

---

Programação de múltiplos *cross-docks* com  
múltiplas docas

***Pâmella Sátiko Miyazaki Tenório***

---



SERVIÇO DE PÓS-GRADUAÇÃO DO ICMC-USP

Data de Depósito:

Assinatura: \_\_\_\_\_

**Pâmella Sátiko Miyazaki Tenório**

## Programação de múltiplos *cross-docks* com múltiplas docas

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação – ICMC-USP, como parte dos requisitos para obtenção do título de Mestra em Ciências – Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

Área de Concentração: Ciências de Computação e Matemática Computacional

Orientadora: Profa. Dra. Franklina Maria Bragion de Toledo

**USP – São Carlos**  
**Agosto de 2016**

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi  
e Seção Técnica de Informática, ICMC/USP,  
com os dados fornecidos pelo(a) autor(a)

M618p Miyazaki Tenório, Pâmella Sátiko  
Programação de múltiplos *cross-docks* com  
múltiplas docas / Pâmella Sátiko Miyazaki Tenório;  
orientadora Franklina Maria Bragion de Toledo. - São  
Carlos - SP, 2016.  
94 p.

Dissertação (Mestrado - Programa de Pós-Graduação  
em Ciências de Computação e Matemática Computacional)  
- Instituto de Ciências Matemáticas e de Computação,  
Universidade de São Paulo, 2016.

1. *Cross-docking*. 2. Modelagem matemática.  
3. Sincronização. 4. Heurística. I. Toledo, Franklina  
Maria Bragion de, orient. II. Título.

**Pâmella Sátiko Miyazaki Tenório**

## Scheduling of multiple cross-docks with multiple docks

Master dissertation submitted to the Instituto de Ciências Matemáticas e de Computação – ICMC-USP, in partial fulfillment of the requirements for the degree of the Master Program in Computer Science and Computational Mathematics. *FINAL VERSION*

Concentration Area: Computer Science and Computational Mathematics

Advisor: Profa. Dra. Franklina Maria Bragion de Toledo

**USP – São Carlos**  
**August 2016**



*Este trabalho é dedicado à minha família, pelo amor e apoio, e a toda comunidade científica, que sempre busca melhorar o dia-a-dia com pesquisas.  
Em especial, a todos os pesquisadores do Instituto de Ciências Matemáticas, Computação e Estatística (ICMC).*





# AGRADECIMENTOS

---

---

Os agradecimentos principais são direcionados à banca, ao apoio financeiro do CNPq (132271/2014-1, 306918/2014-5), da FAPESP (2013/07375-0) e a Deus.

Agradecimentos especiais são direcionados à Professora Doutora Franklina Maria Bragion de Toledo, pelo apoio e excelente trabalho de orientação, a todos que compõem o Instituto de Ciências matemáticas, Computação e Estatística da Universidade de São Paulo (ICMC/USP) e ao Laboratório de Otimização do ICMC/USP (LOt - [www.otm.icmc.usp.br](http://www.otm.icmc.usp.br)).



*“Os que se encantam com a prática sem ciência  
são como os timoneiros que entram no navio  
sem timão nem bússola, nunca tendo  
certeza do seu destino.”  
(Leonardo da Vinci)*



# RESUMO

MIYAZAKI TENÓRIO, P. S.. **Programação de múltiplos *cross-docks* com múltiplas docas.** 2016. 94 f. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

Cadeias de suprimentos podem ter operações seguindo diferentes estratégias de distribuição e a utilização de cada uma dessas estratégias pode resultar em diferentes operações e custos. A estratégia de *cross-docking* auxilia na redução dos custos de distribuição de produtos, consolidando cargas, e a redução de tempo e custos de armazenamento, uma vez que o tempo máximo de estoque permitido pela estratégia é de cerca de 24 horas. O objetivo deste trabalho é apresentar um modelo para o problema de *cross-docking*, em que cargas são entregues e reorganizadas de forma a atender a outras cargas que são coletadas e garantir que as janelas de tempo para início das operações sejam atendidas. Devido à falta de instâncias para o problema disponíveis na literatura, buscou-se gerar um *benchmark* e disponibilizá-las à comunidade científica. Uma vez que o problema é de difícil solução exata, um método heurístico para a resolução do problema foi desenvolvido. Os resultados mostraram que o modelo proposto resulta em boas soluções quando comparado ao modelo da literatura. O estudo de calibração do software IBM CPLEX mostrou que a calibração dos parâmetros pode resultar em melhores soluções e, por fim, a matheurística se mostrou competitiva com o CPLEX, principalmente para cenários em que a proporção de entregas e coletas diverge.

**Palavras-chave:** *Cross-docking*, Modelagem matemática, Sincronização, Heurística.



# ABSTRACT

MIYAZAKI TENÓRIO, P. S.. **Programação de múltiplos *cross-docks* com múltiplas docas.** 2016. 94 f. Dissertação (Mestrado em Ciências – Ciências de Computação e Matemática Computacional) – Instituto de Ciências Matemáticas e de Computação (ICMC/USP), São Carlos – SP.

Supply chains may have operations which follow different distribution strategies and each one of these strategies may result in different operations and costs. The Cross-docking strategy helps to reduce the products' distribution costs by consolidating loads and reducing storage costs as the maximum inventory time is approximately 24 hours. The aim of this research is to present a model for the cross-docking problem where loads are delivered and reorganized so as to cater for other loads that are collected and ensure that time windows are respected. Due to the lack of instances available in the literature, a benchmark was generated and was made available to the scientific community. As the problem is difficult to obtain the exact solution, a heuristic method was developed. The results showed that the proposed model has good solutions when compared to the literature model. A study of the IBM CPLEX software showed that tuning can result in better solutions and the matheuristics was competitive with the software, mainly in scenarios where deliveries and pickups are very different.

**Key-words:** Cross-docking, Mathematical modelling, Synchronization, Heuristics.





# LISTA DE ILUSTRAÇÕES

---

---

Figura 1 – Percentual de cada operação em um armazém. . . . .	26
Figura 2 – <i>Cross-docking</i> com <i>cross-dock</i> único. . . . .	29
Figura 3 – Centro de distribuição ( <i>cross-docking</i> ) com três <i>cross-docks</i> . . . . .	30
Figura 4 – Organização de problemas de <i>Cross-docking</i> . . . . .	31
Figura 5 – Uma solução para um exemplo do problema estudado. . . . .	35
Figura 6 – Horizonte de planejamento do exemplo da Figura 5 . . . . .	36
Figura 7 – Ilustração das janelas de tempo, flexível e rígida. . . . .	43
Figura 8 – Ilustração do resultado de uma instância pequena a partir do modelo de Javanmard, Vahdani e Tavakkoli-Moghaddam (2014). . . . .	54
Figura 9 – Ilustração do resultado de uma instância pequena a partir do modelo proposto MPJT. . . . .	54
Figura 10 – Dados do exemplo para a ordenação . . . . .	72
Figura 11 – Exemplo para a ordenação $O_{cr}$ . . . . .	72
Figura 12 – Exemplo para a ordenação $O_{fo}$ . . . . .	73
Figura 13 – Gráfico <b>solução x tempo</b> para o <i>kernel</i> inicial de problemas de pequena dimensão. . . . .	74



# LISTA DE ALGORITMOS

---

---

Algoritmo 1 – <i>Kernel search</i> adaptado de Angelelli, Mansini e Speranza (2010). . . . .	71
--	----



# LISTA DE TABELAS

---

---

Tabela 1 – Resumo dos artigos. . . . .	35
Tabela 2 – Grupos de instâncias gerados e quantidade de variáveis correspondentes. . .	49
Tabela 3 – Resultados dos modelos para mesma proporção de entregas e coletas. . . . .	50
Tabela 4 – Resultados dos modelos para muitas entregas e poucas coletas. . . . .	51
Tabela 5 – Resultados dos modelos para poucas entregas e muitas coletas . . . . .	52
Tabela 6 – Resultados de instâncias para calibração do CPLEX, com tempo limite de 30min. . . . .	55
Tabela 7 – Melhores soluções encontradas para as instâncias-teste de calibração do CPLEX com 10min com os parâmetros padrão do software. . . . .	55
Tabela 8 – Soluções encontradas para as instâncias-teste de calibração após calibração automática do CPLEX com 10min de execução. . . . .	56
Tabela 9 – Soluções das instâncias-teste de calibração para o parâmetro de ênfase com 10min de execução. . . . .	56
Tabela 10 – Soluções das instâncias-teste de calibração para o parâmetro <i>LP method</i> com 10min. . . . .	58
Tabela 11 – Soluções para as instâncias-teste de calibração para o parâmetro de seleção de variáveis com 10min de execução. . . . .	59
Tabela 12 – Soluções das instâncias-teste de calibração para a heurística LB com 10min de execução. . . . .	60
Tabela 13 – Soluções das instâncias-teste de calibração para a heurística <i>RINS</i> com 10min de execução. . . . .	60
Tabela 14 – Soluções das instâncias-teste de calibração para a heurística FP ( <i>Feasibility Pump</i> ) com 10min de execução. . . . .	61
Tabela 15 – Soluções das instâncias-teste de calibração para todos os cortes ao mesmo tempo com 10min de execução. . . . .	61
Tabela 16 – Resultados da calibração das instâncias-teste dos cortes por cliques com 10min de execução. . . . .	62
Tabela 17 – Resultados das instâncias-teste de calibração para o parâmetro de cortes <i>covers</i> com 10min de execução. . . . .	62
Tabela 18 – Resultados das instâncias-teste de calibracção para o parâmetro de cortes <i>flow covers</i> com 10min de execução. . . . .	63
Tabela 19 – Resultados das instâncias-teste de calibração para o parâmetro de cortes <i>flow path</i> com 10min de execução. . . . .	63

Tabela 20 – Comparação entre os resultados da calibração automática, das calibrações manuais e o padrão do software com 10min de execução. . . . .	64
Tabela 21 – Resultados após a calibração do modelo proposto para instâncias com mesma proporção de entregas e coletas com 10min de execução. . . . .	65
Tabela 22 – Resultados após a calibração do modelo proposto para instâncias com muitas entregas e poucas coletas. . . . .	66
Tabela 23 – Resultados após a calibração do modelo proposto para instâncias com poucas entregas e muitas coletas. . . . .	67
Tabela 24 – Resultados dos testes da combinação da estratégia $S_{xy}$ com as estratégias $O_{cr}$ e $O_{fo}$ para o <i>kernel search</i> com até 10min de execução. . . . .	78
Tabela 25 – Resultados dos testes da combinação da estratégia $S_x$ com as estratégias $O_{cr}$ e $O_{fo}$ para o <i>kernel search</i> com até 10min de execução. . . . .	79
Tabela 26 – Resultados dos testes da combinação da estratégia $S_y$ com as estratégias $O_{cr}$ e $O_{fo}$ para o <i>kernel search</i> com até 10min de execução. . . . .	79
Tabela 27 – Resultados obtidos pelo <i>kernel search</i> para a combinação das estratégias $KS_{xy,cr}$ e $KS_{xy,fo}$ à estratégia de adaptação $A_k$ com tempo máximo de execução do método de 10min. . . . .	80
Tabela 28 – Resultados obtidos pelo <i>kernel search</i> para a combinação das estratégias $KS_{x,cr}$ e $KS_{x,fo}$ à estratégia de adaptação $A_k$ com tempo máximo de execução do método de 10min. . . . .	81
Tabela 29 – Resultados obtidos pelo <i>kernel search</i> para a combinação das estratégias $KS_{y,cr}$ e $KS_{y,fo}$ à estratégia de adaptação $A_k$ com tempo máximo de execução do método de 10min. . . . .	81
Tabela 30 – Resultados obtidos pelo <i>kernel search</i> para as configurações da Fase 1 para a seleção $S_{xy}$ combinadas à estratégia de adaptação $A_b$ . . . . .	82
Tabela 31 – Resultados obtidos pelo <i>kernel search</i> para as configurações da Fase 1 para a seleção $S_x$ combinadas à estratégia de adaptação $A_b$ . . . . .	82
Tabela 32 – Resultados obtidos pelo <i>kernel search</i> para as configurações da Fase 1 para a seleção $S_y$ combinadas à estratégia de adaptação $A_b$ . . . . .	83
Tabela 33 – Resultados obtidos pela combinação $KS_{xy,cr,k}$ sem e com a calibração manual definida na Seção 4.3. . . . .	83
Tabela 34 – Resultados obtidos pela combinação $KS_{x,fo,k}$ sem e com a calibração manual definida na Seção 4.3. . . . .	84
Tabela 35 – Resultados obtidos pela combinação $KS_{y,cr,b}$ sem e com a calibração manual definida na Seção 4.3. . . . .	84
Tabela 36 – Resultados obtidos para 15 instâncias de calibração utilizando as 3 melhores estratégias para o <i>kernel search</i> com no máximo 10min de execução. . . . .	85
Tabela 37 – Resultados do <i>kernel search</i> com tempo limite de 10min e de 30min para o Cenário 1 (mesma proporção de entregas e coletas). . . . .	87

Tabela 38 – Resultados do <i>kernel search</i> com tempo limite de 10min e de 30min para o Cenário 2 (muitas entregas e poucas coletas). . . . .	88
Tabela 39 – Resultados do <i>kernel search</i> com tempo limite de 10min e de 30min para o Cenário 3 (poucas entregas e muitas coletas). . . . .	89





# SUMÁRIO

---

---

1	INTRODUÇÃO . . . . .	25
2	DEFINIÇÃO DO PROBLEMA E REVISÃO BIBLIOGRÁFICA . . . . .	29
2.1	Problema de integração do planejamento e da programação de uma rede de distribuição . . . . .	32
2.2	Considerações . . . . .	35
3	MODELAGEM . . . . .	39
3.1	Modelo da literatura . . . . .	40
3.2	Modelo proposto . . . . .	42
4	ANÁLISE DOS MODELOS . . . . .	47
4.1	Geração de instâncias . . . . .	47
4.2	Comparação entre os modelos . . . . .	48
4.3	Calibração do CPLEX para o MPJT . . . . .	53
4.4	Conclusões da calibração . . . . .	64
5	HEURÍSTICA PROPOSTA . . . . .	69
5.1	<i>Kernel search</i> . . . . .	69
5.2	<i>Kernel search</i> para o problema estudado . . . . .	70
5.2.1	<i>Critério de ordenação das variáveis</i> . . . . .	71
5.2.2	<i>Seleção de variáveis para a construção do kernel inicial e dos buckets</i> . . . . .	73
5.2.3	<i>Melhorias propostas</i> . . . . .	76
6	ANÁLISE DA HEURÍSTICA . . . . .	77
6.1	Parametrização . . . . .	78
6.2	Análise dos cenários . . . . .	85
7	CONCLUSÕES E PESQUISAS FUTURAS . . . . .	91
	REFERÊNCIAS . . . . .	93



---

## INTRODUÇÃO

---

A estratégia de distribuição de produtos é uma importante atividade para o gerenciamento eficiente de uma cadeia de suprimentos. Sua utilização contribui para a redução de custos operacionais e busca garantir que as operações internas da rede de distribuição sejam realizadas da forma mais eficiente, como exemplo garantir que mercadorias sejam entregues nos prazos estabelecidos.

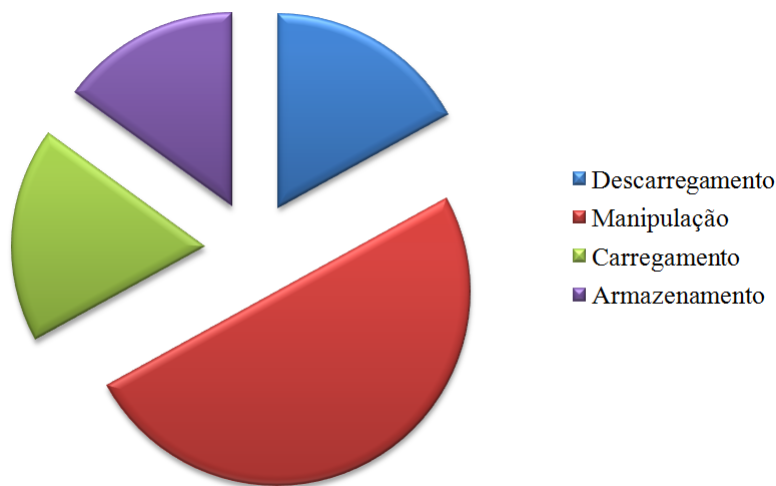
As estratégias mais comuns para atividades de distribuição de produtos, segundo [Buijs, Vis e Carlo \(2014\)](#), são: entrega direta ou por rotas de entrega, entrega utilizando-se armazéns (*warehousing*), e entrega utilizando-se docas cruzadas (*cross-docking*). A escolha adequada da estratégia a utilizar garante redução dos custos e o melhor gerenciamento da rede.

As estratégias de entrega direta, em que produtos são enviados diretamente da origem ao destino, e de rotas de entrega, em que existem grupos de rotas conectando origens e destinos, possuem o benefício de ter custos fixos baixos, uma vez que não dependem de instalações logísticas intermediárias. Porém, se as cargas forem pequenas e/ou os destinos de entrega (coleta) estiverem geograficamente dispersos, tais tipos de entrega (coleta) podem se tornar dispendiosos, uma vez que os veículos tendem a operar parcialmente vazios. Para tentar reduzir os custos intermediários nestas situações, as empresas podem utilizar armazéns (*warehousing*) ou docas de distribuição (*cross-docks*).

A utilização de armazéns permite a consolidação de cargas de veículos, ou seja, as cargas são recebidas e estocadas num armazém (ou centro de distribuição) em que os produtos são desembarcados e armazenados para que posteriormente sejam enviados a seus destinos. Desta forma, tanto as cargas de entrega quanto de envio podem ser organizadas para obter um melhor aproveitamento dos meios de transporte. [Buijs, Vis e Carlo \(2014\)](#) destacam que as principais operações em um armazém são: o descarregamento das cargas, o armazenamento dos produtos, a localização dos produtos armazenados, e a elaboração das cargas de entrega (carregamento). Por um lado, estas operações permitem organizar as cargas em busca da redução de custos de

transporte, por outro, resultam em custos adicionais de estoque e de manipulação de mercadorias. Segundo Ghiani, Laporte e Musmanno (2004), a manipulação de mercadorias representa a maior porcentagem dos custos em um armazém, cerca de 50%, enquanto o descarregamento (recebimento) representa 17%, o armazenamento 15% e o carregamento 18%, como ilustra a Figura 1.

Figura 1 – Percentual de cada operação em um armazém.



Fonte: adaptado de Ghiani, Laporte e Musmanno (2004).

*Cross-docking* é uma estratégia de distribuição em que os produtos recebidos não permanecem em estoque nas estações de distribuição por longos períodos de tempo, o que tipicamente ocorre nos armazéns. Logo, uma das vantagens do *cross-docking* é a redução dos custos de armazenamento e de manipulação de produtos nos centros de distribuição, o que leva a uma redução nos custos operacionais de distribuição, em comparação aos armazéns. Em resumo, na estratégia de *cross-docking* não se trabalha com estoque, apenas admitem-se estoques temporários que, no entanto, possuem um custo de armazenagem. Terminais dedicados às operações de *cross-docking* são chamados *cross-docks* (Belle, Valckenaers e Cattrysse (2012)) e são compostos por docas de descarregamento (em que entregas são feitas) e docas de carregamento (em que coletas são captadas). Devido a sua relevância econômica, vários autores têm estudado estratégias para o projeto e gerenciamento das operações de *cross-docking*. Revisões recentes sobre o tema foram apresentadas em Buijs, Vis e Carlo (2014) e Belle, Valckenaers e Cattrysse (2012).

Esta dissertação de mestrado tem por objetivo estudar o problema de *cross-docking* em que as cargas a serem descarregadas (entregas) e as cargas a serem compostas/carregadas (coletas) têm janelas de tempo para suas operações. Além disso, é imposto um limite para o estoque de produtos e custos de transferência entre *cross-docks* são contabilizados. O objetivo é encontrar um plano de distribuição que minimize os custos de transporte e de estoque. A possibilidade de múltiplas docas não é tratada de forma detalhada.

A revisão bibliográfica aponta que existem lacunas na literatura com relação aos modelos,

---

desta forma foi proposto um modelo baseado no modelo de [Javanmard, Vahdani e Tavakkoli-Moghaddam \(2014\)](#), cobrindo as lacunas identificadas e que gerou o artigo [Miyazaki-Tenório, Toledo e Armentano \(2015\)](#). Dados os bons resultados obtidos foi realizado o estudo da calibração dos parâmetros do software CPLEX, que retornou resultados de melhor qualidade. Com a falta de instâncias disponíveis na literatura para o problema foi criado um conjunto de instâncias (*benchmark*), disponível à comunidade científica em [Miyazaki-Tenório e Toledo \(2016\)](#). Como este é um problema NP-difícil ([Lim et al. \(2005\)](#), [Chen et al. \(2006\)](#), [Ma et al. \(2011\)](#), [Miao et al. \(2012\)](#)) um método heurístico foi proposto para sua resolução. Os testes computacionais indicaram que a heurística apresenta resultados competitivos com o software utilizado, principalmente para cenários em que há diferentes proporções de entregas e coletas no problema.

O texto está estruturado como segue. No Capítulo 2, o problema estudado é descrito e há uma breve revisão bibliográfica no contexto de *cross-docking*, assim como a ideia principal para o desenvolvimento do modelo proposto nesse trabalho. No Capítulo 3, um modelo matemático para representação do problema estudado é proposto. O Capítulo 4 é destinado a testes computacionais comparando modelos e de calibração do *software* comercial utilizado, buscando melhoria das soluções. A proposta de uma heurística é apresentada no Capítulo 5 e os resultados computacionais acerca da heurística são apresentados e discutidos no Capítulo 6. Por fim, no Capítulo 7, são apresentadas as conclusões do trabalho desenvolvido e perspectivas de trabalhos futuros.

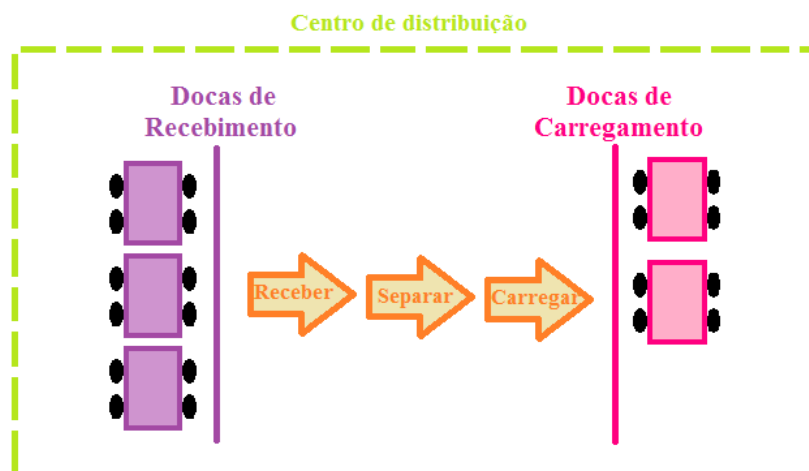


## DEFINIÇÃO DO PROBLEMA E REVISÃO BIBLIOGRÁFICA

A estratégia de *cross-docking* consiste em um sistema de distribuição intermediário entre fornecedores e consumidores. A ideia central é transferir cargas através de uma instalação logística intermediária da forma mais direta possível, ou seja, sem armazenagem entre os processos (Belle, Valckenaers e Cattrysse (2012)), ou buscando manter um estoque pelo menor tempo possível. O objetivo é efetuar o processo de entrega, organização e coleta de produtos com custo mínimo de transporte e armazenagem.

A Figura 2 ilustra um centro de distribuição composto por um único *cross-dock*, com docas de recebimento em que os entregadores devem descarregar seus produtos e docas de carregamento destinadas aos captadores. Na ilustração, há três entregadores e dois captadores efetuando as operações destacadas (recebimento/d Descarregamento da carga dos entregadores, separação das cargas para compor o carregamento das coletas realizadas pelos captadores).

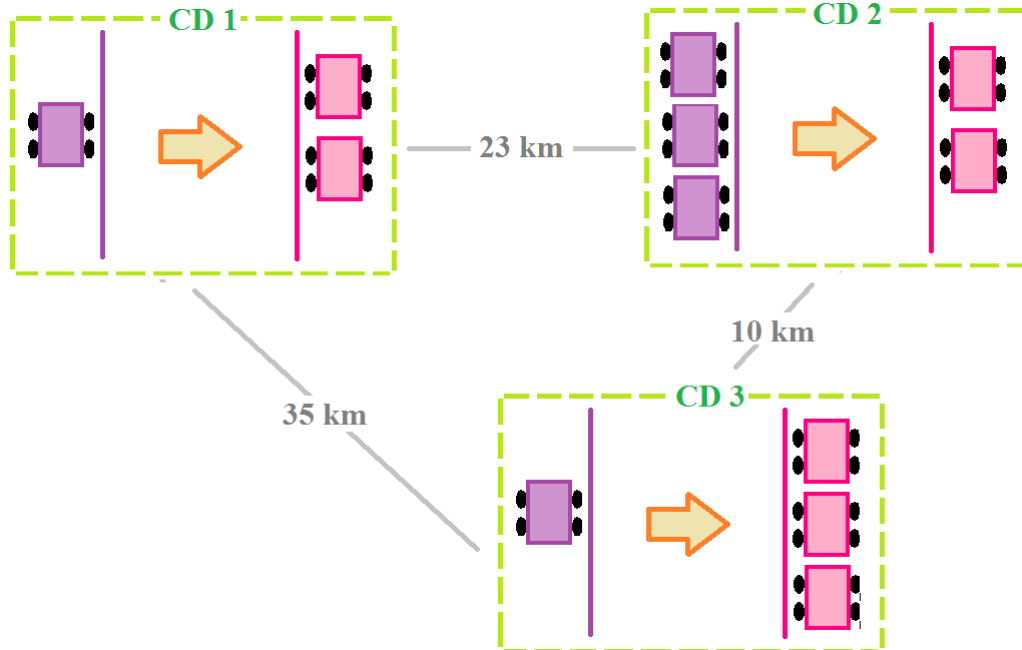
Figura 2 – *Cross-docking* com *cross-dock* único.



Fonte: Adaptado de Yu e Egbebu (2008).

A Figura 3 ilustra um centro de distribuição composto por três *cross-docks*, em que as distâncias entre os *cross-docks* são destacadas. Para o problema abordado é importante citar que as distâncias entre os *cross-docks* não podem ser muito grandes, uma vez que se considera que transferências entre os *cross-docks* podem ocorrer em cerca de uma hora.

Figura 3 – Centro de distribuição (*cross-docking*) com três *cross-docks*.



Fonte: própria.

Em geral, os centros de distribuição são compostos por um ou mais *cross-docks* que contém várias docas de recebimento e várias docas de carregamento. Como dito no Capítulo 1, a operação de armazenagem é uma das mais caras dentro de uma estratégia de distribuição, o que torna importante o estudo da estratégia de *cross-docking*. Segundo Belle, Valckenaers e Cattrysse (2012), as principais vantagens de se utilizar a estratégia de *cross-docking* com relação aos outros tipos de centros de distribuição são: redução de custos (carregamento, descarregamento, manuseio nos armazéns e armazenagem), entregas mais rápidas (muitas vezes entregas são descarregadas e os produtos são imediatamente direcionados a compor a carga das coletas), redução do espaço de armazenagem e redução do risco de perdas (no caso de produtos perecíveis), devido ao tempo de estoque limitado.

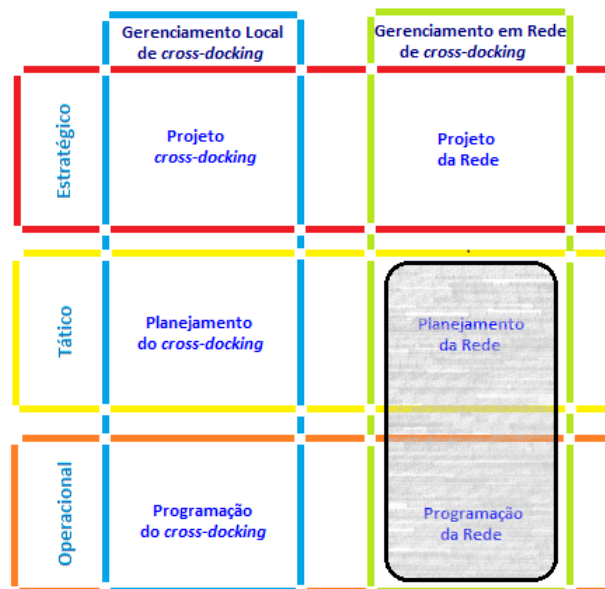
Uma das principais aplicações práticas para esta estratégia é encontrada nas redes de varejo de todo o mundo, em que os produtos chegam a um centro de distribuição de uma rede (por exemplo, a matriz) e devem ser carregados para envio a filiais ou diretamente para os consumidores. Ladier e Alpan (2015) citam o Wal-Mart como um dos pioneiros na utilização de *cross-docking*, tendo implementado a estratégia por volta de 1980. No Brasil, empresas de logística utilizam o *cross-docking* e também algumas redes de supermercado têm usado a estratégia combinada à estratégia de *warehouses*, alguns exemplos são Maroni (2013), Carraro (2008), Covisi (2016).



Segundo Buijs, Vis e Carlo (2014), na estratégia de *cross-docking*, três elementos são recorrentes: as operações básicas, os objetivos a serem atingidos; e o número de docas disponíveis. As operações básicas realizadas são: o descarregamento da carga, a classificação dos produtos com base em seu destino, a movimentação dos produtos, e o despacho imediato para coleta. Dentre os objetivos, os autores destacam: minimizar o tempo de atraso no despacho das mercadorias, minimizar o tempo de espera para entregas e coletas, e minimizar o manuseio dos produtos no interior das instalações. A restrição mais comum é quanto ao tempo limite de armazenamento dos produtos, por exemplo, 24 horas. Também o número de docas disponíveis no *cross-dock* é relevante pois, em muitos casos, é importante haver capacidade para tratar o descarregamento de várias cargas simultaneamente, o que permite que o despacho de mercadorias seja mais ágil e as cargas sejam melhor consolidadas.

O problema de *cross-docking* pode ser dividido em classes, que são: problemas locais e problemas da rede de distribuição. Problemas locais se restringem a um *cross-dock*, enquanto problemas de rede estão relacionados a uma rede de *cross-docks*. Estes problemas são subdivididos de acordo com seu nível de tomada de decisão, isto é, estratégico, tático e operacional, como ilustra a Figura 4 (ver Buijs, Vis e Carlo (2014)).

Figura 4 – Organização de problemas de *Cross-docking*.



Fonte: adaptado de Buijs, Vis e Carlo (2014).

De acordo com Buijs, Vis e Carlo (2014), no gerenciamento local de *cross-docking*, a classe de Projeto de *cross-docking* especifica as configurações do centro de distribuição como, por exemplo, determinar o número de docas a ser utilizado (planejamento estratégico). Na classe de Planejamento de *cross-docking* são definidas as operações das docas (planejamento tático). Um objetivo comum nessa classe é minimizar o manuseio de material nas docas, enquanto a classe de Programação de *cross-docking* especifica o planejamento das operações internas do

*cross-dock* no dia-a-dia (planejamento operacional).

As decisões associadas ao gerenciamento da rede de *cross-docking* são similares às do gerenciamento local de *cross-docking*. Assim, a classe de Projeto de Rede determina a infraestrutura física da rede de *cross-docking* (por exemplo, número de *cross-docks*), de forma a atender a demanda com o menor custo possível, a classe de Planejamento da Rede aloca e utiliza configurações logísticas de forma a economizar e atender às demandas, e a Programação de Rede considera restrições de tempo, respeitando as janelas de tempo para os serviços de transporte da rede.

De acordo com esta classificação, o problema abordado nessa dissertação é um problema de integração do planejamento e da programação de uma rede de distribuição, região destacada em cinza na Figura 4. O foco principal, entretanto, está no nível operacional, uma vez que se deseja programar a ordem e o instante de início das operações de entregas e coletas. Desta forma, a revisão bibliográfica apresentada se restringe aos artigos relacionados ao problema estudado. Uma revisão mais ampla sobre o tema pode ser encontrada em [Boysen e Fliedner \(2010\)](#), [Belle, Valckenaers e Cattrysse \(2012\)](#), [Buijs, Vis e Carlo \(2014\)](#) e [Ladier e Alpan \(2015\)](#) que também apresentam uma boa descrição do problema e algumas classificações para o mesmo.

## 2.1 Problema de integração do planejamento e da programação de uma rede de distribuição

O problema de integração do planejamento e da programação de uma rede de distribuição consiste em determinar a ordem e o instante para início das operações de descarregamento de entregas e carregamento de coletas em um ou mais *cross-docks* de um centro de distribuição. O foco principal é sincronizar os carregamentos e os descarregamentos de forma que a soma dos custos de distribuição e de armazenagem sejam minimizados. Como mencionado na introdução deste capítulo, a revisão aqui apresentada se restringe aos artigos relacionados a esta classe de problemas de *cross-docking*.

[Lim et al. \(2005\)](#) apresentam uma comparação entre os modelos propostos para o problema de *cross-docking* sem janelas de tempo e terminam por dar foco aos *cross-dockings* com janelas de tempo. Além disso, os autores classificaram o problema com relação a sua programação: rígida (quando os tempos são fixos), flexível (quando os tempos de entrada e saída são flexíveis) e mista (junção da programação rígida e da programação flexível).

[Chen et al. \(2006\)](#) destacam que o planejamento das operações de *cross-docking* é extremamente complexo, exigindo de fornecedores, consumidores e distribuidores um alto grau de coordenação. Em especial, a sincronização dos tempos de chegada e saída de cargas é crucial para que as operações de distribuição sejam eficazes. Sendo assim, os autores focam na política "*just-in-time*", ou seja, os produtos que chegam às docas devem ser despachados, evitando,

assim, os custos de armazenamento. Além disso, é desenvolvido um modelo matemático para o problema de *cross-docking* considerando janelas de tempo para entregadores (*deliveries*) e captadores (*pickups*, que realizam as coletas), de forma que os atrasos são penalizados (as janelas de tempo devem ser respeitadas). Os autores provaram que o problema é de difícil solução exata e desenvolveram métodos heurísticos (heurística gulosa, *Simulated Annealing*, *Tabu Search* e *Simulated Annealing com Tabu Search*) para a resolução de 40 instâncias geradas por eles. As soluções obtidas foram comparadas às melhores soluções obtidas pelo *software* ILOG CPLEX 8.0. *Tabu Search* (TS) e *Simulated Annealing com Tabu Search* (SA+TS) obtiveram, em média, as melhores soluções, uma vez que encontraram valores mais próximos da solução do *software* CPLEX em mais instâncias. Comparando-se somente TS e SA+TS, concluiu-se que SA+TS é a melhor abordagem para os problemas tratados, uma vez que SA+TS encontra mais soluções próximas das melhores soluções do *software* CPLEX que TS.

Ma *et al.* (2011) seguem uma abordagem semelhante a de Chen *et al.* (2006), porém, em sua proposta apenas um tipo de produto compõe a carga e os entregadores são, também, os captadores. Desta forma, em alguns casos, os *cross-docks* nem chegam a ser utilizados. Quando há a necessidade de *cross-docking*, os produtos são descarregados nas docas e recarregados, respeitando as janelas de tempo de entregadores e de captadores. O objetivo do problema é reduzir os custos de transporte, manuseio dos produtos dentro dos *cross-docks* e estoque. Os autores mostram que o problema é NP-completo e por isso desenvolvem duas heurísticas para sua resolução: fluxos em redes com *squeaky wheel optimization* (NF+SWO) e fluxos em redes com *genetic algorithm* (NF+GA). Foram resolvidas 24 instâncias de testes, geradas aleatoriamente pelos autores, e comparadas às soluções dadas pelo *software* CPLEX (os autores não informam a versão) em um tempo limite (2700s). Respeitando o tempo limite, o *software* CPLEX não obteve a solução ótima para nenhuma das instâncias avaliadas, os *gaps* obtidos foram de no mínimo 6,6% e no máximo 21,9%. A heurística NF+SWO resolveu as instâncias em no máximo 41,8s, e teve *gaps* entre 8,1% e 22,5%, enquanto a heurística NF+GA teve seu tempo limitado em 133,5s e *gaps* entre 8,0% e 20,1%. Assim, analisando-se os gráficos comparativos e os resultados obtidos, em geral, as heurísticas (principalmente NF+GA) tiveram melhor desempenho que o *software* CPLEX com relação ao tempo computacional, além de apresentarem soluções de qualidade similar às soluções encontradas pelo *software*.

Miao *et al.* (2012) seguem a mesma linha que Ma *et al.* (2011), considerando cargas com apenas um tipo de produto. Porém, Miao *et al.* (2012) sugerem uma nova abordagem para as janelas de tempo dos captadores, em que elas são divididas em dois tipos: janelas de tempo flexíveis (*soft time windows*) e janelas de tempo rígidas (*hard time windows*). As janelas de tempo flexíveis representam o período em que se deseja que as coletas sejam carregadas, e as janelas de tempo rígidas representam o período em que os captadores devem ser atendidos. As janelas de tempo flexíveis não precisam, necessariamente, ser atendidas. Já as janelas de tempo rígidas precisam, necessariamente, ser atendidas. Como o problema é NP-completo (Lim *et al.* (2005)), os autores desenvolvem meta-heurísticas para sua resolução. As meta-heurísticas

propostas foram: busca tabu adaptativa (BTA) e algoritmo genético adaptativo (AGA). Foram utilizados 8 grupos de instâncias de testes de pequeno porte (cada um com 4 instâncias), geradas aleatoriamente seguindo uma distribuição uniforme, e 8 grupos de instâncias de teste para problemas de grande porte, gerados da mesma forma que os problemas de pequeno porte também com 4 instâncias cada grupo. Os resultados obtidos foram comparados aos resultados obtidos utilizando o *software* ILOG CPLEX 11.0 com tempo limitado a 3600s. A BTA teve melhores tempos computacionais que os demais, e a AGA teve os melhores *gaps*; porém, para os problemas de grande escala, a BTA obteve *gaps* e tempos computacionais melhores. Desta forma, através de uma análise de sensibilidade, os autores mostram que, na maioria dos casos, a BTA tem um desempenho superior a AGA, principalmente para instâncias de grande porte. Como esperado, as duas meta-heurísticas possuem um tempo computacional muito menor que o utilizado pelo *software* CPLEX e para quase todas as instâncias testadas os *gaps* do *software* foram maiores que os encontrados pelas meta-heurísticas propostas.

Marjani, Hussein e Karimi (2012) também seguem a mesma linha de problemas que Ma *et al.* (2011), porém, permitindo o transporte de múltiplos tipos de produtos, transporte entre *cross-docks* e atraso para alguns captadores (com penalidade na função objetivo). Além destas características, os autores permitem que o estoque inicial de cada *cross-dock* seja não-nulo, e o problema é tratado como bi-objetivo, em que os objetivos são: minimizar o custo total (de manuseio, transporte dos entregadores, dos captadores, e entre *cross-docks*) e minimizar o atraso permitido. Os autores propõem três meta-heurísticas para o problema: busca em vizinhança variável (ou *variable neighborhood search* - VNS), busca tabu (TS) e *simulated annealing* híbrido (HSA). Foram geradas 24 instâncias de teste, e as meta-heurísticas foram implementadas em Matlab 7.0. Desta forma, a partir dos resultados, o algoritmo VNS mostrou melhor desempenho que os demais (menores valores de função objetivo), porém TS se mostrou mais rápida e mantendo soluções próximas de VNS.

Javanmard, Vahdani e Tavakkoli-Moghaddam (2014), uma extensão de Chen *et al.* (2006), também abordam problemas com múltiplos tipos de produtos, múltiplos *cross-docks* e janelas de tempo tanto para entregadores quanto para captadores, porém cada entrega ou coleta pode passar por até um *cross-dock* por tipo de produto. Os autores, após mostrar que o problema é NP-difícil, propõem uma heurística imperialista, *Imperialist Competitive Algorithm* (ICA), para resolução do problema. Este método é semelhante aos algoritmos genéticos e, comparando seus resultados aos resultados obtidos utilizando a linguagem GAMS 22.9 (com tempo limite de 2h), a heurística chega a soluções de boa qualidade (com *gaps* inferiores a 2%, quando comparadas as soluções ótimas do GAMS às soluções encontradas pela heurística) para as 40 instâncias geradas e encontra soluções factíveis para as instâncias maiores, para as quais o *software* GAMS não encontrou solução.

## 2.2 Considerações

Como visto no desenvolver deste capítulo, os problemas de *cross-docking* têm grande importância na área de distribuição de produtos. Na Tabela 1 é apresentado um breve resumo dos principais aspectos dos problemas abordados pelos autores anteriormente citados. Embora possa ocorrer na prática, nota-se que em nenhum dos artigos todos os aspectos (vários *cross-docks*, vários produtos, estoque limitado, janelas de tempo para entregadores e captadores (rígidas e flexíveis), podendo passar por mais de um *cross-dock* e com transporte entre *cross-docks*) são abordados simultaneamente. Identificou-se, portanto, uma lacuna na literatura que será estudada neste trabalho e destacado na última coluna da tabela.

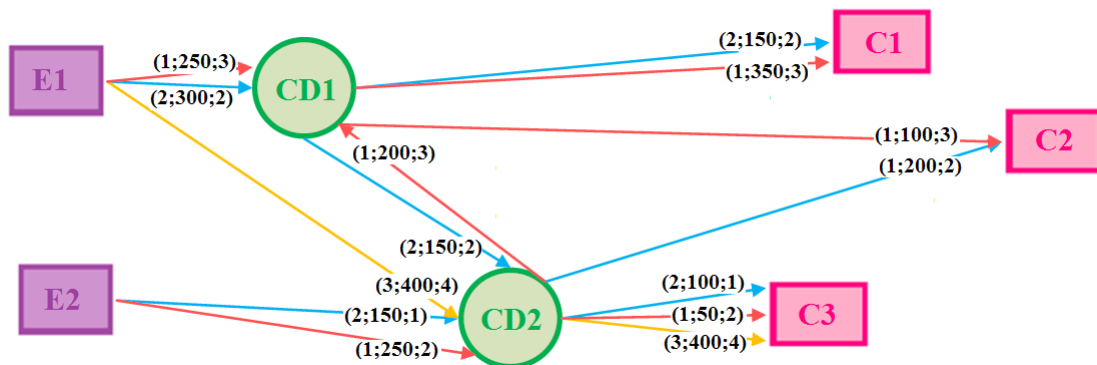
Tabela 1 – Resumo dos artigos.

Modelo	Chen et al. (2006)	Ma et al. (2011)	Miao et al. (2012)	Marjani et al. (2012)	Javanmard et al. (2014)	Miyazaki-Tenório e Toledo (2015)
<b>Cross-docks</b>	Vários	Vários	Vários	Vários	Vários	Vários
<b>Tipo de carga</b>	Heterogênea	Homogênea	Homogênea	Heterogênea	Heterogênea	Heterogênea
<b>Janelas de tempo</b>	Rígidas para entregas e coletas	Rígidas para entregas e coletas	Rígidas para entregas e rígidas e flexíveis para coletas	Rígidas para entregas e coletas	Rígidas para entregas e coletas	Rígidas para entregas e rígidas e flexíveis para coletas
<b>Pode passar por quantos cross-docks</b>	Somente um	Somente um	Somente um	Somente um	Vários (até um para cada tipo de produto)	Vários (até um para cada tipo de produto)
<b>Transporte entre cross-docks</b>	Não	Não	Não	Sim	Não	Sim
<b>Objetivo</b>	Minimizar custos de transporte e estoque	Minimizar custos de transporte e estoque	Minimizar custos de transporte, estoque e penalizações por não cumprir as janelas de tempo flexíveis das coletas	Minimizar custos de estoque, transporte e transferências e Minimizar o atraso permitido	Minimizar custos de transporte e estoque	Minimizar custos de transporte, estoque, transferências e penalizações por não cumprir as janelas de tempo flexíveis das coletas

Fonte: própria.

Uma solução para um exemplo deste problema é ilustrada na Figura 5. No exemplo são considerados 2 *cross-docks* (CD1 e CD2), 2 entregadores (E1 e E2), 3 captadores (C1, C2 e C3) e 3 tipos de produtos, ilustrados pelas setas de cores amarela, vermelha e azul. Os valores (P;Q;T) representam, respectivamente, o tipo de produto da carga, a quantidade e o instante de início da operação (entrega, coleta ou transferência), as setas indicam a direção da operação.

Figura 5 – Uma solução para um exemplo do problema estudado.



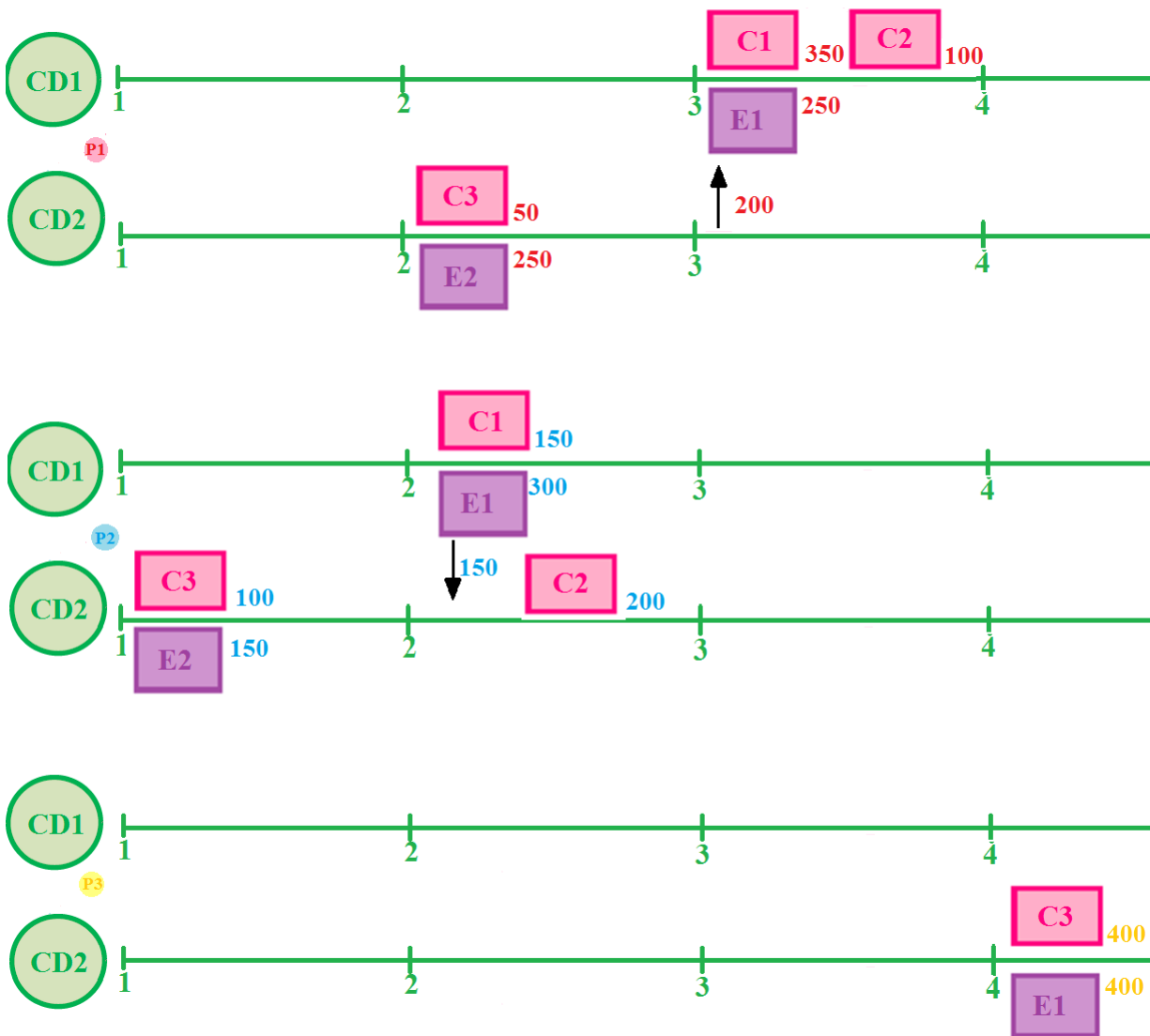
Fonte: própria.

Por hipótese, admite-se a utilização de um período de tempo para cada operação seja

ela descarregamento, transferência ou carregamento. No exemplo da Figura 5, no período 3, o *cross-dock* 1 recebe uma carga com 250 itens do produto 1 e outra carga com 200 itens do mesmo produto vinda do *cross-dock* 2.

A Figura 6 ilustra o horizonte de planejamento do exemplo exibido na Figura 5. Na figura é apresentada uma linha de tempo para cada produto, respectivamente, tipo 1 (vermelho), tipo 2 (azul) e tipo 3 (amarelo).

Figura 6 – Horizonte de planejamento do exemplo da Figura 5



Fonte: própria.

Para os produtos do tipo 1, o entregador E1 inicia sua entrega de 250 itens no instante 2 no *cross-dock* CD2 e 50 itens são direcionados a compor a carga do captador C3, deixando ao final do período 2 um estoque de 200 itens. No instante 3, os 200 itens em estoque são transferidos de CD 2 para CD1 e Cd1 recebe 250 itens de E1. Dos 450 itens recebidos 350 são direcionados a C1 e 100 para C2. O estoque de produtos do tipo 1 ao final do período 3 é nulo em ambos os *cross-docks*.

Os produtos do tipo 2 são entregues no instante 1 por E2 e 100 itens são redirecionados à carga de C3, deixando um estoque no *cross-dock* CD2 de 50 itens ao final do período 1. No instante 2, E1 inicia a entrega de 300 itens em CD1 dos quais 150 são carregados em C1, enquanto 150 são transferidos para CD2. Em CD2, 200 itens passam a compor a carga de C2. O estoque final de CD1 e de CD2 ao final do período 2 é nulo.

Por fim, os produtos do tipo 3 possuem entrega de 400 itens iniciada no instante 4 no *cross-dock* CD2 que instantaneamente passam a compor a coleta C3, deixando um estoque nulo ao final do período 4.





---

## MODELAGEM

---

O problema de *cross-docking* estudado neste trabalho é, conforme definido em [Buijs, Vis e Carlo \(2014\)](#), um problema de integração do planejamento e da programação de uma rede de distribuição. Ele consiste em determinar a ordem e o instante em que entregas são destinadas a dar início ao descarregamento e as coletas são destinadas a iniciar os carregamentos em algum dos *cross-docks* de um centro de distribuição. O foco principal é sincronizar os descarregamentos e os carregamentos de cargas de forma que a soma dos custos de distribuição e de estoque sejam minimizados. Este problema é frequente em centros de distribuição de produtos como, por exemplo, em grandes empresas de eletrodomésticos e em redes de lojas de varejo como o clássico exemplo do Wal-Mart ([Simchi-Levi, Kaminsky e Simchi-Levi \(2003\)](#) e [Miranda \(2015\)](#)). No Brasil, esta estratégia de distribuição é frequente em redes de logística como [Maroni \(2013\)](#), [Carraro \(2008\)](#), [Covisi \(2016\)](#) e no crescente *e-commerce*, como mencionado em [CARGOBR \(2013\)](#).

Neste trabalho, é abordado o problema integrado de planejamento e da programação da distribuição com as seguintes características: vários *cross-docks* e múltiplos tipos de produto (cargas heterogêneas), tanto para entregas quanto para coletas, transferência de cargas entre *cross-docks* de um mesmo centro de distribuição, entregadores e captadores podem passar por até um *cross-dock* para cada tipo de produto e janelas de tempo para entregas e coletas. Além disso, são tratados dois tipos de janelas de tempo para as coletas: janelas de tempo rígidas e janelas de tempo desejáveis (ou flexíveis). As janelas de tempo rígidas devem, obrigatoriamente, ser respeitadas; já as janelas de tempo flexíveis representam o período desejável para que cada coleta inicie a operação de carregamento.

Este mesmo problema, exceto pela transferência de cargas entre *cross-docks* de um mesmo centro de distribuição e janelas de tempo flexíveis para os consumidores, foi abordado por [Javanmard, Vahdani e Tavakkoli-Moghaddam \(2014\)](#). Desta forma, o modelo aqui proposto é uma extensão do trabalho desses autores que é descrito a seguir.

### 3.1 Modelo da literatura

O problema estudado por [Javanmard, Vahdani e Tavakkoli-Moghaddam \(2014\)](#) possui as seguintes características:

- as cargas de entregas e de coletas podem ser compostas por mais de um tipo de produto, ou seja, são cargas heterogêneas;
- todas as demandas das coletas devem ser atendidas;
- cada *cross-dock* possui capacidade limitada, ou seja, o estoque ao final de cada período de tempo não deve ultrapassar sua capacidade de armazenagem;
- o estoque inicial de cada *cross-dock* é nulo;
- entregas e coletas têm janelas de tempo que devem ser respeitadas, ou seja, os carregamentos e descarregamentos devem ser (obrigatoriamente) iniciados nos *cross-docks* dentro destas janelas;
- entregadores e captadores podem passar por mais de um *cross-dock*, mais especificamente, podem passar por até um *cross-dock* para cada tipo de produto, não sendo proibido fazer as operações de carga e descarga de mais de um tipo de produto.

Em geral, o horizonte de planejamento para problemas deste tipo não ultrapassa 24 horas e usualmente é subdividido em períodos de uma hora. Para a modelagem, considera-se que o horizonte de planejamento é discretizado e subdividido em períodos e que apenas um carregamento (descarregamento) é realizado por coleta (entrega) a cada período. Também se considera que uma vez que uma entrega seja destinada a um determinado *cross-dock*, ela deve ser inteiramente descarregada, ou seja, não se admite a entrega parcial de uma carga por um determinado entregador. O objetivo é minimizar os custos de estoque e de transporte. O modelo proposto pelos autores é dado por (3.1) - (3.9), em que:

Índices:

- $i \in \{1, 2, \dots, I\}$  - em que  $I$  é o total de tipos de produtos;
- $k \in \{1, 2, \dots, K\}$  - em que  $K$  é o total de *cross-docks* disponíveis no centro de distribuição;
- $t \in \{1, \dots, T\}$  - em que  $T$  é o total de períodos do horizonte de planejamento;
- $e \in \{1, 2, \dots, E\}$  - em que  $E$  é o total de entregadores;
- $c \in \{1, 2, \dots, C\}$  - em que  $C$  é o total de captadores.

Parâmetros:

- $HC_{ik}$  - penalidade por período por unidade do produto  $i$  em estoque no *cross-dock*  $k$ ;
- $ETC_{ik}^e$  - custo de transporte do produto  $i$  da entrega  $e$  até o *cross-dock*  $k$ ;
- $CTC_{ik}^c$  - custo de transporte do produto  $i$  da coleta  $c$  a partir do *cross-dock*  $k$ ;
- $TI^e(TF^e)$  - limitante inferior (superior) da janela de tempo da entrega  $e$ ;
- $TI^c(TF^c)$  - limitante inferior (superior) da janela de tempo da coleta  $c$ ;
- $CAP_k$  - capacidade de armazenagem do *cross-dock*  $k$ , dado em termos de itens;
- $M_i^e$  - quantidade do produto  $i$  na entrega  $e$ ;
- $M_i^c$  - demanda do produto  $i$  na coleta  $c$ .

Variáveis:

- $x_{ikt}^e = \begin{cases} 1, & \text{se o produto } i \text{ da entrega } e \text{ é descarregado no } \textit{cross-dock} k \text{ no instante } t, \\ 0, & \text{caso contrário;} \end{cases}$
- $y_{ikt}^c = \begin{cases} 1, & \text{se o produto } i \text{ da coleta } c \text{ é carregado no } \textit{cross-dock} k \text{ no instante } t, \\ 0, & \text{caso contrário;} \end{cases}$
- $s_{ikt}$  = a quantidade do produto  $i$  em estoque no *cross-dock*  $k$  ao final do período iniciado no instante  $t$ .

$$\begin{aligned} \text{Minimizar } & \sum_{i=1}^I \sum_{k=1}^K \sum_{t=1}^T HC_{ik} s_{ikt} + \sum_{i=1}^I \sum_{k=1}^K \sum_{t=1}^T \sum_{e=1}^E ETC_{ik}^e x_{ikt}^e \\ & + \sum_{i=1}^I \sum_{k=1}^K \sum_{t=1}^T \sum_{c=1}^C CTC_{ik}^c y_{ikt}^c \end{aligned} \quad (3.1)$$

Sujeito a:

$$\sum_{k=1}^K \left( \sum_{t=1}^{TF^e-1} x_{ikt}^e + \sum_{t=TF^e+1}^T x_{ikt}^e \right) = 0, \quad \forall i, e \quad (3.2)$$

$$\sum_{k=1}^K \left( \sum_{t=1}^{TF^c-1} y_{ikt}^c + \sum_{t=TF^c+1}^T y_{ikt}^c \right) = 0, \quad \forall i, c \quad (3.3)$$

$$\sum_{k=1}^K \sum_{t=TF^e}^{TF^e} x_{ikt}^e \leq 1, \quad \forall i, e \quad (3.4)$$

$$\sum_{k=1}^K \sum_{t=TF^c}^{TF^c} y_{ikt}^c = 1, \quad \forall i, c \quad (3.5)$$

$$\sum_{i=1}^I s_{ikt} \leq CAP_k, \quad \forall k, t \quad (3.6)$$

$$s_{ikt} = s_{ik,t-1} + \sum_{e=1}^E x_{ikt}^e M_i^e - \sum_{c=1}^C y_{ikt}^c M_i^c, \quad \forall i, k, t \quad (3.7)$$

$$s_{ik,0} = 0, \quad \forall i, k \quad (3.8)$$

$$s_{ikt} \in \mathbb{Z}_+; x_{ikt}^e \in \mathbb{B}; y_{ikt}^c \in \mathbb{B} \quad \forall i, k, t, e, c \quad (3.9)$$

A função objetivo (3.1) consiste em minimizar a soma dos custos de estoque de produtos nos *cross-docks*, dos custos de transporte dos produtos dos entregadores para os *cross-docks* e dos custos de transporte dos produtos dos *cross-docks* dos captadores. As restrições (3.2) determinam que um entregador não pode chegar a um *cross-dock* fora de sua janela de tempo, enquanto as restrições (3.4) garantem que se houver entrega ela terá início dentro de sua janela de tempo. De forma análoga, as restrições (3.3) asseguram que as janelas de tempo das coletas são satisfeitas. As restrições (3.5) garantem o atendimento da demanda das coletas. As restrições (3.6) impõem que a capacidade de armazenamento dos *cross-docks* é respeitada em todos os períodos. As restrições (3.7) representam o balanço de estoque de produtos nos *cross-docks*, enquanto as restrições (3.8) determinam, sem perda de generalidade, que o estoque inicial nos *cross-docks* é nulo. Por fim, as restrições (3.9) definem o domínio das variáveis.

## 3.2 Modelo proposto

Em Miao *et al.* (2012), os autores estudam dois tipos de janelas de tempo: desejável (isto é, flexível) e obrigatória (isto é, rígida). A janela de tempo desejável, definida apenas para os captadores, é o período mais apropriado para o serviço. Os autores tratam esta característica penalizando na função objetivo o serviço iniciado fora da janela de tempo desejável. Vale destacar

que as janelas de tempo obrigatórias devem ser respeitadas tanto pelos entregadores quanto pelos captadores.

No problema estudado são abordadas **janelas de tempo flexíveis (desejáveis) e rígidas para coletas** como em [Miao \*et al.\* \(2012\)](#), em que as janelas de tempo rígidas são as mesmas utilizadas anteriormente, ou seja,  $[TI^c, TF^c]$ . Já as janelas de tempo flexíveis constituem um período em que se deseja que o captador inicie atendimento no *cross-docking*, caso contrário, impõe-se uma penalidade à função objetivo. Definem-se, então, os dados:

- $PC^c$  - o custo por não atender a janela de tempo flexível da coleta  $c$ ;
- $FI^c$  - o limitante inferior da janela de tempo flexível da coleta  $c$ ;
- $FF^c$  - o limitante superior da janela de tempo flexível da coleta  $c$ .

A Figura 7 ilustra a diferença entre as janelas de tempo, ou seja, a janela de tempo flexível (em azul) está contida na janela de tempo rígida (em vinho), uma vez que a janela de tempo flexível representa o período preferível para atendimento do captador e a janela de tempo rígida representa o período em que há a obrigação de atendimento da coleta.

Figura 7 – Ilustração das janelas de tempo, flexível e rígida.



Fonte: própria.

Para adicionar a restrição desejada, define-se, também, a variável binária:

- $\delta^c = \begin{cases} 1, & \text{se a janela de tempo desejável da coleta } c \text{ é atendida,} \\ 0, & \text{caso contrário;} \end{cases}$

Assim, novas restrições para determinar se as janelas de tempo desejáveis são ou não atendidas são adicionadas ao problema:

$$\sum_{k=1}^K \sum_{t=FI^c}^{FF^c} y_{ikt}^c \leq \delta^c, \quad \forall i, c$$

$$\delta^c \in \mathbb{B},$$

e adiciona-se à função objetivo uma **penalidade** caso o consumidor seja atendido fora da janela de tempo flexível, logo a nova função objetivo é dada por:

$$\begin{aligned} & \sum_{i=1}^I \sum_{k=1}^K \sum_{t=1}^T HC_{ik} s_{ikt} + \sum_{i=1}^I \sum_{k=1}^K \sum_{t=1}^T \sum_{e=1}^E FTC_{ik}^e x_{ikt}^e \\ & + \sum_{i=1}^I \sum_{k=1}^K \sum_{t=1}^T \sum_{c=1}^C CTC_{ik}^c y_{ikt}^c + \sum_{i=1}^I \sum_{c=1}^C PC^c (1 - \delta_i^c). \end{aligned}$$

Deseja-se agora, permitir o **transporte de produtos entre *cross-docks***, como em [Marjani, Husseini e Karimi \(2012\)](#). Assim, é necessário conhecer  $TC_{ikk'}$  - o custo de transportar uma unidade de um produto de tipo  $i$  do *cross-dock*  $k$  para o *cross-dock*  $k'$ . Definimos, também, as variáveis:

- $z_{kk't} = \begin{cases} 1, & \text{se há transferência de produtos do } \textit{cross-dock} \ k \text{ para } k' \text{ no período } t, \\ 0, & \text{caso contrário;} \end{cases}$
- $w_{ikk't}$  = a quantidade transferida do produto  $i$  do *cross-dock*  $k$  para  $k'$  no período  $t$ .

As restrições (3.7), de atualização de estoque, são, então, substituídas por:

$$s_{ikt} = s_{ik,t-1} + \sum_{e=1}^E x_{ikt}^e M_i^e - \sum_{c=1}^C y_{ikt}^c M_i^c + \sum_{k'=1, k' \neq k}^K w_{k'kt}^i - \sum_{k'=1, k' \neq k}^K w_{kk't}^i, \quad \forall i, k, t,$$

em que a parcela  $\sum_{k'=1, k' \neq k}^K w_{k'kt}^i$  representa os produtos que chegaram ao *cross-dock*  $k$  transferidos de outros *cross-docks* e  $\sum_{k'=1, k' \neq k}^K w_{kk't}^i$  representa os produtos que foram transferidos de  $k$  para outros *cross-docks* no instante  $t$ . Além disso, são adicionadas as restrições:

$$\sum_{i=1}^I w_{ikk't} \leq \min \{CAP_k, CAP_{k'}\} z_{kk't}, \quad \forall k, k' \neq k, t,$$

que validam as variáveis  $z_{kk't}$ , responsáveis pelo custo adicionado à função objetivo quando há transferência de produtos entre *cross-docks*.

Como não é permitido estoque negativo e como são dependentes somente de variáveis inteiras, o domínio das variáveis  $w_{kk't}^i$  é dado por:

$$w_{kk't}^i \in \mathbb{R}_+, \quad \forall i, k, k', t.$$

O modelo proposto é, então, dado por (3.10)-(3.16).

$$\begin{aligned} \text{Minimizar } & \sum_{t=1}^T \sum_{k=1}^K \left( \sum_{i=1}^I HC_{ik} s_{ikt} + \sum_{e=1}^E \sum_{i=1}^I FTC_{ik}^e x_{ikt}^e + \sum_{c=1}^C \sum_{i=1}^I CTC_{ik}^c y_{ikt}^c \right) \\ & + \sum_{c=1}^C PC^c (1 - \delta^c) + \sum_{k=1}^K \sum_{k'=1, k' \neq k}^K \sum_{i=1}^I TC_{ikk'} \sum_{t=1}^T z_{kk't} \end{aligned} \quad (3.10)$$

Sujeito a:

$$(3.2) - (3.6)$$

$$\begin{aligned} s_{ikt} = & s_{ik,t-1} + \sum_{e=1}^E x_{ikt}^e M_i^e - \sum_{c=1}^C y_{ikt}^c M_i^c \\ & + \sum_{k'=1, k' \neq k}^K w_{ik'kt} - \sum_{k'=1, k' \neq k}^K w_{ikk't}, \end{aligned} \quad \forall i, k, t \quad (3.11)$$

$$\sum_{k=1}^K \sum_{t=FT^c}^{FF^c} y_{ikt}^c \leq \delta^c, \quad \forall i, c \quad (3.12)$$

$$\sum_{i=1}^I w_{ikk't} \leq \min \{CAP_k, CAP_{k'}\} z_{kk't}, \quad \forall k, k' \neq k, t, \quad (3.13)$$

$$s_{ik,0} = 0, \quad \forall i, k \quad (3.14)$$

$$x_{ikt}^e, y_{ikt}^c, z_{kk't}, \delta^c \in \mathbb{B}; s_{ikt} \in \mathbb{Z}_+ \quad \forall i, k, t, c, e \quad (3.15)$$

$$w_{ikk't} \in \mathbb{R}_+, \quad \forall i, k, k', t. \quad (3.16)$$

A função objetivo (3.10) visa minimizar a soma dos custos de estoque de produtos nos *cross-docks*, dos custos de transporte das entregas e das coletas, das penalidades caso as janelas de tempo desejáveis das coletas não sejam atendidas e custos de transferência de produtos entre *cross-docks*.

As equações (3.11) representam o balanço de estoque de produtos nos *cross-docks*, incluindo as quantidades de produtos transferidas para outros *cross-docks* e recebidas de outros *cross-docks*. Essas restrições foram adaptadas do modelo de [Javanmard, Vahdani e Tavakkoli-Moghaddam \(2014\)](#) para permitir a transferência de produtos entre *cross-docks*. Nas restrições (3.12), cada uma das variáveis  $\delta^c$  assume o valor um se a janela de tempo flexível a ela associada for atendida. Essas restrições foram adaptadas de [Miao et al. \(2012\)](#). Nas restrições (3.13), cada uma das variáveis  $z_{kk't}$  assume o valor um se existir transferência do *cross-dock*  $k$  para o  $k'$  no tempo  $t$ . Sem perda de generalidade, as restrições (3.14) impõem que o estoque inicial dos produtos nos *cross-docks* é nulo. Finalmente, as restrições (3.15) - (3.16) definem o domínio das variáveis do modelo.





---

## ANÁLISE DOS MODELOS

---

Neste capítulo são apresentados resultados computacionais referentes aos modelos descritos no Capítulo 3. Os testes computacionais foram realizados utilizando-se o software comercial ILOG CPLEX 12.6 em um computador Intel(R) Core(TM) Xeon (2.00GHz), com 32 GB de memória RAM, 24 núcleos e sistema operacional Linux. Além de resultados utilizando-se a configuração padrão do software, também são reportados resultados obtidos a partir do estudo da calibração de seus parâmetros. Como na literatura não há *benchmark* de instâncias disponível para o problema estudado, foram geradas instâncias como descrito na Seção 4.1.

### 4.1 Geração de instâncias

O conjunto de instâncias gerado para os testes aqui apresentados está disponível em [Miyazaki-Tenório e Toledo \(2016\)](#).

Para avaliar a influência do número de entregas e de coletas, decidiu-se gerar três cenários:

Cenário 1. mesma proporção de entregas e coletas;

Cenário 2. muitas entregas e poucas coletas;

Cenário 3. poucas entregas e muitas coletas.

Com base nos dados de [Chen et al. \(2006\)](#) e seguindo uma distribuição uniforme ( $U[a,b]$ ), foram gerados os parâmetros:

- demanda de produtos por entrega:  $U[200,500]$  itens;
- custo unitário de estoque:  $U[2,10]$  por item por período de tempo;
- custo de transporte de uma carga (tanto para entregas quanto para coletas):  $U[1000,2000]$ ;
- capacidade de estoque dos *cross-docks*:  $U[600,1500]$  itens.

A demanda das coletas foi gerada a partir dos valores das entregas, isto é, entre 50% e 95% das entregas e possuem  $U[200,500]$  itens. Além disso, como não há custo de transferência entre *cross-docks* em [Chen et al. \(2006\)](#), a geração destes custos foi baseada em uma visita técnica realizada durante a pesquisa. Assim, assumiu-se que o custo de transporte interno (entre *cross-docks*) é menor que o custo de transporte externo (entre a origem e *cross-dock* e entre *cross-dock* e destino), ou seja, o custo de transferência é gerado uniformemente entre  $U[500,1000]$ .

Considera-se um horizonte de planejamento com 24 períodos em que cada período equivale a 1 hora. A geração das janelas de tempo fixas e flexíveis foi adaptada de [Miao et al. \(2012\)](#). Assim, o início da janela de tempo da entrega  $e$  está entre  $U[1,3]$  e que o fim está entre  $U[3+P^e,22]$ , em que  $P^e$  é a quantidade de tipos de produtos da entrega  $e$ . Para as coletas, o início da janela de tempo fixa ( $TI^c$ ) está entre  $U[2,6]$  e o fim ( $TF^c$ ) está entre  $U[21,24]$ . Já o início da janela de tempo flexível da coleta  $c$  está entre  $U[TI^c+2, TI^c+3]$  e o fim está entre  $U[TF^c-3, TF^c-2]$ .

A quantidade de coletas (C), quantidade de entregas (E), número de *cross-docks* (K) e número de tipos de produtos (I) foram fixados, e as combinações utilizadas são ilustradas na Tabela 2. Cada linha da tabela representa um grupo de instâncias, para o qual foram geradas 5 instâncias-teste. Desta forma, há 40 instâncias para cada Cenário, totalizando 120 instâncias-teste geradas. As duas últimas colunas da tabela representam, respectivamente, a quantidade de variáveis (#var) e a quantidade de restrições (#rest), a menos das restrições do domínio das variáveis, no modelo proposto MPJT.

## 4.2 Comparação entre os modelos

As instâncias de teste geradas foram resolvidas utilizando-se o software comercial CPLEX com sua configuração padrão. A fim de avaliar a redução de custos que pode ser obtida ao se permitir transferências entre *cross-docks*, as instâncias-teste foram resolvidas pelo modelo de [Javanmard, Vahdani e Tavakkoli-Moghaddam \(2014\)](#), em que transferências entre *cross-docks* não são permitidas, e pelo modelo proposto, primeiramente, sem considerar as restrições referentes às janelas de tempo flexíveis e em seguida considerando essa possibilidade.

As Tabelas 3 - 5 apresentam os resultados para os três modelos, sendo: JVTM o modelo da literatura, MP o modelo proposto sem janelas de tempo flexíveis e com a possibilidade de transferência de produtos entre *cross-docks* e MPJT o modelo proposto com transferência de produtos entre *cross-docks* e janelas de tempo flexíveis para as coletas. As instâncias são representadas por  $(t,k,i,e,c)$ , com  $t$  o tamanho do horizonte de planejamento (em horas),  $k$  o número de *cross-docks*,  $i$  o número de tipos de produtos,  $e$  o número de entregas disponíveis e  $c$  o número de coletas a realizar.

Na Tabela 3, são apresentados os resultados para o Cenário 1, enquanto as Tabelas 4 e 5 exibem os resultados para os Cenários 2 e 3, respectivamente. Para cada modelo é apresentado

Tabela 2 – Grupos de instâncias gerados e quantidade de variáveis correspondentes.

	I	K	E	C	#var	#rest
<b>Cenário 1</b>	5	3	10	10	8434	841
	5	3	20	20	15644	1091
	5	4	10	10	11818	1134
	5	4	20	20	21428	1384
	10	3	10	10	16714	1466
	10	3	20	20	31124	1966
	10	4	10	10	23338	1884
	10	4	20	20	42548	2384
<b>Cenário 2</b>	5	3	5	10	6634	791
	5	3	5	20	10244	941
	5	4	5	10	9418	1084
	5	4	5	20	14228	1234
	10	3	5	10	13114	1366
	10	3	5	20	20324	1666
	10	4	5	10	18538	1784
	10	4	5	20	28148	2084
<b>Cenário 3</b>	5	3	10	5	6629	766
	5	3	20	5	10229	866
	5	4	10	5	9413	1059
	5	4	20	5	14213	1159
	10	3	10	5	13109	1316
	10	3	20	5	20309	1516
	10	4	10	5	18533	1734
	10	4	20	5	28133	1934

Fonte: própria.

o valor da melhor solução encontrada (Sol) respeitando o tempo limite de 1800s, o tempo de resolução dado em segundos (Ti) e o desvio percentual desta solução para seu limite inferior (LI) fornecido pelo CPLEX e dado em porcentagem ( $GAP = \frac{(Sol-LI)}{Sol} * 100$ ). Para MP há, ainda, a diferença com relação ao modelo JVTM (D1) dado em porcentagem ( $D1 = \frac{(Sol_{MP}-Sol_{JVTM})}{Sol_{JVTM}} * 100$ ). A melhor solução e o menor *GAP* encontrados pelos modelos é destacada em negrito, *NIS* significa que nenhuma solução foi encontrada no tempo limite e, conseqüentemente, o *GAP* não é apresentado (-).

Na Tabela 3 para 32 das 40 instâncias MP teve melhores soluções que Javanmard, sendo que para 6 dessas o modelo da literatura não retornou solução inteira factível no tempo limite. O modelo MP, como esperado, resulta em soluções de menor custo, apesar de ter *GAPs* mais altos. Como se nota pela diferença *D1*, com média -4,09%, em geral, há um ganho na qualidade da solução quando se utiliza o modelo proposto MP.

Na Tabela 4 para 23 das 40 instâncias MP teve melhores soluções que Javanmard e para

Tabela 3 – Resultados dos modelos para mesma proporção de entregas e coletas.

	JVTM			MP				MPJT		
	Sol	Ti	GAP	Sol	Ti	GAP	D1	Sol	Ti	GAP
(24,3,5,10,10)_1	77421	1806	<b>39,56</b>	74669	1803	40,49	-3,55	<b>74630</b>	1804	40,46
(24,3,5,10,10)_2	50458	1802	<b>20,14</b>	48605	1805	25,99	-3,67	<b>48361</b>	1808	25,71
(24,3,5,10,10)_3	59504	1803	<b>14,09</b>	58242	1805	21,05	-2,12	<b>58138</b>	1806	20,82
(24,3,5,10,10)_4	48579	1803	<b>16,97</b>	<b>46514</b>	1804	21,70	-4,25	46981	1806	22,48
(24,3,5,10,10)_5	52126	1802	<b>19,10</b>	<b>49573</b>	1807	22,09	-4,90	51170	1809	24,59
(24,3,5,20,20)_1	120923	1807	17,74	115724	1803	8,02	-4,30	<b>114990</b>	1806	<b>7,46</b>
(24,3,5,20,20)_2	114043	1803	21,33	<b>97432</b>	1805	<b>9,07</b>	-14,57	97577	1805	9,21
(24,3,5,20,20)_3	97098	1805	18,77	<b>85652</b>	1807	<b>10,42</b>	-11,79	87385	1805	12,23
(24,3,5,20,20)_4	100129	1807	15,98	90411	1807	9,09	-9,71	<b>88998</b>	1806	<b>7,63</b>
(24,3,5,20,20)_5	84382	1803	<b>10,16</b>	<b>82577</b>	1805	10,74	-2,14	82587	1806	10,74
(24,4,5,10,10)_1	35800	1804	<b>13,19</b>	<b>34971</b>	1807	20,05	-2,32	35355	1808	19,97
(24,4,5,10,10)_2	<b>52108</b>	1804	<b>9,73</b>	54356	1805	33,72	4,31	54300	1805	33,64
(24,4,5,10,10)_3	52643	1803	<b>34,06</b>	<b>52040</b>	1806	44,33	-1,15	52064	1805	44,35
(24,4,5,10,10)_4	<b>65773</b>	1803	<b>8,01</b>	66087	1806	59,68	0,48	66433	1807	60,91
(24,4,5,10,10)_5	47520	1804	<b>8,99</b>	<b>47137</b>	1806	12,10	-0,81	47197	1807	12,22
(24,4,5,20,20)_1	78589	1803	18,00	75418	1808	17,28	-4,03	<b>75180</b>	1806	<b>17,01</b>
(24,4,5,20,20)_2	97737	1806	20,27	84182	1805	9,69	-13,87	<b>84132</b>	1807	<b>9,63</b>
(24,4,5,20,20)_3	100105	1806	18,05	95742	1808	15,45	-4,36	<b>90273</b>	1806	<b>10,33</b>
(24,4,5,20,20)_4	129797	1806	27,01	103897	1808	10,66	-19,95	<b>103488</b>	1808	<b>10,31</b>
(24,4,5,20,20)_5	125945	1804	34,40	<b>98108</b>	1807	<b>18,81</b>	-22,10	103186	1808	22,79
(24,3,10,10,10)_1	<b>98721</b>	1803	<b>24,30</b>	99754	1806	32,47	1,05	99760	1806	32,47
(24,3,10,10,10)_2	121406	1801	<b>36,80</b>	<b>116423</b>	1807	45,32	-4,10	120856	1807	47,32
(24,3,10,10,10)_3	101703	1802	<b>10,74</b>	<b>99161</b>	1804	38,91	-2,50	<b>99161</b>	1804	38,90
(24,3,10,10,10)_4	118857	1804	<b>15,81</b>	112639	1807	48,49	-5,23	<b>111984</b>	1805	48,18
(24,3,10,10,10)_5	105422	1802	18,30	<b>96320</b>	1805	<b>17,90</b>	-8,63	96760	1807	18,15
(24,3,10,20,20)_1	NIS	1802	-	<b>179707</b>	1807	<b>16,97</b>	-	180410	1807	17,30
(24,3,10,20,20)_2	NIS	1802	-	168625	1808	15,36	-	<b>166128</b>	1807	<b>14,09</b>
(24,3,10,20,20)_3	185574	1803	21,18	178667	1805	20,68	-3,72	<b>173947</b>	1803	<b>18,09</b>
(24,3,10,20,20)_4	NIS	1802	-	183710	1808	14,64	-	<b>181240</b>	1807	<b>13,48</b>
(24,3,10,20,20)_5	NIS	1802	-	160461	1805	9,20	-	<b>159472</b>	1808	<b>8,64</b>
(24,4,10,10,10)_1	<b>125955</b>	1802	<b>23,20</b>	126964	1809	33,51	0,80	127340	1807	33,70
(24,4,10,10,10)_2	<b>103271</b>	1801	<b>17,25</b>	106301	1808	45,36	2,93	106301	1808	45,36
(24,4,10,10,10)_3	<b>112736</b>	1803	<b>29,32</b>	114957	1806	47,63	1,97	115123	1809	47,70
(24,4,10,10,10)_4	<b>90194</b>	1804	<b>21,67</b>	92317	1809	38,60	2,35	92317	1810	38,60
(24,4,10,10,10)_5	140261	1805	<b>31,76</b>	137570	1806	50,90	-1,92	<b>136686</b>	1808	50,57
(24,4,10,20,20)_1	NIS	1803	-	187036	1806	<b>24,32</b>	-	193080	1808	26,69
(24,4,10,20,20)_2	150418	1802	21,20	146910	1808	21,92	-2,33	<b>144646</b>	1806	<b>20,70</b>
(24,4,10,20,20)_3	<b>158989</b>	1802	<b>17,43</b>	185495	1806	31,23	16,67	171316	1808	25,54
(24,4,10,20,20)_4	179049	1802	22,80	<b>158059</b>	1809	<b>13,82</b>	-11,72	158470	1809	14,05
(24,4,10,20,20)_5	NIS	1807	-	173861	1807	20,15	-	<b>171753</b>	1809	<b>19,17</b>
Média	99507			107157			-4,09	106729		
Mínimo	35800			34971			-22,10	35355		
Máximo	185574			187036			16,67	193080		

Fonte: própria.

Tabela 4 – Resultados dos modelos para muitas entregas e poucas coletas.

	JVTM			MP				MPJT		
	Sol	Ti	GAP	Sol	Ti	GAP	D1	Sol	Ti	GAP
(24,3,5,10,5)_1	53960	1805	<b>18,73</b>	50676	1807	24,23	-6,09	<b>50054</b>	1807	23,77
(24,3,5,10,5)_2	41079	786	<b>0,01</b>	<b>39446</b>	1805	15,38	-3,98	<b>39446</b>	1804	15,38
(24,3,5,10,5)_3	<b>43641</b>	1804	<b>6,90</b>	44016	1807	28,08	0,86	44016	1807	28,07
(24,3,5,10,5)_4	35241	1803	<b>14,40</b>	<b>34414</b>	1807	25,47	-2,35	<b>34414</b>	1805	25,47
(24,3,5,10,5)_5	32407	1804	<b>1,54</b>	<b>31586</b>	1807	28,65	-2,53	31606	1808	28,70
(24,3,5,20,5)_1	41225	1803	<b>19,56</b>	<b>40764</b>	1803	21,75	-1,12	41239	1804	22,55
(24,3,5,20,5)_2	<b>30564</b>	1803	<b>7,62</b>	31086	1805	13,60	1,71	31682	1808	15,17
(24,3,5,20,5)_3	<b>48231</b>	1802	<b>41,77</b>	48524	1805	48,94	0,61	49700	1808	50,15
(24,3,5,20,5)_4	31642	1802	<b>18,27</b>	<b>29613</b>	1805	21,85	-6,41	<b>29613</b>	1809	21,84
(24,3,5,20,5)_5	34489	1802	<b>10,05</b>	<b>34299</b>	1804	21,99	-0,55	<b>34299</b>	1805	21,99
(24,4,5,10,5)_1	35375	37	<b>0,00</b>	<b>34431</b>	1808	34,50	-2,67	<b>34431</b>	1807	34,51
(24,4,5,10,5)_2	<b>31601</b>	1806	<b>22,50</b>	31715	1807	38,46	0,36	31715	1806	38,46
(24,4,5,10,5)_3	54315	1331	<b>0,01</b>	<b>54231</b>	1805	54,69	-0,15	54342	1807	54,79
(24,4,5,10,5)_4	41609	1804	<b>13,97</b>	<b>38764</b>	1805	39,84	-6,84	39182	1807	40,48
(24,4,5,10,5)_5	<b>45897</b>	1806	<b>23,31</b>	46438	1804	44,61	1,18	46574	1806	45,21
(24,4,5,20,5)_1	<b>25063</b>	646	<b>0,01</b>	26356	1806	31,93	5,16	26421	1804	32,12
(24,4,5,20,5)_2	36922	1804	<b>16,52</b>	<b>36079</b>	1808	19,54	-2,28	<b>36079</b>	1808	19,54
(24,4,5,20,5)_3	33477	1802	<b>9,87</b>	<b>32672</b>	1807	36,76	-2,40	<b>32672</b>	1806	36,76
(24,4,5,20,5)_4	<b>31304</b>	1803	<b>3,97</b>	31822	1807	17,32	1,65	31947	1807	18,18
(24,4,5,20,5)_5	<b>20839</b>	569	<b>0,01</b>	22809	1806	41,12	9,45	24647	1806	45,58
(24,3,10,10,5)_1	62503	1690	<b>0,01</b>	<b>59240</b>	1806	18,29	-5,22	59308	1807	18,53
(24,3,10,10,5)_2	90520	1806	<b>2,34</b>	88701	1806	34,92	-2,01	<b>88649</b>	1806	34,80
(24,3,10,10,5)_3	113037	1803	<b>22,39</b>	112024	1807	61,38	-0,90	<b>111813</b>	1807	61,30
(24,3,10,10,5)_4	<b>77208</b>	1802	<b>8,19</b>	80023	1804	28,60	3,65	80197	1806	28,75
(24,3,10,10,5)_5	66602	1803	<b>2,63</b>	<b>66404</b>	1805	38,10	-0,30	<b>66404</b>	1805	38,11
(24,3,10,20,5)_1	64273	403	<b>0,01</b>	65897	1806	48,86	2,53	<b>64238</b>	1806	46,51
(24,3,10,20,5)_2	<b>68040</b>	1427	<b>0,01</b>	69630	1805	53,00	2,34	68925	1805	52,66
(24,3,10,20,5)_3	78819	1805	<b>19,43</b>	<b>78597</b>	1807	28,42	-0,28	80554	1806	30,16
(24,3,10,20,5)_4	<b>86977</b>	1804	<b>24,89</b>	93907	1806	42,16	7,97	93147	1807	41,69
(24,3,10,20,5)_5	86920	1805	<b>39,14</b>	83179	1806	45,33	-4,30	83179	1806	45,33
(24,4,10,10,5)_1	59222	1803	<b>12,92</b>	59004	1807	41,85	-0,37	<b>58883</b>	1807	41,73
(24,4,10,10,5)_2	60670	1804	<b>2,64</b>	<b>59590</b>	1802	35,41	-1,78	<b>59590</b>	1804	35,41
(24,4,10,10,5)_3	73193	1803	<b>17,24</b>	<b>70482</b>	1806	28,07	-3,70	70540	1807	28,13
(24,4,10,10,5)_4	74716	1806	<b>3,99</b>	<b>73384</b>	1806	39,38	-1,78	73988	1807	39,89
(24,4,10,10,5)_5	<b>62123</b>	1804	<b>16,41</b>	62667	1805	43,11	0,88	62992	1805	43,41
(24,4,10,20,5)_1	<b>69384</b>	1804	<b>16,49</b>	<b>69384</b>	1802	37,20	0,00	<b>69384</b>	1803	37,20
(24,4,10,20,5)_2	62674	1804	<b>12,94</b>	63151	1808	45,81	0,76	<b>62319</b>	1807	45,09
(24,4,10,20,5)_3	75280	1903	<b>28,36</b>	<b>84217</b>	1805	57,12	11,87	<b>84217</b>	1804	57,12
(24,4,10,20,5)_4	<b>60531</b>	1806	<b>16,26</b>	61983	1808	45,45	2,40	62479	1807	45,86
(24,4,10,20,5)_5	<b>57022</b>	1802	<b>17,20</b>	59154	1806	41,03	3,74	59154	1807	41,03
<b>Média</b>	54965			55009			-0,02	55101		
<b>Mínimo</b>	20839			22809			-6,84	24647		
<b>Máximo</b>	113037			112024			11,87	111813		

Fonte: própria.

Tabela 5 – Resultados dos modelos para poucas entregas e muitas coletas

	JVTM			Sol	MP			D1	MPJT		
	Sol	Ti	GAP		Ti	GAP	Sol		Ti	GAP	
(24,3,5,5,10)_1	86564	1642	<b>0,01</b>	<b>84425</b>	1806	49,20	-2,47	<b>84425</b>	1807	45,67	
(24,3,5,5,10)_2	49089	4	<b>0,00</b>	<b>48630</b>	195	0,01	-0,94	<b>48630</b>	196	0,01	
(24,3,5,5,10)_3	55845	1807	<b>13,21</b>	<b>53671</b>	1807	20,18	-3,89	54005	1806	20,63	
(24,3,5,5,10)_4	82196	47	<b>0,01</b>	<b>79571</b>	1805	4,57	-3,19	<b>79571</b>	1805	4,57	
(24,3,5,5,10)_5	40135	459	<b>0,01</b>	<b>39191</b>	1805	18,60	-2,35	39593	1807	19,36	
(24,3,5,5,20)_1	80119	1803	<b>16,45</b>	77703	1807	33,23	-3,02	<b>77368</b>	1808	39,03	
(24,3,5,5,20)_2	<b>82164</b>	1803	<b>1,34</b>	82612	1802	7,77	0,55	82200	1808	6,72	
(24,3,5,5,20)_3	89198	689	<b>0,01</b>	<b>88126</b>	1808	6,95	-1,20	88198	1809	6,96	
(24,3,5,5,20)_4	57321	1805	<b>8,53</b>	<b>54301</b>	1805	12,54	-5,27	54364	1808	12,64	
(24,3,5,5,20)_5	53681	1806	<b>11,06</b>	<b>51813</b>	1806	16,92	-3,48	<b>51813</b>	1808	16,95	
(24,4,5,5,10)_1	63123	1803	<b>21,70</b>	<b>60501</b>	1805	24,22	-4,15	60750	1807	24,53	
(24,4,5,5,10)_2	53012	1806	<b>10,63</b>	<b>48165</b>	1806	18,73	-9,14	<b>48165</b>	1807	18,73	
(24,4,5,5,10)_3	58180	1804	<b>5,88</b>	57529	1805	40,16	-1,12	<b>57476</b>	1804	40,11	
(24,4,5,5,10)_4	57372	1803	<b>16,93</b>	55596	1809	31,07	-3,10	<b>55459</b>	1805	31,16	
(24,4,5,5,10)_5	56293	1803	<b>11,68</b>	<b>55777</b>	1806	33,54	-0,92	<b>55777</b>	1806	33,54	
(24,4,5,5,20)_1	49586	1805	<b>7,33</b>	<b>49082</b>	1805	18,71	-1,02	<b>49082</b>	1806	18,71	
(24,4,5,5,20)_2	64875	1807	<b>21,84</b>	61942	1806	22,71	-4,52	<b>61807</b>	1805	22,54	
(24,4,5,5,20)_3	60634	1807	<b>13,78</b>	58403	1807	30,00	-3,68	<b>58288</b>	1808	29,87	
(24,4,5,5,20)_4	50048	1806	<b>16,35</b>	<b>49470</b>	1807	18,09	-1,15	<b>49470</b>	1807	18,09	
(24,4,5,5,20)_5	57251	1112	<b>0,01</b>	56321	1806	20,68	-1,62	<b>56218</b>	1807	20,53	
(24,3,10,5,10)_1	94136	1803	<b>18,42</b>	<b>87134</b>	1806	35,63	-7,44	<b>87134</b>	1806	35,62	
(24,3,10,5,10)_2	179658	1805	<b>5,27</b>	<b>178481</b>	1807	36,85	-0,66	<b>178481</b>	1807	36,85	
(24,3,10,5,10)_3	82053	1803	<b>2,76</b>	<b>80096</b>	1805	28,53	-2,39	80522	1808	28,68	
(24,3,10,5,10)_4	152869	1703	<b>0,01</b>	<b>146566</b>	1807	40,01	-4,12	<b>146566</b>	1808	40,02	
(24,3,10,5,10)_5	158794	1803	<b>1,64</b>	156314	1805	53,79	-1,56	<b>155840</b>	1806	54,52	
(24,3,10,5,20)_1	90962	1803	<b>17,19</b>	88419	1806	29,70	-2,80	<b>88199</b>	1805	29,52	
(24,3,10,5,20)_2	<b>100399</b>	1805	<b>11,46</b>	101856	1807	19,75	1,45	101919	1805	20,42	
(24,3,10,5,20)_3	145559	1803	<b>23,15</b>	<b>133860</b>	1808	28,89	-8,04	134525	1806	29,24	
(24,3,10,5,20)_4	116393	1802	<b>21,79</b>	110173	1805	33,84	-5,34	<b>109582</b>	1805	32,90	
(24,3,10,5,20)_5	166604	1803	<b>9,52</b>	<b>161152</b>	1806	40,37	-3,27	<b>161152</b>	1807	40,87	
(24,4,10,5,10)_1	91990	1803	<b>2,30</b>	<b>88676</b>	1806	42,28	-3,60	<b>88676</b>	1807	42,15	
(24,4,10,5,10)_2	122869	1804	<b>12,72</b>	<b>121368</b>	1807	29,77	-1,22	<b>121368</b>	1807	29,77	
(24,4,10,5,10)_3	<b>66655</b>	1804	<b>9,15</b>	66679	1808	38,93	0,04	66679	1807	38,92	
(24,4,10,5,10)_4	80272	1803	<b>0,71</b>	<b>75218</b>	1807	30,72	-6,30	75271	1807	30,77	
(24,4,10,5,10)_5	73134	1398	<b>0,01</b>	71358	1807	28,14	-2,43	<b>71191</b>	1805	27,98	
(24,4,10,5,20)_1	73536	1803	<b>15,62</b>	71896	1805	33,88	-2,23	<b>71500</b>	1807	33,54	
(24,4,10,5,20)_2	80993	1806	<b>11,50</b>	<b>77205</b>	1807	43,19	-4,68	77841	1805	49,78	
(24,4,10,5,20)_3	102150	1803	<b>3,75</b>	100583	1805	26,20	-1,53	<b>98448</b>	1804	27,74	
(24,4,10,5,20)_4	64299	1803	<b>4,79</b>	<b>62364</b>	1806	37,05	-3,01	<b>62364</b>	1807	37,06	
(24,4,10,5,20)_5	93878	1803	<b>5,32</b>	93494	1803	39,05	-0,41	<b>93235</b>	1803	38,88	
<b>Média</b>	84597			82143			-2,88	82079			
<b>Mínimo</b>	40135			39191			-9,14	39593			
<b>Máximo</b>	179658			178481			1,45	178481			

Fonte: própria.

1 instância encontrou solução de mesmo valor. As soluções de MP são de melhor qualidade com média de diferença de  $-0,02\%$  apesar dos *GAPs* superiores aos do modelo de Javanmard, Vahdani e Tavakkoli-Moghaddam (2014). E na Tabela 5 para 37 das 40 instâncias MP teve melhores soluções que Javanmard e a diferença média de  $-2,88\%$  também indica que o modelo MP retorna soluções que superam as de Javanmard.

Comparando-se os modelos propostos com e sem janela de tempo desejável para os captadores, na Tabela 3 para 19 instâncias MPJT teve melhores soluções e para 3 instâncias obteve soluções iguais. 8 soluções das apresentadas na Tabela 4 para MPJT são de melhor qualidade, enquanto que em 15 instâncias a solução encontrada pelo modelo MPJT foi de qualidade igual à encontrada por MP. Para 14 das 40 instâncias da Tabela 5 MPJT teve melhores soluções e para 16 teve soluções iguais às de MP. A diferença média de solução apresentada entre os modelos MPJT e MP é de  $-0,13\%$ ,  $0,34\%$  e  $-0,05\%$  para os cenários 1, 2 e 3, respectivamente.

Nos Cenários 1, 2 e 3 as soluções de menor custo foram obtidas pelo modelo proposto com janelas de tempo desejáveis (MPJT) e os melhores *GAPs* encontrados foram para o modelo da literatura. Os *GAPs* médios de MP e MPJT são muito próximos (menos de  $1,0\%$  de diferença), porém as soluções de MPJT são de melhor qualidade para todos os cenários. Apenas para uma das 120 instâncias dos 3 cenários ((24,3,5,5,10)<sub>2</sub>), os três modelos encontraram solução ótima.

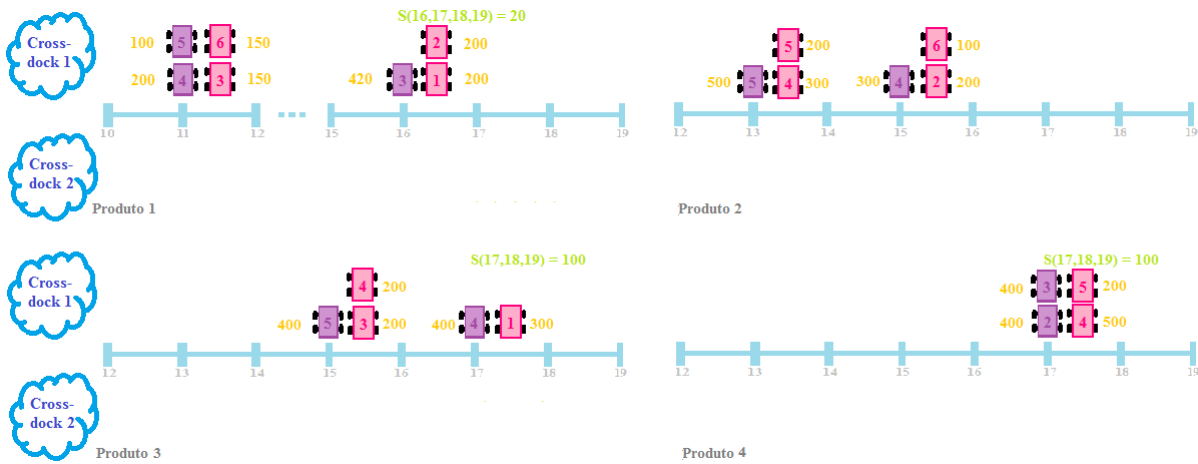
Em geral, para todas as instâncias o modelo proposto com janelas de tempo flexíveis retornou as melhores soluções, mas apresentando *GAPs* relativamente altos quando comparados ao modelo de Javanmard, Vahdani e Tavakkoli-Moghaddam (2014). O modelo da literatura apresentou os melhores *GAPs*, porém a qualidade das soluções dos modelos propostos foi melhor. Desta forma, mostramos que o modelo proposto (inserindo a transferência entre *cross-docks* e, até mesmo com as janelas de tempo desejáveis - que às vezes representam aumento no custo da função objetivo) pode auxiliar a encontrar soluções de melhores qualidades.

As Figuras 8 e 9 ilustram, respectivamente, os resultados do modelo de Javanmard, Vahdani e Tavakkoli-Moghaddam (2014) e do modelo proposto MPJT para um problema de pequena dimensão, com 2 *cross-docks*, 5 entregadores (veículos roxos), 6 captadores (veículos rosa) e 4 tipos de produto. O valor da solução encontrada pelo modelo da literatura é de 65062,70 (encontrada em 3,20s) e em MPJT é de 63207,00 (encontrada em 1,80s), ou seja, há melhoria na qualidade da solução. A Figura 9 mostra que o ganho foi, principalmente, ao se permitir a transferência de produtos do tipo 1 e do tipo 4 (representada por setas vermelhas).

### 4.3 Calibração do CPLEX para o MPJT

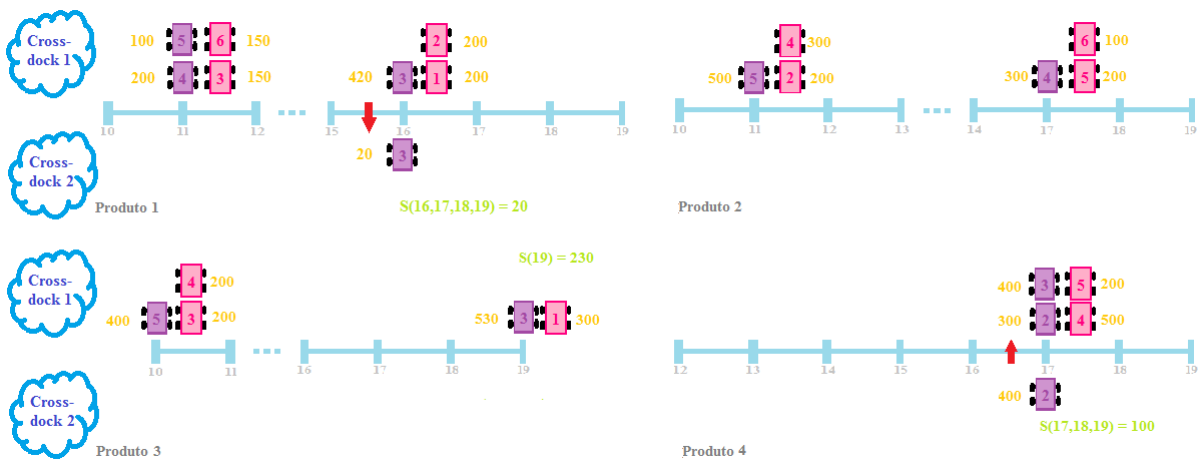
A fim de buscar melhores resultados para o modelo proposto, foi realizada a calibração dos parâmetros do software comercial ILOG CPLEX 12.6. Desta forma, buscou-se obter melhores configurações para resolver as instâncias geradas para o problema de *cross-docking*. Na

Figura 8 – Ilustração do resultado de uma instância pequena a partir do modelo de Javanmard, Vahdani e Tavakkoli-Moghaddam (2014).



Fonte: própria.

Figura 9 – Ilustração do resultado de uma instância pequena a partir do modelo proposto MPJT.



Fonte: própria.

versão utilizada do software CPLEX existe a possibilidade de realizar a calibração automática dos parâmetros. A escolha dos parâmetros testados individualmente foi baseada no conhecimento prévio do problema e com base dos parâmetros mais modificados usualmente. A definição dos parâmetros, bem como seus valores são apresentados no Manual do software.

Das 120 instâncias geradas na Seção 4.1, 15 instâncias de pequena dimensão foram selecionadas e executadas com tempo limite de execução de 1800s com o software na configuração padrão. Na seleção das 15 instâncias, foram escolhidas 5 de cada um dos 3 cenários descritos. As soluções obtidas são resumidas na Tabela 6. Para a calibração (automática e manual), foram selecionadas as 10 piores soluções (com maiores GAPS, destacadas em negrito) uma vez que para todas as instâncias não foi possível encontrar a melhor solução dentro do tempo limite estipulado.

Para a calibração do software CPLEX, fixou-se um tempo limite de 600s de execução



Tabela 6 – Resultados de instâncias para calibração do CPLEX, com tempo limite de 30min.

	<b>Sol</b>	<b>T</b>	<b>GAP</b>
<b>(24,3,5,10,10)_1</b>	74630	1804	<b>40,46</b>
<b>(24,3,5,10,10)_2</b>	48361	1808	<b>25,71</b>
<b>(24,3,5,10,10)_3</b>	58138	1806	<b>20,82</b>
<b>(24,3,5,10,10)_4</b>	46981	1806	<b>22,48</b>
<b>(24,3,5,10,10)_5</b>	51170	1809	<b>24,59</b>
<b>(24,3,5,5,20)_1</b>	77368	1808	<b>39,03</b>
<b>(24,3,5,5,20)_2</b>	82200	1808	6,72
<b>(24,3,5,5,20)_3</b>	88198	1809	6,96
<b>(24,3,5,5,20)_4</b>	54364	1808	12,64
<b>(24,3,5,5,20)_5</b>	51813	1808	16,95
<b>(24,3,5,20,5)_1</b>	41239	1804	<b>22,55</b>
<b>(24,3,5,20,5)_2</b>	31682	1808	15,17
<b>(24,3,5,20,5)_3</b>	49700	1808	<b>50,15</b>
<b>(24,3,5,20,5)_4</b>	29613	1809	<b>21,84</b>
<b>(24,3,5,20,5)_5</b>	34299	1805	<b>21,99</b>

Fonte: própria.

para cada uma das instâncias em cada configuração testada. Na Tabela 7, são apresentadas as soluções das 10 instâncias-teste selecionadas no passo anterior, com tempo limite de execução de 600s. A solução média apresentada na tabela é 52229,7 e o GAP médio é 31,06.

Tabela 7 – Melhores soluções encontradas para as instâncias-teste de calibração do CPLEX com 10min com os parâmetros padrão do software.

	<b>Sol</b>	<b>GAP</b>
I1 <b>(24,3,5,10,10)_1</b>	75691	41,40
I2 <b>(24,3,5,10,10)_2</b>	48978	26,81
I3 <b>(24,3,5,10,10)_3</b>	60527	27,24
I4 <b>(24,3,5,10,10)_4</b>	47257	23,09
I5 <b>(24,3,5,10,10)_5</b>	51170	24,75
I6 <b>(24,3,5,5,20)_1</b>	77969	39,71
I7 <b>(24,3,5,20,5)_1</b>	46712	31,94
I8 <b>(24,3,5,20,5)_3</b>	49840	50,45
I9 <b>(24,3,5,20,5)_4</b>	29749	22,71
I10 <b>(24,3,5,20,5)_5</b>	34404	22,51
Média		<b>31,06</b>

Fonte: própria.

Através da calibração automática do CPLEX, nota-se que todos os parâmetros devem ser mantidos em seus valores padrão, exceto pelo parâmetro *mircuts*, que deve assumir o valor 1 (moderado). As soluções obtidas para as 10 instâncias-teste com a configuração sugerida após a calibração automática do CPLEX são exibidas na Tabela 8. O valor médio das soluções encontradas é de 51709,4 e o GAP médio de 31,88%, ou seja, há redução no valor das soluções

e aumento de 0,82% no GAP médio.

Tabela 8 – Soluções encontradas para as instâncias-teste de calibração após calibração automática do CPLEX com 10min de execução.

	<b>Sol</b>	<b>GAP</b>
I1	77354	43,53
I2	<b>48289</b>	26,15
I3	60527	33,49
I4	<b>46908</b>	22,99
I5	<b>49706</b>	22,27
I6	<b>77608</b>	39,07
I7	<b>41668</b>	23,86
I8	<b>47225</b>	48,24
I9	31085	26,17
I10	36724	33,01
Média	<b>31,88</b>	

Fonte: própria.

Foi realizada, também, a calibração manual com tempo limite de 600s para cada instância e variando alguns parâmetros. O primeiro parâmetro analisado é a ênfase de resolução, os resultados obtidos são ilustrados na Tabela 9. As Tabelas 9-20 apresentam as soluções para cada um dos valores que o parâmetro pode assumir. A primeira coluna é o parâmetro padrão do CPLEX, e as melhores soluções e os melhores GAPs são destacados em negrito. É apresentada na última linha de cada tabela a média de soluções e as melhores médias são destacadas em negrito. O parâmetro selecionado é o que tiver melhor solução média, uma vez que buscamos soluções de melhor qualidade.

Tabela 9 – Soluções das instâncias-teste de calibração para o parâmetro de ênfase com 10min de execução.

	<b>balance (0)</b>		<b>feasibility (1)</b>		<b>optimality (2)</b>		<b>best bound (3)</b>		<b>hidden (4)</b>	
	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP
I1	75634	<b>41,35</b>	76653	45,26	80530	44,74	76223	42,33	<b>74120</b>	40,75
I2	48978	26,81	48972	28,01	50964	29,27	51109	<b>19,05</b>	<b>48294</b>	26,08
I3	58931	27,62	<b>58163</b>	34,58	60459	32,00	59491	<b>14,43</b>	58917	15,72
I4	46832	22,39	<b>46104</b>	23,97	47335	23,19	46273	<b>15,15</b>	46512	21,75
I5	52101	26,02	<b>50403</b>	22,48	51499	24,97	50662	<b>18,66</b>	50771	23,61
I6	78099	39,81	77610	40,67	78673	33,76	78119	<b>13,74</b>	<b>77451</b>	34,95
I7	44950	29,28	41548	26,27	49086	34,98	41551	<b>19,66</b>	<b>40718</b>	21,86
I8	49450	50,06	<b>47706</b>	51,11	49913	50,19	48562	<b>25,27</b>	49053	49,66
I9	<b>29749</b>	22,70	31146	30,53	29613	21,46	30000	<b>13,37</b>	29842	23,33
I10	<b>34404</b>	22,50	34946	28,36	38473	30,25	36724	<b>17,81</b>	36917	27,66
	51913		51325		53655		51871		<b>51260</b>	

Fonte: própria.

Para o parâmetro ênfase, o valor selecionado (que possui melhores soluções) é o *hidden*

*feasible solutions* (3). Com média de soluções de 51260 e GAP médio de 28,54% (redução de 2,52% quando comparadas com as soluções da Tabela 7).

Fixado o parâmetro ênfase, o próximo parâmetro verificado foi o de método de programação linear, cujas soluções são apresentadas na Tabela 10. Nota-se, então, que o valor padrão 0 – *automatic*, apresenta a melhor solução, com soluções de menor custo para 6 instâncias, além de melhores *GAPs* para 4 instâncias.

As soluções obtidas para o parâmetro de escolha de variáveis são apresentadas na Tabela 11. A seleção por pseudo-custos (*2-pseudo costs*) apresentou as melhores soluções, com médias de 50994 para a solução e 30,38% para o GAP.

Os próximos parâmetros verificados são heurísticas. As heurísticas verificadas foram: Limitante Inferior (LB), *RINS* e *Feasibility Pump* (FP). Para a heurística LB, os resultados são resumidos na Tabela 12. A solução média encontrada é de 50938 e o GAP médio é de 29,74% para o valor *yes*.

Para a heurística *RINS*, os resultados são resumidos na Tabela 13. Verifica-se que as melhores soluções mantêm o valor padrão para esse parâmetro.

Finalmente, para a heurística FP, os resultados são exibidos na Tabela 14. A melhor média de solução é 50829 para o valor -1 (*do not generate*), com GAP médio 29,64.

Também foram analisados os parâmetros de cortes e, para tal, dividimos esta verificação em duas etapas:

Etapa 1: todos os cortes ao mesmo tempo; e

Etapa 2: cada um dos cortes mais adequados ao problema individualmente (*cliques*, *covers*, *flow covers* e *flow path*).

Os resultados obtidos quando se verificam todos os cortes ao mesmo tempo (Etapa 1) são apresentados na Tabela 15. As melhores soluções obtidas são com o valor -1 (*do not generate*), com médias de 50789 para solução e 34,21% para GAP.

Na Etapa 2, foram testados, nesta ordem, os parâmetros de corte: *cliques*, *covers*, *flow covers* e *flow path*. Seus resultados são apresentados nas Tabelas 16-19.

Os resultados apresentados na Tabela 16, quando comparados com os resultados da Tabela 14, mantêm-se com a melhor média de soluções no valor padrão.

Já na Tabela 17, os melhores valores encontrados para média de solução e GAP médio são 50518 e 14,33%, respectivamente. Ou seja, há uma redução de 15,31% quando comparado ao GAP médio da Tabela 16. Como o valor *2-aggressive* para o parâmetro de cortes *covers* apresentou mais soluções de boa qualidade, este foi o valor selecionado para a calibração.

Na Tabela 18, o parâmetro *flow covers cuts* apresentou melhores resultados com o valor *1-moderate*, com solução média de 50446 e GAP médio de 12,52% (redução de 1,82% quando

Tabela 10 – Soluções das instâncias-teste de calibração para o parâmetro *LP method* com 10min.

	automatic (0)		primal (1)		dual (2)		network (3)		barrier (4)		sifting (5)		dual+barrier (6)	
	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP
I1	<b>74120</b>	<b>40,75</b>	<b>74120</b>	40,76	<b>74120</b>	<b>40,75</b>	<b>74120</b>	<b>40,75</b>	<b>74120</b>	<b>40,75</b>	<b>74120</b>	<b>40,75</b>	<b>74120</b>	40,76
I2	48294	26,08	<b>48250</b>	<b>26,01</b>	<b>48250</b>	<b>26,01</b>	49051	27,21	48294	26,08	48294	26,08	<b>48250</b>	<b>26,01</b>
I3	<b>58917</b>	<b>15,72</b>	59506	16,57	59506	16,57	59506	16,57	59506	16,57	59148	16,04	59509	16,56
I4	<b>46512</b>	<b>21,75</b>	<b>46512</b>	<b>21,75</b>	<b>46512</b>	21,76	46512	21,76	<b>46512</b>	21,75	46520	21,79	<b>46512</b>	21,76
I5	<b>50771</b>	23,61	<b>50771</b>	23,61	<b>50771</b>	23,61	<b>50771</b>	23,61	<b>50771</b>	23,61	<b>50771</b>	<b>23,60</b>	<b>50771</b>	23,61
I6	<b>77451</b>	34,95	<b>77451</b>	35,04	<b>77451</b>	34,96	<b>77451</b>	35,08	<b>77451</b>	<b>34,93</b>	<b>77451</b>	<b>35,06</b>	<b>77451</b>	34,94
I7	40718	21,86	40734	21,89	40734	21,88	40734	21,88	40734	21,88	40734	21,88	<b>40350</b>	<b>21,14</b>
I8	<b>49053</b>	<b>49,66</b>	49389	50,00	49389	50,00	<b>49053</b>	<b>49,66</b>	49053	49,66	49389	50,00	<b>49053</b>	<b>49,66</b>
I9	29842	23,33	29734	23,06	29842	23,34	29842	23,33	29842	23,33	29842	23,33	<b>29734</b>	<b>23,06</b>
I10	36917	27,66	<b>36886</b>	<b>27,60</b>	36917	27,66	36917	27,67	36917	27,66	36917	27,66	36917	27,66
	<b>51260</b>		51335		51349		51396		51320		51319		51267	

Fonte: própria.

Tabela 11 – Soluções para as instâncias-teste de calibração para o parâmetro de seleção de variáveis com 10min de execução.

	automatic (0)		minimum (-1)		maximum (1)		ps. costs (2)		strong (3)		ps.red. costs (4)	
	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP
<b>I1</b>	74120	40,75	75731	42,27	73854	41,05	<b>72280</b>	<b>39,25</b>	73524	40,20	74254	41,13
<b>I2</b>	48294	26,08	49994	28,97	48733	27,19	48942	26,97	49192	26,61	<b>48065</b>	<b>25,99</b>
<b>I3</b>	58917	<b>15,72</b>	<b>58365</b>	31,80	60054	33,86	58688	31,45	58822	30,71	60268	34,05
<b>I4</b>	46512	21,75	46918	22,96	46524	22,21	47139	23,21	47469	23,36	<b>46033</b>	<b>21,27</b>
<b>I5</b>	50771	23,61	50406	23,86	51784	25,97	49635	21,89	<b>50367</b>	<b>19,80</b>	50964	24,17
<b>I6</b>	77451	<b>34,95</b>	<b>77195</b>	38,84	78536	40,17	77763	39,29	79510	39,80	77705	38,81
<b>I7</b>	<b>40718</b>	<b>21,86</b>	42771	26,10	43682	27,99	42727	25,32	43749	27,36	41260	23,27
<b>I8</b>	49053	49,66	50117	51,40	50032	51,04	<b>47254</b>	<b>47,35</b>	47802	48,58	49134	49,86
<b>I9</b>	29842	<b>23,33</b>	30056	24,85	30994	26,60	30722	25,24	31159	25,50	<b>29831</b>	23,53
<b>I10</b>	36917	27,66	34509	23,34	34654	24,01	34788	23,87	<b>34404</b>	<b>22,67</b>	34721	22,78
	51260		51606		51885		<b>50994</b>		51600		51224	

Fonte: própria.

Tabela 12 – Soluções das instâncias-teste de calibração para a heurística LB com 10min de execução.

	no		yes	
	Sol	GAP	Sol	GAP
<b>I1</b>	<b>72280</b>	39,25	73277	<b>38,83</b>
<b>I2</b>	48942	26,97	<b>48244</b>	<b>25,52</b>
<b>I3</b>	<b>58688</b>	31,45	58936	<b>29,22</b>
<b>I4</b>	47139	23,21	<b>46519</b>	<b>21,75</b>
<b>I5</b>	49635	<b>21,89</b>	<b>49573</b>	22,11
<b>I6</b>	<b>77763</b>	39,29	77864	<b>38,72</b>
<b>I7</b>	42727	25,32	<b>41189</b>	<b>22,48</b>
<b>I8</b>	<b>47254</b>	<b>47,35</b>	47562	48,20
<b>I9</b>	<b>30722</b>	<b>25,24</b>	31302	26,85
<b>I10</b>	<b>34788</b>	23,87	34911	<b>23,70</b>
	50994		<b>50938</b>	

Fonte: própria.

Tabela 13 – Soluções das instâncias-teste de calibração para a heurística RINS com 10min de execução.

	automatic (0)		none (-1)	
	Sol	GAP	Sol	GAP
<b>I1</b>	73277	<b>38,83</b>	79085	44,26
<b>I2</b>	<b>48244</b>	<b>25,52</b>	54982	34,92
<b>I3</b>	<b>58936</b>	29,22	59232	<b>22,41</b>
<b>I4</b>	<b>46519</b>	<b>21,75</b>	52687	31,19
<b>I5</b>	<b>49573</b>	<b>22,11</b>	52356	26,47
<b>I6</b>	<b>77864</b>	<b>38,72</b>	79377	40,01
<b>I7</b>	<b>41189</b>	<b>22,48</b>	46698	31,63
<b>I8</b>	<b>47562</b>	<b>48,20</b>	50113	50,43
<b>I9</b>	<b>31302</b>	<b>26,85</b>	31832	27,67
<b>I10</b>	34911	<b>23,70</b>	<b>34489</b>	27,10
	<b>50938</b>		54085	

Fonte: própria.

comparado ao GAP obtido na Tabela 17).

A Tabela 19 apresenta os resultados para o parâmetro *flow path cuts* que obteve as melhores soluções para o valor -1 (*do not generate*), com solução média de 50437 e GAP médio de 12,50.

Assim, na Tabela 20, exibimos a comparação entre os resultados obtidos pela calibração automática do software CPLEX, a calibração manual com a Etapa 1, a calibração manual com a Etapa 2 e a configuração padrão do CPLEX. Para 6 das 10 instâncias-teste, a calibração manual com cortes pela Etapa 1 obteve melhores soluções que a calibração automática do CPLEX. Enquanto para 9 das 10 instâncias-teste, a calibração manual com cortes pela Etapa 2 obteve melhores soluções que a calibração automática.

Tabela 14 – Soluções das instâncias-teste de calibração para a heurística FP (*Feasibility Pump*) com 10min de execução.

	<b>automatic (0)</b>		<b>none (-1)</b>		<b>feasibility (1)</b>		<b>both (2)</b>	
	<b>Sol</b>	<b>GAP</b>	<b>Sol</b>	<b>GAP</b>	<b>Sol</b>	<b>GAP</b>	<b>Sol</b>	<b>GAP</b>
<b>I1</b>	<b>73277</b>	<b>38,83</b>	73417	39,95	74550	40,91	75232	41,09
<b>I2</b>	48244	25,52	<b>47794</b>	<b>24,82</b>	48581	26,39	48883	26,89
<b>I3</b>	58936	29,22	<b>58654</b>	28,90	58920	<b>28,35</b>	60067	33,02
<b>I4</b>	<b>46519</b>	21,75	<b>46519</b>	21,75	46805	22,04	46550	<b>21,65</b>
<b>I5</b>	<b>49573</b>	<b>22,11</b>	<b>49573</b>	<b>22,11</b>	50200	23,09	50768	24,13
<b>I6</b>	77864	<b>38,72</b>	77864	<b>38,72</b>	77802	39,24	<b>77622</b>	39,02
<b>I7</b>	41189	22,48	<b>41104</b>	<b>22,29</b>	42931	25,64	44973	29,38
<b>I8</b>	47562	48,20	<b>47490</b>	<b>48,12</b>	51814	52,27	50404	51,20
<b>I9</b>	31302	26,85	<b>30968</b>	<b>26,06</b>	31442	27,22	32329	29,17
<b>I10</b>	<b>34911</b>	23,70	<b>34911</b>	23,69	<b>34911</b>	<b>23,12</b>	37169	28,35
	50938		<b>50829</b>		51796		52400	

Fonte: própria.

Tabela 15 – Soluções das instâncias-teste de calibração para todos os cortes ao mesmo tempo com 10min de execução.

	<b>automatic (0)</b>		<b>do not (-1)</b>		<b>moderate (1)</b>		<b>aggressive (2)</b>	
	<b>Sol</b>	<b>GAP</b>	<b>Sol</b>	<b>GAP</b>	<b>Sol</b>	<b>GAP</b>	<b>Sol</b>	<b>GAP</b>
<b>I1</b>	<b>73417</b>	39,95	73941	43,29	73973	41,44	75033	<b>27,82</b>
<b>I2</b>	<b>47794</b>	24,82	48410	27,15	48192	25,44	48071	<b>17,02</b>
<b>I3</b>	58654	28,90	<b>58404</b>	35,60	59709	33,36	58437	<b>6,90</b>
<b>I4</b>	46519	21,75	<b>45738</b>	23,25	46934	23,24	46359	<b>13,46</b>
<b>I5</b>	<b>49573</b>	22,11	50771	26,23	50212	24,17	50497	<b>15,32</b>
<b>I6</b>	77864	38,72	78536	45,07	77795	39,52	<b>77306</b>	<b>2,69</b>
<b>I7</b>	41104	22,29	40350	24,15	40744	22,45	<b>40114</b>	<b>13,83</b>
<b>I8</b>	47490	48,12	47389	51,07	50696	51,59	<b>46953</b>	<b>9,88</b>
<b>I9</b>	30968	26,06	<b>30049</b>	26,93	30871	26,52	31207	<b>16,29</b>
<b>I10</b>	34911	23,69	<b>34299</b>	39,32	35025	24,64	34489	<b>13,54</b>
	50829		<b>50789</b>		51415		50847	

Fonte: própria.

Tabela 16 – Resultados da calibração das instâncias-teste dos cortes por cliques com 10min de execução.

	automatic (0)		do not (-1)		moderate (1)		aggressive (2)		very aggr. (3)	
	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP
<b>I1</b>	73417	39,95	<b>73277</b>	<b>38,83</b>	<b>73277</b>	<b>38,83</b>	76676	42,84	76676	42,84
<b>I2</b>	<b>47794</b>	<b>24,82</b>	48019	25,19	48019	25,19	48882	26,77	48882	26,77
<b>I3</b>	58654	<b>28,90</b>	58936	29,36	59257	29,80	<b>58285</b>	31,68	58693	32,03
<b>I4</b>	46519	21,75	<b>46269</b>	<b>21,08</b>	46898	22,38	46867	22,31	46877	22,33
<b>I5</b>	<b>49573</b>	22,11	<b>49573</b>	22,10	<b>49573</b>	22,10	<b>49573</b>	<b>21,99</b>	<b>49573</b>	<b>21,99</b>
<b>I6</b>	77864	<b>38,72</b>	77887	38,74	77887	38,74	<b>77195</b>	<b>38,72</b>	77610	39,05
<b>I7</b>	<b>41104</b>	<b>22,29</b>	<b>41104</b>	<b>22,29</b>	<b>41104</b>	22,31	42600	25,15	42600	25,15
<b>I8</b>	47490	48,12	<b>47444</b>	<b>48,07</b>	47562	48,20	48822	49,41	48822	49,41
<b>I9</b>	<b>30968</b>	<b>26,06</b>	31038	26,23	31038	26,23	31311	26,95	31736	27,93
<b>I10</b>	34911	23,69	34911	23,70	34911	23,70	<b>34489</b>	<b>22,49</b>	<b>34489</b>	<b>22,49</b>
	<b>50829</b>		50846		50953		51470		51596	

Fonte: própria.

Tabela 17 – Resultados das instâncias-teste de calibração para o parâmetro de cortes *covers* com 10min de execução.

	automatic (0)		do not(-1)		moderate (1)		aggressive (2)		very aggr. (3)	
	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP
<b>I1</b>	73417	39,95	74735	41,49	73960	40,17	<b>72955</b>	<b>33,32</b>	<b>72955</b>	<b>33,32</b>
<b>I2</b>	<b>47794</b>	24,82	49885	28,25	49008	27,09	48948	<b>17,38</b>	48948	17,39
<b>I3</b>	58654	28,90	<b>58209</b>	31,32	<b>58209</b>	31,32	58484	<b>6,36</b>	58484	<b>6,36</b>
<b>I4</b>	46519	21,75	46421	21,55	46421	21,55	<b>45880</b>	<b>12,39</b>	45924	12,47
<b>I5</b>	<b>49573</b>	22,11	51191	25,15	51191	25,15	50403	<b>15,96</b>	50403	<b>15,96</b>
<b>I6</b>	77864	38,72	77958	39,53	77643	39,20	<b>77195</b>	3,16	<b>77195</b>	<b>3,15</b>
<b>I7</b>	<b>41104</b>	<b>22,29</b>	41224	22,62	41224	22,62	<b>40221</b>	<b>15,34</b>	<b>40221</b>	<b>15,34</b>
<b>I8</b>	47490	48,12	49820	50,55	49820	50,55	<b>46934</b>	<b>16,62</b>	<b>46934</b>	<b>16,62</b>
<b>I9</b>	<b>30968</b>	26,06	30187	24,26	30925	25,92	<b>29864</b>	<b>10,80</b>	<b>29864</b>	10,81
<b>I10</b>	34911	23,69	34911	23,63	34911	23,63	<b>34299</b>	<b>12,01</b>	<b>34299</b>	<b>12,01</b>
	50829		51454		51331		<b>50518</b>		50523	

Fonte: própria.

A calibração manual com cortes pela Etapa 2 apresentou melhores resultados, até mesmo quanto às médias, com o menor valor para solução (50437) e o menor valor para GAP (12,50%). Ainda assim, há uma redução de 19,00% quando comparados às soluções do padrão do CPLEX. Portanto a melhor solução encontrada é: calibração manual utilizando a Etapa 2, ou seja, mudando os parâmetros:

- ênfase 3 (*hidden feasible solutions*),
- escolha de variáveis 2 (*pseudo costs*),
- heurística LB *yes*,
- heurística FP 1 (*do not generate*),



Tabela 18 – Resultados das instâncias-teste de calibração para o parâmetro de cortes *flow covers* com 10min de execução.

	automatic (0)		do not (-1)		moderate (1)		aggressive (2)	
	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP
<b>I1</b>	72955	33,32	<b>71110</b>	<b>23,24</b>	71541	25,60	71541	25,60
<b>I2</b>	48948	17,38	48003	16,18	<b>47971</b>	<b>16,16</b>	<b>47971</b>	<b>16,16</b>
<b>I3</b>	58484	<b>6,36</b>	59317	18,11	<b>58440</b>	7,14	<b>58440</b>	7,14
<b>I4</b>	45880	12,39	45587	<b>10,70</b>	<b>45292</b>	11,20	<b>45292</b>	11,20
<b>I5</b>	<b>50403</b>	15,96	51054	18,50	51868	<b>14,54</b>	52168	14,97
<b>I6</b>	77195	3,16	77449	18,37	<b>77191</b>	<b>3,06</b>	<b>77191</b>	3,11
<b>I7</b>	<b>40221</b>	15,34	40223	15,20	40835	15,07	40767	<b>14,98</b>
<b>I8</b>	<b>46934</b>	16,62	47316	17,93	46985	<b>9,19</b>	46985	<b>9,19</b>
<b>I9</b>	29864	10,80	29668	11,06	<b>29613</b>	<b>10,10</b>	<b>29613</b>	<b>10,10</b>
<b>I10</b>	<b>34299</b>	<b>12,01</b>	34489	12,29	34721	13,13	34721	13,13
	50518		50422		<b>50446</b>		50469	

Fonte: própria.

Tabela 19 – Resultados das instâncias-teste de calibração para o parâmetro de cortes *flow path* com 10min de execução.

	automatic (0)		do not (-1)		moderate (1)		aggressive (2)	
	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP
<b>I1</b>	<b>71541</b>	<b>25,60</b>	<b>71541</b>	<b>25,60</b>	<b>71541</b>	<b>25,60</b>	<b>71541</b>	<b>25,60</b>
<b>I2</b>	<b>47971</b>	<b>16,16</b>	<b>47971</b>	<b>16,16</b>	<b>47971</b>	<b>16,16</b>	49179	18,19
<b>I3</b>	<b>58440</b>	7,14	<b>58440</b>	7,14	<b>58440</b>	7,14	<b>58440</b>	<b>7,09</b>
<b>I4</b>	<b>45292</b>	<b>11,20</b>	<b>45292</b>	<b>11,20</b>	<b>45292</b>	<b>11,20</b>	<b>45292</b>	<b>11,20</b>
<b>I5</b>	<b>51868</b>	<b>14,54</b>	<b>51868</b>	<b>14,54</b>	52168	14,97	52418	15,85
<b>I6</b>	<b>77191</b>	<b>3,06</b>	<b>77191</b>	<b>3,06</b>	<b>77191</b>	3,11	<b>77191</b>	3,07
<b>I7</b>	40835	15,07	40744	14,88	40835	15,07	<b>40733</b>	<b>14,85</b>
<b>I8</b>	<b>46985</b>	<b>9,19</b>	<b>46985</b>	<b>9,19</b>	<b>46985</b>	<b>9,19</b>	<b>46985</b>	<b>9,19</b>
<b>I9</b>	<b>29613</b>	10,10	<b>29613</b>	10,10	<b>29613</b>	10,10	<b>29613</b>	<b>10,09</b>
<b>I10</b>	<b>34721</b>	13,13	<b>34721</b>	13,13	<b>34721</b>	13,13	<b>34721</b>	<b>13,11</b>
	50446		<b>50437</b>		50476		50611	

Fonte: própria.

- *covers 2 (aggressive)*,
- *flow covers 1 (do not generate)*, e
- *flow path 1 (do not generate)*.

Tabela 20 – Comparação entre os resultados da calibração automática, das calibrações manuais e o padrão do software com 10min de execução.

	automático		manual + Etapa 1		manual + Etapa 2		padrão	
	Sol	GAP	Sol	GAP	Sol	GAP	Sol	GAP
<b>I1</b>	77354	43,53	73941	43,29	<b>71541</b>	<b>25,60</b>	75691	41,40
<b>I2</b>	48289	26,15	48410	27,15	<b>47971</b>	<b>16,16</b>	48978	26,81
<b>I3</b>	60527	33,49	<b>58404</b>	35,60	<b>58440</b>	<b>7,14</b>	59099	27,24
<b>I4</b>	46908	22,99	45738	23,25	<b>45292</b>	<b>11,20</b>	47257	23,09
<b>I5</b>	<b>49706</b>	22,27	50770	26,23	51868	<b>14,54</b>	51170	24,75
<b>I6</b>	77608	39,07	78536	45,07	<b>77191</b>	<b>3,06</b>	77969	39,71
<b>I7</b>	41668	23,86	<b>40350</b>	24,15	40744	<b>14,88</b>	46712	31,94
<b>I8</b>	47225	48,24	47389	51,07	<b>46985</b>	<b>9,19</b>	49840	50,45
<b>I9</b>	31085	26,17	30049	26,93	<b>29613</b>	<b>10,10</b>	29749	22,71
<b>I10</b>	36724	33,01	<b>34299</b>	39,32	34721	<b>13,13</b>	34404	22,51
<b>Média</b>	51709	31,88	50789	34,21	<b>50437</b>	<b>12,50</b>	52087	31,06

Fonte: própria.

## 4.4 Conclusões da calibração

Nesta seção são apresentados os resultados obtidos resolvendo as instâncias-teste pelo CPLEX com os parâmetros definidos como proposto na Seção 4.3. Os resultados obtidos são exibidos nas Tabelas 21 - 23, em que o tempo máximo de execução é de 1800s. MPJT representam os resultados do software na configuração padrão, enquanto MPJT\* representam a resolução utilizando a configuração proposta.

A Tabela 21 mostra que houve uma redução média de 12,10% no GAP quando utilizada a configuração de parâmetros proposta, além de apresentar 34 soluções de melhor qualidade quando comparadas às encontradas utilizando o CPLEX. Na Tabela 22 nota-se uma redução de 28,05% no GAP médio e uma melhoria na qualidade de 37 das 40 instâncias testadas neste grupo. Por fim, na Tabela 23 nota-se uma redução de 20,47% no GAP médio e uma melhoria na qualidade de 34 instâncias.

Ou seja, em geral, houve uma redução média de 20,21% do GAP e uma melhoria na qualidade de 105 das 120 instâncias-teste utilizando-se a calibração manual do CPLEX.

Tabela 21 – Resultados após a calibração do modelo proposto para instâncias com mesma proporção de entregas e coletas com 10min de execução.

	MPJT			MPJT*		
	Sol	T	GAP	Sol	T	GAP
(24,3,5,10,10)_1	74630	1804	40,46	<b>71541</b>	1804	<b>24,97</b>
(24,3,5,10,10)_2	48361	1808	25,71	<b>47971</b>	1806	<b>15,57</b>
(24,3,5,10,10)_3	58138	1806	20,82	<b>58121</b>	1804	<b>6,22</b>
(24,3,5,10,10)_4	46981	1806	22,48	<b>45139</b>	1805	<b>10,17</b>
(24,3,5,10,10)_5	51170	1809	24,59	<b>49573</b>	1806	<b>7,19</b>
(24,3,5,20,20)_1	114990	1806	7,46	<b>114324</b>	1808	<b>6,28</b>
(24,3,5,20,20)_2	97577	1805	9,21	<b>96012</b>	1808	<b>7,49</b>
(24,3,5,20,20)_3	87385	1805	12,23	<b>85230</b>	1805	<b>9,55</b>
(24,3,5,20,20)_4	<b>88998</b>	1806	<b>7,63</b>	89696	1807	8,23
(24,3,5,20,20)_5	82587	1806	10,74	<b>81296</b>	1806	<b>8,45</b>
(24,4,5,10,10)_1	<b>35355</b>	1808	19,97	35413	1806	<b>14,38</b>
(24,4,5,10,10)_2	54300	1805	33,64	<b>51696</b>	1807	<b>8,69</b>
(24,4,5,10,10)_3	52064	1805	44,35	<b>49662</b>	1807	<b>30,27</b>
(24,4,5,10,10)_4	66433	1807	60,91	<b>64231</b>	1807	<b>8,15</b>
(24,4,5,10,10)_5	<b>47197</b>	1807	<b>12,22</b>	48185	1807	12,65
(24,4,5,20,20)_1	75180	1806	17,01	<b>73040</b>	1806	<b>13,59</b>
(24,4,5,20,20)_2	84132	1807	9,63	<b>83796</b>	1807	<b>8,90</b>
(24,4,5,20,20)_3	90273	1806	10,33	<b>89182</b>	1806	<b>8,91</b>
(24,4,5,20,20)_4	103488	1808	10,31	<b>102178</b>	1807	<b>8,55</b>
(24,4,5,20,20)_5	103186	1808	22,79	<b>98564</b>	1805	<b>17,53</b>
(24,3,10,10,10)_1	99760	1806	32,47	<b>93967</b>	1804	<b>13,77</b>
(24,3,10,10,10)_2	120856	1807	47,32	<b>115449</b>	1803	<b>15,23</b>
(24,3,10,10,10)_3	<b>99161</b>	1804	38,90	99832	1800	<b>8,57</b>
(24,3,10,10,10)_4	111984	1805	48,18	<b>111401</b>	1808	<b>5,21</b>
(24,3,10,10,10)_5	96760	1807	18,15	<b>95814</b>	1807	<b>12,15</b>
(24,3,10,20,20)_1	180410	1807	17,30	<b>175438</b>	1800	<b>13,43</b>
(24,3,10,20,20)_2	166128	1807	14,09	<b>163338</b>	1800	<b>11,73</b>
(24,3,10,20,20)_3	173947	1803	<b>18,09</b>	<b>172167</b>	1806	20,01
(24,3,10,20,20)_4	181240	1807	13,48	<b>177403</b>	1805	<b>11,54</b>
(24,3,10,20,20)_5	<b>159472</b>	1808	<b>8,64</b>	164632	1806	12,12
(24,4,10,10,10)_1	127340	1807	33,70	<b>119184</b>	1804	<b>20,02</b>
(24,4,10,10,10)_2	106301	1808	45,36	<b>98186</b>	1800	<b>12,71</b>
(24,4,10,10,10)_3	<b>115123</b>	1809	47,70	106150	1800	<b>18,43</b>
(24,4,10,10,10)_4	92317	1810	38,60	<b>86939</b>	1800	<b>13,01</b>
(24,4,10,10,10)_5	136686	1808	50,57	<b>130602</b>	1800	<b>18,40</b>
(24,4,10,20,20)_1	193080	1808	26,69	<b>165179</b>	1809	<b>13,34</b>
(24,4,10,20,20)_2	144646	1806	<b>20,70</b>	<b>146463</b>	1800	21,25
(24,4,10,20,20)_3	171316	1808	25,54	<b>143573</b>	1805	<b>10,78</b>
(24,4,10,20,20)_4	158470	1809	<b>14,05</b>	<b>156547</b>	1808	14,11
(24,4,10,20,20)_5	171753	1809	19,17	<b>166252</b>	1806	<b>15,70</b>
<b>Média</b>	106729			103084		
<b>Mínimo</b>	35355			35413		
<b>Máximo</b>	193080			177403		

Fonte: própria.

Tabela 22 – Resultados após a calibração do modelo proposto para instâncias com muitas entregas e poucas coletas.

	MPJT			MPJT*		
	Sol	T	GAP	Sol	T	GAP
(24,3,5,10,5)_1	<b>50054</b>	1807	23,77	<b>50054</b>	1806	<b>13,74</b>
(24,3,5,10,5)_2	39446	1804	15,38	<b>39249</b>	1806	<b>3,32</b>
(24,3,5,10,5)_3	<b>44016</b>	1807	28,07	<b>44016</b>	1803	<b>6,30</b>
(24,3,5,10,5)_4	34414	1805	25,47	<b>34333</b>	1807	<b>16,04</b>
(24,3,5,10,5)_5	31606	1808	28,70	<b>31357</b>	100	<b>0,01</b>
(24,3,5,20,5)_1	41239	1804	22,55	<b>40114</b>	1805	<b>12,76</b>
(24,3,5,20,5)_2	31682	1808	15,17	<b>31005</b>	1805	<b>9,81</b>
(24,3,5,20,5)_3	49700	1808	50,15	<b>46934</b>	1800	<b>5,33</b>
(24,3,5,20,5)_4	<b>29613</b>	1809	21,84	<b>29613</b>	1808	<b>9,27</b>
(24,3,5,20,5)_5	<b>34299</b>	1805	21,99	34721	1809	<b>12,99</b>
(24,4,5,10,5)_1	34431	1807	34,51	<b>34401</b>	1185	<b>0,01</b>
(24,4,5,10,5)_2	31715	1806	38,46	<b>31496</b>	1806	<b>9,38</b>
(24,4,5,10,5)_3	54342	1807	54,79	<b>53953</b>	1807	<b>6,92</b>
(24,4,5,10,5)_4	39182	1807	40,48	<b>38764</b>	1800	<b>3,56</b>
(24,4,5,10,5)_5	46574	1806	45,21	<b>46424</b>	1807	<b>10,54</b>
(24,4,5,20,5)_1	26421	1804	32,12	<b>25591</b>	1800	<b>13,74</b>
(24,4,5,20,5)_2	36079	1808	19,54	<b>36065</b>	1805	<b>17,06</b>
(24,4,5,20,5)_3	<b>32672</b>	1806	36,76	32811	1806	<b>4,53</b>
(24,4,5,20,5)_4	31947	1807	18,18	<b>30861</b>	1806	<b>10,32</b>
(24,4,5,20,5)_5	24647	1806	45,58	<b>20839</b>	1536	<b>0,01</b>
(24,3,10,10,5)_1	59308	1807	18,53	<b>59240</b>	1800	<b>1,48</b>
(24,3,10,10,5)_2	88649	1806	34,80	<b>87978</b>	1805	<b>15,33</b>
(24,3,10,10,5)_3	111813	1807	61,30	<b>111669</b>	233	<b>0,01</b>
(24,3,10,10,5)_4	80197	1806	28,75	<b>77324</b>	1808	<b>5,63</b>
(24,3,10,10,5)_5	66404	1805	38,11	<b>65303</b>	168	<b>0,01</b>
(24,3,10,20,5)_1	64238	1806	46,51	<b>62380</b>	348	<b>0,01</b>
(24,3,10,20,5)_2	68925	1805	52,66	<b>67320</b>	1800	<b>6,97</b>
(24,3,10,20,5)_3	80554	1806	30,16	<b>76772</b>	1808	<b>10,96</b>
(24,3,10,20,5)_4	93147	1807	41,69	<b>82299</b>	1808	<b>9,78</b>
(24,3,10,20,5)_5	83179	1806	45,33	<b>76714</b>	1806	<b>10,75</b>
(24,4,10,10,5)_1	58883	1807	41,73	<b>57611</b>	1800	<b>8,68</b>
(24,4,10,10,5)_2	<b>59590</b>	1804	35,41	59722	1806	<b>10,67</b>
(24,4,10,10,5)_3	70540	1807	28,13	<b>68121</b>	1800	<b>1,64</b>
(24,4,10,10,5)_4	73988	1807	39,89	<b>72940</b>	1800	<b>7,86</b>
(24,4,10,10,5)_5	62992	1805	43,41	<b>59928</b>	1800	<b>5,15</b>
(24,4,10,20,5)_1	69384	1803	37,20	<b>62048</b>	1800	<b>7,89</b>
(24,4,10,20,5)_2	62319	1807	45,09	<b>58430</b>	1800	<b>9,47</b>
(24,4,10,20,5)_3	84217	1804	57,12	<b>73397</b>	1800	<b>5,44</b>
(24,4,10,20,5)_4	62479	1807	45,86	<b>59361</b>	1800	<b>6,78</b>
(24,4,10,20,5)_5	59154	1807	41,03	<b>55990</b>	1808	<b>19,55</b>
<b>Média</b>	55101			53179		
<b>Mínimo</b>	24647			20839		
<b>Máximo</b>	111813			111669		

Fonte: própria.

Tabela 23 – Resultados após a calibração do modelo proposto para instâncias com poucas entregas e muitas coletas.

	MPJT			MPJT*		
	Sol	T	GAP	Sol	T	GAP
(24,3,5,5,10)_1	<b>84425</b>	1807	45,67	<b>84425</b>	1806	<b>1,77</b>
(24,3,5,5,10)_2	<b>48630</b>	196	0,01	<b>48630</b>	34	<b>0,00</b>
(24,3,5,5,10)_3	54005	1806	20,63	<b>53671</b>	1808	<b>12,23</b>
(24,3,5,5,10)_4	<b>79571</b>	1805	4,57	<b>79571</b>	64	<b>0,01</b>
(24,3,5,5,10)_5	39593	1807	19,36	<b>39165</b>	187	<b>0,01</b>
(24,3,5,5,20)_1	77368	1808	39,03	<b>77191</b>	1804	<b>2,31</b>
(24,3,5,5,20)_2	82200	1808	6,72	<b>81629</b>	1800	<b>0,25</b>
(24,3,5,5,20)_3	88198	1809	6,96	<b>88105</b>	1806	<b>6,69</b>
(24,3,5,5,20)_4	54364	1808	12,64	<b>54007</b>	1808	<b>9,30</b>
(24,3,5,5,20)_5	<b>51813</b>	1808	16,95	<b>51813</b>	1800	<b>0,24</b>
(24,4,5,5,10)_1	60750	1807	24,53	<b>60701</b>	1806	<b>20,11</b>
(24,4,5,5,10)_2	<b>48165</b>	1807	18,73	48177	1805	<b>6,58</b>
(24,4,5,5,10)_3	57476	1804	40,11	<b>57179</b>	1806	<b>3,52</b>
(24,4,5,5,10)_4	55459	1805	31,16	<b>55085</b>	1807	<b>4,01</b>
(24,4,5,5,10)_5	<b>55777</b>	1806	33,54	<b>55777</b>	1807	<b>16,89</b>
(24,4,5,5,20)_1	<b>49082</b>	1806	18,71	49590	1806	<b>10,79</b>
(24,4,5,5,20)_2	61807	1805	<b>22,54</b>	<b>61702</b>	1807	23,29
(24,4,5,5,20)_3	58288	1808	29,87	<b>57939</b>	1806	<b>8,17</b>
(24,4,5,5,20)_4	49470	1807	18,09	<b>49064</b>	1806	<b>10,98</b>
(24,4,5,5,20)_5	<b>56218</b>	1807	20,53	56321	1806	<b>12,23</b>
(24,3,10,5,10)_1	87134	1806	35,62	<b>86076</b>	1806	<b>5,23</b>
(24,3,10,5,10)_2	<b>178481</b>	1807	36,85	178522	1807	<b>6,33</b>
(24,3,10,5,10)_3	80522	1808	28,68	<b>79552</b>	1342	<b>0,01</b>
(24,3,10,5,10)_4	<b>146566</b>	1808	40,02	<b>146566</b>	1807	<b>1,68</b>
(24,3,10,5,10)_5	155840	1806	54,52	<b>155700</b>	1806	<b>3,82</b>
(24,3,10,5,20)_1	88199	1805	29,52	<b>87052</b>	1807	<b>10,52</b>
(24,3,10,5,20)_2	101919	1805	20,42	<b>99125</b>	1800	<b>5,60</b>
(24,3,10,5,20)_3	134525	1806	29,24	<b>133761</b>	1803	<b>12,57</b>
(24,3,10,5,20)_4	<b>109582</b>	1805	32,90	109831	1805	<b>12,04</b>
(24,3,10,5,20)_5	161152	1807	40,87	<b>161097</b>	182	<b>0,01</b>
(24,4,10,5,10)_1	88676	1807	42,15	<b>87914</b>	1800	<b>7,85</b>
(24,4,10,5,10)_2	121368	1807	29,77	<b>118387</b>	1800	<b>1,07</b>
(24,4,10,5,10)_3	66679	1807	38,92	<b>65450</b>	1800	<b>9,93</b>
(24,4,10,5,10)_4	75271	1807	30,77	<b>75218</b>	1800	<b>1,89</b>
(24,4,10,5,10)_5	71191	1805	27,98	<b>71025</b>	1804	<b>5,36</b>
(24,4,10,5,20)_1	71500	1807	33,54	<b>69216</b>	1803	<b>14,55</b>
(24,4,10,5,20)_2	77841	1805	49,78	<b>77663</b>	1800	<b>12,32</b>
(24,4,10,5,20)_3	<b>98448</b>	1804	<b>27,74</b>	100889	1805	28,98
(24,4,10,5,20)_4	62364	1807	37,06	<b>61632</b>	1800	<b>5,77</b>
(24,4,10,5,20)_5	93235	1803	38,88	<b>88130</b>	1805	<b>21,98</b>
Média	82079			81564		
Mínimo	39593			39165		
Máximo	178481			178522		

Fonte: própria.



---

## HEURÍSTICA PROPOSTA

---

A heurística proposta é derivada do *Kernel search*, uma heurística baseada em programação matemática que foi inicialmente desenvolvida para resolver um problema de mochila multidimensional (Angelelli, Mansini e Speranza (2010)). Segundo os autores, esta é uma heurística estável e facilmente adaptável para problemas lineares inteiros mistos em que as variáveis inteiras são binárias, o que levou à escolha desta heurística para buscar soluções para o problema.

O método consiste, basicamente, em classificar as variáveis de decisão e identificar um conjunto de variáveis promissoras, chamado de núcleo (*kernel*). As demais variáveis são divididas em subconjuntos chamados de *buckets*. A ideia da heurística consiste em resolver subproblemas restritos a conjuntos de variáveis, buscando soluções para estes subproblemas. Desta forma, primeiramente, se resolve o subproblema restrito somente às variáveis do *kernel*. A partir de sua resolução, são adicionadas variáveis promissoras ao *kernel* de acordo com a construção dos *buckets*. Então o subproblema resultante é resolvido. Todos os *buckets* construídos são avaliados e, por fim, o melhor resultado obtido pelo método é retornado.

A heurística *Kernel search* foi aplicada com sucesso para resolver diferentes problemas de otimização, tais como o problema de seleção de portfólios (Angelelli, Mansini e Speranza (2010)) e mais recentemente, o de localização de facilidades (Guastaroba e Speranza (2014)).

Frente aos bons resultados obtidos na literatura, neste trabalho propõe-se uma heurística *Kernel search* para resolução do problema de *cross-docking* estudado. Na Seção 5.1 descreve-se brevemente a ideia da heurística e, em seguida, o método desenvolvido é detalhado.

### 5.1 *Kernel search*

O *Kernel search* é construído, essencialmente, com base na relaxação linear do problema original (Angelelli, Mansini e Speranza (2010)). O método consiste em ordenar as variáveis do problema priorizando as variáveis consideradas promissoras (a partir da solução obtida na

solução do problema relaxado). São resolvidos subproblemas restritos a conjuntos de variáveis promissoras e, a cada passo, adicionam-se novas variáveis até que todas as variáveis do problema original sejam verificadas.

O Algoritmo 1 apresenta a estrutura básica do método *Kernel search*, adaptada de Angelelli, Mansini e Speranza (2010). Os Passos 1-6 são chamados de fase de inicialização, pois é nesta fase que o *kernel* e os *buckets* são definidos. Basicamente, resolve-se a relaxação linear do problema original (Passo 2) e, a partir de uma solução ótima, as  $N$  variáveis binárias do problema são ordenadas de acordo com um critério de ordenação específico (Passo 3). Constrói-se, então, um núcleo inicial  $\Lambda$  com as  $NK$  primeiras variáveis da ordenação (Passo 4). As demais variáveis são distribuídas em subconjuntos  $B_i$  (*buckets*), que podem ou não ter a mesma cardinalidade (Passo 5). Por fim, o subproblema inteiro misto restrito às variáveis pertencentes ao conjunto  $\Lambda$  é resolvido e sua solução armazenada (Passo 6).

A próxima fase, representada pelos Passos 7-15, é chamada de fase de extensão. Nesta fase, para cada *bucket*  $B_i$  é construído um *kernel* temporário definido pela união do *kernel* original com o *bucket*:  $\Lambda_i = \Lambda \cup B_i$  (Passo 9). Resolve-se, então, o subproblema inteiro restrito às variáveis do *kernel* temporário adicionando duas restrições: I) o valor da melhor solução encontrada até o momento é um limitante para o subproblema atual, e II) impõe-se que pelo menos uma das variáveis de  $B_i$  deve ser incluída numa nova solução (Passo 10).

Caso o subproblema encontre uma solução factível, o *kernel* original  $\Lambda$  deve ser atualizado, recebendo todas as variáveis do *bucket* atual ( $B_i$ ) que tiveram valor um na solução atual (Passos 11-13). Caso o subproblema não encontre uma solução viável no tempo limite ou seja infactível, o *bucket* atual é adicionado ao *kernel*. A etapa de extensão é repetida para todos os subconjuntos  $B_i$ .

## 5.2 *Kernel search* para o problema estudado

Para o problema de *cross-docking* estudado, temos 4 conjuntos de variáveis binárias, representadas por:

- $\mathcal{X} = \{x_{ikt}^e \mid f \in F, i \in I, k \in K, t \in T\}$ , em que  $x_{ikt}^e$  assume o valor 1 se o produto  $i$  da entrega  $f$  tem descarregamento iniciado no instante  $t$  no *cross-dock*  $k$  e, zero caso contrário;
- $\Upsilon = \{y_{ikt}^c \mid c \in C, i \in I, k \in K, t \in T\}$ , em que  $y_{ikt}^c$  assume o valor 1 se o produto  $i$  da coleta  $c$  tem carregamento iniciado no instante  $t$  no *cross-dock*  $k$  e, zero caso contrário;
- $\Delta = \{\delta^c \mid c \in C\}$ , em que  $\delta^c$  assume o valor 1 se a janela de tempo desejável da coleta  $c$  é atendida e zero, caso contrário;
- $\zeta = \{z_{kk't} \mid k \in K, k' \neq k \in K, t \in T\}$ , em que  $z_{kk't}$  assume o valor 1 se há transferência de produtos do *cross-dock*  $k$  para  $k'$  no período  $t$  e zero, caso contrário.

Para adaptar a heurística *Kernel search* a outros problemas é necessário definir dois



---

**Algoritmo 1:** *Kernel search* adaptado de [Angelelli, Mansini e Speranza \(2010\)](#).

---

- 1  $x^* = \emptyset$  e  $z^* = \infty$ ;
  - 2 Resolva a relaxação linear do problema  $PI(N)$ ,  $RL[PI(N)]$ ;
  - 3 **Ordenação:** Ordene as variáveis binárias do problema de acordo com um critério;
  - 4 **Seleção:** Construa o *kernel* inicial  $\Lambda$ , selecionando as  $NK$  variáveis de acordo com a ordenação estabelecida;
  - 5 Divida as  $N \setminus \Lambda$  variáveis restantes em  $\{B_i\}$  *buckets* disjuntos, em que  $i = 1, 2, \dots, NB$  e  $NB$  é o número de *buckets* gerados;
  - 6 Resolva  $PI(\Lambda)$ . Tome  $x^*$  e  $z^*$  como a melhor solução obtida e seu valor, que é um limitante primal para o problema;
  - 7  $i = 1$ ;
  - 8 **Enquanto**  $i \leq NB$  **faça**
    - 9 Construa o *kernel* temporário  $\Lambda_i = \Lambda \cup B_i$ ;
    - 10 Resolva  $PI(\Lambda_i)$  adicionando duas restrições: I)  $z^*$  é um limitante primal para o subproblema, e II) pelo menos uma das variáveis de  $B_i$  deve pertencer à solução;
    - 11 **Se**  $PI(\Lambda_i)$  *é factível* **então**
      - 12 Sejam  $x'$  e  $z'$  a solução encontrada e seu valor, então:  $x^* = x'$  e  $z^* = z'$ ;
      - 13 Defina  $\Lambda_i^*$  como o conjunto de variáveis de  $B_i$  que têm valores positivos em  $x'$ ;
      - 14 Faça:  $\Lambda = \Lambda \cup \Lambda_i^*$ ;
    - 15 **senão**
      - 16 Faça:  $\Lambda = \Lambda \cup \Lambda_i$ ;
    - 17  $i = i + 1$ ;
  - 18 Se  $z^* \neq \infty$ , retorne a melhor solução factível ( $x^*$ ) encontrada e seu valor ( $z^*$ ).
- 

aspectos: I) o critério de ordenação das variáveis; e II) como construir o *kernel* inicial e os *buckets*, ou seja, definir a quantidade de variáveis a constituir-los e o número de subconjuntos.

As variáveis ordenadas são apenas as variáveis dos conjuntos  $\chi$  e  $\Upsilon$ , uma vez que as variáveis de transferência (conjunto  $\zeta$ ) são fortemente dependentes das entregas e das coletas realizadas nos *cross-docks*, ou seja, das variáveis  $x_{ikt}^e$  e  $y_{ikt}^c$ . Desta forma, no critério de ordenação utilizado, as variáveis  $x_{ikt}^e$  e  $y_{ikt}^c$  têm prioridade na construção do *kernel*. As variáveis de transferência são, então, selecionadas para compor o *kernel* e os *buckets* a partir das variáveis previamente selecionadas de  $\chi$  e  $\Upsilon$ . Como a quantidade de variáveis  $\delta^c$  é razoavelmente pequena (cardinalidade igual à quantidade de coletas  $C$ ) todas são adicionadas ao *kernel*.

### 5.2.1 Critério de ordenação das variáveis

Quanto ao critério de ordenação das variáveis e tendo em mãos o resultado obtido na relaxação linear do problema, foram analisadas duas opções:

**Estratégia de ordenação 1 ( $O_{cr}$ ):** as variáveis  $x_{ikt}^e$  e  $y_{ikt}^c$  são **ordenadas**, primeiramente, de acordo com seus valores na solução do problema relaxado, em ordem não-crescente. No caso em que as variáveis sejam iguais a um, elas são ordenadas de acordo com o valor de seu custo relativo na solução atual, dando preferência às variáveis com menor custo relativo. Caso existam

duas ou mais variáveis com custo relativo igual, utiliza-se como critério de desempate seu custo na função objetivo, pois o objetivo é minimizar os custos da solução, logo variáveis com custo mais baixo têm preferência. Em seguida, ordenam-se as variáveis básicas (com valor menor que um e maior que zero) de acordo com o valor de seu custo na função objetivo. Por fim, são ordenadas as variáveis com valor nulo e, neste caso, o critério de ordenação utilizado é o mesmo das variáveis com valor um.

A Figura 10 ilustra um conjunto de variáveis para as quais são reportados os custos na função objetivo, o valor e o custo relativo na solução ótima do problema relaxado. Na Figura 11 são apresentados os passos (E1 e E2) para a ordenação  $O_{cr}$ . Como descrito, primeiramente se ordena pelos valores na solução (E1). Desta forma, em amarelo ficaram as variáveis com valor 1 ( $q7$  e  $q9$ ), em rosa as variáveis com valores entre 0 e 1 ( $q1$ ,  $q2$  e  $q8$ ), e em azul as variáveis com valor 0 ( $q3$ ,  $q4$ ,  $q5$ ,  $q6$ ,  $q10$  e  $q11$ ). As variáveis destacadas em negrito possuem mesmo valor na solução, o que leva a próxima etapa de ordenação (E2), em que as variáveis com valores iguais a 1 e 0 são ordenadas de acordo com seu custo relativo e as variáveis com valores entre 1 e 0 são ordenadas de acordo com seu custo na função objetivo. As variáveis em negrito são as que empatam quanto ao valor do custo relativo, sendo necessário ordená-las pelo terceiro critério, que é o custo na função objetivo. Assim, o resultado da ordenação é representado pelo vetor  $O_{cr} = \{q9, q7, q1, q2, q8, q5, q6, q11, q10, q3, q4\}$ .

Figura 10 – Dados do exemplo para a ordenação .

Nome da variável	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	q11
Custo na f.o.	5320,00	5320,00	19110,20	19110,20	1064,00	1064,00	3420,00	1915,20	1064,00	10358,80	1900,00
Valor na solução	0,40	0,40	0,00	0,00	0,00	0,00	1,00	0,29	1,00	0,00	0,00
Custo relativo	0,00	0,00	13752,20	13752,20	0,00	0,00	0,00	0,00	0,00	5042,17	844,80

Fonte: própria.

Figura 11 – Exemplo para a ordenação  $O_{cr}$ .

	q7	q9	q1	q2	q8	q3	q4	q5	q6	q10	q11
	q7	q9	q1	q2	q8	q5	q6	q11	q10	q3	q4
$O_{cr}$	q9	q7	q1	q2	q8	q5	q6	q11	q10	q3	q4

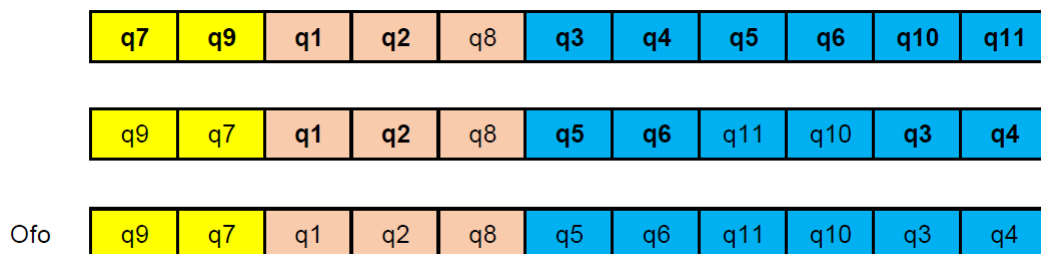
Fonte: própria.

**Estratégia de ordenação 2 ( $O_{fo}$ ):** as variáveis  $x_{ikt}^e$  e  $y_{ikt}^c$  são **ordenadas**, primeiramente, de acordo com seus valores na solução do problema relaxado, em ordem não-crescente. Caso existam duas ou mais variáveis com valor igual, elas são ordenadas de acordo com o valor de seu custo na função objetivo da solução atual, de forma que variáveis com custo mais baixo têm

preferência. Utilizamos como critério de desempate o valor do custo relativo na solução atual, dando preferência às variáveis com menor custo relativo.

A Figura 12 ilustra os passos da estratégia de ordenação  $O_{fo}$  para o exemplo da Figura 10. Como descrito, primeiramente se ordena pelos valores na solução (E1). As variáveis que possuem mesmo valor na solução são destacadas em negrito o que leva a próxima etapa (E2), em que as variáveis são ordenadas de acordo com seu custo na função objetivo. As variáveis em negrito são as que empatam quanto ao valor do custo na função objetivo, sendo necessário ordená-las pelo terceiro critério, que é o custo relativo. Assim, o resultado da ordenação é representado pelo vetor  $O_{fo} = \{q9, q7, q1, q2, q8, q5, q6, q11, q10, q3, q4\}$ .

Figura 12 – Exemplo para a ordenação  $O_{fo}$ .



Fonte: própria.

### 5.2.2 Seleção de variáveis para a construção do kernel inicial e dos buckets

O número de variáveis do *kernel* inicial foi definido com o auxílio de testes computacionais realizados para instâncias de pequena dimensão e considerando que o tempo de execução do método é limitado a 10min. Ele é dado por:

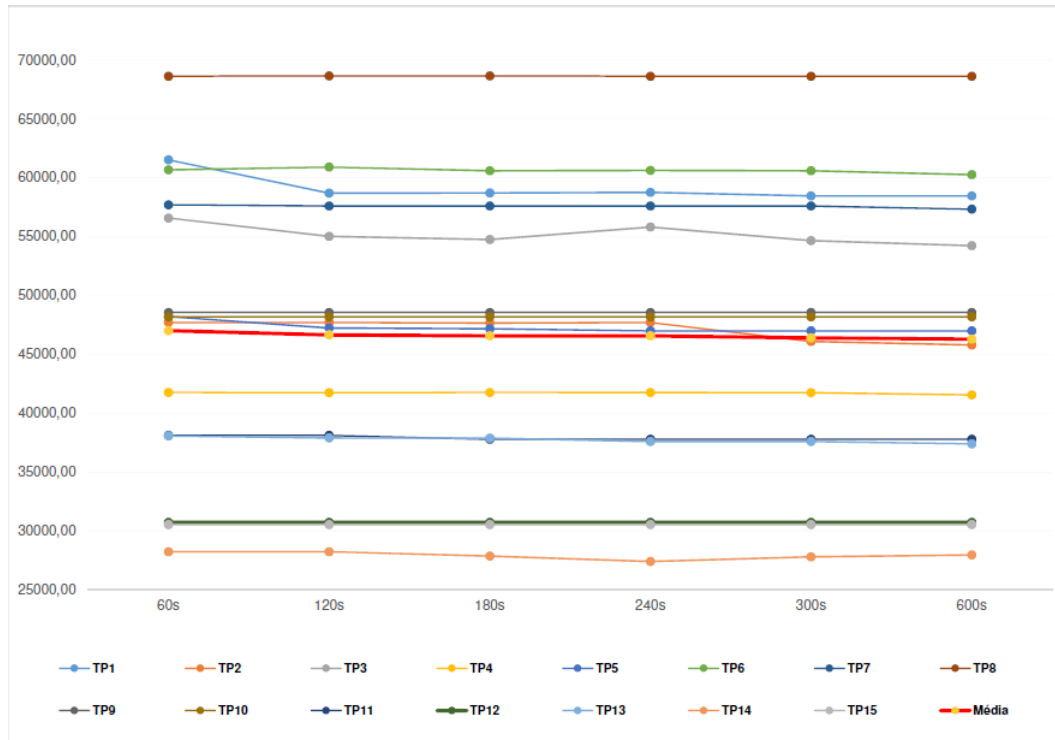
$$TK = \min \{40\% \text{ das variáveis pertencentes aos conjuntos } \chi \text{ e/ou } \Upsilon, 1800\}.$$

Estes valores foram definidos através da análise das soluções encontradas com o problema restrito ao *kernel* inicial com tempo máximo de 5min. Notou-se que subproblemas com até 40% das variáveis resultavam em soluções ótimas ou com uma diferença média com relação à solução do CPLEX (com 30min de execução) de 0,24%. Outro fator analisado foi a dimensão dos problemas para a qual o software encontrava a solução ótima para as instâncias pequenas e o tamanho encontrado foi de cerca de 1800 variáveis (considerando-se somente as variáveis dos conjuntos  $\chi$  e  $\Upsilon$ ).

Para a execução do método foi necessário definir o tempo de execução do primeiro subproblema (somente com o *kernel* inicial). A Figura 13 apresenta o gráfico com as soluções para as instâncias de pequena dimensão (eixo vertical) encontradas para o subproblema contendo somente o *kernel* inicial com tempo de execução máximo de 60s, 120s, 180s, 240s, 300s

e 600s (eixo horizontal). Nota-se que a partir de 180s a qualidade da solução obtida não melhora significativamente. Assim, o tempo máximo de execução estabelecido para o primeiro subproblema foi de 180s, ou seja, 30% do tempo total da heurística (10min).

Figura 13 – Gráfico **solução x tempo** para o *kernel* inicial de problemas de pequena dimensão.



Fonte: própria.

O conjunto  $\hat{\chi}$  representa as variáveis de  $\chi$  selecionadas na criação do *kernel* inicial, enquanto  $\hat{\Upsilon}$  representa as variáveis de  $\Upsilon$  que compõem o *kernel* inicial. A partir dos conjuntos  $\hat{\chi}$  e  $\hat{\Upsilon}$  que compõem o *kernel* é criado um conjunto  $\Phi$  de *cross-docks*, que corresponde a todos os *cross-docks*  $k$  relacionados às variáveis destes conjuntos. Por exemplo, se uma variável  $x_{i2t}^e$  é adicionada ao *kernel* inicial, então o *cross-dock* 2 é incluído no conjunto  $\Phi$ . Analogamente, se uma variável  $y_{i6t}^c$  é adicionada ao *kernel* inicial, então o *cross-dock* 6 é incluído no conjunto  $\Phi$ .

Para a **construção do *kernel*** e seleção de variáveis para construir cada um dos *buckets*, foram analisadas 3 estratégias de seleção de variáveis:

**Seleção 1 ( $S_{xy}$ ):** são selecionadas todas as variáveis  $x_{ikt}^e$  e  $y_{ikt}^c$  com valores maiores que zero. Caso o total de variáveis seja inferior a  $TK$ , complementa-se o conjunto seguindo a ordenação escolhida. A partir do conjunto  $\Phi$  de *cross-docks*, as variáveis  $z_{kk't}$  correspondentes a esses *cross-docks* também são adicionadas ao *kernel* inicial.

Mais especificamente, a princípio constroem-se dois sub-núcleos, um para as variáveis do conjunto  $\chi$  e outro para as variáveis do conjunto  $\Upsilon$ . O núcleo de  $\chi$  é composto por todas as variáveis com valor positivo na relaxação linear e, caso a quantidade de variáveis seja inferior a  $TK = \min \{40\% \text{ das variáveis pertencentes a } \chi, 1800 * \text{porcentagem de variáveis de } \chi\}$

são adicionadas as variáveis seguindo a ordenação previamente realizada até que se obtenha tal dimensão. Analogamente, para o núcleo de  $\Upsilon$  entrarão todas as variáveis com solução da relaxação linear estritamente positiva e, caso a quantidade de variáveis seja inferior a  $TK = \min\{40\% \text{ das variáveis pertencentes ao conjunto } \Upsilon, 1800 * \text{porcentagem de variáveis de } \Upsilon\}$  são adicionadas as variáveis seguindo a ordenação até que se obtenha tal dimensão. Desta forma, unindo os dois sub-núcleos, obtém-se um núcleo com variáveis de  $\chi$  e de  $\Upsilon$ .

**Seleção 2 ( $S_x$ ):** são selecionadas todas as variáveis  $x_{ikt}^e$  com valores maiores que zero. Deseja-se que o *kernel* tenha pelo menos  $TK$  variáveis, então caso esta quantidade de variáveis não seja atingida, complementa-se o *kernel* seguindo a ordenação escolhida. A partir do conjunto de variáveis  $\widehat{\chi}$ , as variáveis do conjunto  $\Upsilon$  correspondentes a essas variáveis também passam a compor o *kernel*. Além disso, as variáveis  $z_{kk't}$  correspondentes aos *cross-docks* pertencentes ao conjunto  $\Phi$  de *cross-docks* selecionados através das variáveis selecionadas  $x_{ikt}^e$  também são selecionadas para compor o *kernel* inicial.

Nesta estratégia  $S_x$  são adicionadas ao *kernel* todas as variáveis de  $\chi$  com solução da relaxação linear maior que zero. Em seguida, as variáveis de  $\Upsilon$  correspondentes às variáveis pertencentes ao *kernel* são adicionadas, ou seja, caso uma variável de  $\chi$  com  $i = \widehat{i}$ ,  $k = \widehat{k}$  e  $t = \widehat{t}$  esteja no *kernel*, todas as variáveis de  $\Upsilon$  com  $i = \widehat{i}$ ,  $k = \widehat{k}$  e  $t \geq \widehat{t}$  também estarão no *kernel*. Caso a dimensão do núcleo até então criado seja inferior a  $TK$ , adicionam-se uma a uma as variáveis  $x_{ikt}^e$  de acordo com a ordenação e suas respectivas variáveis do conjunto  $\Upsilon$  até que tal dimensão seja atingida.

**Seleção 3 ( $S_y$ ):** análoga à seleção  $S_x$ , porém a prioridade de seleção deixa de ser das variáveis do conjunto  $\chi$  e passa a ser das variáveis do conjunto  $\Upsilon$ , ou seja, são adicionadas ao *kernel* todas as variáveis de  $\Upsilon$  com solução da relaxação linear maior que zero. Adicionam-se as variáveis de  $\Upsilon$  correspondentes às variáveis pertencentes ao conjunto  $\widehat{\Upsilon}$ . Desta forma, se uma variável  $y_{ikt}^c$  estiver no *kernel*, as variáveis  $\{x_{ikt}^e \mid t \leq \widehat{t}, e \in E\}$  também estarão no *kernel*. Caso a dimensão do núcleo até então criado seja inferior a  $TK$ , adicionam-se uma a uma as variáveis  $y_{ikt}^c$  da ordenação e todas as suas respectivas variáveis do conjunto  $\chi$  até que tal dimensão seja atingida.

Para a **construção dos buckets**, as variáveis restantes são divididas em subconjuntos de tamanho fixo, totalizando  $NB$  buckets. O número de buckets foi definido a partir de testes preliminares e é dado por  $\frac{(N_x - N_{kx}) + (N_y - N_{ky})}{TB}$ . Desta forma, o tamanho dos buckets é dado por:

$$TB = \max \left\{ 10\% \text{ das variáveis pertencentes aos conjuntos } \chi \text{ e/ou } \Upsilon; 180; \frac{(N_x - N_{kx}) + (N_y - N_{ky})}{\frac{\text{tempo restante}}{30}} \right\},$$

em que  $N_x$  é a cardinalidade do conjunto de variáveis  $\chi$  e  $N_y$  é a cardinalidade do conjunto  $\Upsilon$ .  $N_{kx}$  e  $N_{ky}$  representam a quantidade de variáveis do conjunto  $\widehat{\chi}$  e do conjunto  $\widehat{\Upsilon}$ , respectivamente, selecionadas para compor o *kernel* inicial. Para definir a fórmula acima foram realizados testes computacionais com as instâncias utilizadas para definir a cardinalidade do *kernel* inicial. Para

que os subproblemas não crescessem de forma repentina, utilizou-se um percentual de variáveis, assim como na construção do núcleo. Assim, estabeleceram-se os valores de 10% e 180 variáveis (uma vez que 180 representa 10% das 1800 variáveis que podem pertencer ao *kernel*). O outro fator da fórmula determina uma quantidade de variáveis para cada *bucket* de forma que ao adicionar cada um dos *buckets* haja, pelo menos, 30s para buscar soluções para o subproblema criado  $(\frac{(N_x - N_{kx}) + (N_y - N_{ky})}{\text{tempo restante}})$ .

Para a seleção das variáveis dos *buckets*, seguiu-se a ordenação utilizada após se extrair as variáveis do *kernel* inicial, selecionando  $TB$  variáveis para cada *bucket* a menos do último, que pode ter menos variáveis.

### 5.2.3 Melhorias propostas

Com o objetivo de buscar soluções de melhor qualidade, buscou-se analisar estratégias de adaptação para o *kernel search*. Foram exploradas duas estratégias:

**Adaptação 1 ( $A_k$ ):** consiste em tentar garantir que o núcleo não tenha uma dimensão muito grande com o desenvolvimento do método (que não fique com a cardinalidade muito próxima da cardinalidade do problema original). Desta forma, quando o subproblema utilizando o *bucket* atual possui solução infactível ou não encontra solução no tempo limite estabelecido para o subproblema, este subconjunto se une ao próximo, ou seja, o novo subproblema tem um *bucket* maior, porém o *bucket* que não retornou um resultado significativo não é adicionado ao *kernel*, é apenas unido ao próximo *bucket*.

Assim, estratégia  $A_k$  consiste em: se a união do *bucket* 3 ao *kernel* não retorna uma solução factível, então na próxima etapa serão adicionados os *buckets* 3 e 4 ao *kernel* como se a etapa anterior não tivesse ocorrido, ou seja, o *bucket* atual será composto por dois *buckets* e terá o dobro da dimensão original.

**Adaptação 2 ( $A_b$ ):** consiste em tornar menos restritivo um problema ao inserir cada novo *bucket*, buscando encontrar soluções factíveis para mais subproblemas. Para muitos subproblemas não se encontram soluções factíveis no tempo limite estabelecido ou o subproblema é infactível devido à imposição da restrição (II) adicionada no Passo 10 do Algoritmo 1. Então, o que se faz nesta nova estratégia é não inserir tal restrição no próximo subproblema sempre que a solução atual for infactível ou que não se encontre solução.

Assim, estratégia  $A_b$  consiste em: se a união do *bucket*  $B_i$  ao *kernel* não retorna uma solução factível, na próxima etapa ao unir o *bucket*  $B_{i+1}$  ao *kernel*, o novo subproblema não terá a imposição de que pelo menos uma das variáveis do novo *bucket* adicionado deve estar na nova solução.

No Capítulo 6 são apresentados testes computacionais realizados para avaliar a heurística desenvolvida.

---

## ANÁLISE DA HEURÍSTICA

---

Como destacado anteriormente, uma heurística *Kernel search* está baseada na escolha das variáveis que compõem o *kernel* e os *buckets*. No Capítulo 5 foram propostas duas estratégias de ordenação das variáveis dos conjuntos  $\chi$  e  $\Upsilon$ , três estratégias de seleção de variáveis para a construção do *kernel* e dos *buckets*, e duas estratégias de adaptação do *Kernel search*, com o objetivo de buscar soluções de melhor qualidade. Logo, na primeira etapa dos testes computacionais descrita na Seção 6.1, foram avaliadas as combinações destas estratégias. Uma vez apontada a melhor configuração, na Seção 6.2, foram realizados testes para todas as instâncias geradas na Seção 4.1.

As estratégias descritas na Seção 5.2 foram combinadas em busca da melhor configuração para a heurística *Kernel search* proposta. Como as estratégias de adaptação surgiram como tentativas de melhoria para as estratégias até então desenvolvidas, as combinações foram divididas em 3 fases:

**Fase 1:** consiste em combinar as estratégias de seleção  $S_{xy}$ ,  $S_x$  e  $S_y$  às duas estratégias de ordenação apresentadas ( $O_{cr}$  e  $O_{fo}$ );

**Fase 2:** agrega às combinações da Fase 1 a estratégia de adaptação  $A_k$ ; e

**Fase 3:** agrega às combinações da Fase 1 a estratégia de adaptação  $A_b$ .

Assim, a heurística *Kernel search* desenvolvida é chamada  $KS_{S,O,A}$ , por exemplo,  $KS_{xy,cr}$  é a heurística *Kernel search* considerando a estratégia de seleção  $S_{xy}$  e a de ordenação  $O_{cr}$ . Outro exemplo seria  $KS_{x,fo,k}$ , que representa a heurística *Kernel search* considerando a estratégia de seleção  $S_x$ , a estratégia de ordenação  $O_{fo}$  e a estratégia de adaptação  $A_k$ .

## 6.1 Parametrização

Utilizando-se as definições da Seção 5.2 para ordenação e construção do *kernel* e dos *buckets* foram testadas as combinações de estratégias das 3 fases apresentadas.

A análise de cada fase foi realizada a partir da estratégia de seleção utilizada. O desempenho destas variações da heurística foi avaliado utilizando as mesmas 10 instâncias de calibração do Capítulo 4, ou seja, as 10 com maiores GAPs das 15 instâncias (5 de cada um dos 3 cenários existentes) inicialmente selecionadas, como descrito na Seção 4.3.

As Tabelas 24-36 resumem os resultados obtidos pela heurística *kernel search* durante sua parametrização tendo como tempo de execução máximo 10 min. *Sol* representa a melhor solução encontrada no tempo  $T_i$  (dado em segundos) e o fator  $D_{30}$ , dado por  $D_{30} = 100 \frac{Sol.KS - Sol.MPJT}{Sol.MPJT}$ , que representa a diferença percentual entre a solução encontrada pelo CPLEX no tempo limite de 30min (Tabela 6) e a solução encontrada pela heurística no tempo limite de 10min e é dado em porcentagem. As melhores diferenças percentuais para cada variação são destacadas em negrito, assim como a melhor média para cada variação de construção do *kernel* inicial.

Nas Tabelas 24-26 são exibidos os resultados das seis combinações da Fase 1. A Tabela 24 resume os resultados obtidos para estratégia de seleção de variáveis  $S_{xy}$  combinada às duas estratégias de ordenação propostas. Para 7 das 10 instâncias a combinação  $KS_{xy,cr}$  apresentou soluções de melhor qualidade com diferença percentual média com relação à solução encontrada pelo CPLEX em 30min de 10,94%, ou seja, 4,51% melhor que  $KS_{xy,fo}$ .

Tabela 24 – Resultados dos testes da combinação da estratégia  $S_{xy}$  com as estratégias  $O_{cr}$  e  $O_{fo}$  para o *kernel search* com até 10min de execução.

	CPLEX		$KS_{xy,cr}$			$KS_{xy,fo}$		
	Sol	Ti	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	74630	1804	77287	607	<b>3,56</b>	101301	609	35,74
(24,3,5,10,10)_2	48361	1808	48733	608	<b>0,77</b>	57029	611	17,92
(24,3,5,10,10)_3	58138	1806	60632	610	<b>4,29</b>	63651	606	9,48
(24,3,5,10,10)_4	46981	1806	48489	610	<b>3,21</b>	51932	608	10,54
(24,3,5,10,10)_5	51170	1809	51928	608	<b>1,48</b>	54613	610	6,73
(24,3,5,5,20)_1	77368	1808	81107	613	<b>4,83</b>	93839	602	21,29
(24,3,5,20,5)_1	41239	1804	44188	603	<b>7,15</b>	49250	612	19,43
(24,3,5,20,5)_3	49700	1808	60953	607	22,64	55398	604	<b>11,46</b>
(24,3,5,20,5)_4	29613	1809	37121	609	25,35	34777	605	<b>17,44</b>
(24,3,5,20,5)_5	34299	1805	46701	603	36,16	35818	604	<b>4,43</b>
Média					<b>10,94</b>			15,45

Fonte: própria.

Na Tabela 25 os resultados apresentados mostram que para 7 instâncias a combinação  $KS_{x,fo}$  apresentou as melhores soluções e a diferença percentual média com relação ao CPLEX



é de 14,32%.

Tabela 25 – Resultados dos testes da combinação da estratégia  $S_x$  com as estratégias  $O_{cr}$  e  $O_{fo}$  para o *kernel search* com até 10min de execução.

	$KS_{x,cr}$			$KS_{x,fo}$		
	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	89482	604	19,90	88957	603	<b>19,20</b>
(24,3,5,10,10)_2	54405	605	12,50	49632	603	<b>2,63</b>
(24,3,5,10,10)_3	87199	604	49,99	68864	604	<b>18,45</b>
(24,3,5,10,10)_4	53292	612	13,43	53145	605	<b>13,12</b>
(24,3,5,10,10)_5	57323	611	<b>12,02</b>	58848	608	15,00
(24,3,5,5,20)_1	80093	604	<b>3,52</b>	81757	605	5,67
(24,3,5,20,5)_1	46008	605	11,56	44189	611	<b>7,15</b>
(24,3,5,20,5)_3	60505	606	21,74	53529	605	<b>7,70</b>
(24,3,5,20,5)_4	37741	612	27,45	34362	606	<b>16,04</b>
(24,3,5,20,5)_5	46521	604	<b>35,63</b>	47426	605	38,27
Média			20,78			<b>14,32</b>

Fonte: própria.

Os resultados da Tabela 26 mostram que para 9 das 10 instâncias  $KS_{y,cr}$  apresentou melhores resultados que  $KS_{y,fo}$  e a diferença percentual média é de 12,87%. Vale destacar que para uma das instâncias a solução média encontrada pelo *Kernel search* em 10min é melhor que a encontrada pelo CPLEX em 30min (-0,43%).

Tabela 26 – Resultados dos testes da combinação da estratégia  $S_y$  com as estratégias  $O_{cr}$  e  $O_{fo}$  para o *kernel search* com até 10min de execução.

	$KS_{y,cr}$			$KS_{y,fo}$		
	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	107522	605	<b>44,07</b>	107522	604	<b>44,07</b>
(24,3,5,10,10)_2	50740	417	<b>4,92</b>	56014	610	15,82
(24,3,5,10,10)_3	71569	370	23,10	68294	604	<b>17,47</b>
(24,3,5,10,10)_4	53847	501	<b>14,61</b>	53847	605	<b>14,61</b>
(24,3,5,10,10)_5	56529	417	<b>10,47</b>	60261	604	17,77
(24,3,5,5,20)_1	80928	613	<b>4,60</b>	95251	605	23,11
(24,3,5,20,5)_1	43360	512	<b>5,14</b>	48462	605	17,51
(24,3,5,20,5)_3	49487	459	<b>-0,43</b>	62966	606	26,69
(24,3,5,20,5)_4	35424	409	<b>19,62</b>	43132	605	45,65
(24,3,5,20,5)_5	35195	473	<b>2,61</b>	42027	605	22,53
Média			<b>12,87</b>			24,53

Fonte: própria.

As diferenças percentuais médias encontradas foram de, no mínimo, 10,94%, o que motivou a busca por estratégias de adaptação para o *kernel search*. Os resultados mostraram que para duas das 3 estratégias de seleção a melhor combinação é com a ordenação  $O_{cr}$ , ou seja, quando há prioridade em se ordenar pelo custo reduzido.

As Tabelas 27-29 exibem resultados obtidos pelo método com tempo máximo de 10min para as seis variações iniciais (da Fase 1) aliadas à estratégia de adaptação  $A_k$ , já descrita (Fase 2).

Na Tabela 27 os resultados mostram que a configuração  $KS_{xy,cr,k}$  é a melhor, com 10 soluções melhores que  $KS_{xy,fo,k}$  e encontrando soluções melhores que o CPLEX em 30min para 2 instâncias. A redução da diferença percentual quando comparada à melhor combinação encontrada para esta estratégia de seleção de variáveis sem o uso da estratégia de adaptação ( $KS_{xy,cr}$ ) é de 7,37%.

Tabela 27 – Resultados obtidos pelo *kernel search* para a combinação das estratégias  $KS_{xy,cr}$  e  $KS_{xy,fo}$  à estratégia de adaptação  $A_k$  com tempo máximo de execução do método de 10min.

	$KS_{xy,cr,k}$			$KS_{xy,fo,k}$		
	Sol	Ti	D_30	Sol	Ti	D_30
<b>(24,3,5,10,10)_1</b>	77287	607	<b>3,56</b>	101301	608	35,74
<b>(24,3,5,10,10)_2</b>	48326	578	<b>-0,07</b>	54982	609	13,69
<b>(24,3,5,10,10)_3</b>	60632	602	<b>4,29</b>	72437	610	24,59
<b>(24,3,5,10,10)_4</b>	49120	607	<b>4,55</b>	52152	605	11,01
<b>(24,3,5,10,10)_5</b>	51266	528	<b>0,19</b>	58842	605	14,99
<b>(24,3,5,5,20)_1</b>	81600	605	<b>5,47</b>	83327	590	7,70
<b>(24,3,5,20,5)_1</b>	41089	603	<b>-0,36</b>	46966	603	13,89
<b>(24,3,5,20,5)_3</b>	53245	449	<b>7,13</b>	63747	604	28,26
<b>(24,3,5,20,5)_4</b>	31376	256	<b>5,95</b>	36928	604	24,70
<b>(24,3,5,20,5)_5</b>	36001	421	<b>4,96</b>	36218	604	5,59
Média			<b>3,57</b>			18,02

Fonte: própria.

A Tabela 28 mostra que a combinação  $KS_{x,fo,k}$  apresenta melhores soluções para 7 instâncias e a diferença percentual média encontrada também é inferior à encontrada pela melhor combinação definida sem a estratégia de adaptação ( $KS_{x,fo}$ ) com uma redução de 2,34%.

Os resultados exibidos na Tabela 29 mostram que para 9 instâncias a combinação  $KS_{y,cr,k}$  obtém melhores resultados, porém a combinação sem a estratégia de adaptação ( $KS_{y,cr}$ ) apresenta um ganho médio melhor, com uma diferença de 4,97% no valor da diferença percentual média com relação ao CPLEX.

A menor diferença percentual média passa a ser, então, de 3,57%, o que representa uma redução média de 7,37%. Estes testes reforçam que para duas das estratégias de seleção analisadas devem ser combinadas com a ordenação  $O_{cr}$ .

Outra estratégia de adaptação explorada é a  $A_b$ , descrita na Subseção 5.2.3. As Tabelas 30-32 exibem resultados da execução do *kernel search* com tempo até 10min das seis estratégias iniciais (Fase 1) aliadas a essa estratégia de adaptação (Fase 2).

A Tabela 30 mostra que, apesar de  $KS_{xy,cr,b}$  apresentar melhores soluções para 6 instâncias

Tabela 28 – Resultados obtidos pelo *kernel search* para a combinação das estratégias  $KS_{x,cr}$  e  $KS_{x,fo}$  à estratégia de adaptação  $A_k$  com tempo máximo de execução do método de 10min.

	$KS_{x,cr,k}$			$KS_{x,fo,k}$		
	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	89482	604	<b>19,90</b>	91495	603	22,60
(24,3,5,10,10)_2	53570	604	10,77	49014	605	<b>1,35</b>
(24,3,5,10,10)_3	87385	605	50,31	66377	602	<b>14,17</b>
(24,3,5,10,10)_4	53346	604	13,55	49010	606	<b>4,32</b>
(24,3,5,10,10)_5	57488	611	<b>12,35</b>	58782	605	14,88
(24,3,5,5,20)_1	80093	604	3,52	79877	604	<b>3,24</b>
(24,3,5,20,5)_1	46575	606	12,94	44066	603	<b>6,86</b>
(24,3,5,20,5)_3	59915	612	<b>20,55</b>	53751	607	8,15
(24,3,5,20,5)_4	37741	606	27,45	34569	604	<b>16,74</b>
(24,3,5,20,5)_5	46521	605	35,63	43736	603	<b>27,51</b>
Média			20,70			<b>11,98</b>

Fonte: própria.

Tabela 29 – Resultados obtidos pelo *kernel search* para a combinação das estratégias  $KS_{y,cr}$  e  $KS_{y,fo}$  à estratégia de adaptação  $A_k$  com tempo máximo de execução do método de 10min.

	$KS_{y,cr,k}$			$KS_{y,fo,k}$		
	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	107522	604	<b>44,07</b>	107522	605	<b>44,07</b>
(24,3,5,10,10)_2	50898	421	<b>5,25</b>	56014	611	15,82
(24,3,5,10,10)_3	99225	370	70,67	69089	605	<b>18,84</b>
(24,3,5,10,10)_4	53847	499	<b>14,61</b>	53847	606	<b>14,61</b>
(24,3,5,10,10)_5	56400	417	<b>10,22</b>	59954	604	17,17
(24,3,5,5,20)_1	81210	611	<b>4,97</b>	94999	605	22,79
(24,3,5,20,5)_1	43749	512	<b>6,09</b>	48462	605	17,51
(24,3,5,20,5)_3	49878	459	<b>0,36</b>	63220	605	27,20
(24,3,5,20,5)_4	35424	409	<b>19,62</b>	42102	605	42,17
(24,3,5,20,5)_5	35179	475	<b>2,57</b>	42027	605	22,53
Média			<b>17,84</b>			24,27

Fonte: própria.

testadas com relação a  $KS_{xy,fo,b}$ , a melhor combinação quando é realizada a seleção  $S_{xy}$  é  $KS_{xy,cr,k}$ .

Tabela 30 – Resultados obtidos pelo *kernel search* para as configurações da Fase 1 para a seleção  $S_{xy}$  combinadas à estratégia de adaptação  $A_b$ .

	$KS_{xy,cr,b}$			$KS_{xy,fo,b}$		
	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	77287	607	<b>3,56</b>	101307	611	35,75
(24,3,5,10,10)_2	48700	608	<b>0,70</b>	57029	607	17,92
(24,3,5,10,10)_3	60253	609	<b>3,64</b>	63527	609	9,27
(24,3,5,10,10)_4	48489	612	<b>3,21</b>	51932	607	10,54
(24,3,5,10,10)_5	51848	607	<b>1,32</b>	67452	610	31,82
(24,3,5,5,20)_1	80969	605	4,65	80598	602	<b>4,17</b>
(24,3,5,20,5)_1	44904	609	<b>8,89</b>	49250	613	19,43
(24,3,5,20,5)_3	60039	607	20,80	55398	605	<b>11,46</b>
(24,3,5,20,5)_4	38289	604	29,30	34705	603	<b>17,20</b>
(24,3,5,20,5)_5	46530	604	35,66	35015	603	<b>2,09</b>
Média			<b>11,17</b>			15,96

Fonte: própria.

Nota-se, pelos resultados apresentados na Tabela 31, que  $KS_{x,fo,k}$  ainda se mantém melhor para a estratégia de seleção  $S_x$ . Neste caso,  $KS_{x,cr,b}$  é melhor que  $KS_{x,fo,b}$  para 6 das 10 instâncias, porém a diferença percentual com relação ao CPLEX é 4,70% pior que quando se utiliza a estratégia de adaptação  $A_k$ .

Tabela 31 – Resultados obtidos pelo *kernel search* para as configurações da Fase 1 para a seleção  $S_x$  combinadas à estratégia de adaptação  $A_b$ .

	$KS_{x,cr,b}$			$KS_{x,fo,b}$		
	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	89482	604	<b>19,90</b>	89696	603	20,19
(24,3,5,10,10)_2	52131	609	7,80	50482	603	<b>4,39</b>
(24,3,5,10,10)_3	67962	605	<b>16,90</b>	68993	605	18,67
(24,3,5,10,10)_4	53292	604	13,43	49777	610	<b>5,95</b>
(24,3,5,10,10)_5	57488	605	<b>12,35</b>	59046	604	15,39
(24,3,5,5,20)_1	80072	604	<b>3,49</b>	80724	604	4,34
(24,3,5,20,5)_1	46330	604	<b>12,35</b>	57981	606	40,60
(24,3,5,20,5)_3	61185	604	23,11	53987	604	<b>8,63</b>
(24,3,5,20,5)_4	36095	605	21,89	34276	614	<b>15,75</b>
(24,3,5,20,5)_5	46521	603	<b>35,63</b>	52045	609	51,74
Média			<b>16,68</b>			18,56

Fonte: própria.

Na Tabela 32 os resultados mostram que  $KS_{y,cr,b}$  encontrou 9 melhores soluções que  $KS_{y,fo,b}$  e que foi a única combinação dos testes realizados na Fase 3 que resultou em melhoria com relação à melhor configuração encontrada até então quando se trata da seleção pela estratégia  $S_y$  ( $KS_{y,cr}$ ).

Tabela 32 – Resultados obtidos pelo *kernel search* para as configurações da Fase 1 para a seleção  $S_y$  combinadas à estratégia de adaptação  $A_b$ .

	$KS_{y,cr,b}$			$KS_{y,fo,b}$		
	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	107522	604	44,07	107510	603	<b>44,06</b>
(24,3,5,10,10)_2	54746	605	<b>13,20</b>	68694	611	42,04
(24,3,5,10,10)_3	66956	604	<b>15,17</b>	68782	603	18,31
(24,3,5,10,10)_4	53847	612	<b>14,61</b>	53847	612	<b>14,61</b>
(24,3,5,10,10)_5	56288	604	<b>10,00</b>	59856	604	16,97
(24,3,5,5,20)_1	81314	604	<b>5,10</b>	94999	605	22,79
(24,3,5,20,5)_1	41933	605	<b>1,68</b>	47369	605	14,86
(24,3,5,20,5)_3	49635	604	<b>-0,13</b>	62709	605	26,18
(24,3,5,20,5)_4	31943	604	<b>7,87</b>	42440	611	43,32
(24,3,5,20,5)_5	37304	605	<b>8,76</b>	42027	607	22,53
Média			<b>12,03</b>			26,57

Fonte: própria.

Neste caso, a menor diferença percentual média encontrada continuou a ser a mesma (3,57%), porém houve redução das diferenças encontradas para a combinação das estratégias de ordenação e adaptação à estratégia de seleção  $S_y$ .

Após tais testes, assim como para a utilização do software CPLEX, aplicou-se a modificação dos parâmetros, ou seja, utilizou-se a calibração manual definida na Seção 4.3 para as três melhores combinações obtidas, que são  $KS_{xy,cr,k}$ ,  $KS_{x,fo,k}$  e  $KS_{y,cr,b}$ . Os resultados são exibidos nas Tabelas 33-35 e mostram que, em geral, a utilização da modificação dos parâmetros reduz os custos.

Tabela 33 – Resultados obtidos pela combinação  $KS_{xy,cr,k}$  sem e com a calibração manual definida na Seção 4.3.

	$KS_{xy,cr,k}$			$KS_{xy,cr,k}^*$		
	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	77287	607	3,56	75702	602	<b>1,44</b>
(24,3,5,10,10)_2	48326	578	<b>-0,07</b>	49388	172	2,12
(24,3,5,10,10)_3	60632	602	4,29	59509	203	<b>2,36</b>
(24,3,5,10,10)_4	49120	607	4,55	45921	608	<b>-2,26</b>
(24,3,5,10,10)_5	51266	528	<b>0,19</b>	51447	308	0,54
(24,3,5,5,20)_1	81600	605	5,47	78618	239	<b>1,62</b>
(24,3,5,20,5)_1	41089	603	<b>-0,36</b>	44188	605	7,15
(24,3,5,20,5)_3	53245	449	7,13	50100	227	<b>0,80</b>
(24,3,5,20,5)_4	31376	256	5,95	30883	242	<b>4,29</b>
(24,3,5,20,5)_5	36001	421	<b>4,96</b>	37446	264	9,18
Média			3,57			<b>2,72</b>

Fonte: própria.

As três melhores combinações para a heurística foram avaliadas para mais 5 instâncias.

Tabela 34 – Resultados obtidos pela combinação  $KS_{x,fo,k}$  sem e com a calibração manual definida na Seção 4.3.

	$KS_{x,fo,k}$			$KS_{x,fo,k}^*$		
	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	91495	603	22,60	80830	604	<b>8,31</b>
(24,3,5,10,10)_2	49014	605	<b>1,35</b>	50680	604	4,80
(24,3,5,10,10)_3	66377	602	<b>14,17</b>	67892	604	16,78
(24,3,5,10,10)_4	49010	606	<b>4,32</b>	54273	605	15,52
(24,3,5,10,10)_5	58782	605	14,88	57684	601	<b>12,73</b>
(24,3,5,5,20)_1	79877	604	<b>3,24</b>	81808	604	5,74
(24,3,5,20,5)_1	44066	603	<b>6,86</b>	45691	605	10,80
(24,3,5,20,5)_3	53751	607	8,15	53406	603	<b>7,46</b>
(24,3,5,20,5)_4	34569	604	16,74	33952	605	<b>14,65</b>
(24,3,5,20,5)_5	43736	603	27,51	35206	605	<b>2,64</b>
Média			11,98			<b>9,94</b>

Fonte: própria.

Tabela 35 – Resultados obtidos pela combinação  $KS_{y,cr,b}$  sem e com a calibração manual definida na Seção 4.3.

	$KS_{y,cr,b}$			$KS_{y,cr,b}^*$		
	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	107522	604	44,07	107510	604	<b>44,06</b>
(24,3,5,10,10)_2	54746	605	13,20	49646	604	<b>2,66</b>
(24,3,5,10,10)_3	66956	604	15,17	62678	605	<b>7,81</b>
(24,3,5,10,10)_4	53847	612	<b>14,61</b>	53847	604	<b>14,61</b>
(24,3,5,10,10)_5	56288	604	<b>10,00</b>	65910	602	28,81
(24,3,5,5,20)_1	81314	604	<b>5,10</b>	81426	601	5,25
(24,3,5,20,5)_1	41933	605	<b>1,68</b>	43206	605	4,77
(24,3,5,20,5)_3	49635	604	<b>-0,13</b>	50694	604	2,00
(24,3,5,20,5)_4	31943	604	7,87	31030	605	<b>4,79</b>
(24,3,5,20,5)_5	37304	605	<b>8,76</b>	37531	605	9,42
Média			<b>12,03</b>			12,42

Fonte: própria.

Os resultados são exibidos na Tabela 36 (executando o *kernel search* por 10 min). Os resultados obtidos corroboram que a melhor configuração é a  $KS_{xy,cr,k}^*$ , que apresenta uma diferença percentual com relação ao CPLEX de apenas 2,09% e melhores soluções que  $KS_{x,fo,b}^*$  e que  $KS_{y,cr,b}$  para 11 das 15 instâncias testadas.

Tabela 36 – Resultados obtidos para 15 instâncias de calibração utilizando as 3 melhores estratégias para o *kernel search* com no máximo 10min de execução.

	$KS_{xy,cr,k}^*$			$KS_{x,fo,b}^*$			$KS_{y,cr,b}$		
	Sol	Ti	D_30	Sol	Ti	D_30	Sol	Ti	D_30
(24,3,5,10,10)_1	75702	602	<b>1,44</b>	80830	604	8,31	107522	604	44,07
(24,3,5,10,10)_2	49388	172	<b>2,12</b>	50680	604	4,80	54746	605	13,20
(24,3,5,10,10)_3	59509	203	<b>2,36</b>	67892	604	16,78	66956	604	15,17
(24,3,5,10,10)_4	45921	608	<b>-2,26</b>	54273	605	15,52	53847	612	14,61
(24,3,5,10,10)_5	51447	308	<b>0,54</b>	57684	601	12,73	56288	604	10,00
(24,3,5,5,20)_1	78618	239	<b>1,62</b>	81808	604	5,74	81314	604	5,10
(24,3,5,5,20)_2	81862	174	<b>-0,41</b>	82715	604	0,63	82536	603	0,41
(24,3,5,5,20)_3	90691	166	2,83	91158	603	3,36	88630	605	<b>0,49</b>
(24,3,5,5,20)_4	54415	274	<b>0,09</b>	59754	605	9,91	55017	606	1,20
(24,3,5,5,20)_5	52118	291	<b>0,59</b>	61320	604	18,35	52358	605	1,05
(24,3,5,20,5)_1	44188	605	7,15	45691	605	10,80	41933	605	<b>1,68</b>
(24,3,5,20,5)_2	31994	252	<b>0,98</b>	35046	604	10,62	32851	604	3,69
(24,3,5,20,5)_3	50100	227	0,80	53406	603	7,46	49635	604	<b>-0,13</b>
(24,3,5,20,5)_4	30883	242	<b>4,29</b>	33952	605	14,65	31943	604	7,87
(24,3,5,20,5)_5	37446	264	9,18	35206	605	<b>2,64</b>	37304	605	8,76
Média			<b>2,09</b>			9,49			8,48

Fonte: própria.

## 6.2 Análise dos cenários

Como visto na Seção 6.1, a melhor configuração é a  $KS_{xy,cr,k}^*$ , ou seja, com a ordenação das variáveis primeiramente pelo custo relativo e pelo custo na função objetivo, em caso de empate ( $O_{cr}$ ); com a construção do *kernel* inicial selecionando variáveis dos conjuntos  $\chi$  e  $Y$  estritamente positivas na solução da relaxação linear ( $S_{xy}$ ); não inserindo a restrição II do Passo 10 do algoritmo (que afirma que pelo menos uma das variáveis do *bucket* analisado deve ser positiva na solução buscada) caso a solução do subproblema atual não seja encontrada ou seja infactível; e utilizando a mudança de parâmetros segundo a calibração manual da Seção 4.3.

As Tabelas 37 - 39 apresentam os resultados obtidos considerando um tempo máximo de execução de 10min e de 30min para os 3 cenários definidos na Seção 4.1. Nas tabelas,  $D_{30}^*$  representa a diferença percentual entre a solução obtida e o resultado obtido pelo CPLEX com a calibração manual apresentada na Seção 4.3. Vale ressaltar que apenas para a instância (24, 3, 5, 5, 10)\_2 o resultado do CPLEX em 30min de execução com a parametrização foi ótimo.

A Tabela 37 apresenta os resultados obtidos para o cenário com mesma proporção de entregas e de coletas. Quando o tempo máximo foi de 10min a diferença percentual média com relação à solução do CPLEX com a configuração padrão é de 33,19% enquanto que com relação à encontrada quando há calibração é de 37,52%. Apesar das médias altas, foram encontradas 2 soluções de qualidade melhor que o CPLEX sem calibrar e 1 solução de qualidade melhor que o CPLEX calibrado, porém para 15 das 40 instâncias as diferenças  $D_{30}$  e para 18 casos as  $D_{30}^*$  estão acima de 50%.

Ainda na Tabela 37, quando o tempo limite da heurística foi de 30min as diferenças percentuais médias passaram a ser de 20,22% e 24,34% (ou seja, reduções de 22,97% e 23,18%, respectivamente). Existem 13 e 6 soluções de qualidade melhor que as soluções do CPLEX sem e com a calibração manual, respectivamente. Quanto às diferenças percentuais de pior qualidade (maiores que 50%), são 9 quando não há mudança dos parâmetros e 10 quando há.

A Tabela 38 apresenta os resultados obtidos para o cenário que contém mais entregas do que coletas. Quando o tempo máximo foi de 10min a diferença percentual média com relação à solução do CPLEX com a configuração padrão é de 8,99% enquanto que com relação à encontrada quando há calibração é de 12,76%, uma melhoria significativa quando comparado aos valores obtidos no Cenário 1. Foram encontradas 7 soluções de qualidade melhor ou igual ao CPLEX com a configuração padrão e 3 soluções de qualidade melhor que o CPLEX calibrado. Além disso, apenas para 2 das 40 instâncias as diferenças  $D_{30}$  e  $D_{30}^*$  estão acima de 50%.

Quando o tempo limite foi de 30min as diferenças percentuais médias passaram a ser de 2,78% e 6,17% (ou seja, reduções de 6,21% e 6,59%, respectivamente). Existem 13 e 6 soluções de qualidade melhor que o CPLEX sem e com a calibração manual. Quanto às diferenças percentuais, os piores valores encontrados foram de 29,77% para  $D_{30}$  e de 45,11% para  $D_{30}^*$ .

A Tabela 39 resume os resultados obtidos para o cenário que contém poucas entregas e muitas coletas. Quando o tempo máximo foi de 10min a diferença percentual média com relação à solução do CPLEX sem calibrar é de 5,89% enquanto que com relação à encontrada quando há calibração é de 6,58%, os melhores valores quando comparados os 3 cenários. O método encontrou 4 soluções de qualidade melhor ou igual a do CPLEX sem calibrar e 1 solução de qualidade igual a do CPLEX calibrado. Todas as instâncias possuem  $D_{30}$  e  $D_{30}^*$  abaixo de 35%.

Quando o tempo limite foi de 30min as diferenças percentuais médias passaram a ser de 1,87% e 2,52% (ou seja, reduções de 4,02% e 4,08%, respectivamente). Existem 10 e 3 soluções de qualidade melhor ou igual à encontrada pelo CPLEX sem e com a calibração manual. Quanto às diferenças percentuais, existem apenas 3 valores de  $D_{30}$  e 5 de  $D_{30}^*$  acima de 5%.

Nota-se, pelos resultados obtidos, que o Cenário 1 representa os problemas de solução mais difícil para a heurística *kernel search*, havendo casos em que a solução chega a estar a



Tabela 37 – Resultados do *kernel search* com tempo limite de 10min e de 30min para o Cenário 1 (mesma proporção de entregas e coletas).

	$KS_{xy,cr,k}^* - 10min$				$KS_{xy,cr,k}^* - 30min$			
	Sol_10	T	D_30	D_30*	Sol_30	T	D_30	D_30*
(24,3,5,10,10)_1	75702	602	1,44	5,82	74282	1460	<b>-0,47</b>	3,83
(24,3,5,10,10)_2	49388	172	2,12	2,95	48914	499	1,14	1,97
(24,3,5,10,10)_3	59509	203	2,36	2,39	59349	507	2,08	2,11
(24,3,5,10,10)_4	45921	608	<b>-2,26</b>	1,73	45035	508	<b>-4,14</b>	<b>-0,23</b>
(24,3,5,10,10)_5	51447	304	0,54	3,78	51397	956	0,44	3,68
(24,3,5,20,20)_1	180061	609	56,59	57,50	120704	1314	4,97	5,58
(24,3,5,20,20)_2	115348	602	18,21	20,14	98173	1810	0,61	2,25
(24,3,5,20,20)_3	94435	610	8,07	10,80	85157	1811	<b>-2,55</b>	<b>-0,09</b>
(24,3,5,20,20)_4	101969	603	14,57	13,68	89641	1809	0,72	<b>-0,06</b>
(24,3,5,20,20)_5	85465	608	3,48	5,13	81556	1808	<b>-1,25</b>	0,32
(24,4,5,10,10)_1	36934	603	4,47	4,30	34816	1802	<b>-1,52</b>	<b>-1,69</b>
(24,4,5,10,10)_2	57165	210	5,28	10,58	56129	532	3,37	8,58
(24,4,5,10,10)_3	52860	603	1,53	6,44	49186	1802	<b>-5,53</b>	<b>-0,96</b>
(24,4,5,10,10)_4	65242	238	<b>-1,79</b>	1,57	64898	565	<b>-2,31</b>	1,04
(24,4,5,10,10)_5	47810	603	1,30	<b>-0,78</b>	47127	1807	<b>-0,15</b>	<b>-2,20</b>
(24,4,5,20,20)_1	83546	601	11,13	14,38	77934	1809	3,66	6,70
(24,4,5,20,20)_2	132333	602	57,29	57,92	99802	1808	18,63	19,10
(24,4,5,20,20)_3	105994	602	17,41	18,85	100903	1808	11,78	13,14
(24,4,5,20,20)_4	176728	609	70,77	72,96	123259	1808	19,10	20,63
(24,4,5,20,20)_5	155059	602	50,27	57,32	155059	1809	50,27	57,32
(24,3,10,10,10)_1	102625	602	2,87	9,21	95491	1803	<b>-4,28</b>	1,62
(24,3,10,10,10)_2	137087	296	13,43	18,74	127217	717	5,26	10,19
(24,3,10,10,10)_3	107311	263	8,22	7,49	100541	541	1,39	0,71
(24,3,10,10,10)_4	198954	601	77,66	78,59	114637	545	2,37	2,90
(24,3,10,10,10)_5	155616	602	60,83	62,41	96178	513	<b>-0,60</b>	0,38
(24,3,10,20,20)_1	299764	601	66,16	70,87	299764	1816	66,16	70,87
(24,3,10,20,20)_2	268208	602	61,45	64,20	268208	1810	61,45	64,20
(24,3,10,20,20)_3	372054	602	113,89	116,10	222429	1809	27,87	29,19
(24,3,10,20,20)_4	354371	601	95,53	99,75	354371	1814	95,53	99,75
(24,3,10,20,20)_5	262998	602	64,92	59,75	262998	1810	64,92	59,75
(24,4,10,10,10)_1	188873	602	48,32	58,47	151332	1802	18,84	26,97
(24,4,10,10,10)_2	114623	601	7,83	16,74	103451	1812	<b>-2,68</b>	5,36
(24,4,10,10,10)_3	117230	496	1,83	10,44	109644	1492	<b>-4,76</b>	3,29
(24,4,10,10,10)_4	92651	519	0,36	6,57	89854	712	<b>-2,67</b>	3,35
(24,4,10,10,10)_5	181986	573	33,14	39,34	181986	1196	33,14	39,34
(24,4,10,20,20)_1	278672	603	44,33	68,71	278672	1812	44,33	68,71
(24,4,10,20,20)_2	247058	603	70,80	68,68	247058	1802	70,80	68,68
(24,4,10,20,20)_3	325152	602	89,80	126,47	325152	1801	89,80	126,47
(24,4,10,20,20)_4	279771	602	76,55	78,71	279771	1809	76,55	78,71
(24,4,10,20,20)_5	286339	603	66,72	72,23	286339	1802	66,72	72,23
<b>Média</b>			33,19	37,52			20,22	24,34
<b>Mínimo</b>			-2,26	-0,78			-5,53	-2,20
<b>Máximo</b>			113,89	126,47			95,53	126,47

Fonte: própria.

Tabela 38 – Resultados do *kernel search* com tempo limite de 10min e de 30min para o Cenário 2 (muitas entregas e poucas coletas).

	$KS_{xy,cr,k}^* - 10min$				$KS_{xy,cr,k}^* - 30min$			
	Sol_10	T	D_30	D_30*	Sol_30	T	D_30	D_30*
(24,3,5,10,5)_1	50627	611	1,14	1,14	50093	1222	0,08	0,08
(24,3,5,10,5)_2	43410	609	10,05	10,60	40948	646	3,81	4,33
(24,3,5,10,5)_3	47056	239	6,91	6,91	47056	638	6,91	6,91
(24,3,5,10,5)_4	34196	607	<b>-0,63</b>	<b>-0,40</b>	34196	1803	-0,63	<b>-0,40</b>
(24,3,5,10,5)_5	32354	151	2,37	3,18	32354	352	2,37	3,18
(24,3,5,20,5)_1	44188	605	7,15	10,16	44188	1812	7,15	10,16
(24,3,5,20,5)_2	31994	267	0,98	3,19	31824	1442	0,45	2,64
(24,3,5,20,5)_3	50100	228	0,80	6,75	50226	530	1,06	7,01
(24,3,5,20,5)_4	30883	217	4,29	4,29	30663	578	3,55	3,55
(24,3,5,20,5)_5	37446	262	9,18	7,85	36168	731	5,45	4,17
(24,4,5,10,5)_1	34431	481	<b>0,00</b>	0,09	34431	705	<b>0,00</b>	0,09
(24,4,5,10,5)_2	34019	257	7,26	8,01	31297	540	<b>-1,32</b>	<b>-0,63</b>
(24,4,5,10,5)_3	53547	275	<b>-1,46</b>	<b>-0,75</b>	53547	474	<b>-1,46</b>	<b>-0,75</b>
(24,4,5,10,5)_4	39182	215	<b>0,00</b>	1,08	39182	419	<b>0,00</b>	1,08
(24,4,5,10,5)_5	50687	541	8,83	9,18	50214	684	7,82	8,16
(24,4,5,20,5)_1	24992	309	<b>-5,41</b>	<b>-2,34</b>	26451	630	0,11	3,36
(24,4,5,20,5)_2	39308	601	8,95	8,99	38175	999	5,81	5,85
(24,4,5,20,5)_3	35562	344	8,85	8,38	33861	486	3,64	3,20
(24,4,5,20,5)_4	37297	604	16,75	20,85	33035	1813	3,41	7,04
(24,4,5,20,5)_5	21099	218	<b>-14,40</b>	1,25	20890	733	<b>-15,24</b>	0,24
(24,3,10,10,5)_1	59790	325	0,81	0,93	59286	400	<b>-0,04</b>	0,08
(24,3,10,10,5)_2	89543	603	1,01	1,78	88761	1809	0,13	0,89
(24,3,10,10,5)_3	114583	269	2,48	2,61	112474	670	0,59	0,72
(24,3,10,10,5)_4	86118	603	7,38	11,37	80955	1186	0,95	4,70
(24,3,10,10,5)_5	66243	269	<b>-0,24</b>	1,44	67043	1795	0,96	2,66
(24,3,10,20,5)_1	66791	272	3,97	7,07	65872	418	2,54	5,60
(24,3,10,20,5)_2	70243	603	1,91	4,34	69822	720	1,30	3,72
(24,3,10,20,5)_3	93228	603	15,73	21,43	78340	849	<b>-2,75</b>	2,04
(24,3,10,20,5)_4	93557	604	0,44	13,68	87912	1814	<b>-5,62</b>	6,82
(24,3,10,20,5)_5	107979	603	29,82	40,76	83939	874	0,91	9,42
(24,4,10,10,5)_1	62453	603	6,06	8,40	58552	1803	<b>-0,56</b>	1,63
(24,4,10,10,5)_2	62568	352	5,00	4,77	61299	739	2,87	2,64
(24,4,10,10,5)_3	74323	355	5,36	9,10	72884	750	3,32	6,99
(24,4,10,10,5)_4	86764	476	17,27	18,95	82780	992	11,88	13,49
(24,4,10,10,5)_5	96864	603	53,77	61,63	70073	811	11,24	16,93
(24,4,10,20,5)_1	88711	601	27,86	42,97	90037	1801	29,77	45,11
(24,4,10,20,5)_2	93968	604	50,79	60,82	71216	1804	14,28	21,88
(24,4,10,20,5)_3	108325	604	28,63	47,59	81193	1804	<b>-3,59</b>	10,62
(24,4,10,20,5)_4	72102	602	15,40	21,46	66383	1803	6,25	11,83
(24,4,10,20,5)_5	67758	604	14,55	21,02	61504	1804	3,97	9,85
<b>Média</b>			8,99	12,76			2,78	6,17
<b>Mínimo</b>			-14,40	-2,34			-15,24	-0,75
<b>Máximo</b>			53,77	61,63			29,77	45,11

Fonte: própria.

Tabela 39 – Resultados do *kernel search* com tempo limite de 10min e de 30min para o Cenário 3 (poucas entregas e muitas coletas).

	$KS_{xy,cr,k}^* - 10min$				$KS_{xy,cr,k}^* - 30min$			
	Sol_10	T	D_30	D_30*	Sol_30	T	D_30	D_30*
(24,3,5,5,10)_1	85091	130	0,79	0,79	85091	336	0,79	0,79
(24,3,5,5,10)_2	48630	102	<b>0,00</b>	<b>0,00</b>	48630	166	<b>0,00</b>	<b>0,00</b>
(24,3,5,5,10)_3	53889	315	<b>-0,21</b>	0,41	53889	926	<b>-0,21</b>	0,41
(24,3,5,5,10)_4	79906	157	0,42	0,42	79906	158	0,42	0,42
(24,3,5,5,10)_5	39191	160	<b>-1,02</b>	0,07	39191	370	<b>-1,02</b>	0,07
(24,3,5,5,20)_1	78747	604	1,78	2,02	78320	1810	1,23	1,46
(24,3,5,5,20)_2	81862	176	<b>-0,41</b>	0,29	81862	332	<b>-0,41</b>	0,29
(24,3,5,5,20)_3	90691	167	2,83	2,94	90691	315	2,83	2,94
(24,3,5,5,20)_4	54667	276	0,56	1,22	54079	747	<b>-0,52</b>	0,13
(24,3,5,5,20)_5	52118	287	0,59	0,59	52118	442	0,59	0,59
(24,4,5,5,10)_1	64495	602	6,16	6,25	64282	1812	5,81	5,90
(24,4,5,5,10)_2	49382	603	2,53	2,50	49382	1808	2,53	2,50
(24,4,5,5,10)_3	57850	215	0,65	1,17	56939	458	<b>-0,93</b>	<b>-0,42</b>
(24,4,5,5,10)_4	57391	256	3,48	4,19	56522	665	1,92	2,61
(24,4,5,5,10)_5	55877	219	0,18	0,18	55777	399	0,00	0,00
(24,4,5,5,20)_1	50616	353	3,13	2,07	49727	682	1,31	0,28
(24,4,5,5,20)_2	71561	603	15,78	15,98	71143	1810	15,11	15,30
(24,4,5,5,20)_3	65348	382	12,11	12,79	57870	798	<b>-0,72</b>	<b>-0,12</b>
(24,4,5,5,20)_4	51267	240	3,63	4,49	50346	780	1,77	2,61
(24,4,5,5,20)_5	58610	603	4,25	4,06	57129	1615	1,62	1,43
(24,3,10,5,10)_1	107232	268	23,07	24,58	87729	443	0,68	1,92
(24,3,10,5,10)_2	181027	205	1,43	1,40	178715	468	0,13	0,11
(24,3,10,5,10)_3	80678	190	0,19	1,42	80353	354	<b>-0,21</b>	1,01
(24,3,10,5,10)_4	147282	288	0,49	0,49	146566	474	0,00	0,00
(24,3,10,5,10)_5	158133	224	1,47	1,56	156638	734	0,51	0,60
(24,3,10,5,20)_1	91320	554	3,54	4,90	88752	1811	0,63	1,95
(24,3,10,5,20)_2	104997	277	3,02	5,92	100270	381	<b>-1,62</b>	1,16
(24,3,10,5,20)_3	141585	604	5,25	5,85	135245	1360	0,54	1,11
(24,3,10,5,20)_4	116165	571	6,01	5,77	113760	588	3,81	3,58
(24,3,10,5,20)_5	166862	292	3,54	3,58	166465	400	3,30	3,33
(24,4,10,5,10)_1	95199	602	7,36	8,29	91815	1803	3,54	4,44
(24,4,10,5,10)_2	132819	536	9,43	12,19	122964	1802	1,32	3,87
(24,4,10,5,10)_3	88173	387	32,24	34,72	71980	599	7,95	9,98
(24,4,10,5,10)_4	87814	428	16,66	16,75	76545	616	1,69	1,76
(24,4,10,5,10)_5	73642	287	3,44	3,68	71848	448	0,92	1,16
(24,4,10,5,20)_1	83310	603	16,52	20,36	74919	1388	4,78	8,24
(24,4,10,5,20)_2	83394	605	7,13	7,38	80981	846	4,03	4,27
(24,4,10,5,20)_3	122833	603	24,77	21,75	110053	1534	11,79	9,08
(24,4,10,5,20)_4	64751	493	3,83	5,06	62994	710	1,01	2,21
(24,4,10,5,20)_5	101424	591	8,78	15,08	91347	1809	<b>-2,02</b>	3,65
<b>Média</b>			5,89	6,58			1,87	2,52
<b>Mínimo</b>			-1,02	0,00			-2,02	-0,42
<b>Máximo</b>			32,24	34,72			15,11	15,30

Fonte: própria.

mais de 100% da solução encontrada pelo software. Já os Cenários 2 e 3 tiveram resultados com diferenças percentuais de melhor qualidade indicando que o método é competitivo com o software CPLEX, em especial para as instâncias do Cenário 3.

---

## CONCLUSÕES E PESQUISAS FUTURAS

---

O problema de *cross-docking* é de grande relevância econômica, uma vez que tem sido utilizado na área de logística e transportes visando a redução dos custos operacionais que, em geral, tendem a ser muito altos. Essa dissertação apresenta uma abordagem do problema considerando várias características em conjunto que, em geral, são tratadas separadamente na literatura; foi gerado um conjunto de instâncias *benchmark*; e foi proposto um método heurístico para a resolução do problema proposto.

A primeira etapa desse trabalho contou com a revisão da literatura na qual o objeto de estudo se encaixava (o problema de integração do planejamento e da programação de uma rede de distribuição) foi descrito e evidenciou-se a falta de um modelo na literatura que reunisse todas as características avaliadas individualmente nos modelos já existentes. Desta forma, foi desenvolvido um modelo para cobrir esta lacuna de pesquisa. O modelo busca se aproximar do problema real, unindo a ideia da existência de múltiplos *cross-docks*, possibilidade de transferência de produtos entre eles e janelas de tempo desejáveis e obrigatórias para as coletas. Desta primeira etapa foi redigido o artigo publicado no "XLVII Simpósio Brasileiro de Pesquisa Operacional", em 2015.

Para os testes computacionais, uma vez que não se conseguiu acesso aos dados da literatura, optou-se por gerar instâncias que resultaram num *benchmark*, que também está disponível para a comunidade científica.

Na análise computacional do modelo, o modelo proposto foi comparado com um modelo da literatura (Javanmard, Vahdani e Tavakkoli-Moghaddam (2014)). A partir dos resultados obtidos, nota-se que a transferência de produtos entre *cross-docks* pode reduzir custos, uma vez que facilita a composição de cargas para coleta. Quanto às janelas de tempo flexíveis para coletas, além delas serem importantes do ponto de vista prático, a penalidade associada à sua violação reduz o número de soluções alternativas, facilitando a obtenção de soluções de melhor qualidade.

Após verificar que a nova proposta de modelo pode resultar em redução de custos operacionais, foi estudada a possibilidade de calibração do software utilizado, o ILOG CPLEX,

com o intuito de buscar soluções de melhor qualidade. Esta ideia tem sido utilizada com grande frequência para buscar a melhor configuração do software na resolução de cada tipo de problema, uma vez que pode ocorrer que a configuração padrão do CPLEX pode não ser a mais indicada para a resolução do problema estudado. Nesta etapa, notou-se que a calibração manual, uma vez que conta com um conhecimento prévio de algumas características do problema pode retornar soluções de melhor qualidade que ao utilizar a calibração automática do software.

Propôs-se, então, para a resolução do problema, um método heurístico *Kernel search*. O método teve de ser adaptado ao problema de *cross-docking* e seus parâmetros calibrados como apresentado no Capítulo 5. Os resultados computacionais apresentados no Capítulo 6 mostram que para os cenários em que entregas e coletas ocorrem em proporções diferentes o método é competitivo com o software CPLEX calibrado para o problema de *cross-docking*, havendo casos em que a heurística em 10min encontra soluções melhores que o CPLEX em 30min de execução.

De forma geral, foi proposto um modelo matemático; foram criadas instâncias *benchmark* para posterior uso da comunidade; foi realizado o estudo de calibração do software buscando por soluções de melhor qualidade; e, por fim, foi proposta uma heurística para a resolução do problema de *cross-docking* através do modelo proposto.

Como perspectiva para trabalhos futuros, sugere-se adaptar o modelo para aproximá-lo ainda mais da realidade, uma vez que cada vez mais as empresas estão utilizando essa estratégia de distribuição. Outra sugestão é que instâncias baseadas em dados reais sejam avaliadas, para posterior comparação às estratégias utilizadas na prática. Por fim, os resultados indicam que a heurística *Kernel search* ainda poderia ser avaliada com outras estratégias em busca de resultados melhores, em especial para situações em que o número de coletas e entregas sejam muito próximos.

## REFERÊNCIAS

---

---

- ANGELELLI, E.; MANSINI, R.; SPERANZA, M. G. Kernel search: A general heuristic for the multi-dimensional knapsack problem. **Computers and Operational Research**, v. 37, p. 2017–2026, 2010. Citado 4 vezes nas páginas 17, 69, 70 e 71.
- BELLE, J. V.; VALCKENAERS, P.; CATTRYSSSE, D. Cross-docking: State of the art. **Omega**, v. 40, n. 6, p. 827 – 846, 2012. Citado 4 vezes nas páginas 26, 29, 30 e 32.
- BOYSEN, N.; FLIEDNER, M. Cross dock scheduling: Classification, literature review and research agenda. **Omega**, v. 38, n. 6, p. 413–422, 2010. Citado na página 32.
- BUIJS, P.; VIS, I. F.; CARLO, H. J. Synchronization in cross-docking networks: A research classification and framework. **European Journal of Operational Research**, v. 239, n. 3, p. 593–608, 2014. Citado 5 vezes nas páginas 25, 26, 31, 32 e 39.
- CARGOBR. **Cross-Docking - Inovação na logística**. 2013. Disponível em: <<http://cargobr.com/blog/cross-docking-inovacao-na-logistica-cargobr/>>. Acesso em: 10/02/2016. Citado na página 39.
- CARRARO, L. e. T. **Cross-Docking**. 2008. Disponível em: <[http://carrarologistica.com.br/serv\\_cross.html](http://carrarologistica.com.br/serv_cross.html)>. Acesso em: 10/02/2016. Citado 2 vezes nas páginas 30 e 39.
- CHEN, P.; GUO, Y.; LIM, A.; RODRIGUES, B. Multiple crossdocks with inventory and time windows. **Computers & operations research**, v. 33, n. 1, p. 43–63, 2006. Citado 6 vezes nas páginas 27, 32, 33, 34, 47 e 48.
- COVISI, T. **Conceitos e principais objetivos do Cross Docking**. 2016. Disponível em: <[http://covisi.com.br/site/index.php?option=com\\_content&view=article&id=54&Itemid=63](http://covisi.com.br/site/index.php?option=com_content&view=article&id=54&Itemid=63)>. Acesso em: 10/02/2016. Citado 2 vezes nas páginas 30 e 39.
- GHIANI, G.; LAPORTE, G.; MUSMANNO, R. **Introduction to logistics systems planning and control**. [S.l.]: John Wiley & Sons, 2004. Citado na página 26.
- GUASTAROBA, G.; SPERANZA, M. G. A heuristic for blip problems: The single source capacitated facility location problem. **European Journal of Operational Research**, v. 238, p. 438 – 450, 2014. Citado na página 69.
- JAVANMARD, S.; VAHDANI, B.; TAVAKKOLI-MOGHADDAM, R. Solving a multi-product distribution planning problem in cross docking networks: An imperialist competitive algorithm. **The International Journal of Advanced Manufacturing Technology**, v. 70, n. 9-12, p. 1709–1720, 2014. Citado 10 vezes nas páginas 15, 27, 34, 39, 40, 45, 48, 53, 54 e 91.
- LADIER, A.-L.; ALPAN, G. Cross-docking operations: Current research versus industry practice. **Omega**, Elsevier, 2015. Disponível em: <<http://dx.doi.org/10.1016/j.omega.2015.09.006>>. Citado 2 vezes nas páginas 30 e 32.

- LIM, A.; MIAO, Z.; RODRIGUES, B.; XU, Z. Transshipment through crossdocks with inventory and time windows. **Naval Research Logistics (NRL)**, v. 52, n. 8, p. 724–733, 2005. Citado 3 vezes nas páginas 27, 32 e 33.
- MA, H.; MIAO, Z.; LIM, A.; RODRIGUES, B. Crossdocking distribution networks with setup cost and time window constraint. **Omega**, v. 39, n. 1, p. 64–72, 2011. Citado 3 vezes nas páginas 27, 33 e 34.
- MARJANI, M. R.; HUSSEINI, S. M. M.; KARIMI, B. Bi-objective heuristics for multi-item freights distribution planning problem in crossdocking networks. **The International Journal of Advanced Manufacturing Technology**, v. 58, n. 9-12, p. 1201–1216, 2012. Citado 2 vezes nas páginas 34 e 44.
- MARONI, G. **Armazenagem e Cross docking**. 2013. Disponível em: <<http://transmaroni.com.br/armazenagem-e-cross-docking/>>. Acesso em: 10/02/2016. Citado 2 vezes nas páginas 30 e 39.
- MIAO, Z.; YANG, F.; FU, K.; XU, D. Transshipment service through crossdocks with both soft and hard time windows. **Annals of Operations Research**, v. 192, n. 1, p. 21–47, 2012. Citado 6 vezes nas páginas 27, 33, 42, 43, 45 e 48.
- MIRANDA, S. **Cross Docking and Supply Chain Success**. 2015. Disponível em: <<http://usacrossdocking.com/cross-docking-and-supply-chain-success/>>. Acesso em: 10/02/2016. Citado na página 39.
- MIYAZAKI-TENÓRIO, P. a. S.; TOLEDO, F. M.; ARMENTANO, V. A. Um modelo para o problema de programação de múltiplos *cross-docks*. **XLVII SBPO**, 2015. Citado na página 27.
- MIYAZAKI-TENÓRIO, P. S.; TOLEDO, F. M. **Benchmark**. 2016. Disponível em: <[https://www.dropbox.com/sh/8ot2agt5i3rm175/AAAiww0G4TPs\\_nrj9YzYoMPKa?dl=0](https://www.dropbox.com/sh/8ot2agt5i3rm175/AAAiww0G4TPs_nrj9YzYoMPKa?dl=0)>. Acesso em: 15/02/2016. Citado 2 vezes nas páginas 27 e 47.
- SIMCHI-LEVI, D.; KAMINSKY, P.; SIMCHI-LEVI, E. **Designing and managing the supply chain: concepts, strategies, and case studies**. 2. ed. ed. Boston, Mass. [u.a.]: McGraw-Hill/Irwin, 2003. (The McGraw-Hill/Irwin series in operations and decision sciences). Citado na página 39.
- YU, W.; EGBELU, P. J. Scheduling of inbound and outbound trucks in cross docking systems with temporary storage. **European Journal of Operational Research**, v. 184, n. 1, p. 377–396, 2008. Citado na página 29.