

# Heurísticas para o Vehicle Routing Problem with Time Windows

UFMG - UNIVERSIDADE FEDERAL DE MINAS GERAIS

DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

RODRIGO MARINHO PASSOS

EVANDRINO BARROS

[PASSOS,BARROS]@DCC.UFMG.BR

BELO HORIZONTE, AGOSTO/2003

**Resumo:** O *Vehicle Routing Problem with Time Windows (VRPTW)* é um problema de otimização combinatorial complexo que pertence à classe de problemas *NP – Completo* [Joh79], e que vem sendo bastante estudado nos últimos anos. Este artigo apresenta o estado da arte dos métodos exatos e heurísticas para encontrar soluções exatas ou próximas ao ótimo para este problema. Após a descrição dos métodos existentes, este artigo concentra-se no estudo de um problema real de roteamento de veículos e apresenta uma solução aproximada utilizando algoritmos genéticos (GA) [Hol75].

## 1 Introdução

Nos tempos atuais, muitas empresas têm se preocupado cada vez mais com problemas de transporte, gerando uma crescente demanda por soluções tecnológicas que sejam capazes de minimizar os custos com este setor.

A resolução de problemas deste tipo, que envolvem principalmente roteamento e itinerário de veículos, é uma importante sub-área do campo da *Pesquisa Operacional*, mais especificamente da otimização combinatorial, onde os problemas podem ser basicamente classificados como *maximização* ou *minimização* de uma função objetivo. Desta maneira, a aplicação de métodos de otimização capazes de tornar uma frota de veículos otimizada, minimizando a distância percorrida por exemplo, traz muitos benefícios, pois evita gastos desnecessários, garante entregas nos períodos estipulados, diminui a poluição causada por estes veículos, entre outras vantagens.

Problemas de roteamento de veículos, mais conhecidos como *VRP - Vehicle Routing Problems*, têm sido amplamente estudados nas suas mais diversas formas e

variações [Lim00, Des88, Tai97, Lar99, Tom03]. Modelos mais realísticos deste tipo de problema, consideram não apenas o problema geográfico, ou seja, distância, mas também, a restrição de tempo. Assim, este tipo de problema é denominado *VRPTW - Vehicle Routing Problems with Time Windows*, onde cada veículo possui uma “janela de tempo”, na qual o trabalho deve ser realizado.

Na Seção 2 deste artigo, uma breve descrição do *VRPTW* é apresentada. Na Seção 3, alguns métodos exatos e abordagens não exatas, mais utilizadas na literatura para resolver o problema, são descritas. Na Seção 4, apresenta-se uma modelagem para um problema real, utilizando o algoritmo genético, uma heurística não exata para resolver problemas de busca por soluções próximas ao ótimo. Finalmente, na Seção 5, alguns resultados e trabalhos futuros são discutidos.

## 2 Vehicle Routing Problem with Time Windows - VRPTW

Segundo [Jue91], o *VRPTW* envolve o roteamento de frotas de veículos com limitações de capacidade e tempos de viagem, partindo de um depósito central para um conjunto de clientes dispersos geograficamente, cada qual com suas demandas e “janelas de tempo” específicas. Assim, um veículo deixa seu depósito, viaja para pontos de carga específicos até que a capacidade de carga do veículo se esgote, finalmente o veículo retorna para o depósito dentro do prazo de tempo estipulado. O *VRPTW* pode ser simples, onde o prazo estipulado de chegada do veículo não pode ser ultrapassado, mas nada impede que o veículo chegue antes do prazo. No *VRPTW* duplo, as rotas impõe prazos de chegada mínimos e máximos, para evitar que um

veículo fique em tempo ocioso de espera, pois se este tempo for longo, pode prejudicar o atendimento a outros clientes.

Os problemas de roteamento de veículos, da classe *VRP* ou *VRPTW*, são geralmente problemas “difíceis” de ser resolvidos, pois o número de clientes, veículos e restrições pode variar bastante em cada problema, tornando uma solução exata inviável na maioria dos casos. Segundo [Qil], *VRPTW*, com um veículo, pertence à classe *NP – Completo*. No caso mais comum, onde o número de veículos é sempre maior que um, o problema se torna *NP – Hard* [Joh79]. Isto significa que a solução ótima para a maioria das instâncias do *VRPTW* podem ser obtidas apenas em tempo exponencial.

Com o custo proibitivo para encontrar a solução ótima, a alternativa resume-se em utilizar heurísticas ou aproximações capazes de encontrar soluções próximas ao ótimo nestas situações. Existem uma série de abordagens para resolver o *VRPTW* na literatura. Na Seção 3 deste artigo, algumas abordagens não exatas são evidenciadas.

## 3 Solucionando o VRPTW

Nesta seção, apresenta-se uma revisão bibliográfica de alguns dos principais métodos abordados na literatura para encontrar soluções ótimas ou próximas ao ótimo para o *VRPTW*.

### 3.1 Métodos exatos

Os métodos exatos procuram encontrar a solução ótima para o *VRPTW*. As principais publicações que tratam desse método contemplam três princípios:

#### 3.1.1 Programação dinâmica

A programação dinâmica baseia-se na técnica de branch-and-bound para alcançar o ótimo. Um limite inferior para o ótimo é verificado para cada nó incluído na solução. Quando algum novo nó propicia um limite inferior ao atual, esse limite inferior torna-se o novo limite inferior. Cada nó referente ao cliente é visitado somente uma vez. Esse algoritmo é implementado em [Kaa87] e resolvido para até 15 clientes.

#### 3.1.2 Relaxação de Lagrange

O modelo matemático do *VRPTM* [Lar99], com suas restrições, como por exemplo garantir que todo cliente só é visitado uma única vez, sofre um relaxamento

através de relaxação de Lagrange. A partir desse relaxamento um limite inferior global é calculado. A partir desse limite inferior a técnica do branch-and-bound, é aplicada. A cada novo cliente visitado verifica-se se o limite inferior global, não está sendo ultrapassado pela rota estabelecida caso esse novo cliente seja incluído na solução.

### 3.2 Métodos aproximados por heurísticas

Nesse tipo de solução encontramos na bibliografia consultada os seguintes principais métodos:

#### 3.2.1 Heurística de construção de rotas

[Sch86] é o primeiro artigo sobre heurística de construção de rotas para o *VRPTW*. O algoritmo começa com todas as rotas simples possíveis com apenas um cliente (depósito-cliente-depósito). E cada interação calcula-se quais duas rotas podem ser combinadas com a maior economia de recursos.

#### 3.2.2 "Time-oriented-nearest-neighbour heuristic"

Proposta em [Sol86]. Toda rota nesta heurística é iniciada por um cliente ainda não roteado mais próximo do depósito. Essa relação de proximidade é tanto geográfica quanto temporal. A cada interação o cliente mais próximo, geográfica e temporalmente, ao último cliente adicionado é considerado para inserção da rota em questão. Quando a pesquisa por um novo cliente é feita, e um cliente não é encontrado, uma nova rota é iniciada.

### 3.3 Heurísticas baseada em melhoria de rotas

A base para quase todas heurísticas de melhoria de rota é a noção de vizinhança. A vizinhança da solução  $S$  é um conjunto  $N(S)$  que pode ser gerado com uma simples "modificação" de  $S$ .

A idéia é, a partir dos vizinhos, alterar rotas até chegar na melhor solução. Como as pesquisas são baseadas em melhorias localizadas em um rota, esse tipo de técnica é chamada de busca local "local search".

#### 3.3.1 Heurística r-Opt

Uma das mais usadas em roteamento é a heurística *r-Opt*. Aqui  $r$  arcos são removidos e trocados por outros  $r$  arcos. Usualmente o  $r$  é 3, mas 2 é outro valor bastante referenciado.

### 3.3.2 Heurística shift-sequence

Proposta por Schulze e Fahle em [uSF99]. Nela um cliente é movido de um rota para outra então, checando todas as possíveis posições de inserção. Se uma inserção é viável, o cliente é então inserido em outra rota. Esse procedimento se repete até que a viabilidade das rotas alteradas seja atingida.

## 3.4 Métodos baseados em Meta-Heurísticas

As meta-heurísticas são tipicamente baseadas em buscas locais "local search", mas com outros métodos para encontrar as melhores soluções avaliando outros espaços de busca, ou técnicas.

### 3.4.1 Simulate annealing

Essa foi a primeira meta-heurística desenvolvida. Quando se usa essa heurística não se busca pela melhor solução na vizinhança a partir da solução corrente. Ao invés disso, simplesmente, escolhe-se uma solução aleatória a partir dessa vizinhança. A partir dessa configuração inicial verifica-se se ela é aceitável a partir de uma certa probabilidade (temperatura) [Lar99]. Pequenas variações são feitas nessa configuração, de acordo com um número de tentativas ou com o alcance da estabilidade, encontra-se a melhor solução após algumas tentativas. A probabilidade que citamos aqui é equivalente a "temperatura" de processos termodinâmico e metalúrgico. Nesses processos quando um metal em fusão é resfriado lentamente ele tende a se solidificar em um estrutura de energia mínima sob uma determinada temperatura.

### 3.4.2 TABU Search

Assim como a heurística "Simulated Annealing", a "TABU Search" é antiga. Ela foi introduzida por Glover em 1989 [Lau94]. Nela cada interação a partir da vizinhança da solução corrente é explorada e a melhor solução é selecionada com a nova solução corrente. [Via98]. Para evitar ciclos, a visita à soluções recentemente visitadas é proibida. Essa proibição é possível graças a uma lista "TABU", que armazena as soluções proibidas ou que já foram melhores em outras interações. O critério mais usado para parar a busca "TABU" é quando se chega a um número constante de interações sem qualquer ganho na melhor solução até então encontrada.

Como solução inicial a heurística de inserção de Solomon [Sol86] é utilizada. As duas heurísticas de me-

lhoria, 2-opt e Or-opt, são usadas para explorar a vizinhança e gerar soluções locais. O algoritmo da meta-heurística alterna as duas estratégias, quando uma não gera melhoria na solução global, a partir de um determinado número de interações.

### 3.4.3 Algoritmo Genético - GA

Devido à complexidade do problema de roteamento de veículos, métodos baseados em inteligência artificial são os mais promissores na tentativa de se encontrar boas soluções. Um dos métodos mais utilizados é o algoritmo genético (GA), uma classe de algoritmos de busca heurística adaptativa baseados na genética da população [Hol75], mais especificamente, nas regras da teoria de reprodução, seleção e evolução, proposta por Darwin [Dar59].

O GA é baseado nos cromossomos ou indivíduos de uma população, onde cada indivíduo representa uma possível solução. Quando novos indivíduos são criados, novas propostas de solução são geradas, através da combinação dos cromossomos presentes nos indivíduos atuais. Os indivíduos são agrupados em gerações. Um conjunto de indivíduos resulta em uma população. A propriedade mais importante do indivíduo é o "*fitness*", baseado na função objetivo. Este valor determina a habilidade relativa de um indivíduo de sobreviver e repassar seus genes para a próxima geração. Gerações sucessivas são realizadas até que algum critério de parada seja alcançado, normalmente, o tempo computacional gasto ou o número de indivíduos gerados. O Programa 1 mostra o fluxo do algoritmo genético, que possui os seguintes componentes: uma população de *strings* binárias, parâmetros de controle, uma função *fitness*, operadores genéticos (cruzamento e mutação), um mecanismo de seleção e um mecanismo para codificar as soluções como *strings* de bits.

```
Geracao = 0;
Inicializa populacao inicial;
Avalia populacao;
LOOP (condicao de termino)
    Geracao = Geracao + 1;
    Seleciona;
    Cruzamento;
    Mutacao;
    Avalia populacao;
FIM DO LOOP
```

Programa 1. Fluxo do algoritmo genético.

De fato, cada etapa do GA pode utilizar métodos ou heurísticas diferentes. Assim, a forma como o GA é

implementado, depende da natureza do problema. Na Seção seguinte, uma modelagem de uma instância real do VRPTW é apresentada.

### 3.5 Heurísticas híbridas

A partir da combinação de diferentes características das metas-heurísticas obtém-se um heurística híbrida. Uma tendência bastante usada na comunidade que lida com meta-heurísticas [Larsen99] é utilizar a meta-heurística TABU Search com outras meta-heurísticas, tais como "Simulated Annealing" e mecanismos de melhoria de rotas, como  $\lambda$ -interchange que troca um subconjunto de  $\lambda$  clientes de um rota para outra [Sun94].

A meta-heurística "Simulated Annealing", quando usada sozinha, também utiliza o mecanismo  $\lambda$ -interchange para gerar soluções locais utilizadas nas tentativas de melhoria por essa meta-heurística.

## 4 Uma instância VRPTW real

Nesta seção, um problema real VRPTW é identificado, e uma solução utilizando GA apresentada. O GA foi escolhido para resolver esta instância do VRPTW por ter demonstrado sucesso na resolução de outros complexos problemas de roteamento de veículos e alocação de horários, como pode ser verificado em [Des88, Nyg93]. Além disso, o algoritmo é relativamente simples, como mostrado no Programa 1.

Um problema VRPTW real, pode ser observado na operação de extração de minério em minerações de ferro de céu aberto em geral [Alv97]. Nestas minerações, os veículos (caminhões) são despachados para rotas pré-definidas, que possuem pontos de carga e descarga de material, no caso, minério extraído da mina. Além disso, como em todo problema VRPTW, os caminhões recebem um prazo para carregamento e basculamento (descarga) que devem ser respeitados, para evitar tempo ocioso de caminhões e falta de atendimento de demanda de material para os depósitos, no caso, dos britadores, onde o material extraído é processado. Desta maneira, o problema de roteamento de veículos em minerações configura um problema VRPTW clássico, mas os veículos precisam respeitar outras restrições, além da capacidade de transporte, tempo para cumprimento da rota e distância percorrida.

Nas minerações, os pontos de carga (frentes de lavra) possuem características próprias, como quantidade de material disponível e qualidade do material (entende-se por qualidade a porcentagem da concentração de elementos químicos no material). Pode exis-

tir mais de um ponto de descarga (britadores), mas geralmente existe um único ponto central de processamento de minério. Desta maneira, a operação de extração de minério consiste em definir as rotas e a quantidade de caminhões, para atingir as metas de produção (quantidade) e qualidade estipuladas para cada depósito (britador). Para uma descrição mais detalhada do processo de operação de uma mineração, veja [Alv97].

Conhecendo-se a produção e qualidade desejada nos depósitos, os tempos médios das possíveis rotas, as distâncias médias entre pontos de carga e descarga, os tempos médios de carregamento e basculamento dos caminhões, a quantidade de caminhões disponíveis e as características de cada ponto de carga, o processo de otimização visa atender os seguintes objetivos:

1. obter o menor tempo de fila, ou seja, tempo de espera para carregamento ou basculamento dos caminhões;
2. obter o menor desvio de qualidade do minério extraído;
3. obter o máximo de produção (quantidade de material).

Pela descrição do processo, o problema consiste em determinar o número de caminhões a serem despachados em cada rota. Assim, um dos mais importantes fatores para o processo de otimização, é a determinação de uma nova rota (ou a mesma rota) para cada término de carregamento ou basculamento dos caminhões, ou seja, determinar um novo despacho. Este trabalho consiste em determinar qual a melhor rota possível, visando atender as metas estipuladas.

### 4.1 Solucionando o VRPTW em mineração utilizando o GA

A modelagem de um problema VRPTW real utilizando o GA, deve-se concentrar em representar as variáveis do problema, por exemplo, como os indivíduos serão representados, e informar como cada etapa do algoritmo genético representando no Programa 1, deverá proceder. Mais especificamente, as etapas de seleção, cruzamento, mutação etc, podem utilizar os métodos tradicionais encontrados na literatura, ou podem propor novos métodos que melhor se adequem ao problema em questão. Existem vários métodos híbridos para solucionar o VRPTW utilizando outras heurísticas conhecidas em uma das etapas do GA [Tom03, Lim00].

#### 4.1.1 Função Objetivo

Geralmente a função objetivo consiste em maximizar a produção, garantir a qualidade programada no destino e minimizar a fila de carregamento e basculamento dos caminhões. Estes objetivos devem ser trabalhados de forma concomitante pelo GA e não de forma concorrente, pois se algum destes objetivos for focalizado pode-se não atingir uma forma de operação adequada sob o ponto de vista global. No entanto, esta modelagem permite que sejam definidas prioridades para cada um dos objetivos citados, tornando o resultado da otimização mais real. Isto significa que em certos momentos da operação, é necessário focalizar um dos objetivos de forma prioritária, para refletir as realidades da operação. Por exemplo, às vezes se faz necessário garantir totalmente a qualidade do material extraído, mesmo que isto implique em uma diminuição da produção ou aumento das filas. A prioridade pode ser “alta”, “média” ou “baixa” para cada um dos objetivos, que na verdade, é apenas um fator para avaliação dos indivíduos do GA.

#### 4.1.2 Representação dos indivíduos e cromossomos

Um indivíduo no GA significa uma possível solução para o problema em questão. Assim, no contexto das minerações, um indivíduo é composto por um conjunto de cromossomos, no qual cada cromossomo representa uma decisão de despacho para um caminhão. Assim, o indivíduo é representado por um vetor de decisões de despacho (rotas) de caminhões  $s$ , no qual  $s = d_1, d_2, \dots, d_n$ .

O significado de  $d_i$  dependerá do ponto onde, para uma dada simulação, houver a  $i$ -ésima necessidade de se optar por uma entre várias rotas disponíveis. Os valores possíveis para  $d_i$  são números entre 0 e 255. Cada byte corresponde a uma decisão de despacho. A rota escolhida dependerá do valor deste byte e o número de rotas disponíveis no ponto onde será realizada a escolha.

#### 4.1.3 População Inicial

O GA necessita de uma população inicial para começar o processo de seleção dos indivíduos para cruzamento. Nesta modelagem, a população inicial é gerada de forma totalmente randômica, para garantir a diversificação de indivíduos na primeira geração. Na literatura, a heurística PFIH proposta por [Des88], tem sido frequentemente utilizada.

#### 4.1.4 Avaliação dos indivíduos

A avaliação dos indivíduos consiste em calcular a adaptabilidade (*fitness*) de cada indivíduo de acordo com a função objetivo. Esta adaptabilidade é utilizada como principal característica no processo de seleção dos indivíduos. Neste trabalho, de acordo com a função objetivo descrita anteriormente, o processo de avaliação consiste em realizar uma simulação da mineração em questão, para a próxima hora. Neste processo, uma “foto” da situação atual da mina é obtida, ou seja, dados como tempo de ciclo dos caminhões, rotas, caminhões disponíveis, estado atual dos caminhões (por exemplo: carregando, deslocando cheio, basculando, manutenção etc), são obtidos. Então, através de uma matriz de simulação, todos os estados por caminhão são simulados para a próxima hora. Ao fim deste processo, cada indivíduo recebe uma “nota” *fitness*, de acordo com o tempo de fila, produção e qualidade obtidos para este indivíduo, levando em consideração as prioridades definidas pela função objetivo.

#### 4.1.5 Seleção

Na etapa de seleção, pares de indivíduos são selecionados para cruzamento. Na literatura, duas técnicas de seleção são amplamente utilizadas [Hol75]: *roulette wheel* e *tournament*. No primeiro método, a probabilidade de um indivíduo participar do cruzamento está diretamente relacionada com a sua adaptabilidade (*fitness*) relativa. Este método é bastante sensível à função de avaliação, e quase sempre alguns controles extras são necessários. No método torneio (*tournament*),  $k$  indivíduos são selecionados randomicamente. Aqueles que apresentarem maior adaptabilidade (*fitness*) são os escolhidos. Este processo é repetido até que o número necessário de indivíduos para cruzamento seja atingido. Este trabalho utiliza a técnica *tournament* e a quantidade  $k$  de indivíduos é um parâmetro configurável.

#### 4.1.6 Cruzamento

O cruzamento (*crossover*) é a operação mais importante do GA, onde informações de indivíduos diferentes são trocadas para se formar um novo indivíduo. Neste trabalho, a forma mais simples de cruzamento foi utilizada. De acordo com uma taxa de cruzamento (probabilidade), um ponto de “quebra” da *string* de bits que representa o indivíduo é selecionado randomicamente. Assim, a partir deste ponto, todas os cromossomos que estiverem antes do ponto de quebra serão trocados entre os dois indivíduos, formando um

novo indivíduo, ou seja, uma nova possível solução para o problema.

#### 4.1.7 Mutação

O processo de mutação é uma operação importante no GA, pois é capaz de inserir novas características não presentes em todos os indivíduos atuais, diversificando o espaço de busca, e evitando que a solução se concentre nos mínimos locais. No entanto, esta operação deve ser feita com cuidado para evitar a modificação do indivíduo a ponto de inutilizá-lo. Neste trabalho, a mutação é realizada de acordo com uma taxa de mutação (probabilidade), na qual alguns cromossomos de um indivíduo são selecionados de forma randômica e modificados. A modificação de um cromossomo, ou seja, de uma rota para um caminhão, é realizada de forma cuidadosa, para evitar gerar uma rota incoerente, resultando em um cromossomo “defeituoso”.

### 4.2 Fluxo de otimização

Para que o processo de otimização da operação nas mineração seja eficiente, os dados necessários para o GA devem refletir, o mais fiel possível, a situação em que se encontra a mina naquele instante. Desta maneira, a otimização é um processo cíclico, como pode ser observado na Figura 1.

O processo de otimização utiliza dados estatísticos, como tempos médios de ciclo dos caminhões, distâncias entre os pontos de carga e descarga, dados estáticos, como número de caminhões, pontos de carga disponíveis, e dados em tempo real, como posição real de cada caminhão, estado em que se encontra cada caminhão (operando, disponível, manutenção, abastecendo), produção até o momento etc. Com todas estas informações disponíveis, o GA é iniciado e sugere uma nova rota para cada caminhão de acordo com a função objetivo especificada.

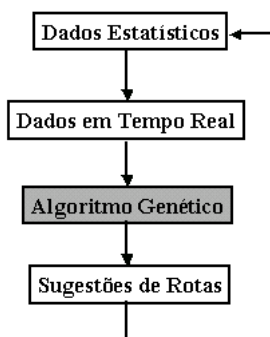


Figura 1: Fluxo do processo de otimização

Assim, a cada 10 segundos, o processo de otimização realiza uma nova simulação, refletindo todas as mudanças que podem ocorrer ao longo da operação da mina, melhorando cada vez mais a solução. Desta forma, os dados em tempo real são fundamentais para a tomada de decisão correta no instante adequado. Por exemplo, se um caminhão apresenta algum problema e não pode mais operar, um outro caminhão pode ser enviado instantaneamente para substituir este caminhão. Com a posição real de cada caminhão, pode-se trocá-lo de rota a qualquer instante, para evitar por exemplo, fila de carregamento em um dos pontos de carga que esteja com problemas de operação.

O processo de otimização deste trabalho, assume que todos os caminhões da mineração possuam um computador de bordo capaz de enviar informações via rede *wireless* e possuam um GPS (Global Positioning System), para informar posição geográfica real no mundo a cada instante.

## 5 Resultados

A maioria dos trabalhos encontrados na literatura, que trabalham com heurísticas para resolver o VRPTW, utilizam uma série de *benchmarks* propostos por [Des88] para comparar a eficiência de suas heurísticas, em relação aos resultados ótimos conhecidos até o momento para algumas instâncias desta classe de problemas.

Neste trabalho, estes *benchmarks* não foram utilizados para verificar a eficiência do GA, pois esta heurística foi totalmente adaptada para o contexto do problema de roteamento de veículos em minerações de minério de ferro. O processo de otimização aqui descrito, encontra-se em implantação na mineração “Córrego do Feijão” da Ferteço e na mineração “Capitão do Mato” da MBR (Minerações Brasileiras Reunidas), na presente data.

Em um ambiente real, pode-se avaliar de forma acertada a eficiência da otimização através do GA, medindo-se os ganhos de produtividade e qualidade nas minerações após a implantação do processo de otimização. Na situação atual, ainda não é possível medir os ganhos em termos de diminuição de custos e aumento de produção.

Apesar disso, pode-se observar que o modelo de operação destas minerações têm sofrido grandes modificações devido ao processo de otimização, pois a operação manual, realizada por um controlador de tráfego, não é capaz de lidar com a quantidade de variações que ocorrem durante a operação destas minerações, preju-

dicando o resultado final. O processo manual, é realizado com base apenas nos dados estatísticos, ou seja, no início da operação calcula-se a quantidade de caminhões necessários por rota para se atingir as metas de produção e qualidade, e estes caminhões então são fixados nestas rotas. Este modelo é incapaz de se adaptar as mudanças do ambiente. Um método de busca adaptativo como o GA, lida muito bem com este tipo de situação. No pior caso, se o GA não convergir para encontrar uma solução viável, a operação continua a ser como a operação manual, ou seja, fixa.

Apesar da medição dos resultados não ser uma tarefa trivial, alguns resultados preliminares foram obtidos, baseados em simulações. Os resultados a seguir foram obtidos a partir de simulações com o algoritmo genético e comparando com resultados de operação com caminhões fixos por rota, denominada frente fixa.

No gráfico da Figura 2, pode-se observar os resultados obtidos pelo GA na simulação para quatro caminhões e três rotas possíveis, sendo a prioridade da função objetivo a qualidade. Pode-se constatar que a qualidade desejada foi obtida logo nas primeiras gerações, enquanto ao tempo de fila a redução não foi imediata, e mesmo após trinta gerações, ela não foi completamente eliminada. Não se sabe se a situação de fila zero existe. Para verificar é necessário gerar todas as combinações possíveis, de custo proibitivo.

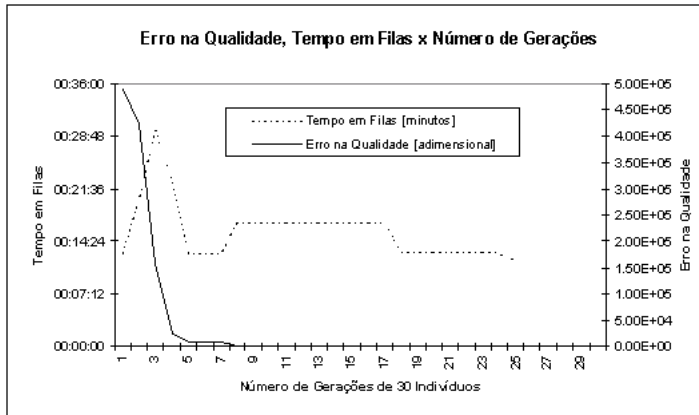


Figura 2: Evolução do tempo em fila e do erro na qualidade, com o avanço nas gerações do GA, para a operação da mina com 4 caminhões, sendo a prioridade a qualidade

No gráfico da Figura 3, os mesmos dados foram analisados, sendo que a prioridade da função objetivo foi o fator fila. Pode-se notar que ocorreu uma diminuição do tempo de fila, o que sacrificou um pouco o termo qualidade, em relação ao gráfico da Figura 2. O termo produção se manteve praticamente constante, mas

com um valor aceitável.

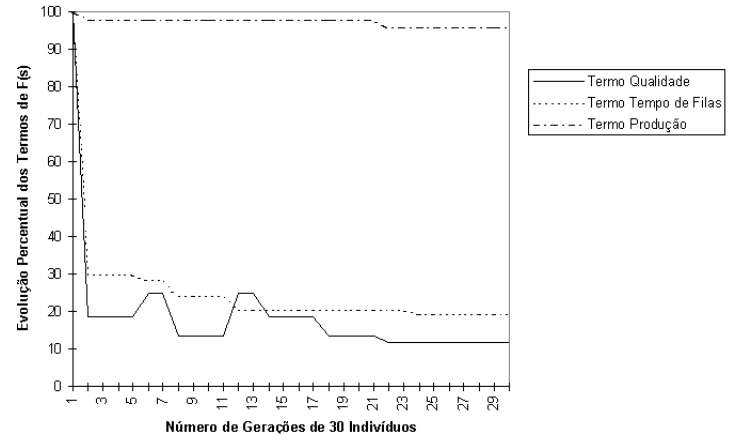


Figura 3: Redução percentual de cada termo da função objetivo com prioridade para fila.

No gráfico da Figura 4, a prioridade da função objetivo foi o termo produção. Pode-se perceber que para se conseguir ganhos de produção foi necessário aumentar o tempo de fila e piorar o fator qualidade.

Apesar de cada fator de prioridade implicar em relativa piora para os termos não priorizados, ocorreu uma redução dos termos em todos os testes realizados em relação à frente fixa. Logo nas primeiras gerações, esta diminuição pode ser observada. Isto comprova que até mesmo um resultado próximo do aleatório obtido pelo GA (fato ocorrido nas primeiras gerações, onde os indivíduos ainda não evoluíram o suficiente), provou ser melhor do que a operação frente fixa.

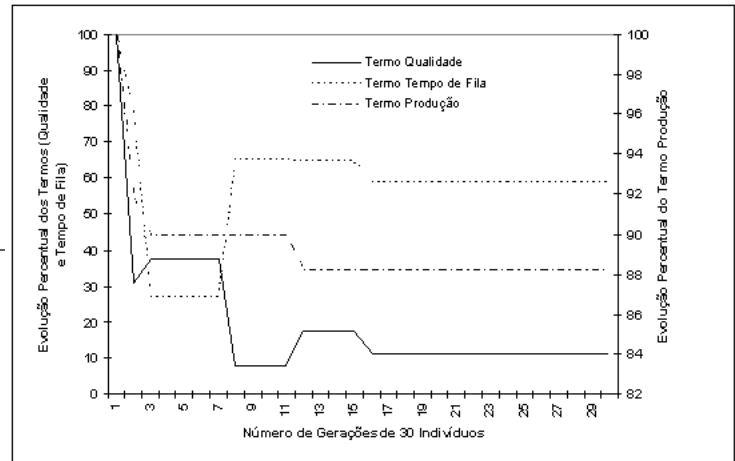


Figura 4: Redução percentual de cada termo da função objetivo com prioridade para produção.

Outro fato que pode ser comprovado, foi que aumentando-se o número de caminhões e o número de

rotas, o tempo de resposta do algoritmo piora bastante, devido ao aumento exponencial de possibilidades de solução a serem analisados pelo GA. No entanto, para um problema real com cinquenta caminhões, o tempo de resposta do algoritmo ficou em torno de dez segundos, com um total de trinta gerações.

## 6 Conclusão

O VRPTW é um problema da classe *NP – Completo* amplamente estudado na literatura. Este trabalho apresentou os principais métodos utilizados para resolver problemas computacionais de custo proibitivo. A tendência atual para resolução deste tipo de problema, tem se concentrado na modelagem de heurísticas híbridas, utilizando uma ou mais heurísticas conhecidas de forma conjunta, como pode ser visto em [Tom03, Lim00]. Além disso, métodos paralelos para resolver o VRPTW também têm sido utilizados [Lar99], bem como a utilização de heurísticas distribuídas [Alv97].

A escolha do GA para modelar o problema real de roteamento de veículos em minerações, foi devida à grande utilização desta heurística na literatura para resolver problemas da classe VRP e suas variações. Na prática, pode-se observar que o GA é bastante versátil, permitindo uma implementação facilitada e o aperfeiçoamento contínuo do método. Por exemplo, se for necessário modificar a operação de mutação acrescentando mais operadores, basta alterar apenas esta parte do algoritmo, sem modificar a representação dos indivíduos, cruzamento etc.

O GA apresentou bons resultados nas simulações realizadas, mas ainda não podemos avaliar os benefícios que este método pode trazer para as minerações estudadas. Espera-se um aumento de produtividade de 5% a 15% em relação à produtividade obtida atualmente, ou seja, com operação manual.

Pode-se perceber que o problema de roteamento de veículos nas minerações é um problema muito específico, pois apresenta muitas restrições a mais, do que o VRTPW padrão. Questões como qualidade, capacidade de combustível, formato das rotas (difícil acesso por exemplo) etc, devem ser consideradas além da capacidade do veículo e a janela de tempo para que este cumpra a rota. Assim, pode-se perceber uma grande quantidade de trabalhos futuros a serem realizados para otimização de frota de veículos em minerações.

Este trabalho não abordou a questão da capacidade de combustível do equipamento, o que pode ser um

fator decisivo para se conseguir atingir os objetivos da operação. Por exemplo, a decisão de enviar um caminhão para abastecimento pode evitar uma sobrecarga em um ponto de carga, diminuindo o tempo de fila para todos os equipamentos.

Este trabalho também não aborda a questão de trocar o caminhão de rota quando este estiver deslocando. A troca de rota é feita apenas nos pontos de carga e descarga. Com a posição real do equipamento via GPS, pode-se decidir trocar o caminhão de rota, mesmo que este já esteja chegando em algum ponto de carga, para refletir as mudanças de objetivo da operação.

Enfim, existem muitas características a serem acrescentadas no modelo proposto. Além disso, convém implementar e testar outras heurísticas para resolver o VRPTW em minerações, pois existem poucos trabalhos na literatura que abordam este problema.

## Referências

- [Alv97] Guilherme Bastos Alvarenga. Despacho ótimo de caminhões numa mineração de ferro utilizando o algoritmo genético com processamento paralelo. Master's thesis, Universidade Federal de Minas Gerais, Departamento de Engenharia Elétrica, 1997.
- [Dar59] Charles Darwin. *On The Origin of Species*. 1st edition, Harvard University Press, MA, 1859.
- [Des88] M. M. Solomon; J. Desrosiers. Time window constrained routing and scheduling problems: A survey. 1988.
- [Hol75] J.H. Holland. *Adaptation in natural and artificial systems*. 1975.
- [Joh79] R. Michael Garey; David Johnson. *Computers and intractability: A guide to the theory of  $np$  – completeness*. 1979.
- [Jue91] Sam R. Thangiah; Kendall Nygard; Paul L. Juell. Gideon: A genetic algorithm system for vehicle routing with time windows. 1991.
- [Kaa87] A.W.J.Kolen; A.H.G.Rinnooy Kan; H.W.J.M.Trienekens. Vehicle routing with time windows. 1987.
- [Lar99] J. Larsen. *Paralellization of the Vehicle Routing Problem with Time Windows*. PhD thesis, Department of Mathematical Modelling, Technical University of Denmark, 1999.

- [Lau94] Per S. Laursen. General optimization heuristics: an introduction. 1994.
- [Lim00] Ang Juay Chin; Ho Wee Kit; Andrew Lim. A new ga approach for the vehicle routing problem. 2000.
- [Nyg93] Sam R. Thangiah; Kendall E. Nygard. Dynamic trajectory routing using an adaptive search method. 1993.
- [Qil] Kenny Zhu Qily. Heuristic methods for vehicle routing problem with time windows. Department of Electrical Engineering, National University of Singapore, 1999.
- [Sch86] Edward K. Baker; Joanne R. Schaer. Solution improvement heuristics for the vehicle routing and scheduling problem with time window constraints. 1986.
- [Sol86] Marius M. Solomon. On the worst-case performance of some heuristics for the vehicle routing and scheduling problem with time window constraints. 1986.
- [Sun94] Sam R. Thangiah; Ibrahim H. Osman; Tong Sun. Hybrid genetic algorithm, simulated annealing and tabu search methods for vehicle routing problems with time windows. 1994.
- [Tai97] E. Taillard; P. Badeau; M. Gendreau; F. Geurtin. A tabu search heuristic for the vehicle routing problem with time windows. 1997.
- [Tom03] G.B. Alvarenga; G.R. Mateus; G. de Tomi. Finding near optimal solutions for vehicle routing problems with time windows using hybrid genetic algorithm. 2003.
- [uSF99] J. urgen Schulze; Torsten Fahle. A parallel algorithm for the vehicle routing problem with time window constraints. 1999.
- [Via98] Valdisio Viana. Meta-heurísticas em programação paralela e otimização combinatória. 1998.