

# Resolução do Problema de Carregamento e Descarregamento 3D de Contêineres em Terminais Portuários para Múltiplos Cenários via Representação por Regras e Algoritmo Genético

Aníbal Tavares de Azevedo (UNICAMP) [anibal.azevedo@fca.unicamp.br](mailto:anibal.azevedo@fca.unicamp.br)

Luiz Leduino Salles Neto (UNIFESP) [luiz.leduino@unifesp.br](mailto:luiz.leduino@unifesp.br)

Antônio Carlos Moretti (UNICAMP) [moretti@ime.unicamp.br](mailto:moretti@ime.unicamp.br)

Cassilda Maria Ribeiro (UNESP) [cassilda@feg.unesp.br](mailto:cassilda@feg.unesp.br)

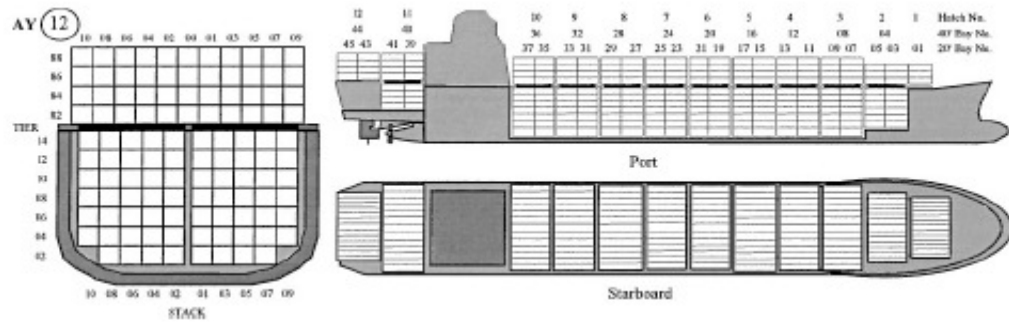
Galeno José de Sena (UNESP) [gsena@feg.unesp.br](mailto:gsena@feg.unesp.br)

*Resumo:* Neste artigo é apresentada uma nova formulação e uma nova representação para a solução do problema de carregamento de contêineres 3D em terminais portuários (PCCTP 3D) para múltiplos cenários que consiste em determinar como carregar e descarregar um conjunto de contêineres de um navio porta-contêiner (containership), respeitando restrições operacionais relacionadas aos contêineres e a estrutura do navio e visando a minimização do número de movimentos e a distância do centro de massa e do centro de gravidade. Assim como o problema 2D é NP-Completo (AVRIEL et al. 2000), o problema 3D também o é, bem como o problema 3D para múltiplos cenários, e é aconselhável a utilização de métodos heurísticos. É proposto um Algoritmo Genético que emprega uma nova representação das soluções, denominada representação por regras, para resolver o PCCTP 3D para múltiplos cenários. A representação por regras tem a grande vantagem de utilizar uma representação bastante compacta que assegura a geração de soluções factíveis e possibilita a inserção de conhecimento do tomador de decisões. Além disso, soluções próximas do limitante inferior do número de movimentos podem ser obtidas em tempo computacional razoável.

*Palavras-chave:* Algoritmo Genético; Carregamento de Contêiner; Otimização Combinatória.

## 1. Introdução

A eficiência de um terminal portuário especializado em movimentação de contêineres depende da ordenação e agilidade do processo de lidar com os contêineres, especialmente durante o processo de carregamento dos navios. A estiva e o plano de carregamento associado são determinados fundamentalmente por dois critérios: estabilidade do navio e o número mínimo de remanejamento requerido nos diversos pontos de entrega (AVRIEL et al., 2000; WILSON e ROACH, 2000; AMBROSINO et al., 2006). O último critério é baseado no fato de que muitos navios possuem uma estrutura celular, conforme pode ser observado na Figura 1, e os contêineres devem ser carregados de modo a formarem pilhas verticais, o que acarreta, em muitos casos, a necessidade de movimentar alguns contêineres para descarregar outros posicionados na parte inferior da pilha.



**Figura 1:** Estrutura celular de um navio. Fonte: WILSON e ROACH (2000).

Concomitantemente, outra restrição emerge durante a escolha dos contêineres para carregamento no pátio do terminal, onde geralmente os contêineres são empilhados formando blocos a espera do momento de serem carregados. Se os contêineres alvos, que devem ser carregados mais tarde, são posicionados nas pilhas abaixo de outros, então a tarefa de carregamento requer remanejamento de modo a remover e reposicionar os contêineres alvos. Esta situação ocorre com frequência, uma vez que a ordem de carregamento não é conhecida quando as cargas chegam ao pátio do terminal e é denominada de realocação. No entanto, mesmo quando esta informação é disponibilizada a tempo, o arranjo ideal de contêineres na área de armazenamento é praticamente impossível de ser obtido devido à chegada aleatória de diversas outras cargas. Mais especificamente, o problema de carregamento de contêineres em terminais portuários (PCCTP) consiste em determinar como carregar um conjunto de contêineres de diferentes tipos em um navio porta-contêiner (*containership*), respeitando restrições operacionais relacionadas aos contêineres, navio e pátio do terminal portuário.

Neste artigo será apresentado um Algoritmo Genético para a solução do PCCTP 3D para múltiplos cenários, ou seja, para a minimização do tempo de carregamento e posterior descarregamento dos contêineres em terminais portuários, bem como a minimização da instabilidade medida como a distância entre o centro de massa e o centro geométrico contemplando vários cenários. Avriel et al. (AVRIEL et al., 2000) apresentaram uma formulação 2D do problema e mostraram que o problema 2D é NP-Completo. Na seção 2 é apresentado o problema 3D para múltiplos cenários a ser resolvido bem como sua formulação matemática binária. Na seção 3 é apresentada a representação por regras combinada com a implementação do algoritmo genético para a resolução de problemas com múltiplos cenários. Na seção 4 são apresentados os resultados computacionais obtidos e na seção 5 as conclusões e sugestões de trabalhos futuros.

## 2. Apresentação do Problema

Um navio porta contêiner tem sua capacidade medida em TEU (*Twenty-foot Equivalent Units*) ou Unidade Equivalente de Vinte pés. Por exemplo um navio com capacidade de 8000 TEUs pode carregar 8000 contêineres de vinte pés. Nos navios tem uma estrutura celular (vide Figura1) onde são alojados os contêineres. Essas células são agrupadas por seções ou baias (em inglês *bays*) e os contêineres são empilhados nessas seções formando pilhas verticais. Então uma baía é um agrupamento de células, com capacidade de se empilhar um certo número de contêineres. A baía tem então linhas horizontais numeradas  $r = 1, 2, \dots, R$ , (a linha 1 é a linha que está em baixo, e a linha  $R$  é a linha do topo da pilha) e colunas numeradas  $c = 1, 2, \dots, C$  (coluna 1 é a primeira coluna da esquerda).

O problema PCCTP 3D para múltiplos cenários que será resolvido aqui consiste em reduzir ao máximo possível dois objetivos em cada cenário: o número de realocações dos contêineres para um certo número de portos  $N$  e a instabilidade do arranjo dos contêineres. Para o primeiro objetivo, minimiza-se a realocação decorrente do descarregamento temporário de contêineres, da pilha de contêineres, com a finalidade de descarregar, num terminal portuário  $p$ , um contêiner que está na parte inferior da pilha. Isto é necessário porque os contêineres que estão numa pilha só podem ser acessados pelo topo. Então um contêiner que está no meio da pilha só pode ser descarregado num determinado porto  $p$  se os contêineres que estão acima dele forem removidos. Para o segundo objetivo define-se a distância do centro de massa ao centro geométrico como medida da instabilidade do navio em cad(AVRIEL et al.,2000) a porto. A seguir será apresentada a formulação deste problema como sendo um problema de programação linear inteira com variáveis binárias 0-1. Esta formulação respeita as restrições operacionais relacionadas aos contêineres e ao navio e aparece de modo mais detalhado em (AVRIEL at al , 1998).

## 2.1 Modelo Matemático

Considere um navio de transporte de contêineres que possui  $D$  baias numeradas  $d = 1, \dots, D$ . Cada baia tem  $R$  linhas horizontais numeradas  $r = 1, 2, \dots, R$ , (a linha 1 é a linha que está em baixo, e a linha  $R$  é a linha do topo da pilha) e  $C$  colunas verticais numeradas  $c = 1, 2, \dots, C$  (coluna 1 é a primeira coluna da esquerda). Apesar da baia ter um formato tridimensional com baias de diferentes tamanhos, a mesma pode ser representada, sem perda de generalidade, por um formato tridimensional com baias de mesma capacidade, em particular um vetor de matrizes. Então, uma baia pode alocar no máximo  $R \times C \times D$  contêineres. É assumido também que todos os contêineres têm o mesmo tamanho e peso. O navio chega no porto 1 completamente vazio e sequencialmente ele visita os portos 2, 3, ...,  $N$ . Em cada porto  $i = 1, \dots, N-1$ , o navio recebe o carregamento de contêineres com destino aos portos  $i+1, \dots, N$ . No último porto ele descarrega os contêineres e fica totalmente vazio. Seja  $T^s = [T_{ij}^s]$  a matriz de transporte de dimensão  $(N-1) \times (N-1)$ , onde  $T_{ij}^s$  é o número de contêineres com origem em  $i$  e destino em  $j$  no cenário  $s$ . A matriz é triangular superior porque  $T_{ij} = 0$  para todo  $i \geq j$ .

A formulação de programação linear inteira do PCCTP 3D para múltiplos cenários é dada pelas Eqs. (1)-(8).

$$\text{Min } f(x) = \sum_{s=1}^S (\alpha\phi_1(x) + \beta\phi_2(y))\theta_s \quad \begin{array}{l} i = 1, \dots, N-1, j = i+1, \dots, N; \\ s = 1, \dots, S; \end{array} \quad (1)$$

$$\text{s.a: } \sum_{v=i+1}^j \sum_{r=1}^R \sum_{c=1}^C \sum_{d=1}^D x_{ijv}^s(r, c, d) - \sum_{k=1}^{i-1} \sum_{r=1}^R \sum_{c=1}^C \sum_{d=1}^D x_{kji}^s(r, c, d) = T_{ij}^s \quad \begin{array}{l} i = 1, \dots, N-1, j = i+1, \dots, N; \\ s = 1, \dots, S; \end{array} \quad (2)$$

$$\sum_{k=1}^j \sum_{j=i+1}^N \sum_{v=i+1}^j x_{kiv}^s(r, c, d) = y_i^s(r, c, d) \quad \begin{array}{l} i = 1, \dots, N-1, r = 1, \dots, R; \\ c = 1, \dots, C; d = 1, \dots, D; \\ s = 1, \dots, S; \end{array} \quad (3)$$

$$y_i^s(r, c, d) - y_{i+1}^s(r, c, d) \geq 0 \quad \begin{array}{l} i = 1, \dots, N-1, r = 1, \dots, R-1; \\ c = 1, \dots, C; d = 1, \dots, D; \\ s = 1, \dots, S; \end{array} \quad (4)$$

$$\sum_{i=1}^{j-1} \sum_{p=j}^N x_{ipj}^s(r, c, d) + \sum_{i=1}^{j-1} \sum_{p=j+1}^N \sum_{v=j+1}^p x_{ipv}^s(r, c, d) \leq 1 \quad \begin{array}{l} i = 2, \dots, N, r = 1, \dots, R-1; \\ c = 1, \dots, C; d = 1, \dots, D; \\ s = 1, \dots, S; \end{array} \quad (5)$$

$$x_{ijv}^s(r, c, d) = 0 \text{ ou } 1; \quad y_i^s(r, c, d) = 0 \text{ ou } 1 \quad (6)$$

onde: a variável binária  $x_{ijv}^s(r, c, d)$  é definida de forma que assume o valor 1 se no cenário  $s$  existir um contêiner no compartimento  $(r, c, d)$  que foi ocupado no porto  $i$  e tem como destino final o porto  $j$  e movido no porto  $v$ ; caso contrário assume valor zero. Por compartimento  $(r, c, d)$  entende-se a linha  $r$ , a coluna  $c$  e a baía  $d$  no compartimento de carga do navio. É necessário salientar que a numeração das linhas é feita de baixo para cima, assim a linha de número 5 está acima da linha de número 4, a numeração das colunas é feita da esquerda para a direita e a profundidade é iniciada da popa até a proa. Similarmente, a variável  $y_i^s(r, c, d)$  possui valor 1 se no cenário  $s$  saindo do porto  $i$  o compartimento  $(r, c, d)$  for ocupado por um contêiner; caso contrário assume valor 0.

A restrição (2) é a restrição de conservação de fluxo, onde  $T_{ij}^s$  é o elemento da matriz de transporte que representa o número de contêineres que embarcam no porto  $i$  com destino ao porto  $j$  no cenário  $s$ . A restrição (3) garante que cada compartimento  $(r, c, d)$  tem no máximo um único contêiner. A restrição (4) é necessária para garantir a existência de contêineres embaixo do contêiner que ocupa o compartimento  $(r, c, d)$ . A restrição (5) é responsável por definir a movimentação dos contêineres: se um contêiner que ocupa a posição  $(r, c, d)$  é descarregado no porto  $j$ , então, ou não existem contêineres acima dele, ou o índice  $v$  do contêiner que ocupa o compartimento  $(r+1, c, d)$  não é maior que  $j$ .

A função objetivo da Eq. (1) possui é uma composição de dois diferentes critérios em um cenário  $s$  ponderado pela probabilidade  $\theta_s$  de ocorrência do cenário: o primeiro critério depende da movimentação dos contêineres,  $\phi_1^s(x)$ , e o segundo depende posição ocupada pelos contêineres em cada porto,  $\phi_2^s(y)$ . Os dois critérios podem ser combinados através de valores escalares fornecidos para os pesos  $\alpha$  e  $\beta$  dentro de um modelo de otimização bi-objetivo.

O termo  $\phi_1^s(x)$  relativo ao custo total de movimentação dos contêineres (assumindo que a movimentação de um contêiner possui um custo unitário e igual para todos os portos) em todos os portos para um cenário  $s$  e é dado pela Eq. (7).

$$\phi_1^s(x) = \sum_{i=1}^{N-1} \sum_{j=i+1v=i+1r=1}^N \sum_{c=1}^{j-1} \sum_{d=1}^R \sum_{d=1}^C x_{ijv}^s(r, c, d) \quad (7)$$

O termo  $\phi_2^s(y)$  é relativo à estabilidade do navio de acordo com a posição dos contêineres (assumindo que cada contêiner possui um peso unitário e igual para todos os portos) em todos os portos no cenário  $s$  e é dado pela Eq. (8).

$$\phi_2^s(y) = \sum_{i=1}^N \sum_{d=1}^D (\Delta xcm_{di}^s)^2 + \sum_{d=1}^D (\Delta zcm_{di}^s)^2 \quad (8)$$

$$\text{onde: } \Delta zcm_{di}^s = zcm_{di}^s - R/2, \quad \Delta xcm_{di}^s = xcm_{di}^s - C/2 \quad e$$

$$zcm_{di}^s = \left( \sum_{r=1}^R \sum_{c=1}^C (y_i^s(r, c, d) \times (r - 0.5)) \right) / \left( \sum_{r=1}^R \sum_{c=1}^C y_i^s(r, c, d) \right)$$

$$xcm_{di}^s = \left( \sum_{c=1}^C \sum_{r=1}^R (y_i^s(r, c, d) \times (c - 0.5)) \right) / \left( \sum_{r=1}^R \sum_{c=1}^C y_i^s(r, c, d) \right)$$

Observe na Eq. (8), que em cada cenário  $s$  para se calcular a soma dos desvios  $\Delta xcm_{di}^s$  e  $\Delta zcm_{di}^s$  entre as coordenadas do centro de massa ( $xcm_d^s, zcm_d^s$ ) e as coordenadas do centro geométrico ( $R/2, C/2$ ) de cada baía  $d$ , é necessário calcular as coordenadas do centro de massa  $xcm_{di}^s$  e  $zcm_{di}^s$ , relativas respectivamente ao eixo das colunas  $c$  e linhas  $r$  de cada baía  $d$ .

Infelizmente, o tamanho que o problema assume com a formulação dada pelas Eqs. (1)-(8) é proibitivo para problemas reais e só pode ser resolvido para problemas pequenos. Além disso, em (AVRIEL E PENN, 1993) é demonstrado que o PCCTP 2D é um problema NP-Completo, e portanto também o PCCTP 3D com múltiplos cenários, o que justifica o uso de heurísticas para encontrar boas soluções, em particular, Algoritmos Genéticos.

Outro problema é relacionado com a questão da representação da solução por meio de variáveis binárias. A formulação (1)-(8) é tal que para se representar uma solução de uma instância com  $S = 10, D = 5, R=6, C = 50$  e  $P = 30$  serão necessárias  $S \times D \times R \times C \times P^3$  variáveis  $x_{ijv}(r,c)$ , ou seja, 405.000.000 variáveis  $x$ , e  $S \times D \times R \times C \times P$  variáveis  $y_i(r,c)$ , ou seja, 450.000 variáveis  $y$ . Ou seja, um total de 405.450.000 variáveis para representar uma única solução.

### 3. Representação por Regras

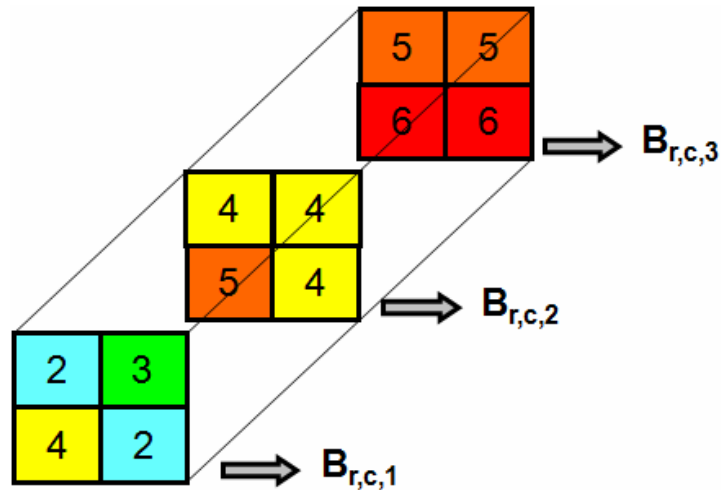
#### 3.1 Representação por Regras para um Cenário

Aqui será apresentada a representação matricial desenvolvida para a resolução PCCTP 3D. Esta representação tem a vantagem de ser compacta e de assegurar que todas as soluções geradas pelo método sejam factíveis.

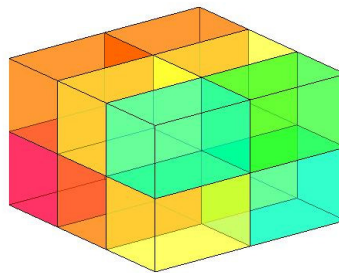
Na Figura 1 viu-se que os navios possuem uma estrutura celular de modo que os locais onde os contêineres serão alocados são pré-determinados fazendo com que os contêineres sejam empilhados verticalmente. Este empilhamento sugere uma representação por meio de um vetor de matrizes dos contêineres no navio. Deste modo pode-se definir um vetor de matrizes de ocupação  $B$  que fornece a quantidade de espaços disponíveis e a localização dos contêineres no navio em cada porto. Para tanto, cada elemento do vetor de matrizes é uma matriz  $B_{rcd}$  que representa o estado de uma célula  $(r,c,d)$ , isto é se  $B_{rcd}=0$  significa que a célula que ocupa a linha  $r$ , a coluna  $c$  e a baía  $d$  está vazia e se  $B_{rcd}=j$  significa que a célula contém um contêiner cujo destino é o porto  $j$ . Assim, no Exemplo da Figura 2, o elemento  $(2,2,1)$  pertencente a baía 1 ( $B_{rc1}$ ), linha 2 e coluna 2 é igual a 3 significando que neste local existe um contêiner que será descarregado no porto 3. De modo análogo, o elemento  $(1,1,3)=6$  significa que esta célula da terceira baía ( $B_{rc3}$ ), primeira linha e primeira coluna contém um contêiner cujo destino é o porto 6. Observando que a linha 2 representa o topo da pilha de carregamento e a linha 1 representa a parte inferior da pilha.

Cada vetor de matrizes  $B_{rcd}$  representa uma ocupação do navio de acordo com um desenho tridimensional. Assim, o vetor de matrizes  $B_{rcd}$  mostrado na Figura 2 será associado ao desenho tridimensional da Figura 3.

É importante observar que a matriz de carregamento  $B$  é modificada em cada porto devido a entrada e saída de novos contêineres em cada porto, pois quando o navio chega num porto  $j$  é necessário realizar dois movimentos obrigatórios, a saber: descarregar os contêineres cujo destino é o porto  $j$  em questão e carregar os contêineres com destinos aos portos  $j+1, j+2, \dots, N$ . Então, para cada porto  $j$  foram estabelecidas regras para se fazer o carregamento e descarregamento dos contêineres.



**Figura 2:** Matriz de Ocupação para navio com capacidade de 16 contêineres e transporte para 6 portos.



**Figura 3:** Representação tridimensional da Matriz de Ocupação mostrada na Figura 2.

Muitas vezes, para se fazer o descarregamento no porto  $j$ , de um contêiner cujo destino é o porto  $j$ , é necessário fazer operações de remanejamento dos contêineres cujo destino são os portos de  $j+1$  até  $N$ , porque a posição que eles ocupam na pilha está acima da posição do contêiner do porto  $j$ . Veja por exemplo que na matriz  $B$  da Figura 3, para se descarregar os contêineres do porto 2, serão necessário descarregar os contêineres contidos nas células  $(1,1,1)$ ,  $(1,2,1)$  e  $(1,2,2)$ . Assim, com intuito de reduzir o número de remanejamentos, ao se fazer o carregamento de contêineres num dado porto  $j$  deve-se levar em conta os contêineres que já estão no navio, porque foram embarcados nos portos anteriores (portos de 1 até  $j-1$ ) com destino aos portos  $j+1$  até  $N$ . Observe então que existe uma relação íntima entre as operações de carregamento e descarregamento, tendo em vista que, a forma como é realizado o carregamento num porto  $j$  vai influenciar no descarregamento a ser efetuado nos demais portos (portos de  $j+1$  até  $N$ ). Portanto, para reduzir as operações de remanejamento é necessário estabelecer regras para o carregamento e descarregamento de contêineres em cada porto que leve em conta esta relação. Para tanto foram criadas doze regras, sendo seis para o carregamento (Rc1, Rc2, Rc3, Rc4, Rc5, Rc6) e duas para o descarregamento (Rd1, Rd2). A combinação de uma regra de carregamento com uma de descarregamento fornece a regra  $k$  para o porto  $j$ .

**Tabela 1:** Regras  $k$  a serem utilizadas em cada porto  $j$ .

Regra de carregamento	Regra de descarregamento	Regra $k$ usada no porto $j$
Rc1	Rd1	<b>1</b>
	Rd2	<b>2</b>
Rc2	Rd1	<b>3</b>
	Rd2	<b>4</b>
Rc3	Rd1	<b>5</b>
	Rd2	<b>6</b>
Rc4	Rd1	<b>7</b>
	Rd2	<b>8</b>

Rc5	Rd1	<b>9</b>
	Rd2	<b>10</b>
Rc6	Rd1	<b>11</b>
	Rd2	<b>12</b>

Observe na Tabela 1 que a regra 2 foi obtida utilizando a regra Rc1 para o carregamento dos contêineres e a regra Rd2 para o descarregamento. Isto foi feito com objetivo de se obter uma representação compacta da solução.

A aplicação destas regras em cada porto  $j$  vai atualizar a Matriz de Ocupação  $B_{rcd}$  no porto  $j$ . Vale lembrar que inicialmente a matriz  $B$  está com todos seus elementos iguais a zero e ela começa a ser preenchida no porto 1. Para melhor ilustrar a utilização das regras, será utilizada a matriz de transporte  $T$ , que fornece a quantidade de contêineres que devem ser embarcados em cada porto  $i$  com destino a cada porto  $j$ .

Para maiores detalhes acerca da descrição e funcionamento das regras em (AZEVEDO et al., 2012).

As vantagens do emprego de regras são:

- A facilidade de incorporar conhecimento prévio do decisor sob a forma de regras.
- As regras só podem produzir matrizes de Ocupação factíveis, facilitando e garantindo, a obtenção de soluções factíveis por métodos heurísticos (neste trabalho um Algoritmo Genético).
- A codificação da solução que determina como será realizado o carregamento e o descarregamento de um navio para  $N$  portos é um vetor de tamanho  $N-1$ . Esta representação é muito mais compacta se comparada com outras abordagens da literatura como, por exemplo, a utilizada em (AVRIEL et al., 2000).

Por fim, de posse das regras é possível verificar que a representação por regras fornece uma representação mais compacta da solução do problema. Por exemplo, para se representar uma solução por meio de variáveis binárias para uma instância com  $S = 10$ ,  $D = 5$ ,  $R=6$ ,  $C = 50$  e  $P = 30$  serão necessárias  $S \times D \times R \times C \times P^3$  variáveis  $x_{ijv}(r,c)$ , ou seja,  $405.000.000$  variáveis  $x$ , e  $S \times D \times R \times C \times P$  variáveis  $y_i(r,c)$ , ou seja,  $450.000$  variáveis  $y$ . Ou seja, um total de  $405.450.000$  variáveis para representar uma única solução. Já empregando a representação por regras o número de variáveis para se representar uma única solução corresponde ao número de portos menos um, ou seja, neste caso, 29. Tal economia em termos de representação só é possível, pois a avaliação de uma solução *vsol* combina avaliação das regras com a simulação do estado do navio ao longo dos portos após a aplicação de cada regra tal como descrito em (AZEVEDO et al., 2010). Para o PCCTP 3D o procedimento dado em (AZEVEDO et al., 2010) foi modificado para retornar não só o número de movimentos, mas também a medida de estabilidade (AZEVEDO et al., 2012) tal como dado pelas Equações de (1) até (8) e o presente artigo permite a resolução do PCCTP 3D para múltiplos cenários tal como detalhado na seção 3.2.

### 3.2 Representação por Regras para Múltiplos Cenários

É importante observar que a consideração de múltiplos cenários na representação por regras pode ser obtida empregando-se o Algoritmo da Avaliação de uma Solução da Figura 4. Para tanto, é necessário observar que um vetor solução  $s$  contendo  $N$  regras de carregamento e descarregamento para  $N$  portos deverá ser avaliado para  $S$  cenários. Depois, o número de movimentos deverá ser multiplicado pela probabilidade de ocorrência de cada cenário  $s$ . Este procedimento, portanto, engloba a avaliação mono cenário tal como descrito na Figura 4.

#### Avaliação de uma Solução para Múltiplos Cenários

```
[nmov,instab] = AvaliarMC(v,T)
Início
s ← 1; nmov ← 0;
Enquanto (s < S) faça
Início
  [aux1,aux2] = Avaliar3D(v,T(s));
  nmov ← nmov + aux1*p(s);
  instab ← instab + aux2*p(s);
  s ← s + 1;
Fim
retornar nmov e instab;
Fim
```

Figura 4: Algoritmo para avaliação de uma solução por meio de regras.

Os símbolos e funções empregados no algoritmo da Figura 4 são descritos a seguir:

- s - variável contadora que indica o cenário atual de avaliação.
- S - número total de cenários.
- T - vetor de matrizes de transporte tal que o elemento  $T(s)$  contém a matriz de transporte para o cenário  $s$ .
- p - vetor tal que o elemento  $p(s)$  contém a probabilidade de ocorrência do cenário  $s$  associado a matriz de transporte  $T(s)$ .
- nmov - número de movimentos realizados para carregar ou descarregar o navio ao longo dos  $N$  portos para o cenário  $s$  tal como dado pela Eq. (7).
- instab - medida de instabilidade para o cenário  $s$  com matriz de transporte  $T(s)$  calculada de acordo com a Eq. (8).

A vantagem da abordagem destacada na Figura 4 é que toda a codificação relativa a solução via representação por regras para um único cenário pode ser reaproveitada para a consideração e avaliação de múltiplos cenários.

Portanto, a mesma lógica de avaliação via simulação descrita e utilizada na solução do PCCTP 2D, descrita em (AZEVEDO *et al.*, 2010), é modificada para avaliar uma solução do PCCTP 3D, bastando realizar uma única modificação no processo de simulação que agora passa a considerar uma representação mais realista do navio (3D). Assim a função **Avaliar3D** da Figura 4 fornece dois valores combinados em uma única função: número de movimentos e medida de instabilidade (vide Equações (1), (7) e (8)). Para se considerar múltiplos cenários e **3D** basta fazer a função **Avaliar3D** ser englobada na avaliação de múltiplos cenários, isto é, para o vetor solução  $v$  contendo regras são extraídos o número de movimentos e a medida de instabilidade para  $s$  cenários ( $s$  matrizes de transporte  $T$ ).

## 4. Resultados obtidos

Para testar o algoritmo, foram geradas automática e aleatoriamente 15 instâncias tal que cada instância contempla 10 cenários diferentes e equiprováveis. Essas instâncias são classificadas de acordo com o número de portos e o tipo da matriz de transporte. Para cada instância é gerada dez matrizes de transporte  $T$ , tal que para cada matriz a capacidade do



navio não será excedida em nenhum porto, isto é, o valor de  $\theta_p$  dado pela equação (9) deve ser menor ou igual a  $D \times R \times C$  para todo porto  $p$ , isto porque a matriz de transporte é factível se:

$$\sum_{i=1}^p \sum_{j=p+1}^N T_{ij} \leq D \times R \times C \text{ para todo } p= 1, \dots, N \quad (9)$$

De acordo com AVRIEL *et al.* (1998) podem ser gerados três tipos de matriz de transporte: 1-Mista, 2-Longa distância e 3-Curta distância. Uma matriz do tipo 3 se refere ao transporte de contêineres que vão percorrer poucos portos antes de serem desembarcados. Já uma matriz do tipo 2 se refere a contêineres que vão percorrer muitos portos antes de serem desembarcados. Uma matriz do tipo mista combina os dois tipos anteriores. As instâncias foram classificadas de acordo com a quantidade de portos a serem percorridos, o tipo da matriz de transporte e a capacidade do navio. Neste trabalho foi suposto um navio com as seguintes dimensões ( $D \times R \times C$ ):  $5 \times 6 \times 50$ , resultando na seguinte capacidade máxima: 1500 contêineres. As instâncias utilizadas nos testes encontram-se disponíveis em: <https://sites.google.com/site/projetonavio/home>.

A Tabela 2 mostra os resultados obtidos com a melhor solução após 5 rodadas de algoritmo genético com população de 10 indivíduos, 1000 gerações, taxa de crossover igual a 80% e taxa de mutação igual a 15% para as 15 instâncias. Como cada instância compreende 10 cenários (ou seja, 10 matrizes de transporte diferentes), então, cada linha da tabela corresponde ao desempenho médio para os 10 diferentes cenários da melhor solução obtida. A tabela inclui as colunas relacionadas a seguir: **I**: o número da instância; **M**: o tipo da matriz de transporte; **N**: o número de portos; **NMin**: o número mínimo de movimentos a serem realizados com os contêineres; **F.O.1**: os valores da função objetivo em termos do número total de movimentos realizados com os contêineres até a chegada no último porto; **F.O.2**: os valores de estabilidade de acordo com o arranjo de contêineres e com a equação (8); e **T**: tempo computacional em segundos. Note-se que os valores de **F.O.1**, **F.O.2** e **T** são apresentados para dois pares de  $\alpha$  e  $\beta$ : ( $\alpha=1, \beta=0$ ) e ( $\alpha=0, \beta=1$ ). Note-se ainda que os resultados apresentados para ( $\alpha=1, \beta=0$ ) correspondem à melhor solução obtida com o objetivo de minimizar o número de movimentos para cada instância e que para 7 instâncias o número de movimentos (**FO1**) encontrado é até 4% maior que o número mínimo (Nmin), para 6 instâncias o número mínimo é até 25% maior e apenas 2 instâncias até 49%. Já para ( $\alpha=0, \beta=1$ ) como o objetivo é minimizar a medida de instabilidade apresenta maior número de movimentos (**FO1**), mas soluções com medida de instabilidade significativa menor (**FO2**). Estes dados confirmam que a minimização do número de movimentos e da instabilidade são objetivos conflitantes. O número mínimo de movimentos é obtido multiplicando-se por dois o valor do somatório de  $T_{ij}$  calculado de acordo com a equação (9). Observe-se que esse valor é o limitante inferior para o número total de movimentos a serem realizados ao longo do percurso do navio. Os resultados do GA foram obtidos com um programa desenvolvido em Matlab 7.0 e executado num computador Intel Core 2Duo 2.20 GHz (E4500), 2GB RAM, Windows XP Versão 2002 (SP 3).

**Tabela 2:** Resultados do Algoritmo Genético para dois pares de valores de  $\alpha$  e  $\beta$ .

I	N	M	NMin	$(\alpha=1, \beta=0)$			$(\alpha=0, \beta=1)$		
				FO1	FO2	T	FO1	FO2	T
1	10	1	6499,00	6698,60	506,56	1344,91	13067,20	66,26	1336,84
2	10	2	3908,20	4019,00	453,23	1238,53	7329,40	62,22	1208,63
3	10	3	17202,40	17255,00	690,54	1471,28	18699,40	26,71	1427,30
4	15	1	8752,40	9328,00	393,19	2200,74	15196,20	12,86	2139,79

5	15	2	4531,80	5234,80	452,95	8897,97	10729,60	80,24	1910,00
6	15	3	26573,80	27010,00	569,45	6517,74	29307,60	46,35	2288,97
7	20	1	11293,40	12856,60	322,50	3415,65	24943,00	29,43	2979,87
8	20	2	4937,00	5792,40	458,31	2698,28	11759,60	83,69	2549,33
9	20	3	33734,80	34377,00	811,93	3176,09	37606,80	151,36	3031,79
10	25	1	12946,40	15403,80	262,33	3793,48	29236,00	15,10	3664,56
11	25	2	5378,60	7032,00	378,51	3299,93	14309,00	86,58	3187,03
12	25	3	43028,40	44629,40	478,33	3948,62	47908,60	152,22	3890,08
13	30	1	12660,20	15681,60	203,24	4493,20	27650,80	17,25	4282,49
14	30	2	5729,80	8512,40	328,47	4080,54	17185,80	79,73	3767,08
15	30	3	52699,40	54540,40	534,64	5089,76	57995,20	133,14	4725,34

A primeira observação importante acerca dos resultados da Tabela 2 é sobre o tempo computacional gasto pelo algoritmo genético. É importante observar que para a formulação dada pelas equações (1)-(8) as instâncias com 30 portos são problemas com 405.450.000 variáveis inteiras (10 cenários, 30 portos, 5 baías, 6 linhas e 50 colunas). Para estas instâncias o algoritmo genético consegue produzir boas soluções em mais ou menos 1 hora e 20 minutos. Pode-se observar também que, de forma geral, um aumento de 5 no número de portos a serem percorridos, de uma instância para outra, produz em média um aumento de mais ou menos 13 minutos e 30 segundos no tempo computacional gasto pelo algoritmo genético. Por exemplo, em instâncias com 10 portos leva-se 22 minutos para se obter uma solução, ao passo que em instâncias com 15 portos leva-se, em média, 35 minutos. Espera-se ainda, que futuras implementações em linguagem C venham a reduzir o tempo computacional de solução pelo algoritmo genético.

Os resultados indicam que para as instâncias em que a função objetivo visa minimizar o número de movimentos ( $\alpha=1$ ,  $\beta=0$ ) e a matriz de transporte é do tipo curta distância (tipo 3) as regras são bastante adequadas e produzem resultados muito próximos do limite inferior do número de movimentos, e chegam mesmo a quase atingir este limite, como no caso para 10, 15 e 20 portos produzem soluções com distância de 0.30%, 1.61% e 1.87%, respectivamente. Já para as instâncias em que a matriz de transporte é do tipo média distância (tipo 1) e longa distância (tipo 2), o algoritmo genético apresentou soluções cujo número de movimentos é bem maior que o limitante inferior (na instância de 30 portos e matriz tipo 2 cerca de 30%). Estes resultados indicam a necessidade de se incorporar ao sistema um número maior de regras que levem em consideração a arrumação de contêineres que permanecerão um longo período de tempo dentro do navio ou uma melhor estimativa de limitante inferior.

Porém, quando a função objetivo visa minimizar a medida de instabilidade ( $\alpha=0$ ,  $\beta=1$ ) as soluções apresentadas pelo GA têm um razoável aumento no número de movimentos, mas com uma significativa melhora na medida de instabilidade. Em algumas instâncias essa medida pode ser até 26 vezes menor em comparação com a solução encontrada na minimização do número de movimentos (instância 3).

## 5. Conclusões e trabalhos futuros

Este artigo apresentou pela primeira vez uma nova representação de soluções para o problema de carregamento de contêineres 3D em terminais portuários (PCCTP 3D) considerando múltiplos cenários. Esta nova representação permite reduzir a quantidade de informações necessárias para se representar uma solução tal que soluções próximas do limitante inferior do número de movimentos podem ser obtidas em pequeno tempo computacional.

Este artigo tem três inovações, a saber: a primeira delas é que ele apresenta uma nova formulação tridimensional para o problema de carregamento de contêineres em terminais portuários (PCCTP 3D) para múltiplos cenários. Esta nova representação permite a obtenção

de soluções que consideram medidas de instabilidade acerca do arranjo dos contêineres no navio, além da medida acerca do número de movimentos para carregar e descarregar um navio em vários cenários simultaneamente.

A segunda inovação é o uso da representação com regras, cuja aplicação teve êxito no problema 3D, e que também no problema 3D com múltiplos cenários reduz consideravelmente o número de variáveis do problema reduzindo assim a quantidade de informações necessárias para se representar uma solução. Na formulação dada pelas Eqs. (1)-(8) as informações devem ser armazenadas num vetor de tamanho  $(S \times D \times R \times C) \times (N + N^3)$ . Na representação aqui utilizada o tamanho do vetor é  $N-1$ . Tendo em vista que  $S$  é o número de cenários,  $D \times R \times C$  expressa o número de contêineres que podem ser armazenados em um navio e que  $N$  representa o número de portos, a representação da solução aqui utilizada permite uma enorme redução do tempo computacional para instâncias com grande número de portos. Para tanto, a representação emprega regras que definem como será carregado e descarregado o navio e as regras garantem que as soluções obtidas são sempre factíveis qualquer que seja o cenário considerado.

A terceira inovação é que a utilização de múltiplos cenários por meio de múltiplas matrizes de transporte tal que permite que a avaliação de uma solução, formada por um sequência de regras de carregamento e descarregamento, para vários cenários por meio de um processo de simulação. Assim, pode-se verificar o conhecimento do profissional responsável por esse serviço no porto, sob a forma de regras, para diversas situações. Esta maneira de representar o PCCTP 3D para múltiplos cenários em conjunto com a aplicação de regras, é uma alternativa promissora na resolução deste problema, até porque assim como se podem acrescentar novas regras, também se podem utilizar outras heurísticas, como por exemplo, Simulated Annealing, ILS, VNS, dentre outras. Futuramente pretende-se testar um método multiobjetivo que considere o uso de fronteira pareto-ótima para a obtenção de soluções que atendam bem os dois objetivos conflitantes, a elaboração de novas regras que forneçam arranjos dos contêineres com bons valores em termos de número de movimentos e medida de instabilidade, bem como testes com outras medidas de instabilidade do navio.

Agradecimentos à FAPESP pelo apoio financeiro através dos processos 2009/15107-0, 2010/51274-5, 2010-16517-4, 2010/16622-2 e 2011/01667-3.

## Referências

- AMBROSINO, D.; SCIOMACHEN A.; TANFANI, E. A decomposition heuristics for the container ship stowage problem, *J. Heuristics*, v.12, p. 211–233, 2006.
- AVRIEL, M.; PENN, M. Container ship stowage problem, *Computers and Industrial Engineering*, v. 25, p. 271-274, 1993.
- AVRIEL, M.; PENN, M.; SHPIRER, N.; WITTENBOON, S. Stowage planning for container ships to reduce the number of shifts, *Annals of Operations Research*, v. 76, p. 55-71, 1998.
- AVRIEL, M.; PENN, M.; SHPIRER, N. Container ship stowage problem: complexity and connection to the coloring of circle graphs, *Discrete Applied Mathematics*, v. 103, p. 271-279, 2000.

AZEVEDO, A. T. ; RIBEIRO, C. M. ; DEUS, N. M. R. (2010) Resolução do Problema de Carregamento e Descarregamento de Contêineres em Terminais Portuários via Algoritmo Genético. *Revista INGEPRO*, v. 2, p. 38-51, 2010.

AZEVEDO, A. T. ; RIBEIRO, C. M. ; CHAVES, A. A., SENA, G. J., SALLES NETO, L. L., MORETTI, A. C. (2012) Solving the 3D Containership Stowage Loading Planning Problem by Representation by Rules and Beam Search. *First International Conference on Operations Research and Enterprise Systems – ICORES 2012*, p. 132-141, 2012.

WILSON, I.; ROACH, P. Container stowage planning: a methodology for generating computerised solutions, *Journal of the Operational Research Society*, v. 51, p. 1248-1255, 2000.