

PLANEJAMENTO DE CENTROS DE DISTRIBUIÇÃO TIPO CROSSDOCKING

Antonio G.N. Novaes, Lalt, Unicamp

Edson T. Bez, Univ. Vale do Itajaí, SC

Daniel D. Adriano, Univ. Vale do Itajaí, SC

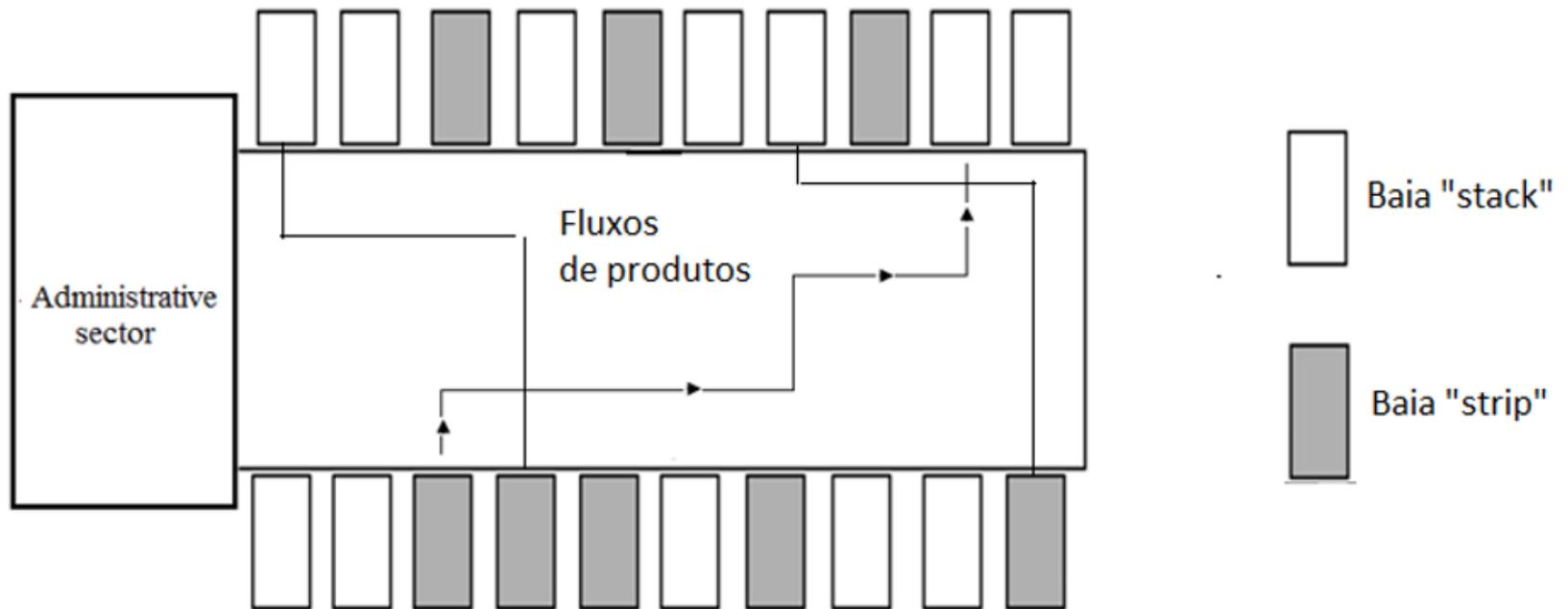
Abstract

- **Cross docking is a form of operating break-bulk freight terminals in which the receiving products are not stocked in the traditional way, but, instead, the incoming cargo is transferred internally to be loaded into distribution trucks as quickly as possible, as to take the products to retail stores. With this approach, inventory costs and lead-times are significantly reduced. The objective function of the problem is to minimize the sum of the rent cost of the terminal facility, the waiting cost of inbound trailers when being attended at the terminal, and the cost of handling and moving freight inside the terminal. These costs are deeply interrelated and are analysed in an integrated way.**

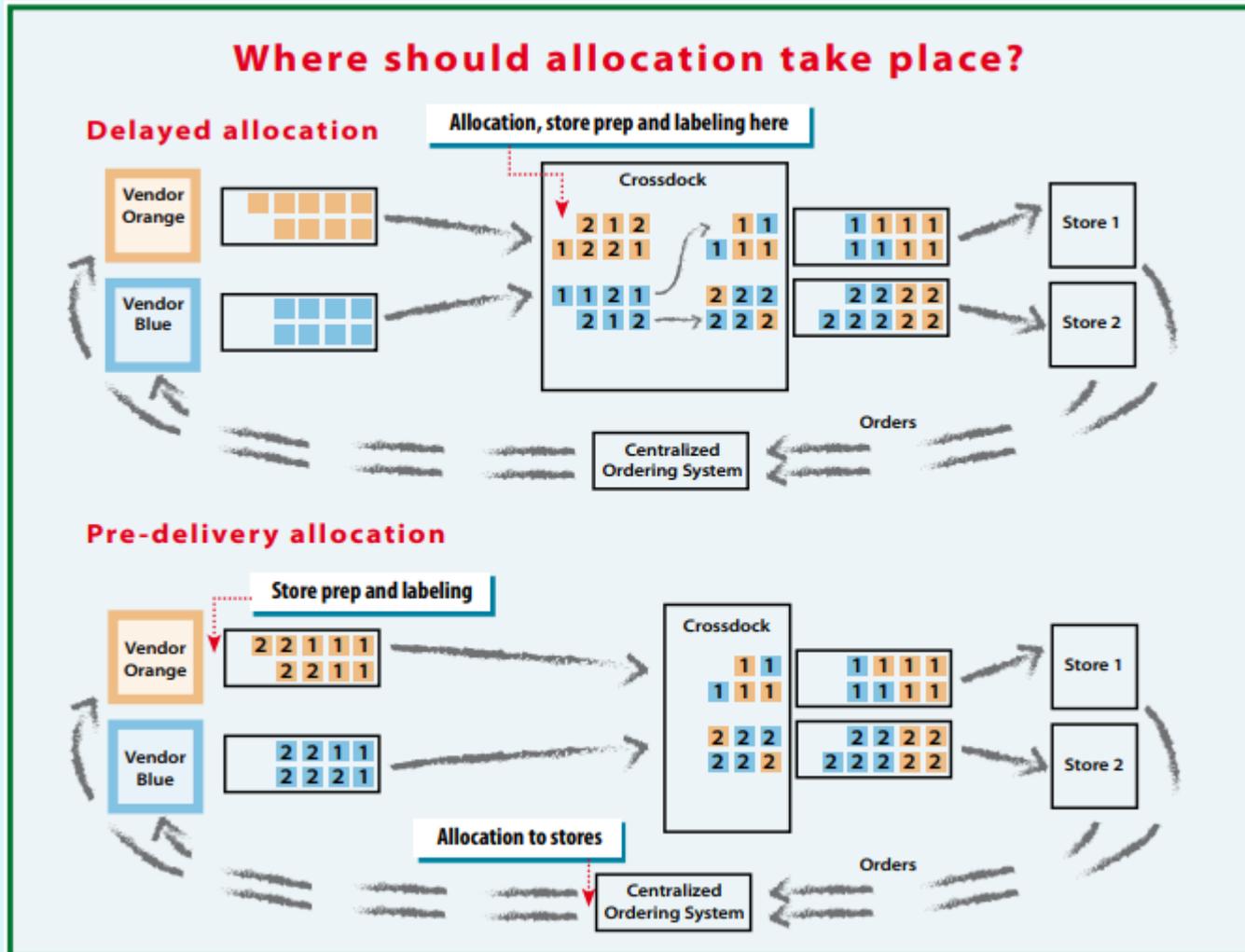
Esquema geral

- Centros de crossdocking são terminais que operam produtos *less-than-truckload (LTL)* nos quais os produtos são descarregados, conferidos, endereçados, triados, depois consolidados, e finalmente carregados em veículos locais, que levam a mercadoria para pontos na cadeia de suprimento (lojas). Além de gerar custos apreciáveis, o manuseio de produtos é importante porque o tempo que uma remessa gasta no terminal é desperdiçado sob o ponto de vista de que a mercadoria não está se movimentando em direção ao seu destino final, a prateleira do supermercado, ou mais apropriadamente, às mãos do consumidor (Gue, 1999).
- Neste *case* vamos abordar um centro de triagem e distribuição de produtos para uma rede de supermercados, na qual os fornecedores transportam os produtos até o terminal em horários pré-marcados, que são descarregadas em baias específicas denominadas **strip-doors**. Em seguida os produtos são endereçados e submetidos à triagem interna. Depois são direcionados às baias voltadas à distribuição para as lojas (**stack-doors**), e daí saem nos veículos locais para as entregas às lojas.

Esquema de movimentação interna



Fluxos de produtos dentro do terminal (Gue, 2007)



Otimização do layout de um crossdocking

- A operação eficiente de um terminal de crossdocking exige uma coordenação apropriada entre os fluxos de chegada e de saída de produtos. De fato, uma parte substancial dos custos operacionais de crossdocking é devida ao intenso movimento de funcionários e de equipamentos de manuseio de materiais dentro da instalação, quando os produtos são transferidos das baias de recepção, para as baias dos veículos de entrega dos produtos às lojas, e essa operação costuma ter custos relativamente altos. Como resposta a esses custos, grande parte dos modelos de otimização de terminais de crossdocking, principalmente em casos de redes de supermercados e de outras grandes organizações varejistas, têm sido desenvolvidos e aplicados com o objetivo de chegar a layouts adequados do terminal, com a minimização dos custos de movimentação interna do material. E, para isso, é necessário buscar um arranjo otimizado das baias de recepção (*strip doors*) e das baias de despacho (*stack doors*).
-

Gerando fluxos de produtos no abastecimento do terminal

- **Dados reais sobre o mix de produtos em cada caminhão de abastecimento no terminal são difíceis de obter no Brasil, em parte devido à razões de sigilo, mas também porque as empresas não organizam os dados na forma preconizada neste tipo de operação. Além disso, os operadores logísticos, que operam veículos LTL, não registram normalmente os dados de forma a identificar detalhadamente a formação do conteúdo de cada remessa (Gue, 1999). Assim, para maior detalhamento da análise, adotamos a carga representada por “*cartons*”, isto é, caixas de papelão contendo os produtos, que, em geral são transportados em *pallets*, contendo os cartons. E, seguindo Bartholdi & Gue (2004), os fluxos de produtos na sequência de abastecimento do terminal, são gerados neste caso através de um experimento de Monte Carlo, suprimindo assim o modelo com dados pseudo-realistas. O algoritmo para a geração de tais informações é mostrado na Fig. 2, a seguir.**

Fig. 2 - Pseudocode do processo de Monte Carlo para gerar lotes de *cartons* no suprimento

Algorithm 1. Cargo Assignment

```
1  Set  $\emptyset_j$  the total number of cartons to be sent to destination  $j$  per working day
2  for all  $k$ , given trailer  $k$  is available, do
3      Stage 1 (greedy loading)
4      for all  $j$ , given destination  $j$  is available, do
5          Set a random load of cartons respecting  $\emptyset_j$  and trailer capacity limits
6          Set temporary cargo load of vehicle  $k$ , for destination  $j$ 
7      end for
8      if trailer  $k$  is fully loaded then
9          Next trailer
10     else
11         Stage 2 (step-by-step loading)
12         while trailer is not fully loaded do
13             for all  $j$ , given destination  $j$  is available do
14                 Add one carton unit for destination  $j$ , if truck load is not complete
15             end for
16         end while
17     end if
18 end for
```

Estágios do no processo de Monte Carlo

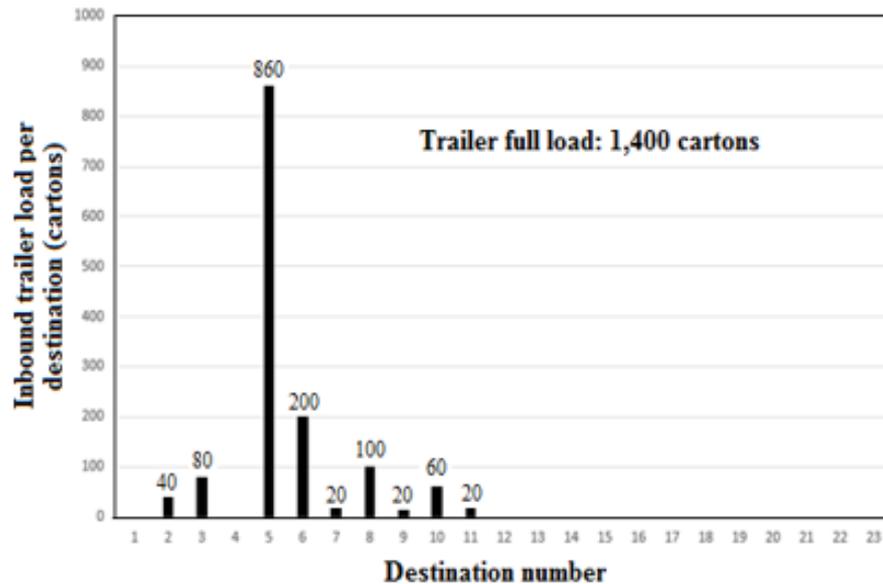
- Conforme mostra a Fig. 2, o processo de Monte Carlo é composto de dois estágios. O primeiro é uma rotina gulosa, que tenta colocar no veículo de uma vez a maior quantidade de *cartons* por vez. Mas, ao chegar próximo à situação de lotação do veículo, a operação de carregamento fica cada vez mais minuciosa, tentando colocar poucas unidades por vez, de forma a não extrapolar o espaço ainda disponível no caminhão.
- O processo de Monte Carlo gera remessas de produtos, onde $V_{k,j}$ expressa a remessa k ($k = 1, 2, \dots, n$), medida em *cartons*, e transportada pelo caminhão fornecedor, e destinada à loja j ($j = 1, 2, \dots, m$) da empresa supermercadista. Cada caminhão fornecedor tem capacidade de C_T *cartons*. Assim, O processo de Monte Carlo deve respeitar a capacidade do veículo através da restrição:

$$\bullet \sum_{j=1}^m V_{k,j} \leq C_T, \quad \text{for } k = 1, 2 \dots n, \quad (1)$$

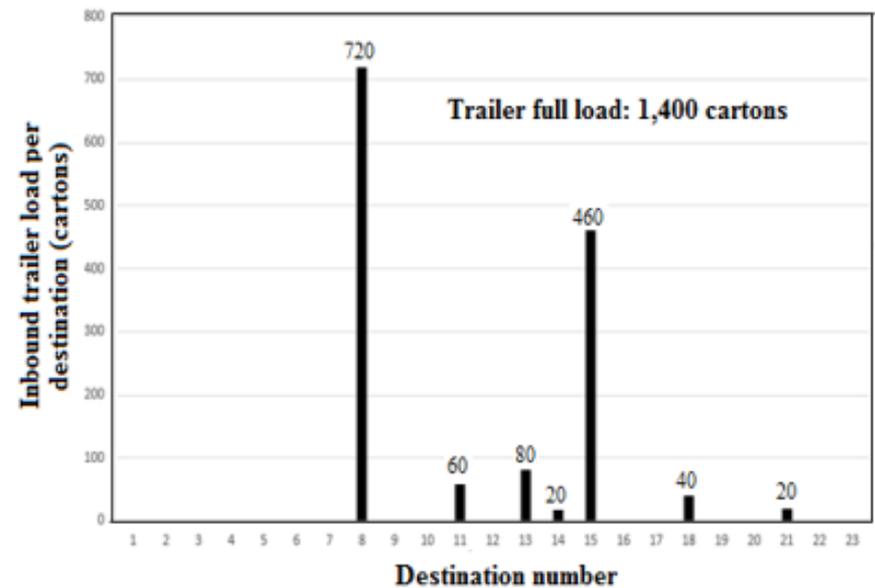
- Por outro lado, cada loja j da rede de supermercados apresenta uma demanda diária de \emptyset_j *cartons*, observando assim à seguinte relação:

$$\bullet \sum_{k=1}^n V_{k,j} = \emptyset_j, \quad \text{for } j = 1, 2, \dots, m, \quad (2)$$

Fig. 3 – Resultados, processo de Monte Carlo



(a)



(b)

Exemplos de carregamento de caminhões fornecedores (Monte Carlo)

Modelo de alocação de caminhões às baias strip

- Modelos mais recentes sobre otimização do layout de terminais de crossdocking consideram problemas com múltiplas baias do tipo *strip*, que recebem só caminhões fornecedores de produtos, como também consideram múltiplas baias do tipo *stack*, que recebem só veículos menores, locais, que fazem as entregas dos produtos às lojas (Van Belle et al. 2012).
- Mas tais modelos geralmente só consideram a alocação ótima dos caminhões fornecedores que usam as baias do tipo *strip*, porque a alocação dos veículos às baias *stack* é normalmente realizada para prazos bem maiores (uma estação, por exemplo), não mudando em prazos curtos (Van Belle et al. 2012). Assim, no nosso *case*, a alocação dos veículos às respectivas baias *stack*, veículos esses que fazem as entregas dos produtos às lojas, segue um processo fixo por um bom período de tempo.

Alocação das baias *strip* para os caminhões fornecedores

- Em geral, a alocação das baias *strip* aos caminhões fornecedores visa minimizar as distâncias percorridas dentro do terminal, desde o ponto de chegada do lote na doca de recebimento, até a baia de despacho, na doca respectiva. Quando esse deslocamento é realizado manualmente, utiliza-se o trajeto Manhattan, em segmentos perpendiculares, a 90° (Bartholdi & Gue, 2004).
- Nesse tipo de deslocamento, em terminais retangulares (I-shaped), é fácil perceber que as baias *strip* no centro da doca são as mais convenientes por apresentarem menores distâncias médias em relação às baias *stack*, que ficam geralmente no outro lado do edifício (Bartholdi & Gue, 2004). Na prática, um critério razoável de decisão é considerar, dentre todas as baias *strip* ainda não ocupadas, aquela apresentando a menor distância média até as baias *stack*.

Variáveis

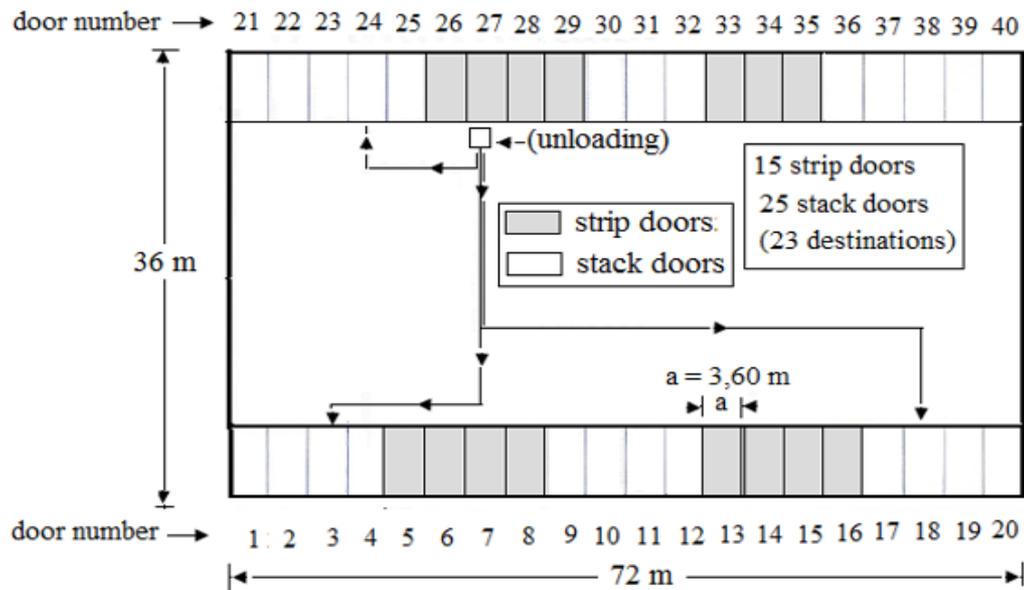
- Seguindo Barthold & Gue (2004), são as seguintes as variáveis:
- n_V = número de caminhões fornecedores por dia útil;
- n = número de baias *strip*;
- m = número de baias *stack* (igual ao número de destinos (lojas));
- i = uma baia *strip*, $i = 1, 2, \dots, n$;
- j = um destino (loja), para o qual uma remessa diária de produtos $V_{k,j}$, é enviada;
- $d_{i,j}$ = distância retangular entre a baia *strip* i até a baia *stack* j ;
- C_T = capacidade de carga do caminhão fornecedor (em *cartons*);
- Φ_j = quantidade de produto enviado diariamente ao destino j , em *cartons*:

Modelo para otimizar a alocação de baias *strip*

- De acordo com Bartholdi & Gue (2004), o objetivo é minimizar a distância total interna média percorrida pelos funcionários, multiplicada pelas respectivas quantidades de cartons

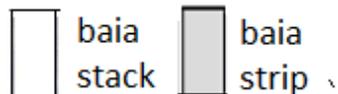
- Minimize $\bar{d} = \frac{\sum_{i=1}^n \sum_{j=1}^m \phi_j \times d_{i,j}}{\sum_{j=1}^m \phi_j}$. (3)

- A distância percorrida media \bar{d} é uma aproximação relativa para o custo de deslocamento interno no terminal, (Bartholdi & Gue, 2000, 2004). O problema de otimização (4) é um problema combinatório com a seguinte estrutura:
 - de um total de $n + m$ baias disponíveis, selecionar n baias como *strip*, e m baias como *stack*;
 - alocar cada uma das baias *stack* para um destino, de acordo com esquemas operacionais internos;
 - As baias *strip*, por outro lado, são alocadas às diversas remessas de caminhões fornecedores, ao longo do dia, de acordo com a disciplina pré-determinada de atendimento e de outros fatores;



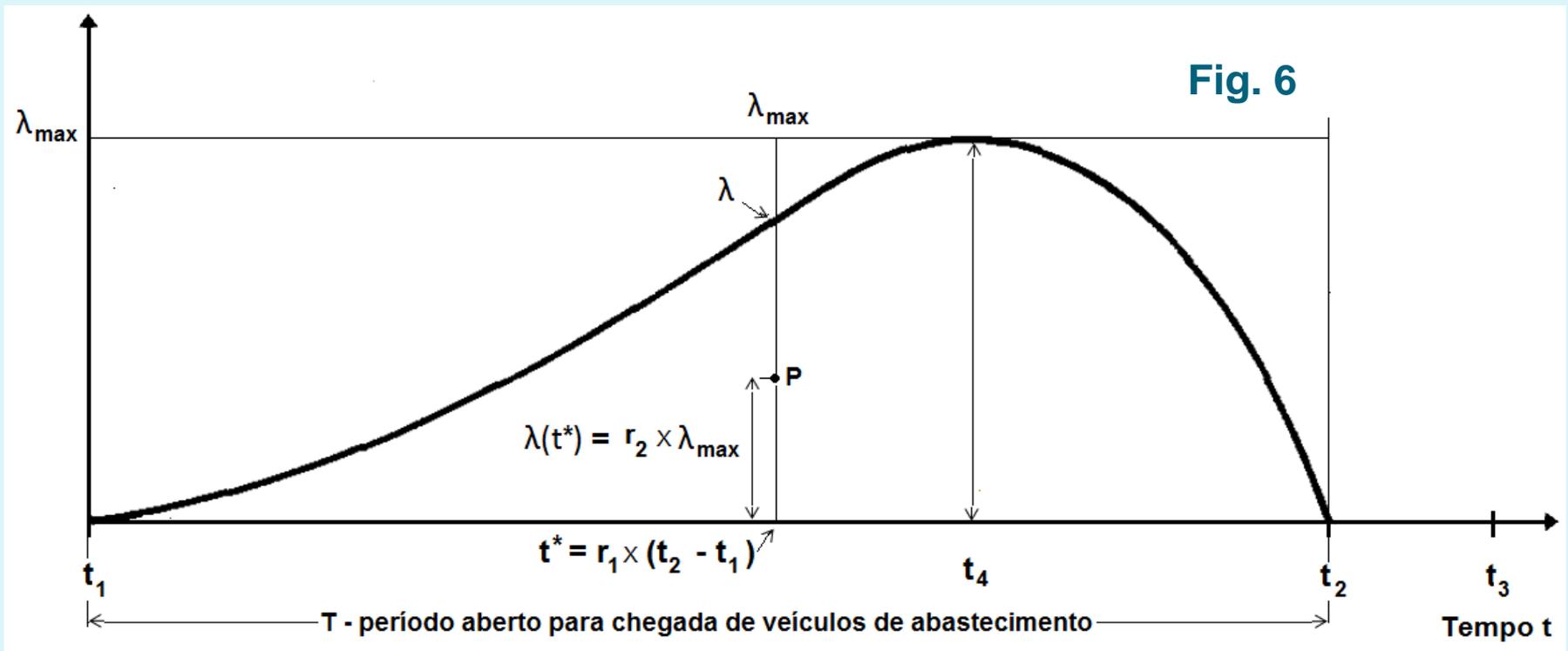
Stack door assignment

(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)
Destina- tion	Cartons/ day	Stack door	Average dist. (m)	Destina- tion	Cartons/ day	Stack door	Average dist. (m)
1	7,740	9,10	29.52	13	3,340	18	44.16
2	5,040	11,12	30.00	14	3,340	23	45.84
3	3,900	30	32.16	15	3,340	38	46.56
4	3,620	31	32.40	16	3,340	2	47.04
5	3,460	32	32.64	17	3,340	19	47.76
6	3,400	25	38.64	18	3,340	22	49.44
7	3,380	36	39.36	19	3,340	39	50.16
8	3,360	4	39.84	20	3,340	1	50.64
9	3,340	17	40.56	21	3,340	20	51.36
10	3,340	24	42.24	22	3,340	21	53.04
11	3,340	37	42.96	23	3,340	40	53.76
12	3,340	3	43.44				



Alocação de baias strip e stack

A simulação das chegadas de caminhões de abastecimento no terminal (mesmo método usado no *case 1*)

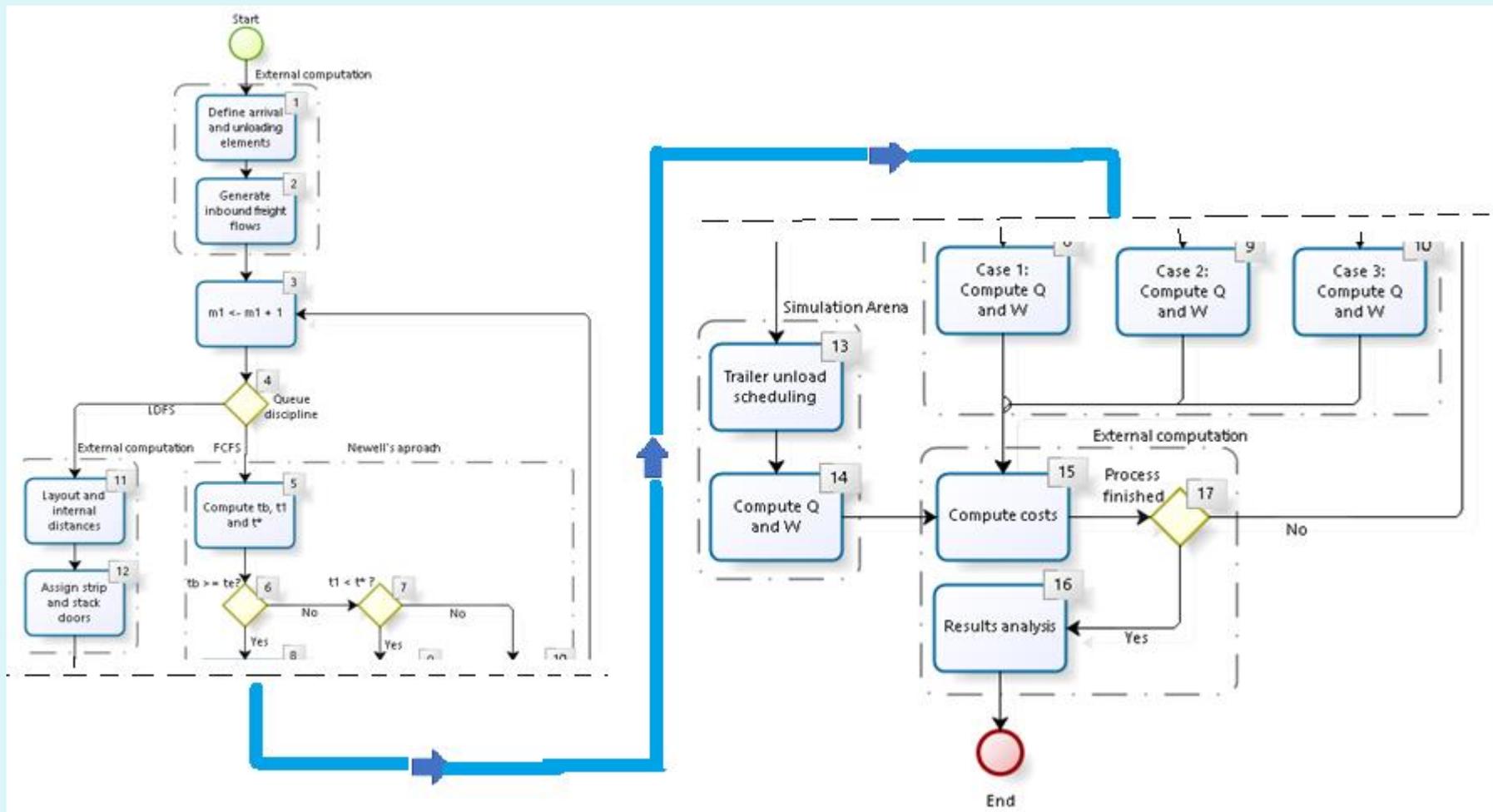


Pseudo-code para gerar sequência de chegadas de caminhões fornecedores no terminal

Algorithm 2. Routine Lambda

```
1  for all  $k$ , given trailer  $k$  is available do
2      while arrival time of trailer  $k$  has been not accepted do
3          Generate a random number  $0 \leq r_1 \leq 1$ 
4          Determine time  $t_K = r_1 \times (t_E - t_0)$ 
5          Compute  $\lambda(t_k)$ 
6          Generate a new random number  $0 \leq r_2 \leq 1$ 
7          Compute  $z = r_2 \times \lambda_{max}$  where  $\lambda_{max} = \lambda(t^*)$ , with  $t^* = \left(\frac{2}{3}\right)L$ 
8          if  $z \leq \lambda(t_k)$  then
9              Accept  $t_k$  as the arrival time of trailer  $k$ 
10         end if
11     end while
12 end for
13 Sort the set  $t_K$  according to the increasing order of magnitude of  $t_k$ , thus forming the simulated arrival sequence of vehicles and the corresponding arrival times
```

Fluxograma do modelo

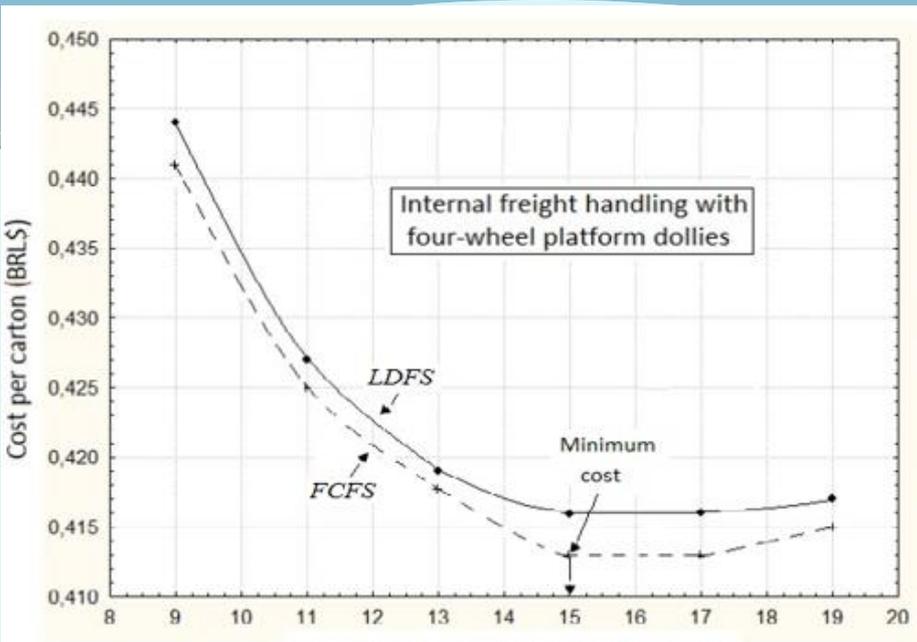


Input de dados para o modelo

Input symbol	Description	Value	Unit
G	Total freight flow per day	84,000	Cartons
n_V	Number inbound vehicles	60	Vehicles/day
m_1	Number of strip doors (main variable)	9,11,13,15, 17,19	Units
m_2	Number of freight destinations	23	Units
$m_2 + 2$	Number of stack doors (see Section 4.2.1)	25	Units
C_T	Trailer capacity	1,400	Cartons
L	Trailer unloading time window extension	8	Hours
$\bar{\lambda}$	Average trailer arrival rate	10,500	Cartons/hour
λ_{max}	Maximum trailer arrival rate	18,666.67	Cartons/hour
\bar{s}	Average trailer unloading rate ^(*)	933.33	Cartons/hour
σ_s	Standard deviation trailer unloading rate ^(*)	280.00	Cartons/hour

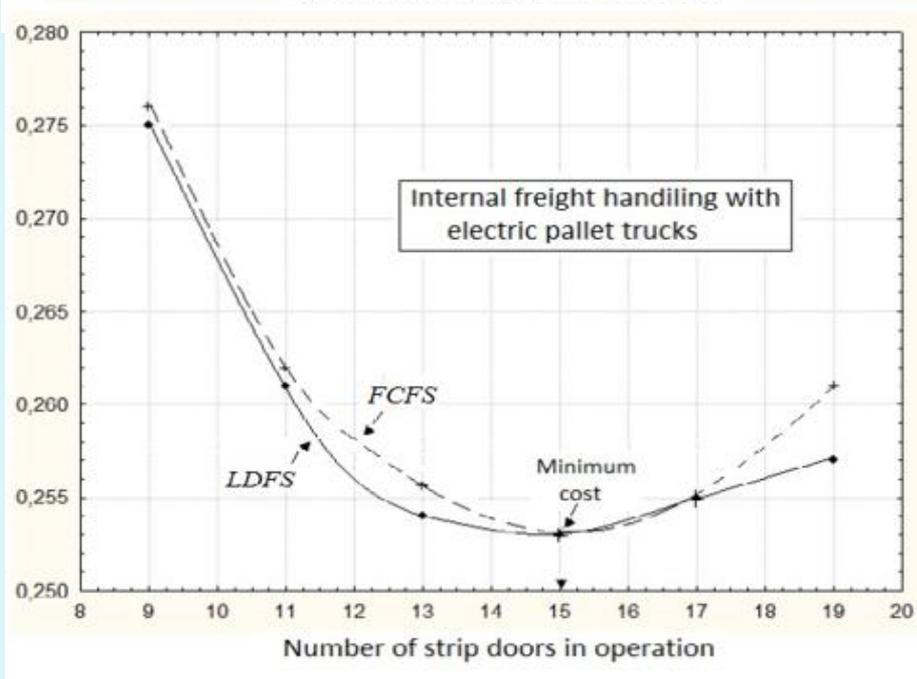
Critério LDFS para priorizar caminhões na fila

- No que se refere à fila dos caminhões fornecedores a serem atendidos na doca de recepção, a estratégia tradicional era a *FCFS* (first-come, first-served), com atendimento na ordem da chegada. Mas no caso do crossdocking, em que as distâncias percorridas pelos fluxos internos no armazém geram muitas vezes custos mais altos do que os custos de espera dos caminhões, usa-se o critério *LDFS* (least-distance, first-served). Nesses casos, quando uma baia *strip i* fica vaga, busca-se o caminhão fornecedor na fila de espera tal que a distância média entre a baia vaga *i* e as baias *k stack*, seja mínima. Noutras palavras, agora a ordem de chegada do caminhão não influencia a ordem de atendimento, sendo a distância média esperada o critério de seleção.



Deslocamento interno com carrinhos 4 rodas

(Número ótimo de baias strip e custo médio por carton)



Deslocamento interno com paleteiras elétricas

Resultados finais

- Um total de 40 baias é indicado, distribuídas ao longo dos dois lados mais longos de um terminal de formato “T” (Fig. 5);
- Um total de 15 baias strip;
- Um total de 23 destinos (lojas), 25 baias stack, em que os destinos 1 e 2 exigem duas baias stack cada um, em razão da maior demanda;
- O layout das baias stack e respectivas alocações como indicadas na Figura 5;
- A descarga dos caminhões fornecedores obedece o critério *LDFS* de preferência no atendimento;
- Os caminhões fornecedores esperam, em média 31 minutos para serem descarregados (modelo simplificado de fila apud Newell, 1971);
- A fila média dos caminhões fornecedores é de 1,2 veículos (apud Newell, 1971)
- Uma área interna de cerca 3.000 m² será necessária;
- A utilização de paleteiras elétricas apresentou menor custo por “carton”;
- Custo esperado médio de BRL\$ 0,25 por “carton”;