

USO DA INTELIGÊNCIA ARTIFICIAL NA CRIAÇÃO DE UM AGENTE JOGADOR DE GO

Pablo Vieira dos Santos^{1*}, André Bevilaqua²

¹Graduando em Sistemas de Informação, pela Universidade Luterana do Brasil – ULBRA, Unidade Itumbiara - GO, *pablovieira@outlook.com, ²Mestre em Inteligência Artificial, Professor da Universidade Luterana do Brasil – ULBRA, Unidade Itumbiara – GO.

PALAVRAS-CHAVE: Go, Agente Jogador, Algoritmos Genéticos.

RESUMO – Jogos combinatórios são aqueles que possuem informação completa, basicamente, não dependem do fator sorte e os jogadores sabem tudo sobre a atual posição no jogo. Existem diversos jogos deste tipo e alguns deles, como o Go, que será o objeto de estudo deste trabalho, possuem algumas relações com problemas do cotidiano, situações reais complexas que se assemelham em alguns aspectos com situações do jogo. O Go é um jogo de tabuleiro (conforme mostrado na figura 1) que, de acordo com a tradição chinesa, foi criado há mais de 4000 anos. Devido à complexidade das situações que acontecem no decorrer de uma partida, a implementação de ótimos agentes automáticos se tornou um desafio para a área da computação, mas a utilização de técnicas de Inteligência Artificial (IA) na construção desses agentes, resultou em melhores desempenhos e jogadores de Go extremamente competentes. Assim, o trabalho visa a criação de uma heurística para a construção de um jogador automático, ou agente jogador de Go, utilizando Algoritmos Genéticos (AGs). O objetivo é utilizar o processo evolutivo dos AGs no desenvolvimento de indivíduos que joguem de forma competitiva o Go. A ideia é que, com base nas regras do jogo, o agente consiga desempenhar bem o papel de jogador automático e possa, a cada geração do AG, melhorar a qualidade das jogadas a fim de buscar a competitividade desejada.

INTRODUÇÃO

Jogos têm sido por vezes utilizados em teste de novos conceitos e técnicas de Inteligência Artificial (IA), visto que

possuem uma relativa simplicidade se comparado a problemas mais complexos, como situações do mundo real (Blackman, 2003). De acordo com Shah, Singh e Shah (2012), a resolução de jogos tem se tornando uma das extensivas áreas de estudo da IA e, em especial, os jogos de tabuleiro, exigem uma inteligência sofisticada para encontrar uma melhor definição do problema.

Através dos jogos, trabalhos de IA podem ter sua eficiência medida com relação à capacidade de adquirir inteligência sem que haja a necessidade de colocar vidas humanas ou bens materiais em risco. Diversas técnicas de IA trabalham bem com jogos e, muitas delas, foram desenvolvidas e testadas para esses jogos (Hong e Cho, 2004) (Clune, 2007).

O Go, jogo de tabuleiro milenar com existência datada há mais de 4000 anos pela tradição chinesa, tem a possibilidade de ser testado pelos conceitos de IA devido a alta complexidade dos movimentos e das estratégias utilizadas para se ter um bom jogo. Toda essa complexidade pode ser comparada com problemas do mundo real melhor que a maioria dos jogos (Blackman, 2003). A implementação de agentes jogadores se tornou uma área interessante e diversos jogos possuem tais jogadores com níveis tão bons ou melhores que ótimos jogadores humanos.

Go possui um grande fator de ramificação e a sua quantidade de estados supera a de outros jogos de tabuleiro como o xadrez e o gamão. O fato de possuir esse alto fator de ramificação, torna a avaliação de possíveis posições no tabuleiro uma tarefa

um tanto quando complexa (Campos e Langlois, 2003).

Assim, métodos inteligentes e técnicas de IA são interessantes de serem aplicadas na construção de agentes jogadores de Go. Alguns destes métodos já foram aplicados em agentes jogadores de Go, como a Busca em Árvore Monte-Carlo, utilizada no jogador de código-aberto Fuego (Enzenberger et. al., 2011) e Redes Neurais, utilizadas, por exemplo, no NeuroGo (Enzenberger, 2003). A ideia agora é a aplicação de Algoritmos Genéticos (AGs) na criação de um agente jogador de Go que seja competitivo.

De acordo com Russel (2004), um algoritmo genético pode ser definido como uma busca estocástica, onde os estados sucessores são gerados pela combinação de dois estados pais, no lugar de serem gerados de um único estado. Mitchell (1999) diz que algoritmos genéticos são métodos de busca que se aproximam da genética biológica, simulando evolução, na tentativa de encontrar uma solução para um dado problema.

A complexidade na resolução do jogo de Go possui semelhanças com problemas reais de caráter combinatórios, como o controle de tráfego urbano (Thorpe e Anderson, 1996) e o conhecimento utilizado por empresas na tomada de decisão (Bouzy e Cazenave, 1996), por exemplo. O primeiro problema se relaciona com o jogo de Go, visto que para tomar uma próxima decisão, a estratégia utilizada deve levar em conta a existência de outros agentes que interagem com a situação, o que, conseqüentemente, reflete na próxima decisão. Já o segundo, tem relação com o fato de que os três níveis utilizados na tomada de decisão no jogo, estratégico, tático e operacional, pode ser usado para modelar a tomada de decisão dentro de uma organização.

Com isso, este trabalho tem como objetivo o desenvolvimento de um agente jogador de Go, também descrito como jogador automático, utilizando para isso

técnicas de IA, como os já citados AGs. A ideia é que este agente consiga jogar de forma competitiva o Go, mostrando assim a eficácia do emprego de técnicas de IA, como o AG na resolução de problemas combinatórios.

METODOLOGIA

Para a obtenção de uma parte do conhecimento acerca do jogo de Go, foi necessário uma extensa pesquisa bibliográfica, que possibilitou o entendimento das regras e de algumas estratégias que devem ser adotadas no decorrer de uma partida.

A pesquisa bibliográfica, no entanto, não se fez suficiente na obtenção de todo o conhecimento sobre o jogo. Para entender melhor o funcionamento e as estratégias que um bom jogador de Go deve possuir, foi necessário praticar o jogo, empregar as técnicas e regras que foram obtidas através da pesquisa bibliográfica em um jogo real, ora contra o computador, ora contra pessoas.

Para praticar o jogo foi usada uma ferramenta online chamada Kiseido Go Server (KGS). O KGS é, segundo a Associação Portuguesa de Go, o servidor de Go mais intuitivo e popular fora do Oriente. Além do ambiente de jogo (tabuleiro virtual), o KGS possui também tutoriais para iniciantes, torneios e uma base de dados com todas as partidas disputadas por seus membros. O KGS encontra-se disponível em <https://www.gokgs.com/>.

Assim, o conhecimento adquirido sobre o jogo será de vital importância na implementação deste agente jogador, os resultados e o desempenho do jogador automático se darão em função deste conhecimento.

RESULTADOS E DISCUSSÃO

A aplicação de técnicas de IA no desenvolvimento de agentes jogadores traz excelentes resultados e, segundo Clune (2007), algumas técnicas são próprias para jogos. O objetivo é utilizar as técnicas

evolucionárias dos AGs no desenvolvimento do agente jogador.

Espera-se alcançar tal objetivo e constatar que a aplicação de AGs no desenvolvimento destes agentes jogadores os torna mais competitivos.

Uma limitação importante que deverá ser bem analisada nesta implementação é o tempo de execução do AG. Blackman (2003) cita que conjuntos treinamentos maiores, conjunto de testes maiores, populações maiores e mais gerações podem ajudar na melhora de desempenho do algoritmo, entretanto, influenciam na velocidade de execução do mesmo.

No Go, apesar das regras simples, as situações criadas no decorrer da partida são complexas. O objetivo do jogo é ganhar território, assim, na construção das jogadas são criados diversos cenários que devem ser analisados e avaliados para a decisão do próximo movimento. O agente jogador deverá lidar com todas estas situações utilizando as técnicas evolucionárias oriundas dos AGs.

Para avaliar o desempenho do agente jogador, uma interface gráfica será criada, permitindo assim que o agente desenvolvido jogue contra humanos. A interface deverá ser intuitiva e o algoritmo deverá possuir um tempo de resposta satisfatório.

CONCLUSÕES

Ao longo das pesquisas bibliográficas que deram origem ao início deste trabalho, foi possível verificar as vantagens trazidas pelo uso de técnicas de Inteligência Artificial no desenvolvimento de jogadores automáticos (agentes jogadores). Blackman (2003) diz que tais técnicas têm sido cada vez mais aplicadas em agentes que resolvem um determinado jogo, em especial aqueles jogos de duas pessoas e com informação perfeita.

Trabalhos como o de Enzenberger (2011) e Enzenberger (2012) mostram agentes altamente funcionais e competitivos, o que torna viável e interessante à

continuação de pesquisas e desenvolvimentos como este, que além de desafiadores, são também, como citado por Blackman (2003), relacionados com problemas complexos do cotidiano.

REFERÊNCIAS BIBLIOGRÁFICAS

BLACKMAN, Todd. **Genetic Algorithms for a Multiagent Approach to the Game of Go**. 2003, 265 f. tese – University of Kansas.

BOUZY, Bruno. CAZENAVE, Tristan. Shared concepts between complex systems and the game of Go. Université Pierre et Marie Curie 4, 1996. Disponível em: <http://www.math-info.univ-paris5.fr/~bouzy/publications/UES96.article.pdf> Acessado em 21/06/2014 às 11:48.

CAMPOS, Pedro. LANGLOIS, Thibault. Abalearn: Efficient Self-Play Learning of the game Abalone. In: **NEURAL NETWORKS AND SIGNAL PROCESSING GROUP**, 2003. Portugal: INESC-ID, 2003

CLUNE, J. Heuristic evaluation functions for general game playing. In: **Proc. of AAAI**, 1134–1139, 2007. Disponível em: <http://logic.stanford.edu/classes/cs227/2013/readings/cluneplayer.pdf> Acessado em 20/09/2014 às 14:35.

ENZENBERGER, Markus. **Evaluation in Go by a Neural Network Using Soft Segmentation**. University of Alberta, 2003. Disponível em: <http://webdocs.cs.ualberta.ca/~emarkus/neurogo/neurogo3.pdf> Acessado em 20/09/2014 às 16:04.

ENZENBERGER, Markus. MÜLLER, Marcus. ARNESON, Broderick. SEGAL, Richard. Fuego—An Open-Source Framework for Board Games and Go Engine Based on Monte Carlo Tree Search. In: **IEEE TRANSACTIONS ON COMPUTATIONAL INTELLIGENCE AND AI IN GAMES**, 2011.

HONG, J. H. CHO, S. B. Evolution of emergent behaviors for shooting game characters in robocode. In: **CONGRESS EVOLUTIONARY COMPUTATION**, 2004. Piscataway: CEC, 2004, v. 1, p. 634–638.

MITCHELL, Melanie. **9** The MIT press, 1ª ed., 1999.

RUSSEL, Stuart. NORVING, Peter. **Inteligência Artificial – Um Enfoque Moderno**. Tradução da segunda edição. Rio de Janeiro: Elsevier, 2004.

SHAH, Sanjay. SINGH, Dharm. SHAH, J. S. Using Genetic Algorithm to Solve Game of Go-Moku. *International Journal of Computers Applications*, Udaipur, n. 6, p. 28-31, 2012.

THORPE, Thomas L. ANDERSON, Charles W. **Traffic Light Control Using SARSA with Three State Representations.** Technical Reporte, IBM Corporation, 1996.

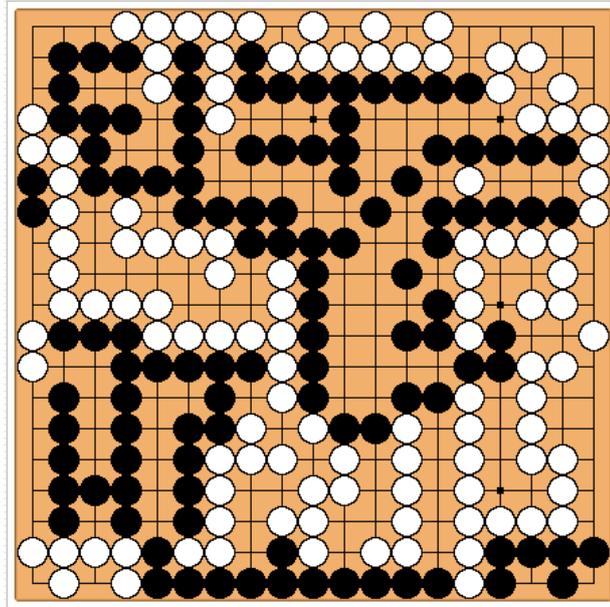


Figura 1: Tabuleiro padrão de Go (19x19). O Go é um jogo de tabuleiro de informação perfeita, jogado por duas pessoas. Assim como outros jogos orientais, é jogado com as pedras nas interseções das linhas do tabuleiro.