

Intelig ncia Artificial para Jogos

Aula 01 - Introdu  o   IA em jogos



Apresentação da aula

Olá!

Esta é a primeira aula da disciplina Inteligência Artificial (IA) para Jogos! Nela você vai estudar conceitos e técnicas importantíssimos no desenvolvimento de jogos, que são aplicados desde que o Pac-man começou a sua dieta exclusiva de bolinhas, e perdura até os jogos mais recentes, quando aquelas hordas de zumbis famintos sempre acabam lhe encontrando, mesmo quando você corre pra se esconder!

E o que seria Inteligência Artificial? Seria uma falsa inteligência? Como assim?

Esse conceito vai ser bem explorado na aula de hoje, mas eu posso dar um *spoiler* para você: ~~no final de Game of Thrones, Jon Snow~~ sempre que se simula um comportamento em objetos ou programas de computador para fazer com que eles pareçam inteligentes, então se diz que está trabalhando uma técnica de Inteligência Artificial. O objeto parece ter inteligência própria, mas na verdade ela foi criada artificialmente por você e pelo conjunto de algoritmos e processos utilizados para criar uma **ilusão** de inteligência para o usuário do programa ou jogo. Captou minha mensagem? Não? Não se preocupe, irei falar bem disso ao longo dessa aula :).

Prepara o lanche para começar os seus estudos!



Objetivos

Compreender a diferença entre a IA tradicional e a IA aplicada em jogos;

Conhecer os tipos de IA aplicadas em jogos e os principais problemas solucionados por essas técnicas;

Identificar problemas de movimentação de personagens em cenários, tomada de decisão inteligente e os relacionados à criação de modelos de comportamento.

O que é IA?

Você já viu o filme de Steven Spielberg chamado A.I. – *Inteligência Artificial*, lançado no ano de 2001? No filme, um “robô menino” foi programado para se comportar como uma criança e “amar” sua mãe adotiva, humana, por toda a eternidade. Para se sentir correspondido, ele sonha, como Pinóquio, em ser humano e, nessa busca, passa por uma série de aventuras. É uma história cheia de metáforas filosóficas sobre o que é ser humano, sobre o que é um ser inanimado e sobre o que nos diferencia, segundo o autor do filme, de máquinas.

Figura 01 - Imagem do filme A.I. – Inteligência Artificial

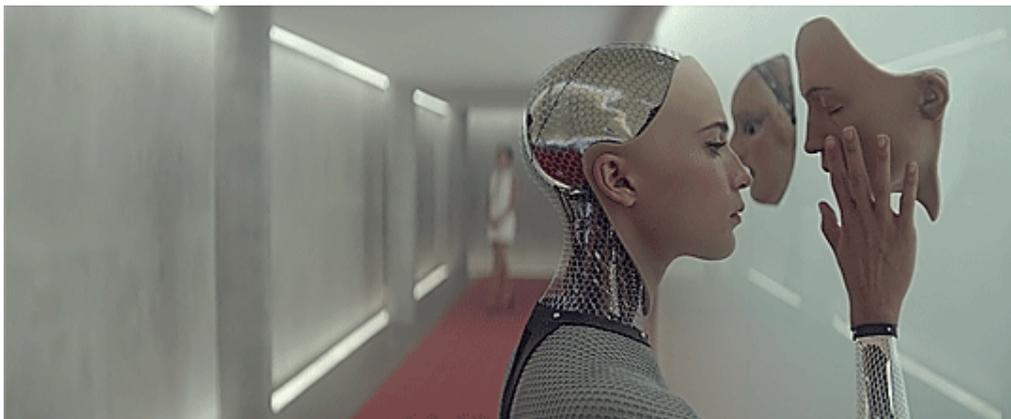


Fonte: Adaptado de GIF GALLERY. Disponível em: <<http://gifsgallery.com/>>. Acesso em: 14 de mar. de 2018.

Além desse filme, Hollywood tem nos proporcionado uma grande quantidade de filmes de ficção com diferentes visões sobre a IA, como os clássicos *Metropolis*, de 1927; *2001: Uma Odisseia no Espaço*, de 1968; *Blade Runner. O Caçador de Androides*, de 1982; *O Exterminador do Futuro*, de 1984; *Matrix*, de 1999; *Eu, robô*, de 2004; *Transcendence: A Revolução*, de 2014; *Ex-Machina: Instinto Artificial*, de 2015; entre outros.

No filme *Ex-Machina: Instinto Artificial*, por exemplo, a robô AVA é capaz de convencer as pessoas as quais sabem que ela é um robô a acreditarem que ela tem consciência e que, portanto, é um ser vivo.

Figura 02 - Imagem do filme Ex-Machina: Instinto Artificial, 2015

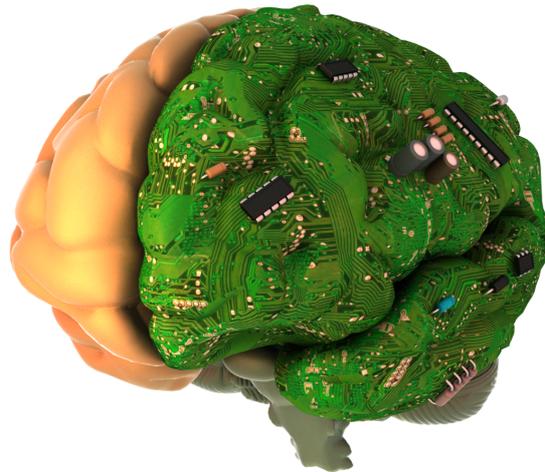


Fonte: GIFER. Disponível em: <<https://gifer.com/en/QY0F>>. Acesso em: 14 de mar. de 2018.

Dos filmes citados, até o mais clássico de todos, *Metropolis*, realizado quando ainda nem existiam computadores, retrata a **IA como uma tecnologia capaz de dar consciência e emoção a um objeto inanimado, que é a máquina**. Essa linha de pensamento, que é denominada por muitos autores como **IA forte**, procura criar um modelo de tomada geral de decisão, assim como nós humanos fazemos. Por exemplo, se uma máquina for submetida ao processo de escrever um texto, ela teria de ter consciência do que escreveu e não somente organizar as palavras para formar frases. Como você deve ter percebido, a **Inteligência Artificial Forte** inspirou e tem inspirado tanto os escritores quanto os roteiristas de ficção científica.

Mas, mesmo com tantos atrativos, essa linha é contestada por alguns autores e pesquisadores. E, eles se utilizam de dois argumentos: o primeiro é a distância em relação à nossa realidade atual. Segundo, talvez o mais importante, é que essa linha assume a possibilidade de a mente humana ser modelada em um computador. Em outras palavras, a mente humana nada mais é do que um software rodando em um hardware potente (cérebro).

Figura 03 - Cérebro humano como sendo o hardware de uma máquina pensante



Pesquisadores dessa linha de IA têm como foco entender a natureza do pensamento e do conceito de inteligência. Eles estão interessados em construir modelos e programas que possam nos trazer, entre outros benefícios, um entendimento sobre como pensamos, como estruturamos nosso conhecimento e como raciocinamos sobre esse conhecimento. Assim, eles tentam **reproduzir nos computadores a forma como pensamos** (ou, pelo menos, a forma como pensamos que pensamos). Bem, esse é um campo fértil para discussões filosóficas sobre mente, alma e inteligência, que já durou séculos e ainda pode durar mais alguns.

Apesar da ótima discussão filosófica que isso pode gerar, existe um lado mais prático. Os algoritmos e as técnicas de IA são usados para nos ajudar a realizar tarefas, deixando a vida com mais comodidade. Diferentemente do que os filmes de ficção científica propagam ou a filosofia discute, a IA não é, necessariamente, uma tecnologia de um futuro distante. Veja o vídeo 01 abaixo e faça suas conclusões.

Vídeo 01 - Entrevista com a IA Sophia para TV CNBC

Fonte: BEST OF WEB. Best Of Web. 2016. Disponível em: <https://www.youtube.com/watch?v=eqvmNhqdD38>. Acesso em: 13 mar. 2018.

E aí, você acredita que ela seja uma tecnologia de um futuro distante?

A IA está presente em muitos aspectos da nossa vida: das recomendações de amizade da sua rede social preferida aos carros autônomos - que serão em breve uma realidade nas nossas ruas, passando pelos corretores ortográficos e

gramaticais dos nossos editores de texto. Ela está presente no nosso dia a dia e muitas vezes nem nos damos conta. Esse é o tipo de IA que alguns autores denominam de **IA fraca** ou **IA aplicada**, em oposição ao conceito de **IA forte**. O fato de chamá-la de fraca não tem nada a ver com a força dela, muito pelo contrário! Chamamos de **IA fraca** as soluções de IA que não são generalistas a ponto de imitar o comportamento humano em geral, como a do vídeo acima.

Por exemplo, hoje em dia, para tirar uma foto, você pega o celular, aponta para onde você quer e o celular se encarrega de ajustar o foco, a intensidade do brilho que vai captar e, de quebra, ainda tira a foto quando alguém sorri para a tela.

Nesse caso, o aplicativo de tirar foto do celular possui algoritmos de reconhecimento de imagem que fazem o ajuste de foco, do brilho, detecta rostos e também quando há sorriso neles. Será que isso não pode ser considerado como “inteligência”? Claro que sim! Isso é um exemplo de IA presente no nosso dia a dia, facilitando as nossas atividades, tornando-as tão simples que até macacos podem fazê-las. Macacos? Isso mesmo, houve um recente caso de disputa de [copyright](#) (É um direito autoral, a propriedade literária, que concede ao autor de trabalhos originais direitos exclusivos de exploração de uma obra artística, literária ou científica, proibindo a reprodução.) de uma selfie feita por um macaco. O proprietário da máquina queria cobrar os direitos autorais da foto... mas, como? Se quem fez a foto foi o macaco?

Figura 04 - Selfie tirada por um macaco



Fonte: WIKIPEDIA. Disponível em: <https://en.wikipedia.org/wiki/Monkey_selfie>. Acesso em: 28 mar. 2018.



Saiba mais:

Foto interessante, não é? Bem, pelo menos, melhor do que as minhas. Essa foto ilustra como os algoritmos de IA presente nas câmeras atuais nos ajudam nas atividades do dia a dia a dia que até um macaco pode fazê-lo! Para saber mais, [acesse aqui](#).

A grande fonte de inspiração da inteligência artificial é a inteligência humana. Esta, por sua vez, tem muito a ver com o conceito de conhecimento. Você pode até se considerar inteligente porque tem conhecimento sobre conceitos e objetos ao seu redor e consegue fazer relações entre eles, não é mesmo? Para criar essas relações, bem como para expressar esse conhecimento, você precisa usar representações simbólicas.

Para melhor entendê-las, podemos classificá-las em dois tipos de abordagens: uma chamada de **simbólica** e outra de **subsimbólica**. É, eu sei, você deve estar se perguntando: Que termos malucos são esses? De onde eles vêm?

Por exemplo, o termo “cachorro” é uma representação simbólica de um conceito. Quando se pensa nessa palavra, pode-se associá-la com as seguintes características:

- É um animal de estimação dócil e fiel;
- É um mamífero peludo;
- Tem quatro patas;
- Gosta de comer a minha sandália e revirar o lixo da casa.

Todos esses conceitos vêm do entendimento que nós temos do que é um cachorro (ok, ok, alguns são muito específicos do MEU cachorro →). Então o símbolo “cachorro” traz uma carga de significados que nosso cérebro imediatamente consegue associar. Algumas técnicas de IA seguem essa estratégia, procurando definir um conhecimento artificial através de representações simbólicas, de suas relações e de manipulações que o computador pode fazer sobre elas. Por exemplo, você diria para o computador que existe um símbolo chamado “cachorro”, quais seriam as características de um cachorro e que tipo de ações um cachorro faz, e a partir daí o computador simularia o comportamento de um cachorro dentro dessas regras estabelecidas. É por isso que essa abordagem se chama **IA simbólica**. Faz sentido? Normalmente, essa abordagem usa lógicas proposicionais, sistemas especialistas, entre outras técnicas para implementar um sistema inteligente.

Porém, as técnicas da **IA simbólica** não são apropriadas para casos em que os dados são imprecisos ou quando não há informação suficiente para se tomar uma decisão “lógica”. Nesses casos, não há símbolos que ajudem! É necessário, então, usar uma abordagem abaixo das representações do conhecimento, ou seja, **subsimbólica** ou **conexional**.

Voltando ao exemplo do cachorro. Imagine que um amigo ~~alienígena~~ seu nunca viu um cachorro na vida! Mas ele já viu outros bichos, então você poderia tentar explicar para ele algumas coisas: um cachorro anda sobre quatro patas, come carne, gosta de viver em bandos... Se o seu amigo conhecer outros animais com

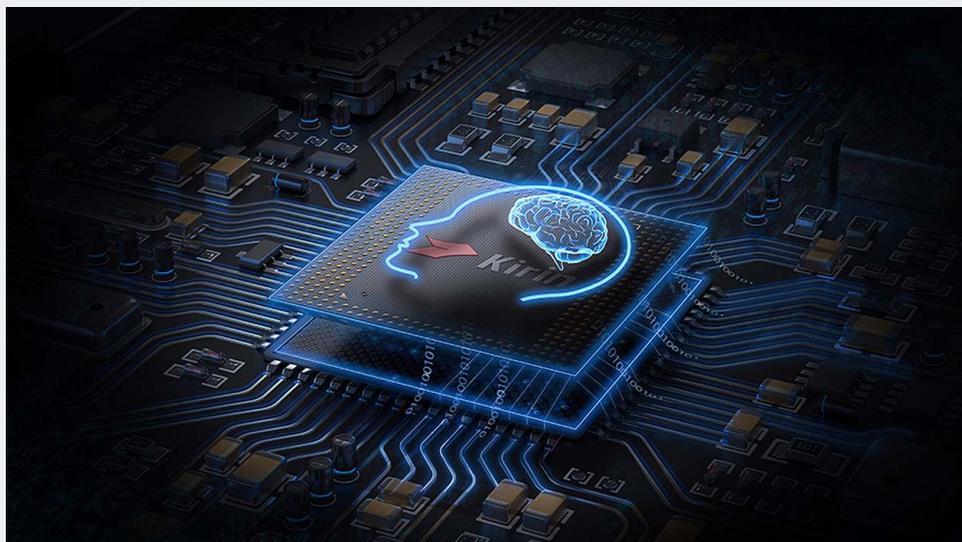
comportamentos parecidos, ele pode começar a construir uma base própria do que é um cachorro, mesmo sem nunca ter visto um. Não vai ser perfeito, mas ele pode até começar a tentar adivinhar comportamentos do cachorro por comparação com outros animais que ele conhece. Estratégias de IA que seguem essa linha, em que a base, em vez do símbolo definido, são as relações e conexões que são possíveis de estabelecer na base de conhecimento existente, e a partir dela construir novos conhecimentos, são denominadas **subsimbólicas** ou **conexionais**.

Por exemplo, um trabalho nessa linha é o de desenvolvimento de mecanismos que fazem os computadores aprenderem com os próprios erros por meio da modelagem de como o cérebro funciona. Assim, se cria uma rede de conexões que simula os neurônios e suas sinapses, chamadas **redes neurais**. Como o nosso cérebro, as sinapses estimulam a rede de neurônios, adaptando-se a novos sinais e, portanto, sendo capaz de aprender com isso.



Curiosidade

Para que serve o chip de inteligência artificial no seu smartphone?

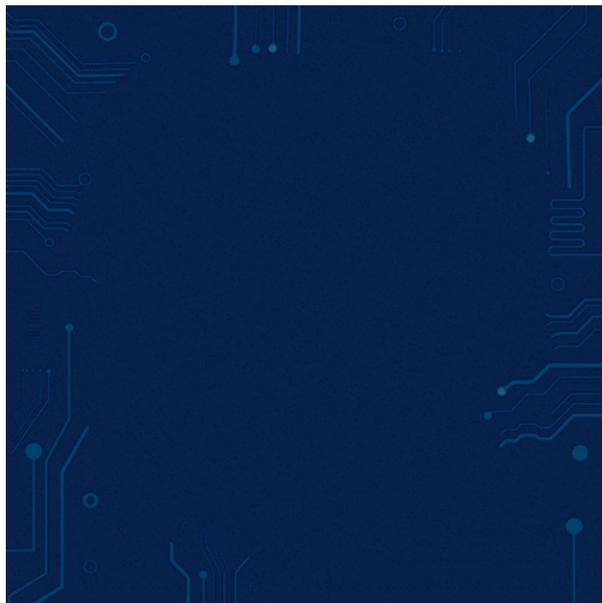


[Confira aqui!](#)

Os pesquisadores que usam essa técnica da **IA subsimbólica** exploram a ideia de que o conhecimento é fruto de uma rede de impulsos elétricos e que se pode fazer o mesmo através de um programa de computador. Assim, os programas teriam de aprender e a se adaptar, tal como o cérebro o faz.



Figura 05 - Rede de conexões entre neurônios artificiais



Enfim, todas essas técnicas e abordagens constituem a área de pesquisa de Inteligência Artificial. Diante de tantas linhas, estratégias e mecanismos da IA, há pelo menos o consenso em considerar que a **IA é a capacidade de um computador ou outra máquina de realizar atividades as quais acredita-se que requerem inteligência**. É essa área de pesquisa e desenvolvimento que se chama de **IA tradicional**, em oposição a que irei lhe apresentar, **Game AI** ou **IA para Jogos**.

O que é Game AI?

Por que mais uma classificação? A **IA tradicional** não é a mesma **IA aplicada em jogos?**

Sim, de fato é, porém, há um movimento recente entre os profissionais e pesquisadores da área de jogos em diferenciar a definição de IA que foi apresentada de algumas das técnicas consideradas na implementação de **IA nos jogos**. Por exemplo, em muitos jogos, os personagens controlados pelo computador não possuem um comportamento que se possa julgar como “inteligente”. Eles são muitas vezes previsíveis, com padrões e procedimentos bem definidos, sem a característica básica de aprendizagem e adaptação do ser humano. Nesse caso, a estratégia de IA implementada para reproduzir o comportamento do **NPC** ([Non-Player Character](#), ou, em português [Personagem não jogável](#) - PNJ)) pode ser extremamente simples, muitas vezes não sendo considerada como uma técnica de IA tradicional. Apesar da simplicidade, considera-se, ainda assim, o algoritmo que governa o comportamento do personagem como uma técnica de IA.

Veja, diferentemente da **IA tradicional**, o objetivo de uma **IA em um jogo** não é resolver problemas do dia a dia, nem entender e reproduzir o conceito de inteligência ou consciência em uma máquina. O objetivo é outro, é fazer com que os jogadores se divirtam! Simplesmente isso! Ou seja, a **IA utilizada em jogos** preocupa-se primordialmente em tornar um jogo divertido, independentemente da complexidade da técnica utilizada. Porém, isso não impede que as técnicas de IA tradicionais sejam usadas nos jogos, pelo contrário! O uso de técnicas de IA são extremamente bem-vindas em um jogo, desde que:

1. Não consumam muito recurso computacional, uma vez que jogos são, em geral, aplicações que precisam de um tempo de resposta extremamente rápido (“pseudo tempo-real”);
2. As soluções usadas pelos NPCs não sejam sempre ótimas. Caso contrário, não teria graça alguma para o jogador, que nunca irá ganhar. 😊

Por causa dessa diferença fundamental de **objetivos**, foi criado um termo para diferenciar os estudos da **IA tradicional** da **IA aplicada em jogos digitais**, o qual se chama de **Game AI** (do inglês *Game Artificial Intelligence*). Dessa forma, **Game AI** objetiva pesquisar e desenvolver algoritmos e técnicas que implementam, para cada jogo, **o grau de inteligência necessário para torná-lo divertido**. Nem mais, nem menos, apenas o suficiente.

Diferentes jogos podem requerer diferentes graus de inteligência... ou ilusão de inteligência. Há jogos que demandam o uso intensivo das técnicas da IA tradicional, através de representação de conhecimento, de mecanismos de inferências ou reconhecimento de padrões. Por exemplo, no jogo *Creatures*, de 1997, o jogador tinha o objetivo de criar seres alienígenas chamados Norns, desde bebês até a sua fase adulta. Como todo bebê, os Norns eram curiosos e queriam descobrir o seu [habitat](#) (É o ambiente natural em que vive um organismo, ou o ambiente físico que envolve uma população de espécies.), que era repleto de perigos, como objetos e plantas os quais podiam fazer-lhes mal; até criaturas inimigas, que os atacavam. O jogador deveria, então, educar os bebês Norns, fazendo-os aprender o que se podia fazer e o que evitar. Inclusive, ensinar nomes de objetos e de outras criaturas para que pudessem melhor dialogar com os Norns. E com essas técnicas o jogo fica bem mais interessante!

Esse é um exemplo no qual foram usadas técnicas de **IA tradicional**, no caso **redes neurais**, para implementar o aprendizado dos Norns.

Vídeo 02 - Abertura do jogo *Creatures*

Fonte: GLITCHFINDER. *Creatures one: introduction*. 2009. Disponível em: <https://www.youtube.com/watch?v=VLEZFgkyhUw>. Acesso em: 28 mar. 2018.

Outros jogos utilizam mecanismos similares, como é o caso do jogo *Kinectimals*, que a Microsoft desenvolveu em 2010 para explorar o potencial de interação do Kinect. Nesse jogo, voltado ao público infantil, o jogador pode ensinar alguns movimentos para seus pets virtuais.



Enfim, apesar da indústria de jogos ter procurado utilizar cada vez mais técnicas que permitam comportamentos mais realistas dos seus personagens, ela não tem deixado de lado técnicas consideradas simples, mas que podem prover horas e horas de diversão.

Tome como exemplo um dos jogos mais famosos e rentáveis da história: Pac-Man. Apesar da idade, criado em 1980, atualmente, ele ainda é jogado, com versões para PS3, Wii e Xbox 360. Nesse jogo, o jogador controla o personagem principal, o Pac-Man, e o computador controla quatro fantasmas que o perseguem dentro de um labirinto. Cada fantasma tem um mecanismo bastante simples de comportamento, definido por apenas dois estados: caçando o jogador ou fugindo dele. Apesar de simples, o comportamento associado à mecânica do jogo permite uma experiência divertida, e isso é muito mais importante do que um comportamento complexo que às vezes pode trazer uma experiência enfadonha.

Figura 06 - Pac-Man e os Fantasmas



Saiba mais:

Desde sua criação há 37 anos, o Pac-Man se transformou em um dos clássicos da indústria de games. A história dele é recheada de fatos desconhecidos mesmo entre os fãs. Para você conhecer um pouco da história desse game, [acesse aqui](#), e para algumas curiosidades, [acesse aqui](#).

Onde a IA é usada nos jogos?

Apesar de normalmente se associar as técnicas de IA aos comportamentos dos personagens de um jogo, elas não são limitadas unicamente a esse uso. Imagine, por exemplo, um jogo 3D em terceira pessoa, no qual a câmera precisa focar nos elementos principais do cenário ao mesmo tempo em que aponta o caminho a ser percorrido pelo jogador, desviando de possíveis obstáculos que possam ocultar a visão dele. Qual é a melhor sequência de tomadas da câmera a ser executada? Como em um filme, as pistas e a ligação com os elementos do cenário precisam, às vezes, estar relacionadas. Quem faz filme sabe da importância das “tomadas”, do efeito que uma aproximação do rosto do personagem faz, do ritmo da câmera a ser definido em momentos mais tensos ou dos movimentos panorâmicos quando deseja-se dar uma visão geral do cenário. Em um filme, todas essas decisões são realizadas pelo diretor. E em um jogo, quem toma essas decisões?

Muitos jogos usam técnicas de IA para definir qual o ângulo e a abertura da melhor tomada em um determinado momento do jogo. Além disso, há também o problema de como fazer a transição de uma tomada para a outra. Como será?

Rápida ou lenta? Como no cinema, a tomada da câmera precisa ser inteligente o suficiente para realçar os elementos importantes da cena, independentemente das interações do jogador.

No exemplo da Figura 07, a visão em terceira pessoa do jogo evidencia que há uma trilha abaixo.

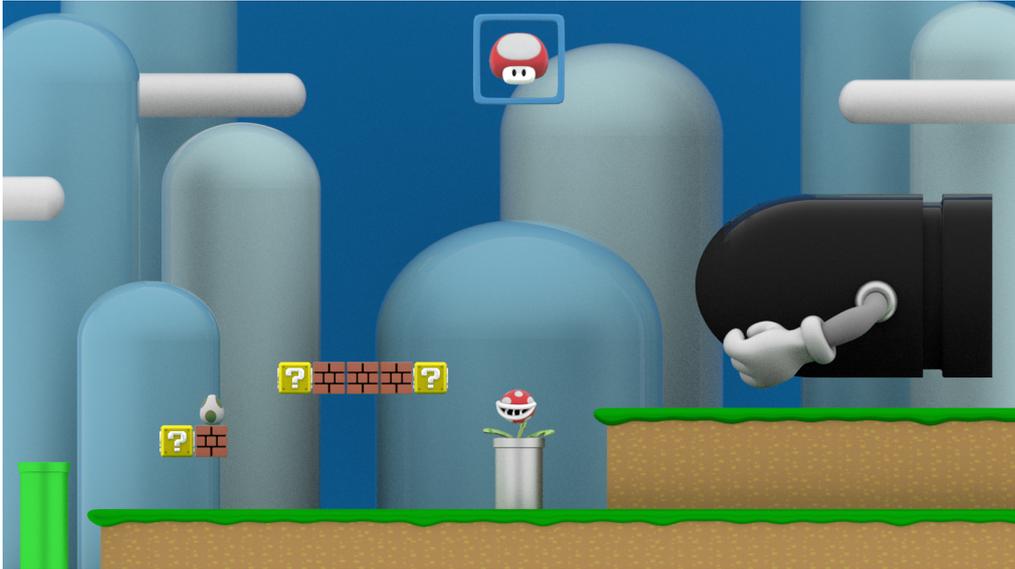
Figura 07 - Visão em terceira pessoa de uma cena do jogo Assassin's Creed



Fonte: TERRY MAJAMAKI. Disponível em: <<http://majamaki.com/2013/01/assassins-creed-iii/>>. Acesso em: 28 mar. 2018.

Outro exemplo de uso das técnicas de IA é o caso em que o *designer* do jogo, para aumentar a sua jogabilidade, deseja que os cenários (ou níveis) sejam criados dinamicamente. Ou seja, os cenários não são pré-definidos pelo designer, mas criados durante o desenrolar do próprio jogo. Para que você visualize melhor o que está se colocando aqui, basta pensar em um jogo simples como o **Super Mario**. É um jogo de plataforma 2D, com áreas de percurso (plataformas) e adversários posicionados em locais estratégicos. Imagine se para criar um nível mais dinâmico, fossem posicionados os elementos do jogo em lugares completamente aleatórios. Será que teria solução (o personagem conseguiria chegar ao final da fase)? Mesmo que haja solução, será que seria muito fácil ou muito difícil para o jogador? Será que isso o faria desestimular do jogo? Enfim, a criação de um nível de forma dinâmica, mesmo em jogos simples como o **Super Mario**, é uma tarefa complexa, que requer técnicas de IA para gerar um cenário que possua solução e dificuldade que seja adequada à evolução do jogador ao longo dos níveis.

Figura 08 - Criação de níveis em Super Mario



Mais um exemplo de uso de IA é prover o jogo de inteligência suficiente para identificar o grau de habilidade do jogador e, com isso, poder adaptar o nível de dificuldade a ser proposto por ele. **Esse tópico é muito importante para a maioria dos jogos atuais, porque é essencial cativar o jogador desde o início e conservá-lo jogando o máximo de tempo possível.** Se o jogo for muito fácil no início, ele pode ficar chato e desinteressante. Por outro lado, se for muito difícil, muitos jogadores vão deixá-lo de lado se não enxergarem a possibilidade de experimentarem o “sabor da vitória”. Então, o segredo é adaptar o jogo ao usuário. Como fazer isso? Com técnicas de aprendizado de máquina. Elas permitem à IA aprender qual o nível do jogador, classificando-o em diferentes categorias a partir das análises de suas ações e dos resultados obtidos. O segredo é então adaptar-se ao jogador!

Além desses usos, há obviamente a utilização de técnicas de IA para fazer com que o comportamento dos personagens seja mais realista. E isso é o que será discutido na próxima seção.

IA na definição do comportamento dos personagens

Foram citados alguns exemplos do uso de **Game AI**, no entanto, nessa disciplina será abordado apenas um deles: **a criação de personagens**, ou melhor, **a criação do comportamento dos personagens**. O intuito de utilizar técnicas de IA na

definição do comportamento dos personagens não controlados pelo jogador (os NPCs) é possibilitar ao jogador uma melhor imersão no contexto do mundo virtual do jogo. Por exemplo, espera-se que um jogo de futebol tenha NPCs que atuem de forma realista, que façam os mesmos movimentos dos “craques da bola”. Se isso não ocorrer, a ilusão de imersão no mundo virtual do jogo pode ser quebrada e o jogo pode perder a graça. Imagine como seria um jogo de futebol no qual os NPCs se movimentam como espaçonaves! Seria qualquer coisa, menos jogo de futebol.



Para uma melhor compreensão desse assunto, a disciplina foi dividida em dois grandes tópicos: Movimentação e Tomada de decisão. No primeiro, serão abordados os problemas associados à criação de movimentos naturais dos personagens de jogos. Ou seja, você irá aprender como evitar que os personagens passem pelo mico de ficar batendo com a cabeça no chão na tentativa de, um dia, encontrar o caminho para seguir, ou de não conseguirem dar a volta em uma árvore para alcançar o objetivo que se encontra logo depois dela. No segundo, você irá aprender várias formas de como criar ilusão de inteligência nos personagens, fazendo-os escolher ações ou estados que pareçam “normais”, ou seja, como se fosse uma pessoa controlando seu próprio *avatar*.

Navegação e descoberta de caminhos

Quem vê os jogos de hoje, com NPCs indo de um lado para o outro, não sabe o quanto é complexo para eles realizarem tarefas que, para nós, são triviais. Uma delas é o deslocamento nos mundos virtuais. Como no exemplo que foi citado na apresentação da disciplina, do pobre cavalo tentando atravessar o chão. Agora, irei apresentar outro, dessa vez, sem esse problema.

Só para ter uma ideia, imagine um jogo com muitos NPCs ao mesmo tempo, como os jogos de Estratégia em tempo real (ou RTS – *Real-Time Strategy*). Posso citar as séries StarCraft, Age of Empires, Age of Mythology, Total War e os sucessos League of Legends e DOTA (bem, esses últimos são chamados de MOBA – *Multiplayer Online Battle Arena*, que é, na verdade, uma subcategoria dos jogos RTS). Como ilustração, usarei um jogo menos conhecido chamado 0 A.D., que você pode jogar enquanto estuda. 😄 Como se trata de um jogo livre e de código aberto, é permitido baixar sem a preocupação de estar usando um jogo pirata! Como desenvolvedor de jogos, você deve se preocupar com esses piratas que não valorizam o seu trabalho, não é mesmo? 😊



Saiba mais!

O 0 A.D. é pouco conhecido, porém, é um jogo de estratégia em tempo real de plataforma aberta ou multiplataforma. Se quiser saber mais sobre o seu desenvolvimento, [clique aqui](#) e divirta-se!

A Figura 09 apresenta uma captura de tela desse jogo. Nela, é possível observar tropas de duas equipes se enfrentando e também identificar um problema clássico de navegação com vários NPCs: passagens mais estreitas do que a formação da tropa gerando uma confusão no início da passagem. Por esse problema, muitas vezes, vemos NPCs “rodando” de um lado para o outro sem saber para onde ir por seu caminho estar bloqueado. A movimentação de uma tropa humana não faria

isso. O guerreiro que ficasse rodando no mesmo ponto seria considerado maluco e, possivelmente, ganharia uma dispensa no próximo ataque. Se bem que, numa guerra, é bem possível que esses considerados malucos ganhem é uma promoção.

Figura 09 - Captura de tela do jogo 0 A.D. com vários NPCs com problemas de deslocamento em um cenário



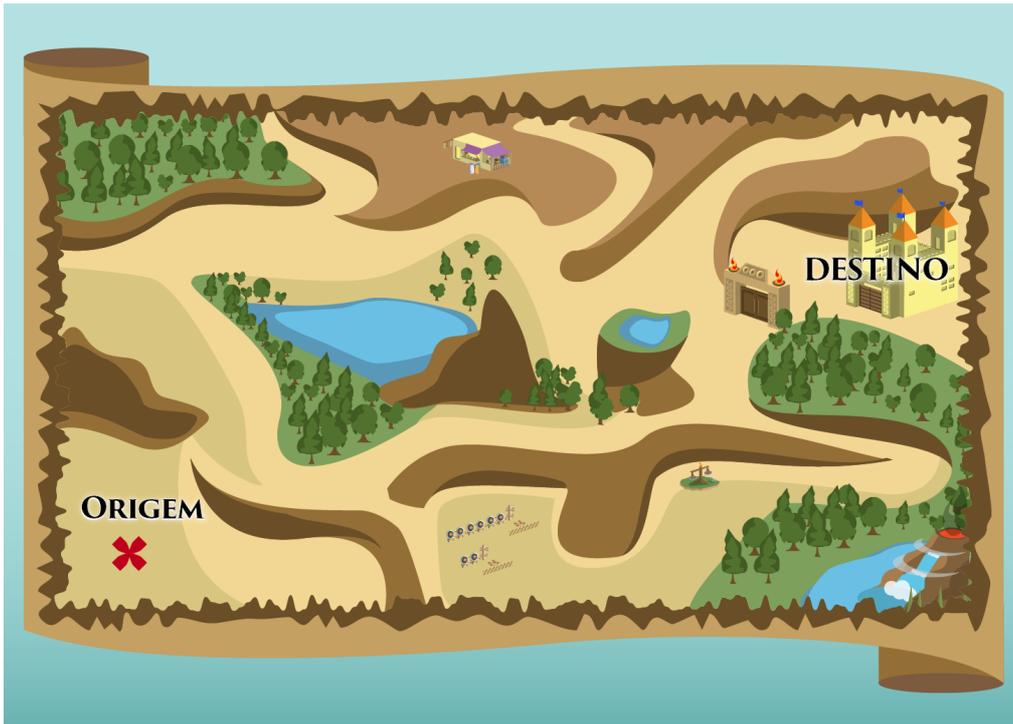
Fonte: WILDFIRE GAMES. Disponível em: <<http://play0ad.com/>>. Acesso em: 23. mar. 2018.

Há inúmeras formas de resolver os problemas de movimentação que foram citados. As mais utilizadas são **a utilização de algoritmos de descoberta de caminhos** (ou *pathfinding*) e a de **comportamentos de navegação** (ou *steering behaviors*). Para a primeira, é necessário ter um “mapa” do ambiente por onde os NPCs irão se deslocar. Com o mapa, utilizam-se algoritmos de busca para achar o menor caminho entre a localização atual do NPC e a localização para onde ele quer ir. Claro, alguém com “inteligência” usaria o menor caminho.

Veja, por exemplo, a Figura 10, nela está o mapa onde o NPC deverá encontrar o melhor caminho entre sua localização (origem) e onde ele quer ir (destino).

Mesmo sabendo qual é o menor caminho entre a origem e o destino, você deverá instruir o computador a calculá-lo. Mas como isso poderá ser feito?

Figura 10 - Exemplo de mapa com obstáculos



Uma opção para o cálculo é o NPC testar todos os caminhos possíveis e, com isso, achar qual o menor deles. Entretanto, lembre-se de que um jogo normalmente requer tempo rápido de resposta, uma vez que precisa gerar cerca de 30 quadros por segundo. Ou seja, o NPC precisa verificar todas as possibilidades de caminho para calcular seu percurso ótimo em menos de 0,03 segundos. Enfatizando que estou deixando de lado o tempo de renderização do jogo, que também é alto. Se o cálculo levar mais tempo do que isso, você terá *lag* (é um termo em inglês que se refere a atrasos que se podem apresentar na comunicação entre computadores (por causa de problemas na internet, por exemplo), podendo ser ampliado a outras situações, como comunicação via satélite ou da tela do computador com o jogador.) no jogo e vários jogadores reclamando (falando mal de você). Então, se o cenário for pequeno e se o jogo apresentar um ou dois NPCs, talvez nem percebam *lag*. Mas se o cenário for grande e/ou tiver uma quantidade grande de NPCs, como os apresentados na Figura 09, vai precisar de uma boa otimização nas técnicas de descoberta de caminhos para que não haja nenhuma reclamação. Até aqui, tudo bem? Então, prepare-se para aprender como fazer isso ainda nessa disciplina!

Tomada de decisão

Neste tópico, você irá entender como os NPCs podem escolher suas ações explorando várias alternativas, das mais simples às de complexidade média. Porém, mesmo com as mais simples, você poderá elaborar para os NPCs comportamentos rebuscados.

Você vai conhecer agora uma técnica chamada de **Máquina de Estados Finitos - MEF**. Ela permite modelar o comportamento de um NPC através da descrição dos possíveis estados em que ele pode se encontrar e dos eventos que podem ocorrer para fazê-lo mudar de estado. Por exemplo, suponha que você esteja modelando o comportamento de um caçador. Uma possível forma de elaborar seu comportamento usando **MEF** é definir três estados para ele, como ilustra a Figura 11:

1. Ou ele está procurando uma presa;
2. Ou ele está perseguindo uma presa;
3. Ou ele está atirando na presa.

Figura 11 - Esquema de Máquina de Estado Finito controlando o comportamento de um NPC caçador



As mudanças de estados são reguladas por eventos que ocorrem no cenário. Assim, o caçador passa a perseguir uma presa (estado 2) quando avistar uma e volta a procurar uma presa (estado 1) quando ela sai do alcance de sua vista. Ao perseguir a presa, se esta encontra-se a uma distância segura para atirar, ele pode parar de perseguir para poder mirar e atirar na presa, e fica assim enquanto a distância ainda é boa para o tiro. Se a presa se distanciar muito, o caçador terá de parar de atirar e voltar a perseguir a presa.

Aliadas a essas mudanças de estado, pode-se incluir probabilidades de transição, em que um mesmo evento poderá gerar dois ou mais eventos possíveis, cada um com sua probabilidade de ocorrer. Chamam essa adaptação da **MEF** de **Máquina de Estados Finitos Probabilística** ou **MEFP**.

Na Figura 12, pode-se ver que há duas possíveis transições a partir do estado 1, ambas quando o evento **A** ocorrer. Mas, como saber para qual estado o NPC irá quando **A** ocorrer? O diagrama indica que há uma probabilidade de 60% de ele ir para o estado 3 e uma probabilidade de 40% dele ir para o estado 2. Ou seja, em aproximadamente 60% dos casos ele irá para o estado 3 e em 40% dos casos irá para o estado 2. De forma similar, quando o evento **B** ocorrer e o NPC estiver no estado 3, ele voltará para o estado 1 cerca de 70% das vezes e, nos casos restantes, 30%, ele irá para o estado 2.

Figura 12 - Esquema de Máquina de Estado Probabilística, na qual um evento pode gerar uma das possíveis transições



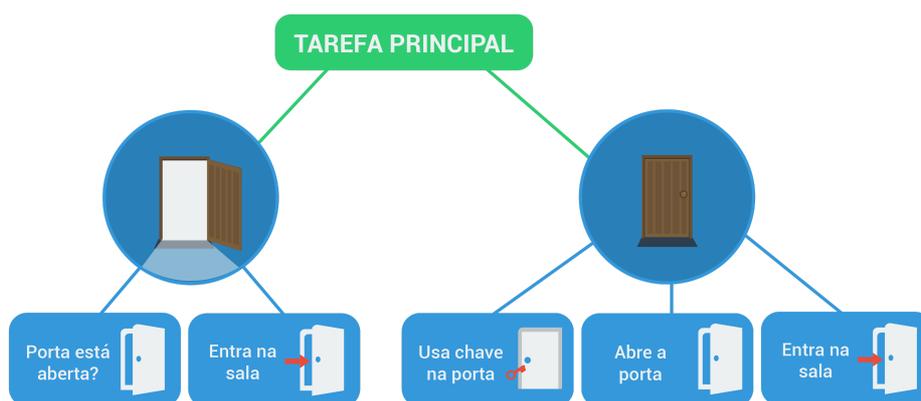
Veja que o mecanismo é simples, mas tem o poder de definir comportamentos bastante complexos, inclusive associando probabilidade às transições de forma que o NPC se comporte diferentemente a cada nova partida ou a cada situação similar. A complexidade dessa técnica não reside na técnica em si, mas do número de estados

que você está atribuindo ao seu NPC e da quantidade de eventos com que se está lidando. Quando o número de estados cresce, fica muito difícil gerenciar todas as possibilidades de transições de forma adequada.

Uma técnica alternativa a ser usada para minimizar o “ninho de cobra” de estados e transições da **MEF** é a **Árvore de Comportamento** (ou *Behavior Tree*). Nessa técnica, organizamos os possíveis estados em uma estrutura hierárquica, como nos organogramas de empresas. Porém, em vez de você organizar os comportamentos em torno de estados, você estrutura através de “tarefas”. Uma tarefa pode, então, ser decomposta em subtarefas e estas em outras subtarefas, assim por diante, criando uma ramificação a partir da tarefa principal. O termo “árvore” vem dessa ramificação.

A Figura 13 ilustra uma árvore cujo objetivo é modelar o comportamento de um NPC para entrar em uma sala. Há duas alternativas para alcançar esse objetivo: em uma, a porta da sala encontra-se aberta e na outra, fechada. Bem, veja com mais detalhes a simbologia desse formalismo em futuras aulas. Por enquanto, quero apenas dar uma ideia de qual tipo de ferramenta é possível utilizar para modelar o comportamento dos personagens para que você, futuro desenvolvedor de jogos, possa ter uma visão geral da disciplina.

Figura 13 - Esquema de organização de estados em uma Árvore de Comportamentos (Behavior Tree)





Resumo

Espero que você tenha assimilado as principais ideias relacionadas à Inteligência Artificial e como elas são aplicadas nos jogos digitais. Você viu o que os pesquisadores da área de IA para jogos (Game AI) consideram importante no desenvolvimento da IA de um jogo e em que ela se diferencia do que chamamos de “IA tradicional”. Também foram apresentados os objetivos principais da Game AI, como: construir técnicas que não consumam muito recurso computacional (“pseudo tempo-real”), e, compreender que nem sempre a solução mais eficaz é a melhor opção a ser tomada pela IA do jogo, pois ela precisa se adaptar ao nível do jogador. Você ainda aprendeu que técnicas simples podem trazer excelentes resultados, como por exemplo as técnicas aplicadas em jogos clássicos como o Pac-Man, mas que também não se deve deixar de lado técnicas avançadas, as que utilizam aprendizagem de máquina, como as usadas no jogo Creatures. Além disso, entendeu onde as técnicas de IA são, em geral, utilizadas nos jogos, podendo citar o cálculo de navegação, a modelagem de comportamentos de personagens NPCs (personagens não controlados por jogadores), bem como a definição de tomadas cinematográficas de câmeras, a geração de cenários de forma dinâmica, a adaptação do jogo (ou balanceamento dele) em função do jogador, entre outros.

Espero que tenha gostado da visão geral dada nesta primeira aula. Ela foi bastante teórica, com apresentação de novos conceitos e informações para você assimilar e, por enquanto, nada para praticar. Mas, não se preocupe, você colocará a mão na massa em breve!

Até a próxima!



Glossário

Copyright: é um direito autoral, a propriedade literária, que concede ao autor de trabalhos originais direitos exclusivos de exploração de uma obra artística, literária ou científica, proibindo a reprodução.

Lag: é um termo em inglês que se refere a atrasos que se podem apresentar na comunicação entre computadores (internet, por exemplo), podendo ser ampliado a outras situações, como comunicação via satélite.

NPC: *Non-Player Character*, ou, em português, Personagem não jogável – PNJ.

Habitat: é o ambiente natural em que vive um organismo, ou o ambiente físico que envolve uma população de espécies.



Atividade

1. Das técnicas de IA utilizadas em jogos que foram citadas nessa aula (descoberta de caminho, comportamentos de navegação, máquinas de estado finito e árvore de comportamento), qual delas você acha que seria a mais apropriada para:
 - a. fazer um carro (em um jogo de corrida) seguir o caminho da pista?
 - b. fazer um NPC passar a perseguir o jogador assim que ele for visível ao NPC?
 - c. organizar o comportamento de um NPC em termos de “objetivos”?
 - d. fazer com que um NPC saiba os corredores por onde passar em um labirinto?

[Clique aqui](#) para verificar suas respostas.

Respostas

1. Das técnicas de IA utilizadas em jogos que foram citadas nessa aula (descoberta de caminho, comportamentos de navegação, máquinas de estado finito e árvore de comportamento), qual delas você acha que seria a mais apropriada para:

a. fazer um carro (em um jogo de corrida) seguir o caminho da pista?

R. O uso de comportamentos de navegação é o mais indicado nesse caso devido à sua similaridade com as mecânicas da Física. Haveria forças de atração no caminho a ser percorrido pelo carro, fazendo-o seguir a pista.

b. fazer um NPC passar a perseguir o jogador assim que ele for visível ao NPC?

R. O uso de máquinas de estado finito é mais evidente, podendo ocorrer pelo menos 2 estados: 1) o NPC não faz nada (fica esperando o jogador ser visível a ele); e 2) ele passa a perseguir o jogador (quando o evento "jogador visível" ocorrer). Quando o jogador deixar de estar visível, o NPC volta ao estado de "fazer nada".

c. organizar o comportamento de um NPC em termos de "objetivos"?

R. A organização de um comportamento em termos de objetivos é mais fácil de ser implementada através de uma árvore de comportamento. O objetivo do NPC será o elemento de maior hierarquia na árvore.

d. fazer com que um NPC saiba os corredores por onde passar em um labirinto?

Nesse cenário, o NPC precisaria conhecer qual o melhor caminho para ir de um ponto a outro do labirinto. Esse é o tipo de problema que algoritmos de descoberta de caminhos resolve.



Midiateca

Quer aprofundar/reforçar um pouco os conceitos vistos na aula?

Os primeiros dois vídeos trazem uma conceituação do que é IA, com alguns exemplos para você ver o quanto ela já está presente nas nossas vidas!

Prepare o coração e o inglês (ou liga o tradutor do youtube e se divirta!), porque os vídeos abaixo são uma introdução (o primeiro) e uma palestra (o segundo) sobre as técnicas que são consideradas IA para Jogos!



Referências

MILLINGTON, Ian; FUNGE, John. **Artificial Intelligence for Games**. 2. ed. Morgan Kaufmann Publishers. 2009.