

# Utilização da Programação Evolucionária para escalonamento de pontes rolantes em uma aciaria

Allysson Steve Mota Lacerda, Renato Dourado Maia, João Antônio de Vasconcelos

Programa de Pós-Graduação em Engenharia Elétrica - PPGEE

Universidade Federal de Minas Gerais – UFMG

{stevelerda, rdmaia, joao}@cpdee.ufmg.br

## I. INTRODUÇÃO

O problema abordado consiste em otimizar o processo de produção em uma aciaria, especialmente no que diz respeito à operação de pontes rolantes. O processo consiste em transportar placas de aço entre diversos pontos da planta, a fim de realizar alguns tratamentos e finalmente despachá-las.

Como em qualquer processo de produção real, há uma série de restrições, sendo que as principais são:

- As pontes nunca podem se cruzar;
- Cada ponte só pode transportar uma placa por vez;
- Cada recurso (conversor, forno etc.) só pode ser utilizado por uma placa de cada vez;
- Algumas pontes ficam restritas a determinados trechos da planta.

Tomando como base tais restrições, o principal objetivo é otimizar a operação de tais pontes, de forma a diminuir o tempo de operação de um dado conjunto de ordens de serviço.

## II. MODELAGEM

A planta foi dividida em 03 regiões (conversores, tratamentos e lingotamentos) e, com isso, verificou-se que o número de rotas para as ordens de serviço se limitam em quatro.

A partir dessas informações, foi criada uma tabela com as probabilidades de utilização de cada ponte em cada trecho. Inicialmente, esses valores foram atribuídos manualmente, mas durante a execução do algoritmo eles são ajustados automaticamente.

Com base nesses dados, os indivíduos foram gerados como sendo a sequência de pontes a ser utilizada por cada ordem de serviço.

TRECHO   PONTE	A	B	C
pos. 01 a pos. 02	0.7	0.3	0.0
pos. 01 a pos. 03	0.4	0.6	0.0
pos. 02 a pos. 02	0.3	0.5	0.2
pos. 02 a pos. 03	0.1	0.3	0.6

**Tabela 1** Probabilidades de utilização das pontes

Numa segunda etapa, tais indivíduos são testados e o menor tempo de execução é armazenado. Logo após, a tabela de probabilidades é modificada aleatoriamente e o processo se repete. Se o resultado dos indivíduos gerados por essa nova tabela for melhor que o anterior, a nova tabela é armazenada e o processo continua até o número de gerações definido. Para os testes, as ordens de serviço foram divididas em blocos de 50 e os tempos de execução somados.

## III. TESTES

Para realização dos testes, foi utilizada uma base fornecida contendo 498 ordens de serviço bastante heterogêneas e com conflitos de utilização de recursos. A base foi tratada apenas para adequação ao modelo de entrada do algoritmo, sem alterações em sua estrutura.

Foram realizados diversos testes com o número de indivíduos, grau de mutação da tabela de probabilidades e número de gerações. Foi observado que um número reduzido de indivíduos pode comprometer a eficiência do algoritmo, já que a região de busca se torna bastante reduzida. Por sua vez, um valor muito pequeno para o grau de mutação influencia apenas o tempo de convergência do algoritmo, porém sem grandes modificações no desempenho do sistema.

### A. Resultados

A partir de uma tabela de probabilidades semelhante à Tabela 1, o algoritmo foi executado repetidamente com 100 indivíduos e 100 gerações. Como os melhores resultados são armazenados e a base utilizada foi a mesma, o sistema apresentou ganhos de desempenho em todas as execuções, porém com menor variação à medida em que os tempos foram diminuindo.

No melhor resultado obtido, todas as ordens de serviço foram executadas em 8702 minutos (6,04 dias) e a média atual gira em torno de 8850 minutos (6,18 dias), o que representa uma diferença em torno de 03 horas (2% do tempo total). É possível notar, então, que o algoritmo está bastante estável e alcançou resultados expressivos, respeitando todas as restrições estabelecidas.

## IV. CONCLUSÕES E TRABALHOS FUTUROS

Embora não haja resultados comparativos de outras técnicas, a Programação Evolucionária apresentou resultados satisfatórios e robustos para a resolução do problema de agendamento de tarefas. Apesar de necessitar de parâmetros externos, como qualquer método evolucionário, seu número é reduzido e podem ser facilmente definidos de acordo com a natureza do problema. Mais importante que a escolha dos parâmetros exógenos é a escolha de uma modelagem adequada e a forma de abordagem do problema.

Para uma próxima etapa, pretende-se medir o tempo médio de espera em cada trecho e, então, evoluir apenas aquele com pior desempenho. Espera-se com isso, realizar um ajuste mais fino no algoritmo e, conseqüentemente, alcançar resultados ainda melhores.