

# Aplicação de Lógica Fuzzy e Redes Neurais em Jogos Computacionais

Maurício T. Pereira Eliane Pozzebon\* Luciana B. Frigo\* Tatiana N. dos Santos\*

Universidade Federal de Santa Catarina, Laboratório de Tecnologias Computacionais, Brasil

\*Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Tecnologias da Informação e Comunicação, Brasil.

## Abstract

The aim of this paper is to present an analysis of two (2) Artificial Intelligence techniques, fuzzy logic and neural networks, applied in two (2) computer games. The paper identifies the characteristics of each of these techniques and evaluates the most appropriate for a particular game. Two examples of games are presented and these resources can be useful to help developers expand their knowledge in the area and also help in choosing between these two AI techniques to other games.

**Palavras-chave:** inteligência artificial, jogos computacionais, redes neurais, lógica fuzzy.

## Informações para Contato:

[mauricio.pereira@msn.com](mailto:mauricio.pereira@msn.com)  
[{eliane Pozzebon, luciana.frigo}@ufsc.br](mailto:{eliane Pozzebon, luciana.frigo}@ufsc.br)  
[tatiana.santos@posgrad.ufsc.br](mailto:tatiana.santos@posgrad.ufsc.br)

## 1. Introdução

Mais que uma indústria de alta tecnologia, os games são considerados atividades artísticas e necessitam de um ser humano com ampla qualificação para a sua produção. A evolução do mercado na área de jogos computacionais é constante e o Brasil, segundo o SEBRAE (2013), já é o quarto maior mercado de games do mundo com cerca de 35 milhões de usuários.

De acordo com BNDES (2014), o movimento do mercado mundial de games foi de US\$ 57 bilhões, comparado ao mercado cinematográfico que atingiu US\$ 31.8 bilhões, em 2010. Como base, o mercado nacional de games movimentou R\$ 5,3 bilhões em 2012, obtendo um crescimento de 32% em relação a 2011 (SEBRAE, 2013).

Com o aumento deste mercado, também aumenta a necessidade de aproximação dos jogos com a realidade, utilizando-se para tal característica as técnicas de inteligência artificial que serão adicionadas durante o processo de criação dos jogos para aumentar

a diversão nos mesmos disponíveis e para proporcionar nos jogos a capacidade de resolver problemas (SOUZA, 2011; FERNANDES et. al., 2015).

A utilização das técnicas de Inteligência Artificial (IA) em jogos computacionais agrega diversas melhorias aos mesmos, como por exemplo, “aumentar a experiência e imersão do jogo, melhorando sua jogabilidade” (SOUZA, 2011).

Desta forma, os jogos exigem cada vez mais uma inteligência elaborada para os seus personagens e também para o enredo e os cenários. Assim, surgem problemas a ser resolvido, o que acaba por disseminar e aprofundar cada vez mais as áreas de estudo da inteligência artificial.

Neste artigo são apresentados 2 (duas) análises a partir de 2 (dois) jogos, sobre a utilização de técnicas de IA nos mesmos. Esta análise será útil para auxiliar os desenvolvedores na escolha da técnica mais adequada para cada tipo de jogo específico.

O artigo está organizado da seguinte maneira, na segunda seção são apresentados trabalhos desenvolvidos de forma equivalente ou similar ao apresentado neste artigo; na terceira seção são descritas as técnicas de inteligência artificial aplicada a jogos computacionais; a quarta seção apresenta os resultados e discussões após a análise das técnicas de IA; a quinta seção apresenta as considerações finais do artigo e por fim, as referências utilizadas.

## 2. Trabalhos Relacionados

Como trabalhos correlatos, pode-se citar o artigo intitulado ‘Estudo do Uso de Técnicas de Inteligência Artificial em Jogos 2D’, de Soraia Teixeira Barbosa, Janaína Veiga e Carlos Vitor de Alencar Carvalho, publicado na Revista Eletrônica TECCEN em 2012 (BARBOSA, VEIGA e CARVALHO, 2012).

Neste artigo, os autores apresentam um estudo sobre sistema especialistas aplicados em um sistema de batalhas de um jogo de RPG (Jogo de Interpretação de Papéis, do inglês Role-Playing Game); neste caso, a única diferença com o presente artigo é apresentar-se na escolha da técnica de IA utilizada, uma vez que o presente artigo utiliza Lógica Fuzzy e Redes Neurais e

o artigo citado acima aplica no jogo a técnica de Sistemas Especialista (BARBOSA, VEIGA e CARVALHO, 2012).

Outro trabalho relacionado é o intitulado, 'Inteligência Artificial Aplicada aos Jogos Eletrônicos', de Fernando Filgueiras e Sérgio Muinhos Barroso Lima, publicado na Revista Eletrônica da Faculdade Metodista Granbery no ano de 2012 (FILGUEIRAS e LIMA, 2012).

O objetivo deste artigo é apresentar um estudo de caso sobre as técnicas de IA utilizadas em diversas funcionalidades e implementações do jogo Left 4 Dead. Novamente, a única diferença apresentada encontra-se na técnica de IA utilizada, que no caso do artigo acima citado foi constatado que grande parte das funções do jogo utiliza o algoritmo de busca A\* (lê-se a-estrela) (FILGUEIRAS e LIMA, 2012).

Também pode-se citar como trabalho correlato, o artigo intitulado 'Analysis of artificial intelligence in computer game', de Ailin Zeng publicado em BioTechnology An Indian Journal em 2014 (ZENG, 2014).

Este trabalho apresenta um estudo da aplicação de IA em jogos computacionais,

“com base em sua distinção funcional, distinção de desempenho e comportamento [...] além de analisar o princípio da construção de sistemas de IA em jogos computacionais” (ZENG, 2014).

O artigo ainda enumera e discute sobre as principais técnicas de IA usadas em jogos computacionais. Em comparação ao presente artigo, a diferença encontra-se na forma como foram abordadas as técnicas de IA, ou seja, no presente artigo se fez de maneira objetiva, focando em duas técnicas, uma para cada jogo; já no artigo citado é apresentado de uma maneira geral as principais técnicas utilizadas para o desenvolvimento de jogos eletrônicos

### 3. Aplicação das técnicas de IA

Esta seção apresenta as análises da aplicação das técnicas de Inteligência Artificial – Redes Neurais (RN) e Lógica Fuzzy (LF) – em dois jogos eletrônicos.

A aplicação da IA aos jogos não se preocupa com a resposta que o sistema gera, nem em como o sistema deve funcionar internamente, mas “o interesse está em como o sistema atua, e não como ele pensa” (Fernandes et. al., 2015),

Nas próximas subseções então, será abordada a metodologia de como os jogos atuam a partir das características de cada técnica de IA aplicada às funções do mesmo.

#### 3.1. Exemplo de jogo utilizando Redes Neurais

Conceição (2009) propõe uma aplicação que consiste em um jogo chamado Jogo da Velha 3D em Pinos; este jogo é uma variação do famoso Jogo da Velha.

O Jogo da Velha 3D em Pinos possui muita semelhança com o jogo da velha jogado no “papel” onde a lógica do jogo é praticamente a mesma: colocar os símbolos X (xis) ou O (bolinha) em linha reta (seja horizontal, vertical ou diagonal) de três símbolos do mesmo tipo que determina a pontuação.

O Jogo da Velha 3D em Pinos é um jogo de tabuleiro para ser jogado por duas pessoas, uma sendo o adversário da outra. Ele consiste em jogadas de pontuação em turnos, ou seja, cada um dos oponentes tem a sua vez de jogar e em sua vez ele tem a chance de marcar pontos.

O tabuleiro tem o formato geométrico quadrado (ou similar) em sua superfície, essa superfície é dividida em 9 partes de mesma área, contabilizando assim 9 pequenos quadrados. No centro de cada quadrado há um pino cilíndrico (Figura 1), contabilizando assim nove pinos distribuídos uniformemente na superfície.

O jogo contém, além do tabuleiro, 24 (vinte e quatro) peças com forma de argola e tamanhos iguais, este formato é para que as peças possam ser encaixadas nos pinos. As peças são divididas em duas cores, contabilizando assim, 12 (doze) peças para cada jogador.

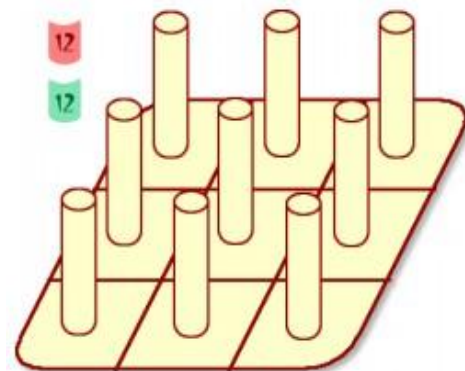


Figura 1: Jogo da velha 3D em pinos  
Fonte: Extraído de Conceição (2009)

Como no Jogo da Velha, a pontuação é feita quando um jogador coloca 3 (três) de suas peças em linha reta. No Jogo da Velha há 9 (nove) locais onde podemos colocar uma peça X (xis) ou O (bolinha), esses locais são chamados de casas e estas numeradas de 1 (um) a 9 (nove) (Figura 2).

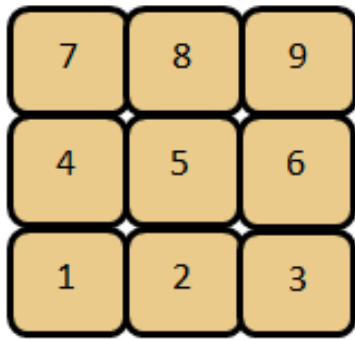


Figura 2: Apresentação das casas

Conforme afirma Conceição (2009), o jogo da velha possui 8 (oito) formas possíveis de ganhar o jogo (Figura 3).

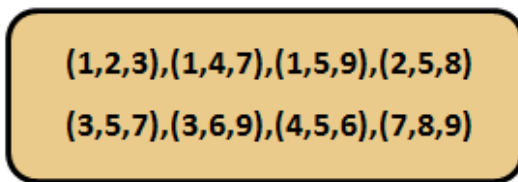


Figura 3: Possibilidades do jogo da velha

Já no Jogo da Velha 3D em Pinos, os jogadores jogam até as 24 (vinte e quatro) peças se esgotarem e no final são contabilizados os pontos marcados de acordo com as combinações feitas por cada jogador (Figura 4).

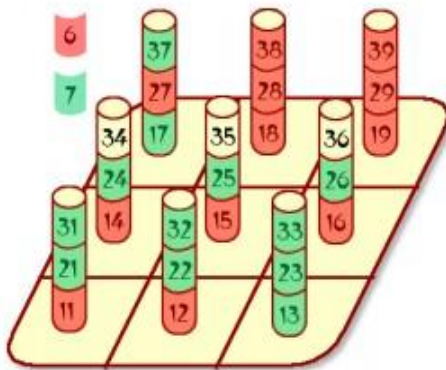


Figura 4: Exemplo de pontuação jogo da velha 3D  
Fonte: Extraído de Conceição (2009)

Como é possível constatar na Figura 4, cada pino possui três níveis (1, 2 e 3 – de baixo para cima, respectivamente), e é sugestivo que cada pino possa receber três argolas; sendo assim, há 49 (quarenta e nove) combinações possíveis.

“A rede tem como objetivo aprender a jogar o Jogo da Velha 3D em Pinos e para isso ela terá que aprender as regras do jogo, armar jogadas, pontuar e bloquear pontos.” (CONCEIÇÃO, 2009).

A estrutura da rede contém três camadas (por causa dos três níveis dos pinos), i) uma camada de entrada que contém 27 (vinte e sete) neurônios (devido a quantidade de posições (9) multiplicado pela quantidade de níveis (3) de cada pino), ii) uma camada oculta que contém 49 (quarenta e nove) neurônios (devido a quantidade de pontuações possíveis) e iii) uma camada de saída que contém 9 (nove) neurônios (que é a quantidade de pinos).

Segundo Conceição (2009), no Jogo da Velha 3D em Pinos, cada posição pode estar em apenas um dos três estados possíveis: 1) sem peça; 2) com a minha peça; e 3) com a peça do oponente.

Para isso serão contabilizadas todas as posições, onde em cada uma há um neurônio responsável na camada de entrada e cada neurônio receberá um valor. Este valor pode ser:

- 0 (zero): quando não há peça para aquele neurônio;
- 1 (um): quando há peça para aquele neurônio e a peça é do jogador;
- -1 (um negativo): quando há peça para aquele neurônio e a peça é do oponente.

As posições onde as peças se encaixam vão de 1 (um) a 27 (vinte e sete) em uma contagem sequencial pela numeração levando em consideração seu nível, sendo assim, haverá peças do primeiro nível numeradas de 11 (onze) a 19 (dezenove); segundo nível numeradas de 21 (vinte e um) a 29 (vinte e nove) e terceiro nível numeradas de 31 (trinta e um) a 39 (trinta e nove) (CONCEIÇÃO, 2009).

No treinamento da rede, foi utilizado o algoritmo de aprendizagem por correção de erro também conhecido como backpropagation, pois se encaixa perfeitamente à estrutura da rede (CONCEIÇÃO, 2009).

Um dos princípios que Conceição (2009) aborda, é que a rede aprenda a jogar sem que um humano intervenha no jogo para que o treinamento não seja lento. Para isso, foi proposta a criação de mais três redes para que pudessem ser oponentes. Logo que os oponentes são criados, é necessário que algum deles saiba a regra do jogo. Para isso, foi criado um juiz, que auxilia as redes na disputa.

O papel do juiz é, a cada turno, avaliar a situação do jogo e a escolha da rede. Como o jogo possui 9 pinos, cada um com 3 (três) posições, dando um total de 27 (vinte e sete) posições, a rede não escolhe em que posição vai jogar e sim em que pino vai jogar, então a rede escolhe um dos 9 (nove) pinos para jogar, e o juiz só precisa avaliar as 9 (nove) posições possíveis.

- Melhor posição: O juiz atribui o valor 1 (um);
- Pior posição: O juiz atribui o valor de -1 (menos um);
- Outras posições: o valor é proporcional, menor que 1 (um) e maior que 1 (um).

Para saber a melhor posição em cada turno para a situação, o juiz varre as 9 (nove) posições. Para cada posição ele analisa as combinações possíveis e verifica:

- Se for possível fazer ponto: Ótimo;
- Bloquear ponto do adversário: Ótimo;
- Tentar caminho para um ponto na próxima jogada: Muito Bom;
- Iniciar um caminho de ponto: Bom,
- Colocar uma peça na combinação onde o adversário já tem uma peça: Ruim;
- Colocar uma peça na combinação onde a rede e o adversário já tem uma peça finalizando assim as posições do pino e não fazendo pontuação alguma: Muito Ruim;
- Colocar uma peça num pino sem espaço vago: Péssimo.

O juiz também faz análise da posição de onde se está fazendo a jogada, quantas combinações são possíveis fazer com aquela posição mesmo que no momento não façam pontos, isso porque é importante pegar posições estratégicas. O juiz pode analisar o jogo e pontuar cada jogada, pois ele tem conhecimento sobre todas as possibilidades de pontuação e as posições de influência. (CONCEIÇÃO, 2009).

Atualmente o juiz considera apenas a situação atual do jogo para criticar a rede, não levando em consideração jogadas passadas e movimentações futuras, por isso é provável que a rede aprenda a cair em armadilhas. (CONCEIÇÃO, 2009).

Para que estas estratégias sejam executadas, existem os algoritmos responsáveis pela execução. O algoritmo do juiz é o responsável por qualificar a jogada da rede e utiliza o Algoritmo 2 e o Algoritmo 3 como auxílio buscando neles as combinações possíveis e as porcentagens das posições de influência respectivamente (CONCEIÇÃO, 2009).

### 3.2 Exemplo de jogo utilizando Lógica Fuzzy

Boscatto, Nakamiti e Toledo (2012), implementaram um jogo onde possui um sistema de controle tempo-real para uma nave autônoma onde a estratégia de controle é testada em um cenário similar ao jogo River Raid.

Neste jogo, a nave deve voar em um local composto por vários obstáculos sem se chocar com obstáculos que ficam em constante movimentação no cenário (Figura 5).



Figura 5: Tela do jogo River Raid  
Fonte: Extraído de BOSCATTO; NAKAMITI e TOLEDO, 2012

A lógica Fuzzy é utilizada na elaboração deste jogo para definir as distâncias aproximadas entre a nave e os demais objetos do cenário do jogo.

A arquitetura do sistema (Figura 6) é composta por diversos módulos. O módulo XNA utiliza as bibliotecas da plataforma de mesmo nome baseada no framework .NET com alguns métodos básicos como: i) Load() que carrega os dados para a memória principal; ii) Unload() que limpar a memória; iii) Update() que atualiza as posições dos objetos e iv) Draw() que desenha a tabela.

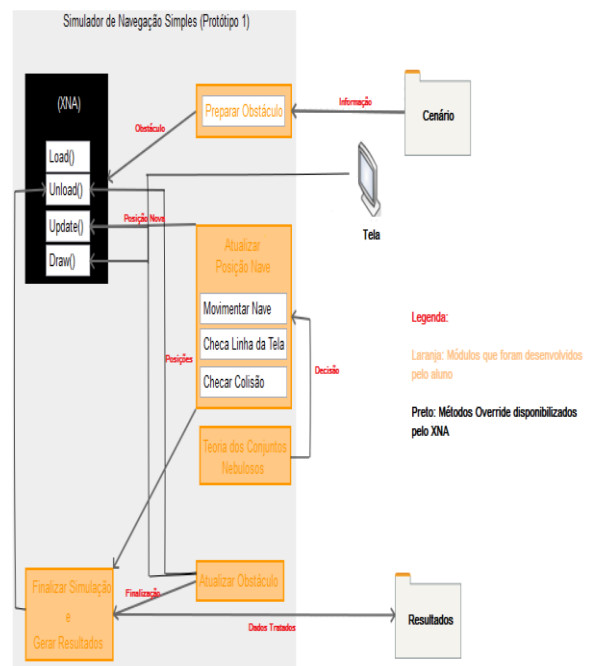


Figura 6: Arquitetura do jogo River Raid  
Fonte: Extraído de BOSCATTO; NAKAMITI e TOLEDO, 2012

Além disso, a arquitetura apresenta os módulos descritos a seguir:

- O módulo “Preparar Obstáculo” obtém informações do cenário como os obstáculos e as respectivas coordenadas;
- O módulo “Atualizar Posição da Nave” obtém informações da movimentação da nave e é dividido em três métodos: i) MovimentarNave() que atualiza a posição da nave principal a partir dos comandos de alteração na direção e intensidade de movimento; ii) ChecarLimites() que verifica se a nave não ultrapassa os limites da tela previamente estabelecidos e iii) ChecarColisão() que verifica a colisão com os objetos do cenário e a intensidade das colisões observando a sobreposição dos pixels para maior precisão em relação aos objetos não retangulares;
- O módulo “Atualizar Obstáculo” tem a função de mover obstáculos da tela para baixo, passando a sensação que o cenário está em constante movimento, fazendo com que a nave pareça se deslocar para frente;
- Os módulos “Finalizar Simulação” e “Gerar Resultados” liberam a memória, pois os objetos a ocuparam durante a simulação e gera um arquivo texto com um relatório com as informações como o número de objetos ultrapassados e o tempo total de simulação;
- Os módulos “Teoria dos Conjuntos Nebulosos” (Fuzzy), “Raciocínio Baseado em Casos” e “Algoritmos Genéticos”, são responsáveis pela orientação da nave de acordo com o cenário;
- O módulo “Fuzzy” realiza o treinamento do sistema, gerando a base inicial de casos que será utilizada pelos outros módulos adaptativos.

Basicamente, ele converte as distâncias entre a nave e os demais objetos do cenário, usando uma função de pertinência “Distância” (Figura 7) (BOSCATTO; NAKAMITI e TOLEDO, 2012).

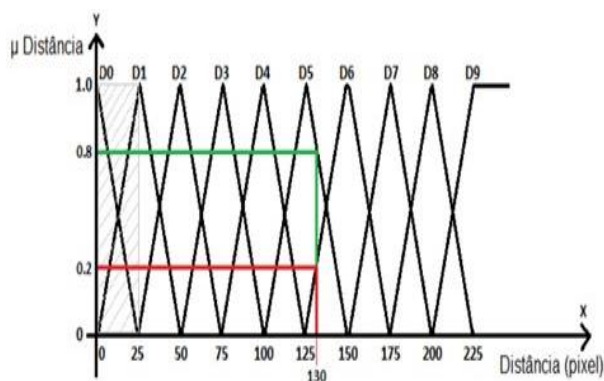


Figura 7: Representação gráfica da variável Distância  
 Fonte: Extraído de BOSCATTO; NAKAMITI e TOLEDO, 2012

As funções D0 a D9, apresentadas na Figura 7, são utilizadas no lugar de termos linguísticos (Ex: Baixa, Média, Alta, etc).

A aplicação das regras Fuzzy e o cálculo da ação a ser tomada é obtido através do centro da área. A fase de treinamento Fuzzy é executada até que gere 50 (cinquenta) casos (valor configurável), que servirão de entrada ao mecanismo híbrido genético-baseado em casos, facilitando sua convergência. (BOSCATTO; NAKAMITI e TOLEDO, 2012).

Conforme os autores deste jogo, em todos os testes em que foram utilizadas as regras de produção Fuzzy, o mesmo resultado foi obtido (a nave conseguiu ou não atravessar o cenário sem colisões). Cabe observar que foram utilizadas regras de produção estáticas, isto é, não foi gerada nova regra em tempo de execução.

Nos testes com o sistema completo, o sistema colidiu com os obstáculos nos cenários mais complexos nas primeiras tentativas. Depois, em todos os casos, o sistema aprendeu uma solução e prosseguiu nos testes sem colidir.

#### 4. Resultados e Discussões

Se tratando das redes neurais, é possível afirmar que elas tentam simular o comportamento do cérebro humano, realizando aprendizado de acordo com as interações que esta faz onde é possível obter uma métrica binária (um ou zero) que podem ser mensurados e utilizados para tomada de decisões (PEREIRA, 2013).

O que difere da lógica Fuzzy é que ela é utilizada para resolver problemas onde os números e as decisões passam a ser tomadas sobre outra perspectiva, como por exemplo, muito, pouco, grande, pequeno, etc., ou seja, os números não são precisos e são feitos com valores aproximados. Essas variáveis não-numéricas são trabalhadas para determinar um padrão de decisões que deve ser feita de acordo com um valor aproximado.

Para os resultados ficarem mais visíveis, seguem abaixo as características de cada técnica conforme os autores citados no decorrer do capítulo que justificam os argumentos citados anteriormente.

Tabela 1: Características das técnicas de IA trabalhadas

Técnica de IA	Características
Redes Neurais	Permite recuperação de informação; Possui tolerância a falhas; Capaz de realizar aprendizado através de exemplos fornecidos; Generalizações; Distribuição da informação de maneira organizada; Não é baseada em regras

## Lógica Fuzzy

Manipula de forma eficaz informações incertas ou incompletas;  
 Sistemas baseados em casos;  
 Possui a capacidade de generalizar a lógica clássica binária de forma a ser aproveitada em situações em que haja um alto grau de incerteza;  
 Traduz em termos matemáticos a informação imprecisa expressa por um conjunto de regras linguísticas  
 Trabalha com valores aproximados;  
 É capaz de resolver processos complexos, de modo que os sistemas de controle resultantes proporcionam um resultado mais acurado;  
 Redução da complexidade de um projeto a um ponto em que problemas anteriormente intratáveis passam agora a ser solúveis.

Como podemos observar, as técnicas de Inteligência Artificial (Redes Neurais e Lógica Fuzzy) permitem que os jogos sejam capazes de resolver problemas considerados complexos.

Como aprendizado pode-se concluir que a rede neural proporciona características ao jogo o mais próximo do real e também que a utilização dos resultados aproximados propostos pela lógica Fuzzy faz com que o jogo possa tomar decisões que aumentam a realidade e o torne mais inteligente e realista.

Como contribuição científica percebe-se que foi possível constatar que cada técnica possui várias características e que elas devem ser analisadas e estudadas para aplicá-las num jogo específico de forma mais adequada.

## 5. Conclusões

A utilização de inteligência artificial em jogos computacionais agrega reais melhorias a estes softwares porque é possível aumentar a experiência e imersão do jogo.

Neste trabalho, buscou-se as informações referente as técnicas de Inteligência Artificial com intuito de propiciar aos leitores um subsídio para que estes ampliem seu conhecimento na área para poder determinar qual a técnica – das duas estudadas – possa ser aplicada ao seu jogo.

A técnica de Redes Neurais é utilizada para problemas que simulam o comportamento humano nos jogos, por exemplo, um personagem do jogo que é autônomo e precisa tomar a decisão para vencer um adversário. Neste personagem terá que ser implementado o treinamento da rede neural para auxiliar este personagem na escolha da melhor ação durante o jogo.

A entrada dos dados deste treinamento da RN será alimentada com as diferentes situações que o personagem enfrentará no jogo e quanto maior o volume de dados para treinar a rede neural, consequentemente mais inteligente serão as decisões futuras deste personagem.

A técnica de lógica Fuzzy é utilizada quando os problemas não são bem definidos num jogo, por exemplo, situações onde o personagem tem dúvidas quanto a mensurar os dados de entrada para tomar a decisão. Os dados de entrada variam entre 0 e 1, quente ou frio, rápido ou devagar, alto ou baixo, etc.

O personagem terá que calcular o grau de pertinência destes dados para tomar a decisão mais adequada para as ações futuras.

Por fim, é imprescindível analisar em cada jogo as características para decidir qual a melhor ou melhores técnicas de Inteligência Artificial a serem aplicadas.

Para trabalhos futuros, sugere-se aplicação de um estudo de caso onde possa ser engloba diversas técnicas, como por exemplo, a técnica híbrida neuro-fuzzy; sugere-se também a análise de outras técnicas de Inteligência Artificial utilizadas nos jogos, como algoritmos genéticos, raciocínio baseado em casos, entre outras.

## Referências

- BNDES. Grupo de Estudos e Desenvolvimento da Indústria de Games. Relatório Final: Mapeamento da Indústria Brasileira e Global de Jogos Digitais. 2014
- BOSCATTO, A. B.; NAKAMITI, G.; TOLEDO, C. T. In: SBGames. 11., 2012, Campinas. Fuzzy River Raid – Uma abordagem adaptativa para navegação Autônoma. Brasília, 2012. Disponível em: <[http://sbgames.org/sbgames2012/proceedings/papers/comp/putacao/comp-short\\_11.pdf](http://sbgames.org/sbgames2012/proceedings/papers/comp/putacao/comp-short_11.pdf)> Acesso em: 22 set. 2013
- BARBOSA, S. T.; VEIGA, J.; CARVALHO, C. V. A., 2012, Estudo do Uso de Técnicas de Inteligência Artificial em Jogos 2D. Revista Eletrônica TECCEN, Vassouras, v. 5, n. 1, p. 5-20.
- CONCEIÇÃO, D. T., 2009, Redes Neurais Artificiais Aplicadas ao Jogo da Velha 3D em Pinos. Trabalho de Conclusão (Graduação) – Tecnólogo da Informação e da Comunicação, Instituto Superior de Tecnologia em Ciência da Computação, Petrópolis – RJ, 85p.
- FERNANDES, A. M.; CAMARA, B. H. P.; PASCHOAL, A. R. e DAMASCENO, E. F. 2015, Inteligência Artificial aplicada a Jogos de Tabuleiro com Realidade Aumentada. In: 15º Workshop de Realidade Virtual Aumentada, Marília-SP, v. 1.
- FILGUEIRAS, F.; LIMA, S. M. B. 2012, Inteligência Artificial Aplicada aos Jogos Eletrônicos. Revista Eletrônica da Faculdade Metodista Granbery, n. 13.
- PEREIRA, M. T. 2013, Análise de técnicas de Inteligência Artificial aplicadas em jogos computacionais. Trabalho de Conclusão de Curso (Graduação), Universidade Federal de Santa Catarina – Campus Araranguá.

SEBRAE. Brasil tem o maior mercado de games no mundo em 2012. Disponível em: <<http://www.sebrae2014.com.br/Sebrae2014/Alertas/Brazil-tem-o-maior-mercado-de-games-no-mundo-em-2012#.Ug6DMtLFXwx>> Acesso em: 16 de agosto de 2013.

SOUZA, M., 2011, Influência dos jogos no campo da Inteligência Artificial. Centro de Educação Superior do Alto Vale do Itajaí, Universidade do Estado de Santa Catarina, Florianópolis - SC.

ZENG, A., 2014, Analysis of artificial intelligence in computer games. BioTechnology An Indian Journal, v. 10, ed. 13.