

# Problema de Alocação de Horários

Arthur Alexander Martins Teodoro<sup>1</sup>, Wesley Henrique Batista Nunes<sup>1</sup>

<sup>1</sup>Instituto Federal de Minas Gerais - Campus Formiga

arthurmteodoro@gmail.com, wesleyhbn@hotmail.com

## 1. Introdução

O problema de alocação de programação de horários consiste em uma intensa área de atividade. A criação de tabelas de jogos e organização de escolas são grandes exemplos da área, porém pode ser aplicada em diversas áreas.

Porém, este se encaixa na categoria *NP-Difícil*, ou seja, utilizando algoritmos determinísticos, fazendo com que a execução demore muito, fazendo com que outros métodos devam ser utilizados.

## 2. O Problema

O problema proposto era a geração de um quadro de horários para exames finais. Neste caso, existe uma quantidade  $t$  de turmas nas quais ocorreram exames. Além disso, existe  $s$  salas disponíveis para a realização do exame em  $h$  horários. A quantidade de dias possíveis para ocorrer os exames foi de 5.

O problema também leva em relação currículo. Uma currículo é um conjunto de disciplinas que são interligadas. No contexto escolar, uma currículo pode ser todas as disciplinas de uma turma.

O sistema possui um total de 3 restrições, sendo elas: (a) Todas as disciplinas devem dar a quantidade de exames pré-determinados; (b) A quantidade de exames que iram ocorrer no mesmo horário não pode exceder a quantidade de salas disponíveis e (c) Disciplinas pertencente à mesma currículo não pode possuir exame final no mesmo horário.

## 3. Modelagem Proposta

Nesta seção será descrito as decisões de projeto para a solução do problema, usando como base em [Martins 2010].

### 3.1. Representação e geração de vizinhos

Cada um dos vizinhos possui cinco matrizes de dimensões  $D_{tXh}$ , cada uma representando um dia. Cada posição da matriz representa se existirá exame nesta turma neste horário.

Para gerar um vizinho, são feitas cinco permutações, cada uma escolhendo um dia, turma e horário randomicamente. O cálculo da função objetivo leva em conta se a quantidade de exames for respeitadas, se a quantidade de salas estiver válidas e caso não exista problemas na alocação por currículos.

### 3.2. Esquema de Resfriamento

O esquema de resfriamento usado é decremento geométrico, cuja função é  $Temp = Temp * 0.95$ . A escolha do valor foi feita com base no fato de que como o problema não possui grande desempenho utilizando a meta-heurística *Simulated Annealing*.

### 3.3. Reaquecimento

Como o problema se encaixa na categoria de busca, o algoritmo deve encontrar uma solução, e ao contrário de outros problemas comuns, somente a solução ótima deve ser aceita. Dada a esta característica do problema, é realizada o reaquecimento do *Simulated Annealing*. Isso permite que o algoritmo possa andar mais pelo espaço de busca, aumentando assim a chance de encontrar a solução.

### 3.4. Simulated Annealing

Foi criado duas versões da meta-heurística *Simulated Annealing*. A primeira versão foi a versão cânone, na qual a temperatura é decrementada sempre. Já na versão modificada pelo grupo, a temperatura é decrementada somente quando não existe troca de melhor vizinho.

## 4. Testes Realizados

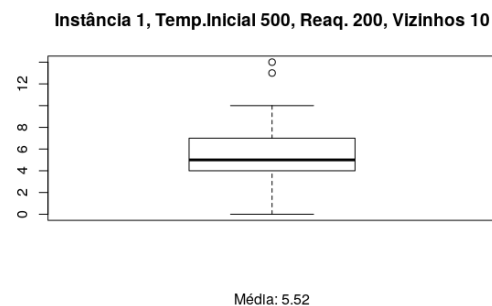
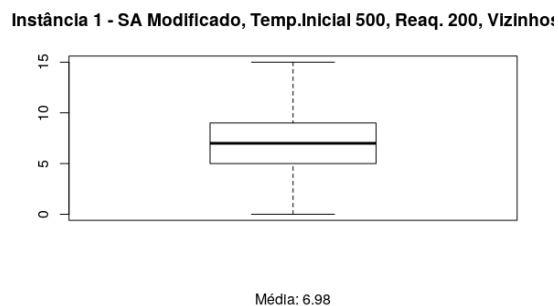
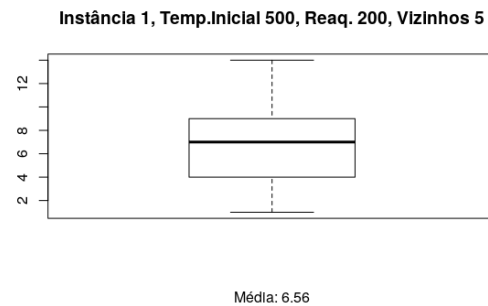
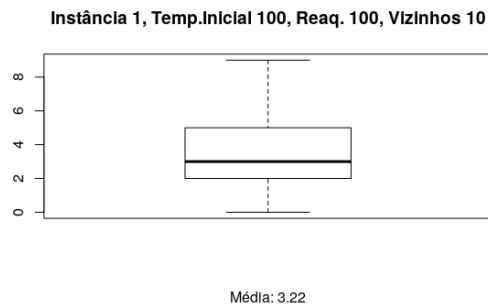
Foi usada três instâncias de testes. As instâncias *curriculas1.dat* e *curriculas2.dat* possuem a mesma distribuição de curriculas (17 disciplinas em 4 currículas), alterando entre as mesmas somente a quantidade de salas e horários disponíveis. Já a instância de teste *curriculas3.dat* possui 25 disciplinas distribuídas dentre 5 currículas.

Foram alterados parâmetros como temperatura inicial e quantidade de reaquecimento para verificar qual mostra melhor resultado. Além disso, existe os mesmos testes para a modificação proposta para o *Simulated Annealing*.

## 5. Resultados

Foi realizada uma bateria de testes com todas as instâncias. Porém, a instância *curriculas2.dat* se mostrou muito instável, por isso, foi retirada da análise final. Porém esta possuiu este comportamento uma vez que a a quantidade de salas e horários é mais restrita.



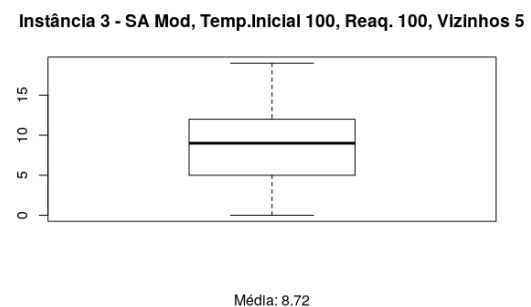
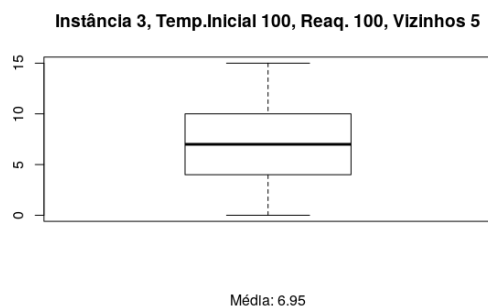


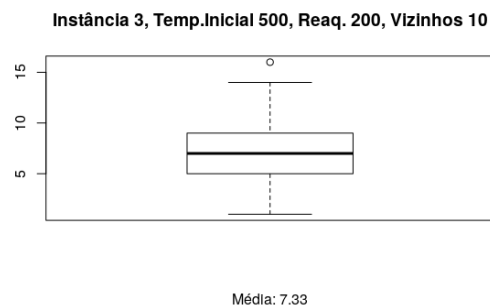
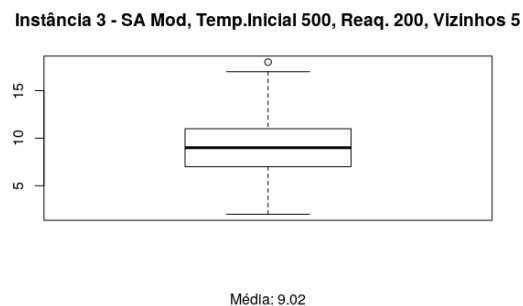
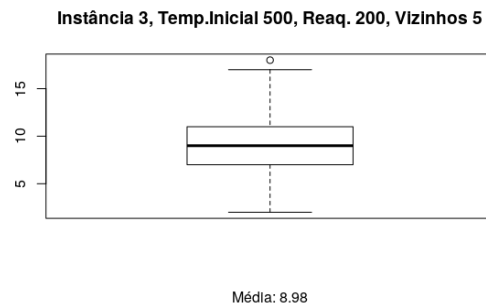
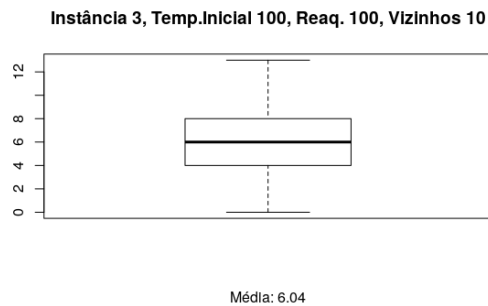
Visualizando os gráficos resultantes da bateria de testes com a instância *curriculas1.dat* foi visto que, para a resolução do problema, uma quantidade maior de vizinhos influencia mais positivamente que o aumento da temperatura inicial e quantidade de reaquecimentos.

Para o instância *curriculas3.dat* também se mostrou mais proveitoso o uso de um maior número de vizinhos que de reaquecimentos ou temperatura inicial.

Tal fato se deve que, com mais vizinhos, um número maior de soluções são verificados na vizinhança, aumentando a chance de encontrar uma boa solução. O aumento de temperatura faz com que exista muita aceitação de vizinhos não tão bons, uma vez que a "agitação" se encontra alta por muito tempo.

Ambas as instâncias apresentaram um valor médio melhor fazendo uso do *Simulated Annealing* cânone do que a melhoria proposta pelo grupo.





## 6. Conclusão

Este projeto trouxe ao grupo, além de um reforço no método *Simulated Annealing* apresentado em sala, um grande conhecimento sobre a complexidade existente em problemas reais. Além disso, despertou na equipe um interesse por métodos não determinísticos. Trouxe também um aprendizado sobre métodos de validação de um algoritmo.

## Referências

Martins, J. P. (2010). O problema do agendamento semanal de aulas. <http://www.inf.ufg.br/mestrado/sites/www.inf.ufg.br.mestrado/files/uploads/Dissertacoes/JeanPaulo.pdf>. Acesso em 03 de Maio de 2017.