

## 5 Estudo de Caso

### 5.1 Planejamento de um Porto Fictício

Para o estudo de caso foi necessário implementar uma ferramenta capaz de otimizar o uso dos recursos de um porto fictício porém, semelhante ao descrito no capítulo 2 (seção 2.2), utilizando um dos modelos de representação com restrições de precedência (algoritmo genético ou co-evolucionário) descritos anteriormente no capítulo 4. A idéia é que o modelo de representação proposto seja capaz de selecionar o píer para a atracação dos navios, selecionar as áreas onde o minério para cada navio será estocado, selecionar quais viradores, correias e empilhadeiras serão utilizados para descarregar o minério e finalmente, quais recuperadoras, correias e carregadores de navios serão utilizados para embarcar o minério. Além disso, ele deve ser capaz de encontrar o melhor momento para realizar cada uma destas operações de modo a atender os objetivos desejados. Nesse estudo de caso, considerou-se um único objetivo, o de minimizar o tempo de espera dos navios ou, em outras palavras, minimizar a multa (ou maximizar o prêmio) por sobrestadia do navio no porto.

As restrições de precedência se fazem necessárias, conforme já foi dito no capítulo 2, devido a uma série de fatores:

- Certas áreas de estocagem podem não ter ligação com certos píeres através de correias transportadoras. Deste modo, é necessário planejar primeiro a atracação do navio para depois se planejar onde o minério será estocado;
- Do mesmo modo, certos viradores podem não ter ligação com certas áreas através de correias transportadoras. Assim, é necessário planejar primeiro a estocagem do minério para depois se planejar através de qual virador o minério será descarregado;
- Só é possível saber o momento em que o navio irá desatracar após ter-se planejado a chegada do minério no porto e o embarque do mesmo no navio;
- Finalmente, dado que deve-se respeitar a fila de navios e o tamanho dos mesmos na hora de planejar a atracação, só é possível saber quando um

navio irá atracar depois que um navio que tenha preferência sobre ele em um píer tiver sua desatracação planejada.

Além das restrições de precedência, a simulação deve levar em conta uma série de regras para a construção do planejamento. Estas regras são listadas abaixo:

- Todos os cálculos relacionados a minério são feitos com base na densidade dos mesmos. Entre outros cálculos que devem considerar a densidade, pode-se citar a quantidade de minério que pode ser transportada por um lote e a quantidade de minério que cabe entre duas balizas de uma área;
- Os trens têm capacidade para transportar um lote com 20.000 toneladas de minério. Devido a isso, em geral, as cargas dos navios só são completadas com várias viagens de trens;
- O trem leva 12 horas para viajar entre a mina e o porto;
- As áreas do porto são divididas em 50 balizas. Entre duas balizas cabem 20.000 toneladas de minério (ou seja, um lote) de minério com densidade 1;
- Caso um navio esteja atracado em um píer que não tenha meios de receber minério de uma determinada área por ausência de uma rota para este transporte, esta área deve ser descartada na hora de selecionar em que área o minério será estocado;
- Todos os equipamentos têm uma taxa de operação própria especificada em toneladas por hora. Como o minério é transportado por diversos equipamentos acoplados (por exemplo, do virador para uma correia, da correia para a empilhadeira, etc) e estes diversos equipamentos podem ter taxas de operações diferentes, a menor taxa de operação é a que conta para fazer a transferência;
- Não são consideradas paradas para manutenção de equipamentos, apesar de ser possível indisponibilizá-los para a simulação;
- As minas têm uma capacidade limitada de produção de minério. Deste modo, para cada tipo de minério, especifica-se uma taxa de produção diária. Esta taxa deve ser sempre respeitada e portanto o otimizador deve

antecipar a solicitação de minério para vários dias antes da previsão de atracação do navio no porto.

O algoritmo completo para a otimização é mostrado na figura abaixo:

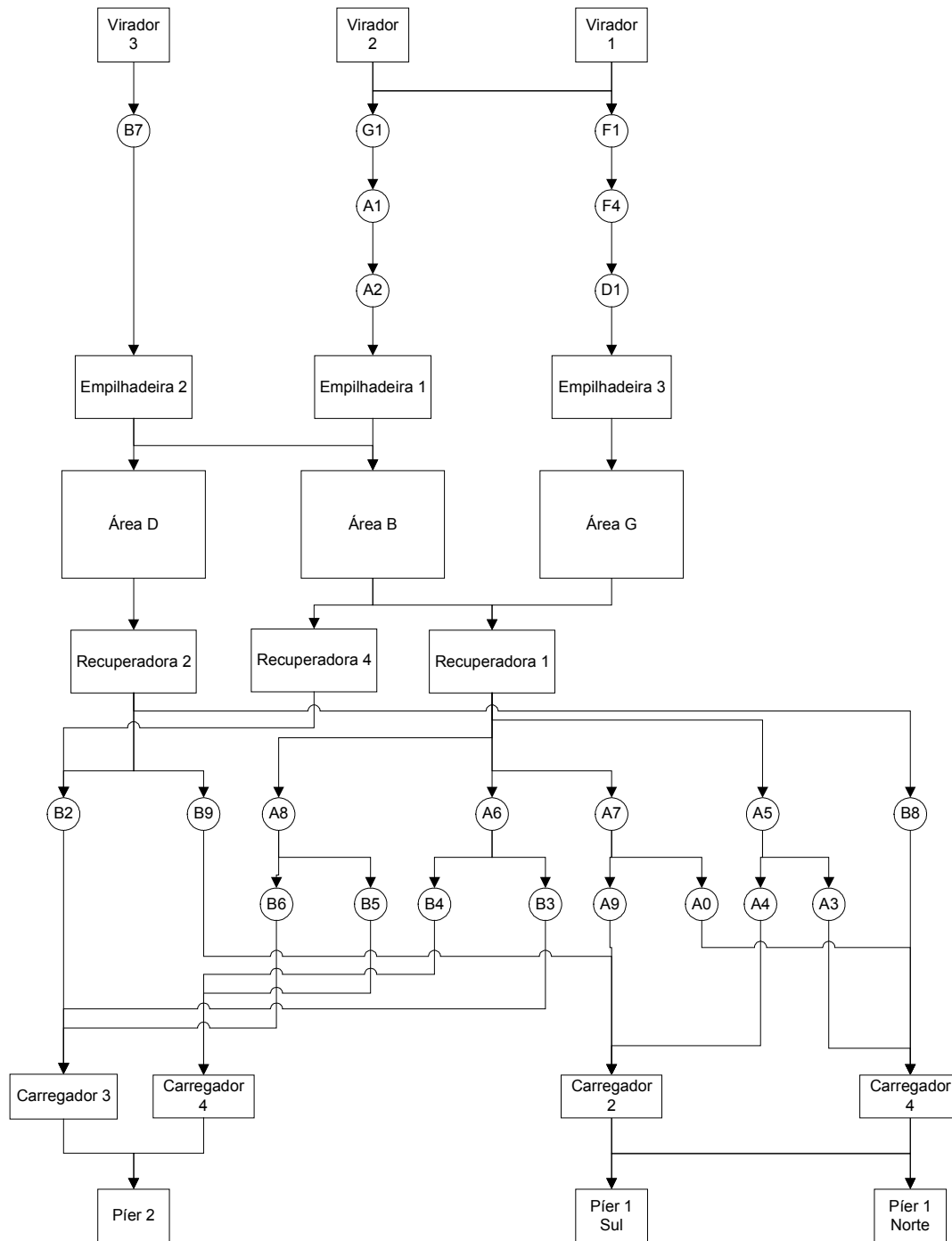
```

procedure otimização
  input população inicial P
  geração ← 0
  while (geração <> total_geração)
    for each indivíduo Si em P
      repeat
        seleciona tarefa J a planejar baseado no grafo de
          precedências
        baseado nas regras que devem ser respeitadas e no
          que já foi planejado selecionar a data
          de início e fim da tarefa
        planejar a tarefa
      until todas as tarefas planejadas
      calcular a multa
    end for each
    seleção
    crossover e mutação
  end while
end procedure

```

**Figura 17 – Algoritmo completo de otimização**

Para fins de simplificação de criação da base de dados (o porto de Vitória conta com mais de 400 diferentes rotas ligando diversos equipamentos e áreas) e devido a necessidade de se preservar sigilo sobre algumas informações da empresa, optou-se por simular um porto simplificado, mas que contemplasse todos os modelos de equipamento do porto original. Deste modo, criou-se um porto cujo diagrama é mostrado na figura a seguir:



**Figura 18 – Diagrama do porto fictício**

Nesta figura, cada correia transportadora é definida por um círculo. As setas identificam as conexões entre equipamentos (ou no caso dos carregadores e píeres ou das áreas e empilhadeiras e recuperadoras, indica quais equipamentos atendem que píeres ou áreas).

Além disso, algumas simplificações foram feitas em termos de heurística: Considera-se que os trens transportam apenas um lote de minério. Não se leva em conta também a questão de planos de carga no momento de embarcar o minério (quantidade de minério que deve ser jogada em cada porão e a ordem no qual a operação deve ser feita), ou seja, o sistema entende que o navio irá suportar qualquer tipo de manobra de embarque de minério. O sistema também não se preocupa com o limite operacional de calado aéreo e informações sobre maré (já que para tratar destas informações seria necessário dispor de uma tábua de marés).

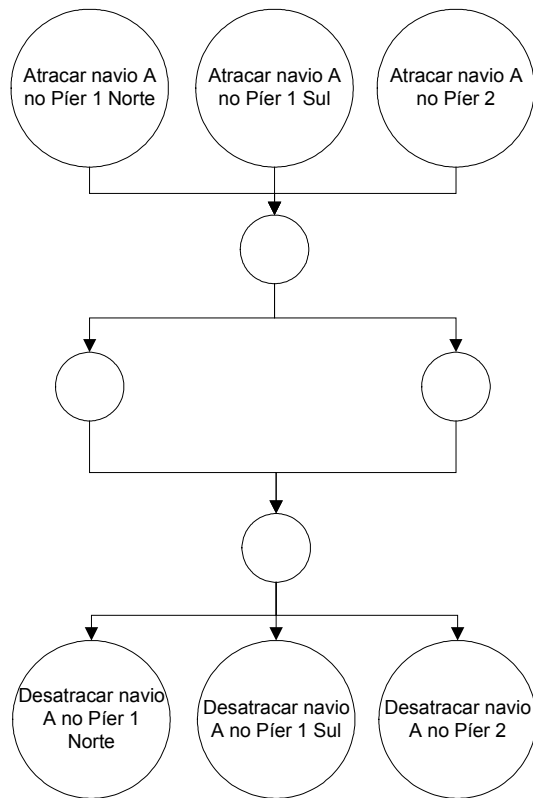
No início da execução, o sistema calcula quantos lotes serão necessários para atender o navio. Este cálculo é feito baseado na capacidade de produção das minas e na densidade do minério para se saber quantas toneladas de cada tipo de minério cabem em um lote. Após gerar esses números, o sistema gera o grafo de precedências. A geração deste grafo segue os seguintes passos:

1. Verificar o tipo de navio e em quais píeres este navio pode atracar;
2. Inserir no grafo uma tarefa de atracação e desatracação para cada píer onde é possível o navio atracar;
3. Inserir no grafo uma tarefa de criação de pilha para cada carga a ser embarcada;
4. Inserir no grafo uma tarefa para descarga de lote para cada lote criado inicialmente, durante o cálculo do número necessário de trens para transportar a carga até o porto;
5. Inserir no grafo uma tarefa de embarque de minério para cada carga;
6. Gerar as ligações do grafo para as atracações e desatracações. Esta geração segue as seguintes regras:
  - a. Considerando-se dois navios que, devido ao tamanho só podem atracar no píer 2 ou dois navios que podem atracar em qualquer píer, gerar uma ligação entre a desatracação do navio que chega antes e a atracação do navio que chega depois;
  - b. Considerando-se dois navios de tamanhos diferentes, sendo que um navio só pode atracar no píer 2 e o outro pode atracar tanto no píer 1 norte e no píer 2, a restrição será criada apenas entre as atracações e desatracações nos píeres em comum. Neste caso, o navio maior terá precedência sobre o navio menor, ou seja,

considerando-se o exemplo acima, cria-se uma ligação entre a desatracação do píer 2 do navio maior e a atracação no píer 2 do navio menor;

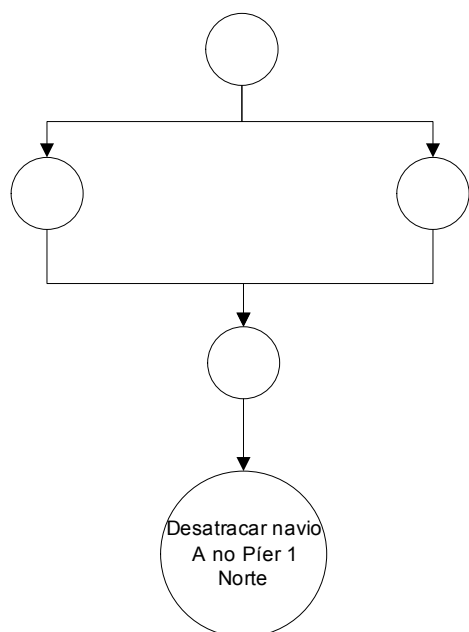
7. Gerar as ligações do grafo entre as atracações dos navios e a criação de pilhas para as suas respectivas cargas, ou seja, as atracações devem ser planejadas antes da criação das pilhas;
8. Gerar as ligações do grafo entre a criação das pilhas e a descarga dos lotes para cada uma das cargas;
9. Gerar as ligações do grafo entre a descarga dos lotes e o embarque do minério no navio;
10. Gerar as ligações do grafo entre o embarque do minério no navio e a desatracação do navio.

É importante notar que para os navios menores, pode-se ter duas ou até três tarefas de atracação e desatracação diferentes (uma para cada píer), contempladas no grafo, considerando-se que estes navios podem utilizar qualquer um dos 3 píeres. A seleção de qual delas será executada é feita pelo cromossoma, utilizando-se a lista de prioridades. Todavia, ao se planejar qualquer uma delas, todas as outras são eliminadas do grafo. Em outras palavras, supondo um navio que possa atracar em qualquer píer, deve-se que criar uma tarefa de atracação e desatracação para este mesmo navio em cada um dos píeres. Após o planejamento da atracação no píer referente a tarefa de maior prioridade no cromossoma, a tarefa deve ser eliminada do grafo bem como as restrições de precedência pelas quais ela é responsável. Feito isso, todas as outras tarefas de atracação e desatracação relativas a este navio, mas em píeres diferentes, são eliminadas assim como as precedências pelas quais elas são responsáveis. Este processo é mostrado em detalhes no exemplo abaixo. Na figura 19 tem-se o grafo antes da execução de uma das tarefas (os itens sem texto correspondem a outras tarefas que não têm relação com atracação ou desatracação):



**Figura 19 – Exemplo de grafo com um navio que pode atracar em diversos píeres**

Após a seleção de uma das tarefas através das prioridades definidas no cromossoma (supor por exemplo que a tarefa de atracação no píer 1 norte seja a escolhida), remove-se as outras tarefas como já foi explicado e o grafo passa a ter a seguinte configuração (figura 20):



**Figura 20 – Grafo com navio que pode atracar em diversos píeres após atracação**

Ou seja, todas as tarefas de atracação referentes àquele navio foram removidas e todas as tarefas de desatracação que não ocorrem no píer selecionado também foram removidas do grafo.

Todas as informações necessárias para o funcionamento do sistema são fornecidas através de um arquivo texto similar aos arquivos de configuração do Windows. Este arquivo é dividido em seções (o nome das seções é colocado entre colchetes). Todas as seções são obrigatórias e são descritas na tabela 5:



Seção	Descrição
Global	Contém informações como o nome do porto, a data corrente a ser considerada pelo sistema e a quantidade de minério de densidade 1 que um lote (80 vagões) pode transportar.
GA	Contém informações para a execução do Algoritmo Genético como a taxa de <i>steady state</i> , taxas de crossover e mutação, número de rodadas, de gerações e indivíduos entre outras.
Pier	Contém uma lista com o nome de cada pier e o limite operacional de calado e tamanho do navio.
Área	Contém um identificador para cada área, a distância entre as balizas, o número de balizas e a quantidade de minério de densidade 1 que pode ser estocada entre duas balizas.
Virador	Contém o nome de cada virador e sua taxa de operação em toneladas/hora.
Empilhadeira	Contém o nome de cada empilhadeira e sua taxa de operação em toneladas/hora.
Recuperadora	Contém o nome de cada recuperadora e sua taxa de operação em toneladas/hora.
Carregador	Contém o nome de cada carregador de navio e sua taxa de operação em toneladas hora.
Correia	Contém o nome de cada correia transportadora e sua capacidade de operação em toneladas/hora.
Rota	Contém o nome e a seqüência de equipamentos que forma cada rota. Esta seqüência é construída utilizando-se os identificadores (nomes) de cada equipamento declarado.
Carga	Contém informações sobre as cargas que cada navio vai levar (é importante notar que um navio pode levar uma ou mais cargas de diferentes minérios). Nestas informações constam o minério a ser transportado, o navio para o qual é destinado, a quantidade de minério, o <i>laytime</i> e a multa por hora excedente (o prêmio é calculado como metade do valor da multa).
Navio	Contém informações sobre a fila de navios. Entre outras informações contém o nome do navio, o tamanho, o calado, a data e a hora de chegada (estimadas) e caso o navio já esteja atracado, o pier no qual ele está.
Minério	Contém informações sobre os tipos de minério. Além do nome, informa a taxa de produção diária do minério e a densidade do mesmo.

**Tabela 5 – Seções do arquivo de configuração do sistema**

Um exemplo de um trecho deste tipo de arquivo é mostrado na figura a seguir:

```
[Global]
Porto=Teste
Data=14/05/2001
TonLote=20000
TempoLote=12

[GA]
SteadyState1=0.10
PMX1=1.0
OX1=0.0
CX1=0.0
PIM1=1.0
SWAPM1=0.0
RLM1=0.0
RRM1=0.0
Crossover1=0.8
Mutation1=0.2
Cycles1=1
Rounds1=5
Generations1=30
Genomes1=40

[Pier]
Total=3

Nome1=P1N
LimiteOperacional1=180000
Calado1=15.2

Nome2=P1S
LimiteOperacional2=100000
Calado2=13.0

Nome3=P2
LimiteOperacional3=350000
Calado3=20.0
```

**Figura 21 – Trecho do arquivo de configuração**

Caso as taxas de *crossover* e *mutação* sejam iguais a zero, o sistema irá executar uma busca aleatória. Esta configuração é permitida para efeitos de comparação do algoritmo genético com a busca aleatória.

## 5.2 Resultados

Foram realizados testes com o Algoritmo Genético com uma espécie e com Algoritmos Co-Evolucionários Cooperativos (duas espécies) além de buscas

exaustivas com representações da solução correspondentes àquelas empregadas pelos algoritmos. Inicialmente, utilizando-se uma fila de 35 navios, realizou-se um teste apenas com o Algoritmo Genético de uma espécie e a busca aleatória correspondente. Todos os demais testes empregaram uma fila de apenas 20 navios de modo a reduzir o tempo de processamento.

Para uma fila de 35 navios utilizando-se o algoritmo genético convencional (isto é, utilizando uma heurística fixa para seleção dos recursos), a melhor combinação de operadores encontrada foi empregando o operador de crossover de mapeamento parcial (PMX), o operador de ciclo (CX) e os operadores de mutação de inversão de posição (PI) e troca (*Swap*). Foram utilizadas taxas de crossover e mutação interpoladas durante a evolução e seleção de operadores por roleta. Para o crossover foram utilizadas taxas entre 100% na primeira geração e 50% na última com 50% de chances de seleção do operador. Para a mutação foram utilizadas taxas entre 10% na primeira geração e 50% na última com 50% de chances de seleção do operador. Foi utilizado *steady state* com *gap* de 20% e foi feita uma média de 5 rodadas com 40 gerações cada uma. Cada geração tinha uma população de 100 indivíduos. O gráfico a seguir mostra os resultados comparativos entre o algoritmo genético e a busca aleatória:

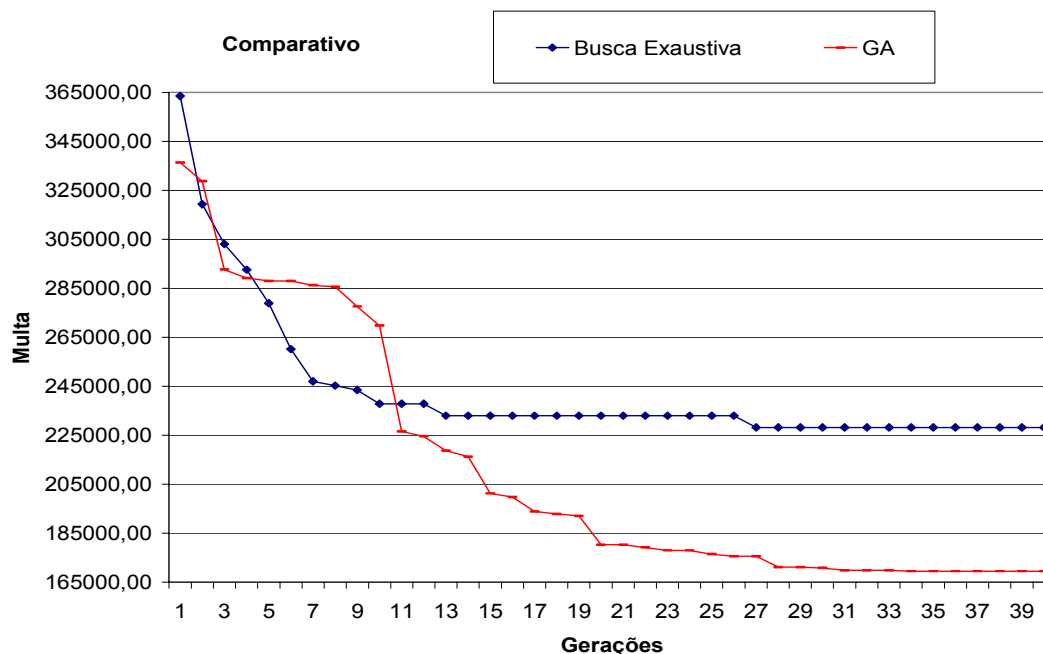


Figura 22 – Resultados com o algoritmo genético com uma espécie e fila de 35 navios

O Algoritmo Genético, após 40 gerações obteve um resultado 25,75% melhor do que a busca aleatória.

Para uma outra configuração, agora com 20 navios, e os mesmos parâmetros para os operadores genéticos, o resultado obtido é mostrado a seguir:

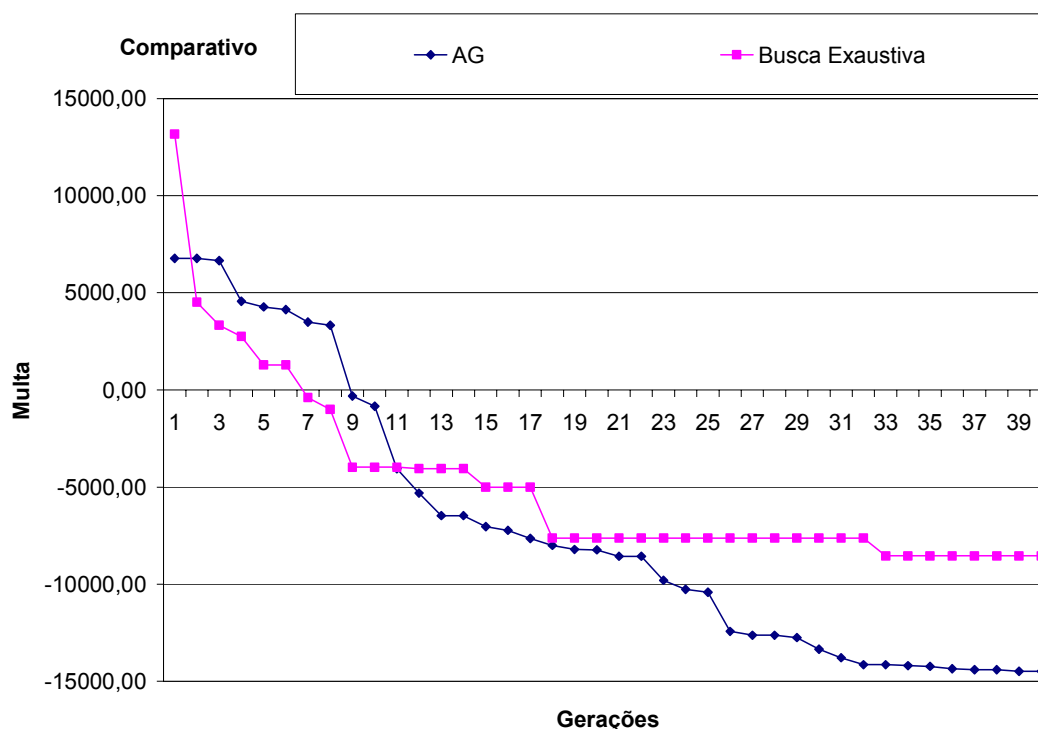


Figura 23 – Resultados com algoritmo genético com uma espécie e fila de 20 navios

Neste caso, o Algoritmo Genético, após 40 gerações obteve um resultado 41% melhor do que a busca aleatória. Esta diferença de resultado (de 25,75% para 41%), com relação ao que foi obtido no experimento anterior se deve ao menor número de navios e ao fato de que os navios que foram retirados continham cargas que demandavam poucas variedades de minério, o que concentrava a produção e saturava a capacidade da mina. Com a capacidade de produção na mina saturada, o algoritmo não tem muitas opções de otimização para estes navios e a multa aumenta consideravelmente.

Os mesmos testes foram realizados utilizando-se o modelo co-evolucionário. Para as duas espécies foram usados os mesmos valores de configuração: *gap* de 10%, taxa de crossover de 80% e de mutação de 5% com 50% de chances para cada operador utilizado (PMX, CX, inversão de posição e

swap). Foram feitas experiências diferentes com relação a avaliação entre os indivíduos escolhidos da população de cada uma das duas espécies:

- Indivíduo Aleatório: cada indivíduo da população de uma espécie é avaliado conjuntamente com um indivíduo aleatório da população da outra espécie;
- Melhor Indivíduo: cada indivíduo de uma população é avaliado conjuntamente com o indivíduo da outra população que teve a melhor avaliação na geração anterior;
- Misto: são feitas as duas avaliações descritas acima e a que tiver o melhor resultado é a escolhida.

O gráfico com os resultados é mostrado abaixo:

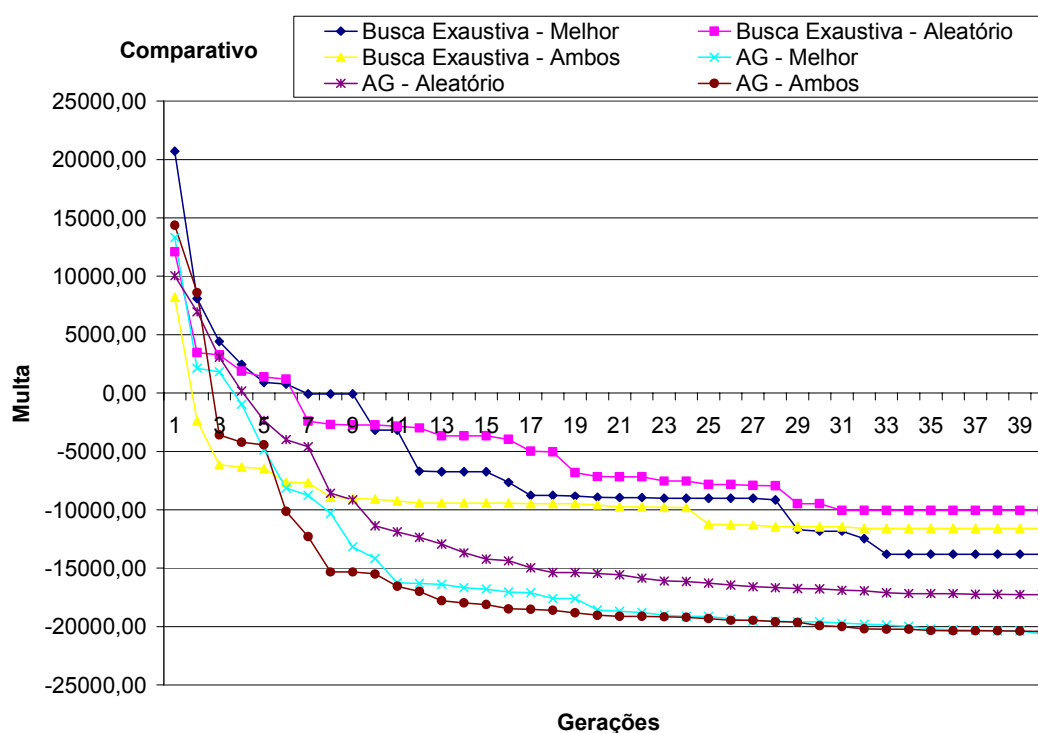


Figura 24 – Comparação dos resultados da busca aleatória e da co-evolução

Os melhores resultados com a co-evolução foram em média 33% melhores do que a busca aleatória empregando a mesma representação (populações de tarefas e recursos) e tratamento de solução (seleção de indivíduos de cada população) usados no modelo co-evolucionário. Além disso, os resultados foram

em média 29% melhores do que o AG com apenas uma espécie. As médias dos melhores resultados de cada experimento para o cenário de 20 navios são mostradas para efeito de comparação na tabela a seguir (o melhor resultado está destacado): Os resultados representam valores de multas e portanto, quando negativo, quanto maior o seu valor absoluto, maior o prêmio.

Tipo	Resultado	% em relação ao melhor
Busca aleatória – Uma Espécie	-8542,64	0,41
Algoritmo Genético – Uma Espécie	-14485,74	0,70
Busca aleatória – Aleatório	-10051,00	0,48
Busca aleatória – Melhor	-13806,11	0,66
Busca aleatória – Ambos	-11586,88	0,56
Co-Evolução – Aleatório	-17277,71	0,83
<b>Co-Evolução – Melhor</b>	<b>-20628,67</b>	<b>1,0</b>
Co-Evolução – Ambos	-20433,34	0,99

**Tabela 6 – Comparação percentual dos resultados**

Na figura 25, pode-se ver um trecho de um arquivo de saída do sistema contendo o planejamento gerado para um cenário semelhante a um cenário observado no porto real no ano de 2001:

```

14/5/2001 19:57:53 - 14/5/2001 19:57:53 - Chegada no porto de lote de minério ES04
(22 - 22222.2222222222 ton) para o navio AAA007
14/5/2001 19:57:53 - 14/5/2001 23:40:07 - Descarga de 22222.2222222222 do lote de
minério ES04 (22) para o navio AAA007 usando a rota R003
14/5/2001 22:05:00 - 14/5/2001 23:05:00 - Atracacao do navio AAA004 no pier P1S
14/5/2001 22:05:00 - 15/5/2001 07:35:00 - Encerramento de pilha para o navio
AAA004 na área D entre as balizas 4 e 6 (RM20) usando a rota R019
15/5/2001 02:00:00 - 15/5/2001 02:47:46 - Descarga de 4777.77777777778 do lote de
minério ES04 (23) para o navio AAA007 usando a rota R004
15/5/2001 04:08:34 - 15/5/2001 04:08:34 - Abertura de pilha para o navio AAA010 na
área G entre as balizas 1 e 1 (SLBF - 30000 ton)
15/5/2001 04:08:34 - 15/5/2001 04:08:34 - Requisição do minério SLBF (31 - 30000
ton) para o navio AAA010
15/5/2001 07:30:00 - 15/5/2001 07:30:00 - Chegada no porto de lote de minério PF
(25 - 20000 ton) para o navio AAA008
15/5/2001 07:35:00 - 15/5/2001 08:35:00 - Desatracacao do navio AAA004 do pier P1S

```

**Figura 25 – Trecho do arquivo de saída contendo o planejamento**

O tempo de execução do sistema para cada um dos modelos, em um computador AMD Athlon XP 2200+ com 512 Mb de memória, é mostrado na tabela abaixo:

Teste	Tempo
Algoritmo Genético – 20 Navios	10 minutos
Algoritmo Genético – 35 Navios	45 minutos
Busca aleatória – AG – 20 Navios	20 minutos
Busca aleatória – AG – 35 Navios	2h20min
Co-Evolução – 20 Navios	20 minutos
Busca aleatória – Co-Evolução – 20 Navios	30 minutos

**Tabela 7 – Tempo de execução dos diversos modelos**

Nota-se portanto a grande melhoria de desempenho que se obtém ao usar o algoritmo genético ou a co-evolução no lugar da busca aleatória. Entretanto, o tempo de execução cresce consideravelmente para um número de navios elevado já que, para cada navio, deve-se acrescentar diversas tarefas de descarga de lotes e embarque de minério, o que sugere, de início, a necessidade de otimização de código na função de avaliação dos dois modelos.

Até a décima geração a busca aleatória e o algoritmo genético convergem aproximadamente na mesma velocidade. Mas a partir daí o algoritmo genético se mostra mais eficaz na solução do problema, convergindo mais rapidamente para o ponto ótimo do que a busca aleatória.

Com relação a co-evolução, deve-se notar que o uso da comparação entre espécies de forma aleatória não produz bons resultados. No caso em que se usou ambos os tipos de comparação (melhor e aleatório) o resultado foi semelhante ao obtido usando apenas a comparação com o melhor.

Além disso, deve-se salientar que, em cada uma das execuções o algoritmo genético buscou de maneira ótima/sub-ótima dentre 20.000 possíveis soluções, todas essas soluções viáveis, além de não ter sido necessário penalizá-las ou corrigi-las. Em contraste com o atual modo de solução do problema no porto de Vitória, por exemplo, que emprega planilhas, a solução por Algoritmos Genéticos avalia centenas de vezes mais soluções em um tempo extremamente menor.

Com relação aos operadores genéticos, foi de extrema importância fazer uma avaliação cautelosa sobre os efeitos dos mesmos na evolução. Foi dado o mesmo peso aos operadores de mutação por haver uma necessidade de se balancear algumas trocas cautelosas de genes (com o operador de *swap*) com algumas trocas mais agressivas (com o operador de inversão de posição). A mutação mais agressiva se fez necessária em alguns momentos devido à extensão do cromossoma, que em alguns experimentos chega a ter mais de 500 posições.

É interessante notar que, ao contrário do que se costuma observar, os operadores de crossover que conseguiram melhor desempenho são os que geralmente menos se destacam em problemas de otimização de planejamento. De acordo com Michalewicz (1996), em testes de otimização de planejamentos com 195 tarefas o operador de ordem se saiu melhor e os operadores de ciclo e mapeamento parcial foram os piores. Nos testes realizados com cada operador de crossover, sendo usado individualmente com uma taxa de 100% e *gap* de 20%, os melhores resultados foram obtidos justamente pelos operadores de ciclo e de mapeamento parcial sendo que o operador de ordem foi o que se saiu pior, conforme se pode observar na figura a seguir:

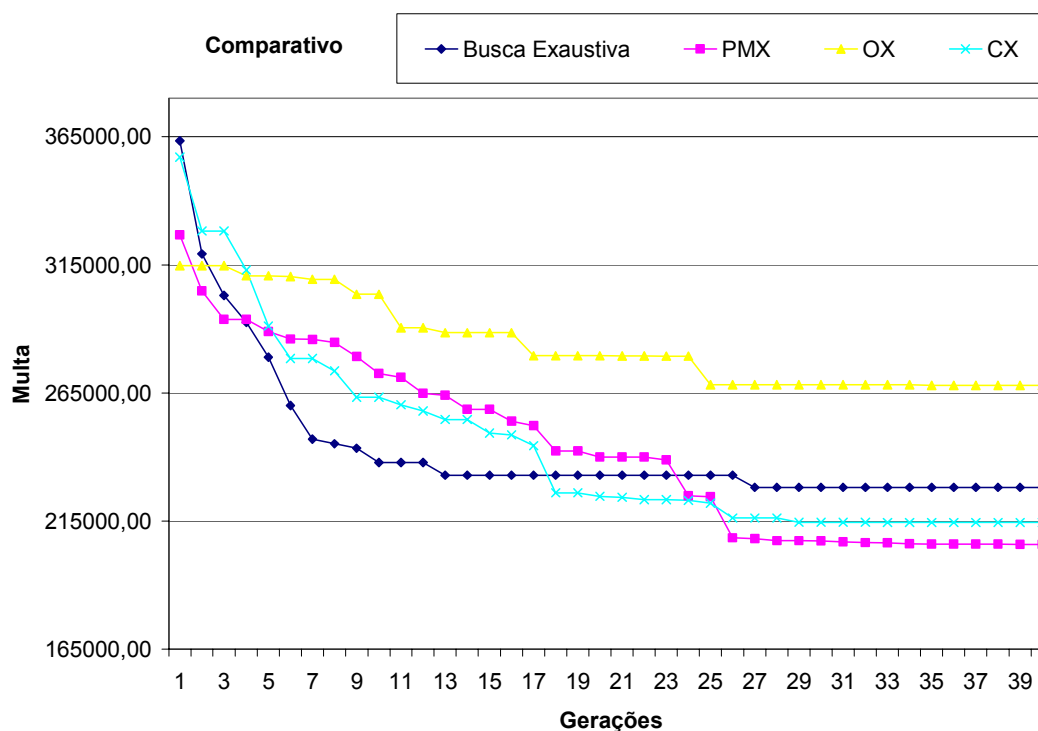


Figura 26 – Comparação dos resultados obtidos



Pode-se notar que mesmo utilizando apenas os operadores de crossover de ciclo e de mapeamento parcial o algoritmo genético se saiu melhor do que a busca aleatória. O mesmo não aconteceu com o operador de ordem.

Com isto, pode-se observar a importância da técnica empregada como um todo: geração de soluções viáveis, respeitando as regras de operação do porto e a necessidade de se planejar as tarefas em ordem, melhoria de até 41% em relação a busca aleatória no caso do algoritmo genético e avaliação de diversas soluções contra poucas soluções que podem ser avaliadas de forma manual.