

Algoritmo Genético para um sistema de *Cross-Docking*

Ubiratan Soares Cavalcante Netto e André Luís Shiguemoto

Universidade Federal do Ceará
ubiratans@gmail.com, shiguemoto@ufc.br

Resumo - *Cross-docking* é um sistema de distribuição no qual produtos que chegam ao armazém (centro de distribuição) são organizados, carregados e enviados para a entrega aos clientes, sem que passem períodos superiores a 24 horas em estoque. Os produtos chegam ao armazém através de caminhões de entrada (*inbound trucks*), são descarregados em docas específicas, verificados e organizados de acordo com seus destinos e a demanda dos clientes, e depois enviados diretamente aos caminhões de saída (*outbound trucks*), que farão as entregas para os clientes. Desta forma, os custos de estoque e o tempo de entrega aos clientes são reduzidos substancialmente. Um parâmetro importante para obter sucesso na implantação do *cross-docking*, é a eficiência relacionada ao escalonamento dos caminhões de saída e entrada, assim como a transferência de produtos entre os mesmos. Nesta pesquisa é apresentado um Algoritmo Genético, para encontrar um bom escalonamento de caminhões, reduzindo o tempo operacional total, denominado *makespan*. A meta-heurística foi testada em 29 instâncias e todas as soluções encontradas são iguais ou melhores que as da literatura.

Palavras-chave – *cross-docking*, *makespan*, logística, escalonamento, algoritmo genético, meta-heurística.

1 Introdução

Em uma realidade caracterizada pela competição e pela maximização dos lucros, as universidades e as empresas reconhecem o grande potencial de otimizar operações de manufatura e distribuição de produtos visando alcançar melhor eficiência e melhores serviços.

Atualmente, estima-se que o custo logístico de uma empresa possa ser equivalente a 19% do seu faturamento (Associação Brasileira de Movimentação Logística - ABML) e, portanto a logística é uma área de estudo promissora para se obter economia significativa nos custos.

Apte e Viswanathan (2000) mostram diversas empresas que alcançaram melhorias significativas em suas operações de manufatura, ao sincronizar o planejamento das operações de produção com o fornecimento de matéria prima. Por esse motivo algumas empresas, deixaram de priorizar as operações de produção e passaram a priorizar as operações de logística e distribuição. Buscando o uso de níveis mínimos de estoque através da cadeia logística e a entrega de baixo volume com maior frequência. Isto porque com a redução dos níveis de estoque, os custos de operação e das atividades de separação e preparação (*picking* na língua inglesa) são reduzidos. Neste contexto, para manter níveis mínimos de estoque surge o sistema de *cross-docking* em que os produtos (mercadorias) que chegam a um armazém (centro de distribuição – CD) são imediatamente classificados e reorganizados a partir das demandas dos clientes, e são carregados e despachados para entrega aos clientes, passando menos de 24 horas no estoque dos armazéns.

Geralmente, as instalações que utilizam *cross-docking* incluem três funções: recebimento, *picking* e distribuição. Estima-se que com a utilização do sistema de *cross-docking* os custos de armazenagem podem ser reduzidos em até 70%. Nesse sistema, a principal idéia é transferir os produtos recebidos, em menos de 24 horas, diretamente aos veículos responsáveis pela distribuição, sem armazená-los em um depósito ou terminal. Para que isso ocorra todas as funções do sistema de *cross-docking* exigem controle e precisão na gestão de produtos, sendo fortemente dependente do fluxo de informações.

O *cross-docking* também é essencial em instalações de transbordo para caminhões que chegam com produtos que devem ser classificados, consolidados com outros produtos, e carregados em caminhões de saída. Em um modelo de *cross-docking*, o destino de um produto é conhecido antes de sua chegada ao armazém, não havendo necessidade de movê-lo para o armazenamento.

O fluxo de produtos em um típico modelo de *cross-docking* (Figura 1) ocorre da seguinte maneira: (i) Os produtos que chegam ao armazém são recebidos nas docas de desembarque (*receiving docks*). (ii) Os produtos são postos em sistemas de organização, os quais são classificados e reorganizados de acordo com seus respectivos destinos. (iii) Os produtos são transferidos para as docas de embarque (*shipping dock*), onde serão carregados e despachados em veículos para a entrega aos clientes.

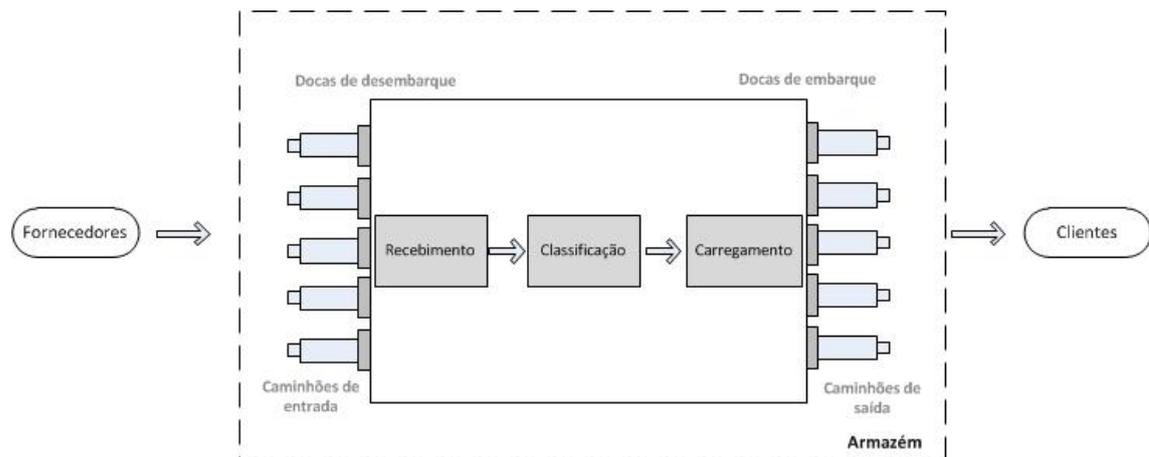


Figura 1 – Fluxo de produtos em um típico modelo de cross-docking.

Em geral, a literatura associada com a programação de entradas e saídas de um sistema de *cross-docking* é escassa. Alguns autores, tais como Forger (1995), Witt (1998), Apte e Viswanathan (2000) e Gue (2002), apresentam sucessos na implantação do sistema de *cross-docking* em empresas, tais como UPS, Toyota e Wal Mart. Lau et al. (2003) apresentam uma heurística de busca tabu que minimiza os custos de distribuição em um sistema de *cross-docking* com janelas de tempo e uma frota finita de veículos. Lee et al. (2006) propõem um modelo que integra um sistema de *cross-docking* com um processo de coleta e entrega de produtos em uma cadeia de suprimentos. Os autores também desenvolveram um modelo matemático para determinar se uma sequência ótima de distribuição considerando um sistema de *cross-docking*. Mais recentemente, Yu e Egbelu (2008), Vahdani e Zandieh (2009) e Arabani, Ghomi e Zandieh (2010) propõem diversas heurísticas que tratam do problema considerando um único modal. Por fim, Boysen e Fließner (2009) apresentam uma revisão da literatura e uma classificação para os problemas que utilizam os sistemas de *cross-docking*.

O foco desta pesquisa são as sequências de transferências de produtos realizadas entre os caminhões de entrada (*inbound trucks*), que descarregam os produtos no setor de desembarque do armazém para o setor de embarque, e os caminhões de saída (*outbound trucks*), que são carregados com os produtos do setor de embarque e então os entregam aos clientes, tendo o objetivo de minimizar o tempo total gasto nessas operações de transferências de produtos. Esse trabalho está organizado da seguinte forma: a seção 2 apresenta uma descrição do problema, e o modelo matemático. A seção 3 apresenta o algoritmo genético (AG). Resultados computacionais são exibidos e comentados na seção 4, e as conclusões na seção 5.

2 Descrição do problema

Define-se *cross-docking* como um sistema de distribuição no qual os produtos (mercadorias) recebidos em um armazém, não são estocados, mas preparados para o carregamento e distribuição a fim de serem entregues ao cliente no menor tempo possível.

Neste trabalho é considerado um sistema de *cross-docking* análogo ao apresentado por Yu e Egbelu (2008). O sistema trata de um problema que envolve um conjunto de veículos que devem ser sequenciados nas docas de embarque e desembarque de um único armazém para distribuir um conjunto de produtos. Neste problema, não se considera as operações que ocorrem dentro do armazém, como classificação, preparação, etc. Portanto, a sequência da chegada de produtos na doca de embarque é a mesma sequência que foi descarregada na doca de desembarque.

No armazém há apenas uma doca de desembarque e uma doca de embarque, e presume-se que há um armazenamento temporário na frente da doca de embarque com capacidade infinita. Se um produto que chega à doca de embarque não se destina ao caminhão de saída, que está na doca, o produto é armazenado temporariamente até que o devido caminhão entre na doca de embarque, a Figura 2 apresenta o fluxo de produtos no modelo em questão. Uma vez que um caminhão de entrada chega à doca de desembarque, a sua partida só é permitida no momento em que ele descarregar todos os seus produtos. Assim como só se permite a partida de um caminhão de saída, da doca de embarque, quando o mesmo carregar todos os produtos requeridos.

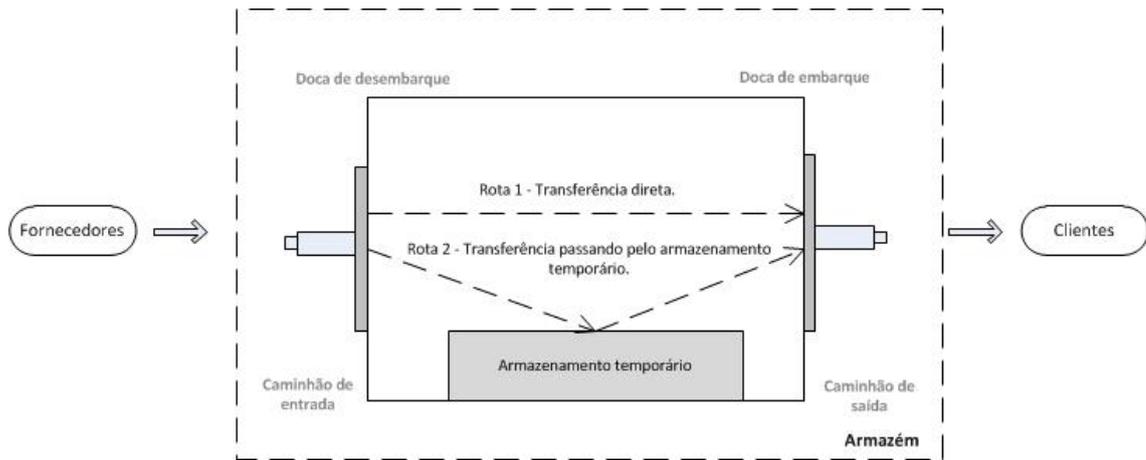


Figura 2 – Fluxo de produtos aplicado ao problema.

Além disso, outras suposições são aplicadas: (i) Um caminhão entra na doca uma única vez. (ii) Todos os caminhões de entrada e de saída estão disponíveis no tempo zero (início do dia). (iii) Todos os produtos recebidos pelo armazém devem ser embarcados; estocagens de grandes durações não são permitidas. (iv) A sequência de descarregamento de um caminhão de entrada pode ser determinada. (v) Permite-se descarregar de um caminhão de entrada, apenas a quantidade de produtos de certo tipo, requerida pelo caminhão de saída que está na doca de embarque, no momento. (vi) Apenas uma unidade de um produto pode ser carregada por vez, em um caminhão de saída. Portanto, um caminhão de saída não pode carregar múltiplos produtos de forma simultaneamente da doca de desembarque e do armazenamento temporário. (vii) O tempo gasto pela troca de caminhões nas docas é o mesmo para todos os caminhões de saída e os de entrada. (viii) A capacidade de estocagem do armazenamento temporário é infinita. (ix) Há apenas uma doca de desembarque e uma doca de embarque. (x) São assumidas que sejam conhecidas as seguintes informações:

- Os tipos de produtos e a quantidade de cada produto carregado em um caminhão de entrada e requerido por um caminhão de saída.
- O tempo de carregamento e descarregamento dos produtos.
- O tempo de movimentação dos produtos da doca de desembarque à doca de embarque.
- O tempo de troca de caminhões.

Nesta pesquisa, o tempo de operação total da operação de *cross-docking*, denomina-se *makespan*. Ou seja, *makespan* é o tempo gasto desde o descarregamento do primeiro produto do primeiro caminhão de entrada, até o carregamento do último produto no último caminhão de saída.

Modelo Matemático

A formulação matemática que representa o problema apresentado baseia-se em modelos de programação inteira mista e considera as seguintes variáveis:

- R quantidade de caminhões de entrada no conjunto.
 S quantidade de caminhões de saída no conjunto.
 N quantidade de tipos de produtos no conjunto.
 r_{ik} quantidade inicial de produtos do tipo k carregadas no caminhão de entrada i .
 s_{jk} quantidade inicial de produtos do tipo k que precisam ser carregadas no caminhão de saída j .
 D tempo de troca de caminhões (*truck changeover time*).
 V tempo de movimentação de produtos da doca de desembarque para a doca de embarque.
 M número grande.

Variáveis contínuas:

- T *makespan*.
 c_i horário em que o caminhão de entrada i , entra na doca de desembarque.
 f_i horário em que o caminhão de entrada i , deixa a doca de desembarque.
 d_j horário em que o caminhão de saída j , entra na doca de embarque.
 l_j horário em que o caminhão de saída j , deixa a doca de embarque.

Variáveis inteiras:

- x_{ijk} quantidade de produtos do tipo k que é transferida do caminhão de entrada i para o caminhão de saída j .

Variáveis binárias:

$$v_{ij} = \begin{cases} 1, & \text{se } x_{ijk} \neq 0, \text{ para algum } k. \\ 0, & \text{caso contrário} \end{cases}$$

$$p_{ij} = \begin{cases} 1, & \text{se o caminhão de entrada } i \text{ precede } j, \text{ na sequência de caminhões de entrada.} \\ 0, & \text{caso contrário.} \end{cases}$$

$$q_{ij} = \begin{cases} 1, & \text{se o caminhão de saída } i \text{ precede } j, \text{ na sequência de caminhões de saída.} \\ 0, & \text{caso contrário.} \end{cases}$$

Modelo matemático:

Min T

Sujeito a:

$$T \geq L_j, \quad \text{para todo } j, \quad (1)$$

$$\sum_{j=1}^S x_{ijk} = r_{ik} \quad \text{para todo } i, k, \quad (2)$$

$$\sum_{i=1}^R x_{ijk} = s_{jk} \quad \text{para todo } j, k, \quad (3)$$

$$x_{ijk} \leq Mv_{ij} \quad \text{para todo } i, j, k, \quad (4)$$

$$F_i \geq c_i + \sum_{k=1}^N r_{ik} \quad \text{para todo } i, \quad (5)$$

$$c_j \geq F_i + D - M(1 - p_{ij}) \quad \text{para todo } i, j, \text{ aonde } i \neq j, \quad (6)$$

$$c_i \geq F_j + D - Mp_{ij} \quad \text{para todo } i, j, \text{ aonde } i \neq j, \quad (7)$$

$$p_{ii} = 0 \quad \text{para todo } i, \quad (8)$$

$$L_j \geq d_j + \sum_{k=1}^N s_{jk} \quad \text{para todo } j, \quad (9)$$

$$d_j \geq L_i + D - M(1 - q_{ij}) \quad \text{para todo } i, j, \text{ aonde } i \neq j, \quad (10)$$

$$d_i \geq L_j + D - Mq_{ij} \quad \text{para todo } i, j, \text{ aonde } i \neq j, \quad (11)$$

$$q_{ii} = 0 \quad \text{para todo } i, \quad (12)$$

$$L_j \geq c_i + V + \sum_{k=1}^N x_{ijk} - M(1 - v_{ij}) \quad \text{para todo } i, j, \quad (13)$$

todas as variáveis ≥ 0 .

A restrição (1) garante que o *makespan* seja maior ou igual ao tempo (horário) em que o último caminhão de saída, deixa a doca de embarque. A restrição (2) garante que o número total de unidades de produto do tipo k que se transfere de um caminhão de entrada i para todos os caminhões de saída seja exatamente a mesma quantidade de unidades do produto do tipo k do caminhão de entrada i . A restrição (3) garante que o número total de unidades do produto do tipo k que se transfere de todos os caminhões de entrada para o caminhão de saída j , seja exatamente a mesma quantidade de produtos do tipo k requerida pelo caminhão de saída j . A restrição (4) reforça o relacionamento correto entre as variáveis x_{ij} e v_{ij} . As restrições (5)-(7) validam a sequência (horários/tempos) de chegadas e partidas dos caminhões de entrada baseando-se pela ordem em que eles foram designados. A restrição (8) garante que nenhum caminhão de entrada pode preceder-se no escalonamento de caminhões de entrada. As restrições (9)-(11) validam a sequência (horários/tempos) de chegadas e partidas dos caminhões de saída baseando-se pela ordem em que eles foram designados. A restrição (12) garante que nenhum caminhão de saída pode preceder-se na sequência de caminhões de saída. A restrição (13) associa o horário de partida de um caminhão de saída com o horário de chegada de um caminhão de entrada, caso algum produto seja transferido entre os mesmos. O tempo de desembarque de cada produto de um caminhão de entrada e o tempo de embarque em um caminhão de saída, assim como o tempo de desembarque, dos caminhões de entrada para o armazenamento temporário, e o tempo de embarque do armazenamento temporário para um caminhão de saída, são os mesmos para todos os tipos de produtos, com custo de 1 unidade de tempo por transferência. O tempo de troca de caminhões nas docas é de 75 unidades de tempo, e o tempo de transferência da doca de desembarque para a doca de embarque é de 100 unidades de tempo. Todas as operações de transferências podem ser feitas simultaneamente, exceto as operações de carregamento diretamente de um caminhão de entrada para um caminhão de saída e operações de carregamento do armazenamento temporário para um caminhão de saída.

3 Algoritmo Genético (AG)

Os algoritmos genéticos (AGs) são algoritmos de busca que utilizam mecanismos baseados na seleção natural e na genética natural. São algoritmos geralmente aplicados a problemas de otimização, no qual não se conhece um algoritmo determinístico de tempo polinomial que leve a alguma solução ótima ou próxima da ótima, ou nem mesmo se conhece uma solução ótima.

Dada uma população, em que cada indivíduo, denominado cromossomo, representa uma possível solução para o problema, os cromossomos mais aptos (as melhores soluções) da população tenderão a se reproduzir e passar seus genes à gerações seguintes, gerando populações de indivíduos cada vez mais aptos. Neste problema, cada cromossomo é codificado da seguinte maneira: a primeira porção é composta por uma sequência de caminhões de saída. A outra parte é composta pela sequência de caminhões de entrada, como ilustrado na Figura 3. Cada caminhão na sequência é denominado gene.

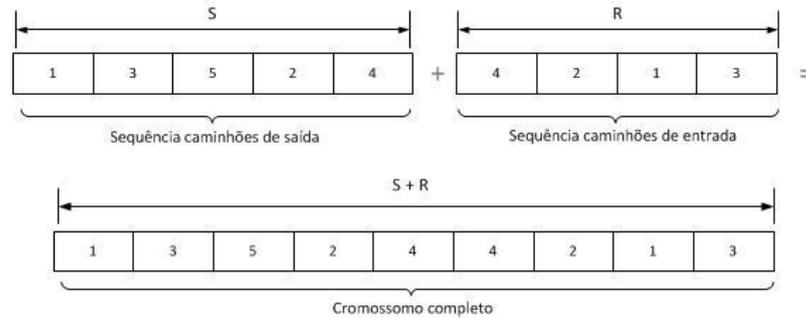


Figura 3 – Formato de um cromossomo.

O AG aplicado ao problema, ilustrado na Figura 4, trata o problema da seguinte forma, primeiramente gera-se aleatoriamente um conjunto de indivíduos (possíveis soluções), formando a população inicial, que no momento também é considerada a população corrente. Em seguida avalia-se a aptidão de cada cromossomo da população corrente. Essa aptidão é medida pelo valor do *makespan* do cromossomo, e quanto menor for mais apto o cromossomo é. Seleciona-se uma parte da população corrente através dos processos de seleção: elitismo, roleta e torneio. Os cromossomos selecionados são copiados para uma população intermediária. A população intermediária passa por mecanismos de reprodução, gerando novos indivíduos que serão acrescentados à mesma. Mecanismos de mutação são aplicados a alguns cromossomos da população corrente que não foram escolhidos pelos processos de seleção, e são acrescentados à população intermediária. Em seguida, a população intermediária passa a ser população corrente e o processo é repetido desde a fase de avaliação de aptidão dos cromossomos, até que um critério de parada seja satisfeito. O critério de parada utilizado é um número máximo de iterações sem que haja melhoras no melhor indivíduo da população.

Algoritmo Genético

1. gerar população inicial $P(0)$
2. $t \leftarrow 0$
3. Avaliar_aptidão($P(t)$)
4. Enquanto critério de parada não for satisfeito faça
5. $t \leftarrow t + 1$
6. $S \leftarrow$ Seleção de $P(t-1)$
7. $R \leftarrow$ Reprodução(S)
8. $M \leftarrow$ Mutação($P(t-1) / S$)
9. $P(t) \leftarrow S \cup R \cup M$
10. Avaliar_aptidão de $P(t)$
11. Retorne (cromossomo mais apto).

Figura 4 – Algoritmo genético aplicado ao problema.

Processos de seleção

Durante o processo de seleção escolhem-se os indivíduos que se reproduzirão, gerando um número determinado de descendentes para a próxima geração. A probabilidade de um indivíduo ser escolhido nesta fase está diretamente relacionada ao seu grau de aptidão. O processo de seleção ocorre após a população corrente ser avaliada. Utilizam-se três formas de escolhas:

- *Elitismo* - Nesta etapa, os melhores cromossomos da população são escolhidos e passam à próxima geração. O número de cromossomos que serão escolhidos é regido pela taxa de elitismo Te . A taxa de elitismo geralmente possui valores entre 0,01 e 0,2, pois taxas muito altas comprometerão a diversidade populacional. Esta escolha garante que os melhores indivíduos da população permanecerão na próxima geração, acelerando a convergência da busca.
- *Torneio* - Neste processo de escolha, seleciona-se um número n (geralmente $n = 3$) de cromossomos da população corrente, então o cromossomo mais apto dentre os n cromossomos é escolhido para a população intermediária. Repete-se esse processo até que um determinado número de cromossomos tenha sido escolhido.
- *Roleta* - Este método simula uma roleta em que cada indivíduo representa um intervalo, de forma que quanto mais apto for o indivíduo, maior será o seu intervalo na roleta. Portanto indivíduos mais aptos têm mais chances de serem escolhidos. A cada iteração, seleciona-se um indivíduo, até que um número determinado de cromossomos tenha sido escolhido.

Mecanismos de reprodução / Crossover

Neste processo, escolhem-se aleatoriamente dois cromossomos da população intermediária, P1 e P2, gerando dois novos indivíduos, F1 e F2, que conterão características de P1 e P2. A ocorrência do *crossover* é quantificada pela taxa de reprodução Tr , que representa a porcentagem em relação aos cromossomos escolhidos na etapa de seleção que se reproduzirão. Os mecanismos de *crossover* utilizados nessa pesquisa foram:

- *Crossover de dois pontos* - Escolhem-se aleatoriamente dois pontos, denominados pontos de *crossover*, referentes aos genes dos cromossomos P1 e P2, os genes de P1, contidos entre os pontos de *crossover* são copiados para F2, os genes contidos entre o mesmo intervalo de pontos de P2 são passados à F1, na mesma sequência. Logo após, os genes restantes de P1 são passados à F1, e os genes restantes de P2 são passados à F2. Note que, após serem gerados, os cromossomos F1 e F2 podem conter genes repetidos. Estes genes devem ser trocados para se obter uma sequência correta. Este mecanismo é aplicado separadamente para a sequência de caminhos de entrada e a sequência de caminhos de saída.
- *Crossover uniforme* - A sequência de caminhos de saída do cromossomo P1 é copiada para o cromossomo F2, o mesmo processo é feito de maneira análoga entre P2 e F1. A sequência de caminhos de entrada de P1 é copiada para F1, o mesmo processo é feito de maneira análoga entre P2 e F1.

Mecanismos de mutação:

A mutação garante uma diversificação na população, através de pequenas alterações nos cromossomos. A probabilidade de ocorrência de mutação, Tm , geralmente está entre 0,00 e 0,1. A mutação é aplicada separadamente para a sequência de caminhos de saída e para a sequência de caminhos de entrada. Usam-se três mecanismos de mutação:

- *Troca de posições* - Escolhem-se aleatoriamente dois genes do cromossomo e então suas posições são trocadas.
- *Inversão* - Neste mecanismo, escolhe-se dois genes, trocam-se suas posições, e todos os genes que estão contidos entre eles são invertidos.
- *Inserção* - Nesse processo, um gene do cromossomo é escolhido aleatoriamente. Este gene então passa para a última posição da respectiva sequência de genes do cromossomo.

4 Testes computacionais

A meta-heurística foi codificada na linguagem C++ e os testes foram realizados em um computador com processador Intel core I5 2,67GHz com 4 GB de memória RAM e sistema operacional Windows 7 Ultimate. A meta-heurística foi testada em um conjunto de instâncias (1-20) geradas por Yu e Egbelu (2008) e em um segundo conjunto de instâncias de maior porte (21-29), geradas aleatoriamente.

Para determinar os melhores parâmetros do AG, foram realizados cerca de 5 mil testes (aproximadamente 173 testes para cada instância do problema), variando-se a taxa de reprodução, seleção, e mutação, assim como os mecanismos usados em cada uma delas. O tamanho da população foi 100. Os parâmetros de reprodução (Tr) foram divididos entre: taxa de *crossover* de dois pontos (Trd), e taxa *crossover* uniforme (Tru). Os parâmetros de seleção foram divididos em: taxa de elitismo (Te), taxa de seleção por roleta (Tsr) e taxa de seleção por torneio (Tst). Os melhores resultados foram obtidos para os parâmetros ($Te = 0,01$; $Tsr = 0,2$; $Tst = 0,1$; $Tr = 1$; $Tru = 0,5$; $Trd = 0,5$; $Tm = 0,05$). O critério de parada usado foi 500 iterações sem melhora na melhor solução.

Resultados

As informações das instâncias, e os resultados computacionais são apresentados na Tabela 1. A primeira coluna apresenta o número de cada instância. A segunda e terceira colunas apresentam, respectivamente, a quantidade de caminhos de entrada e saída. A quarta coluna apresenta a quantidade de tipos de produtos do problema. A quinta coluna apresenta o total de produtos envolvidos. A sexta coluna apresenta os valores ótimos, a sétima coluna apresenta o melhor valor obtido pelas heurísticas propostas por Yu e Egbelu (2008). As colunas 8-11 apresentam os valores obtidos pelo AG. A última coluna apresenta o tempo computacional do AG. O AG foi testado 5 vezes para cada instância com os parâmetros mencionados anteriormente. Para todas as instâncias do conjunto 1-20, a meta-heurística AG obteve soluções ótimas superando os resultados apresentados por Yu e Egbelu (2008). O tempo computacional utilizado pela meta-heurística supera o tempo apresentado por Yu e Egbelu (2008). Para as 9 instâncias (21-29) é inviável obter a solução ótima utilizando pacotes de otimização. Para essas instâncias aplicou-se somente o algoritmo genético.

Instâncias	Tamanho				Makespan					Tempo
	# caminhões de entrada	# caminhões de saída	# Tipos	# Total de produtos	Ótimo	Algoritmo Genético (AG)				
						Yu-Egbelu (2008) Melhor	Melhor	Pior	Médio	
1	4	5	4	990	1557	1569	1557	1557	1557	0,224
2	5	4	6	1030	1577	1577	1577	1577	1577	0,284
3	3	3	8	890	1372	1372	1372	1372	1372	0,192
4	5	5	8	1000	1749	1789	1749	1749	1749	0,436
5	5	3	8	960	1579	1652	1579	1579	1579	0,282
6	4	4	5	1020	1546	1546	1546	1546	1546	0,214
7	5	4	6	980	1535	1535	1535	1535	1535	0,292
8	3	5	7	890	1525	1525	1525	1525	1525	0,259
9	5	4	8	900	1473	1473	1473	1473	1473	0,301
10	6	4	9	930	1452	1452	1452	1452	1452	0,258
11	5	4	6	1620	2232	2232	2232	2232	2232	0,291
12	6	4	8	1950	2833	2862	2833	2833	2833	0,516
13	5	6	8	1610	2386	2490	2386	2386	2386	0,510
14	5	5	8	1680	2385	2413	2385	2385	2385	0,448
15	6	5	4	2030	2745	2762	2745	2745	2745	0,312
16	5	6	6	1690	2407	2407	2407	2407	2407	0,455
17	4	4	7	1180	1867	1885	1867	1867	1867	0,267
18	6	6	7	1770	2502	2642	2502	2502	2502	0,620
19	5	5	10	1720	2553	2639	2553	2553	2553	0,555
20	6	6	9	2020	2732	3036	2732	2732	2732	0,673
21	10	9	10	7846	---	---	9850	10382	10068	2,894
22	16	15	12	5188	---	---	7164	8021	7476,8	10,753
23	12	15	15	9073	---	---	12132	12518	12360,4	10,008
24	15	15	15	5233	---	---	7458	7759	7607,4	9,166
25	16	17	18	7884	---	---	10588	11225	11012,4	18,609
26	16	16	20	9515	---	---	12742	13568	13200,2	17,796
27	19	16	15	6413	---	---	8999	9519	9289,4	18,231
28	18	19	16	7050	---	---	9744	10216	9919,2	20,025
29	20	20	18	8700	---	---	11983	12692	12386,2	29,618

Tabela 1 – Resultados computacionais e informações sobre as instâncias.

5 Conclusões

Esta pesquisa apresenta um Algoritmo Genético, para escalonar os caminhões de entrada e saída de um modelo de *cross-docking*. A meta-heurística foi aplicada em um conjunto contendo 29 instâncias. Para as instâncias 1 à 20 a meta-

heurística obteve resultados ótimos em todas as instâncias. Pesquisas futuras incluem o estudo e o desenvolvimento de meta-heurísticas capazes de diminuir o esforço computacional para tratar do problema de *cross-docking* com diversas docas de embarque e desembarque e do *cross-docking* integrado a problemas de roteamento envolvendo, janelas de tempo na distribuição dos produtos e frota de veículos heterogênea. Os autores agradecem o apoio financeiro do programa PIBIC/CNPQ e FUNCAP.

6 Referências

- [1] Apte U. M. e Viswanathan S. Effective Cross Docking for Improving Distribution Efficiencies. **International Journal of Logistics: Research and Applications**, 291-302 (2000).
- [2] Arabani A. R., Ghomi F. e Zandieh M., Meta-heuristics implementation for scheduling of trucks in a *cross-docking* system with temporary storage, **Expert System with Applications**, 38 (2010), 1964-1979.
- [3] Boysen N. e Fließner M., Cross dock scheduling: Classification, literature review and research agenda. **Omega, Article In Press** (2009).
- [4] Forger e Gary, 1995. UPS starts World's premiere *cross-docking* operation. **Modern Materials Handling**, 36-38
- [5] Gue K. R., The effect of trailer scheduling on the layout of freights terminals. **Transportation Science**, 33 (1999), 419-428.
- [6] Lau H. C., Sim M. e Teo K. M., Vehicle routing problem with time Windows and a limited number of vehicles. **European Journal of Operation Research**, 148 (2003), 559-569.
- [7] Lee Y. H., Jung W. J. e Lee K. M., Vehicle routing scheduling for *cross-docking* in the supply chain. **Computers & Industrial Engineering**, 51 (2003), 247-256.
- [8] Vahdani B. e Zandieh M., Scheduling trucks in *cross-docking* systems: Robust meta-heuristics, **Computer & Industrial Engineering, Article In Press** (2010).
- [9] Yu W. e Egbelu P. J., Scheduling of inbound and outbound trucks in cross docking systems with temporary storage. **European Journal of Operational Research**, 184 (2008), 377-396.
- [10] Witt e Clyd E., Crossdocking: Concepts demand choice. **Material Handling Engineering**, 53 (1998), 44-49.