Proposta de solução do Problema de Dominação de Rainhas Utilizando ILS*

Maria Edoarda Vallim Fonseca
Institute of Computing – UFF
Niterói, Brazil
medoarda@id.uff.br

Thales Athayde Santos
Institute of Computing – UFF
Niterói, Brazil
thalesathaydesantos@id.uff.br

ABSTRACT

In this paper, we propose a solution to the Dominating Queens Problem using Iterative Local Search and compare our results with a previous solution using Genetic Algorithm.

KEYWORDS

Dominating Queen Problem, ILS, Metaheuristic

ACM Reference Format:
Maria Edoarda Vallim Fonseca and Thales Athayde Santos. 2018. Proposta de solução do Problema de Dominação de Rainhas Utilizando ILS. In *Proceedings of Trabalho Final da disciplina de Pesquisa Operacional da UFF, 2018.2 (Pesquisa Operacional).* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUÇÃO

O problema de dominação de rainhas é muito bem conhecido dentre os problemas de xadrez. Nele, dado um tabuleiro *nxn*, temos n quadrados dispostos nas linhas e n quadrados nas colunas. Quando uma rainha Q é disposta no tabuleiro, ela domina a linha, a coluna, e as diagonais que passam pela sua posição. O objetivo do problema é descobrir a disposição da menor quantidade de rainhas possível de forma à dominar todo o tabuleiro.

Muitos pesquisadores focaram em descobrir os limites superior e inferior do problema desde a década de 70, e o número mínimo possível de rainhas para solucionar o problema ($\gamma(\mathbf{Qn})$) foi calculado para diversos tamanhos de tabuleiro (vide tabela 1).

Algoritmos evolucionários provaram ter sucesso para resolver e otimizar uma grande variedade de problemas complexos, incluindo problemas combinatórios, como o estudado aqui, em um tempo computacional razoavelmente aceitável. [2]

Local Search, ou Busca Local, é um método heurístico para resolver problemas computacionalmente difíceis. Busca local pode ser usada em problemas que possam ser formulados como achar a solução maximizando um critério entre várias soluções possíveis. Algoritmos de busca local movem de solução à solução no espaço de soluções possíveis aplicando mudanças locais, até uma solução dita ótima ser encontrada. [3]

Um dos problemas da Busca Local é que ela pode ficar presa em um mínimo local, sem conseguir melhorar seu resultado. Para contornar esse problema, utiliza-se *Iterative Local Search*, ou Busca Local Iterativa. Essa modificação consiste em iterar sobre chamadas da busca local, cada vez começando de um ponto diferente do conjunto de solução perturbando o mínimo local atual de modo que faça a solução chegar em outro ótimo local. Esta perturbação não pode ser muito forte nem muito fraca, pois corre o risco dela acabar encontrando o mesmo mínimo local ou servir como uma inicialização aleatória. [4]

Neste artigo, propomos uma solução baseada em *Iterative Local Search* para o Problema de Dominação de Rainhas. Nossos resultados serão comparados diretamente com o método descrito em [1], que utiliza Algoritmo Genético e é o mais comumente encontrado para solucionar este problema. Depois disso, iremos debater os resultados alcançados.

2 MOTIVAÇÃO

Decidimos escolher o *Dominating Queens Problem* por ter sido um dos temas abordados por um dos membros do grupo durante as apresentações de trabalho da disciplina, o que nos deu um certo grau de familiaridade com o assunto. Pesquisando mais à fundo, vimos que as soluções mais comuns para a solução do Problema de Dominação de Rainhas eram com *Backtracking* e Algoritmo Genético. Além disso, de acordo com [1], existem muitos artigos procurando os limites superior e inferior do problema, mas não existe muito esforço de pesquisa na busca de soluções práticas para o problema.

Visto essa situação, decidimos propor uma solução baseada em *Iterative Local Search* para o problema e comparar os resultados com uma solução em Algoritmo Guloso.

3 TRABALHOS RELACIONADOS

Figure 1 mostra a dominação de uma rainha em um tabuleiro *8x8.*

Figure 2 shows a sdfsdfwererWboat.

4 METHODOLOGY

O *Local Search* foi implementado com 1 de distância pela nossa implementação estar utilizando uma matriz e não um vetor contendo todas as posições do tabuleiro. Embora usar um *Local Search* com distâncias maiores gere resultados melhores, isso teria a consequência negativa de aumentar exponencialmente o tempo computacional da execução do algoritmo.

Pseudocódigo do ILS

*Produces the permission block, and copyright information

Pesquisa Operacional, 2018

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM.
https://doi.org/10.1145/nnnnnnn.nnnnnnn

Tabela 1: Número mínimo de rainhas para dominação total $\gamma(\mathbf{Qn})$ de um tabuleiro de tamanho *n*.

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
$\gamma(\mathbf{Qn})$	1	1	1	3	3	4	4	5	5	5	5	7	7	≤8	≤9	≤9	9	9

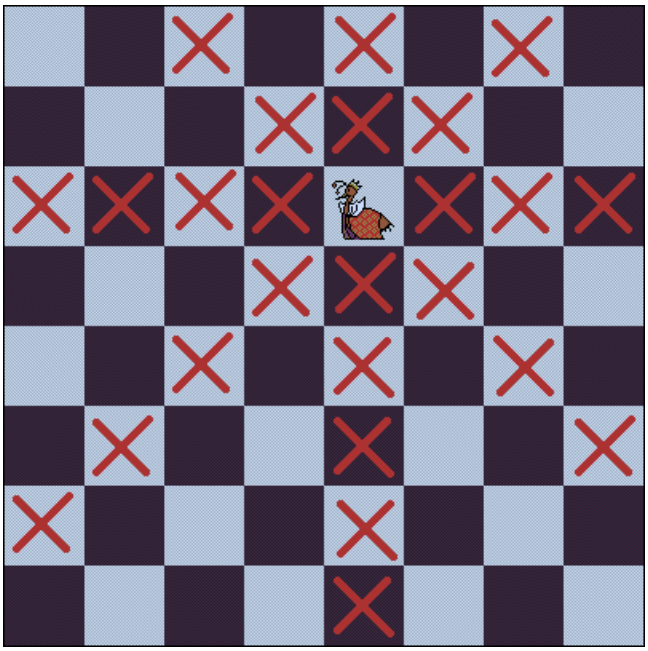

Figura 1: Exemplo da linha de dominação de uma rainha em um tabuleiro de xadrez *8x8*

Figura 2: Exemplo de um tabuleiro de tamanho *8x8* sendo completamente dominado por quatro rainhas

Data: rainhas, tabuleiro
Result: how to write algorithm with LaTeX2e
initialization;
listaMovimentos = [0..7];
melhorResultado;
while *rainhas* **do**
  embaralha(rainhas);
  if *understand* **then**
    go to next section;
    current section becomes this one;
  else
    go back to the beginning of current section;
  end
end

Algorithm 1: Pseudocódigo do algoritmo de ILS utilizado

Data: rainhas, tabuleiro
Result: how to write algorithm with LaTeX2e
initialization;
listaMovimentos = [0..7];
melhorResultado;
while *rainhas* **do**
  embaralha(rainhas);
  if *understand* **then**
    go to next section;
    current section becomes this one;
  else
    go back to the beginning of current section;
  end
end

Algorithm 2: Pseudocódigo do Algoritmo Genético utilizado

5 RESULTADOS

Nossos testes foram rodados em uma máquina Intel Core i5-7200U com 8GB de RAM, usando o sistema operacional Manjaro Linux com o pacote gráfico KDE Plasma. A linguagem de programação utilizada foi Python 3.7.1.

6 CONCLUSÃO

Text...

REFERÊNCIAS

[1] Saad Alharbi and Ibrahim Venkat. 2017. A Genetic Algorithm Based Approach for Solving the Minimum Dominating Set of Queens Problem. *Journal of Optimization* 2017 (2017).
[2] Benjamin Doerr, Anton Eremeev, Frank Neumann, Madeleine Theile, and Christian Thyssen. 2011. Evolutionary algorithms and dynamic programming. *Theoretical computer science* 412, 43 (2011), 6020–6035.
[3] Holger H Hoos and Thomas Stützle. 2004. *Stochastic local search: Foundations and applications.* Elsevier.

[4] Helena R Lourenço, Olivier C Martin, and Thomas Stützle. 2010. Iterated local search: Framework and applications. In *Handbook of metaheuristics*. Springer, 363–397.