

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

## *Proposed solution to the Minimum Dominating Set of Queens Problem using ILS*

Maria Edoarda Vallim Fonseca

Thales Athayde Santos

2018, v-1.0.0

Figure 1 shows a sdfsfwererWboat.

Um problema semelhante, proposto em 1850, conhecido como problema das n-rainhas teve muitos esforços focados nele para sua solução. O problema das n-rainhas é descrito como: dado um tabuleiro  $n \times n$ , qual seria a disposição das rainhas de modo que nenhuma rainha consiga atacar a outra. Houve muita pesquisa em torno deste problema, e suas soluções utilizam desde teoria matemática até teoria dos grafos. O estudo desse tipo

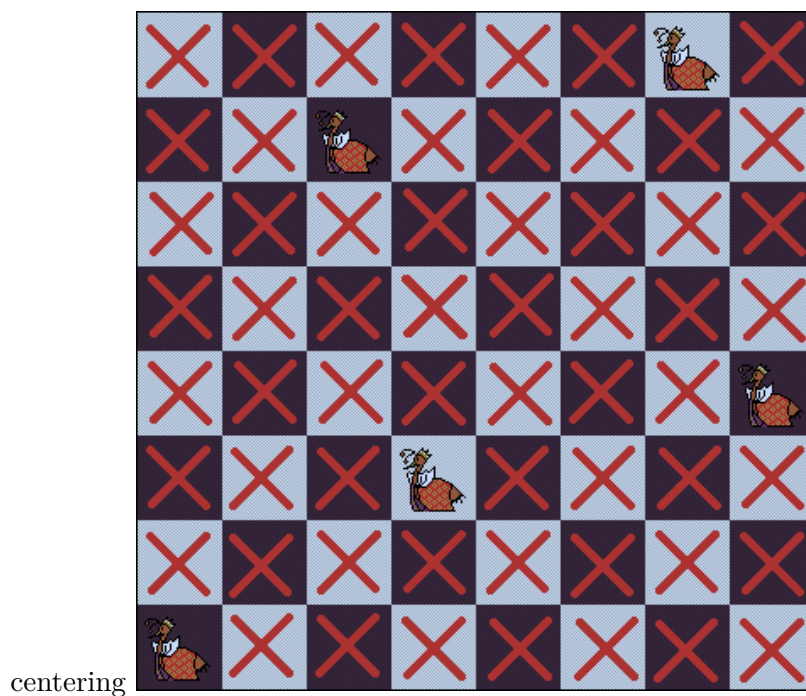


Figura 1 – Exemplo de um tabuleiro de tamanho  $8 \times 8$  sendo completamente dominado por quatro rainhas

de problema pode beneficiar várias áreas como controle de tráfego, prevenção de *deadlocks* e armazenamento de memória paralela. (BELL; STEVENS, 2009)

## 1 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

```

Data: rainhas, best, maxIterações
Result: best - melhor solução
rainhas  $\leftarrow$  randomStart(rainhas);
rainhas  $\leftarrow$  LocalSearch(rainhas);
best  $\leftarrow$  rainhas;
repeat
    rainhas  $\leftarrow$  perturbation(rainhas);
    rainhas  $\leftarrow$  LocalSearch(rainhas);
    if fitness(rainhas) > fitness(best) then
        | best  $\leftarrow$  rainhas;
    end
until maxIterações OU fitness(best) = 1;

```

**Algorithm 1:** Pseudocódigo do algoritmo de ILS utilizado

```

Data: rainhas, tabuleiro
Result: how to write algorithm with LATEX2ε
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

## 2 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.

### 3 Conclusão

Text...

### Referências

BELL, J.; STEVENS, B. A survey of known results and research areas for n-queens. *Discrete Mathematics*, Elsevier, v. 309, n. 1, p. 1–31, 2009. Citado na página [2](#).