

Uma proposta para resolver o problema de loop no LSDraughts e VisionDraughts: criação de um jogador de final de jogo

Autor: Valquíria Aparecida Rosa Duarte¹,
Orientadora: Rita Maria da Silva Julia¹

¹Programa de Pós-Graduação em Ciência da Computação
Universidade Federal do Uberlândia (UFU)
Uberlândia – MG – Brasil

valquiriaduarte@gmail.com, rita@facom.ufu.br

Nível: Mestrado

Ano de ingresso no programa: 2008

Época esperada de conclusão: Julho / 2009

Etapa concluída: Defesa da proposta de dissertação Data da aprovação da proposta de dissertação: 19/19/2008

Resumo. *O objetivo deste trabalho é criar um jogador de final de jogo de Damas visando resolver, ou o mínimo, amenizar o problema de loop existente nos jogadores LSDraughts de [NETO 2007] e VisionDraughts de [CAIXETA 2008]. Este jogador será acoplado ao jogador de [CAIXETA 2008], o qual jogará até o meio da partida e, deste ponto em diante, o jogador de fim de jogo entra na partida e a conduz até o final. O segundo jogador, na verdade, será uma segunda rede neural que receberá como entrada para seu treinamento um AG selecionador de características análogo ao usado em [NETO 2007] e estados de tabuleiros característicos de final de jogo filtrados de uma base de dados com mais de 7 milhões de estados através de Redes de Kohonen [COSTA 1999]. A aprendizagem consiste em aproximar a rede neural MLP (MultiLayer Perceptron) através do método de aprendizagem por reforço TD(λ) e treinamento self-play com clonagem. O processo de aprendizagem deste jogador de final de jogo é similar ao processo utilizado pelos sistemas LSDraughts e VisionDraughts.*

Palavras-Chave. *Inteligência Artificial, Teoria dos jogos, Redes Neurais, Algoritmos Genéticos, Busca Minimax, Aprendizagem de Máquina, Aprendizagem por Reforço, Busca em Árvores de Jogos, Algoritmos de Clusterização, Redes Kohonen.*

1. Introdução e Motivação

Desde o início dos estudos na área de inteligência artificial (IA), tem havido grande interesse em fazer com que um computador aprenda por si próprio a jogar um jogo de habilidades, a ponto de se tornar competitivo contra especialistas humanos. Tal interesse se justifica pelo fato do domínio das técnicas necessárias para que um software jogue um determinado jogo serem as mesmas utilizadas nas principais áreas da aprendizagem automática.

Em jogos de tabuleiro é normal encontrar situações onde os movimentos das peças se tornam repetitivos e o estado do jogo permanece inalterado, o que faz com que um estado de vitória não seja alcançado por nenhum jogador, caracterizando um empate. Essa situação de indeterminação do resultado do jogo devido a movimentos repetitivos caracteriza um *loop* nos jogos de tabuleiro, mais especificamente, no domínio de jogo de Damas. Um *loop* ocorre quando o agente, mesmo em vantagem, não consegue pressionar o adversário e alcançar a vitória. Ao invés disso, o agente começa uma sequência repetitiva de movimentos (*loop*) que passa a alternar entre posições inúteis do tabuleiro.

O agente jogador de Damas que será usado neste trabalho como base para os estudos de problema de *loop* será o jogador LSDraughts [NETO et al. 2008] e o VisionDraughts [CAIXETA 2008], ambos inspirados no jogador original de Mark Lynch, o NeuroDraughts [LYNCH 2007], onde o *loop* também é relatado como um dos problemas essenciais.

A grande incidência de empates envolvendo esses jogadores, mesmo em situações em que eles têm grande vantagem de peças sobre o adversário no momento do empate, é o fator principal que motiva a decisão de investigar e propor métodos de resolução para tal problema. Para tanto, buscaram-se informações, tanto em jogos de tabuleiro, quanto em outras áreas, com o objetivo de extrair idéias práticas ou técnicas que pudessem auxiliar na resolução ou amenização do *loop*.

Em outras palavras, este trabalho visa resolver ou, no mínimo, amenizar o problema de *loop* existentes na junção dos jogadores LSDraughts e VisionDraughts [Caixeta and da Silva Julia 2008], além de melhorar a eficiência dos mesmos.

2. Trabalhos Relacionados

Serão apresentados nesta seção alguns jogos de tabuleiro onde a presença de *loop* interfere no resultado final do jogo. Os jogos analisados aqui são de damas e de GO. Entre os agentes jogadores de damas, temos o campeão mundial de damas Chinook, o NeuroDraughts, o LSDraughts, o VisionDraughts e o Anaconda, sendo que os quatro últimos são jogadores que não têm supervisão humana no seu processo de aprendizagem. O último jogo tratado nesta seção é o GO, o qual apresenta uma técnica capaz de resolver o problema de *loop* neste jogo.

2.1. NeuroDraughts, LSDraughts e VisionDraughts

Mark Lynch [LYNCH 2007] desenvolveu seu jogador de damas, NeuroDraughts, utilizando o mínimo de intervenção humana possível. Seu objetivo foi criar um agente que pudesse aprender a jogar damas através de partidas contra si próprio, dadas apenas as regras do jogo e um pequeno conjunto de características específicas do jogo de damas. Apesar da proposta de pouca intervenção humana, o NeuroDraughts ainda utiliza um subconjunto das características extraídas do jogador de Samuel [SAMUEL 1959, SAMUEL 1967], as quais representam algumas especificidades de damas definidas por mestres deste jogo. Essas características foram selecionadas manualmente por Lynch.

Uma extensão do NeuroDraughts é o LSDraughts [NETO 2007], um dos nossos principais objeto de estudos, o qual é um sistema gerador de agentes jogadores de Damas em que os

jogadores correspondem a redes neurais treinadas pelo método de aprendizagem por reforço e das diferenças temporais, aliado com a busca minimax e técnica de treinamento por self-play com clonagem. As características usadas por este jogador são selecionadas, automaticamente através de um algoritmo genético.

Outra extensão do NeuroDraughts é o VisionDraughts de [CAIXETA 2008]. O VisionDraughts, acrescenta dois módulos à arquitetura do NeuroDraughts: um módulo de busca eficiente em árvores de jogos com alfa-beta, aprofundamento iterativo e tabelas de transposição, fornecendo ao agente jogador maior capacidade de analisar jogadas futuras (estados do tabuleiro mais distantes do estado corrente) e um módulo para acessar bases de dados de fim de jogo, permitindo obter informações perfeitas para combinações de oito, ou menos, peças no tabuleiro. Neste jogador o tempo de execução para o treinamento do jogador foi reduzido em 95% comparado ao LSDraughts e seu desempenho significativamente melhorado.

Os resultados experimentais obtidos com o VisionDraughts mostram que as ocorrências do problema do *loop* foram consideravelmente reduzidas com o uso das bases de dados e da busca em profundidade variável com aprofundamento iterativo.

2.2. Chinook

O mais famoso agente jogador de damas e atual campeão mundial chama-se Chinook [SCHAEFFER 1996]. Chinook é produto de esforços de pesquisas em estratégia de jogos iniciados em 1989. O projeto tinha dois objetivos que já foram alcançados: desenvolver um programa capaz de bater o campeão mundial humano e resolver a teoria do jogo de damas.

O desempenho deste jogador depende de quatro características fundamentais, que são:

1. Busca - eficiente algoritmo paralelo baseado em poda alfa-beta.
2. Função de avaliação - ajustada manualmente e é capaz de decidir o quão boa é uma dada posição.
3. Bases de dados de fim de jogo - informação perfeita de qual fim de jogo é vitória, derrota ou empate.
4. Opening book - base de dados com diretrizes de início de jogo.

A resistência do Chinook vem de profunda busca, boas funções de avaliação e base de dados de fim de jogo que contém informações perfeitas de todas as posições com 10 peças ou menos. O Chinook depois de ter terminado sua base de dados não perde para nenhum jogador, sendo que no máximo outros jogadores podem conseguir empatar o jogo.

A grande diferença entre o Chinook e os jogadores usados para estudo neste trabalho é que o Chinook é dotado de supervisão humana em todas as fases do jogo que vai desde o banco de dados boas jogadas de início de partida, ajustes manuais nas funções de avaliação e base de dados com todas as possíveis jogadas a partir de 10 peças no tabuleiro, o que justifica a eficiência deste jogador. E a intenção deste trabalho é conseguir um bom jogador de Damas, porém com o mínimo de intervenção humana possível. O Chinook será usado neste trabalho durante a fase de teste do jogador, com o objetivo de verificar o quão bom é o nosso jogador.

2.3. Anaconda

Outro bem sucedido programa jogador de damas é o Anaconda [CHELLAPILLA and FOGEL 2000]. Seus criadores exploraram o potencial de aprender jogar damas sem depender da interferência humana, na forma de características do jogo, que acreditam ser importantes para se jogar bem. Foi utilizada uma população de redes neurais

na qual cada indivíduo serve como uma função de avaliação, para descrever a qualidade da posição corrente do jogo.

Em vez de pesquisar em tabelas, grandes bases de dados com jogadas perfeitas de fim de jogo, o Anaconda considera a possibilidade de redes neurais aprenderem a jogar damas sem esses conhecimentos, e consequentemente com o mínimo de intervenção humana possível. Segundo Fogel [CHELLAPILLA and FOGEL 2000], os esforços despendidos com técnicas de aprendizagem automática devem minimizar o montante de intervenção humana no processo e aprendizagem do jogador. Ainda segundo David Fogel o problema de *loop* no Anaconda foi desconsiderado e foi deixado por conta das regras do jogo decidir o resultado final do mesmo.

2.4. Go

Go também é um jogo de tabuleiro, porém com várias características diferentes do jogo de damas. A teoria dos jogos combinatória existente neste jogo fornece ferramentas as quais são mais eficientes para localizar jogadas que uma busca minimax completa pelo tabuleiro. Uma dessas ferramentas, termografia (thermography em inglês), foi recentemente generalizada no Go para jogos locais, ou sub-jogos, onde contenham *loops*.

Para resolver o problema de *loop* existente no Go, Elwys Berlekamp [BERLEKAMP 1996] desenvolveu uma teoria de temperatura, Generalized Thermography, que conta com varias suposições de como atacar e resolver o problema de *loop* no jogo. Com o objetivo de resolver o problema de *loop* no jogo de damas, verificou-se a possibilidade de adaptar tal técnica ao ambiente de jogo de damas, porém sem sucesso.

De acordo com Muller, Thermography é uma técnica para jogos de soma zero, onde para cada movimento existe uma pergunta se você deve jogar aquele jogo, ou um sub-jogo diferente. No jogo de GO a técnica Generalized thermography direciona o jogo para o sub-jogo onde tem *loop* e faz as suposições necessárias para resolver o *loop*. No caso de jogo de damas, isso não é possível, pois existe somente um jogo, e não é possível subdividir esse jogo em sub-jogos diferentes para serem tratados pela técnica.

3. Metodologia e Estado da Pesquisa

Diante das características apresentadas pelos jogadores estudados, atacar-se-á dois pontos fracos existentes nos mesmos:

1. Ordenação da árvore de busca: ao invés de se ordenar somente a raiz a árvore de busca, como é feito no VisionDraughts, pretende-se ordenar parcialmente toda a árvore, de modo a agilizar o processo de busca e, consequentemente, aumentar o nível da profundidade de busca (*lookahead*) e a possibilidade de encontrar bons movimentos na mesma.
2. Criação de um jogador de fim de jogo: criar uma segunda rede neural a qual será acoplada ao jogador VisionDraughts, que começará a atuar no jogo quando o mesmo se encontrar com apenas 10 peças sobre o tabuleiro. Esta segunda rede será treinada de modo análogo ao treinamento do LSDraughts, inclusive no que diz respeito ao uso de características de jogo de Damas.

Um dos objetivos da utilização do aprofundamento iterativo no VisionDraughts foi atacar o problema do *loop*. Para isso, a ordenação parcial da árvore de busca, utilizando o melhor movimento descoberto nas iterações anteriores (iteraões já concluídas) e possivelmente armazenado na tabela de transposição, foi realizada apenas na raiz da árvore. Assim, utilizar o procedimento iterativo com tabela de transposição para ordenar, parcialmente, todos os estados da árvore anteriormente pesquisados (e não apenas o estado raiz) é uma boa opção na tentativa

de reduzir o tempo de busca e consequentemente aumentar a profundidade de busca e encontrar um maior número de possíveis melhores movimentos.

A ordenação total da árvore de busca consiste em colocar nos ramos mais a esquerda da árvore os filhos de melhor avaliação (avaliação retornada pela rede neural), fazendo com que os possíveis melhores movimentos sejam avaliados primeiro e um melhor resultado possa ser alcançado mais rapidamente. Este melhor resultado será alcançado pois a ordenação dos ramos da árvore possibilitará o aumento do número de podas alfa-beta que acontecerão no decorrer da construção da mesma.

E quanto ao jogador de fim de jogo, como este será criado?

Uma versão multi-agente será utilizada de forma que cada um dos agentes se especialize em uma das fases do jogo, sendo que o agente jogador VisionDraughts atuará do início até o meio do jogo e o segundo agente, o qual será criado neste trabalho, atuará do meio até o final do jogo. Serão utilizadas duas redes neurais artificiais (RNA) sendo cada uma responsável por uma fase do jogo.

O treinamento das redes neurais será idem ao treinamento feito no trabalho de [CAIXETA 2008], ou seja, o treinamento será feito através de Aprendizagem por reforço TD (λ) com *self-play* com clonagem [RUSSELL and NORVIG 2004] e sem supervisão humana. Uma diferença que pode ser notada entre o treinamento das duas redes é que a segunda rede contará com um Algoritmo Genético (AG) o qual selecionará as características de jogo de Damas [SAMUEL 1959] que deverão estar presentes nos indivíduos durante seu treinamento, e com uma base de dados fornecedora de estados iniciais de jogo.

Os estados iniciais de jogo do agente jogador de final de jogo (segunda rede neural) é diferente do estado inicial usado no VisionDraughts, visto que início de jogo para o jogador de final de jogo é tabuleiro com 10 peças. A base de dados que conterá esses estados de tabuleiros com 10 peças será gerada através do jogador VisionDraughts.

A escolha em caracterizar 10 peças como final de jogo foi feita observando os tabuleiros em vários estágios da partida, e no estágio de 10 peças sobre o tabuleiro ainda não existe um jogador tendencioso a ganhar o jogo diferentemente do que pôde ser observado num tabuleiro de 8 peças.

Devido ao grande número de estados gerados pelo VisionDraughts e o tempo dispendioso gasto no treinamento do jogador, não será possível treiná-lo em todos os estados de tabuleiro gerados, sendo que um filtro deverá ser aplicado à base de dados de modo a deixar na base estados mais prováveis de ocorrer durante uma partida. A estratégia usada para filtrar os dados da base foi agrupar os estados em Clusters [JAIN and DUBES 1988] e posteriormente classificá-los por incidência na base.

O algoritmo usado para a clusterização foi o Algoritmo de Kohonen (SOM), também conhecido como Redes de Kohonen [COSTA 1999], o qual agrupa os dados em grupos baseados na similaridade entre eles. A similaridade é fornecida por uma função de distância, chamada distância euclidiana, definida entre pares de dados.

O AG existente no jogador de final de jogo é responsável por selecionar um conjunto mínimo de características e fornece-los como entrada da rede neural juntamente com os estados selecionados da base de dados de estados de fim de jogo. A partir do momento que já estiver disponível à rede os estados de tabuleiros e o conjunto mínimo de características, a fase de treinamento poderá ser iniciada. Vale lembrar que o treinamento será feito por de aprendizagem por reforço TD(λ) e *self-play* com clonagem.

Diferentemente do Chinook, neste jogador a intenção é usar o mínimo de intervenção humana possível, de modo que o jogador possa aprender por si próprio, através de retropropagação de erro, a se comportar no jogo e conseguir bons resultados.

Acredita-se que ao final da criação da segunda rede ter-se-á um conjunto mínimo de características necessárias para representar tabuleiro de Damas em final de jogo e uma rede neural apta a jogar em tabuleiros a partir de 10 peças.

Logo, a metodologia proposta para combater o problema de *loop* dos jogadores é implementar os dois mecanismos citados acima, e a partir daí comparar, através de treinamento e teste, a eficiência entre os jogadores já existentes e o novo jogador obtido.

4. Cronograma do Trabalho até a Defesa

A figura abaixo mostra o detalhamento do cronograma de trabalho a ser desenvolvido.

Atividade/Mês	Ago 2008	Set 2008	Out 2008	Nov 2008	Dez 2008	Jan 2009	Fev 2009	Mar 2009	Abr 2009	Mai 2009	Jun 2009	Jul 2009
Pesquisas Bibliográficas	X	X	X	X	X	X	X					
Estudo do jogador NeuroDraughts	X	X										
Estudo do jogador LsDraughts	X	X	X	X	X	X	X	X				
Estudo do jogador VisionDraughts	X	X	X	X	X	X	X	X				
Ordenação da árvore de busca	X	X										
Testes ordenação		X										
Junção do LsDraughts e VisionDraughts	X	X	X	X	X							
Testes junção				X	X	X						
Implementação da segunda RNA			X	X	X	X	X	X				
Testes com a rede							X	X	X			
Análise de resultados obtidos em cada implementação									X	X		
Comparação de resultados									X	X		
Submissão de artigos				X	X	X	X					
Redação da dissertação				X	X	X	X	X	X	X	X	
Apresentação/Defesa												X

References

- BERLEKAMP, E. (1996). The economist's view of combinatorial games. *Proceedings of MSRI Workshop on Combinatorial Games*.
- CAIXETA, G. S. (2008). Visiondraughts -um sistema de aprendizagem de jogos de damas baseado em redes neurais, diferenças temporais, algoritmos eficientes de busca em Árvores e informações perfeitas contidas em bases de dados. Master's thesis, Universidade Federal de Uberlândia.
- Caixeta, G. S. and da Silva Julia, R. M. (2008). A draughts learning system based on neural networks and temporal differences: The impact of an efficient tree-search algorithm. *The 19th Brazilian Symposium on Artificial Intelligence, SBIA 2008*.
- CHELLAPILLA, K. and FOGEL, D. B. (2000). Anaconda defeats hoyle 6-0: A case study competing an evolved checkers program against commercially available software. *Proceedings of the 2000 Congress on Evolutionary Computation CEC00, La Jolla Marriott Hotel La Jolla, California, USA*, page 857.
- COSTA, J. A. F. (1999). Classificação automática e análise de dados por redes neurais auto-organizáveis. Master's thesis, Universidade Estadual de Campinas.

- JAIN, A. K. and DUBES, R. (1988). *Algorithms for Clustering Data*. Prentice-Hall.
- LYNCH, M. (2007). Neurodraughts - an application of temporal difference learning to draughts. Master's thesis, Dept. of CSIS, University of Limerick, Ireland.
- NETO, H. C. (2007). Ls-draughts - um sistema de aprendizagem de jogos de damas baseado em algoritmos genéticos, redes neurais e diferenças temporais. Master's thesis, Universidade Federal de Uberlândia.
- NETO, H. C., JULIA, R. M. S., and Caixeta, G. (2008). *LS-Draughts- A Draughts Learning System Based on Genetic Algorithms, Neural Network and Temporal Differences*, pages 73–82. Kirchengasse: I-Tech Educational and Publishing KG,2008.
- RUSSELL, S. and NORVIG, P. (2004). *Inteligência Artificial - Uma Abordagem Moderna*. Editora Campus.
- SAMUEL, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*.
- SAMUEL, A. L. (1967). Some studies in machine learning using the game of checkers ii - recent progress. *IBM Journal of Research and Development*.
- SCHAEFFER, J. e. a. (1996). Chinook - the world man-machine checkers champion. *AI Magazine*, 17(1):21–29.