

Universidade Federal do Maranhão
Centro de Ciências Exatas e Tecnologia
Curso de Pós-Graduação em Engenharia de Eletricidade

VICTOR HUGO BARROS

**PROBLEMA DE ALOCAÇÃO DE
BERÇOS EM PORTOS
GRANELEIROS COM RESTRIÇÕES
DE ESTOQUE E CONDIÇÕES
FAVORÁVEIS DE MARÉ**

São Luís
2010

VICTOR HUGO BARROS

**PROBLEMA DE ALOCAÇÃO DE
BERÇOS EM PORTOS
GRANELEIROS COM RESTRIÇÕES
DE ESTOQUE E CONDIÇÕES
FAVORÁVEIS DE MARÉ**

Dissertação apresentada ao Curso de Pós-Graduação em Engenharia de Eletricidade da Universidade Federal do Maranhão para obtenção do grau de Mestre em Engenharia Elétrica.

São Luís
2010

Barros, Victor Hugo

Problema de alocação de berços em portos graneleiros com restrições de estoque e condições favoráveis de maré / Victor Hugo Barros. - São Luís, 2010.

73f.

Impresso por computador.(fotocópia).

Orientador: Anselmo Cardoso de Paiva. Co-Orientador: Alexandre César Muniz de Oliveira.

Dissertação (Mestrado) - Universidade Federal do Maranhão, Programa de Pós Graduação em Engenharia de Eletricidade, 2010.

1.Pesquisa Operacional 2.Programação Linear 3.Algoritmos heurísticosI.Título.

CDU 519.8

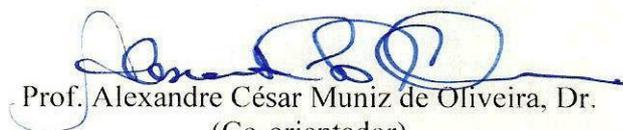
**PROBLEMA DE ALOCAÇÃO DE BERÇOS EM PORTOS
GRANELEIROS COM RESTRIÇÕES DE ESTOQUE
E CONDIÇÕES FAVORÁVEIS DE MARÉ**

Victor Hugo Barros Silva

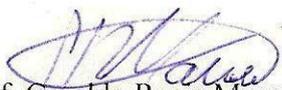
Dissertação aprovada em 22 de março de 2010.



Prof. Anselmo Cardoso de Paiva, Dr.
(Orientador)



Prof. Alexandre César Muniz de Oliveira, Dr.
(Co-orientador)



Prof. Geraldo Regis Mauri, Dr.
(Membro da Banca Examinadora)



Prof. Aristófanes Corrêa Silva, Dr.
(Membro da Banca Examinadora)

Aos meus pais, aos meus irmãos, à minha mulher e ao meu filho.

Agradecimentos

Aos meus pais, Edmilson e Sonia, que sempre me apoiaram e me deram todo o suporte para eu trilhar no caminho que escolhi.

Aos meus irmãos, Andréia e Marcel, e aos outros familiares, que, de alguma forma *fuzzy*, também me apoiaram.

À minha esposa, Selmira, e ao meu filho, Victinho, por fazerem parte da minha vida e por serem minha motivação nos dias difíceis.

Aos meus amigos (considerem-se todos citados), que sempre estiveram ao meu lado, incentivando-me para o bom caminho e para o mau também.

Ao pessoal do Laboratório de Aprendizagem Computacional e Métodos de Otimização (LACMO), Tarcísio, Paulo, Saulo e Ícaro, pela força no dia-a-dia e ao sexto elemento, Euziel, pela vital colaboração no trabalho.

Ao professor Alexandre Oliveira pela confiança, sem a qual eu teria desistido; pela tranquilidade passada e pela amizade construída durante os últimos anos.

Ao professor Anselmo Paiva e ao Programa de Pós-Graduação em Engenharia de Eletricidade (PPGEE) pela oportunidade me dada para realização deste trabalho.

Ao CNPq pelo apoio financeiro por meio do programa REUNI, assim como, ao pessoal da Pró-Reitoria de Ensino pela compreensão das dificuldades enfrentadas.

*“Nunca ande pelo caminho traçado, pois ele
conduz somente até onde os outros já foram”*

Alexander Graham Bell

RESUMO

O Problema de Alocação de Berços em Portos Graneleiros com Restrições de Estoque e Condições Favoráveis de Maré é abordado neste trabalho. Um berço define um local específico ao longo do cais onde um carregador de navio está disponível para carregar ou descarregar navios, acomodando apenas um navio por vez. Em portos que sofrem a influência da variação das marés, as condições de navegação dependem de condições favoráveis de maré, uma vez que a profundidade na maré baixa restringe a movimentação de navios. Alguns terminais no complexo portuário de São Luís, Maranhão, estão associados a importantes empresas multinacionais que mantêm um forte controle sobre os níveis de estoque de seus produtos. Uma vez que o nível de estoque, por vezes, depende de um processo contínuo de consumo ou produção de graneis, a tomada de decisão de carregar ou descarregar navios deve levar em conta as cargas armazenadas nos pátios do porto. Desta forma, um critério básico para tomadas de decisão é dar prioridade aos navios relacionada aos níveis mais críticos de estoque. Este trabalho apresenta dois modelos de programação linear baseado no problema de transporte para representar o problema abordado. Algumas instâncias do problema puderam ser resolvidas por um *solver* comercial. Como alternativa às instâncias maiores, que exigem grande tempo de execução, uma implementação do *Simulated Annealing* (SA) e do algoritmo conhecido como Algoritmo de Treinamento Populacional para Programação Linear (ATP/PL) são empregadas para resolução do problema.

Palavras-Chave: Programação linear, Problema de alocação de berços, Portos com restrições de maré, Controle de estoque

ABSTRACT

The problem of allocating berth positions for vessels in tidal grain port terminals is discussed in this work. A berth defines a specific location alongside a quay where a ship loader is available for loading or unloading vessels, accommodating only one vessel at time. In tidal ports, draft conditions depend on high tide conditions, since available depth under the low tide is not adequate to the movement of ships. Some port terminals at the port complex of São Luís, Maranhão, are associated to important transnational enterprises which maintain a strong control over the stock level of their goods. Since the stock level sometimes depends on a continuous process of consumption or production of grains, the decision making of loading or unloading vessels must contemplate the amount of the grain stored in the port yards. Therefore, a basic criterion for decision making is to give priority to the vessels related to the most critical grain stock level. This paper presents two integer linear programming models based on the transportation problem to represent the discussed problem. Some problem instances could be solved by a commercial solver. As an alternative to larger instances, which require large running time, an implementation of Simulated Annealing (SA) and the algorithm known as Population Training Algorithm for Linear Programming (PTA/LP) are used to solve the problem.

Keywords: Linear programming, Berth allocation problem, Tidal ports, Stock control

Lista de Tabelas

4.1	Exemplo de solução inicial desenvolvida com variáveis artificiais . . .	48
5.1	Exemplo de instância para terminais com berços homogêneos . . .	55
5.2	Exemplo de instância para terminais com berços heterogêneos . . .	56
5.3	Resultados obtidos pelo CPLEX para berços homogêneos	57
5.4	Resultados do <i>Simulated Annealing</i> para berços homogêneos . . .	60
5.5	Resultados obtidos pelo CPLEX para berços heterogêneos	62
5.6	Solução: navios x TTW's x Berços	64
5.7	Resultados do <i>Simulated Annealing</i> para berços heterogêneos . . .	65
5.8	Resultados do ATP/PL para berços heterogêneos	67

Lista de Figuras

1.1	Porto do Itaqui. Fonte: [EMAP, 2009]	7
1.2	Terminal Marítimo da Ponta da Madeita - Vale. Fonte: [VALE, 2008]	8
2.1	Porto graneleiro	13
2.2	Período de uma maré	14
2.3	Pátios de armazenamento na Vale. Fonte: [VALE, 2008]	16
4.1	PAB modelado como problema de transporte	33
4.2	Duas soluções diferentes com custos teoricamente iguais	42
4.3	High Demurrage Ship heuristic	51
5.1	Sequência de atendimento em 3 berços	59
5.2	Níveis de cada matéria-prima com a restrição de estoque	60
5.3	Níveis de cada matéria-prima sem a restrição de estoque	61

Lista de Abreviaturas e Siglas

AG	Algoritmo Genético
AGC	Algoritmo Genético Construtivo
ATP	Algoritmo de Treinamento Populacional
ATP/PL	Algoritmo de Treinamento Populacional com Programação Linear
COLS	Colunas geradas
D-W	<i>Dantzig-Wolfe</i>
EMAP	Empresa Maranhense de Administração Portuária
FC	Chamadas à Função Objetivo
FS	Solução Encontrada
GC	Geração de Colunas
GMAX	Maior Valor de <i>Fitness</i> no ATP
HDSH	<i>High Demurrage Ship heuristic</i>
PAB	Problema de Alocação de Berços
PAG	Problema de Atribuição de Guindastes
PL	Programação Linear
PLI	Programação Linear Inteira
PM	Problema Mestre
PMR	Problema Mestre Restrito
PPC	Problema de Particionamento de Conjuntos
RT	Tempo de Execução
SA	<i>Simulated Annealing</i>
SAD	Sistemas de Apoio à Decisão
SE	Subida da Encosta
TMPM	Terminal Marítimo de Ponta da Madeira
TTW	<i>Tidal Time Window</i>

Sumário

1	Introdução	5
1.1	Transporte Marítimo	5
1.2	Complexo Portuário Marítimo de São Luís	6
1.3	Objetivo	8
1.4	Revisão da Literatura	9
1.5	Organização do trabalho	11
2	Descrição do Problema	12
2.1	Problema de Alocação de Berços	12
2.1.1	Condições favoráveis de maré	14
2.1.2	Controle de estoque	15
2.2	Requisitos	15
2.3	Cenários	18
3	Fundamentos Teóricos	19
3.1	Programação Linear	19
3.2	Decomposição de Dantzig-Wolfe	20
3.3	Geração de Colunas	22
3.4	Algoritmos Genéticos	25
3.4.1	Algoritmo genético construtivo - AGC	26
3.4.2	Algoritmo de treinamento populacional - ATP	27
3.5	ATP/PL	29
3.6	Simulated Annealing	30
4	Modelos Propostos	32
4.1	Berços homogêneos	33

4.1.1	Dados de entrada	34
4.1.2	Função Objetivo	34
4.1.3	Restrições	35
4.2	Berços heterogêneos	37
4.2.1	Dados de entrada	38
4.2.2	Função Objetivo	39
4.2.3	Restrições	39
4.3	Variações no modelo	41
4.3.1	Variações de função objetivo	41
4.3.2	Restrições na utilização de equipamentos	44
4.3.3	Restrições de berços e temporais no atendimento de navios	45
4.4	ATP/PL aplicado ao problema	45
4.4.1	Problema Mestre Restrito	46
4.4.2	ATP	49
4.5	<i>Simulated Annealing</i> aplicado ao problema	49
4.5.1	Berços homogêneos	49
4.5.2	Berços heterogêneos	51
5	Resultados	53
5.1	Ambiente de testes	53
5.1.1	Instâncias	53
5.2	Experimentos Computacionais	56
5.2.1	Problema com berços homogêneos	57
5.2.2	Problema com berços heterogêneos	62
6	Conclusão	69

CAPÍTULO 1

Introdução

1.1 Transporte Marítimo

A importância dos portos cresce com os avanços tecnológicos na construção de grandes navios. A maior capacidade de navegação resulta em um número maior de operações de carregamento e descarregamento realizados por um navio. Além disso, o transporte marítimo é invariavelmente mais intensificado entre as nações devido à globalização e ao crescimento econômico mundial. Estes fatores são responsáveis pelo aumento da demanda pelos serviços portuários. É comum encontrar, em regiões próximas aos portos, vários navios aguardando para serem atendidos, por não existir posição de berço disponível para atracação. Assim, é comum os portos operarem perto de suas capacidades máximas.

O crescente aumento do fluxo de mercadorias entre nações acirra a competitividade econômica mundial, e os portos, por serem vértices importantes para o advento do comércio exterior, são exigidos cada vez mais para operarem de maneira eficiente. Uma outra questão importante a respeito dos sistemas portuários é o próprio papel sócio-econômico que desempenham em uma região. Algumas das características determinantes para a definição de um porto ideal são a maximização do desenvolvimento da nação, ou região, valorizando os conceitos político-econômicos, humanos e sociais; ou seja, o porto ideal seria capaz de atender navios de grande porte, com alto nível de mecanização, oferecendo fluidez ao transporte e elevada produtividade [SILVA, 2008]. Além disso, o crescimento constante da população mundial gera grandes demandas de itens relacionados

a alimentos e combustíveis, mantendo, desta forma, o comércio marítimo em constante crescimento [PEIXOTO, 1977] (apud [SILVA, 2008]).

Tendo em vista esses desafios, surge a necessidade de aproveitar ao máximo os recursos disponíveis nos portos. Usar os recursos de forma inteligente é fundamental porque, apesar das reformas e ampliações serem mecanismos adotados corriqueiramente pelas administrações portuárias, com objetivo de oferecer mais recursos e agilidade às operações com os navios, é fácil concluir que os portos dificilmente serão projetados para disporem de recursos acima da demanda exigida.

Durante vários anos, o sistema portuário brasileiro sofreu devido a carências no âmbito estratégico, administrativo e operacional em relação aos portos internacionais. As péssimas condições de operação, como máquinas sem conservação e obsoletas, trabalhadores mal preparados e ausência de planejamento administrativo, foram fatores decisivos para a baixa competitividade dos produtos brasileiros. Em 2000, uma determinada operação com *contêineres* em portos brasileiros podia tomar a atenção de até dezenas de trabalhadores, enquanto que em portos mais modernos como o de Roterdã, na Holanda, não era preciso mais que um operador [SILVA, 2008].

Na maioria dos países com superávit na balança comercial, o fator predominante para atingir tal feito foi a privatização dos portos. Desta maneira, foi possível ter controle absoluto de todas as operações portuárias, viabilizando o investimento em equipamentos mais modernos, no treinamento de trabalhadores e na consequente modernização dos processos de atendimento aos navios [SILVA, 2008].

1.2 Complexo Portuário Marítimo de São Luís

No Maranhão, existe o complexo portuário marítimo de São Luís, composto por três portos: Porto do Itaqui, Terminal Marítimo de Ponta da Madeira (TMPM) e o Porto da Alumar. A Empresa Maranhense de Administração Portuária (EMAP) é responsável pela administração do Porto do Itaqui, tendo como principal movimentação a importação de derivados de petróleo. A mineradora Companhia Vale do Rio Doce (Vale) opera no TMPM. A exportação de minério de ferro e pelotas, derivados do minério de ferro, é o seu principal negócio. O Porto da

Alumar está associado ao Consórcio de Alumínio do Maranhão (Alumar), que produz alumínio primário e alumina, derivados do alumínio primário. Neste caso, ao contrário da Vale, é realizada tanto a exportação de seus produtos quanto a importação de matérias-primas e insumos para fabricação destes [EMAP, 2009]; [ALUMAR, 2009]; [VALE, 2008]. Nas figuras 1.1 e 1.2 são mostrados os portos do Itaqui e da Ponta da Madeira, respectivamente.



Figura 1.1: Porto do Itaqui. Fonte: [EMAP, 2009]

O complexo portuário maranhense se destaca pela movimentação de granéis sólidos. Em 2008, 105,4 milhões de toneladas passaram pelos portos de São Luís [PORTOSMA, 2010], com 1451 navios sendo atracados. A principal diferença de granéis sólidos para *contêineres* ou cargas gerais soltas, do ponto de vista das operações portuárias, está no transporte e estocagem. Em terminais graneleiros não existe o problema de alocação de contêineres, por exemplo. Desta forma, o tempo de atracação está vinculado, principalmente, à capacidade de vazão dos carregadores/descarregadores de navios.

Outra característica do complexo portuário maranhense é a presença acentu-



Figura 1.2: Terminal Marítimo da Ponta da Madeira - Vale. Fonte: [VALE, 2008]

ada do fenômeno conhecido como *avanço das marés*. Existe uma grande amplitude dos níveis de maré de São Luís, limitando, desta forma, a navegação dos navios, nas operações de atracação ou desatracação, a determinados períodos do dia.

Diante da importância dos três portos marítimos maranhenses no cenário nacional e até internacional, é levantado o problema de alocação de berços (PAB), ou posições de berços, em terminais graneleiros. O problema é determinar onde e quando o navio deve ser atracado, minimizando o custo total provocado, onde considera-se as peculiaridades da região portuária de São Luís. Em geral, os berços são heterogêneos, ou seja, possuem infra-estruturas variadas, o que acarreta em capacidades de vazão diferentes, mas, frequentemente, atendem quaisquer tipos de mercadorias em cada uma das posições de berços.

1.3 Objetivo

Muitos trabalhos têm considerado o PAB em portos que movimentam *contêineres*. Nestes portos, surgem outros problemas de otimização relacionados ao transporte e armazenagem dos *contêineres*. Poucos trabalhos têm considerado portos que operam granéis sólidos. Além disso, o fenômeno conhecido como *avanço das marés*

é pouco influente nas demais regiões portuárias do mundo, e por isso também é pouco abordado.

De uma outra perspectiva, a financeira, a prevista expansão do complexo portuário de São Luís [GOVERNO DO MARANHÃO, 2009] deve favorecer o investimento em pesquisa e desenvolvimento tecnológico, visando a produção de sistemas de apoio a decisão (SAD) que auxiliem os gestores a examinar vários cenários segundo diferentes critérios de desempenho.

O objetivo deste trabalho é apresentar o PAB em um caso específico, com características peculiares à região portuária industrial de São Luís. Pretende-se formalizar este problema através da proposição de um modelo matemático, e resolvê-lo por meio de meta-heurísticas.

1.4 Revisão da Literatura

O objetivo do PAB é geralmente minimizar o tempo de serviço total dos navios. A distinção entre a importância dos navios pode ser feita pelo emprego de um regime de preço baseado no *demurrage* sobre as operações atrasadas. Penalidades podem ser explicitamente incluídas na função objetivo, considerando custos de *demurrage* quando o tempo de serviço do navio excede o tempo contratado [IMAI et al., 2006].

No caso discreto, o PAB pode ser modelado como um problema de escalonamento de máquinas paralelas não-relacionadas, onde os navios são tratados como tarefas e os berços como máquinas. No caso contínuo, o PAB é um problema de corte com restrições adicionais, posicionando os navios, sem sobreposição, em um plano bi-dimensional em tempo e espaço [PINEDO, 1995].

Em [LIM, 1998], o cais é representado como um linha contínua e o problema é heurísticamente resolvido por meio da alocação de pontos de atracação para os tempos de atracação do navio. Esta abordagem não resolve o problema geral no qual o tempo de atracação é uma variável de decisão e o tempo de atendimento varia ao longo do cais.

Em [NISHIMURA et al., 2001], um modelo de programação inteira não-linear e o algoritmo genético são apresentados. A meta-heurística é baseada em uma representação diferente da dimensão espacial na qual o cais é uma coleção de

segmentos, permitindo a dois navios compartilharem o mesmo segmento ao mesmo tempo se seus tamanhos forem compatíveis com o berço. Restrições adicionais considerando a profundidade do berço também são introduzidas.

Em [PARK e KIM, 2003], outro modelo de programação inteira não-linear considerando o problema de atribuição de guindastes (PAG) foi introduzido. A principal hipótese que permite integrar o PAB e o PAG é que os tempos de atendimento variam linearmente com o número de guindastes atribuídos a um navio. Os autores reconhecem que isto é uma aproximação da realidade. A função objetivo minimiza a soma das penalidades sobre todos os navios, assumindo que um ponto de atracação ótimo é conhecido e uma penalidade pode ser aplicada sempre que uma escolha diferente é feita. O algoritmo emprega a relaxação lagrangiana e o método de otimização por subgradientes.

Em [KIM e MOON, 2003], um modelo de programação linear inteira para o caso contínuo foi formulado. Um *solver* comercial é capaz de derivar a solução ótima para instâncias envolvendo sete navios em um horizonte de planejamento de três dias. Uma heurística baseada no *Simulated Annealing* é proposta para resolver instâncias de dimensões reais.

Em [LORENZONI et al., 2006], o problema de atender navios dentro de prazos acordados sob a condição do primeiro que chegar deve ser o primeiro a ser atendido foi formulado. Além disso, o uso da ferramenta desenvolvida, baseada em um modelo matemático de um problema de escalonamento de recursos restritos, é indicado, além de uma extensão de um algoritmo evolutivo diferencial. Os testes computacionais com dados gerados a partir de características do ambiente portuário real também são apresentados. Os autores fazem alusão a condições de maré, que podem restringir a entrada de navios no porto em certos intervalos de tempo.

[SILVA, 2008] faz uma ampla revisão sobre operações portuárias tanto no contexto operacional quanto no econômico, comparando os cenários brasileiro e mundial. Um algoritmo genético é proposto para resolver o problema de alocação de berços no Porto de Itajaí, em Santa Catarina.

Mais recentemente, em [MAURI, 2008b], uma estratégia híbrida, que combina o método conhecido como *geração de colunas* e o *Algoritmo de Treinamento Popu-*

lacional (ATP), da família dos algoritmos genéticos, é proposto para resolver uma formulação do PAB para o caso discreto, apresentado em [CORDEAU et al., 2005].

Em nenhum caso estudado, o PAB é considerado em portos que movimentam apenas granel, e nem o fenômeno de *avanço das marés* impõe restrições a atracação ou desatracação em janelas de tempo.

1.5 Organização do trabalho

Este trabalho está organizado em capítulos, sendo este destinado à contextualização do problema de alocação de berços, assim como a definição do objetivo e a justificativa para o trabalho. No Capítulo 2, o problema é descrito de forma detalhada, assumindo suposições que vão nortear a modelagem. Os principais conceitos utilizados neste trabalho são revisados no Capítulo 3. O problema formalizado e os modelos desenvolvidos para resolvê-lo são descritos no Capítulo 4. No último Capítulo, os resultados computacionais são apresentados. Ao final, na Conclusão, os principais achados são sumarizados.

Descrição do Problema

O problema de alocação de berços (PAB) é apresentado em um caso específico devido às particularidades observadas no complexo portuário de São Luís. São consideradas algumas restrições naturais do ambiente e alguns interesses típicos dos gestores dos portos que movimentam granéis sólidos na formalização do problema geral. Nas seções a seguir, é descrito o *Problema de Alocação de Berços em Portos Graneleiros com Restrições de Estoque e Condições Favoráveis de Maré*, além de alternativas de cenários operacionais.

2.1 Problema de Alocação de Berços

O Problema de Alocação de Berços (PAB) é definido como o planejamento do *tempo* e da *posição* de atracação (berço) referentes a cada navio [MOON, 2000] e [CORDEAU et al., 2005]. Se não houver posição de atracação adequada para o navio, este deve aguardar em uma fila de espera [FERNANDES, 2001]. Um cenário típico de um porto graneleiro é ilustrado na Figura 2.1. Todavia, para se adequar a cada caso, algumas suposições devem ser feitas.

As posições de atracação podem ser contínuas, onde outro desafio é encontrar espaços ao longo do cais para alocar os navios; ou podem ser discretas, sendo a dimensão espacial ignorada e o número de berços pré-determinado. O PAB pode ser ainda tratado de forma estática e dinâmica. Na primeira abordagem, considera-se que os navios já estão no porto esperando para serem atendidos. Na

segunda, uma generalização da primeira, considera-se também os navios que estão para chegar [MAURI, 2008b] [CORDEAU et al., 2005].

[BROWN, 1994], [NISHIMURA et al., 2001] e [SILVA, 2008] complementam observando que algumas variáveis, como profundidade e comprimento do berço e força dos cabos utilizados, podem inviabilizar a atracação de um navio em um berço. [CORDEAU et al., 2005] e [MAURI, 2008b] também mencionam que os horários de atracação podem ser limitados em *janelas de tempo*.

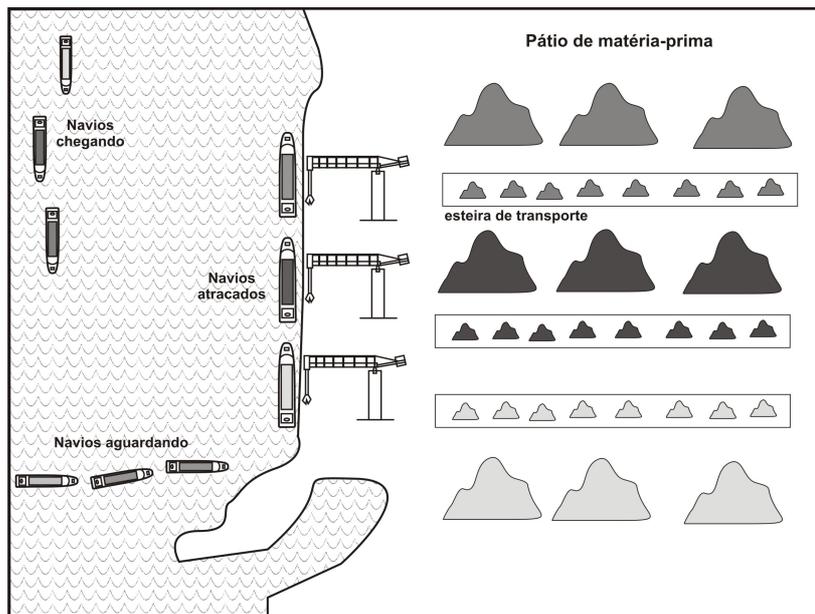


Figura 2.1: Porto graneleiro

O objetivo do PAB é minimizar os custos referentes ao porto e ao navio, o que é geralmente relacionado ao *tempo de serviço* dos navios, i.e., tempos em espera para ser atendido mais os tempos em atendimento [MAURI, 2008b]. Porém, outros critérios de desempenho também foram identificados, como por exemplo, a multa paga por sobrestadia do navio no porto devido a atrasos no atendimento, conhecida como *demurrage*. O *demurrage* é uma taxa paga diariamente para cada dia além do tempo limite estabelecido em contrato para a liberação do navio (término do atendimento), a partir do instante de chegada do mesmo. Altos valores pagos a título de *demurrage* indicam problemas na operação do terminal portuário, como má utilização de recursos ou problemas na logística de compra e venda dos produtos nele comercializados, desde que contratos mal elaborados

possam sobrecarregar o terminal desnecessariamente em determinados períodos no mês.

Assim, o escalonamento dos navios aos berços pode ser direcionado à redução do *demurrage*. Em contrato, é estabelecido o período máximo que o navio fica à disposição do porto para ser atendido. Se este valor é ultrapassado, uma taxa proporcional aos dias de espera a mais é cobrada. Identifica-se também que a sobrecarga, ou uso excessivo de um determinado carregador de navios (equipamento do berço responsável pelo carregamento dos navios) aumenta a probabilidade de manutenções corretivas, reduzindo a vazão do berço ou mesmo deixando-o inativo temporariamente. Desta forma, reduzir a taxa de uso dos carregadores de navios é algo desejável na programação portuária.

2.1.1 Condições favoráveis de maré

Em São Luís, o fenômeno conhecido por *avanço das marés* é bem acentuado. Durante um dia, existem dois períodos de maré cheia, *preamar*, e dois de maré baixa, *baixamar*. Desta forma, o ciclo de uma maré inicia-se em uma *preamar* e termina na seguinte, passando por uma *baixamar*, com duração total de, aproximadamente, 12 horas [CHM, 2010]. Na Figura 2.2, o período de uma maré é ilustrado.

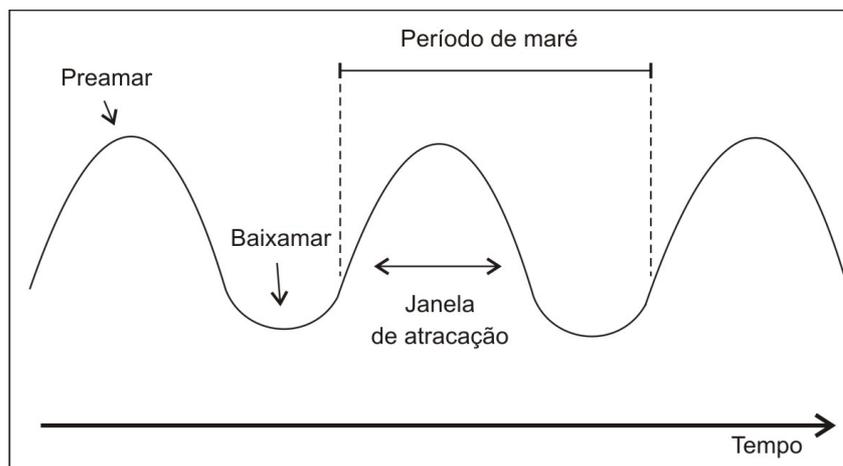


Figura 2.2: Período de uma maré

Na *baixamar*, as atracações ou desatracções, em geral, são inviabilizadas devido a pouca profundidade do canal de navegação até os berços. Apesar de ser con-

siderada uma das regiões portuárias mais profundas do mundo, os portos de São Luís sofrem com a variação dos níveis de maré. Além disso, grandes oscilações dos níveis de maré em pouco tempo acarretam em correntes marítimas fortes que também podem impossibilitar a navegação. Assim, sob a influência dos níveis de maré e das correntes marítimas, as atracções de navios são limitadas em janelas de tempo predeterminadas.

2.1.2 Controle de estoque

Na Alumar, a bauxita é refinada e transformada em alumina [ALUMAR, 2009]. O Porto da Alumar tem o papel de escoar a produção e de receber as matérias-primas e insumos necessários para o refino. A Vale extrai minério de ferro, entre outros, para exportação através do seu porto. Parte do minério extraído também é refinada e transformada em pelotas de ferro [VALE, 2008]. Estar associado à produção industrial traz um ônus a mais ao porto no processo de atracção de navios, tendo em vista que a falta de determinado insumo pode afetar a produção nas refinarias e, em casos de exportação, a falta de algum produto pode inviabilizar a sequência natural das operações de atracção.

Um controle dos níveis de estoque das matérias-primas, insumos e produtos torna-se necessário à medida que estes influenciem as importações/exportações. Considerando cada taxa de consumo/produção de cada carga somada à quantidade já armazenada nos pátios é possível controlar os estoques e evitar que navios permaneçam atracados ao porto sem carga para carregá-los ou que a produção pare por falta de insumos e matérias-primas. Na Figura 2.3, a produção da Vale “aguarda” em pátios de estocagem à espera de ser transportada a um navio. O fluxo de cargas é executado dos pátios até os navios por esteiras

2.2 Requisitos

Como visto na Seção 2.1, o PAB tem uma proposta geral mas pode assumir uma série de suposições que o torna especializado para um problema específico. As suposições que são apresentadas nesta Seção como requisitos e foram levantadas a partir das principais características identificadas na região portuária de São Luís, seguem abaixo:



Figura 2.3: Pátios de armazenamento na Vale. Fonte: [VALE, 2008]

- Como a programação dos navios é feita, em geral, em um horizonte de planejamento de, aproximadamente, uma semana, deve-se tratar o problema na forma dinâmica. Navios que estão para chegar fazem parte do planejamento, onde são conhecidos os momentos estimados de chegada.
- O *tempo de atendimento* do navio, também chamado de *tempo de tratamento*, ou seja, o tempo em que navio permanece atracado no porto carregando/descarregando, é a razão entre quantidade de carga transportada pelo navio e a velocidade de atendimento, ou vazão, do berço. Apesar de várias cargas serem manipuladas, considera-se que todas exigem a mesma velocidade de processamento de um berço. É importante ressaltar que quando é sabido o tempo de atendimento de cada navio em cada berço, esta aproximação é desnecessária.
- Um berço pode acomodar apenas um navio por vez, ou seja, trata-se do problema discreto.
- Não deve existir mudanças de berço. Um navio não pode atracar em um berço para posteriormente ser deslocado para outro, salvo em casos de quebra de equipamentos onde todo planejamento deve ser refeito a partir do momento de parada.

- Como os portos considerados neste problema têm indústrias importadoras ou exportadoras associadas, os pátios de armazenagem fazem parte do problema. Por isso, os níveis de carga (matérias-primas, insumos e produtos) devem ser mantidos em patamares seguros para que não haja paradas na produção industrial, ou os navios não fiquem esperando por cargas ainda não disponibilizadas pelo porto.
- As atracações devem ser limitadas a janelas de tempo devido aos efeitos do *avanço das marés*. É importante ressaltar que as características de cada porto determinam quais são as melhores janelas para atracação, ou seja, não é, necessariamente, apenas o período de *preamar* ou *baixamar* que as determinam. O porto pode ter o canal de navegação e berços profundos, ou manipular apenas navios de pequeno e médio porte, com calados pequenos, por exemplo.

Critérios de Desempenho

Alguns critérios de desempenho foram identificados. Como foi visto na Seção 2.1, normalmente, o objetivo é minimizar o total dos *tempos de serviço* dos navios. A minimização do tempo de serviço de cada navio, no entanto, não necessariamente implica na minimização do tempo total de atendimento no porto. Existem situações em que precisa-se retardar o atendimento de um ou outro navio para obter-se um ganho de tempo na programação total do porto.

O tempo de atendimento de um navio varia de um berço para o outro em função da vazão que este berço provê, e o tempo de espera de um navio pelo momento e local de atracação são variáveis que interferem no tempo de serviço. O tempo de serviço é, portanto, o tempo total em que o navio fica à disposição do porto para ser liberado (soma dos tempos de espera e atendimento).

Outro objetivo também foi levantado: *demurrage*. Pode-se dizer que o *demurrage* atribui pesos ao tempo de serviço, definindo prioridades flexíveis de atracação. Apesar de não ser o primeiro critério a ser pretendido na programação do porto, é algo que pode ser investigado. Minimizar a taxa de uso do berço é um objetivo secundário, que pode ser considerado como uma penalização.

2.3 Cenários

A região portuária contém três portos administrados por empresas distintas. Assim, naturalmente, existem algumas diferenças estruturais entre eles. Dois cenários semelhantes, porém com possibilidades de tratamento diferentes, são apontados:

- O primeiro cenário consiste em assumir que todos os berços tenham a mesma infraestrutura e, conseqüentemente, a mesma vazão no atendimento do navio. Desta forma, *onde* o navio foi atracado é irrelevante, ou seja, não tem influência no planejamento. Neste cenário, define-se como critério de desempenho a minimização do *demurrage*, em parte, para fins de investigação.
- No segundo cenário, os berços são heterogêneos, ou seja, possuem infraestruturas diferentes, que conduzem os berços a tempos de processamento diferentes para o mesmo navio. Neste caso, decidir onde o navio será atracado tem importância fundamental na obtenção de seu tempo de serviço. Este caso é mais geral e, por isso, define-se como critério de desempenho o tempo de serviço de todos os navios, que julga-se o mais adequado.

Fundamentos Teóricos

3.1 Programação Linear

A programação matemática lida com a otimização de uma função objetivo, satisfazendo a um conjunto de restrições de igualdade e/ou desigualdade. Um caso especial de programação matemática é a programação linear (PL), na qual as funções envolvidas são todas lineares. O termo programação refere-se a planejamento de atividades, devido a sua concepção está ligada a este objetivo [BAZARAA et al., 1990] [HILLIER e LIEBERMAN, 2006].

Existem várias formas de modelagem de problemas, i.e., formas de representação de problemas para fins de estudo. Um modelo matemático tem a finalidade de descrever, sem ambiguidades, um dado problema ou situação do mundo real. A modelagem matemática é especialmente interessante pois permite o uso da programação matemática para que se extraia resultados numéricos, o que permite uma análise mais precisa dos objetivos e requisitos do problema.

Os principais desenvolvimentos teóricos sobre PL são atribuídos a Leonid Kantorovich, mas foi George Dantzig o responsável pelo desenvolvimento do principal método de resolução de problemas de PL, chamado *Simplex*, no final da década de 1940 [BAZARAA et al., 1990] [TAVARES e CORREIA, 1999]. O método Simplex é capítulo básico em Pesquisa Operacional por viabilizar o entendimento sobre métodos mais complexos de resolução de problemas de otimização, incluindo métodos de pontos interiores que, em tese, são seus concorrentes bem menos populares [HILLIER e LIEBERMAN, 2006]. Apesar da restrição de linearidade,

a modelagem matemática para resolução através do Simplex se mostra como um desafio e passo elementar para um melhor entendimento do problema.

PL é utilizada com sucesso nos mais variados tipos de problemas, como em problemas de *fluxo em redes, logística, roteamento de veículos, cobertura e particionamento de conjuntos*, entre tantos outros. Todavia, é importante ressaltar que a sua importância não surge apenas pelas suas numerosas aplicações práticas, mas também pelo fato de sua criação estar ligada ao próprio desenvolvimento da teoria sobre otimização [GOLDBARG e LUNA, 2005],[TAVARES e CORREIA, 1999].

3.2 Decomposição de Dantzig-Wolfe

Não é incomum modelos de programação linear serem produzidos com muitas milhares de linhas e um número aparentemente ilimitado de colunas. Nestes problemas, algum método deve ser aplicado para reduzir o problema grande em um ou mais problemas menores apropriadamente coordenados e de tamanho tratável [BAZARAA et al., 1990].

A decomposição de Dantzig-Wolfe (D-W) é uma técnica utilizada na resolução de problemas de programação linear e de programação inteira de grandes dimensões [DANTZIG e WOLFE, 1960], em geral, associada com sucesso ao método de Geração de Colunas (GC) em diversos tipos de problemas, especialmente, ao longo da última década [PIMENTEL, 2005]. A decomposição D-W é uma técnica baseada na idéia de “dividir para conquistar” [MAURI, 2008b], onde um problema maior, de difícil ou até mesmo impossível resolução, é decomposto em problemas menores, tipicamente bem estruturados, mais fáceis de serem resolvidos [PIMENTEL, 2005]. Uma descrição é apresentada a seguir, conforme visto em [BAZARAA et al., 1990].

Primeiramente, considere o problema de PL em (3.1)-(3.4).

$$\text{Minimizar} \quad \mathbf{cx} \quad (3.1)$$

$$\text{Sujeito a} \quad \mathbf{Ax} = \mathbf{b} \quad (3.2)$$

$$\mathbf{Dx} = \mathbf{d} \quad (3.3)$$

$$\mathbf{x} > 0 \quad (3.4)$$

Definindo-se $X = \{\mathbf{D}\mathbf{x} = \mathbf{d}, \mathbf{x} > 0\}$, tem-se o problema de PL nas equações (3.5)-(3.7), onde X é um conjunto poliédrico representando restrições de estrutura especial, \mathbf{A} é uma matriz $m \times n$ e \mathbf{b} é um vetor m .

$$\text{Minimizar} \quad \mathbf{c}\mathbf{x} \quad (3.5)$$

$$\text{Sujeito a} \quad \mathbf{A}\mathbf{x} = \mathbf{b} \quad (3.6)$$

$$\mathbf{x} \in X \quad (3.7)$$

Para simplificar a apresentação, assume-se que X é limitado. Uma vez que X é um conjunto poliédrico limitado, então qualquer ponto $x \in X$ pode ser representado como uma combinação linear convexa do número finito de pontos extremos de X . Denotando estes pontos por x_1, x_2, \dots, x_t , qualquer $x \in X$ pode ser representado como

$$\mathbf{x} = \sum_{j=1}^t \lambda_j \mathbf{x}_j \quad (3.8)$$

$$\sum_{j=1}^t \lambda_j = 1 \quad (3.9)$$

$$\lambda_j \geq 0, j = 1, 2, \dots, t \quad (3.10)$$

Substituindo por \mathbf{x} , o problema de PL em (3.5)-(3.7) pode ser reformulado nas variáveis $\lambda_1, \lambda_2, \dots, \lambda_t$, como abaixo.

$$\text{Minimizar} \quad \sum_{j=1}^t (\mathbf{c}\mathbf{x}_j) \lambda_j \quad (3.11)$$

$$\text{Sujeito a} \quad \sum_{j=1}^t (\mathbf{A}\mathbf{x}_j) \lambda_j = \mathbf{b} \quad (3.12)$$

$$\sum_{j=1}^t \lambda_j = 1 \quad (3.13)$$

$$\lambda_j \geq 0, j = 1, 2, \dots, t \quad \mathbf{x} \in X \quad (3.14)$$

O problema de encontrar os pontos extremos de X é chamado de *subproblema* e o problema em (3.11)-(3.14) é denominado *problema mestre* (PM).

Estabelecendo um paralelo entre o problema mestre e o problema original (3.5)-(3.7), verifica-se um grande número de variáveis a mais no PM, mais precisamente o número de pontos extremos em X . Em compensação, há grande redução no número de restrições. As restrições que constituem o domínio de X no problema original são substituídas por uma única restrição de convexidade no PM.

Devido a quantidade enorme de pontos extremos, enumerá-las explicitamente pode ser uma tarefa árdua ou até mesmo, em alguns casos, impossível. A solução, entretanto, pode ser determinada sem necessariamente enumerar todas as variáveis. Na prática, existem muitos pontos extremos que recebem valor nulo numa solução ótima do problema mestre. Assim, nem todos são necessários para a obtenção da sua solução ótima [PIMENTEL, 2005]. A técnica de geração de colunas trabalha exatamente esta característica dos problemas de PL, inserindo gradativamente as variáveis “interessantes” no PM até culminar na solução ótima do problema original.

3.3 Geração de Colunas

Geração de Colunas (GC) é um método utilizado na Programação Linear (PL) que visa resolver problemas de grande dimensão, com alto custo computacional, em tempos aceitáveis de execução. Esta técnica foi desenvolvida a partir do princípio da decomposição de Dantzig-Wolfe (ver Seção 3.2) e seus melhores resultados estão no tratamento de problemas de Programação Linear Inteira (PLI), onde as variáveis são inteiras. A primeira aplicação prática foi na determinação de padrões de corte unidimensionais [GILMORE e GOMORY, 1961] [GILMORE e GOMORY, 1963] e, desde então, seu uso difundiu-se de forma intensa [LORENA et al., 2003].

Como já mencionado na Seção 3.2, a decomposição de Dantzig-Wolfe retira um subconjunto de restrições X no problema mestre, reformulado a partir do problema original, deixando a cargo do subproblema determinar os pontos extremos λ_p do poliedro de soluções factíveis formado por X . Em compensação, são obtidas muitas variáveis a mais que o problema original. O método de geração de colunas propõe uma interação entre o subproblema e o problema mestre, onde o primeiro

fornece, gradativamente, apenas as colunas *favoráveis* para melhora na solução atual fornecida pelo PM. Este, por sua vez, é executado e suas variáveis duais são passadas para o subproblema, direcionando a geração de boas colunas. O método pára quando novas colunas favoráveis não forem mais geradas.

Quando se considera apenas um subconjunto de pontos extremos, o problema mestre passa a ser denominado *problema mestre restrito* (PMR). O subproblema é chamado de *subproblema de geração de colunas*, pois as variáveis obtidas pela resolução deste problema constituirão novas colunas para o problema-mestre restrito [SALOMÃO, 2005].

Como no método Simplex, a cada iteração é necessário encontrar variáveis promissoras para entrar na base. Uma iteração consiste de otimizar o problema mestre restrito para determinar o valor objetivo ótimo atual \bar{z} e os multiplicadores duais π e θ ; e, de encontrar uma variável λ_p com *custo reduzido* negativo, se ainda existir uma. A busca implícita por um custo reduzido mínimo conduz a otimização de um subproblema [DESAULNIERS et al., 2005]. O custo reduzido pode ser calculado como nas equações (3.15)-(3.17).

$$\bar{c}^* = \text{Minimizar } (\mathbf{c} - \pi \mathbf{A})\mathbf{x} - \theta \quad (3.15)$$

$$\text{Sujeito a } \quad \mathbf{D}\mathbf{x} = \mathbf{d} \quad (3.16)$$

$$\mathbf{x} > 0 \quad (3.17)$$

Se $\bar{c}^* \geq 0$, não existe variável promissora para entrar na base e o problema mestre não pode encontrar solução melhor. Caso contrário, a variável é inserida no problema mestre e o processo se repete. Todo o método de geração de colunas é resumido no Algoritmo 1.

```

1 gerarModeloInicial();
2 solucaoPL = resolverPMR();
3 solucaoSub = resolverSubproblema(solucaoPL. $\pi$ );
4 while solucaoSub.custo_reduzido < 0 do
5     addColuna();
6     solucaoPL = resolverPMR();
7     solucaoSub = resolverSubproblema(solucaoPL. $\pi$ );
8 end
9 return solucaoPL.otimo

```

Algoritmo 1: Geração de colunas

Uma vez evitadas as soluções cíclicas no problema-mestre e a degeneração no subproblema, o algoritmo converge em um número finito de iterações, pois consiste do algoritmo Simplex Revisado. Além disso, o subproblema não precisa ser completamente otimizado em cada iteração. É necessário apenas que as colunas geradas tenham custos reduzidos negativos para que seus respectivos λ_p sejam candidatos a entrar na base [SALOMÃO, 2005].

Em problemas de PLI, após determinada a solução final do problema mestre, as restrições de integralidade são inseridas no problema e este pode ser resolvido pelo algoritmo *branch-and-bound*. Uma alternativa para derivar soluções exatas diretamente usando GC é o algoritmo conhecido como *branch-and-price*, que trabalha sobre o *framework* do *branch-and-bound*. Para mais informações, [BARNHART, 1998] fornece uma introdução neste método.

Outras abordagens de uso do método de Geração de colunas também são comuns. Apesar de GC ser desenvolvida a partir do método de decomposição de Dantzig-Wolfe, outras decomposições ganharam espaço. Em geral, quando a dimensão do problema de PLI for muito grande, a GC é uma alternativa para a obtenção de soluções em tempos aceitáveis. Além disso, para melhorar ainda mais os tempos de execução, a substituição de um modelo de PL ou PLI no subproblema por um algoritmo heurístico pode ser de grande valia.

3.4 Algoritmos Genéticos

Em meados do século XIX, Charles Darwin propôs a *teoria da evolução* biológica. Esta teoria observa que as espécies de animais e plantas com variações individuais benéficas a sua sobrevivência pela melhor adaptação ao ambiente têm mais chances de sobrevivência na geração seguinte. Este fenômeno ficou conhecido como *sobrevivência dos mais adaptados* [HILLIER e LIEBERMAN, 2006].

A genética moderna forneceu explicação melhor sobre a evolução e a *seleção natural* quando demonstrou que, em qualquer reprodução sexuada, as características dos *pais* são transmitidas aos *filhos* através dos *cromossomos*. Subgrupos que constituem os cromossomos, denominados *genes*, determinam as características dos filhos. Assim, os indivíduos que herdaram as melhores características dos pais têm mais chances de sobreviver na idade adulta e, conseqüentemente, transmitir suas boas características para as gerações seguintes. A tendência é, desta forma, que a população de indivíduos melhore lentamente ao longo do tempo [HILLIER e LIEBERMAN, 2006] [GOLDBARG e LUNA, 2005].

O processo de evolução originou os chamados *algoritmos genéticos* (AG), que são métodos de busca que exploram o espaço de soluções formado por uma população de indivíduos, evoluindo-os até atingir uma população “superior”. John Holland é considerado o pioneiro por seus trabalhos em [HOLLAND, 1962] e [HOLLAND, 1970], mas, quando fez a analogia do processo evolutivo a partir do modelo de cromossomos, realizou o primeiro grande feito em algoritmos genéticos, [HOLLAND, 1973] e [HOLLAND, 1975], ao determinar máximos e mínimos de funções matemáticas [GOLDBARG e LUNA, 2005].

Um algoritmo genético básico pode ser resumido como uma população inicial de indivíduos que representam uma solução cada, onde operadores genéticos, *seleção*, *cruzamento* e *mutação*, são responsáveis pelo processo de evolução até que um critério de parada seja atingido. Os principais elementos de um AG são descritos abaixo:

Codificação Primeiramente, deve-se definir como uma solução será representada como um indivíduo.

Fitness Função que avalia cada solução. Determina o quão adaptado é um indi-

víduo ao ambiente, ou o quão é boa a solução. Se a solução for inviável é comum descartá-la (aborto) ou atribuir uma penalidade a ela.

População inicial Pode ser determinada heurísticamente ou aleatoriamente.

Seleção Os pais são selecionados onde é desejável utilizar algum critério para que os indivíduos mais adaptados tenham mais chances de serem escolhidos.

Cruzamento Produz um ou mais filhos através do cruzamento de informações dos pais. Como a seleção, também deve ser tendencioso para extrair as melhores combinações. Por isso, a estratégia depende muito do problema abordado.

Mutação Causa uma deformidade no indivíduo. Ela é necessária porque a tendência do algoritmo é convergir para um mínimo que nem sempre é o global. Aplicar a mutação representa dar um “salto” no espaço de soluções, o que melhora a *diversidade* da população.

Critério de parada Geralmente, usa-se um tempo de processamento ou um número de iterações máximos.

Algoritmos genéticos são bem aceitos no meio acadêmico, principalmente pela sua capacidade de resolver problemas complexos. Entretanto, a robustez prejudica o tempo computacional em problemas menores [OLIVEIRA, 2004]. Em [OLIVEIRA, 2004] também é relacionado outros prós e contras do uso de algoritmos evolucionários e, em [GOLDBARG e LUNA, 2005], são listados alguns fatores que tornam os AG's a parte da computação evolucionária que encontrou a possibilidade de aplicação geral.

3.4.1 Algoritmo genético construtivo - AGC

O algoritmo genético construtivo (AGC) é uma variação do algoritmo genético tradicional resultado da observação de que as boas soluções são obtidas através da combinação de blocos construtivos [FURTADO, 1998].

O AGC é composto por uma população de *esquemas* e *estruturas*. Os esquemas são indivíduos incompletos com trechos “genéticos” ausentes, formando blocos de construção, e as estruturas são os indivíduos completos. A idéia do AGC é evoluir

os esquemas e as estruturas, onde os melhores esquemas são privilegiados, e por meio das recombinações sucessivas a população seja construída [FURTADO, 1998] [CORRÊA e LORENA, 2006]. Um ponto importante é a estratégia de avaliação dos esquemas, tendo em vista que, por não serem soluções completas, comparações com estruturas podem não ser representativas. [FURTADO, 1998] apresenta também uma estratégia de avaliação de esquemas.

Uma população dinâmica de esquemas e estruturas e a possibilidade de serem empregadas heurísticas específicas para a avaliação dos indivíduos são características que tornam o AGC diferente das abordagens tradicionais. Além disso, originalmente, a avaliação do AGC é feita através de duas funções, f e g (avaliação $f - g$), que são elaboradas considerando características inerentes ao problema de tal forma que $|f - g| \approx 0$ represente a melhor adaptação [OLIVEIRA, 2004]; [LORENA e FURTADO, 2001].

3.4.2 Algoritmo de treinamento populacional - ATP

O Algoritmo de Treinamento Populacional (ATP) foi desenvolvido a partir do Algoritmo Genético Construtivo (AGC) apresentado na Seção 3.4.1. A idéia fundamental do ATP é avaliar os indivíduos através de uma heurística de treinamento e evoluir a população de indivíduos privilegiando os mais adaptados a esta heurística.

Segundo [OLIVEIRA, 2002], a idéia de treinamento sugere um processo no qual um modelo consegue se adaptar ou assimilar determinado ambiente inicialmente desconhecido por ele. O termo treinamento empregado no ATP diz respeito ao processo de minimizar a avaliação chamada $f - g$ onde g é a função de *fitness* do indivíduo e f a heurística de treinamento adotada.

O ATP pode ser definido através da tupla $\{P, \Theta, f, H, \varphi, \delta\}$, onde P é uma população amostrada do espaço de busca codificado S , conseqüentemente indivíduos $s_k \in P \subset S$. Indivíduos são gerados por um conjunto de operadores específicos Θ e avaliados por uma função objetivo f que mapeia S em \mathfrak{R} . A heurística de treinamento H é definida pelo par de $\{\varphi^H, g\}$, onde g avalia heurísticamente cada solução gerada através da relação de vizinhança φ^H . A relação de vizinhança φ^H pode ser entendida como o conjunto de soluções obtidas a partir da solução original, s_k , através de movimentos de heurísticas (movimentos H):

$$\varphi^H(s_k) = \{s_k, s_{v1}, s_{v2}, \dots, s_{vl}\} \quad (3.18)$$

onde $l + 1$ é o tamanho da vizinhança de s_k , incluindo ele mesmo.

O conhecimento heurístico sobre o problema é então usado para definir g . Um típico g é a função objetivo f calculada sobre $\varphi^H(s_k)$. Desta forma, para problemas de minimização:

$$g(s_k) = f(s_{vb}) = \min\{f(s_k), f(s_{v1}), f(s_{v2}), \dots, f(s_{vl})\} \quad (3.19)$$

O melhor vizinho de s_k é denotado por s_{vb} . O conceito de proximidade, \wp , está preocupado com o esforço necessário para atingir s_{vb} a partir de s_k através de movimentos H . Mais proximidade significa mais adaptação de s_k à heurística que gerou s_{vb} . Dependendo da codificação sendo empregada na aplicação, algumas métricas de distância podem ser usadas, como *Hamming* para codificação binária, *Euclidiana* para codificação real e distância heurística.

Independentemente da natureza do problema (minimização ou maximização), o $\wp(s_k, s_{vb})$ sempre reflete quanto s_{vb} é melhor que s_k . Entretanto, se uma outra métrica de distância foi usada, $\wp(s_k, s_{vb})$ significaria apenas a adaptação para o treinamento heurístico. Finalmente, a população inteira é classificada através da função δ que atribui um *rank* a cada indivíduo. A função δ é mostrada em 3.20.

$$\delta_{ncons}(s_k) = d \cdot [G_{max} - g(s_k)] - |f(s_k) - g(s_k)| \quad (3.20)$$

Considerando G_{max} uma estimativa do limite superior para todos os possíveis valores da função g (mesma função f), o intervalo $G_{max} - g(s_k)$, dá a distância de adaptação entre o indivíduo s_k e o limite superior. Esta distância é usada em duas formas diferentes. Na classificação, o intervalo apenas considera a avaliação da função objetivo, uma vez em problemas de minimização, o maior é $G_{max} - g(s_k)$, o melhor é o indivíduo. A constante d é usada para equilibrar ambas equações de classificação (geralmente, sobre $1/G_{max}$). No início da evolução, o limite superior G_{max} pode ser analiticamente calculado, considerando a instância do problema, ou estimado através de amostragem.

3.5 ATP/PL

O ATP/PL é um método híbrido apresentado em [MAURI, 2008a] que combina o ATP (ver Seção 3.4.2) com a programação linear (ver Seção 3.1). O algoritmo trabalha sob o *framework* do método de geração de colunas (descrito na Seção 3.3).

A idéia fundamental é aproveitar a característica do método de geração de colunas de não exigir que o subproblema retorne, necessariamente, a melhor coluna em cada interação. Basta as colunas terem custo reduzido negativo (em problemas de minimização) e já são candidatas a entrar na base e, conseqüentemente, produzir solução melhor. Desta forma, com mais colunas sendo geradas por um método heurístico, ao invés da melhor por interação através de um método exato, é possível acelerar a convergência da GC.

O ATP/PL é constituído no problema mestre por um *problema de particionamento de conjuntos* (PPC) e no subproblema pelo ATP, que é responsável por evoluir boas colunas (com boa cobertura de conjuntos e custo baixo). A formulação do PPC é feita como segue:

$$\text{Minimizar} \quad \sum_{p=1}^n c_p \lambda_p \quad (3.21)$$

$$\text{Sujeito a} \quad \sum_{p=1}^n a_{ip} \lambda_p \quad \forall i \in M \quad (3.22)$$

$$\lambda_p \in \{0, 1\} \quad p = 1, 2, \dots, n \quad (3.23)$$

onde c_p é o custo da coluna p , M é o conjunto de *tarefas* existentes e a_{ip} determina se a atividade i faz parte da coluna p , sendo n o número de colunas. Caso i faça parte de coluna p , a_{ip} assume 1 e $a_{ip} = 0$, caso contrário. Se a coluna p pertencer à solução, a λ_p é atribuído 1 senão λ_p recebe 0.

Como no método de geração de colunas, o primeiro passo do ATP/PL é definir o modelo inicial, ou o conjunto de colunas inicial. Deve existir a garantia de solução viável para que hajam variáveis duais a serem transmitidas ao subproblema. Assim, aleatoriamente ou heurísticamente, deve-se ser gerar colunas combinadas que formem uma solução.

O ATP desenvolvido para o subproblema segue as mesmas definições apre-

sentadas na Seção 3.4.2. Cada indivíduo deve representar uma coluna no PPC. Assim, o PPC e ATP interagem para a geração de bons indivíduos. Originalmente, o subproblema de GC avalia as colunas pelo seu custo reduzido, atualizado a cada interação através dos duais obtidos pela resolução do problema mestre. No entanto, no ATP/PL o *fitness* utiliza os mesmos termos do cálculo do custo reduzido mas formulada de forma a valorizar a boa cobertura de conjuntos. A função g é definida na Equação 3.24.

$$g_p = \begin{cases} \frac{c_p}{\sum_{i=1}^m \pi_i a_{ip}} & , \text{ se } \sum_{i=1}^m \pi_i a_{ip} > 0 \\ c_p & , \text{ se } \sum_{i=1}^m \pi_i a_{ip} \leq 0 \end{cases} \quad (3.24)$$

onde π_i é a variável dual associada à restrição i .

O valor de custo reduzido \bar{c}_p é, entretanto, necessário para determinar quais colunas serão incorporadas ao PPC. Todas as colunas com $\bar{c}_p < 0$ (em problemas de minimização) são agregadas ao problema mestre porque são candidatos a entrar na base. Na Equação 3.25 é apresentada o cálculo de \bar{c}_p .

$$\bar{c}_p = c_p - \sum_{i=1}^m \pi_i a_{ip} \quad (3.25)$$

O processo se repete até o critério de parada ser atingido. O número de colunas geradas, o tempo de execução, o número total de interações ou o número de interações sem melhora na solução do problema mestre são algumas possíveis métricas que podem ser utilizadas. Após terminada a execução, o PPC resultante pode, então, ser resolvido com as restrições de integralidade pelo método *branch and bound*.

As informações presentes nesta Seção e maiores detalhes sobre o ATP/PL são encontrados em [MAURI, 2008a] e [MAURI, 2008b].

3.6 Simulated Annealing

O *Simulated Annealing* (SA) é um algoritmo aproximativo de busca local proposto por [KIRKPATRICK et al., 1983]. Ele é inspirado no processo físico de resfriamento de sólidos, que visa atingir formas bem definidas após o fim do processo.

O processo de resfriamento é antes antecipado por um aquecimento até que o sólido entre em estado de fusão. Na sequência, é realizado o resfriamento lentamente. O objetivo de realizar-se este lento resfriamento se dá pelo fato de que a forma final do material desejada seja a mais perfeita possível, fazendo com que as partículas altamente dispersas possam se reajustar gradativamente. Caso isto não fosse feito, certamente teria-se sólidos deformados.

O SA é dotado de parâmetros como temperatura inicial, taxa de resfriamento e número máximo de iterações por grau de temperatura, que devem ser devidamente ajustados, pois este ajuste é fundamental para a obtenção de bons resultados. O SA, ao contrário de algoritmos como *subida da encosta* (SE), aceita movimentos que melhorem ou que piorem o estado atual. Ao passo que o SE aceita apenas movimentos de melhoria. As vezes, movimentos de piora podem fazer com que o algoritmo escape de pontos de convergência locais.

O aceite de soluções que piorem o valor atual não é sempre realizado e dependerá de dois fatores: temperatura atual e a distância da solução nova à atual; essa distância se refere a quanto a solução piorou. Isto é demonstrado pela Equação 3.26.

$$P = e^{(-\Delta/T)} \quad (3.26)$$

onde Δ é a diferença entre a solução nova e a atual e P é a probabilidade de aceitação da nova solução. O que se pode depreender da equação é que quanto maior o valor da temperatura T , a tendência é que o valor entre parênteses seja, na maioria das vezes, muito pequeno. Sendo assim, tem-se um alto valor para a probabilidade de aceitação, mas que ainda dependerá do valor de Δ , ou seja, mesmo sendo grande o valor de T , se Δ for alto também, a probabilidade irá diminuir a proporção que Δ aumenta.

CAPÍTULO 4

Modelos Propostos

Dois modelos matemáticos são apresentados neste Capítulo, diferindo, principalmente, sobre as suposições que são feitas quanto às infraestruturas de seus berços. O primeiro considera todos os berços homogêneos, ou seja, com capacidades de carregamento ou descarregamento iguais. No segundo modelo, os berços são considerados com estruturas diferentes entre si, sendo assim, cada berço possui capacidade de serviço, carregamento ou descarregamento, diferente.

A heterogeneidade dos berços considerada refere-se apenas à velocidade de atendimento nas operações com os navios. Não há restrições entre navios e berços, ou seja, qualquer navio pode atracar em qualquer berço e todas as cargas podem ser movimentadas por todos os berços. O cenário com berços heterogêneos é claramente o mais próximo da realidade, sendo o respectivo modelo evoluído a partir do primeiro.

O fato das janelas de tempo disponíveis para atracação conduz ao tratamento do tempo na forma discreta, supondo que os períodos com condições favoráveis sejam suficientemente pequenos. Em geral, para portos com grande amplitude dos níveis de maré, é fornecido pequenos intervalos para as operações de atracamento e desatracamento. Desta forma, o horizonte de planejamento pode ser dividido em $|M|$ janelas de tempo. Neste documento, o acrônimo em inglês TTW (*Tidal Time Window*) é utilizado ao invés do equivalente em português.

Neste trabalho, o PAB é modelado na forma discreta como um problema de transporte no qual N navios são vistos como consumidores e os berços L e as janelas de marés favoráveis M (*Tidal Time Window* - TTW), como fornecedores.

Cada navio i deve ser alocado a um berço l em um subconjunto de TTW's cuja cardinalidade depende do número de marés necessárias h_{il} para que a operação de carga/descarga seja completada.

Na Figura 4.1, é ilustrado o PAB modelado como problema de transporte.

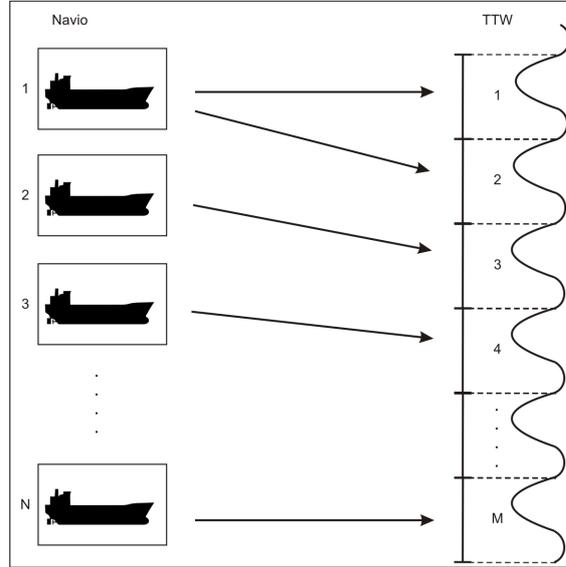


Figura 4.1: PAB modelado como problema de transporte

4.1 Berços homogêneos

A variável de decisão x_{ij} é dada por:

$$x_{ij} = \begin{cases} 1 & \text{se o navio } i \text{ está alocado à TTW } j \\ 0 & \text{caso contrário} \end{cases} \quad (4.1)$$

A decisão é determinar uma sequência de x_{ij} 's para cada navio i que determine o subconjunto de TTW's em que o navio permanece atracado, sendo atendido pelo porto. O número de marés em atendimento é definido pelo parâmetro h_{il} , mas para o caso com berços homogêneos, pode ser resumido a h_i por não haver distinção entre as infra-estruturas dos berços.

4.1.1 Dados de entrada

Os dados de entrada para o PAB neste problema são:

- N : conjunto de navios;
- M : conjunto de TTW's;
- K : conjunto de matérias-prima operadas no porto;
- a_i : TTW de chegada do navio i ;
- h_i : tempo de tratamento do navio i ;
- t_i : quantidade de TTW's contratualmente permitidas para o navio i ;
- d_i : *demurrage* para o navio i ;
- e_k : nível de estoque inicial da carga de granel k ;
- c_k : quantidade de consumo ou produção da carga de granel k ;
- q_{ik} : quantidade de carga do navio i em relação a carga k .

Uma matéria-prima k pode ser importada ou exportada (granéis de entrada ou de saída), uma vez que o tipo de operação (descarga ou carga) é definido através dos sinais de c_k e q_{ik} . Quando $q_{ik} = 0$, o navio i não movimenta a carga k .

- importação: $c_k, q_{ik} > 0$;
- exportação: $c_k, q_{ik} < 0$.

4.1.2 Função Objetivo

A função objetivo considera o custo acumulado sobre todas as operações em um dado horizonte de planejamento. O custo está relacionado ao tempo de serviço de todos os navios. Com intuito de dar prioridades diferentes a cada navio, optou-se por complementar o fator custo com a multa por sobrestadia (*demurrage*), proporcional ao tempo em que o navio permanece em serviço além daquele definido em contrato, t_i . Assim, o custo de cada navio pode ser representado por:

$$custo_i = (s_i - t_i) \times d_i \quad (4.2)$$

onde s_i é o tempo de serviço dado por:

$$s_i = c_i - a_i \quad (4.3)$$

e o tempo de término do carregamento/descarregamento, c_i , é dado por:

$$c_i = b_i + h_i \quad (4.4)$$

A maré de atracação, b_i , é obtida através da variável de decisão, x_{ij} . A função de custo do navio i torna-se:

$$custo_i = (b_i + h_i - a_i - t_i) \times d_i \quad (4.5)$$

A partir da Equação 4.5, foi desenvolvida a função objetivo na Equação 4.6, levando em conta que b_i é em função de x_{ij} e pode ser encontrado pelo cálculo por meio de uma progressão aritmética. A função objetivo representa o custo total, considerando todos os $|N|$ navios no horizonte de planejamento de $|M|$ TTW's.

$$\min \sum_{i=1}^N \sum_{j=1}^M j \frac{d_i}{h_i} x_{ij} + \sum_{i=1}^N d_i \left(\frac{1}{2} h_i - a_i - t_i + \frac{1}{2} \right) \quad (4.6)$$

Pode-se observar que o custo total é expresso no forma $\mathcal{A}x + \mathcal{C}$, onde o termo \mathcal{C} é necessário para obter o valor de custo real, mas não para decidir a programação dos navios. O termo \mathcal{A} representa uma mediana TTW de atracação, $j \frac{x_{ij}}{h_i}$, ponderando a prioridade do navio, d_i .

4.1.3 Restrições

O modelo proposto considera restrições rígidas expressas pelas equações 4.7 a 4.11. As equações 4.7 e 4.8 impõem que navios devem atracar após as respectivas marés de chegada, a_i , e permanecem carregando/descarregando durante h_i TTW's para completar a operação. Sem a restrição na Equação 4.7, a situação impossível de um navio atracar antes mesmo de chegar ao porto seria pertimida.

O número de navios alocados a uma dada TTW não pode exceder o total de posições de atracação, $|L|$ (Equação 4.9). A homogeneidade dos berços possibilita

a simplificação da decisão de *onde* alocar os navios. Na realidade, esta decisão é irrelevante, alocam-se navios a medida que berços são disponibilizados.

$$\sum_{j=1}^{a_i-1} x_{ij} = 0, \quad \forall i \in N \quad (4.7)$$

$$\sum_{j=a_i}^M x_{ij} = h_i, \quad \forall i \in N \quad (4.8)$$

$$\sum_{i=1}^N x_{ij} \leq |L|, \quad \forall j \in M \quad (4.9)$$

A Equação 4.10 previne a interrupção da operação de um navio com baixo *demurrage* para substituição por um outro de alto *demurrage*, ou seja, de maior prioridade. Uma vez iniciado o atendimento de um navio, este não deve ser interrompido por nenhuma razão, incluindo vantagens no custo final das operações. Nas operações em portos reais, interrupções de serviço são altamente indesejadas, mas podem ocorrer devido a muitos fatores como mau funcionamento de equipamentos e perdas ou danos, porém nunca para priorizar navios. Do ponto de vista do planejamento, interrupções de serviço são imprevisíveis e, em muitos casos, a simples resolução do cenário atualizado é suficiente para contornar este problema. Através da Equação 4.10, 1's não consecutivos, que representariam interrupções em operações, são eliminados. Do contrário, a solução resultante seria inviável.

$$\sum_{l=1}^{j-1} x_{il} - jx_{ij-1} + jx_{ij} \leq j, \quad \forall i \in N, \forall j > 1 \in M : \quad (4.10)$$

Por fim, na Equação 4.11, as condições de níveis de estoque são consideradas, anulando-se sequências de atracções nas quais o nível de qualquer carga manipulada atinja um valor abaixo de um determinado limiar pré-definido. Em refinarias industriais, níveis de armazenagem seguros são usados como limiares para garantir que ações de emergência possam ser executadas imediatamente. Como nessas refinarias é comum operar com baixos níveis de armazenagem, a verificação dos estoques precisa ser realizada em janelas de tempo estreitas. Neste trabalho, o controle de estoque é considerado a cada TTW e os limiares para todas as cargas são definidos como 0.

$$\sum_{i=1}^N \sum_{l=1}^j \frac{q_{ik}}{h_i} x_{il} \geq jw_k - e_k, \quad \forall j \in M, \forall k \in K \quad (4.11)$$

Desta forma, o modelo completo pode ser resumido pelas equações (4.12)-(4.17):

$$\min \sum_{i=1}^N \sum_{j=1}^M j \frac{d_i}{h_i} x_{ij} + \sum_{i=1}^N d_i \left(\frac{1}{2} h_i - a_i - t_i + \frac{1}{2} \right) \quad (4.12)$$

$$\sum_{j=1}^{a_{i-1}} x_{ij} = 0, \quad \forall i \in N \quad (4.13)$$

$$\sum_{j=a_i}^M x_{ij} = h_i, \quad \forall i \in N \quad (4.14)$$

$$\sum_{i=1}^N x_{ij} \leq |L|, \quad \forall j \in M \quad (4.15)$$

$$\sum_{l=1}^{j-1} x_{il} - jx_{ij-1} + jx_{ij} \leq j, \quad \forall i \in N, \forall j > 1 \in M : \quad (4.16)$$

$$\sum_{i=1}^N \sum_{l=1}^j \frac{q_{ik}}{h_i} x_{il} \geq jw_k - e_k, \quad \forall j \in M, \forall k \in K \quad (4.17)$$

4.2 Berços heterogêneos

Nesta Seção, é apresentada a modelagem do problema com terminal com berços heterogêneos. A questão da heterogeneidade dos berços está ligada à infraestrutura que estes comportam. Infraestruturas diferentes resultam em vazões, ou velocidades de atendimento, diferentes. O diferencial de vazão é comumente obtido equipando o berço com dois ou mais carregadores de navios. Deve-se ressaltar que a heterogeneidade dos berços, para este trabalho, não se estende às limitações de um navio atracar em determinado berço devido a falta de calado, por exemplo, ou outros fatores ligados a navegação. Supõe-se também cada berço pode movimentar com a mesma vazão todos os tipos de cargas.

O modelo foi evoluído a partir daquele que lida com berços homogêneos (ver Seção 4.1). Assim, algumas suposições feitas para o primeiro caso, que não são refeitas nesta Seção, permanecem válidas.

As variáveis de decisão são dadas por x_{ij} , u_{il} e y_{ijl} :

$$x_{ij} = \begin{cases} 1 & \text{se o navio } i \text{ estiver alocado a TTW } j \\ 0 & \text{de outra forma} \end{cases} \quad (4.18)$$

$$u_{il} = \begin{cases} 1 & \text{se o navio } i \text{ estiver alocado ao berço } l \\ 0 & \text{de outra forma} \end{cases} \quad (4.19)$$

Como no caso anterior, as decisões a serem tomadas são de qual subconjunto de TTW's cada navio i deve ser alocado, representadas pela variável x_{ij} . A principal diferença entre os dois problemas é na existência da decisão de *onde* atracar o navio. Cada berço tem vazão diferente e a decisão afeta o custo final da operação no horizonte de planejamento considerado. A variável u_{il} determina se o navio i atracou no berço l , mas, ao contrário de x_{ij} , para cada i , admite-se apenas um l tal que $u_{il} = 1$, indicando que apenas um berço pode ser escolhido pelo navio.

Para associar as variáveis x_{ij} e u_{il} sem tornar o modelo não-linear, utiliza-se uma técnica onde cria-se uma variável auxiliar y_{ijl} . Essa técnica é baseada na idéia de contigência de variáveis e é implementada pelas equações 4.29 a 4.31.

4.2.1 Dados de entrada

Os dados de entrada para o PAB neste problema são:

- N : conjunto de navios;
- M : conjunto de TTWs;
- L : conjunto de posições de atracação (berços);
- K : conjunto de matérias-prima operadas no porto;
- a_i : TTW de chegada do navio i ;
- v_l : velocidade de carregamento/descarregamento do berço l ;
- e_k : nível de estoque inicial da carga de granel k ;
- c_k : quantidade de consumo ou produção da carga de granel k ;
- q_{ik} : capacidade de carga do navio i em relação a carga k .

Outro parâmetro usado no modelo, h_{il} , diz respeito ao tempo de processamento necessário para um navio i completar a operação de carga/descarga em um berço l . Este parâmetro, entretanto, é definido em função de outros e pode ser obtido através da seguinte expressão:

$$h_{il} = \left\lceil \frac{\sum_{k=1}^{|K|} q_{ik}}{v_l} \right\rceil \quad (4.20)$$

O parâmetro h_{il} é apenas uma aproximação pois outros fatores, como tempo de inicialização do serviço, não são levados em conta. Se outros fatores forem determinantes para a definição do tempo de processamento, h_{il} continuaria sendo um parâmetro previamente calculado, que não prejudicaria o formato do modelo.

4.2.2 Função Objetivo

No caso heterogêneo, o objetivo continua sendo minimizar o custo acumulado sobre todas as operações em um horizonte de planejamento dado. Nesta abordagem, porém, optou-se por não considerar o *demurrage* diferenciado como fator de priorização de navios. Sendo assim, apenas o tempo de serviço total é utilizado, obtido com base na utilização de TTW's por cada navio. O tempo de serviço é dado por:

$$s_i = \sum_{j=a_i}^M \sum_{l=1}^L \left\lceil \frac{j - a_i + 1}{h_{il}} \right\rceil \times y_{ijl} \quad (4.21)$$

Minimizar o somatório dos tempos de serviço, s_i , de cada navio corresponde, portanto, à função objetivo:

$$\min \sum_{i=1}^N \sum_{j=a_i}^M \sum_{l=1}^L \left\lceil \frac{j - a_i + 1}{h_{il}} \right\rceil \times y_{ijl} \quad (4.22)$$

4.2.3 Restrições

As restrições do modelo são expressas pelas equações 4.23 a 4.31. As equações 4.23 e 4.24 obrigam os navios a atracarem após suas TTW's de chegada, a_i , e permanecerem em operação durante h_{il} TTW's para completar a operação. O

número de navios alocados a uma TTW não pode exceder o total de posições de berços, $|L|$ (Equação 4.25).

$$\sum_{i=1}^{a_i-1} x_{ij} = 0, \quad \forall i \in N \quad (4.23)$$

$$\sum_{j=a_i}^M x_{ij} = \sum_{l=1}^L h_{il} \times u_{il}, \quad \forall i \in N \quad (4.24)$$

$$\sum_{i=1}^N y_{ijl} \leq 1, \quad \forall j \in M, l \in L \quad (4.25)$$

$$\sum_{l=1}^L u_{il} = 1, \quad \forall i \in N \quad (4.26)$$

A Equação 4.27 evita soluções com interrupções prematuras nas operações do navio.

$$\sum_{z=1}^{j-1} x_{iz} - j \times x_{ij-1} + j \times x_{ij} \leq j, \quad \forall i \in N, j > a_i \in M \quad (4.27)$$

Através da Equação 4.28, os níveis mínimos de estoque maiores que zero são garantidos.

$$\sum_{i=1}^N \sum_{z=1}^j \sum_{l=1}^L \frac{q_{ik}}{h_{il}} \times y_{izl} \geq j \times w_k - e_k, \quad \forall j \in M, k \in K \quad (4.28)$$

As equações 4.29 a 4.31 garantem a consistência do modelo. O objetivo é obter $y_{ijl} = 1$ quando $x_{ij} = 1$ e $u_{il} = 1$, e $y_{ijl} = 0$ quando pelo menos uma das variáveis, x_{ij} ou u_{il} , seja igual a 0.

$$y_{ijl} \geq x_{ij} + u_{il} - 1 \quad \forall i \in N, j \in M, l \in L \quad (4.29)$$

$$y_{ijl} \leq x_{ij} \quad \forall i \in N, j \in M, l \in L \quad (4.30)$$

$$y_{ijl} \leq u_{il} \quad \forall i \in N, j \in M, l \in L \quad (4.31)$$

4.3 Variações no modelo

Os modelos apresentados são fiéis às principais características do problema. Os níveis mínimos de estoque devem ser mantidos durante todo o horizonte de planejamento, e as condições de maré delimitam as janelas de atracação. Nos cenários apresentados, várias suposições foram feitas baseadas no conhecimento sobre a operação de navios no complexo portuário de São Luís. Entretanto, essas suposições podem ser estendidas para outros cenários e, em um problema real, muitos detalhes devem ser incorporados.

Nesta Seção, são apresentadas algumas variações ou possíveis extensões para os dois modelos apresentados. Para tanto, toma-se como base o segundo modelo (ver Seção 4.2), tendo em vista que este abrange a maioria dos cenários reais (ver Seção 4.1).

4.3.1 Variações de função objetivo

A função objetivo pode ser formulada utilizando-se como critério o *demurrage*, discutido no primeiro modelo. Para simplificação, o termo constante pode ser mais uma vez ignorado tendo em vista que não influencia o processo de decisão. Desta forma, tem-se:

$$\min \sum_{i=1}^N \sum_{j=1}^M \sum_{l=1}^L j \frac{d_i}{h_{il}} y_{ijl} \quad (4.32)$$

A formulação difere daquela expressa na Equação 4.6 apenas pela inclusão de um somatório sobre o universo de berços, e as variáveis relevantes para o cálculo do custo são y_{ijl} , em vez de x_{ij} .

Na Figura 4.2, primeiramente, supondo que os navios A e B tem os mesmos parâmetros, as soluções têm significados diferentes para o custo final calculado pela Equação 4.32. A linha tracejada representa a TTW limite para o término de ambos os navios sem a cobrança de multa. As *folgas* no atendimento não deveriam fazer diferença para o custo final, afinal, o limite contratual não é violado. Apesar de não existir penalização prevista para as duas situações, navios com maior *demurrage* devem ser priorizando. Sendo assim, o navio A tem custo menor que B, segundo a Equação 4.32.

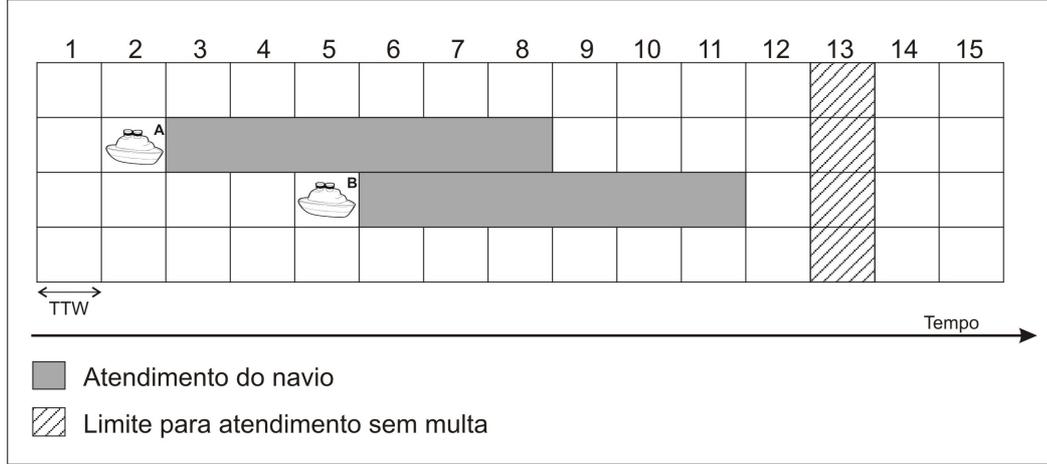


Figura 4.2: Duas soluções diferentes com custos teoricamente iguais

Como já foi dito na Seção 4.1, esta formulação representa uma mediana TTW de atracação ponderada pelo *demurrage*, d_i , do navio. No entanto, o real sentido de *demurrage* é a penalização ou multa aplicada ao porto por atrasos no atendimento do navio segundo um contrato pré-estabelecido. Assim, só o tempo acima do contratado deve ser penalizado.

Uma outra formulação para minimização do *demurrage* é proposta. A quantidade de TTW's a mais, δ_i , necessárias para o atendimento sobre a TTW máxima contratual, ζ_i , ponderada pelo *demurrage*, d_i , determina o custo de cada navio.

$$\text{custo}_i = \delta_i \times d_i \quad (4.33)$$

onde d_i é em função de y_{ijl} e pode ser obtido através de:

$$\delta_i = \sum_{j=\zeta_i+1}^M \sum_{l=1}^L \left[\frac{j - \zeta_i}{h_{il}} \right] \times y_{ijl} \quad (4.34)$$

sendo ζ_i previamente calculado por:

$$\zeta_i = a_i + t_i - 1 \quad (4.35)$$

Apesar das duas soluções na Figura 4.2 não terem custo algum em relação ao *demurrage*, é sempre desejável que o navio seja atendido o quanto antes. Assim, uma segunda parte tem como objetivo minimizar a folga ρ_i entre a_i e a TTW de atracação do navio i . Isto é possível através da maximização do *lucro* formado pelo

valor pago por cada TTW de folga antes de ζ_i (*dispatch*) ou, mesmo, por qualquer mecanismo onde é atribuída uma penalidade à função objetivo proporcional ao tamanho de ρ_i . O lucro é dado por:

$$\text{lucro}_i = \rho_i \times di_i \quad (4.36)$$

onde di_i é o *dispatch*. ρ_i pode ser obtido pela seguinte equação:

$$\rho_i = \sum_{l=1}^L \sum_{j=a_i}^{\xi_{il}-1} \left\lceil \frac{\xi_{il} - j}{h_{il}} \right\rceil \times y_{ijl} \quad (4.37)$$

onde ξ_{il} é a última TTW na qual o navio i deve entrar para não haver multa. Esta posição depende do berço de atracação do navio pois os berços possuem velocidades de operação diferentes. Sendo assim, é possível que haja relação navio/berço onde o *dispatch* nunca possa ser pago. ξ_{il} é dado por:

$$\xi_{il} = \zeta_i - h_{il} + 1 \quad (4.38)$$

A função objetivo representa a minimização do custo _{i} e maximização do lucro _{i} , tomando todos os $|N|$ navios no horizonte de planejamento de $|M|$ TTWs.

$$\min \sum \left(\sum_{j=\zeta_i+1}^M \sum_{l=1}^L \left\lceil \frac{j - \zeta_i}{h_{il}} \right\rceil \times d_i \times y_{ijl} - \sum_{l=1}^L \sum_{j=a_i}^{\xi_{il}-1} \left\lceil \frac{\xi_{il} - j}{h_{il}} \right\rceil \times di_i \times y_{ijl} \right) \quad (4.39)$$

Uma outra abordagem para a função objetivo também merece destaque: *makespan*. *Makespan* é um termo utilizado em Pesquisa Operacional, geralmente, associado à diferença entre os tempos de início e fim de uma sequência de tarefas. Considerando navios como clientes, o *makespan* é medido em função do tamanho da janela de tempo entre a atracação do primeiro navio até a desatracação do último num horizonte de planejamento. A idéia, então, é atender todos os navios na menor janela possível. Pode-se inferir que a minimização dos tempos de serviço de todos os navios, expresso pela Equação 4.21, não necessariamente implica na minimização do *makespan* do porto.

Na Equação 4.40, a função objetivo considerando o *makespan* é apresentada. A principal característica dessa formulação é o acréscimo de mais uma variável, $z_j, \forall j \in M$.

$$\min \sum_{j=1}^M j z_j \quad (4.40)$$

O cálculo do *makespan*, da forma proposta na Equação 4.40, pressupõe que o primeiro navio é sempre atendido no instante zero ($TTW = 1$) do horizonte de planejamento em questão. A variável z_j pode assumir valores 1 ou 0, mas devido às restrições a seguir, apenas o z_j da última TTW na qual há navio em serviço vai assumir o valor 1, cabendo ao coeficiente j o peso apropriado.

Acrescenta-se ainda mais duas restrições para garantir que apenas um único $z_j = 1$ (Equação 4.41) seja suficiente para equilibrar a restrição na Equação 4.42.

$$\sum_{j=1}^M \sum_{i=1}^N 2^j \left\lceil \frac{x_{ij}}{L} \right\rceil - \sum_{j=1}^M \sum_{l=1}^j 2^j z_j \leq 0 \quad (4.41)$$

$$\sum_{j=1}^M z_j = 1 \quad (4.42)$$

Por fim, considera-se ainda a possibilidade de um modelo multi-objetivo, agregando todos os critérios de desempenho apresentados. Afinal, em um cenário operacional real este conjunto de objetivos são almejados direta ou indiretamente, cabendo ao especialista determinar sua ordem de prioridade.

4.3.2 Restrições na utilização de equipamentos

As paradas corretivas, causadas, em geral, por quebra de equipamentos, ao contrário das programadas para manutenção, são inesperadas e podem causar grandes transtornos às operações do porto. Por esse motivo, pode-se incluir uma restrição de capacidade máxima de utilização para cada berço, contabilizando o número de TTW's associadas a cada um deles e inviabilizando programações que extrapolem a capacidade. Um berço, dessa forma, seria considerado um recurso finito, requerendo parada preventiva programada sempre que esgotado. Um efeito secundário dessa restrição seria equilibrar o uso de todos os berços, evitando sobrecarga e conseqüente quebra de equipamentos.

4.3.3 Restrições de berços e temporais no atendimento de navios

Uma restrição comum em problemas de alocação de berços é a impossibilidade de um berço de atender a determinados navios. Isto é causado por questões de infra-estrutura do berço, que não opera determinada carga, ou condições ruins de navegação, como pouca profundidade, por exemplo.

É simples alterar o modelo para satisfazer estas limitações. Basta determinar que certo navio não seja atendido por um ou mais berços, atribuindo 0 a respectiva variável ($u_{il} = 0$). Apenas uma restrição é necessária para evitar todas as atracações indevidas, como na Equação 4.43.

$$\sum_{i=0}^N \sum_{j=0}^M p_{il} \times u_{il} = 0 \quad (4.43)$$

onde p_{il} é um parâmetro binário que determina se o navio i não é atendido pelo berço l , sendo $p_{il} = 1$. Caso contrário, $p_{il} = 0$.

Por um mecanismo similar pode-se inviabilizar associações de navios a certas TTWs, $x_{ij} = 0$, nas quais, por exemplo, o calado do navio ou o horário de atracação (manobra noturna) inviabilizem a atracação.

4.4 ATP/PL aplicado ao problema

A primeira questão que surge na aplicação deste método é como o problema será decomposto. A reformulação do problema para uso do ATP/PL consiste em identificar quais conjuntos de variáveis representariam uma coluna (variável) para o PMR. Além disso, é necessário a formulação de um PPC, para o PMR, fundamental para a utilização da técnica.

Assim, uma coluna $p \in P'$ foi definida como uma sequência de atracações de navios em um determinado berço l , como pode ser visto na Equação 4.44.

$$p = \{l, \{i^1, i^2, \dots, i^k\}\}, \quad (4.44)$$

onde P' é um subconjunto do conjunto P de índices das colunas ($P' \subseteq P$) e k representam a ordem de atracação dos navios i presentes em p .

Cada coluna gerada é uma solução do ATP, ou seja, um indivíduo. Na ilustração abaixo, o primeiro valor significa o berço que este atracou e, os valores restantes, a sequência de atracação dos navios.

l	$i + 2$	i	$i + 5$
-----	---------	-----	---------

4.4.1 Problema Mestre Restrito

O PMR foi formulado como um PPC⁺ porque contém restrições adicionais que mantêm as características de preservar os níveis de estoque e a utilização de um berço apenas uma vez na solução.

As variáveis de decisão são denotados por λ_p e indicam se a coluna p foi usada na solução.

$$\lambda_p = \begin{cases} 1, & \text{se a coluna } p \text{ faz parte da solução} \\ 0, & \text{caso contrário} \end{cases}$$

Uma questão importante é que o PMR tem suas restrições de integralidade relaxadas, sendo estas necessárias apenas ao final das interações, na busca por uma solução inteira para o problema. Assim, no modelo relaxado, $\lambda_p \in [0, 1]$, $\forall p \in P$. Abaixo, o PMR é apresentado.

$$\min \quad z = \sum_p^{P'} c_p \lambda_p \quad (4.45)$$

sujeito a:

$$\forall i \in N, \quad \sum_p^{P'} a_{ip} \lambda_p = 1 \quad (4.46)$$

$$\forall l \in L, \quad \sum_p^{P'} b_{lp} \lambda_p \leq 1 \quad (4.47)$$

$$\forall j \in M, k \in K, \quad \sum_p^{P'} q_{jkp} \lambda_p \geq j c_k - e_k \quad (4.48)$$

onde c_p é o custo da coluna, que representa o somatório dos tempos de serviço dos navios. A restrição na Equação 4.46, que caracteriza o PPC, evita que o mesmo navio participe mais de uma vez na solução. Assim, a_{ip} é definida como:

$$a_{ip} = \begin{cases} 1, & \text{se o navio } i \text{ foi gerado para a coluna } \lambda_p \\ 0, & \text{caso contrário} \end{cases}$$

Como não pode haver mais de um berço sendo utilizado em cada maré, a restrição na Equação 4.47 deve ser inserida no PMR. A constante b_{lp} é assim definida:

$$b_{lp} = \begin{cases} 1, & \text{se a coluna } \lambda_p \text{ refere-se ao berço } l \\ 0, & \text{caso contrário} \end{cases}$$

Uma última restrição (Equação 4.48) foi inserida para preservar os níveis de estoque. Nesta restrição, q_{jkp} é a carga k carregada/descarregada por todos os navios pertencentes à sequência de atracação até a TTW j . Assim, em cada TTW j , é possível verificar se alguma carga k violou seu respectivo nível mínimo de estoque.

Modelo inicial

Como a geração das colunas é direcionada pelas variáveis duais obtidas da execução do PMR, o primeiro passo do método é a geração de um modelo de PL inicial. Pode haver uma dificuldade em identificar um conjunto de colunas que formará um modelo com solução. Assim, optou-se pelo uso de variáveis artificiais, que satisfaçam as restrições, com coeficientes da função objetivo muito ruins, garantindo que ao final das interações estas não façam parte da solução.

O número de colunas geradas é igual ao número de berços. Isto não tem um papel importante porque o principal requisito das colunas é retornar soluções válidas para o PL. Um exemplo de colunas geradas em uma instância com três berços e duas matérias-primas é apresentada na Tabela 4.1.

Um detalhe importante são os valores adotados para os valores de carga nas TTW's: bem elevados ($= 100$), escolhidos arbitrariamente para garantir que os níveis mínimos de estoque não sejam violados. Como mencionado anteriormente, os custos da função objetivo também são elevados ($= 100$), escolhidos arbitrariamente.

Heurística para solução inicial

Adicionalmente ao modelo inicial gerado com variáveis artificiais, uma heurística gulosa também foi desenvolvida para prover uma solução inicial melhor, i.e., mais próxima da solução ótima. A heurística é resumida no Algoritmo 2.

	$p = 1$	$p = 2$	$p = 3$
coef. f.o.	100	100	100
navio 1	1	0	0
navio 2	0	1	0
navio 3	0	0	1
⋮	⋮	⋮	⋮
navio i	0	1	0
berço 1	1	0	0
berço 2	0	1	0
berço 3	0	0	1
1,bauxita	100	100	100
1,alumina	100	100	100
2,bauxita	100	100	100
2,alumina	100	100	100
⋮	⋮	⋮	⋮
j ,bauxita	100	100	100
j ,alumina	100	100	100

Tabela 4.1: Exemplo de solução inicial desenvolvida com variáveis artificiais

```

1 Criar (lista N com todos os navios);
2 Ordenar (N pela ordem de chegada);
3 for cada navio  $i$  em N do
4   if alguma carga  $k$  precisa ser movimentada then
5     Atribuir (próximo navio  $i$  com carga  $k$  no berço com saída mais
        rápida);
6   else
7     Selecionar (um berço em L);
8     Atribuir (navio  $i$  ao final do berço  $l$  escolhido);
9   end
10 end

```

Algoritmo 2: Algoritmo heurístico para solução inicial

4.4.2 ATP

A cargo do ATP fica a evolução de indivíduos (colunas) que possam ser inseridos no PMR e melhorem a solução atual. Assim, um indivíduo do ATP tem a mesma estrutura de uma variável no PMR.

A geração dos indivíduos iniciais é similar ao Algoritmo 2. As diferenças são que os níveis de estoque não são considerados e a seleção dos berços é feita de forma aleatória.

A operação de cruzamento é baseada na estratégia do *ponto único*. Um ponto no meio da sequência de navios dos pais é selecionado. A primeira metade do filho é herdada do início do indivíduo base e a metade restante do final do indivíduo guia.

Na heurística de avaliação, uma busca local é executada, através de uma operação simples com os navios em um determinado número de iterações. A cada iteração, a operação de troca de posições entre dois navios é executada e seu valor de adaptação é calculado. O menor valor entre todas as iterações é considerado. Na mutação, este mesmo critério é utilizado, mas com apenas uma iteração.

4.5 *Simulated Annealing* aplicado ao problema

Por o problema apresentado ser de otimização combinatória, um algoritmo aproximativo foi proposto por colaboradores em [BARROS et al., 2009a] e [BARROS et al., 2009b] como uma alternativa válida para encontrar boas soluções para instâncias difíceis. O algoritmo *Simulated Annealing* (SA) foi escolhido para resolver rapidamente as instâncias mais difíceis devido a sua simplicidade, robustez e facilidade para implementação.

4.5.1 Berços homogêneos

Uma solução no *Simulated Annealing* proposto é representada através de uma sequência de navios que é uma ordem de serviço, não sendo necessariamente a sequência de atracação, mas a ordem de prioridade. O primeiro navio s_1 é o primeiro a ser considerado, respeitando o tempo de chegada, na programação do

berço. Os navios subsequentes na permutação podem obter um conflito de berço com os navios anteriores. Uma vez que o conflito acontece, o navio de menor prioridade é atrasado.

A solução atual, em geral, é a base para gerar nova solução no SA. Isto é feito através de um procedimento de vizinhança \mathcal{P} . A idéia básica é manter algumas características fixas e tentar mudar as outras. A solução modificada s' é considerada uma solução vizinha de s através de um movimento de \mathcal{P} . Por exemplo, um procedimento comum para soluções representadas por permutação, como $s = \{1, 2, 3, 4, 5\}$, é trocar duas posições na permutação, resultando em $s' = \{5, 2, 3, 4, 1\}$ (vizinhança 2-troca).

Em [BARROS et al., 2009b], um procedimento desenvolvido especialmente para o modelo com berços homogêneos é proposto: a *heurística de Navio de Demurrage Alto* (*High Demurrage Ship heuristic* - HDSH). HDSH é baseada no conhecimento inerente sobre o problema: um navio mal posicionado causa um *demurrage* alto. Seguindo esta idéia, uma troca de navios não-cega deveria levar em conta tais conhecimentos sobre a composição da função objetivo para tentar melhorar as soluções.

A HDSH trabalha calculando os custos de cada navio em uma dada solução. O $custo_k$ é uma contribuição que o navio s_k dá ao custo total da função objetivo e é dado pela Equação 4.5. Por exemplo, se um navio aleatório s_k atraca exatamente na maré de chegada dele, o custo é provavelmente um valor negativo, ou seja, o navio não espera por berço quando ele chega. A idéia básica é tentar melhorar a solução pela troca de um navio de custo alto por um de custo baixo na permutação.

Inicialmente, um navio qualquer com custo positivo, s_h , é escolhido, não necessariamente o mais alto. Um segundo passo é achar um outro navio aleatoriamente, s_l , para o qual $custo_l < custo_h, \forall l < h$. Como se pode ver pela condição $l < h$, a prioridade do navio é levada em conta neste procedimento, evitando o atraso de um navio que já tem um alto custo.

A Figura 4.3 ilustra a HDSH em 3 iterações, considerando uma permutação com 5 navios.

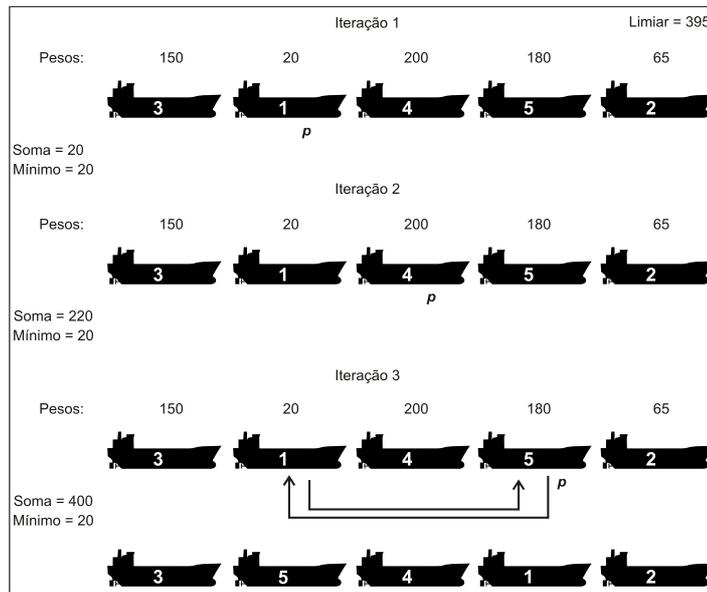


Figura 4.3: High Demurrage Ship heuristic

4.5.2 Berços heterogêneos

Nesta Seção, é apresentado o algoritmo SA proposto em [BARROS et al., 2009a]. Foi utilizado um método heurístico baseado em distribuição de navios aos berços, apresentado em [MAURI, 2008b]. Cada berço possui uma sequência de navios que é sua ordem de atendimento. Para gerar a solução inicial um navio é escolhido por vez e alocado a um berço para que seja atendido até que todos os navios estejam comportados nas filas. Depois disto os navios, em cada berço, são ordenados de acordo com suas marés de chegada.

Na heurística de alocação, se um navio i for destinado a um berço l e já existir outro sendo atendido naquela maré j , ele irá esperar na fila até que exista uma quantidade necessária de tempo para que seja atendido. Os procedimentos de distribuição e alocação dos navios podem ser vistos no Algoritmo 3.

```

1 Criar (lista L com todos os berços vazios);
2 Criar (lista N aleatória com todos os navios);
3 Ordenar (N pela ordem de chegada);
4 for cada navio i em N do
5     Selecionar (um berço em L);
6     for cada maré j em M do
7         Procurar (intervalo de marés para o navio i no berço l);
8         Atribuir (navio i ao berço l nas marés encontradas);
9     end
10 end

```

Algoritmo 3: Algoritmo heurístico de distribuição e alocação

Estrutura de vizinhança

Foram utilizados três movimentos para alterar a estrutura da solução: a) permutar navios no mesmo berço, b) permutar navios em berços diferentes e c) mover navios de um berço para outro. Estes movimentos acontecem uma vez a cada iteração. A escolha de um deles é feita aleatoriamente, mas de forma a ficar bem distribuída pelo espaço de busca.

Depois de escolhido o movimento a ser realizado, a nova solução é re-avaliada e os critérios do SA irão decidir aceitar ou não a nova solução. Quando os movimentos b) e c) são escolhidos, os navios presentes nos berços que fizeram trocas e aqueles presentes no berço para onde um navio foi deslocado são, respectivamente, re-ordenados pelas suas respectivas ordens de chegada.

Resultados

5.1 Ambiente de testes

Software e Hardware

Para derivar soluções a partir dos modelos matemáticos propostos neste trabalho, foi utilizado o *solver* comercial CPLEX [CPLEX, 2006], na sua versão 11.1. O CPLEX foi escolhido para realização dos experimentos por ter bastante aceitação no meio acadêmico, científico e industrial, sendo atualmente uma das mais importantes ferramentas no âmbito da pesquisa operacional.

As heurísticas desenvolvidas neste trabalho utilizam a linguagem de programação C++. O ambiente de desenvolvimento usado foi *Microsoft Visual C++ 2008 Express Edition*.

Para efeito de comparação, todos os testes foram realizados em um único computador com processador *Xeon 2.0GHz* com 4 núcleos e com *16GB* de memória RAM.

5.1.1 Instâncias

Para a validação do modelo, diversos testes foram realizados com dados gerados de forma semi-aleatória, correspondendo a possíveis cenários reais. A maior parte do dados foi gerada aleatoriamente, mas com características que determinam instâncias de difícil resolução, onde cada conjunto de restrições tem influência na solução. Uma diferença pequena entre toda a carga produzida e toda a carga

consumida gera níveis de estoque, em algum momento das operações, próximos aos limiares. Como resultado, decisões de ultrapassagem podem ser necessárias para evitar colapsos nos estoques.

A linguagem *GNU MathProg* [MAKHORIN, 2000] foi utilizada para criar as instâncias. Nesta Seção, são apresentados exemplos de instâncias utilizadas pelos modelos descritos nas seções 4.1 e 4.2, respectivamente.

Na Tabela 5.1, é ilustrado, de forma resumida, uma instância para o problema com terminais com berços homogêneos. São 20 navios, 70 marés e 4 tipos de matéria-prima. O conjunto de navios é definido por N , o conjunto de TTW's é definido por M e as matérias-primas são definidas em K . Na sequência, são atribuídos os valores dos parâmetros do modelo. Cada navio recebe um valor de TTW de chegada, a , as taxas de *demurrage* são atribuídas em d , h refere-se à quantidade de TTW's em atendimento e t à quantidade de TTW's contratualmente permitidas, para cada navio. Os níveis de estoque iniciais de cada matéria-prima são definidos por e e o consumo de cada matéria-prima por ck . Por fim, tem-se a relação navio por matéria-prima, q , indicando a quantidade de carga movimentada por cada navio. Em todas as instâncias, os navios só movimentam um tipo de carga.

Uma instância para berços heterogêneos é apresentada de forma resumida, na Tabela 5.2. Ela é similar ao primeiro caso, mas com algumas diferenças significativas, a começar por um conjunto de berços definido por L . Tem-se também o parâmetro v , indicando a velocidade ou vazão de atendimento de cada berço. Como d e t são parâmetros para cálculo do *demurrage*, para este problema eles não são utilizados. O tempo de atendimento h pode ser estimado através de v e q , e, por isso, não é necessário.

Dois conjuntos de instâncias foram criados. Na Tabela 5.3, as 21 instâncias usadas para o problema com berços homogêneos são apresentadas. Ela contém o número de navios, número de berços e custo mínimo encontrado pelo *solver* comercial CPLEX. Para o problema com berços heterogêneos, foram criadas instâncias de 10, 15, 20 e 30 navios, sendo 3 de 10 e de 15, uma de 20 navios e mais duas com 30 navios, variando o número de berços considerados, no total de 12 instâncias, conforme a Tabela 5.5. Nos dois casos, o número de marés necessário para servir todos os navios é estimado. Quanto menos berços ou mais navios

```

set N := 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20;
set M := 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 ... 70;
set K := Alumina Bauxita Manganes Ferro;
param a :=
1 1
2 2
:
:
20 17;
param d :=
1 37
2 25
:
:
20 33;
param h :=
1 2
2 4
:
:
20 4;
param t :=
1 4
2 5
:
:
20 6;
param e :=
Bauxita 46
Alumina 44
Manganes 46
Ferro 41;
param ck :=
Bauxita 1
Alumina 2
Manganes 1
Ferro 2;
param q : Bauxita Alumina Manganes Ferro :=
1 0 11 0 0
2 0 0 16 0
3 15 0 0 0
:
19 0 0 14 0
20 0 36 0 0;

```

Tabela 5.1: Exemplo de instância para terminais com berços homogêneos

```

set N := 1 2 3 4 5 6 7 8 9 10;
set M := 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 ... 35;
set K := Alumina Bauxita Manganes Ferro;
set L := 1 2 3;
param v :=
1 5
2 4
3 2;
param a :=
1 1
2 2
:
10 11;
param e :=
Bauxita 49
Alumina 71
Manganes 54
Ferro 57;
param ck :=
Bauxita 3
Alumina 3
Manganes 2
Ferro 2;
param q : Bauxita Alumina Manganes Ferro :=
1 6 0 0 0
2 3 0 0 0
3 4 0 0 0
4 0 5 0 0
:
8 0 0 0 18
9 0 0 11 0
10 10 0 0 0;

```

Tabela 5.2: Exemplo de instância para terminais com berços heterogêneos

houver maior deve ser a quantidade de marés, exigindo, assim, maior cuidado para evitar superestimações ou subestimações.

5.2 Experimentos Computacionais

Diversos experimentos computacionais foram realizados com o objetivo de validar os dois modelos propostos nas seções 4.1 e 4.2, por meio da análise de seu comportamento obtido mediante o desejado. Os resultados dos experimentos com o

SA também são discutidos. A discussão é dividida em duas seções: problema com terminais com berços homogêneos e com berços heterogêneos.

5.2.1 Problema com berços homogêneos

As soluções encontradas pelo *solver* são todas ótimas, incluindo a mais difícil: instância 14 de 20 navios na qual o CPLEX precisou de mais de 13 horas de execução (48.604,41 segundos).

Instância	Navios	Berços	TTW	Tempo (s)	Objetivo	Gap (%)
1	10	1	31	104,70	3926,0	0,0
2	10	1	28	0,50	3939,5	0,0
3	10	1	29	0,30	4626,5	0,0
4	10	1	32	230,45	4384,5	0,0
5	10	1	34	57,84	4603,5	0,0
6	15	1	44	12061,09	7671,0	0,0
7	15	1	47	9,64	9844,5	0,0
8	15	1	44	228,64	8202,0	0,0
9	15	1	41	1,30	8052,5	0,0
10	15	2	26	3,30	5451,0	0,0
11	20	1	58	3470,78	14671,0	0,0
12	20	1	57	12,94	15151,5	0,0
13	20	1	60	2596,34	15153,5	0,0
14	20	1	63	48604,41	16965,0	0,3
15	20	1	54	15,52	16087,0	0,0
16	30	2	44	253,83	18480,0	0,0
17	30	3	31	5,73	14456,0	0,0
18	30	1	85	1130,33	30285,0	0,0
19	30	2	47	3388,41	18264,0	0,0
20	30	2	47	2469,52	19244,0	0,0
21	30	2	48	501,00	20910,5	0,0

Tabela 5.3: Resultados obtidos pelo CPLEX para berços homogêneos

O tempo de *solver* foi incluído para dar uma noção da dificuldade de cada instância, entretanto estes tempos não devem ser estritamente considerados. O

CPLEX levou mais de 3 horas para obter a solução na instância 6, com 15 navios, por exemplo, e menos de 1 hora para atingir o valor ótimo na instância 18, com 30 navios. Apesar do número de navios, a instância 6 é definitivamente mais difícil de ser resolvida pelo CPLEX do que a instância 18.

Quanto mais berços a instância tem, mais rápido a solução é encontrada. A instância 14 precisou de mais 48.000 segundos, com apenas um berço, para ser resolvida e ainda assim com *gap* de 0,3%. Para outras instâncias não resolvidas, foram acrescentados mais berços e elas tornaram-se mais fáceis. Por último, com respeito à TTW's mínimas, minimizando o custo, o *solver* também minimiza o número de TTW's necessárias para atracar todos os navios no horizonte de planejamento.

Alguns experimentos especiais são apresentados agora para outras observações. Na Figura 5.1, a sequência de atracação de navios para a instância 17 é mostrada. Com 3 berços, esta instância toma apenas 5 segundos para ser resolvida. A alocação de navios aos berços é denotado pela associação número-símbolo. A sequência de navios para o berço 1, por exemplo, é representada pelos círculos, seguindo a linha tracejada: 1, 4, 7, 8, 15, 13, \dots , 28, 30.

O comportamento do nível de estoque ao longo do horizonte de planejamento para a instância 11 é representada na Figura 5.2 e 5.3. Os níveis de estoque de 4 matérias-primas é sempre positivo quando as restrições da Equação 4.11 são aplicadas. Sem restrições de estoque, a matéria-prima 4 assume valores negativos após a TTW 51 o que representa um colapso por falta de carga.

A instância 11 é interessante neste ponto do trabalho porque ambas as figuras são bastante similares. Examinando a solução ótima para ambos os casos, as seguintes sequências de navios podem ser extraídas:

<p>(1) 1 5 4 12 7 16 17 14 8 13 15 18 9 11 [20 3] 10 19 [6 2] (aplicando a Eq. 4.11)</p> <p>(2) 1 5 4 12 7 16 17 14 8 13 15 18 9 11 [3 20] 10 19 [2 6]</p>
--

Existem duas diferenças nas sequências: trocas entre os navios 20 e 3 e entre os navios 6 e 2. A primeira troca não afeta o nível de estoque nem a subsequência restante. Desta forma, esta troca não é relevante para o valor objetivo. Entretanto, a segunda troca causa uma pequena diferença na viabilidade e no custo ótimo. Uma vez que a Equação 4.11 é aplicada, o navio 6 é avançado para manter

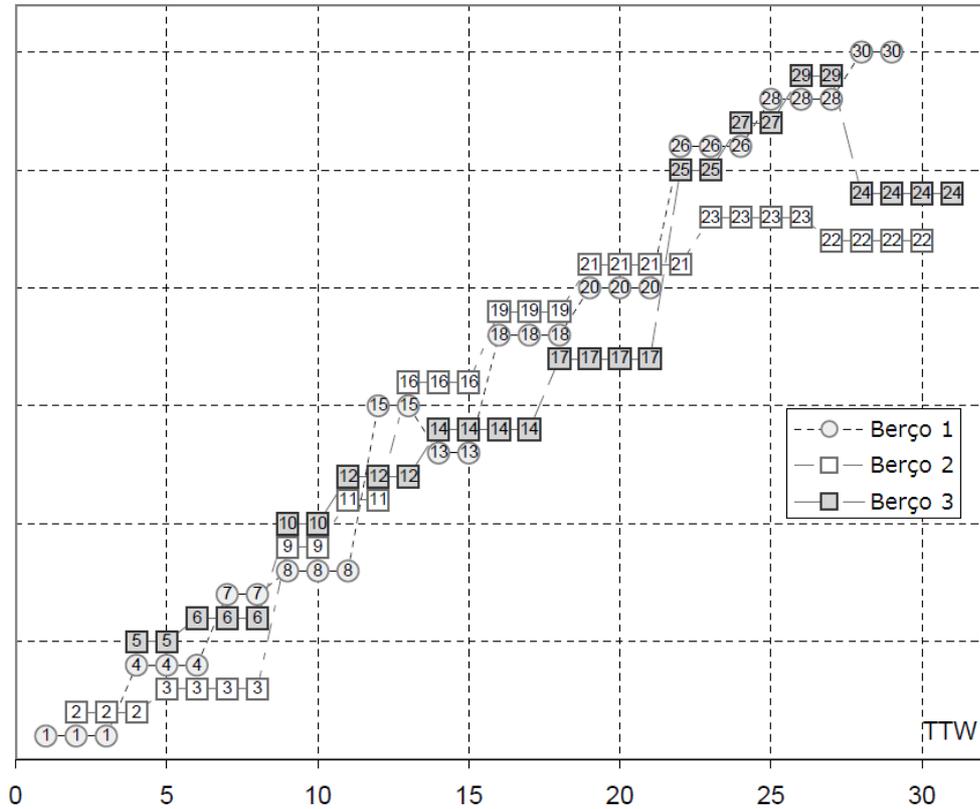


Figura 5.1: Sequência de atendimento em 3 berços

o nível de estoque positivo o que não acontece na Figura 5.3. A consequência é um aumento no custo operacional, levando o valor objetivo ótimo a 14.671 contra 14.663 quando as restrições de estoque não são aplicadas.

Simulated Annealing

Diversas execuções de teste foram feitas usando diferentes configurações dos parâmetros do *Simulated Annealing*. Nestas execuções, a temperatura e o número de iterações foram incrementadas até certos valores. O SA foi executado usando vários valores de taxa de resfriamento também. Após todas as execuções, mudando novamente os parâmetros, o algoritmo não foi melhor. Estes valores foram os melhores para a configuração completa: a) temperatura inicial, 2.000, taxa de resfriamento, 0,98, número de iterações, 1.300. Os resultados computacionais podem ser vistos na Tabela 5.4.

Para cada instância do problema, o SA foi executado 20 vezes. A coluna

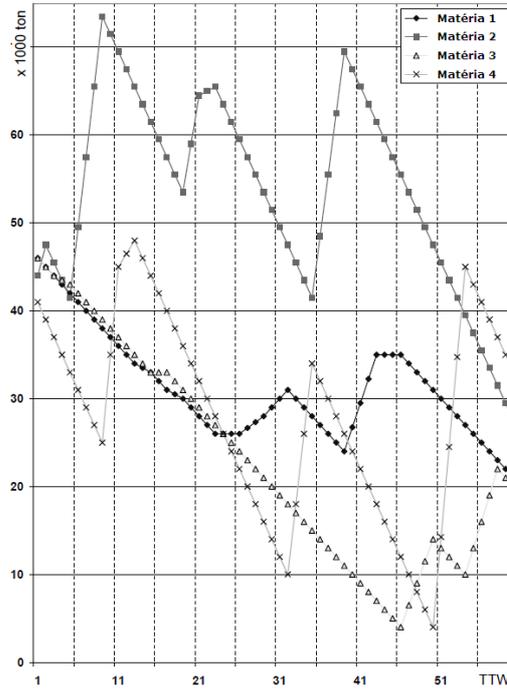


Figura 5.2: Níveis de cada matéria-prima com a restrição de estoque

Instância	Ótimo	$\overline{FS} \pm \tau^2$	$\overline{Erro}\%$	Acertos	\overline{FC}	CPLEX RT (s)	SA $\overline{RT}(s)$
1	3926	3926 ± 0	0%	20	168633.7	104.7	6.64
2	3939.5	3939.5 ± 0	0%	20	28054.5	0.5	1.95
3	4626	4626 ± 0	0%	20	3700.6	0.3	0.6
4	4384	4384 ± 0	0%	20	105408.7	230.45	4.7
5	4603.5	4603.5 ± 0	0%	20	9872.9	57.84	1.1
6	7671	7671 ± 0	0%	20	172666.6	12061.09	14.1
7	9844.5	9844.5 ± 0	0%	20	155665.2	9.64	12.7
8	8202	8202 ± 0	0%	20	178633	228.64	17
9	8052.5	8052.5 ± 0	0%	20	114368.7	1.3	9.7
10	5451	5451 ± 0	0%	20	184902.1	3.3	14.8
11	14671	14672.5 ± 2.66	0.01%	16	370444.7	3470.78	42.1
12	15151.5	15151.5 ± 0	0%	20	195914.3	12.94	28.7
13	15153.5	15153.5 ± 0	0%	20	402075	2596.34	45.4
14	16965	16971 ± 14.17	0.04%	16	359523.13	48604.41	50
15	16087	16087 ± 0	0%	20	281006.2	15.52	37.5
16	18480	18480 ± 0	0%	20	477112.2	253.83	104.2
17	14456	14456 ± 0	0%	20	174602	5.73	11.3
18	30285	30285 ± 0	0%	20	291457.8	1130.33	56
19	18264	18439 ± 216.9	0.96%	10	238275.4	3388.41	83.3
20	19244	19258 ± 8.94	0.07%	1	15223	2469.52	144.8
21	20910.5	20933.5 ± 13.33	0.11%	6	494703.5	501	125.6

Tabela 5.4: Resultados do *Simulated Annealing* para berços homogêneos

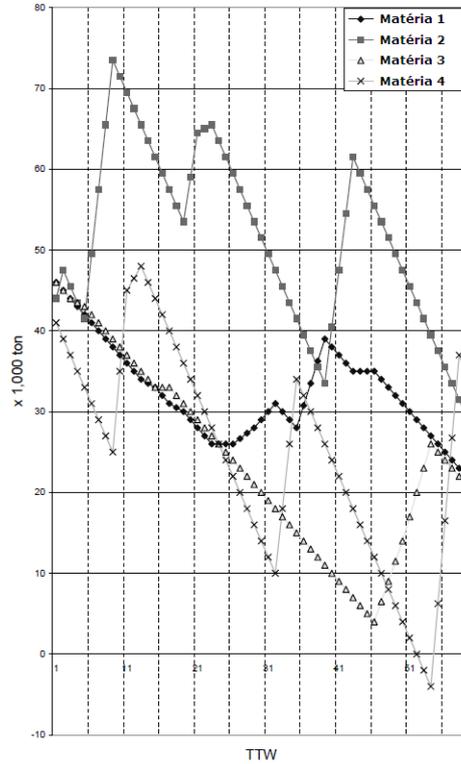


Figura 5.3: Níveis de cada matéria-prima sem a restrição de estoque

com a solução ótima relembra o valor ótimo obtido através do CPLEX. O $\overline{FS} \pm \tau^2$ refere-se à média e o desvio padrão calculado com respeito à melhor solução (FS) encontrada em cada execução. Sobre o \overline{FS} , uma medida de qualidade final, $\overline{Erro}\%$, foi levada em conta, considerando o percentual de erro médio para o SA. Na coluna com acertos, $Acertos$, é mostrado o número de vezes que as soluções ótimas foram atingidas para cada instância. O esforço computacional do SA é adicionado ao número de chamadas a função objetivo, \overline{FC} . Em relação ao tempo de execução (RT), o CPLEX e a média do SA, $\overline{RT}(s)$, estão ambas situadas nas últimas colunas da Tabela 5.4.

A média de chamadas à função objetivo e os tempos de execução foram ambos calculados considerando apenas as execuções bem-sucedidas. Por exemplo, na instância 19, os valores 238.275,4 e 83,3, respectivamente, referem-se a 10 execuções que encontraram soluções ótimas. Como pode ser observado na Tabela 5.4, o algoritmo do SA encontrou a solução ótima (16.965) para a instância mais difícil (instância 14) em 16 de 20 tentativas, com um bom tempo de execução, se

comparado ao CPLEX. Por outro lado, em algumas instâncias pequenas (instâncias 2 e 3) CPLEX foi mais rápido que o SA.

O desempenho do SA é considerado razoável tanto no tempo de execução como na qualidade das soluções. Mesmo na instância 20, para o qual o SA teve uma performance ruim, onde achou o ótimo apenas uma vez, na média, as soluções finais para esta instância foram próximas ao ótimo, dando um baixo percentual de erro (0,07%). De fato, a pior porcentagem de erro (0.96%) foi obtida para a instância 19. O SA encontrou soluções ótimas em 90% das tentativas, executando acima de 30 segundos, na média, e 2,5 minutos para o pior caso (instância 20), enquanto o CPLEX teve picos de tempo acima das 13 horas para achar a solução ótima na instância 14. Para esta instância, o SA não precisou de mais de 50 segundos para atingir a solução ótima.

5.2.2 Problema com berços heterogêneos

Na Tabela 5.5, é relacionado o conjunto de instâncias geradas com suas respectivas soluções derivadas pelo CPLEX. Ela é composta pelo número da instância, número de navios, quantidade de marés estimadas, número de berços considerados, tempo de execução, valor objetivo e foi inserido o *gap* entre a solução encontrada e a solução ótima. Nos experimentos com o CPLEX, definiu-se o tempo limite de 2 horas para as soluções serem derivadas.

Instância	Navios	Berços	TTW	Tempo (s)	Objetivo	Gap (%)
1	10	4	15	1,4	29	0,0
2	10	3	20	3,3	44	0,0
3	10	2	25	3,4	45	0,0
4	15	4	30	36,5	77	0,0
5	15	3	35	1082,6	86	0,0
6	15	2	40	1701,6	121	0,0
7	20	4	45	2684,3	105	0,0
8	30	4	70	-	-	-
9	30	3	75	28800,0	208	15,0

Tabela 5.5: Resultados obtidos pelo CPLEX para berços heterogêneos

As instâncias de 10 e 15 navios foram resolvidas pelo *solver* em tempos consi-

derados “satisfatórios”. O CPLEX precisou de menos de 4 segundos para resolver as instâncias de 10 navios. As instâncias de 15 navios demandaram mais tempo. O CPLEX exigiu do computador 1.701,6 segundos (28 minutos) para determinar a solução da instância 6, a mais difícil com 15 navios. Ainda assim, é considerado um tempo satisfatório devido a natureza do problema.

Na resolução de instâncias com 20 e 30 navios o mesmo desempenho não se repetiu. Estas instâncias exigiram muito mais do *solver*. Apenas uma instância de 20 navios (instância 7) teve solução derivada em um tempo menor que 2 horas (2.684,3 segundos). Como tentativa de derivar alguma solução viável para as instâncias de 30 navios, o limite de tempo foi aumentado para 4 horas, porém sem sucesso, exceto pela instância 9. O tempo limite de execução foi aumentado mais uma vez, desta vez para 28.800 segundos (8 horas) e ao final da execução uma solução viável havia sido encontrada, mas com *gap* de 15%.

É possível observar que os valores objetivos crescem, obviamente, com o aumento do número de navios, porém, são inversamente proporcionais ao número de berços. Quanto mais berços, menos os navios precisam esperar para serem atendidos. O tempo de execução, entretanto, é mais sensível à quantidade de marés utilizadas. O acréscimo de berços, que poderia aumentar o tempo de processamento, na prática, é reduzido devido ao menor número de marés necessárias ou estimadas. Todavia, como visto com as instâncias de 30 navios, a instância 9 com 75 TTW's derivou uma solução viável mais rápida que a instância com 70 TTW's. Assim, a cada berço incluído, como as TTW's estimadas são decrementadas, a resolução *tende* a se tornar mais rápida, mas isto não deve ser estritamente considerado.

Para efeito de validação do modelo, a instância 2 foi tomada para análise mais aprofundada. Este cenário foi escolhido por ter como solução ótima decisões que são influenciadas por todos os conjuntos de restrições.

Uma solução para o caso com berços heterogêneos pode ser representada por duas matrizes. A primeira com a relação navio por TTW, onde a decisão de quais marés o navio atracou é determinada. Na Tabela 5.6, cada navio tem uma sequência de 1's consecutivos, que dão esta informação.

Uma solução para o caso com berços heterogêneos pode ser representada por uma matriz com as atracações dos navios em um subconjunto de marés e por outra

matriz definindo em qual berço cada navio deve ser atracado. Desta forma, cada linha pode ser vista como o i -ésimo navio e cada coluna como a j -ésima TTW. Cada decisão de atracação ou não é identificada pelo valor 1 ou 0, respectivamente. Como pode ser observado na segunda coluna da Tabela 5.6, os 1's consecutivos determinam a sequência de permanência de um navio sendo atendido pelo porto. É possível perceber também que a restrição da Equação 4.10 evita as intercalações de navios, ou seja, nenhum navio interrompe prematuramente sua operação para dar lugar a outro navio, por qualquer motivo.

Navio	TTW's	Berço
1	11000000000000000000	010
2	11110000000000000000	100
3	00110000000000000000	010
4	00001111000000000000	010
5	00000110000000000000	100
6	00000000000001111110	010
7	00000001100000000000	100
8	00000000011111110000	100
9	00000000011110000000	010
10	00000000001111100000	001

Tabela 5.6: Solução: navios x TTW's x Berços

Na Tabela 5.6, é representada também a decisão de qual berço servirá cada navio, através da terceira coluna. Essa escolha é influenciada pela velocidade de operação de cada berço. Através da Equação 4.20, é possível identificar quantas marés o navio permanecerá atracado para completar sua operação. Na instância 2, os três berços 1, 2 e 3 foram definidos com velocidades 3, 4 e 2, respectivamente. É possível observar uma decisão interessante que é tomada em relação ao berço 3. Por ser o mais lento entre todos, em alguns momentos, foi preferível deixá-lo livre, mesmo havendo navio esperando para atracar, a fim de não prejudicar o tempo de serviço do respectivo navio. Portanto, mesmo prejudicando a sequência de atracação de outro berço, os tempos somados, em uma perspectiva geral, podem se tornar melhores. Na solução ótima da instância 2, o berço 3 foi utilizado apenas uma vez, pelo navio 10.

Simulated Annealing

Diversas execuções foram usadas para identificar a melhor configuração dos parâmetros. Após o final dos testes, os melhores parâmetros encontrados foram estes: a) temperatura inicial, 900, taxa de resfriamento, 0,99, número de iterações, 1000. Os resultados dos experimentos podem ser vistos na Tabela 5.7.

Instância	Ótimo	$\overline{FS} \pm \tau^2$	$\overline{Erro}\%$	\overline{FC}	CPLEX RT (s)	SA $\overline{RT}(s)$	MCE
1	29	29 ± 0	0	18.562,10	1,4	0,2	29
2	44	44 ± 0	0	41.396,30	3,3	0,4	44
3	43	43 ± 0	0	13.954	3,4	0,2	43
4	77	77 ± 0	0	436.295,60	36,5	13,3	77
5	86	86,25 ± 0,78	0,3	442.553,70	1082,6	14,2	86
6	121	121,15 ± 0,49	0,1	456.553,70	1701,6	14,3	121
7	105	105 ± 0	0	565.002	2684,3	30,7	105
8	-	189,8 ± 3,8	2,6	565.002	-	71,2	185
9	208	195,3 ± 3,9	1,7	565.002	28800,0	75	192

Tabela 5.7: Resultados do *Simulated Annealing* para berços heterogêneos

Para cada instância do problema, 20 execuções foram realizadas. O SA é configurado para parar a execução assim que são encontrados os valores ótimos conhecidos, mas existem instâncias sem solução ótima ou mesmo nenhuma solução (instâncias 8 e 9). Nestes casos, os critérios de parada foram apenas os parâmetros do algoritmo: temperatura e número máximo de iterações.

As mesmas métricas utilizadas na aplicação do SA no problema com berços homogêneos são apresentadas aqui: a solução ótima obtida pelo CPLEX, o $\overline{FS} \pm \tau^2$ é a média dos custos encontrados e o desvio padrão, $\overline{Erro}\%$ é o percentual de erro médio para o SA, \overline{FC} é o número de chamadas à função objetivo, o tempo de execução do CPLEX e a média dos tempos de execução do SA. Uma coluna a mais foi adicionada, MCE, que mostra o melhor custo encontrado pelo SA.

A média de chamadas a função objetivo e os tempos de execução foram ambos calculados considerando apenas as execuções bem-sucedidas. A coluna com o número de acertos depende da solução ótima. Nos casos onde não existe solução ótima, a melhor solução encontrada a substitui para determinar estes valores.

O SA conseguiu encontrar as soluções ótimas nas instâncias 1, 2, 3, 4 e 7 em todas as 20 execuções. Além disso, o tempo de execução para estas instâncias foi bem satisfatório, principalmente para instâncias 1, 2 e 3 (0,2 segundos, 0,4 segundos e 0,2 segundos, respectivamente). O maior tempo ficou para a instância

7 (30,7 segundos). O SA também obteve tempos satisfatórios para as outras instâncias, se comparados aos tempos obtidos pelo CPLEX. O “destaque negativo” ficou para a instância 8 (71,2 segundos).

As soluções são de boa qualidade tendo em vista que até as instâncias que o SA não obteve o ótimo tiveram um percentual de erro muito pequeno. Sobre as soluções sem o ótimo disponível, pouco se pode concluir a respeito da qualidade dos resultados obtidos. Apenas os valores pequenos de desvio padrão indicam que as soluções para estas instâncias ficaram próximas a melhor solução encontrada.

Pode se observar que a partir da instância 7, todas as execuções tiveram como critério de parada um número de iterações que resultaram na mesma média de chamadas a função objetivo (565.002). A solução do SA para a instância 9 superou bastante a solução derivada pelo CPLEX em relação ao valor objetivo encontrado. O SA obteve 192 e o CPLEX 208. Além disso, o tempo necessário pelo SA (75 segundos) foi insignificante se comparado ao tempo do CPLEX (28.800 segundos).

ATP/PL

O ATP/PL foi executado para diversas instâncias sob várias configurações. Entretanto, o resultado obtido não foi, em nenhum dos experimentos, totalmente satisfatório.

Na Tabela 5.8, são mostrados os resultados de um dos experimentos realizados. Para cada instância, 10 execuções foram feitas. A segunda coluna contém os valores ótimos conhecidos, com ressalva nas instâncias 8 e 9, que não foram obtidos através do CPLEX. Neste caso são usados como parâmetros os melhores valores encontrados pelo SA. Nas colunas $\overline{FS/PL} \pm \tau^2$ e $\overline{FS/PLI} \pm \tau^2$, tem-se a média com desvio padrão dos custos obtidos do modelo relaxado e do modelo inteiro, respectivamente. Nas colunas $\overline{RT/PL}$ e $\overline{RT/PLI}$, as médias dos tempos de execução durante o processo de geração das colunas com o modelo relaxado e o modelo inteiro, respectivamente. As médias das colunas geradas são relacionadas por \overline{COLS} .

Para as instâncias pequenas, com 10 e 15 navios (instâncias 1 a 6), foram atingidos resultados apenas razoáveis tendo em vista o baixo grau de dificuldade. Com exceção da instância 4, todas atingiram pelo menos uma vez o valor ótimo. Esta mesma instância também teve o maior percentual de erro médio (6,23), e

Instância	Ótimo	$\overline{FS} \pm \tau^2$		\overline{Erro}	\overline{RT}			\overline{COLS}	MCE
		PL	PLI		PL	PLI	Total		
1	29	27,56 ± 0	29,1 ± 0,32	0,34	11,46	0,13	11,58	403,8	29
2	44	43,71 ± 0,52	44,8 ± 0,79	1,82	27,73	0,14	27,87	262,3	44
3	43	43 ± 0	43,3 ± 0,48	0,7	13,44	0,12	13,56	290,9	43
4	77	79,4 ± 0,87	81,8 ± 2,44	6,23	48,26	0,57	48,83	397,6	79
5	86	86,79 ± 0,61	87,4 ± 1,07	1,63	43,54	0,5	44,04	589,6	86
6	121	121,76 ± 0,9	127,1 ± 6,33	5,04	53,43	0,47	53,9	615,1	121
7	105	116,41 ± 2,94	136,9 ± 11,29	30,38	74,55	8,64	83,19	826,3	117
8	185*	242,3 ± 15,34	256 ± 0	38,38	145,46	6,65	152,11	1131,7	256
9	192*	213,55 ± 4,02	305 ± 0	58,85	97,02	20,21	117,23	1400,2	305

Tabela 5.8: Resultados do ATP/PL para berços heterogêneos

o maior desvio padrão foi obtido por pela instância 6 (6,33). As execuções do modelo inteiro precisaram de menos de 1 segundo para serem finalizadas. Nos tempos totais, as médias não ultrapassaram 1 minuto de execução.

Entre as instâncias maiores (7 a 9), nenhuma obteve solução ótima em todas as execuções. Os percentuais de erro médio foram superiores a 30%. Os tempos totais de execução médio variaram entre 83,19 a 117,23 e os tempos médios para o modelo inteiro entre 6,65 e 20,21. Isto resulta, em parte, pelo número de colunas geradas ser superior às demais instâncias. À medida que as colunas são geradas, o modelo se torna maior e, assim, exige maior tempo de execução. Um detalhe importante nestes experimentos são os valores de desvio padrão da soluções finais (inteiras) das instâncias 8 e 9, que são iguais a 0. Isto acontece devido à má qualidade das colunas geradas pelo ATP. Durante toda a execução, nenhuma coluna (ou conjunto de colunas) pode melhorar a solução existente, obtida pela heurística geradora da solução inicial. Tanto 256 como 305 foram obtidos por esta heurística.

Quando a solução final obtida pelo PMR for menor ou igual ao valor ótimo, e mesmo assim quando a restrição de integralidade é inserida, mas o resultado inteiro não é o ótimo, significa que o PMR não é uma relaxação, de fato, do problema. As colunas que determinariam como este sendo a relaxação do problema não foram geradas. Assim, há duas justificativas para os resultados não satisfatórios: o número reduzido de colunas e também a qualidade das colunas inseridas.

Alguns parâmetros usados nos experimentos acima são: a) número inicial de indivíduos, 20, b) número máximo de iterações do ATP, 80, c) GMAX, 10, d) peso de GMAX, 0,1, e) *step*, 0,001, f) mutação, 0,2, g) base, 0,2, h) número máximo

de interações entre ATP e PMR, 1000. Outro critério de parada foi o número de interações sem melhora da solução, 100.

CAPÍTULO 6

Conclusão

O problema de alocação de berços em portos graneleiros com restrições de estoque e condições favoráveis de maré, identificado no complexo portuário marítimo de São Luís, apresenta peculiaridades encontradas nos cenários operacionais de portos associados a grandes empresas exportadoras, tais como refinarias e mineradoras. Tais empresas, quando sediadas em São Luís, movimentam granéis em grande quantidade armazenados em seus pátios e realizam operações de atracação e desatracação de navios apenas em condições favoráveis de maré. A literatura especializada em problemas de portos graneleiros, por sua vez, é pouco trabalhada em virtude da maior movimentação de cargas no mundo ser destinada aos contêineres.

A modelagem matemática é o mecanismo adotado para formalizar o problema neste trabalho. Dois modelos são apresentados como propostas para esta formalização. O primeiro modelo trata de casos específicos quando os berços possuem a mesma infraestrutura e, assim, são ditos homogêneos. O segundo considera que os berços podem apresentar diferenças em termos de equipamentos, tais como o número de carregadores de navios e, por esse motivo, possuírem velocidades ou vazões de atendimento diferentes. Apesar de tratar-se da abordagem de um caso particular, várias suposições são necessárias para a descrição do problema, além de variações na modelagem que também são apresentadas, mostrando a flexibilidade do modelo proposto.

Para derivação das soluções a partir dos modelos, um *solver* comercial foi usado. Testes foram realizados e os resultados discutidos e validados. A análise

dos experimentos mostra que o comportamento obtido por meio dos modelos matemáticos foi condizente com o exposto na descrição do problema. Entretanto, como o problema de alocação de berços é abordado como um problema de otimização combinatória, o *solver* é ineficaz quanto ao tempo de resolução para as instâncias maiores dos problemas.

Algoritmos heurísticos baseados no *Simulated Annealing* (SA) e no Algoritmo de Treinamento Populacional para Programação Linear (ATP/PL) são apresentados para obtenção de soluções em tempos hábeis para as instâncias maiores. O SA, uma metaheurística não populacional, foi escolhida por ser relativamente robusta e de simples implementação, sendo desenvolvidas versões para cada modelo. O ATP/PL é uma abordagem evolutiva recente que pode ser combinada com esquemas de geração de colunas para se obter soluções aproximadas de qualidade. Os experimentos com o SA mostraram resultados satisfatórios. Todas as instâncias puderam ser resolvidas em tempos hábeis (pouco mais de 2 minutos) e a taxa de erro foi pequena. Nos experimentos realizados, o ATP/PL não foi tão eficiente quanto o SA. Para instâncias menores, o resultado foi considerado satisfatório, mas, para as instâncias restantes, os resultados não alcançaram os valores desejados. Acredita-se que o ATP/PL necessite de mais ajustes nos operadores e parâmetros utilizados.

Como extensão deste trabalho, pretende-se ampliar a pesquisa sobre problemas relacionados a portos graneleiros, em especial, a integração do problema de alocação de berços ao problema de distribuição de granéis nos pátios. No que tange metaheurísticas, entende-se que algoritmos evolutivos associados a técnicas de geração de colunas formam uma opção promissora para tratar instâncias maiores do problema abordado. Dependendo do horizonte de planejamento em questão, ou da movimentação de navios em expansão, pode-se obter instâncias com mais de 30 navios facilmente, o que vai requerer abordagens heurísticas cada vez mais robustas e velozes que possam ser incorporadas a *softwares* de apoio ao planejamento de operações.

Referências Bibliográficas

- [ALUMAR, 2009] CONSÓRCIO DE ALUMÍNIO DO MARANHÃO (ALUMAR). Disponível em: <http://www.alumar.com.br>, acessado em: 21 de setembro de 2009.
- [BARNHART, 1998] BARNHART, C., JOHNSON, E.D., NEMHAUSER, G.L., SAVELSBERGH, M.W.P. e VANCE P.H. (1998) Branch and Price: Column Generation for Solving Huge Integer Programs, *Operations Research*, 46, 316-329.
- [BARROS et al., 2009a] BARROS, V. H., COSTA, T.S., OLIVEIRA, A.C.M., LORENA, L. A. N. (2009) Problema de alocação de berços heterogêneos em portos graneleiros com restrições de estoque e condições favoráveis de maré. *Simpósio de Engenharia de Produção*, UNESP, Bauru, SP.
- [BARROS et al., 2009b] BARROS, V.H., COSTA, T.S., OLIVEIRA, A.C.M., LORENA, L.A.N. (2009) Model and Heuristic. Berth Allocation in Tidal Bulk Ports with Stock Level Constraints. *Computer & Industrial Engineering* (Submetido).
- [BAZARAA et al., 1990] BAZARAA, M.S., JARVIS, J.J., SHERALI, H.D. (1990) *Linear Programming and Network Flows* (2nd Ed.), Wiley, N. Y.
- [BROWN, 1994] BROWN, G.G., LAWPHONGPANICH, S., THURMAN, K.P. Optimizing ship berthing. *Naval Research Logistics*, v. 41, 1-15, 1994.
- [CHM, 2010] CENTRO DE HIDROGRAFIA DA MARINHA (CHM). Disponível em: <http://www.mar.mil.br/dhn/chm/tabuas/index.htm>, acessado em: 20 de janeiro de 2010.

- [CORDEAU et al., 2005] CORDEAU, J.F., LAPORTE, G., LEGATO, P., MOC-
CIA, L. (2005) Models and Tabu Search Heuristics for the Berth Allocation
Problem. *Transportation Science* 39, 526-538.
- [CORRÊA e LORENA, 2006] CORRÊA, F.A. e LORENA, L.A.N. (2006) Apli-
cação da Relaxação Lagrangeana e do Algoritmo Genético Construtivo na
Solução do Problema Probabilístico de Localização-Alocação de Máxima
Cobertura. *Gestão & Produção*, v. 13, n. 2, p. 233-244, 2006a.
- [CPLEX, 2006] ILOG. (2006) ILOG CPLEX 10.0: user's manual. France: [s.n.],
478 p.
- [DANTZIG e WOLFE, 1960] DANTZIG, G.B. e WOLFE, P. (1960) Decomposi-
tion principle for linear programs. *Oper. Res.*, 8: 101-111.
- [DESAULNIERS et al., 2005] DESAULNIERS, G., DESROSIERS, J., SOLO-
MON, M.M. (eds). (2005) Column Generation, p. 1-32. Springer, Berlin.
- [EMAP, 2009] EMPRESA MARANHENSE DE ADMINISTRAÇÃO PORTUÁ-
RIA (EMAP). Disponível em: <http://www.portodoitaqui.ma.gov.br>, aces-
sado em: 21 de setembro de 2009.
- [FERNANDES, 2001] FERNANDES, M.G. Modelo econômico-operacional para
análise e dimensionamento de terminais de contêineres e veículos. Dissertação
de M. Sc., USP, São Paulo, SP, Brasil.
- [FURTADO, 1998] FURTADO, J.C. (1998) Algoritmo genético construtivo na
otimização de problemas combinatoriais de agrupamentos. São José dos
Campos - SP. Tese (Doutorado em Computação Aplicada) - Instituto
Nacional de Pesquisas Espaciais.
- [GILMORE e GOMORY, 1961] GILMORE, P.C. E GOMORY, R.E. (1961). A
linear programming approach to the cutting-stock problem. *Operations
Research*, 9:849-859.
- [GILMORE e GOMORY, 1963] GILMORE, P.C. E GOMORY, R.E. (1963).
A linear programming approach to the cutting-stock problem. Part II.
Operations Research, 11:863-888.

- [GOLDBARG e LUNA, 2005] GOLDBARG, M.C., LUNA, H.P.L. (2005) Otimização combinatória e programação linear: modelos e algoritmos (2a. Edição). Elsevier, Rio de Janeiro.
- [GOVERNO DO MARANHÃO, 2009] . GOVERNO DO ESTADO DO MARANHÃO. Disponível em: <http://www.ma.gov.br>, acessado em: 19 de setembro 2009.
- [IMAI et al., 2006] IMAI, A., NISHIMURA, E., PAPADIMITRIOU, S. (2006) Berthing ships at a multi-user container terminal with a limited quay capacity. *Transportation Research Part E*.
- [KIM e MOON, 2003] KIM, K.H. e MOON, K.C. (2003) Berth scheduling by simulated annealing. *Transportation Science*. v37 i6. 541-560.
- [KIRKPATRICK et al., 1983] KIRKPATRICK, S., GELLAT, D.C., VECCHI, M.P. (1983) Optimization by simulated annealing. *Science*, 220, 671-680.
- [HILLIER e LIEBERMAN, 2006] HILLIER, F.S., LIEBERMAN, G.J. (2006) *Introduction to Operations research* (8 ed.). McGraw-Hill, San Francisco, California.
- [HOLLAND, 1962] HOLLAND, J.H. (1962) Outline for a logical theory of adaptive systems. *Journal of the Association for Computing Machinery*, 3, 297-314.
- [HOLLAND, 1970] HOLLAND, J.H. (1970) Robust algorithms for adaptation set in a general formal framework, *Proceedings of the IEEE Symposium on Adaptive Processes - Decision and Control* 17, 5.1-5.5.
- [HOLLAND, 1973] HOLLAND, J.H. (1973) Genetic algorithms and the optimal allocation of trials, *SIAM J. Comput.* 2 88-105.
- [HOLLAND, 1975] HOLLAND, J.H. (1975) *Adaptation in Natural and Artificial Systems*. University of Michigan Press, Ann Arbor, MI.
- [LIM, 1998] LIM, A. (1998) The berth planning problem. *Operations Research Letters*. v22. 105-110.

- [LORENA e FURTADO, 2001] LORENA, L.A.N., FURTADO, J.C. (2001) Constructive genetic algorithm for clustering problems. *Evolutionary Computation*, v. 9, n. 3, p. 309-327.
- [LORENA et al., 2003] LORENA, L.A.N., SENNE, E.L.F. e SALOMÃO, S.N.A. (2003) A relaxação lagrangeana/surrogate e o método de geração de colunas: novos limitantes e novas colunas. *Pesquisa Operacional*, Rio de Janeiro, v. 23, n.1, p. 29-47.
- [LORENZONI et al., 2006] LORENZONI, L.L., Ahonen, H., Alvarenga, A.G. (2006) A multi-mode resource-constrained scheduling problem in the context of port operations, *Comput. Ind. Eng.* v50 1, 55-65.
- [MAKHORIN, 2000] MAKHORIN A. (2000) Modeling language GNU MathProg. Relatório técnico, Moscow Aviation Institute.
- [MAURI, 2008a] MAURI, G.R., Oliveira, A.C.M. e Lorena, L.A.N. (2008) A hybrid column generation approach for the berth allocation problem. In J. van Emert and C. Cotta, editors, *EvoCOP 2008*, volume 4972 of *Lecture Notes in Computer Science*, p. 110-122. Springer.
- [MAURI, 2008b] MAURI, G.R. (2008) Novas abordagens para representação e obtenção de limitantes e soluções para alguns problemas de otimização combinatória. São José dos Campos - SP. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais.
- [MOON, 2000] MOON, K.C. (2000) A mathematical model and a heuristic algorithm for berth planning. Brain Korea 21 Logistics Team, July.
- [NISHIMURA et al., 2001] NISHIMURA, E., IMAI, A., PAPADIMITRIOU, S. (2001) Berth allocation planning in the public berth system by genetic algorithms. *European Journal of Operational Research*, 131, 282-292.
- [OLIVEIRA, 2002] OLIVEIRA, A.C.M. (2002) Treinamento Populacional em Heurísticas Aplicações em Otimização. São José dos Campos - SP. Proposta de tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais.

- [OLIVEIRA, 2004] OLIVEIRA, A.C.M. (2004) Algoritmos evolutivos híbridos com detecção de regiões promissoras em espaços de busca contínuos e discretos. São José dos Campos - SP. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais.
- [PARK e KIM, 2003] PARK, Y.M., KIM, K.H. (2003) A scheduling method for berth and quay cranes. OR Spectrum, 25, 1-23.
- [PEIXOTO, 1977] PEIXOTO, J.B. (1977) Os transportes no atual desenvolvimento do Brasil. Rio de Janeiro: Biblioteca do Exército.
- [PIMENTEL, 2005] PIMENTEL, C.M.O. (2005) Algoritmos de partição e geração de colunas para dimensionamento de lotes de produção. Dissertação de mestrado em Engenharia Industrial da Universidade do Minho, Portugal.
- [PINEDO, 1995] PINEDO, M. (1995) Scheduling: Theory, Algorithms and Systems, Prentice-Hall, Englewood Cliffs, NJ.
- [PORTOSMA, 2010] PORTOSMA. Disponível em: <http://www.portosma.com.br>, acessado em: 10 de fevereiro de 2010.
- [TAVARES e CORREIA, 1999] TAVARES, L.V., CORREIA, F.N. (1999) Otimização Linear e Não Linear: Conceitos, Métodos e Algoritmos (2a. Edição) Fundação Calouste Gulbenkian, Lisboa.
- [SALOMÃO, 2005] SALOMÃO, S.N.A. (2005) Métodos de geração de colunas para problemas de atribuição. São José dos Campos - SP. Tese (Doutorado em Computação Aplicada) - Instituto Nacional de Pesquisas Espaciais.
- [SILVA, 2008] SILVA, V.M.D. (2008), Um modelo heurístico para alocação de navios em berços. Dissertação de mestrado do Programa de Pós Graduação em Engenharia de Produção da Universidade Federal de Santa Catarina, Florianópolis.
- [VALE, 2008] Companhia Vale do Rio Doce (VALE). Disponível em: <http://www.cvrld.com.br>, acessado em: 31 de março de 2008.