

SIMULAÇÃO MULTIAGENTE E A SUA INTEGRAÇÃO COM SINAIS FISIOLÓGICOS

Rafael Bulsing¹, Carlos Longo¹, Gustavo Carneiro Fleck¹, Diana F. Adamatti¹

1 – Laboratório de Simulação Social e Ambiental – Centro de Ciências Computacionais – Universidade Federal de Rio Grande (LAMSA/C3/FURG) – Rio Grande – RS

rafabulsing@gmail.com, carlos.longo@yahoo.com.br, gustavofleck@furg.br, dianaada@gmail.com

Resumo. *Este artigo apresenta a proposta de integração de sinais fisiológicos em sistemas multiagentes, com o intuito de tornar as aplicações mais realistas e integrar o usuário ao sistema desenvolvido. Para tanto, um dispositivo de coleta de sinais fisiológicos foi desenvolvido em Arduino e integrado ao sistema multiagente via protocolo Javino.*

1. Introdução

A emoção humana é objeto de estudo em inúmeras áreas do conhecimento sendo o meio computacional uma delas. A Inteligência Artificial (IA) é uma área multidisciplinar que visa a simulação da capacidade humana de pensar, tomar decisões, resolver problemas e, nesse caso, sentir. Os Sistemas Multiagentes nos oferecem a possibilidade de simular essas situações através de interações entre agentes em um meio comum (RUSSEL e NORVIG, 2003).

Assim como nas organizações humanas as atividades são feitas por um grupo de pessoas que trabalham de modo cooperativo, onde existem decisões individuais que afetam o grupo, no Sistema Multiagente (SMA) as pessoas são representadas por agentes artificiais, os quais se relacionam em um ambiente de maneira a buscar soluções para problemas de forma cooperativa, compartilhando informações, evitando conflitos e coordenando a execução de atividades (ADAMATTI, 2003).

A Rede Bayesiana é uma ótima ferramenta a ser usada no caso da simulação de emoções, pois nos oferece um raciocínio probabilístico onde podemos adicionar a imprevisibilidade necessária ao agente e, assim, obter um resultado mais plausível (PEARL, 1988). Neves (2014) desenvolveu a Rede Bayesiana de Emoções tendo como base o Modelo OCC (ORTONY et al., 1988).

A placa Arduino é uma plataforma ideal para a criação de protótipos eletrônicos, pois seu microcontrolador pode ser programado através de uma linguagem própria baseada em C/C++, e por existir uma grande variedade de sensores e atuadores compactáveis disponíveis no mercado (ARDUINO, 2017). Além disso, o Arduino é uma plataforma gratuita e aberta.

Este artigo tem como objetivo, através de uma Rede Bayesiana de Emoções baseada no Modelo OCC, apresentar a proposta de integração entre os componentes, com propósito de ter-se um sistema multiagente que faça uso de sinais fisiológicos do usuário, e que esses sinais interfiram na tomada de decisão dos agentes, em tempo de execução. Para testar essa proposta, um jogo computacional simples está sendo desenvolvido, contando com um agente, que representa um NPC (*Non-Player Character*) do jogo, que tem a rede de emoções implementada, e que recebe do jogador (usuário) os sinais fisiológicos durante sua interação com o jogo e indiretamente com o

NPC, tornando o comportamento do agente mais dinâmico. A proposta de integração é apresentada na Figura 1, onde estão os componentes e como eles são integrados.

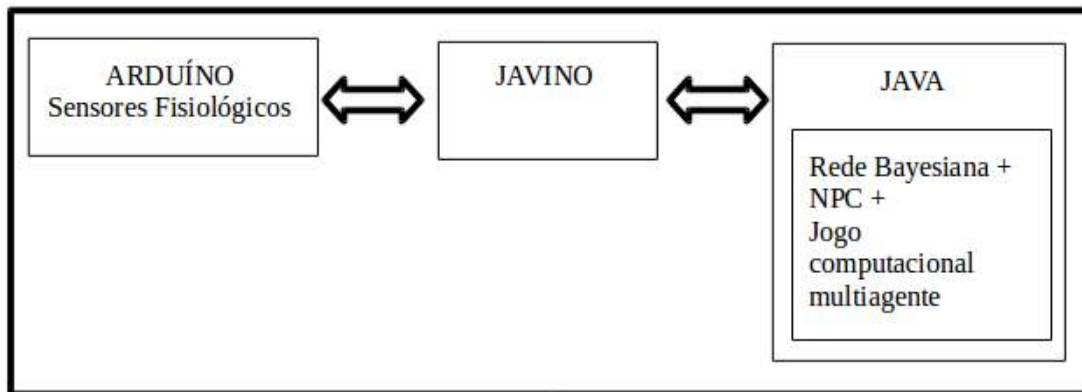


Figura 1. Proposta de Integração de Sinais Fisiológicos ao Sistema Multiagente

O artigo está organizado em 4 seções: na seção 2 é apresentado o referencial teórico do trabalho, tratando da Rede Bayesiana de Emoções, sobre o Arduino e sobre o protocolo de comunicação Javino. Na seção 3 é apresentado o jogo proposto e como será a integração com os sinais fisiológicos. E na seção 4 estão as conclusões preliminares, bem como os trabalhos futuros a serem realizados.

2. Referencial Teórico

2.1 Modelo da Rede Bayesiana de Emoções

A Rede Bayesiana de Emoções foi desenvolvida por Neves (2014) com base no modelo OCC. O Modelo OCC de Emoções foi proposto em (ORTONY et al., 1988). Esse modelo é capaz de identificar a partir de estímulos gerados em um ambiente, as emoções que serão sentidas pelo agente. Ele usa três tipos de geradores de estímulos: eventos, agentes e objetos. Toda emoção gerada é resultado de um ou mais estímulos. O modelo é composto por 22 emoções, onze positivas e onze negativas (NEVES et al, 2016).

Tendo em vista as dificuldades encontradas para modelagem de problemas reais, tais como falta de dados, impossibilidade de coleta e até mesmo imprecisão dos mesmos, a Rede Bayesiana se mostra uma ótima alternativa, pois conta com um raciocínio probabilístico. O tipo de probabilidade utilizado na rede é o condicional que se representa por $P(A|B)$, que significa a probabilidade de que o evento A ocorra dado a ocorrência do evento B (WERHLI, 2007).

A Rede Bayesiana é composta por vários nós e arestas. Na Rede Bayesiana de Emoções esses nós são os estímulos e emoções, tal como é organizado o Modelo OCC de Emoções, como mostra a Figura 2.

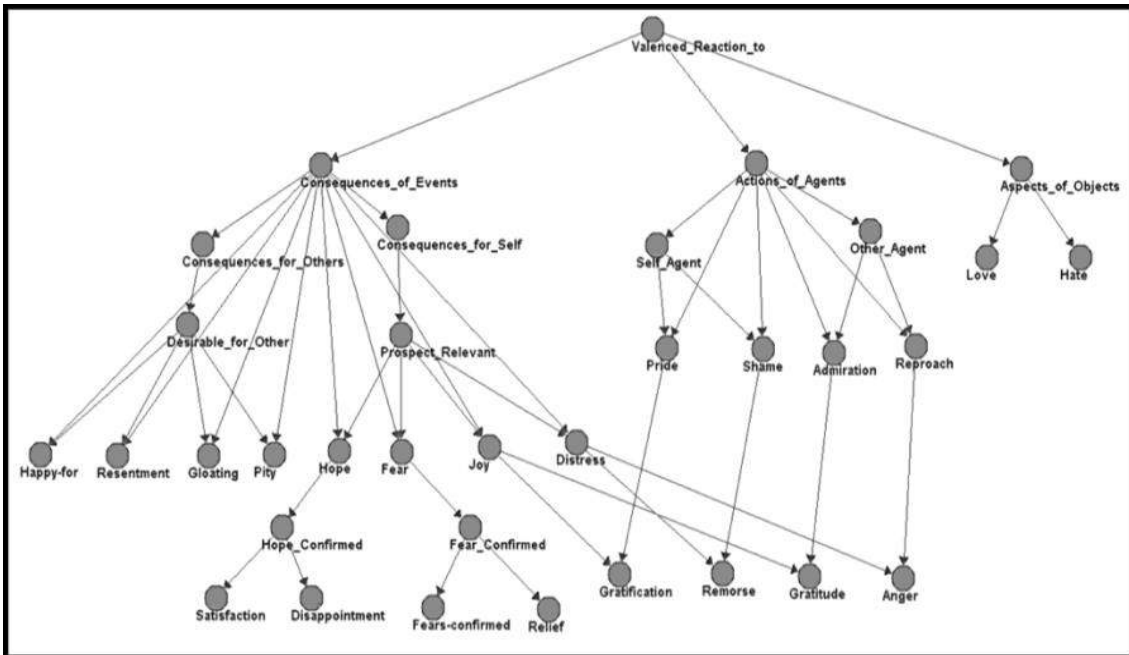


Figura 2. Rede Bayesiana de Emoções (Neves, 2014)

Para modelagem da Rede Bayesiana utilizada neste trabalho, foi usado o software JavaBayes (COZMAN, 2001). Essa ferramenta permite a construção, visualização gráfica e análise de Redes Bayesianas.

2.2 Arduino e Javino

A placa Arduino é uma plataforma ideal para a criação de protótipos eletrônicos, pois seu microcontrolador pode ser programado através de uma linguagem própria baseada em C/C++, e por existir uma grande variedade de sensores e atuadores compatíveis disponíveis no mercado (ARDUINO, 2017).

Para a coleta dos sinais fisiológicos do jogador, o Arduino foi conectado a um sensor de batimentos cardíacos Pulse Sensor SEN-11574 e um sensor de temperatura MLX90614 (Figuras 3 e 4). O sensor de batimentos cardíacos funciona emitindo uma luz na pele do usuário, e medindo as pequenas variações causadas pelo fluxo do sangue na corrente sanguínea. Já o sensor de temperatura, por sua vez, mede a radiação infravermelha emitida pelo jogador, e converte essa informação para a temperatura correspondente em graus Celsius ou Fahrenheit.



Figura 3. Pulse Sensor SEN-11574 **Figura 4. Sensor de Temperatura MLX90614**

Para enviar os dados coletados pela pulseira para o programa Java, foi utilizado Javino, que é um protocolo duplo que permite a troca de mensagens entre o Java e o Arduino através de uma porta serial. O programa Java envia uma mensagem à placa Arduino pedindo que sejam enviados os dados, e o Arduino responde enviando as informações que estão sendo captadas pelos sensores no momento (JAVINO, 2017). Os dados recebidos pelo programa Java são sempre no formato String, porém esses valores podem facilmente ser trabalhados e devidamente convertidos para um formato mais apropriado (inteiro, float, etc). As figuras 5 e 6 apresentam os códigos de Javino para Arduino e de Javino para Java.

```

sketch_oct16a$
1 //JAVINO
2 #include <Javino.h>
3 Javino j; //Objeto Controlador do Javino
4
5 //SENSOR TEMPERATURA
6 #include <Adafruit_MLX90614.h>
7 #include <Wire.h>
8 Adafruit_MLX90614 mlx = Adafruit_MLX90614(); //Objeto Controlador do sensor de temp.
9
10 void setup() {
11     //SENSOR BATIMENTOS
12     mlx.begin();
13
14     //COMUNICAÇÃO SERIAL
15     Serial.begin(9600);
16 }
17
18
19 void loop() {
20     if (j.availablemsg()){
21         answer(j.getmsg());
22     }
23 }
24
25 void answer(String ask){
26     if (ask == "temp")
27         j.sendmsg(String(mlx.readObjectTempC()));
28 }
29

```

Figura 5. Exemplo de código Javino para Arduino

```

*Main.java
1 import br.pro.turing.javino.*;
2
3 public class Main {
4     public static void main(String args[]) {
5
6         Javino j = new Javino();
7
8         String port = "COM3";
9
10        String ask = "temp";
11
12        j.requestData(port, ask);
13
14        if (j.requestData(port, ask)) {
15            System.out.print("Temp: ");
16            System.out.println(j.getData());
17        }
18    }
19 }
20

```

Figura 6. Exemplo de código Javino para Java

3. Estudo do Caso: Jogo Multiagente que utiliza os sinais fisiológicos

O jogo proposto é um labirinto e tem por objetivo que o agente principal, controlado pelo usuário, chegue ao troféu finalizando o jogo. Ao longo desse caminho, o agente se depara com duas placas, uma que o estimula positivamente e outra que o estimula negativamente. Esses estímulos alteram significativamente a jogabilidade, pois se o agente está sobre o efeito de emoções negativas, ele tem sua velocidade reduzida. Caso contrário, se ele está sobre o efeito de estímulos positivos, sua velocidade aumenta. Já o NPC, que recebe valores dos sensores do Arduino, irá interagir com o agente do usuário, desmotivando o mesmo, caso as informações gerem na Rede Bayesiana emoções negativas. Caso sejam emoções positivas, o NPC irá reforçar essas motivações e retirar algum obstáculo do caminho.

Para desenvolver o jogo, foi utilizado o framework multiagente Jason (JASON, 2017), que é desenvolvido em Java e é baseado na arquitetura BDI. A Rede Bayesiana de Emoções também é implementada em Java. Assim, a integração com o Javino é facilitada. A Figura 7 apresenta a interface inicial desenvolvida.

No momento, estamos definindo como cada valor dos sinais fisiológicos afeta as emoções da Rede Bayesiana, e indiretamente o NPC. Contudo, testes de integração entre os componentes (Arduino, Javino e Java) já foram realizados, mostrando a possibilidade da realização do trabalho, em tempo de execução viável, visto que a mudança de comportamento do NPC deverá ocorrer durante a execução do jogo.

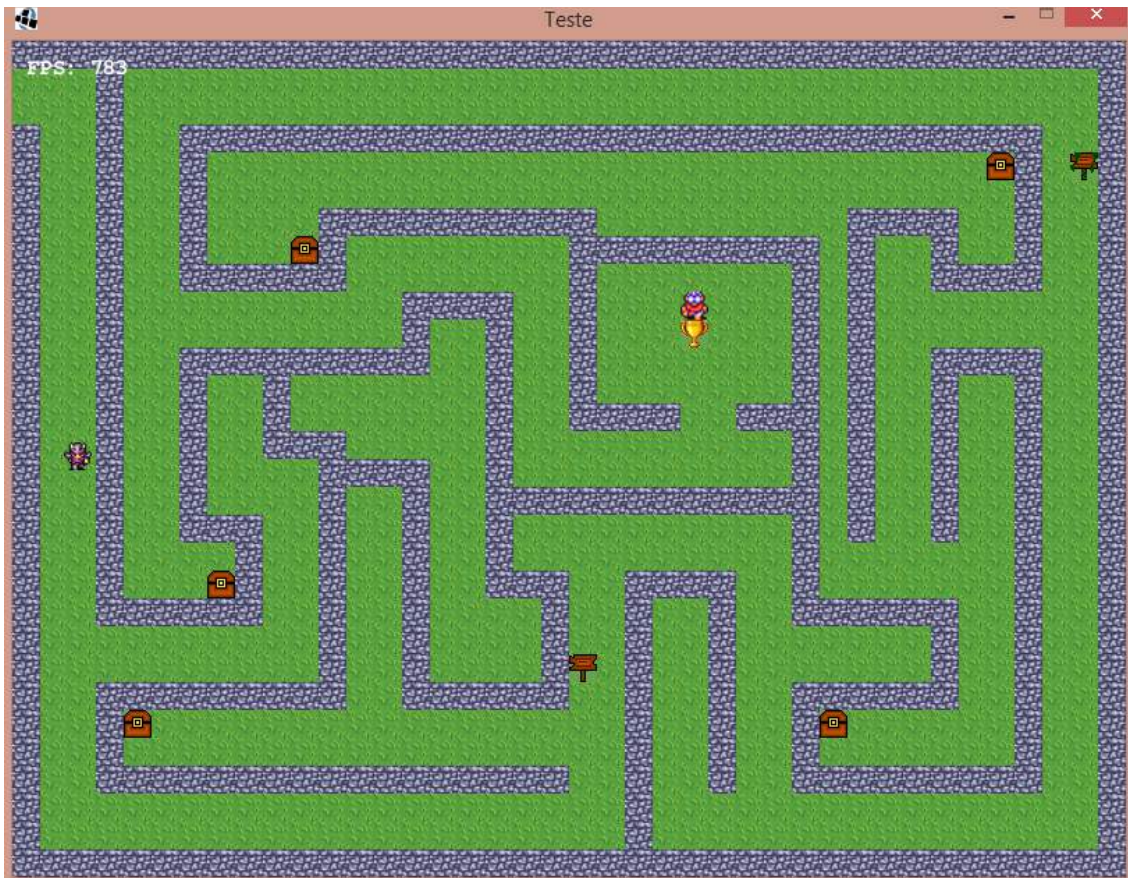


Figura 7. Jogo computacional proposto

4. Conclusões e Trabalhos Futuros

O comportamento humano é determinado por diversas variáveis, muitas delas sem um método de simulação computacional definido, como as emoções, que interferem de forma decisiva no comportamento humano, influenciando a tomada de decisões, ações, memória, atenção, etc. (GRATCH; MARSELHA, 2001).

Através da modelagem de emoções em sistemas multiagentes, baseada em uma Rede Bayesiana, está em desenvolvimento um jogo simples, onde as emoções de um agente (NPC) são alteradas pelos sinais fisiológicos do jogador.

Como resultado final desse projeto, espera-se ter um dispositivo para coleta de sinais fisiológicos e sua integração a Rede Bayesiana de Emoções. Assim, além do jogo computacional que está sendo desenvolvido, outras aplicações poderão fazer uso do dispositivo para tornar mais realísticas as aplicações que interagem com usuários.

6. Referências

- ADAMATTI, D. F. AFRODITE - Ambiente de Simulação Baseado em Agentes com Emoções. Dissertação de Mestrado. Universidade Federal do Rio Grande do Sul. Porto Alegre. 2003.
- ALVAREZ, L. O.; SICHMAN, J. Introdução aos Sistemas Multiagentes. Jornada de Atualização em Informática, Brasília, 1997. 1-38.
- ARDUINO (2017) <http://www.arduino.cc/>, Acesso em 03 de Fevereiro de 2017.
- COZMAN, F. G. Bayesian Networks in Java: User manual and download. JavaBayes, 2001. Disponível em: <<http://www.cs.cmu.edu/~javabayes/Home/index.html>>. Acesso em: 9 Abril 2013.
- FROZZA, R. SIMULA: Ambiente para Desenvolvimento de Sistemas Multiagentes Reativos. Dissertação de Mestrado. Universidade Federal do Rio Grande do Sul. Porto Alegre. 1997.
- GRATCH, J.; MARSELHA, S. Modeling Emotions in the Mission Rehearsal Exercise. Conference on computer generated forces and behavioral representation, 2001. 10.
- JASON (2017). <http://jason.sourceforge.net/wp/>. Acesso em 08 de Fevereiro de 2017.
- JAVINO (2017) <http://sourceforge.net/projects/javino/>. Acesso em 03 de Fevereiro de 2017.
- NEVES, F. S. Modelagem de Emoções Usando Redes Bayesianas. Dissertação de Mestrado. Universidade Federal do Rio Grande. Rio Grande, 2014.
- NEVES, F. S.; WERHLI, A. V.; ADAMATTI, D.F Using Bayesian Networks to Structure the OCC Emotions Model. Journal of Intelligent Computing, v. 7, n. 4, pp.156-171, 2016.
- ORTONY, A.; CLORE, G.; COLLINS, A. The Cognitive Structure of Emotions. Cambridge: Cambridge University Press, 1988.
- PEARL, J. Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. 1a. ed: Morgan Kaufmann, 1988.
- RUSSEL, S.; NORVIG, P. Artificial Intelligence A Modern Approach. 2a. ed. Upper Saddle River, New Jersey: Prentice Hall, 2003.
- WERHLI, A. V. Reconstruction of Gene Regulatory Networks from Postgenomic Data. Tese de doutorado. University of Edinburgh, 2007.