LI Simpósio Brasileiro de Pesquisa Operacional
Limeira–SP, 02 a 06 de setembro de 2019

ABORDAGEM EXATA E HEURÍSTICAS PARA O PROBLEMA DE PLANEJAMENTO DE ORDENS DE MANUTENÇÃO DE LONGO PRAZO: UM ESTUDO DE CASO INDUSTRIAL DE LARGA ESCALA

Roberto Dias Aquino
Programa de Pós-graduação em Ciência da Computação
Universidade Federal de Ouro Preto
Campus Universitário, Morro do Cruzeiro, CEP 35400-000, Ouro Preto - MG
roberto.aquino@aluno.ufop.edu.br

Dissertação de Mestrado
Defesa: novembro de 2018
Orientador: Marcone Jamilson Freitas Souza
Coorientador: Jonatas Batista Costa das Chagas

RESUMO

Este trabalho tem seu foco em um problema real de planejamento de manutenção de longo prazo para uma planta de beneficiamento de minério de ferro no Brasil. Este é um problema complexo de programação de ordens de manutenção preventiva, para o qual é necessário atribuir ordens de manutenção preventiva para as equipes de trabalho disponíveis em um horizonte de 52 semanas. Para resolvê-lo foi desenvolvido um modelo de programação linear inteira mista (*Mixed Integer Linear Programming* - MILP), bem como algoritmos meta-heurísticos baseados nos métodos *Simulated Annealing* (SA), *Variable Neighborhood Search* (VNS) e *Biased Random-Key Genetic Algorithm* (BRKGA). O modelo exato serviu para validar os resultados dos algoritmos heurísticos aplicados a instâncias de dimensões menores. Os algoritmos meta-heurísticos foram capazes de produzir soluções melhores do que aquelas empregadas pela empresa, e em um tempo adequado para a tomada de decisão.

PALAVRAS CHAVE. Escalonamento, Otimização combinatória, Metaheurísticas.

Tópicos: PO na Indústria, Programação Matemática, Metaheurísticas

ABSTRACT

This work has its focus on a real long-term maintenance programming problem of an iron ore processing plant of a company in Brazil. The problem is a complex maintenance programming where machine preventive programming orders have to be assigned to the available work teams over 52-week planning. In order to solve it, we developed a general Mixed Integer Linear Programming (MILP) model as well as metaheuristic algorithms based on *Simulated Annealing* (SA), *Variable Neighborhood Search* (VNS), *Biased Random-Key Genetic Algorithm* (BRKGA). The MILP model was used to validate the results of the metaheuristic algorithms applied to smaller instances. The metaheuristic algorithms were able to find better solutions than those employed by the company, at a time adequate for decision making.

KEYWORDS. Scheduling, Combinatorial Optimization, Metaheuristics.

Paper topics: Industry OR, Mathematical Programming, Metaheuristics

https://proceedings.science/p/107544?lang=pt-br

1. Introdução

Uma prática muito frequente em grandes indústrias para manter o sistema de produção em boas condições de funcionamento é a realização de manutenções preventivas em seus equipamentos. O objetivo é corrigir eventuais falhas antes que elas ocorram, de forma a evitar a interrupção da produção. Essa prática é adotada visto que o custo das manutenções corretivas, aquelas realizadas após falhas do equipamentos, é significativamente maior que as manutenções preventivas.

Tendo em vista que um melhor planejamento das ordens de manutenções preventivas é necessário para maximizar o número de ordens de manutenções executadas e minimizar a mão de obra necessária, este trabalho propõe uma modelagem matemática para o problema, assim como abordagens heurísticas, com o objetivo de encontrar o melhor ou bons planejamentos das ordens de manutenções em cenários de produção. Dessa forma, pode-se ter um maior índice de realização das manutenções. Quanto maior esse índice, menor a probabilidade de ocorrência de paradas para manutenções corretivas e, em consequência, menor custo para as empresas. Assim, destaca-se o grande interesse prático na solução deste problema.

Por outro lado, há interesse de pesquisa em métodos de solução para o problema, visto que ele é da classe NP-difícil. De fato, esse problema se reduz ao de sequenciamento em máquinas paralelas não-relacionadas, que é NP-difícil [Brucker, 2007], ao considerarmos as ordens de serviço como tarefas e as equipes como máquinas. Assim, se justifica o desenvolvimento de algoritmos heurísticos que possam resolvê-lo.

2. Planejamento de Ordens de Manutenção Preventiva de Longo Prazo

O Problema de Planejamento de Ordens de Manutenção Preventiva de Longo Prazo (PPOMPLP) consiste na atribuição das ordens de manutenção para as equipes de manutenção em um horizonte de planejamento, aqui considerado de 52 semanas. Na construção desse mapa de 52 semanas, diversas restrições devem ser consideradas, sendo que a maioria delas está diretamente relacionada a problemas de escalonamento e sequenciamento. Uma ordem de manutenção tem que ser alocada simultaneamente a uma equipe de trabalho e a um equipamento, sendo que a equipe de trabalho não pode executar mais de uma ordem de manutenção simultaneamente e cada equipamento não pode ter mais de uma ordem de manutenção sendo executada simultaneamente. Estas duas alocações simultâneas são as principais restrições que caracterizam o problema; há um limite de horas que cada equipe de trabalho pode trabalhar; há uma janela de tempo que cada ordem de manutenção deve ser executada; cada ordem de manutenção deve ser executada por um tipo específico de mão de obra; pode haver mais de uma equipe de trabalho capaz de executar a mesma ordem de manutenção.

Se considerarmos uma simplificação do PPOMPLP, ele pode ser comparado ao problema de sequenciamento em máquinas paralelas. Nesta comparação, as ordens de manutenção seriam as tarefas e as equipes de trabalho seriam as máquinas. Cada grupo de equipes com a mesma especialidade seria um conjunto de máquinas paralelas. Nessa analogia, o problema abordado neste trabalho seria definido como $P_m \mid r_j M_j \mid \gamma$, no qual o campo γ seria definido de forma a minimizar a mão de obra necessária para executar o maior número de tarefas. A principal diferença seria na função objetivo, pois as funções objetivo dos problemas de sequenciamento são todas relacionadas ao tempo final de alguma tarefa. Como o tempo final é determinado em função do sequenciamento das tarefas, todas as tarefas são alocadas. Já no problema foco deste trabalho, o tempo é fixo e o objetivo é alocar o máximo de tarefas com o mínimo de "máquinas". Dessa forma, não é garantida a alocação de todas as tarefas.

O conceito de multiprocessamento de tarefas pode ser aplicado ao PPOMPLP se considerarmos dois processamentos: um seria pela equipe de trabalho, e o outro pelo equipamento no qual a manutenção deve ser executada. Uma adaptação necessária seria a consideração de que um dos

processadores seria um conjunto de máquinas paralelas, pois várias equipes de trabalho são compatíveis para a execução de uma tarefa. Além disso, é necessário mudar a função objetivo (conforme já discutido anteriormente). Nessa analogia, o problema é definido como $MPT2_m \mid r_j M_j \mid \gamma$.

Na empresa em estudo, o mapa de 52 semanas é feito por uma gerência de engenharia de manutenção utilizando o módulo de manutenção preventiva do software SAP da empresa alemã SAP SE (`https://www.sap.com/corporate/en.html`). O problema é que esse módulo de manutenção não leva em consideração as restrições do problema e, assim, pode gerar soluções inviáveis. Para tentar executar o máximo de manutenções possíveis, a equipe de programação de manutenção de curto prazo, ou seja, aquela que trabalha com um horizonte de planejamento de 30 dias, refaz a programação, incluindo horas extras e a utilização de mão de obra de empresas terceirizadas. Apesar disso, e mesmo com a elevação de custo decorrente dessas medidas, nem todas as manutenções preventivas são realizadas. Portanto, o desafio deste trabalho é justamente a proposição de uma forma mais eficiente de alocação dessas manutenções às equipes.

3. Revisão da Literatura

Os problemas de otimização de manutenção não são novos. Apesar de não ter sido encontrado na literatura um problema igual ao do foco deste trabalho, diferentes problemas relacionados à melhoria ou otimização da manutenção foram objeto de estudos em diversos trabalhos. Uma revisão de trabalhos nesse tema é feita por Simões et al. [2011] e Sharma et al. [2011]. Entretanto, poucos foram os trabalhos revisados relacionados diretamente à otimização do custo de manutenção.

Yamayee et al. [1983] propuseram uma formulação matemática para a solução exata do problema de escalonamento de manutenções preventivas e desenvolveram um *framework* de programação dinâmica para sua resolução. A programação dinâmica foi considerada a técnica mais apropriada para a resolução do problema. O problema estudado consiste no escalonamento de 21 ordens de manutenção com diferentes tamanhos e diferentes custos. Assim como no presente trabalho, a ordem de manutenção, a duração e a janela de tempo já são conhecidos e fazem parte da instância. O objetivo considerado pelos autores, no entanto, é diferente, pois consiste em minimizar o custo esperado de produção e o custo da falta de confiabilidade. Foi observado por eles um grande esforço computacional para resolver o problema e identificado que o motivo é que a função objetivo tem dois componentes. Para estudar o efeito de cada componente da função objetivo, foram realizados dois estudos de caso, um com o custo esperado de produção e o outro com o custo da falta de confiabilidade como função objetivo. Os escalonamentos foram totalmente diferentes. A análise do estudo de caso mostrou que o custo de produção varia muito pouco nas proximidades do ponto ótimo. Foi observado que a minimização do custo de falta de confiabilidade altera muito pouco o custo de produção. Além disso, o tempo computacional para a minimização do custo da falta de confiabilidade é muito menor e, portanto, é a mais recomendada.

Yao et al. [2004] estudaram um problema de manutenção na indústria de semicondutor. Para resolução do problema, foi proposto um *framework* com duas camadas hierárquicas, estando o planejamento de longo prazo na camada externa e o planejamento de curto prazo na camada interna. O foco do trabalho foi a resolução do problema de curto prazo e foi proposto um modelo de programação inteira mista (MIP, das iniciais em inglês de *Mixed Integer Programming*) para a programação de 29 ordens de manutenção a serem executadas em 11 ferramentas diferentes, em uma janela de tempo pré-determinada. O escalonamento de várias ordens de manutenção para a mesma ferramenta é equivalente à programação de várias ordens de manutenção de equipamento simultaneamente no problema em estudo no presente trabalho. O objetivo é maximizar a disponibilidade geral da ferramenta e minimizar a indisponibilidade nos períodos nos quais é esperada uma alta carga de trabalho para a ferramenta. Assim como no problema foco deste trabalho, a função

objetivo não leva em consideração os custos da manutenção. Para a análise dos resultados foi feito um estudo de caso baseado em um modelo de simulação já utilizado e validado pela empresa. Os resultados foram, então, comparados com o planejamento atual da empresa. Das 11 ferramentas, 10 não tiveram um resultado estatisticamente diferente. Houve, então, uma ferramenta para a qual o modelo teve um resultado 13,9% melhor. Apesar de não ser estatisticamente diferente, houve uma melhora média observada de aproximadamente 1,6%, o que representou um ganho financeiro de U$39 milhões por ano.

Adhikary et al. [2016] propuseram um algoritmo genético multiobjetivo para o escalonamento da manutenção preventiva. Os objetivos envolvem a maximização da disponibilidade e a minimização do custo de manutenção de uma planta de operação contínua em série. Foi realizado um estudo de caso de um gerador de uma unidade de geração de energia de 210 MW. Os resultados apresentados pelos autores mostraram que com o algoritmo genético a disponibilidade aumentou em 3% e houve uma redução do custo de manutenção de 437 milhões de rúpias indianas (aproximadamente U$6 milhões).

Saraiva et al. [2011] propuseram um modelo de programação inteira mista (MIP) e um algoritmo baseado na meta-heurística *Simulated Annealing* (SA) para a programação de ordens de manutenção de geradores térmicos. O estudo de caso apresentado envolveu 29 geradores e 29 ordens de manutenção, sendo uma para cada gerador. Assim como neste trabalho, cada ordem de manutenção tem uma janela de tempo pré-determinada. O objetivo foi minimizar as perdas financeiras relacionadas às paradas dos geradores. Foi proposto o SA com uma penalização para cada restrição que pudesse ser violada. Algumas restrições foram consideradas fortes e não podiam ser violadas. Foram realizados dois estudos de caso, nos quais o SA alcançou uma solução sem a violação de nenhuma restrição em um tempo de 15 minutos.

4. Programação Matemática e Heurísticas

O Problema de Planejamento de Ordens de Manutenção Preventiva de Longo Prazo (PPOM-PLP) pode ser descrito como a seguir. Considere $\mathcal{T} = \{1, 2, ..., n\}$ o conjunto de n manutenções a serem realizadas por um conjunto $\mathcal{W} = \{1, 2, ..., m\}$ de m equipes de trabalho. A cada manutenção $i \in \mathcal{T}$ é associado um conjunto $\mathscr{W}_i \subseteq \mathcal{W}$ de equipes de trabalho capazes de realizá-la. A cada manutenção $i \in \mathcal{T}$ é associado um tempo p_i necessário para executá-la; um equipamento A_i no qual a manutenção deve ser prestada; e uma janela de tempo $[\, e_i,\, l_i\,]$, que especifica o intervalo de tempo em que a manutenção i pode ser realizada. A penalidade por não realizar a manutenção $i \in \mathcal{T}$ é dada por w_i. Cada equipe de trabalho $k \in \mathcal{W}$ tem disponibilidade de prestar h_k horas de mão de obra. O conjunto $\mathscr{T}_k \subseteq \mathcal{T}$ indica as manutenções que podem ser realizadas pela equipe de trabalho $k \in \mathcal{W}$, ou seja, $\mathscr{T}_k$ indica as manutenções que a equipe de trabalho k tem habilidade para executar. O objetivo é determinar o plano de execução das manutenções que minimiza o número de equipes de trabalho necessárias às manutenções, respeitando-se todas as restrições do problema, definidas formalmente no modelo matemático.

A partir das variáveis de decisão a seguir, o problema é modelado pelas Equações (1)-(17):

- $x_{ij}^k = 1$, se a manutenção i precede imediatamente a manutenção j executada pela equipe de trabalho k;
- $y_{ik} = 1$, se a manutenção i será executada pela equipe de trabalho k;
- $z_k = 1$, se a equipe de trabalho k é utilizada;
- $c_{ik} \geq 0$, tempo de conclusão da manutenção i pela equipe de trabalho k.
- $r_{ij} = 1$, se a manutenção i precede imediatamente a manutenção j, para o caso em que ambas as manutenções referem-se ao mesmo equipamento ($A_i = A_j$) e que ambas são executadas por equipes diferentes.

$$\min \sum_{k \in \mathcal{W}} z_k \; + \; \sum_{i \in \mathcal{T}_k} w_i \left(1 - \sum_{k \in \mathscr{W}_i} y_{ik}\right) \tag{1}$$

$$\sum_{k \in \mathscr{W}_i} y_{ik} \leq 1 \qquad \forall i \in \mathcal{T} \tag{2}$$

$$\sum_{i \in \mathscr{T}_k \cup \{0\} \setminus \{j\}} x_{ij}^k = y_{jk} \qquad \forall j \in \mathcal{T},\, k \in \mathscr{W}_j \tag{3}$$

$$\sum_{j \in \mathscr{T}_k} x_{0j}^k = z_k \qquad \forall k \in \mathcal{W} \tag{4}$$

$$\sum_{i \in \mathscr{T}_k \cup \{0\} \setminus \{l\}} x_{il}^k = \sum_{j \in \mathscr{T}_k \cup \{0\} \setminus \{l\}} x_{lj}^k \qquad \forall k \in \mathcal{W},\, l \in \mathscr{T}_k \tag{5}$$

$$c_{0k} = 0 \qquad \forall k \in \mathcal{W} \tag{6}$$

$$c_{jk} \geq c_{ik} + p_j - M'_{ij} \cdot (1 - x_{ij}^k) \qquad \forall k \in \mathcal{W},\, i \in \mathscr{T}_k \cup \{0\},\, j \in \mathscr{T}_k \tag{7}$$

$$c_{ik} \geq (e_i + p_i) \cdot y_{ik} \qquad \forall k \in \mathcal{W},\, i \in \mathscr{T}_k \tag{8}$$

$$c_{ik} \leq l_i \qquad \forall k \in \mathcal{W},\, i \in \mathscr{T}_k \tag{9}$$

$$c_{jk'} \leq c_{ik} - p_i + M''_{ij} \cdot r_{ij} \qquad \forall k \in \mathcal{W},\, k' \in \mathcal{W},\, i \in \mathscr{T}_k,\, j \in \mathscr{T}_{k'},\, | \, k \neq k', i < j, A_i = A_j \tag{10}$$

$$c_{jk'} \geq c_{ik} + p_j - M'_{ij} \cdot (1 - r_{ij}) \qquad \forall k \in \mathcal{W},\, k' \in \mathcal{W},\, i \in \mathscr{T}_k,\, j \in \mathscr{T}_{k'},\, | \, k \neq k', i < j, A_i = A_j \tag{11}$$

$$c_{ik} \leq h_k \qquad \forall k \in \mathcal{W},\, i \in \mathscr{T}_k \tag{12}$$

$$x_{ij}^k \in \{0, 1\} \qquad \forall k \in \mathcal{W},\, i \in \mathscr{T}_k \cup \{0\},\, j \in \mathscr{T}_k \cup \{0\} \tag{13}$$

$$y_{ik} \in \{0, 1\} \qquad \forall k \in \mathcal{W},\, i \in \mathscr{T}_k \cup \{0\} \tag{14}$$

$$z_k \in \{0, 1\} \qquad \forall k \in \mathcal{W} \tag{15}$$

$$c_{ik} \geq 0 \qquad \forall k \in \mathcal{W},\, i \in \mathscr{T}_k \tag{16}$$

$$r_{ij} \in \{0, 1\} \qquad \forall i \in \mathcal{T},\, j \in \mathcal{T} \,|\, i < j, A_i = A_j \tag{17}$$

A função objetivo (1) busca reduzir a quantidade de mão de obra necessária para realizar o maior número possível de manutenções. Todas as restrições são detalhadas em Aquino et al. [2017, 2018]. Destaca-se aqui as restrições (7), (10) e (11), as quais garantem que a equipe de trabalho não pode executar mais de uma ordem de manutenção simultaneamente e que cada equipamento não pode ter mais de uma ordem de manutenção sendo executada simultaneamente.

Com essa formulação MILP, no entanto, não foi possível resolver instâncias reais do problema. Em vista disso, também foram desenvolvidos cinco algoritmos meta-heurísticos, a saber: 1) *Simulated Annealing* – SA [Kirkpatrick et al., 1983]; 2) *Variable Neighborhood Search* – VNS

[Mladenović e Hansen, 1997]; 3) MSVNS, que combina os métodos *Multi-Start* – MS e VNS; 4) *Biased Random-Key Genetic Algorithm* – BRKGA [Martinez et al., 2011]; 5) BRKMA, que inclui no BRKGA uma busca local nos indivíduos antes da operação de cruzamento. Todos os algoritmos desenvolvidos, assim como a descrição das instâncias estão detalhados em Aquino [2018] e Aquino et al. [2017, 2018].

5. Experimentos e Resultados Computacionais

A formulação MILP foi programada na linguagem C++ utilizando-se o Concert Technology Library do CPLEX, versão 12.5 acadêmica, na configuração padrão. Os algoritmos SA, VNS, MSVNS, BRKGA e BRKMA foram implementados na linguagem C++.

Os experimentos foram divididos em dois conjuntos de testes. No primeiro, realizados em um computador Intel Xeon CPU E3-1225 v5 @3.30GHz $\times$ 4, com 32 GB de RAM, e sistema operacional Ubuntu 16.04.3 LTS 64 bits, foram analisados os resultados do MILP, do SA e do VNS. No segundo, realizados em um computador Intel(R) Xeon(R) CPU E5-2660 v2 @ 2.20GHz $\times$ 40, com 384 GB de memória RAM sob o sistema operacional CentOS Release 6.8 (Final) Kernel Linux 2.6.32-642.1.1.el6.x86_64, foram analisados os resultados do MSVNS, do BRKGA e do BRKMA, visto que as instâncias desse conjunto têm dimensões maiores e, por isso, foram propostos algoritmos com processamento paralelo para tratá-las. Para garantir uma análise coerente dos resultados de forma a determinar o algoritmo de melhor desempenho, foi feita uma análise estatística (no caso, com teste de Friedman e testes *post-hoc*), considerando um nível de confiança de 95%. Em todos os experimentos os parâmetros dos algoritmos heurísticos foram devidamente calibrados por meio da ferramenta iRACE [López-Ibáñez et al., 2016].

O primeiro conjunto de testes envolveu 100 instâncias pequenas derivadas de uma instância real com 33484 ordens de manutenção, 1032 equipamentos e 145 equipes. Em relação ao primeiro conjunto de testes, observou-se que a formulação MILP e o SA encontraram a solução ótima para 26% das instâncias e o VNS para 24% delas. O SA obteve os melhores resultados para 96% das instâncias, o VNS para 81% e o MILP para 53%. Nesses testes, o tempo de processamento do MILP foi fixado em uma hora, enquanto o tempo médio demandado pelos algoritmos SA e VNS foram de 50,6 segundos e 27,1 segundos, respectivamente. A análise estatística dos resultados mostrou a superioridade do algoritmo SA. A formulação MILP, apesar de só resolver problemas muito pequenos, foi importante para validar o correto funcionamento e desempenho dos algoritmos heurísticos.

Foi feita, também, uma comparação com o indicador que é controlado pela equipe de engenharia de manutenção da empresa, a saber, o percentual de ordens de manutenção que são executadas em relação às programadas. O valor médio histórico é próximo a 50%, isto é, apenas cerca da metade das ordens de manutenção são realizadas. Aplicando-se os algoritmos heurísticos à instância real, com o SA foi possível alocar 90% das ordens de manutenção em 182 horas e com o VNS, 92,5% em 2 horas de processamento. O MILP não foi capaz de resolvê-la. É importante destacar que este indicador. no entanto, não é otimizado pela função objetivo dada pela Equação (1), pois nele não se leva em consideração a prioridade das tarefas e, sim, apenas o número delas. Os resultados desses testes validaram o uso dos algoritmos heurísticos para a resolução de problemas de interesse prático.

Para o segundo conjunto de testes foram adicionadas mais 39 instâncias maiores, todas também derivadas da instância real. As 139 instâncias foram agrupadas em 4 conjuntos, {P, M, G, GG}, de acordo com a quantidade de ordens de manutenção: P (20 a 80), M (150 a 600), G (1200 a 4800) e GG (9600 a 33484). A Tabela 1 mostra os resultados dos desvios médios para cada um desses conjuntos.

Tabela 1: Desvio médio dos algoritmos MSVNS, BRKGA e BRKMA

Conjunto	MSVNS	BRKGA	BRKMA
P	2,09	**1,05**	1,07
M	**0,59**	9,85	7,25
G	**5,27**	32,14	24,85
GG	109,99	7,97	**5,59**

Os testes mostraram que há diferença estatística entre os resultados. A única exceção ocorreu no conjunto M, pois não houve diferença estatística entre o BRKGA (9,85) e o BRKMA (7,25). Esse resultado estatístico facilita a interpretação dos resultados. Observa-se que, dependendo da dimensão da instância, um algoritmo diferente tem o melhor desempenho. Para as instâncias de dimensões pequenas, o algoritmo BRKGA obteve os melhores resultados. Nas instâncias de dimensões médias e grandes o algoritmo de melhor desempenho foi o MSVNS. O BRKMA, por sua vez, foi o melhor nas instâncias de dimensões gigantes. Tendo em vista o interesse prático de resolução do mapa de 52 semanas, este resultado da instância gigante é o que deve ser considerado na empresa em estudo. Na comparação com o indicador controlado pela equipe de engenharia de manutenção da empresa, observa-se que o BRKMA, demandando um tempo de 9 horas de processamento, conseguiu atender a 95% das ordens de manutenção da instância real.

6. Conclusões

Este trabalho tratou o problema de alocação de ordens de manutenção preventiva. Foram propostos uma formulação de programação matemática, bem como algoritmos heurísticos para resolvê-lo. Os resultados obtidos mostraram que os algoritmos desenvolvidos são capazes de produzir soluções bem melhores do que aquelas empregadas pela empresa, e em um tempo adequado para a tomada de decisão.

7. Publicações

Dois artigos foram produzidos com os resultados intermediários da dissertação, a saber:

1. **Título**: A Mixed-Integer Linear Programming Model and a Simulated Annealing Algorithm for the LongTerm Preventive Maintenance Scheduling Problem
   Autores: Roberto Dias Aquino, Jonatas Batista Costa das Chagas e Marcone J. Freitas Souza
   Evento: International Conference on Intelligent Systems Design and Applications - ISDA 2017
   Local: New Delhi, India; **Período**: 14 a 16 de dezembro de 2017
   Qualis CC: B1
   Destaque: Trabalho finalista na área *Artificial Intelligence and Decision Support Systems* ao prêmio *Best Student Paper*
2. **Título**: A Variable Neighborhood Search Algorithm for the Long-Term Preventive Maintenance Scheduling Problem
   Autores: Roberto Dias Aquino, Jonatas Batista Costa das Chagas e Marcone J. Freitas Souza
   Evento: 20th International Conference on Enterprise Information Systems (ICEIS)
   Local: Funchal, Portugal; **Período**: 21 a 24 de março de 2018
   Qualis CC: B2

Agradecimentos

O autor agradece ao orientador Prof. Dr. Marcone Jamilson Freitas Souza, ao coorientador MSc Jonatas Batista Costa das Chagas, à Universidade Federal de Ouro Preto (UFOP) e aos órgãos de fomento CAPES, FAPEMIG e CNPq pelo apoio ao desenvolvimento deste trabalho.

Referências

Adhikary, D. D., Bose, G. K., Jana, D. K., Bose, D., e Mitra, S. (2016). Availability and cost-centered preventive maintenance scheduling of continuous operating series systems using multi-objective genetic algorithm: a case study. *Quality Engineering*, 28(3):352–357.

Aquino, R. D. (2018). Abordagem exata e heurísticas para o problema de planejamento de ordens de manutenção de longo prazo: um estudo de caso industrial de larga escala. Dissertação de mestrado, Programa de Pós-graduação em Ciência da Computação, Universidade Federal de Ouro Preto, Ouro Preto, MG. Disponível em `https://repositorio.ufop.br/handle/123456789/10820`.

Aquino, R. D., Chagas, J. B. C., e Souza, M. J. F. (2017). A mixed-integer linear programming model and a simulated annealing algorithm for the long-term preventive maintenance scheduling problem. In *Proceedings of the 17th Int. Conference on Intelligent Systems Design and Applications (ISDA 2017)*, volume 736, p. 144–153, New Delhi, India. Cham, Switzerland: Springer.

Aquino, R. D., Chagas, J. B. C., e Souza, M. J. F. (2018). A variable neighborhood search algorithm for the long-term preventive maintenance scheduling problem. In *Proceedings of the 20th International Conference on Enterprise Information Systems (ICEIS 2018)*, volume 1, p. 303–310, Funchal, Portugal. Setúbal, Portugal: SciTePress.

Brucker, P. (2007). *Scheduling algorithms*, volume 3. Springer.

Kirkpatrick, S., Gelatt, C. D., e Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598):671–680.

López-Ibáñez, M., Dubois-Lacoste, J., Pérez Cáceres, L., Stützle, T., e Birattari, M. (2016). The irace package: Iterated racing for automatic algorithm configuration. *Operations Research Perspectives*, 3:43–58.

Martinez, C., Loiseau, I., Resende, M. G. C., e Rodriguez, S. (2011). BRKGA algorithm for the capacitated arc routing problem. *Electronic Notes in Theoretical Computer Science*, 281:69–83.

Mladenović, N. e Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11):1097–1100.

Saraiva, J. T., Pereira, M. L., Mendes, V. T., e Sousa, J. C. (2011). A simulated annealing based approach to solve the generator maintenance scheduling problem. *Electric Power Systems Research*, 81(7):1283–1291.

Sharma, A., Yadava, G., e Deshmukh, S. (2011). A literature review and future perspectives on maintenance optimization. *Journal of Quality in Maintenance Engineering*, 17(1):5–25.

Simões, J. M., Gomes, C. F., e Yasin, M. M. (2011). A literature review of maintenance performance measurement: A conceptual framework and directions for future research. *Journal of Quality in Maintenance Engineering*, 17(2):116–137.

Yamayee, Z., Sidenblad, K., e Yoshimura, M. (1983). A computationally efficient optimal maintenance scheduling method. *IEEE Transactions on Power Apparatus and Systems*, 102(2):330–338.

Yao, X., Fernández-Gaucherand, E., Fu, M. C., e Marcus, S. I. (2004). Optimal preventive maintenance scheduling in semiconductor manufacturing. *IEEE Transactions on Semiconductor Manufacturing*, 17(3):345–356.

https://proceedings.science/p/107544?lang=pt-br