



CENTRO DE CIÊNCIAS E TECNOLOGIAS

CURSO DE CIÊNCIA DA COMPUTAÇÃO

INTELIGÊNCIA ARTIFICIAL

**EXPLICANDO OS PRIMÓRDIOS DA INTELIGÊNCIA ARTIFICIAL NOS
JOGOS ELETRÔNICOS: COMO FUNCIONA O JOGO PONG**

**GABRIEL MARQUES DE AMARAL GRAVINA
PAULO ROBERTO CALDAS SIQUEIRA JÚNIOR**

NOVEMBRO, 2022

Apresentação

Os jogos, como um todo, sempre foram uma boa fonte de diversão e aprendizado ao longo dos tempos. Sempre foram utilizados para entreter tanto crianças como adultos, além de serem utilizados como ferramentas lúdicas para ensino em determinadas áreas, principalmente na pré-escola.

Com o passar do tempo, e com a evolução da tecnologia e da computação, era natural que esses jogos evoluíssem também em direção a essa área. Mas o maior desafio era fazer uma interação homem-máquina que parecesse o menos artificial possível, além de permitir que esses jogos “pensassem” de maneira autônoma, e é aí que a inteligência artificial entra.

Resumo

O objetivo dessa dissertação é explicar o funcionamento da inteligência artificial do jogo “Pong”, um dos primeiros jogos eletrônicos criados. Em um período em que não se tinham muitas referências sobre inteligência artificial, mostraremos como uma inteligência artificial rudimentar ainda pode ser eficiente para o seu fim proposto, e como ela se encaixa nas classificações atuais.

Sumário

1. O jogo Pong	5
1.1. Jogabilidade	5
1.2. Desenvolvimento	6
1.3. Legado	7
2. A inteligência artificial de Pong	8
2.1. Descrevendo a inteligência artificial do jogo Pong	8
2.2. Função de agente	10
2.2.1. Predição da barra de colisão	10
2.2.2. Predição da bolinha	11
2.2.3. Calibrando a reação do agente	12
3. Conclusão	13
4. Referências	14

1. O jogo Pong

Pong é um jogo eletrônico de esporte de arcade com temática de tênis de mesa, com gráficos bidimensionais, desenvolvido pela Atari e lançado originalmente em 1972. Foi um dos primeiros jogos de arcade; foi criado por Allan Alcorn como um exercício de treinamento atribuído a ele pelo cofundador da Atari, Nolan Bushnell, mas Bushnell e o cofundador da Atari, Ted Dabney, ficaram surpresos com a qualidade do trabalho de Alcorn e decidiram fabricar o jogo. Bushnell baseou o conceito do título em um jogo eletrônico de pingue-pongue incluído no Magnavox Odyssey, o primeiro console doméstico de jogos eletrônicos. Em resposta, a Magnavox mais tarde processou a Atari por violação de patente.

Pong foi o primeiro jogo comercialmente bem-sucedido e ajudou a estabelecer a indústria de jogos eletrônicos junto com o Magnavox Odyssey. Logo após seu lançamento, várias empresas começaram a produzir jogos que imitavam de perto sua jogabilidade. Eventualmente, os concorrentes da Atari lançaram novos tipos de jogos que se desviaram do formato original de Pong em vários graus, e isso, por sua vez, levou a Atari a incentivar sua equipe a ir além de Pong e produzir jogos mais inovadores.

1.1. Jogabilidade

Pong é um jogo eletrônico de esporte em duas dimensões que simula um tênis de mesa. O jogador controla uma paleta (barra vertical) no jogo movendo-a verticalmente no lado esquerdo da tela, e compete contra o computador ou outro jogador que controlam uma segunda raquete no lado oposto. Os jogadores usam suas paletas para acertar a esfera (bola) e mandá-la para o outro lado. A paleta é dividida em oito segmentos, com o segmento central retornando a bola em um ângulo de 90° em relação a paleta e os segmentos externos retornando a bola em ângulos cada vez menores. A bola aumenta de velocidade cada vez que é rebatida, reiniciando a velocidade caso algum dos jogadores não acerte a bola. O objetivo é fazer mais pontos que seu oponente, fazendo com que o oponente não consiga retornar a bola para o outro lado.

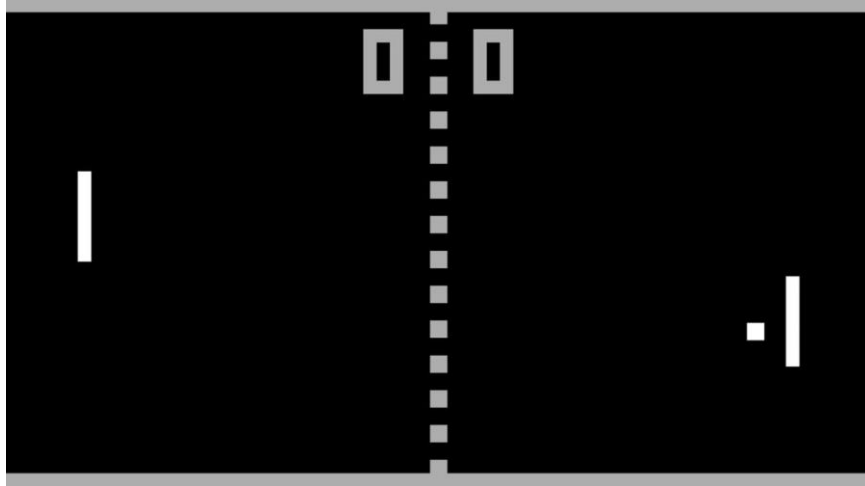


Figura 1: Interface do jogo Pong

1.2. Desenvolvimento

Criado por Allan Alcorn (engenheiro e cientista da computação), o que parecia ser uma ideia “simples” se mostrou ser algo genial. Pong foi desenvolvido em meio a testes realizados pelo engenheiro quando ele ainda não tinha tanta experiência em jogos eletrônicos.

Foi aí que Nolan Bushnell (fundador da Atari) resolveu fazer alguns ajustes pontuais para deixar o jogo mais acessível. Então, melhorando o ângulo da câmera e deixando a física da bolinha mais previsível, o jogo foi ficando mais intuitivo e ganhando forma.

Além disso, Bushnell também adicionou efeitos sonoros simples e aumentou um pouco a velocidade das raquetes para deixar o jogo “mais desafiador” na medida certa. Foi assim que, em 29 de novembro de 1972, a primeira máquina do Pong foi instalada em San Francisco, em um “barzinho” ao lado da sede da Atari, no estado norte-americano da Califórnia.



Figura 2: Primeira máquina a rodar o jogo Pong

1.3. Legado

Muitas publicações consideram Pong o jogo que lançou a indústria de videogames como um negócio lucrativo. O autor de vídeo games David Ellis vê o jogo como um marco no sucesso da indústria de vídeo games, e considera o jogo de arcade "um dos títulos mais significantes da história". Kent atribui o "fenômeno arcade" a Pong e aos jogos da Atari que o seguiram, e considera o lançamento da versão doméstica um início bem-sucedido dos consoles. Bill Loguidice e Matt Barton do Gamasutra se referem ao lançamento do jogo como um começo de um novo meio de entretenimento, e comentou que sua jogabilidade simples e intuitiva o fizeram ser um sucesso. Muitas outras empresas que produziram suas próprias versões de Pong eventualmente se tornaram muito conhecidas no meio.

2. A inteligência artificial de Pong

Pong é um jogo simples, então podemos fornecer uma inteligência artificial com uma estratégia de 2 partes: modelo por tempo de reação e modelo de precisão.

O modelo da inteligência artificial do jogo Pong pode ser classificado dentro da inteligência artificial como “agente racional”. O objetivo da inteligência artificial do jogo é prever o movimento do jogador ao rebater a bola, calcular o local para onde a bola irá se deslocar, e se mover até esse local para rebater a bola.

A inteligência artificial não está preocupada se um humano faria esta tarefa deste mesmo modo. O sistema está mais interessado em saber se está fazendo a atividade correta e se continuará a fazer a atividade correta.

2.1. Descrevendo o agente da inteligência artificial do jogo Pong.

Podemos classificar esse agente como uma agente de software, e abaixo iremos descrever o seus PEAS (Performance, Environment, Actuators, Sensors).

Medida de desempenho	Prever a trajetória da bolinha, não deixar a bolinha passar pela barra, fazer a bolinha passar pela barra do oponente, marcar pontos
Ambiente	Campo do jogo, barra de colisão, bolinha
Atuadores	Função para identificar a trajetória da bolinha, Função para mover a barra de colisão
Sensores	Entrada pelo teclado

Também podemos classificar o ambiente desse atuador, da seguinte forma:

Propriedade	Justificativa
Parcialmente observável	O agente tem acesso a toda área de atuação, porém, não pode determinar a próxima jogada do oponente
Estocástico	Não é possível determinar todas as ações do jogador
Episódico	Percepção e execução de uma única ação
Estático	O ambiente do jogo é sempre o mesmo
Discreto	Por mais que não seja possível determinar as ações do jogador, o ambiente do jogo sempre será o mesmo, não mudando a forma de atuação do agente
Multi Agente	Existem dois agentes atuando durante todo o jogo.

Além do mais, classificamos esse agente como um agente reativo simples, pois as regras condição-ação (regras se-então) fazem uma ligação direta entre a percepção atual e a ação, e a decisão correta pode ser tomada com base apenas na percepção atual.

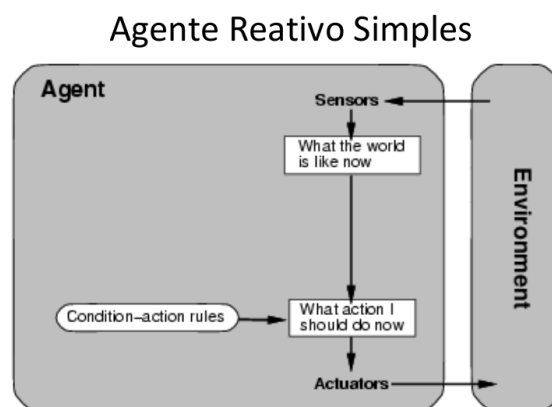


Figura 3: Fluxograma de um agente reativo simples

2.2. Função de agente

Iremos usar como base para estudo uma inteligência artificial construída em Python para mostrar como funciona a função de agente do jogo. Não foi encontrado o código original do jogo, visto que ele é protegido por direitos autorais. Mas o código em Python consegue emular perfeitamente o funcionamento do jogo.

A inteligência artificial se divide em dois agentes, um para movimentar a barra de colisão e outro para prever a trajetória da bolinha.

2.2.1. Predição da barra de colisão

A função de agente da barra de colisão segue a seguinte estratégia: Se a bola estiver se afastando, não faça nada; Caso contrário, preveja onde a bola encontrará a borda barra de colisão; Se tivermos uma previsão, suba ou desça para encontrá-la.

```
ai: function(dt, ball) {
    if (((ball.x < this.left) && (ball.dx < 0)) ||
        ((ball.x > this.right) && (ball.dx > 0))) {
        this.stopMovingUp();
        this.stopMovingDown();
        return;
    }

    this.predict(ball, dt);

    if (this.prediction) {
        if (this.prediction.y < (this.top + this.height/2 - 5)) {
            this.stopMovingDown();
            this.moveUp();
        }
        else if (this.prediction.y > (this.bottom - this.height/2 + 5)) {
            this.stopMovingUp();
            this.moveDown();
        }
        else {
            this.stopMovingUp();
            this.stopMovingDown();
        }
    }
},
```

2.2.2. Predição da bolinha

A função de agente de previsão da bolinha é um pouco mais complicada. A estratégia usada é: Primeiro, não se faz nada até se atingir o tempo de reação (ficar parado até a função de agente ter uma resposta a ação do jogador); Em segundo lugar, calcula-se a interceptação da bola assumindo uma quadra infinitamente alta (isso impede possíveis erros de cálculo de trajetória); Em terceiro lugar, rebata repetidamente a previsão nas paredes superior e inferior até atingir a barra de colisão (calcular todas as possíveis colisões da bolinha com a lateral da quadra e que faça ângulo onde a barra de colisão esteja); Por fim, aplique nosso fator de erro para modelar isso como um 'palpite' (usa-se uma tabela tempo de reação versus fator de erro para modelar a dificuldade do jogo).

```
predict: function(ball, dt) {
  if (this.prediction &&
      ((this.prediction.dx * ball.dx) > 0) &&
      ((this.prediction.dy * ball.dy) > 0) &&
      (this.prediction.since < this.level.aiReaction)) {
    this.prediction.since += dt;
    return;
  }

  var pt = Pong.Helper.ballIntercept(ball, {left: this.left, right:
this.right, top: -10000, bottom: 10000}, ball.dx * 10, ball.dy * 10);
  if (pt) {
    var t = this.minY + ball.radius;
    var b = this.maxY + this.height - ball.radius;

    while ((pt.y < t) || (pt.y > b)) {
      if (pt.y < t) {
        pt.y = t + (t - pt.y);
      }
      else if (pt.y > b) {
        pt.y = t + (b - t) - (pt.y - b);
      }
    }
    this.prediction = pt;
  }
  else {
    this.prediction = null;
  }

  if (this.prediction) {
    this.prediction.since = 0;
    this.prediction.dx = ball.dx;
    this.prediction.dy = ball.dy;
    this.prediction.radius = ball.radius;
    this.prediction.exactX = this.prediction.x;
    this.prediction.exactY = this.prediction.y;
```

```

    var closeness = (ball.dx < 0 ? ball.x - this.right : this.left -
ball.x) / this.pong.width;
    var error = this.level.aiError * closeness;
    this.prediction.y = this.prediction.y + Game.random(-error, error);
  }
},

```

2.2.3. Calibrando a reação do agente

Para tornar o agente justo e competitivo ao mesmo tempo, usa-se uma tabela onde o agente fica com um tempo de reação maior ou menor a depender do placar do jogo (quanto mais pontos o jogador tiver, menor será o tempo de reação e suscetibilidade a erros). Isso está diretamente ligado ao desempenho esperado do agente, pois caso se fizesse um agente que tivesse um tempo de reação extremamente baixo, seria impossível para o jogador marcar pontos.

Abaixo segue a tabela de reação do agente.

```

Levels: [
  {aiReaction: 0.2, aiError: 40}, // 0: ai is losing by 8
  {aiReaction: 0.3, aiError: 50}, // 1: ai is losing by 7
  {aiReaction: 0.4, aiError: 60}, // 2: ai is losing by 6
  {aiReaction: 0.5, aiError: 70}, // 3: ai is losing by 5
  {aiReaction: 0.6, aiError: 80}, // 4: ai is losing by 4
  {aiReaction: 0.7, aiError: 90}, // 5: ai is losing by 3
  {aiReaction: 0.8, aiError: 100}, // 6: ai is losing by 2
  {aiReaction: 0.9, aiError: 110}, // 7: ai is losing by 1
  {aiReaction: 1.0, aiError: 120}, // 8: tie
  {aiReaction: 1.1, aiError: 130}, // 9: ai is winning by 1
  {aiReaction: 1.2, aiError: 140}, // 10: ai is winning by 2
  {aiReaction: 1.3, aiError: 150}, // 11: ai is winning by 3
  {aiReaction: 1.4, aiError: 160}, // 12: ai is winning by 4
  {aiReaction: 1.5, aiError: 170}, // 13: ai is winning by 5
  {aiReaction: 1.6, aiError: 180}, // 14: ai is winning by 6
  {aiReaction: 1.7, aiError: 190}, // 15: ai is winning by 7
  {aiReaction: 1.8, aiError: 200} // 16: ai is winning by 8
],

```

3. Conclusão

O jogo Pong foi um divisor de água para a indústria dos jogos eletrônicos, e foi graças a ele que grandes empresas se consolidaram nesse mercado. Sua importância histórica é inegável, e ele também exerce uma grande influência no campo da inteligência artificial em uma época em que esse campo da computação ainda era pouco discutido.

Ainda em uma época em que a inteligência artificial era apenas um campo teórico e com poucas aplicações práticas, a inteligência artificial do jogo já conseguia se sobrepôr a outros setores dentro da própria indústria. Outra característica interessante do seu agente é o fato de que ela pode ser classificada de maneira correta nas definições formais aceitas nesse campo de estudo, mesmo sendo criada em uma época em que essa classificação não existia.

Conclui-se que podemos compreender o agente do jogo não só como uma inteligência artificial funcional, como também classificável dentro dos parâmetros de estudo da inteligência artificial.

4. Referências

ZAMBIASI, Saulo Popov. Inteligência artificial aplicada a sistemas de informação. 2008, ICPG Criciúma – SC;

ZADROZNY, Bianca. Inteligência artificial. 2015, UFF Rio de Janeiro, RJ;

PONG. Disponível em: <https://pt.wikipedia.org/wiki/Pong>. Acessado em 26 de novembro de 2022;

PONG – COMPUTER AI. Disponível em: <https://codeincomplete.com/articles/javascript-pong/part5/>. Acessado em 26 de novembro de 2022.