, que ao ser comparado ao cenário anterior, houvesse

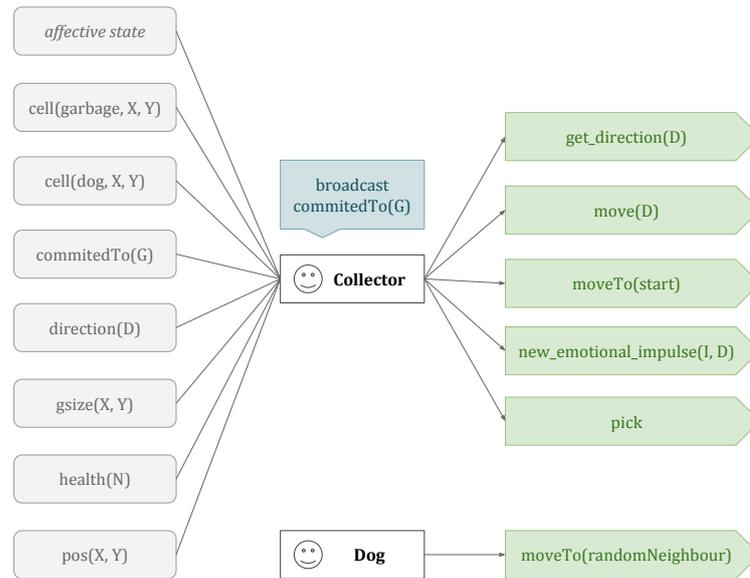


Figura 9: Visão geral do cenário dos coletores de lixo, com as principais crenças e ações de coletores e cachorros.

maior variação entre os experimentos na simulação atual. Surpreendentemente, entretanto, a utilização de emoções deixou de ser uma garantia de melhor desempenho, visto que o agente representado pelo ponto de referência com peso 500 para sua massa obteve o pior resultado entre os agentes simulados e o agente com o maior valor de peso para sua massa foi apenas ligeiramente superior a versão BDI.

Diante das regras impostas no problema dos coletores de lixo e dos resultados obtidos nos experimentos realizados, pode-se concluir que a configuração de um agente que gere o melhor desempenho dependerá da utilização de emoções de modo que ele não seja demasiadamente sensível ao ponto de evitar qualquer cachorro percebido, mas também não insensível o suficiente para apenas ignorá-los.

## 5 Considerações Finais

Este trabalho apresentou um *middleware* que permite que estados emocionais sejam adicionados como crenças e utilizados no ciclo de raciocínio do agente, influenciando na tomada de decisão. Diferentemente dos trabalhos de [19] e [20] que enfatizam como emoções são modeladas, ou da proposta de [4] que define um modelo de agente emocionais, o trabalho ora apresentado integra o modelo de emoções WASABI com a ferramenta para desenvolvimento de agentes Jason.

Ainda que uma avaliação mais profunda dependa de um volume muito maior de experimentos, a verificação feita neste trabalho fornece um panorama bastante interessante sobre a influência da personalidade ao se modelar agentes emocionais. É importante observar que a determinação de uma "personalidade" ao agente, pode resultar em um comportamento que impeça o agente de atingir seus objetivos (modelando um agente que tem medo, por exemplo). Tais variações mostram que, ao utilizar-se de agentes dotados de emoções, não basta definir o comportamento do agente e as limitações do meio. Todas as nuances que constituem a personalidade emocional dos agentes estão diretamente associadas ao seu de-

Tabela 3: Resultados obtidos a partir do cenário dos coletores de lixo.

Total de Lixos Coletados				
Experimento	BDI	EBDI-500	EBDI-5000	EBDI-9900
1	22	18	32	28
2	26	20	25	29
3	26	23	34	17
4	21	14	25	40
5	33	21	24	22
6	17	26	30	27
7	34	27	21	18
8	21	34	32	13
9	21	18	27	34
10	35	11	34	20
<b>Média</b>	<b>24</b>	<b>20,5</b>	<b>28,5</b>	<b>24,5</b>
Variância	40,489	44,622	21,156	69,511
<b>Desvio Padrão</b>	<b>6,363</b>	<b>6,680</b>	<b>4,600</b>	<b>8,337</b>
Variância Populacional	46,15			

sempenho no ambiente. Para determinados casos de uso, agentes com uma menor variação entre estados afetivos, como no cenário apresentado, podem atingir seus objetivos com maior facilidade, no entanto, quando expostos a um ambiente distinto, como no cenário dos coletores de lixo, este agente pode ter um desempenho bastante similar ou mesmo inferior ao de um agente BDI.

Durante a implementação, diversos desafios foram encontrados e nem todos puderam ser endereçados adequadamente. Um dos mais importantes está na própria definição da arquitetura WASABI que, por ser constituída de um sistema massa-mola, promove um retorno gradual ao estado homeostático de cada agente, mas não possibilita que a evolução para um determinado estado, após receber um impulso qualquer, ocorra também de maneira gradual, o que impede a modelagem de agentes que mudam gradativamente seu estado emocional. Apesar desta limitação, integrar o arcabouço Jason com a arquitetura WASABI mostrou-se um conjunto bastante versátil para a construção de agentes EBDI.

Embora integradas, as aplicações WASABI e Jason permanecem operando de maneira independente e com frequências de execução possivelmente distintas, característica esta que pode vir a ser um problema quando da utilização do *mind inspector* do Jason. Uma possível melhoria seria permitir a definição de um passo customizado, através da ampliação da API do Jason.

Além de se considerar a utilização de um outro sistema que não um de massa-mola para determinar as forças que agem sobre um ponto de referência, seria possível fazer com que fossem utilizados quatro sistemas massa-mola, no lugar de apenas dois para as forças  $F_x$  e  $F_y$ . Os dois primeiros seriam mantidos como originalmente definidos, enquanto os dois últimos passariam a existir assim que tenha sido calculada uma nova posição para o ponto no espaço PAD, com sua origem neste novo ponto, permitindo assim a variação gradual também ao receber impulsos.

O próprio *middleware* responsável pela integração WASABI-Jason poderia ser otimizado para, no lugar de enviar um impulso para a arquitetura tão logo o tenha recebido, passasse a adicioná-los em um *buffer* e a enviá-los todos ao mesmo tempo como parte do *loop* de verificação de novas mensagens.

Visto que a interação entre o WASABI e o Jason contempla apenas alterações na base de crenças

dos agentes, o próprio componente de raciocínio do Jason pode ser melhor explorado, de modo a permitir a influência das emoções de maneira mais direta.

Por fim, uma avaliação mais profunda quanto aos quesitos de escalabilidade e tempo de resposta para sistemas envolvendo um grande número de agentes também deve ser conduzida.

## 6 Contribuição dos Autores

- Thiago Ângelo Gelaim: Base teórica sobre emoções em agentes e modelos de Emoções, investigação sobre o funcionamento do WASABI, auxílio no projeto de integração WASABI-JASON; aprimoramento do texto sobre emoções em agentes e modelos de emoções; revisão do texto.
- Jerusa Marchi: Proposta da realização da integração; auxílio nas deficiências e projeto de integração WASABI-JASON; orientação do trabalho; revisão e aprimoramento do texto do artigo.
- Diego Couto: Estudo sobre modelos de emoções e sobre a linguagem Jason; Implementação da WASABI Handler; escolha e implementação dos cenários de testes; execução dos testes; análise dos resultados; escrita do artigo.

## Referências

- [1] CREED, C.; BEALE, R. Agent-based modeling and simulation of the emotional and behavioral dynamics of human civilians during emergency situations. *Proceedings of the 2005 Workshop on the role of emotion in HCI*, v. 2, p. 266–281, 2005.
- [2] BELHAJ, M.; KEBAIR, F.; SAID, L. B. Agent-based modeling and simulation of the emotional and behavioral dynamics of human civilians during emergency situations. In: *German Conference on Multiagent System Technologies*. Cham: Springer, 2014. p. 266–281.
- [3] GOETZ, J.; KIESLER, S.; POWERS, A. Matching robot appearance and behavior to tasks to improve human-robot cooperation. In: *The 12th IEEE International Workshop on Robot and Human Interactive Communication*. Washington, USA: IEEE, 2003. p. 55–60.
- [4] JIANG, H.; VIDAL, J. M.; HUHNS, M. N. EBDI: an architecture for emotional agents. In: *Proceedings of the 6th international joint conference on autonomous agents and multiagent systems*. New York, USA: ACM, 2007.
- [5] LEJMI-RIAHI, H.; KEBAIR, F.; SAID, L. B. Agent decision-making under uncertainty: Towards a new e-bdi agent architecture based on immediate and expected emotions. *International Journal of Computer Theory and Engineering*, v. 6, n. 3, p. 254–259, 2007.
- [6] BECKER-ASANO, C. *WASABI: Affect simulation for agents with believable interactivity*. Clifton, USA: IOS Press, 2008.
- [7] BORDINI, R. H.; HUBNER, J. F.; WOOLDRIDGE, M. *Programming multi-agent systems in AgentSpeak using Jason*. New Jersey, USA: John Wiley & Sons, 2007.
- [8] MARSELLA, S.; GRATCH, J.; PETTA, P. Computational models of emotion. In: *A Blueprint for Affective Computing-A sourcebook and manual*. Oxford, UK: Oxford University Press, 2010. p. 21–46.
- [9] MISNKY, M. *The Society Of Mind*. New York, USA: Simon & Schuster, 1998.

- [10] SCHERER, K. R. Appraisal theory. In: *Handbook of cognition and emotion*. New York, USA: The Guilford Press, 1999. p. 637–663.
- [11] ORTONY, A.; CLORE, G.; COLLINS, A. *Cognitive Structure of Emotions*. Cambridge, UK: Cambridge University Press, 1990.
- [12] MEHRABIAN, A. Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament. *Current Psychology*, v. 14, n. 4, p. 261–292, 1996.
- [13] RUSSELL, J. A. Core affect and the psychological construction of emotion. *Psychological review*, v. 110, n. 1, p. 145–173, 2013.
- [14] EL-NASR, M. S.; YEN, J.; IOERGER, T. R. Flame - fuzzy logic adaptive model of emotions. *Autonomous Agents and Multi-agent systems*, v. 3, n. 3, p. 219–257, 2000.
- [15] GEBHARD, P. Alma: a layered model of affect. In: *Proceedings of the fourth international joint conference on autonomous agents and multiagent systems*. New York, USA: ACM, 2005. p. 29–36.
- [16] MARSELLA, S. C.; GRATCH, J. Ema: A process model of appraisal dynamics. *Cognitive Systems Research*, v. 10, n. 1, p. 70–90, 2009.
- [17] PUICĂ, M.-A.; FLOREA, A.-M. Emotional belief-desire-intention agent model: Previous work and proposed architecture. *International Journal of Advanced Research in Artificial Intelligence*, v. 2, n. 2, p. 1–8, 2013.
- [18] HERNÁNDEZ, D. J. et al. Bdie: a bdi like architecture with emotional capabilities. In: *American Association for Artificial Intelligence Spring Symposium*. Stanford, CA: AAAI, 2004.
- [19] PARUNAK, H. V. D. et al. A model of emotions for situated agents. In: *Proceedings of the fifth international joint conference on autonomous agents and multiagent systems*. New York, USA: ACM, 2006. p. 993–663.
- [20] BITENCOURT, G. K. da S. *Um modelo de confiança baseado em emoções para agentes*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, 2014.
- [21] BECKER-ASANO, C. Wasabi for affect simulation in human-computer interaction. In: *International Workshop on Emotion Representations and Modelling for HCI Systems*. New York, USA: ACM, 2014.
- [22] BORDINI, R. H.; HUBNER, J. F.; TRALAMAZZA, D. M. Using jason to implement a team of gold miners. *Computational Logic in Multi-Agent Systems*, p. 304–313, 2006.