



FACULDADE METODISTA
GRANBERY
1889 • Colégio - Faculdade

Curso de Sistemas de Informação – NOV/2013

INTELIGÊNCIA ARTIFICIAL EM VIDEO GAME ENGINES

Bruno Bartholomeu Bessa¹
Davi Del Esporte Pessanha Filgueiras²
Marco Aurélio Pereira da Silva³
Sérgio Muinhos Barroso Lima⁴

RESUMO

O campo da ciência da computação que estuda o comportamento das máquinas e a forma com que são programadas, a Inteligência Artificial (IA), tem tido um crescimento muito significativo, ainda mais se tratando de jogos eletrônicos. Este artigo se propõe a listar as facilidades que uma *game engine* traz na hora de se desenvolver um jogo eletrônico com IA avançada, e a realizar um estudo de caso sobre a utilização de Inteligência Artificial na *Unreal Engine*.

PALAVRAS-CHAVE: Inteligência Artificial, Jogos Eletrônicos, Motor de Jogos, *Unreal Engine*.

ABSTRACT

The field of computer science that studies the behavior of the machines and the way they are programmed, the Artificial Intelligence (AI), has had a very significant growth, especially in video games environment. This article intends to show the facilities a game engine can bring to the development of a video game with advanced AI, and to conduct a case study about the use of Artificial Intelligence in the Unreal Engine.

KEY-WORDS: Artificial Intelligence, Video Games, Game Engine, Unreal Engine.

Juiz de Fora - MG
2013

¹ Aluno do Curso de Sistemas de Informação, Faculdade Metodista Granbery, bruno_barth_bessa@yahoo.com.br

² Aluno do Curso de Sistemas de Informação, Faculdade Metodista Granbery, davifilgueiras@hotmail.com

³ Aluno do Curso de Sistemas de Informação, Faculdade Metodista Granbery, marcoaurelio.fjf@gmail.com

⁵ Mestre em Ciência da Computação, UNICAMP, smblima@granbery.edu.br

1 INTRODUÇÃO

No início da computação, era comum para os pesquisadores da área se perguntarem se, um dia, uma máquina teria a capacidade de pensar como um ser humano. Alan Turing, homem de grande influência no desenvolvimento da ciência da computação, teria dito em sua publicação para a revista *Mind, Computing Machinery and Intelligence*, que tal indagação é absurda, visto que o conceito de "pensamento" é muito vago. Para Turing, era muito mais interessante ponderar se seria possível uma máquina se passar por um ser humano sem ser notada (TURING, 2013).

Hoje em dia podemos perceber que a resposta para essa dúvida já está sendo desvendada, e isso graças ao avanço da ciência na área da Inteligência Artificial, que cresceu tanto a ponto de permitir aos computadores tomarem decisões baseadas em fatores externos e até mesmo em experiências passadas. Um avanço tão grande se deu não só pela melhoria na tecnologia, mas também pelas mais diversas aplicações da Inteligência Artificial: no setor de finanças, nos hospitais, na aviação, entre muitas outras. Este artigo focará na usabilidade da Inteligência Artificial nos *video games*, ou jogos eletrônicos.

Levando em consideração a grande dificuldade de se construir algoritmos avançados e eficientes de Inteligência Artificial em um curto espaço de tempo, é importante frisar ainda como são úteis as *game engines*, utilizadas para o desenvolvimento mais rápido e organizado de jogos eletrônicos. Neste artigo, também serão descritas a usabilidade das *game engines* e como elas ajudam na aplicabilidade de IA nos jogos eletrônicos.

Como exemplo do avanço da Inteligência Artificial e sua utilização em *game engines*, este artigo apresentará um estudo de caso sobre o *Turing Game/BotPrize*, que tem como objetivo replicar o experimento do Teste de Turing ao desenvolver IAs avançadas na *Unreal Engine* para reproduzir o comportamento humano.

Ao fim do artigo, pretende-se destacar a importância das *game engines* na aplicação da Inteligência Artificial nos jogos eletrônicos, bem como o avanço nessa área trará grandes benefícios aos futuros jogos e à indústria de videogames como um todo.

2 INTELIGÊNCIA ARTIFICIAL

2.1 CRIAÇÃO E EVOLUÇÃO

Como dito anteriormente, para Alan Turing (2013) era interessante saber se máquinas poderiam se passar por humanos sem serem notadas. Ele propôs um pequeno experimento que envolvia três "jogadores" em ambientes isolados: um homem, uma mulher e um interrogador. O objetivo do teste seria o interrogador determinar quem é o homem e quem é a mulher a partir de perguntas e respostas. A partir dessa premissa, Turing se perguntou o que aconteceria caso uma máquina tomasse o lugar de um dos interrogados, e se o resultado dos testes seria muito diferente do que se realizado apenas com seres humanos. Esse desafio seria o "Jogo da Imitação" (TURING, 2013).

Steve Harnard, em seu artigo *Minds, Machines and Turing: The Indistinguishability of the Indistinguishable* (2001), explica que o objetivo do Teste de Turing seria verificar se máquinas poderiam agir de maneira indistinta dos seres pensantes. Portanto, se Inteligência Artificial é o programa de computador capaz de emular ações e pensamentos, tanto humanos, quanto racionais (NORVIG, RUSSEL, 1995), podemos concluir que era a algo semelhante a isso que Turing se referia.

Desde que essas novas dúvidas foram levantadas, o homem tem vislumbrado a ideia de criar um ser pensante artificial. Bruce Buchanan (2013) informa que, na ficção científica, homens mecânicos, máquinas inteligentes e robôs eram e ainda são elementos comuns utilizados para nos fazer pensar sobre nossa própria humanidade. Autores dessa literatura, como Jules Verne, Isaac Asimov, L. Frank Braum e muitos outros, têm sido grandes inspirações para os pesquisadores da área da Inteligência Artificial desde o século XIX (BUCHANAN, 2013).

Jack Copeland relata, no documento digital *A Brief History of Computing* (2013a), que o primeiro programa de Inteligência Artificial funcional foi um "jogador" de damas programado por Christopher Strachey, em 1952. Buchanan (2013) também cita o xadrez, outro jogo de lógica, como mais um foco importante no estudo e no avanço da IA durante suas primeiras décadas.

Outro programa de damas, este criado por Arthur Samuel em meados dos anos 50, foi um marco gigantesco para a história da Inteligência Artificial. Apesar das limitações de memória e processamento geradas pela tecnologia da época, o programa

de Samuel tinha a capacidade de aprender com suas experiências, gerando, assim, novas táticas de jogo e melhorando suas habilidades como jogador (SAMUEL, 1959).

O método utilizado por Samuel é um exemplo de uso do *algoritmo genético* (AG), termo cunhado pelo pesquisador da Universidade de Michigan John Holland em 1975 (COPELAND, 2013b). De acordo com Copeland (2013b), o AG utiliza como base o processo de seleção natural da evolução para que novas "gerações" de um programa possam se adaptar melhor ao seu propósito definido.

Com o passar dos anos, o estudo da Inteligência Artificial tem feito progressos cada vez mais impressionantes. Além da capacidade de resolver problemas pré-determinados, Inteligências Artificiais também têm sido desenvolvidas com o propósito de raciocinar por si só (COPELAND, 2013b). Hoje, programas de IA têm atingido um nível racional capaz de inferir e fazer analogias, o que significa que o raciocínio das máquinas inteligentes contemporâneas não está mais limitado estritamente a deduções lógicas (BUCHANAN, 2013).

2.2 APLICAÇÃO EM JOGOS ELETRÔNICOS

A aplicação de Inteligência Artificial em jogos eletrônicos tem sido um dos assuntos mais evidenciados dessa ciência. Esse sucesso se dá pelo fato de que o videogame, no cenário atual, é entretenimento para todas as faixas etárias. (BRANCO; MALFATTI; LAMAR, 2012). Segundo uma pesquisa realizada nos Estados Unidos, pela Newzoo, 145 milhões de pessoas, o equivalente a 47% da população, passaram, no ano de 2011, 215 milhões de horas por dia jogando *videogames*.

A utilização da IA em jogos eletrônicos é necessária por motivos simples. Michael Samyn (2013), designer e diretor de jogos da série *Tale of Tales*, afirma que os jogos digitais são um tipo de mídia que se destaca das outras com as quais sempre estivemos acostumados. Segundo o desenvolvedor, os *games* têm como principal característica a interatividade do jogador com o que lhe é apresentado. Com isso em mente, pode-se perceber como a Inteligência Artificial é construída para aperfeiçoar as experiências desse jogador no mundo do jogo. (CHARLES et. al, 2008).

A IA também é crucial na hora de criar um desafio balanceado, já que um adversário controlado pelo computador não pode vencer sempre e nem perder sempre. Esse difícil equilíbrio é o maior foco das pesquisas de Inteligência Artificial em videogames nos dias atuais (CHARLES et. al, 2008).

É importante frisar que há uma certa diferença entre a Inteligência Artificial das pesquisas científicas e aquela utilizada nos jogos eletrônicos. Segundo Schwab (2009), a IA de um jogo está muito mais focada no *resultado* de suas escolhas do que em seu *processo* de decisão. O objetivo de muitas IAs em jogos eletrônicos seria, portanto, o de criar uma *ilusão* de comportamento inteligente para o bem da experiência do jogador, ao invés de desenvolver uma inteligência em si (POTTINGER, 2013).

No início da chamada "era de ouro dos videogames", situada entre os anos 70 e 80, jogos famosos como *Space Invaders*, *Asteroids*, *Pac Man* e *Donkey Kong* foram lançados. Nestes, a IA era praticamente inexistente, baseada apenas no rastreamento de seus adversários humanos e movimentos repetidos controlados por máquinas de estado finitos (CHARLES et al., 2008).

Schwab (2009) explica que os jogos primitivos funcionavam dessa maneira para que o jogador pudesse encontrar padrões em seus inimigos e, assim, derrotá-los mais facilmente. Na verdade, os padrões eram mais leves, não necessitavam de cálculos pesados e podiam ser escritos com poucas linhas de código, sendo então perfeitos para as limitações da época (SCHWAB, 2009).

Outra forma encontrada pelos desenvolvedores para se implementar uma IA mais avançada nos jogos foi a "IA trapaceira" (CHARLES et al., 2008). O termo é usado quando a Inteligência Artificial faz uso de informações do jogo indisponíveis ao jogador para tomar decisões (SCHWAB, 2009). Um exemplo seria, no jogo de tênis *Pong* (Figura 1), informações sobre a velocidade da bola e a posição do adversário serem utilizadas para realizar um movimento (CHARLES et al., 2008).



Figura 1 - Pong (1972). Fonte: GIZMODO, 2013.

Nos jogos modernos, técnicas de IA baseadas em padrões e trapaça já foram quase completamente abandonadas. Isso se deve ao fato de que o visual dos gráficos, principal aspecto de um *game* analisado pelos jogadores e críticos nas últimas gerações

de consoles, chegou a um nível tão excepcional (Figura 2) que outros elementos têm roubado a cena, tornando-se mais interessantes (SCHWAB, 2009).

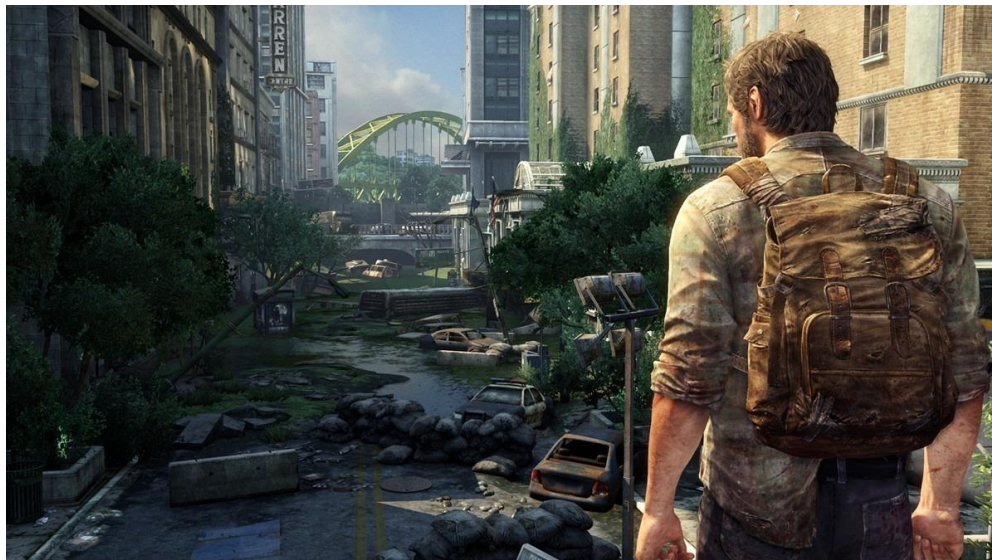


Figura 2 - Tela do jogo "The Last of Us" para PlayStation 3, demonstrando seu visual fantástico.
Fonte: GAMESPOT, 2013

Com *hardwares* cada vez mais poderosos nos consoles e nos PCs, além de visuais extremamente realistas, os jogos eletrônicos dos dias atuais têm como uma de suas principais exigências uma Inteligência Artificial avançada – elevando ainda mais a qualidade da imersão do jogador em seu mundo (CHARLES et al., 2008).

3 GAME ENGINES

3.1 DEFINIÇÃO

As *game engines* surgiram em meados da década de 90, quando jogos de FPS⁵ como o *Doom*, da produtora id Software, começaram a separar claramente os elementos mais técnicos, como gráficos tridimensionais e cálculos de física e de colisão, de recursos mais leves e visuais, como a arte de personagens, cenários e regras de jogo. Com o tempo, desenvolvedores começaram a criar novos personagens, objetos, armas e veículos e lançar jogos novos sem muitas modificações no software principal (GREGORY, 2009).

Hoje em dia, as desenvolvedoras fazem uso das *game engines* para reaproveitar elementos cruciais de seus jogos passados na construção de novos títulos (GREGORY, 2009). Com isso, não há mais a antiga necessidade de recomeçar do zero quando o

⁵ First Person Shooter. Em português: Tiro em Primeira Pessoa

desenvolvimento de outro *game* se inicia, e assim o investimento em tempo e dinheiro diminui significativamente (LILLY, 2013).

As *game engines* possibilitam a união de codificações pesadas e recursos gráficos na criação de um jogo completo. Elas trazem múltiplas funções que auxiliam o desenvolvedor e o *level designer*⁶, dando-lhes a capacidade de observar, em tempo real, as consequências que as mudanças feitas têm dentro do jogo (WILMORE, 2013), conforme ilustra a Figura 3.

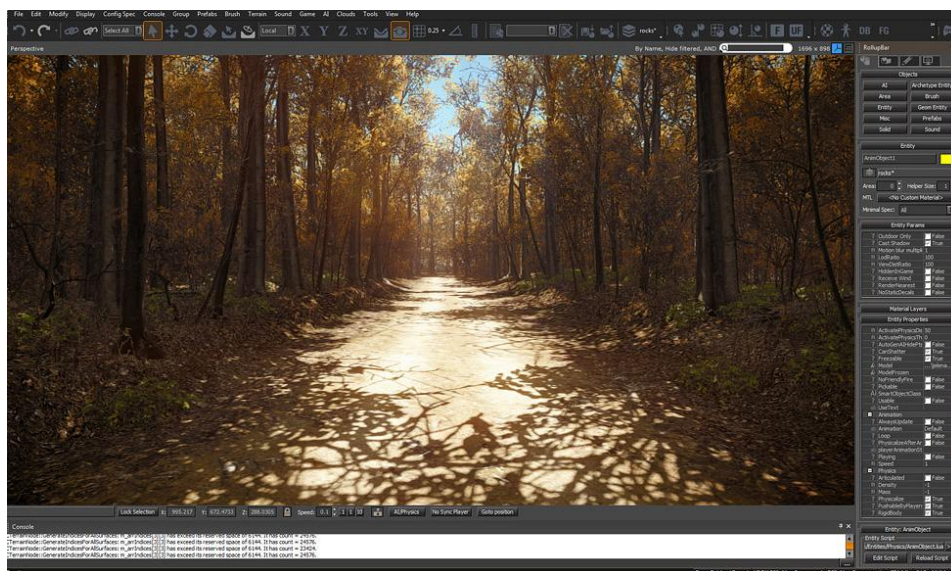
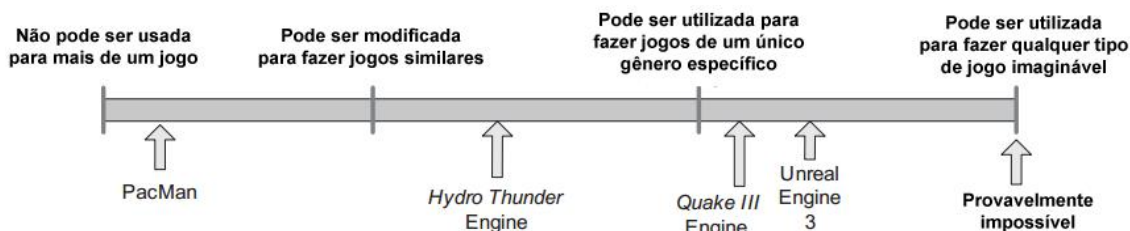


Figura 3 - Tela com vários recursos da CryEngine®3. Fonte: CGEverything, 2011

Devido a essa relação tão íntima com todos os recursos de um jogo, pode-se perguntar se a *game engine* não seria, no fim das contas, o próprio *game*. Gregory (2009) argumenta que uma *arquitetura orientada a dados* é o que diferencia ambos os conceitos. Se um jogo utiliza regras e lógica para tipos específicos de objetos, será muito difícil reaproveitar seu software para fazer outro jogo (Figura 4). Portanto, *game engine* seria um software volátil o suficiente para ser usado como alicerce para vários *games* diferentes sem a necessidade de muitas modificações (GREGORY, 2009).



⁶ Responsável pelo planejamento e construção dos níveis com os quais o personagem, dentro do jogo, irá interagir.

Figura 4 - Escala de Reusabilidade de uma *Game Engine*. Fonte: Gregory, 2009.

3.2 COMPONENTES

Uma *game engine* não é um único software capaz de realizar todas as tarefas necessárias para a criação de um jogo, mas sim um conjunto de subsistemas que interagem entre si para atingir esse mesmo fim, ilustrado pela Figura 5 (NILSON, B.; SÖDERBERG, M., 2007). Segundo os autores, os subsistemas mais importantes são:

- **Audio:** a sonorização é muito importante em uma *game engine*, visto que a mídia dos videogames se utiliza de sons tanto quanto o cinema. O subsistema de áudio abstrai a placa de som contida no hardware do PC ou do console, para que o resto da *engine* possa reproduzir sons sem restrições;
- **Input:** é o sistema responsável pela interação do usuário com a *game engine* e com o *game*, através dos mais diversos dispositivos de entrada como mouse, teclado e controle;
- **Physics Engine:** tendo ao menos alguma inspiração no mundo real, jogos eletrônicos necessitam de física para que a movimentação de personagens e objetos seja realista e consistente com o mundo ao seu redor. Devido à grande complexidade dessa área da ciência, muitas desenvolvedoras utilizam *middlewares*⁷ para realizar os cálculos. Um exemplo famoso é a *Havok*, aplicada no jogo *Half-Life 2*, da *Valve Corporation*;
- **Renderer:** a parte visual de uma *game engine* manuseia tudo o que é exibido na tela para o usuário – desde simples textos até gráficos complexos de cálculo pesado. Assim como o subsistema de áudio, o *Renderer* deve abstrair a placa de vídeo do *hardware* para possibilitar à *engine* moldar a tela à sua maneira;
- **Core:** é o núcleo, o subsistema da *game engine* que organiza e coordena todos os outros subsistemas, tratando também do gerenciamento da memória utilizada. Outra função do *Core* é a de iniciar o jogo e os dispositivos associados;
- **Scripting:** a *game engine* deve conferir aos desenvolvedores uma maneira mais rápida e eficiente de se produzir conteúdo para seus respectivos jogos. Com uma *scripting engine* poderosa, os produtores podem criar modelos, agentes, sonorização, cenários e jogabilidade sem precisar conhecer o código fonte da

⁷ Neologismo criado para designar camadas de software que não constituem diretamente aplicações, mas que facilitam a comunicação entre sistemas.

game engine, tendo assim mais liberdade e facilidade no desenvolvimento de um jogo complexo;

- **Networking:** hoje em dia, na era da internet, a maioria dos jogos eletrônicos tem algum tipo de *multiplayer*⁸ online, necessitando então de uma boa conexão com a rede, gerada através de códigos na *game engine*; e
- **Artificial Intelligence:** como já foi explicado anteriormente, a Inteligência Artificial é extremamente importante nos *games* de hoje, visto que aumentam consideravelmente a imersão do jogador, bem como bons gráficos e física. A maior diferença aqui é que a IA normalmente difere de jogo para jogo, e por isso é sempre mais interessante construí-la na própria *game engine* ao invés de fazer a utilização de *middlewares*.

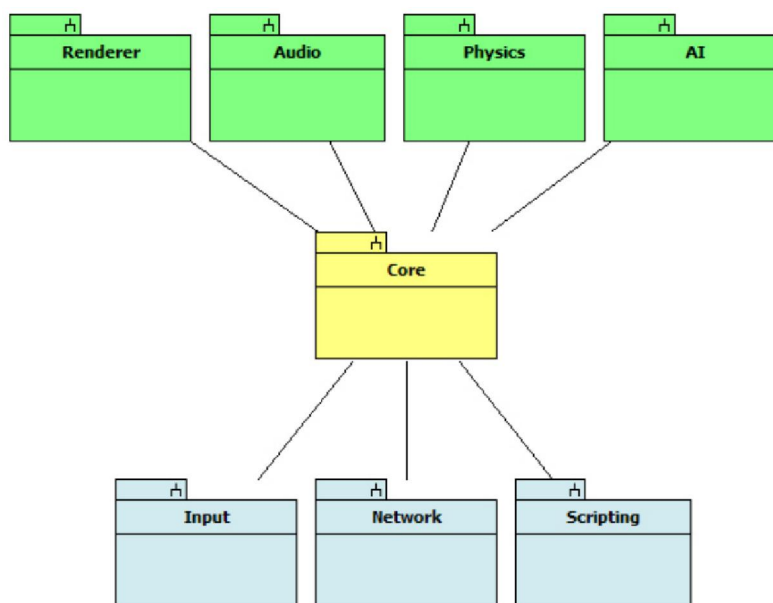


Figura 5 - Os componentes de uma *game engine*. Fonte: NILSON, B.; SÖDERBERG, M., 2007.

4 UNREAL ENGINE

Lançado em 1998 pela *Epic Games*, o FPS *Unreal* (Figura 6) trouxe à indústria de videogames a *Unreal Engine*, principal competidora da *Quake II Engine* para jogos de tiro no espaço (GREGORY, 2009). Dentre seus recursos, os mais notáveis para a época eram a física para detecção de colisão, iluminação multicolorida, e uma versão simples de filtragem de texturas. Assim como sua concorrente, a *Unreal Engine* logo se

⁸ Modo de jogo com mais de um jogador.

tornou popular entre a comunidade de *modders*⁹, principalmente devido à sua linguagem de scripts, a *UnrealScript*, e ao editor de mapas *UnrealEd* (LILLY, 2013).



Figura 6 - Tela do jogo *Unreal*, lançado em 1998 com a *Unreal Engine*. Fonte: Mod Database.

A *Unreal Engine 2* surgiu em 2002 com o jogo *America's Army* (LILLY, 2013), desenvolvido pelo exército dos Estados Unidos e liberado para o público com o objetivo de deixar o serviço militar mais interessante para os jovens da geração do computador (KENNEDY, 2013). Essa nova *game engine* da *Epic Games* era uma versão bastante modificada da primeira, trazendo física integrada e novos efeitos especiais, como água mais realista (LILLY, 2013). É interessante notar que a *Unreal Engine 2* foi a base para o jogo *Unreal Tournament 2004* (UT2004), o qual é até hoje muito utilizado para projetos de universidades (GREGORY, 2009), e é parte do estudo de caso contido neste artigo.

Cinco anos depois, em 2007, chegou ao mercado a *Unreal Engine 3*, projetada para PCs com DirectX10 e para os consoles *PlayStation 3* e *Xbox360* (LILLY, 2013). Essa versão se destacou por trazer ferramentas e recursos ricos para o desenvolvimento de *games*, como, por exemplo, uma interface visual poderosa para a programação, o *Unreal Kismet* (GREGORY, 2009).

⁹ Usuários que modificam um jogo pré-existente dando a este novas características e possibilidades.

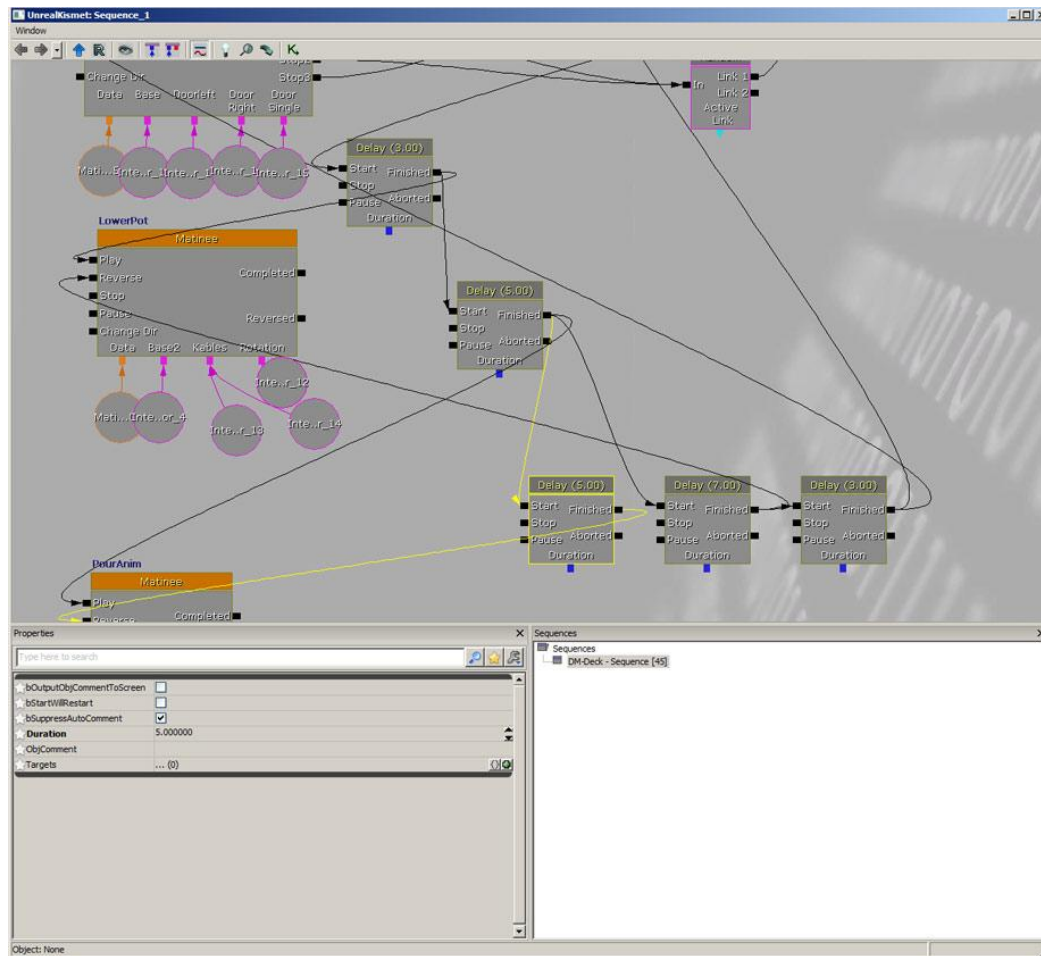


Figura 7 - *Unreal Kismet*, interface visual para scripting.
Fonte: Unreal Engine, 2013.

A *Unreal Engine 3* vem com algumas tecnologias *middleware*, o que, por consequência, traz diversas funcionalidades. Entre elas estão: o sistema de renderização multitarefa *Gemini*; cálculos de física realizados pela *PhysX*, da *Nvidia*; efeitos de partículas feitos pela *Cascade*; a ferramenta para criação de *cutscenes*¹⁰, a *Matinee*; interface para *scripting* (Figura 7), dentre outros (LILLY, 2013).

Na área da Inteligência Artificial, a *Unreal AI* proporciona a personagens controlados por IA uma melhor consciência do mundo à sua volta, dando-lhes mais capacidade para fazer decisões e movimentos inteligentes.

Vários *games* foram desenvolvidos utilizando a *Unreal Engine 3*, incluindo o *Unreal Tournament 3* e o famoso jogo da própria *Epic Games*, *Gears of War*. Há

¹⁰ Um evento ou sequência em um videogame que interrompe a jogabilidade, tirando o controle do jogador sobre os personagens, para avançar o enredo.

também jogos fora do escopo dos *shooters*¹¹, como o aclamado *Batman: Arkham Asylum* e o jogo de *parkour* em primeira pessoa, *Mirror's Edge* (LILLY, 2013).



Figura 8 - Evolução gráfica nas versões da *Unreal Engine*. Fonte: UOL Jogos, 2011

A *Unreal Engine*, como um todo, ficou conhecida por sua imensa gama de recursos e por sua fácil usabilidade. Apesar de não ser perfeita e necessitar de modificações por parte das desenvolvedoras para otimizar seus jogos, a *game engine* da *Epic Games* continua sendo uma das plataformas de desenvolvimento de jogos eletrônicos mais poderosas do mercado, podendo ser utilizada para construir praticamente qualquer tipo de jogo 3D em primeira ou terceira pessoa (GREGORY, 2009). A Figura 8 exemplifica a evolução gráfica entre as versões da *Unreal Engine*.

5 ESTUDO DE CASO: BOTPRIZE

A jornada em busca da máquina pensante chegou ao mundo dos jogos eletrônicos em Dezembro de 2008, na cidade de Perth, Austrália, como parte da *IEEE Symposium on Computational Intelligence and Games*¹², onde aconteceu a primeira edição da competição "2K BotPrize" (HINGSTON, 2013a). O concurso é patrocinado por uma desenvolvedora de jogos eletrônicos Australiana, a 2K Games, e foi idealizado

¹¹ Jogos de Tiro.

¹² Conferências de Inteligência Computacional e Games que reúnem os principais pesquisadores e profissionais de universidades e empresas para discutir os recentes avanços e explorar direções futuras no campo da Inteligência Artificial.

pelo professor Philip Hingston¹³, da Edith Cowan University, em Perth, Austrália. A tarefa da competição é criar um *bot*, uma Inteligência Artificial que controla um personagem de videogame, que seja indistinguível de um jogador humano. Na página inicial do site da competição, pode-se ler:

Computadores são incrivelmente rápidos e precisos nos games, mas eles podem ser programados para serem mais divertidos de se jogar – jogar como uma pessoa? As pessoas gostam de jogar contra adversários que são parecidos com elas mesmas – adversários com personalidade, que podem surpreender, que às vezes cometem erros, mas que não cometem os mesmos erros repetidamente. (HINGSTON, 2013a)

Criar adversários de nível humano para videogames é um "desafio mortal" para a Inteligência artificial, segundo Hingston (2013e). Nesse embate, vários autores apresentam razões para os pesquisadores de Inteligência Artificial trabalharem nesse objetivo:

- Videogames são aplicações interativas que precisam de nível humano de Inteligência Artificial;
- Eles oferecem ambientes propícios para a pesquisa sobre o "tipo certo" de problema que pode levar a Inteligência Artificial a um nível humano;
- IAs com nível humano são necessárias para um treinamento realista usando simulações (e tais simulações interativas são, em essência, jogos de computador);
- Videogames modernos oferecem realismo suficiente sem os problemas práticos do uso de sensores e sistemas reais;
- Jogos de computador são baratos, confiáveis e a maioria tem interfaces de IA; e
- A Inteligência Artificial de videogames é limitada – pesquisadores de IA são necessários para resolver essa questão.

Com essas ideias em mente, vários pesquisadores se empenharam em desenvolver *bots* capazes de “jogar” como uma pessoa, enviando-os para a competição, onde são julgados por jogadores que, dentro do jogo, os classificam como humanos ou máquinas.

¹³ Pesquisador em Inteligência Artificial e Inteligência Computacional, atualmente Professor Associado de Ciências da Computação da Universidade de Edith Cowan, em Perth, Austrália Ocidental. Doutor em Filosofia, Universidade de Monash, 1979.

Este estudo não irá descrever os pormenores das questões técnicas que envolvem a IA utilizada para a construção dos *bots*, nem seus algoritmos; isso poderá ser feito em uma outra oportunidade. Este estudo tem, por fim, mostrar a versatilidade da *Unreal Engine* na criação e desenvolvimento de uma IA mais humana capaz de proporcionar um melhor desafio na experiência de se jogar videogame.

O jogo no qual se desenrola toda a batalha entre jogadores humanos e robôs é uma modificação do clássico FPS da produtora *Epic Games*, o *Unreal Tournament 2004*. Como mencionado anteriormente, o UT2004 foi desenvolvido na *Unreal Engine 2.5* e vem com sua própria linguagem de script, a *UnrealScript*, que facilita o *modding* e consequentemente a criação de um *bot* e seu lançamento dentro do jogo. Um dos *mods* utilizados no torneio é o *GameBots2004*, que permite que os *bots* sejam controlados de forma separada do jogo em si, enviando e recebendo comandos e ações via *socket*. Existem várias versões do *GameBots*; algumas funcionam em títulos anteriores do *Unreal Tournament* e existem pelo menos duas versões compatíveis com o UT2004.

O *GameBots2004* foi desenvolvido pelo projeto *Pogamut 3* (GEMROT et al., 2013), que é um sistema intermediário, uma camada de aplicação que permite criar e controlar agentes virtuais em várias *game engines*. As plataformas atuais que suportam esse sistema intermediário são:

- *Unreal Tournament 2004* (UT2004);
- *UnrealEngine2RuntimeDemo* (UE2);
- *Unreal Development Kit* (UDK); e
- *DEFCON*.

Esse *middleware* fornece uma interface em *Java*, que permite instanciar e controlar os *bots* e também uma GUI (Interface Gráfica de Usuário), através de um *plugin* para o NetBeans.

A "2K BotPrize" possui um sistema de julgamento baseado em uma modificação de uma das armas usadas do jogo. O modo principal dessa arma marca os *bots*, enquanto o modo secundário marca os personagens comandados por humanos. Durante o jogo, quando um jogador acredita que descobriu a natureza de um adversário encontrado, ele atira contra este, utilizando o modo da arma que condiz com suas suspeitas. Esse tiro não causa dano ao oponente, e ao invés disso o jogador que disparou passará a ver uma marcação de "BOT" ou de "HUMANO" nesse adversário, para

lembrá-lo de que esse oponente foi julgado. Se, durante a partida, o jogador mudar de opinião a respeito de um adversário, ele pode acertá-lo novamente, usando um modo diferente da arma, sem restrições (HINGSTON, 2013a).

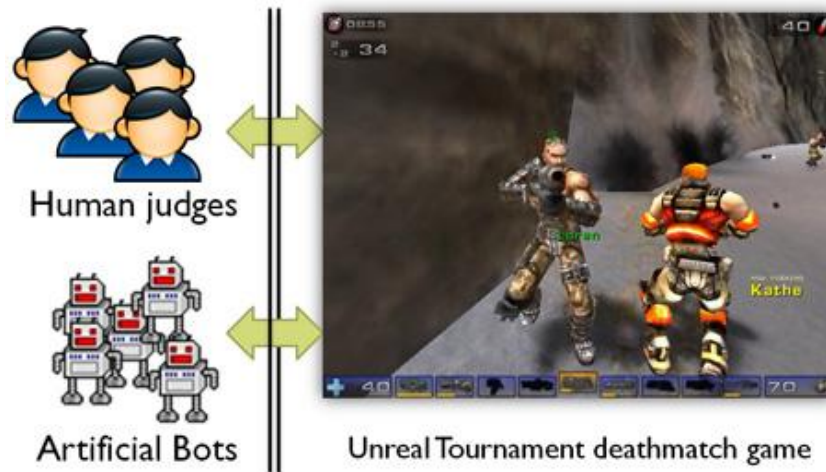


Figura 9 - Juízes avaliam os jogadores e bots. Fonte: (YAMAMOTO, 2013)

É importante notar também que os próprios *bots* também utilizam a arma de julgamento, e esta funciona da mesma forma que a arma dos jogadores humanos (HINGSTON, 2013a).

O vencedor é determinado pelo "índice de humanidade" do *bot*. Esse índice é o percentual de vezes em que o *bot* foi julgado como ser humano pelos juízes durante a série final. Para ganhar o prêmio principal, o *bot* deve atingir um índice de humanidade igual ou maior do que a média do índice de humanidade dos jogadores humanos (HINGSTON, 2013a).

As equipes que chegaram às finais na primeira competição, realizada em 2008 estão listadas na Tabela 1.

The Teams

Team	Affiliation	Members
Amis	Charles University in Prague	Michal Stolba Jakub Gemrot Juraj Simlovic
ICE-UT@RITS	Ritsumeikan University, Japan	Daichi Hirono Yuna Dei Ruck Thawonmas
Intelligent Systems Centre	Nanyang Technological University	Budhitama Subagdja Ah Hwee Tan Di Wang
Underdog	University of Western Australia	Oren Nachman
University of Texas at Austin	University of Texas at Austin	Jacob Schrum Igor Karpov Risto Miikulainen

Tabela 1 - As equipes finalistas de 2008. (HINGSTON, 2013b)

Nessa edição, nenhum dos *bots* enviados conseguiu ser classificado como tendo características mais humanas que as dos jogadores reais, e o vencedor foi o *bot* AMIS (Michal Stolba, Jakub Gemrot, Juraj Simlovic). Como pode ser visto na Tabela 2, os *bots* não obtiveram um índice maior do que 2.4, enquanto o menor índice humano foi 2.6.

Identity	Mean rating	#Judges convinced
Byron (human)	4	5
Andrew (human)	3.8	4
Roderick (human)	3.8	4
Keith (human)	3	2
Seb (human)	2.6	2
AMIS (bot)	2.4	2
ICE (bot)	2.2	1
ISC (bot)	2	2
UTexas (bot)	0.8	0
Underdog (bot)	0.4	0

Tabela 2 - Juízes convencidos pelos *bots*. Fonte: (HINGSTON, 2013b)

O segundo concurso BotPrize foi realizado no Politécnico de Milão, Itália, como parte da *IEEE Symposium on Computational Intelligence and Games 2009*, e contou com a participação das seguintes equipes nas finais (Tabela 3):

Team	Affiliation	Members
ANU	Australian National University, Australia	Chris Pelling
UTAustinite	University of Texas, Austin, USA	Igor Karpov Jacob Schrum Risto Miikulainen
ICE-2009	Ritsumeikan University, Japan	Daichi Hirono Akihiro Kojima Ruck Thawonmas
BradBot	individual	Thomas Curley
sqlitebot	individual	Jeremy Cothran

Tabela 3 - As equipes finalistas de 2009. Fonte: (HINGSTON, 2013c)

Nesse ano, novamente, nenhum dos *bots* foi capaz de enganar todos os juízes e levar o grande prêmio. No entanto, todos os *bots* conseguiram enganar pelo menos um dos avaliadores e o vencedor foi o *sqlitebot*, desenvolvido por Jeremy Cothran.

Em 2010, o desafio foi realizado na Universidade de TI de Copenhagen, Dinamarca, no dia 20 de agosto daquele ano. As equipes finalistas podem ser vistas na Tabela 4.

Team	Affiliation	Members
ANU	Australian National University, Australia	Chris Pelling
UTAustinite	University of Texas, Austin, USA	Igor Karpov Jacob Schrum Risto Miikulainen
ICE-2009	Ritsumeikan University, Japan	Daichi Hirono Akihiro Kojima Ruck Thawonmas
BradBot	individual	Thomas Curley
sqlitebot	individual	Jeremy Cothran

Tabela 4 - As equipes finalistas de 2010. Fonte: (HINGSTON, 2013d)

Embora, mais uma vez, nenhum *bot* tenha sido capaz de levar o prêmio principal, o *bot* vencedor foi Conscious-Robots, pela equipe da Universidade Carlos III de Madrid, Espanha.

Em 2011, a disputa pelo *bot* mais humano foi realizada na Coreia do Sul, na cidade de Seoul, no dia 03 de setembro. A competição não teve vencedor do prêmio máximo, e o *bot* de maior sucesso foi o ICE-CIG2011 da equipe da Universidade de Ritsumeikan, Japão. Notavelmente, dois robôs ficaram bem próximos de chegarem ao topo da classificação de humanidade. A busca pelo *bot* com traços de humanidade estava se aproximando do fim (Tabela 5).

Humanness results					
Most human bots		Most human humans		Most human epic bots	
bot name	humanness %	player name	humanness %	skill level	humanness %
ICE-CIG2011	37.5000 %	KyeongJong Lee	66.6667 %	1	50.0000 %
NeuroBot	35.7143 %	Daniel Beard	60.0000 %	2	50.0000 %
Conscious-Robots	26.6667 %	HyunSoo Park	50.0000 %	5	41.6667 %
UT^2	21.4286 %	Mike Preuss	50.0000 %	4	33.3333 %
		Geoffrey Hingston	20.0000 %	3	33.3333 %
				all	40.0000 %

Judging results			
Best bot judges		Best human judges	
bot name	accuracy %	human name	accuracy %
ICE-CIG2011	50.0000 %	Mike Preuss	66.6667 %
Conscious-Robots	50.0000 %	HyunSoo Park	66.6667 %
NeuroBot	42.8571 %	KyeongJong Lee	58.8235 %
UT^2	41.6667 %	Daniel Beard	52.6316 %
		Geoffrey Hingston	47.6190 %

Tabela 5 - Resultados da competição de 2011. Fonte: (HINGSTON, 2013f)

Em 2012, ano do centenário de Alan Turing, a competição retornou à Austrália, na cidade de Brisbane, juntamente com a *IEEE World Congress on Computational Intelligence*. O congresso teve início no dia 10 de junho e, mais uma vez, as equipes enviaram seus *bots* para o desafio do *Unreal Tournament 2004*.

Em um resultado surpreendente, depois de cinco anos de disputas entre 14 equipes de nove países, duas destas conseguiram romper o "índice de humanidade" – calculado dividindo-se o número de vezes que um competidor (*bot* ou humano) foi julgado como sendo humano por outros humanos, pelo número total de vezes que foi julgado – e levar o prêmio máximo da competição. Cada jogador foi julgado cerca de 25 vezes, eliminando assim a possibilidade de que o alto índice de humanidade alcançado pelos *bots* tenha sido obra do acaso.

Os grandes vencedores, mostrados na Tabela 6, foram o *bot UT^2*, da equipe da Universidade do Texas, em Austin, e o *bot MirrorBot*, feito por Mihai Polceanu, um estudante de doutorado da Romênia, atualmente cursando Inteligência Artificial em Brest, França. A equipe *UT^2* foi composta pelo Professor Risto Miikkulainen e os doutorandos Jacob Schrum e Igor Karpov. As duas equipes dividiram um prêmio de sete mil dólares, oferecido pela 2K Games (HINGSTON, 2013a).

Most human bots		Most human humans		Most human epic bots
bot name	humanness %	player name	humanness %	
MirrorBot	52.2 %	Samaneh Rastegari	53.3 %	average 37.8 %
UT^2	51.9 %	Craig Speelman	52.2 %	
ICE-CIG2012	36.0 %	John Weise	30.8 %	
NeuroBot	26.1 %	Chris Holme	26.3 %	
GladiatorBot	21.7 %	average	41.4 %	
AmisBot	16.0 %			
average	34.2 %			

Tabela 6 - Resultados da competição de 2012. Fonte: (HINGSTON, 2013g)

Após o feito notável das equipes no ano de 2012, a competição teve uma pausa em 2013, prometendo voltar em 2014 com maiores desafios para que assim os desenvolvedores levem suas criações ao próximo nível de desempenho humano.

6 CONCLUSÃO

Sendo a única mídia dependente da interatividade, os jogos eletrônicos têm um potencial gigantesco quando o assunto é a imersão do usuário. Com visuais cada vez mais estupendos, além de física e efeitos extremamente realistas, os *games* têm tido um foco nunca antes visto nas Inteligências Artificiais de seus personagens e inimigos.

No entanto, devido à grande complexidade de diversos elementos diferentes durante o desenvolvimento de um jogo nos dias de hoje, sem uma base já construída seria praticamente impossível não só aplicar uma IA avançada ao produto, mas também terminá-lo a tempo de competir no mercado. Portanto, pode-se perceber o quanto uma *game engine*, responsável pelas ferramentas de criação e pela estruturação e organização de um jogo eletrônico, é importante para as desenvolvedoras.

Com o auxílio das *game engines*, não há mais a preocupação de se iniciar a criação de um *game* do zero, e ao invés disso pode-se focar na agregação de novas funcionalidades ou na expansão de aspectos já existentes na própria *engine*.

Além disso, as *game engines* possibilitam também a ação dos *modders*, incentivando assim o desenvolvimento independente de particularidades ainda mais avançadas, como é o caso dos competidores do "2K BotPrize".

Foi graças à volatilidade da *Unreal Engine 2.5* e sua ferramenta de *scripting*, a *UnrealScript*, que os *bots* MirrorBot e UT^2 conseguiram enganar tanto jogadores, quanto juízes, se passando por seres humanos, em uma ação incrível que poucas Inteligências Artificiais são capazes de realizar.

Mesmo o "2K BotPrize" não sendo exatamente o mesmo teste proposto por Alan Turing em 2013, ainda é de se surpreender que máquinas tenham sido confundidas com seres humanos. Tal feito mostra todo o potencial que a Inteligência Artificial tem não só no mundo dos jogos eletrônicos, mas também na ciência como um todo.

REFERÊNCIAS BIBLIOGRÁFICAS

BRANCO, Marsal; Malfatti, Silvano; LAMAR, Marcus Vinicius. **Jogos Eletrônicos na Prática: Livro de Tutoriais**. 2. ed. Universidade Feevale: Feevale, 2012.

BUCHANAN, Bruce G. A (Very) Brief History of Artificial Intelligence. **AI Magazine**. v. 26, n. 4, p. 53-60, inverno 2005. Disponível em: <<http://www.aaai.org/ojs/index.php/aimagazine/article/view/1848/1746>>. Acesso em: 8 nov. 2013.

CHARLES, Darryl; FYFE, Colyn; LIVINGSTONE, Daniel; MCGLINCHEY, Stephen. **Biologically Inspired Artificial Intelligence for Computer Games**. Hershey: IGI, 2008.

COPELAND, Jack. **A Brief History of Computing**. The Turing Archive, 2000. Disponível em: <http://www.alanturing.net/turing_archive/pages/Reference%20Articles/BriefHistofComp.html>. Acesso em: 6 nov. 2013.

COPELAND, Jack. **What is Artificial Intelligence?** The Turing Archive, 2000. Disponível em: <http://www.alanturing.net/turing_archive/pages/Reference%20Articles/What%20is%20AI.html>. Acesso em: 7 nov. 2013.

GEMROT, J.; KADLEC, R.; BÍDA, M.; BURKERT, O.; PÍBIL, R.; HAVLÍČEK, J.; ZEMČÁK, L.; ŠIMLOVIČ, J.; VANSÁ, R.; ŠTOLBA, M.; PLCH, T.; BROM, C. Pogamut 3 Can Assist Developers in Building AI (Not Only) for Their Videogame Agents. In: **Agents for Games and Simulations**, LNCS 5920, Springer, 2009, pp. 1-15. Disponível em: <http://pogamut.cuni.cz/main/papers/pogamut3_AGS_final.pdf>. Acesso em: 8 nov., 2013.

GREGORY, Jason. **Game Engine Architecture**. Nova York: Taylor & Francis Group, 2009.

HARNARD, Stevan. Minds, Machines and Turing: The Indistinguishability of the Indistinguishables. In: **Journal of Logic, Language and Information**. 2001.

Disponível em: <<http://cogprints.org/2615/1/harnad00.turing.html>>. Acesso em: 5 nov. 2013.

HINGSTON, Philip. **The 2K BotPrize**: Can computers play like people?. 2008. Disponível em: <<http://botprize.org/index.html>>. Acesso em: 28 out. 2013.

HINGSTON, Philip. **2008 BotPrize Contest**. 2008. Disponível em: <<http://www.botprize.org/2008.html>>. Acesso em: 7 nov. 2013.

HINGSTON, Philip. **2009 BotPrize Contest**. 2009. Disponível em: <<http://www.botprize.org/2009.html>>. Acesso em: 7 nov. 2013.

HINGSTON, Philip. **2010 BotPrize Contest**. 2010. Disponível em: <<http://www.botprize.org/2010.html>>. Acesso em: 7 nov. 2013.

HINGSTON, Philip. **A New Design for a Turing Test for Bots**. 2010. Disponível em: <<http://ro.ecu.edu.au/cgi/viewcontent.cgi?article=7339&context=ecuworks>>. Acesso em: 3 nov. 2013.

HINGSTON, Philip. **2011 BotPrize Contest**. 2011. Disponível em: <<http://www.botprize.org/2011.html>>. Acesso em: 7 nov. 2013.

HINGSTON, Philip. **The 2K BotPrize: Result**. 2012. Disponível em: <<http://www.botprize.org/result.html>>. Acesso em: 7 nov. 2013.

KENNEDY, Brian. Uncle Sam Wants You (To Play This Game). **The New York Times**, Nova York, 11 jul. 2002. Disponível em: <<http://www.nytimes.com/2002/07/11/technology/uncle-sam-wants-you-to-play-this-game.html>>. Acesso em: 5 nov. 2013.

LILLY, Paul. **Doom to Dunia**: A Visual History of Game Engines. Maximum PC, 17 jul. 2009. Disponível em: <http://www.maximumpc.com/article/features/doom_dunia_visual_history_3d_game_engines>. Acesso em: 4 nov. 2013.

NEWZOO. **Infographic 2011** – US. Disponível em: <<http://www.newzoo.com/infographics/infographic-2011-us/>>. Acesso em: 7 nov. 2013.

NILSON, Björn; SÖDERBERG, Martin. **Game Engine Architecture**. Mälardalen University Sweden, 2007. Disponível em: <<http://www.idt.mdh.se/kurser/cd5130/jgms/2007lp4/report9.pdf>>. Acesso em: 6 nov. 2013.

NORVIG, Peter; RUSSEL, Stuart J. **Artificial Intelligence: A Modern Approach**. New Jersey: Prentice-Hall, 1995.

POTTINGER, Dave. Game AI: The State of the Industry, Part Two. **Gamasutra**, 8 nov. 2000. Disponível em: <http://www.gamasutra.com/view/feature/131974/game_ai_the_state_of_the_.php>. Acesso em: 8 nov. 2013.

SAMUEL, Arthur L. Some Studies in Machine Learning Using the Game of Checkers. **IBM Journal of Research and Development**, v. 3, n. 3, p. 210-229, jul. 1959.

SAMYN, Michael. Video Games as Media. **Gamasutra**, 16 fev. 2011. Disponível em: <http://www.gamasutra.com/view/feature/134657/video_games_as_media.php>. Acesso em: 3 nov. 2013.

SCHWAB, Brian. **AI Game Engine Programming**. 2. ed. Boston: Course Technology, 2009.

TURING, Alan M. Computing Machinery and Intelligence. **Mind**, Oxford University, v. 59, n. 236, p. 433-460, out. 1950. Disponível em: <<http://www.jstor.org/stable/2251299>>. Acesso em: 4 nov. 2013.

UNREAL ENGINE, **Unreal Kismet, the visual scripting system**. Epic Games, Inc., 2008-2013. Disponível em: <<http://www.unrealengine.com/features/kismet/>>. Acesso em: 7 nov. 2013.

WILMORE, Julie. **Dissecting the Video Game Engine and a Brief History**. Michigan State University, 2009. Disponível em: <<http://www.lenostore.com/images/img/01824066.pdf>>. Acesso em: 6 nov. 2013.

YAMAMOTO, Flavio. **Human-Like Non-Player Characters in UT2004 Arena**. Unicamp, 2013. Disponível em: <<http://faculty.dca.fee.unicamp.br/gudwin/node/129>>. Acesso em: 5 nov. 2013.