

# META-HEURÍSTICA ALNS PARA RESOLUÇÃO DO PROBLEMA DE ALOCAÇÃO DE BERÇOS

**Glaydston Mattos Ribeiro**

Universidade Federal do Rio de Janeiro - UFRJ

**Geraldo Regis Mauri**

Universidade Federal do Espírito Santo - UFES

**Luiz Antonio Nogueira Lorena**

Instituto Nacional de Pesquisas Espaciais - INPE

**Gilbert Laporte**

CIRRELT / HEC Montréal

## RESUMO

O Problema de Alocação de Berços (PAB) consiste em atribuir os navios que chegam a um determinado porto para áreas de atracação disponíveis ao longo de um cais (berços). A principal decisão a ser tomada nesse processo envolve a escolha de onde e quando os navios deverão atracar. Em relação ao local de atracação, existem restrições relativas à profundidade da água (calado), ao tamanho dos navios e outras. Já em relação ao horário de atracação dos navios, as restrições podem ser expressas como janelas de tempo para conclusão de seu atendimento. Neste trabalho, são considerados os casos discreto e contínuo para modelagem do PAB e, para resolvê-los, é utilizada a meta-heurística *Adaptive Large Neighborhood Search* que utiliza como princípio básico o processo de destruição e recriação das soluções. Experimentos computacionais demonstram a qualidade das soluções obtidas em relação a diversos métodos recentes encontrados na literatura.

## ABSTRACT

The Berth Allocation Problem (BAP) consists of assigning ships to berthing areas along a quay in a port. The choice of where and when the ships shall moor is the main decision to be made in this problem. Considering the berthing position, we have restrictions related to the water depth, the size of the ships and others. In addition, restrictions related to the berthing time of the ships can be expressed as time windows for completion of their service. In this work we considered the discrete and continuous models for the BAP and to solve them we used an Adaptive Large Neighborhood Search heuristic which is based on the principle of destruction and recreation of the solutions. Computational experiments demonstrate the quality of our solutions comparing them against several recent methods found in the literature.

## 1. INTRODUÇÃO

Nos dias de hoje, a maior parte do comércio mundial é realizada por meio do transporte marítimo. Além disso, a quantidade de carga transportada, principalmente cargas containerizadas, vem crescendo significativamente. No Brasil, por exemplo, a quantidade de contêineres manipulados pelos portos passou de 7.256.292 TEU (*Twenty foot Equivalent Unit* – unidade de medida que equivale a um contêiner de 20 pés) em 2008 para 8.138.608 TEU em 2010 (UNCTAD, 2013), ou seja, um aumento médio de 6% ao ano.

É evidente a importância do transporte marítimo como um dos pilares fundamentais do crescimento econômico mundial, sendo seus serviços essenciais para ajudar a economia e as empresas competirem em nível mundial. Logo, tem-se um problema crucial, que consiste na otimização do equilíbrio entre os proprietários dos navios, que exigem serviços rápidos, e o uso econômico dos recursos disponíveis no porto (Dragovic *et al.*, 2005). Assim, com o intuito de atender tanto aos portos quanto aos proprietários dos navios, surge o Problema de Alocação de Berço (PAB) que tem por finalidade estabelecer uma sequência de atendimento dos navios para cada berço disponível no cais, visando minimizar o tempo total de operação e de espera dos navios dentro dos portos.

Com o frequente aumento do fluxo nos portos mundiais, surge uma maior dificuldade em

gerenciar navios, dificultando assim a resolução do PAB. Com isso, para competir nesse ambiente, um porto deve operar de forma eficiente (Hansen *et al.*, 2008) e, de acordo com Imai *et al.* (2003), a alocação e a programação de navios a berços têm um impacto primário na eficiência dessas operações.

Devido à complexidade do PAB, diversos trabalhos encontrados na literatura utilizam abordagens baseadas em métodos heurísticos para solucioná-lo, pois tais métodos, apesar de não garantirem a obtenção de soluções ótimas, permitem a modelagem das inúmeras restrições e características do problema de uma forma mais amena.

Assim, este trabalho tem como objetivo apresentar uma aplicação da meta-heurística *Adaptive Large Neighborhood Search* – ALNS (Ropke e Pisinger, 2006) para resolver o PAB considerando duas modelagens distintas do problema: o caso discreto e o caso contínuo. Em ambos os casos, o objetivo é baseado na minimização do tempo total gasto pelos navios dentro do porto (tempo de serviço) que, segundo Hansen *et al.* (2008), é uma função-objetivo apropriada para o PAB.

Com o intuito de verificar o desempenho da ALNS, instâncias utilizadas em diversos trabalhos recentes sobre o PAB foram consideradas. Os resultados obtidos demonstram a competitividade da meta-heurística ALNS uma vez que, para o caso discreto, as soluções ótimas sempre são encontradas para todas as instâncias e com um tempo computacional inferior ao utilizado pelos demais métodos citados na literatura, e, para o caso contínuo, a meta-heurística ALNS utiliza um tempo computacional similar aos demais métodos da literatura, porém, apresenta 10 novas soluções de um conjunto de 30 instâncias.

O restante deste artigo está organizado da seguinte forma: a Seção 2 descreve o PAB e suas principais características. Na Seção 3 é apresentada uma breve revisão da literatura. A Seção 4 apresenta em detalhes a estrutura da meta-heurística ALNS proposta. Os experimentos computacionais são apresentados na Seção 5 e algumas conclusões são sumarizadas na Seção 6.

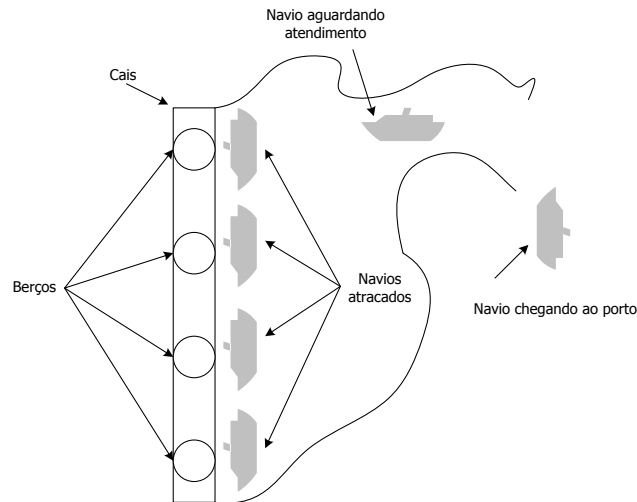
## **2. PROBLEMA DE ALOCAÇÃO DE BERÇOS**

O PAB consiste em atribuir os navios que chegam a um determinado porto para áreas de atracação disponíveis ao longo de um cais. Essas áreas são chamadas de “berços” e, então, deve-se alocar esses berços para o atendimento de diversos navios. A Figura 1 apresenta um cenário típico para o PAB.

Para alocação dos berços, deve-se definir onde e quando os navios deverão atracar. Em relação ao local de atracação, existem restrições relativas à profundidade da água (calado), à distância máxima até o local mais favorável ao longo do cais e também ao tamanho dos navios. Já em relação ao horário de atracação dos navios, as restrições podem ser expressas como janelas de tempo para conclusão de seu atendimento (Cordeau *et al.*, 2005), sendo esse tempo de atendimento dependente de seu ponto de atracação (berço). Em geral, esse tempo está relacionado à distância do berço até a área de carga e descarga de contêineres no pátio do porto, aos tipos e quantidades de equipamentos disponíveis nos berços, a variações significativas de maré, etc.

O PAB pode ser modelado de diferentes formas e, dentre estas, destacam-se os casos: estático

x dinâmico (Imai *et al.*, 2001) e discreto x contínuo (Cordeau *et al.*, 2005). No caso estático, considera-se que todos os navios já estejam no porto aguardando atendimento antes da abertura dos berços. Já no caso dinâmico, os navios podem chegar a qualquer momento ao longo do período de planejamento, porém tem-se um horário previsto para a sua chegada.



**Figura 1:** Cenário típico para o PAB (Mauri, 2008).

No caso discreto, o cais é visto como um conjunto finito de berços e, nesse caso, os berços podem ser descritos como segmentos de comprimento fixo ou como pontos se a dimensão de espaço for ignorada. Assim, apenas um navio pode atracar em cada berço em um mesmo período de tempo, independente de seu tamanho.

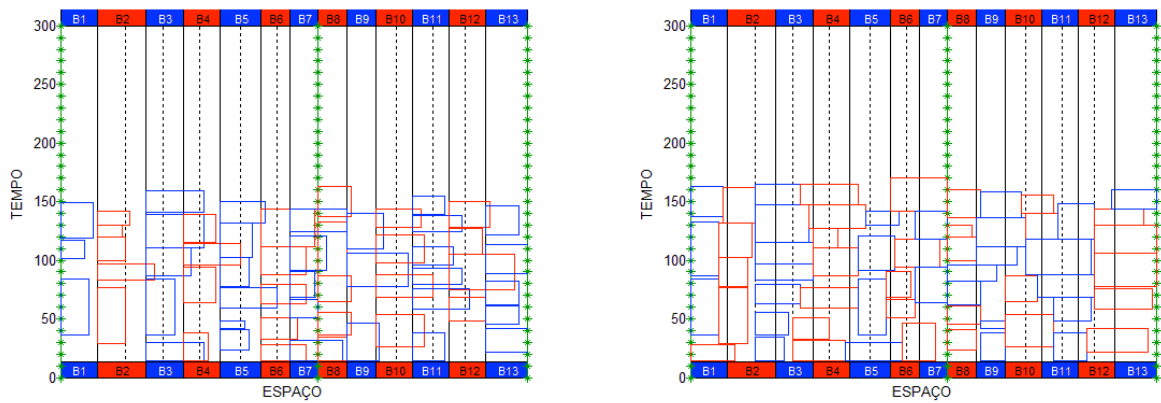
Já no caso contínuo, não há divisão do cais e assim, considera-se que os navios podem atracar em qualquer lugar. Outra concepção para o caso contínuo considera que o cais é dividido em berços, porém o tamanho dos navios e dos berços deve ser considerado. Então, os navios grandes podem ocupar mais de uma posição, ou seja, os navios pequenos podem compartilhar seu berço (Cordeau *et al.*, 2005). Essa abordagem reflete melhor as situações reais do PAB, nas quais os berços possuem diferentes equipamentos com áreas de alcance limitadas e são capazes de operar diferentes tipos de cargas.

Nos trabalhos recentes sobre o PAB, apenas o caso dinâmico é considerado, pois, na prática, o caso estático não representa de forma adequada os casos reais. Assim, neste trabalho, também será considerado o caso dinâmico, porém com as variações dos casos discreto e contínuo.

Assim como em Cordeau *et al.* (2005) e em Mauri *et al.* (2011), neste trabalho o PAB é definido em uma área bi-dimensional na qual os navios são representados como retângulos cuja largura é referente ao seu tamanho (incluindo uma margem de segurança para manobras) e a altura indica o tempo necessário para seu atendimento. Logo, o PAB pode ser redefinido como a alocação desses retângulos em um espaço pré-definido evitando a sobreposição dos mesmos. No caso discreto, como o tamanho dos navios e dos berços é ignorado, sobreposições horizontais são permitidas.

A Figura 2 apresenta duas soluções para os casos discreto e contínuo do PAB. Os berços são

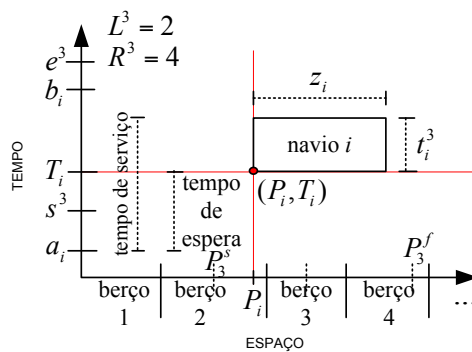
divididos em duas partes, direita e esquerda, e então cada berço  $k$  possui dois vizinhos, a parte direita do berço  $k-1$  e a parte esquerda do vizinho  $k+1$ . Descontinuidades no cais (rochas, curvas, etc. – marcas em verde) também são consideradas e, nesse caso, alguns berços podem ter apenas um vizinho, como os berços 1, 7, 8, 9 e 13 da Figura 2. As cores azul e vermelho são utilizadas apenas para facilitar a visualização. Os navios azuis estão alocados ao berço azul mais próximo e o mesmo acontece para os vermelhos. As linhas pretas sólidas e tracejadas indicam, respectivamente, os tamanhos dos berços e de suas partes (esquerda e direita). Já os retângulos preenchidos (azul e vermelho) nas partes inferior e superior da figura indicam o horário de abertura e fechamento dos berços.



**Figura 2:** Soluções para os casos discreto e contínuo do PAB (Mauri *et al.*, 2011).

Como pode ser observado na Figura 2, a solução para o caso discreto apresenta sobreposições horizontais, pois nesse caso, a dimensão dos navios e dos berços é desconsiderada. Já na solução do caso contínuo, nenhuma sobreposição ou violação nas descontinuidades é permitida. Conforme observado em Cordeau *et al.* (2005), o caso discreto do PAB é uma relaxação do caso contínuo, uma vez que as “restrições espaciais” são ignoradas. Assim, as soluções para o caso discreto são limitantes inferiores para as soluções do caso contínuo.

Neste trabalho, ambos os casos são considerados e o objetivo é o mesmo de Cordeau *et al.* (2005): minimizar a soma dos tempos em que os navios permanecem no porto. Esse tempo é chamado de “tempo de serviço”, ou seja, é o tempo desde a chegada do navio ao porto até a sua saída. A Figura 3 ilustra as principais informações para o cálculo desse objetivo.



**Figura 3:** Informações para o cálculo da função objetivo do PAB (Mauri *et al.*, 2011).

Na Figura 3, é considerada uma possível atracação do navio  $i$  no berço 3. O berço 3 abre no horário  $s^3$  e fecha no horário  $e^3$ , ou seja, o navio deverá ser atendido dentro dessa “janela de tempo” do berço. O berço 3 possui o berço 2 como vizinho à esquerda ( $L^3 = 2$ ) e o berço 4 como vizinho à direita ( $R^3 = 4$ ) e assim, parte desses berços poderá ser usada para atender o navio  $i$ , cujo tamanho é  $z_i$ , ou seja, o navio  $i$  deverá atracar entre as posições  $P_3^s$  e  $P_3^f$ . Já em relação ao tempo, o navio  $i$ , cujo tempo de atendimento no berço 3 é  $t^3_i$ , deverá ser atendido entre sua chegada ( $a_i$ ) e seu horário limite ( $b_i$ ), ou seja,  $[a_i, b_i]$  define uma janela de tempo para o navio  $i$ . Por fim, deve-se então definir o ponto  $(P_i, T_i)$ , que indica a posição e o horário de atracação do navio  $i$ . Feito isso, o tempo de serviço do navio  $i$  é dado por  $T_i + t^3_i - a_i$ . É importante destacar que tanto no caso discreto quanto no contínuo, o objetivo é o mesmo, a única diferença é que no caso discreto não é necessária a definição de  $P_i$ .

### 3. REVISÃO DA LITERATURA

Nos últimos doze anos, diversos trabalhos vêm abordando diferentes características e especificidades referentes ao PAB. Dentre esses, destacam-se aqueles que propõem novas formulações matemáticas que consideram ambos os casos discreto e contínuo e que representam situações reais. A seguir, os trabalhos mais citados e/ou recentes sobre o PAB são brevemente apresentados. Uma revisão mais detalhada sobre o problema é apresentada em Bierwirth e Meisel (2010).

Imai *et al.* (2001) apresentaram uma relaxação Lagrangiana para resolver o PAB e, dois anos depois, Imai *et al.* (2003) aprimoraram sua abordagem considerando diferentes prioridades de atendimento entre os navios e propuseram um Algoritmo Genético como método de solução. Os autores trataram também o caso contínuo do PAB em Imai *et al.* (2005).

Cordeau *et al.* (2005) consideraram tanto o caso discreto quanto o contínuo do PAB. Os autores utilizaram dados baseados no porto de *Gioia Tauro* (Itália), gerando diferentes conjuntos de instâncias, sendo que um desses conjuntos (30 instâncias diferentes com 13 berços e 60 navios) vem sendo utilizado como referência em diversos trabalhos sobre o PAB. Um método heurístico baseado na Busca Tabu foi proposto por Cordeau *et al.* (2005) para resolver os dois casos do PAB.

Mauri *et al.* (2008b) propuseram uma aplicação do *Simulated Annealing* para resolução do caso discreto do PAB. Os resultados computacionais superaram os obtidos pelo *solver* CPLEX e pela Busca Tabu proposta por Cordeau *et al.* (2005). Mauri *et al.* (2008a) apresentaram um método híbrido chamado ATP/PL para o PAB, que utiliza um algoritmo evolutivo em conjunto com um modelo de programação linear por meio da técnica de geração de colunas. Os resultados obtidos superaram os apresentados em Mauri *et al.* (2008b).

Os resultados obtidos por Mauri *et al.* (2008a) foram então superados pelos resultados de um GRASP com *Path-Relinking* proposto por Lopes *et al.* (2011) para o caso discreto do PAB. As melhores soluções obtidas por Lopes *et al.* (2011) são as ótimas para todas as instâncias consideradas, conforme provado pelo método exato proposto por Buhrkal *et al.* (2011). Esses autores resolveram o caso discreto do PAB por meio de um modelo matemático para o Problema de Roteamento de Veículos com Múltiplas Garagens e Janelas de Tempo (PRVMG-JT).

Já para o caso contínuo, as melhores soluções obtidas por Cordeau *et al.* (2005) foram

recentemente superadas por Mauri *et al.* (2011) que propuseram um Algoritmo Memético e um *Simulated Annealing* para resolução do PAB.

Por fim, no ano de 2012, Oliveira *et al.* (2012a,b) propuseram a aplicação do método *Clustering Search* (CS) para resolução dos casos discreto e contínuo do PAB. Para o caso discreto, o CS obteve as soluções ótimas para todas as instâncias. Já para o caso contínuo, o CS superou a Busca Tabu proposta por Cordeau *et al.* (2005), o Algoritmo Memético e o *Simulated Annealing* propostos por Mauri *et al.* (2011), apresentando as melhores soluções conhecidas até então.

#### **4. ADAPTIVE LARGE NEIGHBORHOOD SEARCH**

Proposto por Ropke e Pisinger (2006) e testada em vários problemas de roteamento de veículos, a meta-heurística ALNS vem apresentando bons resultados para diversos problemas de otimização combinatória.

Ropke e Pisinger (2006) utilizaram a meta-heurística ALNS para resolver o problema capacitado de roteamento de veículos com *backhauls* em que rotas são definidas para os veículos de entrega que, no retorno ao depósito, passam a realizar a coleta de mercadorias. Os autores testaram o algoritmo em 338 instâncias diferentes do problema e a meta-heurística ALNS conseguiu encontrar 227 melhores soluções.

Para o problema de roteamento de veículos com coleta e entrega e janelas de tempo, Ropke e Pisinger (2006) testaram a meta-heurística ALNS em mais de 350 instâncias com até 500 clientes e o algoritmo foi capaz de encontrar melhores soluções em mais de 50% das instâncias. Já em Pisinger e Ropke (2007), os autores aplicaram a meta-heurística ALNS em outros problemas de roteamento de veículos: no problema de roteamento de veículos com janelas de tempo, no problema de roteamento de veículos capacitado, no problema de roteamento de veículos com múltiplos depósitos, no problema de roteamento de veículos dependente do local, e no problema aberto de roteamento de veículos. Os autores transformaram todos estes problemas em um rico problema de roteamento de veículos e usaram a meta-heurística. De 486 instâncias testadas, o método foi capaz de encontrar novas melhores soluções em 183 instâncias.

O sucesso da meta-heurística ALNS se deve a vários fatores presentes no algoritmo como uma camada adaptativa e um processo de remoção e inserção de nós (por exemplo, clientes ou navios a serem visitados) que permite diversificar e intensificar a busca.

O ALNS é uma extensão da meta-heurística LNS (*Large Neighborhood Search*) proposta por Shaw (1997). A meta-heurística LNS permite utilizar múltiplos métodos para destruir e reparar uma solução. A meta-heurística ALNS difere da LNS por considerar uma camada adaptativa que permite pontuar e por assim dar uma maior preferência a um dado par de heurísticas de destruição e reconstrução, mais adaptadas à instância sendo resolvida.

##### **4.1 Estratégia da meta-heurística ALNS**

A cada iteração da meta-heurística ALNS, o algoritmo destrói parte de uma de solução corrente  $s$  e a reconstrói de um modo diferente gerando uma nova solução  $s'$ . Essa nova solução é aceita de acordo com um critério de aceitação definido pelo paradigma de busca aplicado em um nível superior, como os critérios de aceitação do *Simulated Annealing*, no

qual se  $s'$  é melhor que  $s$ , a busca continua a partir de  $s'$ , caso contrário, a busca continua a partir de  $s$  com uma dada probabilidade. Os procedimentos de destruição e reconstrução são selecionados de acordo com um mecanismo probabilístico adaptativo.

A cada iteração, a probabilidade de selecionar um dado procedimento depende do seu comportamento no passado. Os procedimentos são na verdade heurísticas que permitem remover nós servidos (destruição) e inserir os nós removidos de volta (reconstrução).

A meta-heurística ALNS apresenta os seguintes componentes: grande vizinhança, mecanismo de busca adaptativa, ajuste de pesos adaptativo das heurísticas de destruição e de reconstrução, critérios de aceitação de soluções piores durante o processo de busca, e ruído no cálculo da função objetivo para evitar que as heurísticas de destruição e reconstrução se tornem repetitivas.

A meta-heurística ALNS é dependente do número de nós removidos e inseridos durante cada iteração do algoritmo. Se somente uma pequena parte da solução é destruída, então o método pode não ser capaz de explorar eficientemente o espaço de busca, porque o efeito da grande vizinhança foi perdido. Por outro lado, se uma grande parte da solução é destruída, então o método degrada muito a solução, aumentando o tempo computacional e produzindo soluções de baixa qualidade (Pisinger e Ropke, 2007).

#### **4.2 ALNS para o PAB**

A meta-heurística ALNS foi implementada de acordo com Ribeiro e Laporte (2012) e Ribeiro *et al.* (2012) considerando a abordagem de Buhrkal *et al.* (2011) que considera o PAB como um PRVMG-JT. Nessa abordagem, cada berço “deixa sua garagem” e “segue” em direção aos navios para servi-los, retornando à sua garagem após o último navio atendido. Assim, uma solução  $s$  consiste em rotas de atendimento para cada berço.

A cada iteração da meta-heurística ALNS, uma heurística de destruição é selecionada, com base no seu histórico de desempenho, para remover  $q$  navios. Quatro heurísticas de destruição foram implementadas:

- 1) Heurística 1 de remoção de Shaw (1997) – essa heurística consiste em remover navios que são similares. Nesse caso, o grau de similaridade entre dois navios é calculado como sendo a diferença absoluta entre os seus custos na solução atual. Inicialmente, um navio é selecionado e, em seguida, o que está mais relacionado a ele é também removido. Desse ponto em diante, um dos navios já removidos é selecionado aleatoriamente e o navio ainda não removido mais relacionado a ele, é então removido da solução. Esse procedimento continua até que  $q$  navios sejam removidos;
- 2) Heurística 2 de remoção de Shaw (1997) – essa heurística é similar à heurística anterior, porém, a similaridade entre dois navios se dá pela diferença absoluta entre seus horários de chegada;
- 3) Heurística de remoção do pior – essa heurística remove navios que apresentam um alto custo na solução atual. A cada remoção, o custo de cada navio ainda não removido é recalculado e o navio com maior custo é removido. Este processo é repetido até que  $q$  navios sejam removidos;

- 4) Heurística de remoção aleatória – essa heurística remove  $q$  navios aleatoriamente. Ela tende a degradar bem a solução atual, porém isso ajuda a diversificar a busca.

Com relação aos procedimentos de reconstrução, três heurísticas foram implementadas:

- 1) Heurística de inserção gulosa ordenada – dada a lista de navios removidos, essa é ordenada de acordo com o horário de chegada dos navios ao porto. Feito isso, o primeiro navio da lista é inserido na sua melhor posição dentre as rotas disponíveis. Considera-se como melhor posição aquele que menos incrementa o custo da solução. Em seguida, o próximo navio da lista é re-inserido seguindo o mesmo critério até que todos os navios removidos tenham sido re-incorporadas à solução;
- 2) Heurística de inserção gulosa – dada a lista de navios removidos, essa heurística avalia a inserção de cada navio na sua melhor posição e aquele navio gerando o menor acréscimo possível é re-incorporado à solução. Essa heurística exige maior esforço computacional que a heurística gulosa descrita anteriormente;
- 3) Heurística de inserção *Regret-K* – essa heurística busca melhorar o comportamento míope das heurísticas gulosas. Para cada navio, é calculado um parâmetro que é igual a diferença entre o custo de inseri-lo na sua 1ª melhor posição em relação à sua 2ª melhor possível, em relação à sua 3ª melhor posição, ..., em relação à sua  $K^{\text{ésima}}$  melhor posição. Em seguida, aquele navio com o maior parâmetro é selecionado para ser inserido na solução atual, na sua 1ª melhor posição.

A Figura 4 apresenta o algoritmo base da meta-heurística ALNS implementada para o PAB. Os parâmetros utilizados no método são detalhados na Tabela 1, na próxima seção.

---

**ALNS** ( $\varphi, \eta, \psi, T_0, T_c, \alpha, \sigma_1, \sigma_2, \sigma_3, S$ )

---

1.  $S^* \leftarrow S; i \leftarrow 0; T \leftarrow T_0;$
2. ENQUANTO (*critério de parada não for alcançado*) FAÇA
3.      $i \leftarrow i + 1; \quad j \leftarrow j + 1;$
4.     SELECIONAR (uma heurística de destruição  $H^-$ );
5.     SELECIONAR (uma heurística de reconstrução  $H^+$ );
6.     APLICAR ( $H^-(S)$  e  $H^+(S)$ );
7.     SE ( $S' > S$ )
8.          $S \leftarrow S';$
9.         SE ( $v(S') > v(S^*)$ )
10.              $S^* \leftarrow S';$
11.             Atualizar a pontuação de  $H^-$  e  $H^+$  com  $\sigma_1$ ;
12.         SENÃO
13.             Atualizar a pontuação de  $H^-$  e  $H^+$  com  $\sigma_2$ ;
14.         FIM-SE;
15.         SENÃO
16.             TOMAR ( $x \in [0,1]$ )
17.             SE ( $x < e^{-\Delta/T}$ )
18.                  $S \leftarrow S';$
19.             Atualizar a pontuação de  $H^-$  e  $H^+$  com  $\sigma_3$ ;
20.             FIM-SE;
21.         FIM-SE;
22.         SE ( $j = \varphi$ )
23.              $j \leftarrow 0;$
24.             Atualizar os pesos para toda  $H^-$  e  $H^+$ ;
25.         FIM-SE;
26.          $T \leftarrow \alpha^* T;$
27.     FIM-ENQUANTO;
28.     RETORNAR ( $S^*$ );

---

**Figura 4:** Algoritmo da meta-heurística ALNS implementada para o PAB.



Para maiores detalhes sobre a meta-heurística ALNS, veja Ropke e Pisinger (2006) e Ribeiro e Laporte (2012) e Ribeiro *et al.* (2012).

## 5. RESULTADOS COMPUTACIONAIS

A meta-heurística aqui apresentada foi implementada na linguagem C++ em um computador equipado com processador AMD Athlon™ 64 de 2,2 GHz e com 1 GB de memória RAM.

As 30 instâncias com 13 berços e 60 navios propostas por Cordeau *et al.* (2005) foram utilizadas nos testes computacionais. Para calibração dos parâmetros da ALNS, foi utilizada uma metodologia similar à adotada por Ribeiro *et al.* (2012). Foram selecionadas aleatoriamente três instâncias: i01, i15 e i30. Cada uma dessas instâncias foi resolvida 5 vezes pela meta-heurística ALNS, variando sempre os valores de cada parâmetro. Assim, um parâmetro recebeu vários valores enquanto os demais continuaram com valores fixos e, então, foi adotado como valor final para cada parâmetro aquele que obteve o melhor resultado médio encontrado entre as 5 execuções para cada instância. Após definir o valor mais adequado para um parâmetro, outro parâmetro foi escolhido por meio do mesmo processo, até que todos os fossem definidos. Detalhes da calibração dos parâmetros da ALNS são apresentados na Tabela 1.

**Tabela 1:** Valores dos parâmetros da ALNS.

Parâmetro	Significado	Valores Testados	Valor Final
$\varphi$	tamanho dos segmentos	[25;50;100]	50
$\eta$	fator de reação	[0,01;0,05;0,1]	0,05
$\psi$	percentual de remoção	[30;40;50]	40
$T_0$	temperatura inicial	[10000;20000;30000]	10000
$T_c$	temperatura de congelamento	[0,01]	0,01
$\alpha$	taxa de resfriamento	[0,975;0,995]	0,975
$\sigma_1$	pontuação se a melhor solução é encontrada	[20;30;50]	50
$\sigma_2$	pontuação se a solução corrente é superada	[10;15;20]	10
$\sigma_3$	pontuação se uma solução pior for aceita	[5;7;10]	7

A meta-heurística ALNS foi aplicada 10 vezes em cada instância e os resultados encontrados são apresentados na Tabela 2, divididos nos casos discreto e contínuo. A coluna “Melhor” indica o valor da melhor solução encontrada dentre as 10 execuções, a coluna “Média” apresenta o valor médio das soluções obtidas, a coluna “Desvio” apresenta o desvio do valor médio em relação à melhor solução encontrada e, por fim, a coluna “T (s)” apresenta o tempo computacional médio das 10 execuções.

Como pode ser observado na Tabela 2, o tempo médio de execução foi inferior a 10 segundos para o caso discreto e inferior a 90 segundos para o caso contínuo. Para o caso discreto, percebe-se um desvio de 0% para todas as instâncias. Isto indica que a meta-heurística ALNS encontrou a melhor solução em todas as 10 execuções do método para cada instância considerada. Já para o caso contínuo, a meta-heurística ALNS apresentou um desvio de apenas 0,82%.

**Tabela 2:** Resultados da ALNS.

Instância	Caso Discreto				Caso Contínuo			
	Melhor	Média	Desvio (%)	T (s)	Melhor	Média	Desvio (%)	T (s)
i01	1409	1409,00	0,00	4,08	1601	1617,50	1,03	82,48
i02	1261	1261,00	0,00	1,30	1312	1314,60	0,20	76,61
i03	1129	1129,00	0,00	1,48	1214	1219,30	0,44	79,49
i04	1302	1302,00	0,00	1,42	1390	1394,00	0,29	60,98
i05	1207	1207,00	0,00	1,16	1278	1283,60	0,44	73,12
i06	1261	1261,00	0,00	1,25	1417	1422,90	0,42	62,42
i07	1279	1279,00	0,00	2,14	1327	1332,20	0,39	66,32
i08	1299	1299,00	0,00	1,94	1401	1416,90	1,13	79,27
i09	1444	1444,00	0,00	1,64	1623	1643,20	1,24	74,76
i10	1213	1213,00	0,00	1,39	1332	1340,90	0,67	76,67
i11	1368	1368,00	0,00	3,21	1529	1556,00	1,77	78,35
i12	1325	1325,00	0,00	1,97	1512	1520,40	0,56	69,21
i13	1360	1360,00	0,00	1,32	1430	1443,40	0,94	66,83
i14	1233	1233,00	0,00	1,33	1277	1278,20	0,09	65,81
i15	1295	1295,00	0,00	1,37	1345	1360,10	1,12	75,35
i16	1364	1364,00	0,00	2,07	1462	1490,80	1,97	77,85
i17	1283	1283,00	0,00	1,16	1303	1305,30	0,18	71,00
i18	1345	1345,00	0,00	1,38	1493	1498,50	0,37	73,94
i19	1367	1367,00	0,00	3,33	1528	1547,10	1,25	75,56
i20	1328	1328,00	0,00	2,67	1406	1415,60	0,68	69,29
i21	1341	1341,00	0,00	5,28	1456	1458,90	0,20	74,73
i22	1326	1326,00	0,00	3,34	1434	1457,00	1,60	76,45
i23	1266	1266,00	0,00	1,40	1386	1394,90	0,64	78,81
i24	1260	1260,00	0,00	1,63	1343	1352,60	0,71	69,07
i25	1376	1376,00	0,00	4,09	1522	1543,80	1,43	77,75
i26	1318	1318,00	0,00	1,85	1459	1482,80	1,63	67,82
i27	1261	1261,00	0,00	1,26	1349	1358,90	0,73	64,76
i28	1359	1359,00	0,00	5,90	1460	1478,40	1,26	81,93
i29	1280	1280,00	0,00	3,15	1327	1336,30	0,70	71,74
i30	1344	1344,00	0,00	3,73	1512	1519,00	0,46	72,45
Média	1306,77	1306,77	0,00	2,31	1414,27	1426,10	0,82	73,03

Uma comparação dos resultados obtidos pela meta-heurística ALNS com diferentes métodos encontrados na literatura é apresentada na Tabela 3. Nessa tabela, são apresentados o tempo médio para resolução de cada instância, o desvio médio obtido, também para cada instância, e o número de soluções ótimas obtidas para o caso discreto e o número de “melhores” soluções encontradas para o caso contínuo. Neste último caso, as soluções ótimas não são conhecidas e a meta-heurística ALNS proposta encontrou 3 soluções idênticas às encontradas pelo método *Clustering Search* e 10 soluções melhores que as conhecidas até então, resultando assim em 13 melhores soluções conhecidas dentre as 30 instâncias.

**Tabela 3:** Comparação dos resultados com a literatura.

Método	Caso Discreto			Caso Contínuo		
	Tempo (s)	Desvio (%)	Soluções Ótimas	Tempo (s)	Desvio (%)	Melhores Soluções
CPLEX <sup>1</sup>	3600,00	-	0/30	-	-	-
<i>Tabu Search</i> <sup>2</sup>	120,00	-	12/30	120,00	-	0/30
<i>Simulated Annealing</i> <sup>3,7</sup>	60,26	-	27/30	64,70	1,57	0/30
PTA/LP <sup>4</sup>	93,99	-	27/30	-	-	-
GSPP <sup>5</sup>	14,98	0,00	30/30	-	-	-
GRASP+PR <sup>6</sup>	15,48	0,06	30/30	-	-	-
<i>Memetic Algorithm</i> <sup>7</sup>	-	-	-	83,13	2,46	0/30
<i>Clustering Search</i> <sup>8</sup>	12,79	0,04	30/30	73,89	1,17	20/30
ALNS	2,31	0,00	30/30	73,03	0,82	13/30

<sup>1</sup> CPLEX 10.0.1<sup>2</sup> Cordeau *et al.* (2005)<sup>3</sup> Mauri *et al.* (2008b)<sup>4</sup> Mauri *et al.* (2008a)<sup>5</sup> Buhkal *et al.* (2011)<sup>6</sup> Lopes *et al.* (2011)<sup>7</sup> Mauri *et al.* (2011)<sup>8</sup> Oliveira *et al.* (2012a,b)

Na Tabela 3, observa-se que o CPLEX, PTA/LP, GSPP e GRASP+PR não foram aplicados para resolução do caso contínuo do PAB. Já o *Memetic Algorithm* não foi aplicado para o caso discreto. Os resultados obtidos pela meta-heurística ALNS demonstram o potencial do método, uma vez que, no caso discreto do PAB, todas as soluções ótimas foram obtidas nas 10 execuções das 30 instâncias e, no caso contínuo, 13 melhores soluções foram encontradas, sendo 10 totalmente novas, com um baixo desvio e um tempo computacional compatível com os demais métodos encontrados na literatura.

## 6. CONCLUSÕES

O Problema da Alocação de Berços (PAB) é um importante e difícil problema a ser resolvido pela administração portuária. Existem altos custos envolvidos e decisões erradas podem levar a prejuízos significativos.

Este artigo apresentou a meta-heurística *Adaptive Large Neighborhood Search* (ALNS) para resolver dois casos distintos do PAB: o discreto e o contínuo. Considerando o conjunto de instâncias utilizado, a meta-heurística ALNS obteve bons resultados, sendo que, para o caso discreto, todas as soluções ótimas foram encontradas em um tempo computacional inferior aos demais publicados da literatura. Já para o caso contínuo, o método encontrou novas soluções em um tempo computacional compatível com os demais métodos da literatura e com um baixo desvio.

Assim, a meta-heurística ALNS se mostrou como uma boa alternativa para resoluções do PAB, de acordo com os resultados encontrados por Ropke e Pisinger (2006).

### Agradecimentos

Os autores agradecem ao CNPq (processos 300692/2009-9, 300747/2010-1, 477148/2011-5, 307002/2011-0 e 476862/2012-4) e à FAPES (processos 54689473/2011 e 59830042/2012) pelo suporte financeiro.

### REFERÊNCIAS BIBLIOGRÁFICAS

- Bierwirth, C. e F. Meisel (2010) A survey of berth allocation and quay crane scheduling problems in container terminals. *European Journal of Operational Research*, v. 202, n. 3, p. 615–627.
- Buhkal, K.; S. Zuglian; S. Ropke; J. Larsen e R. Lusby (2011) Models for the discrete berth allocation problem: a computational comparison. *Transportation Research Part E: Logistics and Transportation Review*, v.

47, p. 461–473.

- Cordeau, J. F.; G. Laporte; P. Legato e L. Moccia (2005) Models and tabu search heuristics for the berth allocation problem. *Transportation Science*, v. 39, n. 4, p. 526–538.
- Dragovic, B; N. K. Park e Z. Radmilovic (2005) Simulation modelling of ship-berth link with priority service. *Maritime Economics & Logistics*, v. 7, p. 316–335.
- Hansen, P.; C. Oğuz e N. Mladenović (2008) Variable neighborhood search for minimum cost berth allocation. *European Journal of Operational Research*, v. 191, n. 3, p. 636–649.
- Imai, A.; E. Nishimura e S. Papadimitriou (2001) The dynamic berth allocation problem for a container port. *Transportation Research Part B: Methodological*, v. 35, p. 401–417.
- Imai, A.; E. Nishimura e S. Papadimitriou (2003) Berth allocation with service priority. *Transportation Research Part B: Methodological*, v. 37, p. 437–457.
- Imai, A.; X. Sun; E. Nishimura e S. Papadimitriou (2005) Berth allocation in a container port: using a continuous location space approach. *Transportation Research Part B: Methodological*, v. 39, p. 199–221.
- Lopes, A. T.; V. M. L. Schulz e G. R. Mauri (2011) Grasp com path relinking para o problema de alocação de berços. *PODes - Pesquisa Operacional para o Desenvolvimento*, v.3, n.3, p. 218–229.
- Mauri, G. R. (2008). *Novas abordagens para representação e obtenção de limitantes e soluções para alguns problemas de otimização combinatória*. Tese (Doutorado em Computação Aplicada), Instituto Nacional de Pesquisas Espaciais (INPE).
- Mauri, G. R.; L. N. Andrade e L. A. N. Lorena (2011) A memetic algorithm for a continuous case of the berth allocation problem. *Anais da III International Conference on Evolutionary Computation Theory and Applications*, ECTA, Paris.
- Mauri, G. R.; A. C. M. Oliveira e L. A. N. Lorena (2008a) A hybrid column generation approach for the berth allocation problem. *Lecture Notes in Computer Science*, v. 4972, p. 110–122.
- Mauri, G. R.; A. C. M. Oliveira e L. A. N. Lorena (2008b) Heurística baseada no simulated annealing aplicada ao problema de alocação de berços. *GEPROS - Gestão da Produção, Operações e Sistemas*, v.1, n.1, p. 113–127.
- Oliveira, R. M.; G. R. Mauri e L. A. N. Lorena (2012a) Clustering search for the berth allocation problem. *Expert Systems with Applications*, v. 39, n. 5, p. 5499–5505.
- Oliveira, R. M.; G. R. Mauri e L. A. N. Lorena (2012b) Clustering search heuristic for solving a continuous berth allocation problem. *Lecture Notes in Computer Science*, v. 7245, p. 49–62.
- Pisinger, D. e S. Ropke (2007) A general heuristic for the vehicle routing problem. *Computers & Operations Research*, v. 34, p. 2403–2435.
- Ribeiro, G. M. e G. Laporte (2012) An adaptive large neighborhood search heuristic for the cumulative capacitated vehicle routing problem. *Computers & Operations Research*, v. 39, p. 728–735.
- Ribeiro, G. M.; G. Laporte e G. R. Mauri (2012) A comparison of three metaheuristics for the workover rig routing problem. *European Journal of Operational Research*, v. 220, n. 1, p. 28–36.
- Ropke, S. e D. Pisinger (2006) An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, v. 40, n. 4, 455–472.
- Shaw, P. (1997) A new local search algorithm providing high quality solutions to vehicle routing problems. *Technical report*, University of Strathclyde Glasgow.
- UNCTAD (2013) United Nations Conference on Trade and Development website. Disponível em: <<http://www.unctad.org>>. Acesso em: 16 jul. 2013.

---

Glaydston Mattos Ribeiro (glaydston@pet.coppe.ufrj.br)

Programa de Engenharia de Transportes, Universidade Federal do Rio de Janeiro - COPPE/UFRJ

Geraldo Regis Mauri (mauri@cca.ufes.br)

Departamento de Computação, Universidade Federal do Espírito Santo - UFES

Luiz Antonio Nogueira Lorena (lorena@lac.inpe.br)

Laboratório Associado de Computação e Matemática Aplicada, Instituto Nacional de Pesquisas Espaciais - INPE

Gilbert Laporte (gilbert.laporte@cirrelt.ca)

CIRRELT / HEC Montréal