

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO E MESTRADO EM ENGENHARIA
MECÂNICA E DE MATERIAIS**

MATHEUS GARIBALDE SOARES DE LIMA

**ALOCAÇÃO DE RECURSOS EM NÍVEL OPERACIONAL COM
INCERTEZA NOS DADOS**

DISSERTAÇÃO

**CURITIBA
2012**

MATHEUS GARIBALDE SOARES DE LIMA

**ALOCAÇÃO DE RECURSOS EM NÍVEL OPERACIONAL COM
INCERTEZA NOS DADOS**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Mecânica e de Materiais da Universidade Tecnológica Federal do Paraná como requisito parcial à obtenção do título de “Mestre em Engenharia” - Área de Concentração em Engenharia de Manufatura.

Orientador: Prof. Luiz Carlos Rodrigues, Dr.

CURITIBA

2012

AGRADECIMENTOS

Primeiramente agradeço a Deus, por sempre iluminar não apenas o meu caminho, mas o de todos. É Dele que nasce toda a força de vontade e amor, a cada dia.

Ao orientador Prof. Dr. Luiz Carlos de Abreu Rodrigues, pela orientação e paciência durante este período. Também pela amizade, carinho e dedicação por acreditar em minha capacidade na execução deste trabalho.

À instituição e ao PPGEM por proporcionar um ensino de qualidade.

À empresa fornecedora do estudo de caso e meus supervisores, que proporcionaram um ambiente propício e construtivo para a elaboração deste trabalho, além da disponibilidade de tempo para o estudo.

À banca examinadora pela oportunidade.

À minha amada Julia R. Vicente, pelo constante incentivo e paciência durante todo o desenvolvimento do presente trabalho.

Aos meus colegas, pelo apoio durante o decorrer do mestrado. Em especial a Alexandre T. Suman, que me apoiou e conduziu com seu extenso conhecimento prático, essencial para a elaboração deste trabalho.

E aos meus amigos e família, pela compreensão, força e amor dedicados.

LIMA, Matheus Garibalde Soares de. Alocação de recursos em nível operacional com incerteza nos dados. 2012. 124f. Dissertação (Mestrado em Engenharia) - Programa de Pós-graduação em Engenharia Mecânica e de Materiais, Universidade Tecnológica Federal do Paraná. Curitiba, 2012.

RESUMO

O estudo tem como finalidade tratar a alocação de recursos no nível operacional com a presença de incertezas. Para isso, foi proposta uma abordagem de otimização usando métodos heurísticos. As soluções de problemas de produção e logística, comumente abordadas em pesquisa operacional, exploram diversos parâmetros dentre os quais o presente estudo considera três como de incerteza: demanda, tempo de execução e indisponibilidade de recursos. Para tal finalidade foi escolhido como estudo de caso a resolução de um problema de logística. O problema consiste na minimização dos custos de operação, na seleção de veículos em uma frota heterogênea, consolidação das cargas para cada cliente e na seleção do tipo de frete utilizado. Quanto ao tipo de frete, são considerados dois, os quais se diferenciam quanto aos ativos envolvidos na produção e ao tipo de prestação de serviço, sendo eles: *i*) frota da empresa com serviço terceirizado; *ii*) frota e serviços totalmente terceirizados. O problema original foi decomposto em duas etapas: *i*) Compartimentalizador e *ii*) Alocador. As duas etapas são solucionadas via a abordagem de Busca Tabu, sendo que a primeira etapa (Compartimentalizador) gera uma lista dos carregamentos factíveis que atenda pedidos de até três clientes distintos. O Alocador se utiliza da lista dos carregamentos factíveis para definir como e quando cada pedido será atendido. Os resultados indicam a viabilidade da adoção desta abordagem para a solução de problemas reais.

Palavras-chave: Pesquisa operacional, heurística, incertezas, alocação de recursos, simulação.

LIMA, Matheus Garibalde Soares de. Impacto da incerteza na alocação de recursos em nível operacional e tático. 2012. 124f. Dissertação (Mestrado em Engenharia) - Programa de Pós-graduação em Engenharia Mecânica e de Materiais, Universidade Tecnológica Federal do Paraná. Curitiba, 2012.

ABSTRACT

The study aims to address the allocation of resources at the operational level under uncertainties. For this reason, it was proposed an optimization approach based on heuristic methods. The resolutions of production and logistics problems, commonly addressed in operational research, explore various parameters among which the present study considers three variables of uncertainty: demand, operation time and resources availability. For this purpose a logistics problem was chosen as study of case. The problem consists in minimizing cost operation, selection of vehicles in a heterogeneous fleet, consolidation of loads for each client and selecting the type of freight payables. Regarding of freight payables types, there are centered in two different tariffs, mainly due to assets and service negotiation, such as: i) fleet controlled by company and service outsource; ii) fleet and service completely outsource. The resolution of the original problem was broke down in two steps: i) Compartmentalizer and ii) Allocator. Both steps are solved through Tabu Search approach; the first step (Compartmentalizer) generates a list of feasible shipments to fulfill orders up to three different customers. The second step, the allocator uses the list of feasible shipments to define how and when each request will be supplied. The results aim the feasibility of assumes this approach in order to solve real problems.

Key words: Operational research, heuristic, uncertainty, generalized assignment problem, simulation.

LISTA DE FIGURAS

Figura 1 - Lacuna entre soluções,.....	12
Figura 2 - Fluxograma Busca Tabu, adaptado de Glover (1990).....	18
Figura 3 - Níveis de Planejamento.....	25
Figura 4 - Arquitetura do problema na implementação.....	27
Figura 5 - Parâmetros dos clientes.....	30
Figura 6 - Parâmetros dos clientes.....	31
Figura 7 - Ordens de serviço	31
Figura 8 - Quadros relacionados a terminais e produtos na implementação.....	32
Figura 9 - Carregamento do tipo Top Loading (TL).....	33
Figura 10 - Carregamento do tipo Bottom Loading (BL)	33
Figura 11 - Quadros de veículos de transporte	37
Figura 12 - Frota de veículos heterogênea – vagões de carga	38
Figura 13 - Frota de veículos heterogênea – compartimentos	38
Figura 14 - Exemplo de um compartimento de 3000 UGL com multimarcas de 100 UGL, podendo ser carregado com até 2600 UGL devido restrição de segurança de 85%	39
Figura 15 - Tarifas FPT versus FT	45
Figura 16 - Quadro de trechos e distâncias utilizado na implementação	45
Figura 17 - Quadro de velocidades utilizado na implementação	46
Figura 18 - Exemplo de zonas z de alto, médio e baixo tráfego	48
Figura 19 - Processo de busca	56
Figura 20 - Etapa 1 –Procedimento de busca	57
Figura 21 - Lógica do método de busca do Compartimentalizador	61
Figura 22 - Lógica da Busca Tabu do Alocador de viagens.....	71
Figura 23 - Distribuição percentual dos módulos	81
Figura 24 - Distribuição de demanda do estudo de caso.....	84
Figura 25 - Distribuição da demanda, cenários 1 e 2.....	96
Figura 26 - Custo diário da operação real executada versus cenários 1 e 6.....	97
Figura 27 - Custo diário da operação real executada versus cenários 6 e 8.....	98
Figura 28 - Programação de entregas do dia 15 de dezembro para os cenários 1 e 4.	100
Figura 29 - Número de viagens compartimentalizadas versus número de pedidos	101
Figura 30 - Função Hash domínio	114
Figura 31 - Função Hash contradomínio.....	114
Figura 32 - Declaração de variáveis composta	115
Figura 33 - Unidade de trabalho.....	116

LISTA DE TABELAS

Tabela 1 - Índices.....	27
Tabela 2 - Parâmetros.....	28
Tabela 3 - Cálculo de disponibilidade de produto.....	36
Tabela 4 - Volume e peso máximo por compartimento.....	38
Tabela 5 - Relação dos índices e parâmetros por forma de pagamento.....	44
Tabela 6 - Cálculo de velocidade de tráfego.....	50
Tabela 7 - Cálculo de velocidade peso.....	50
Tabela 8 - Cálculo de velocidade média.....	51
Tabela 9 - Cálculo final do tempo de viagem.....	51
Tabela 10 - Ordens o1 e o2.....	62
Tabela 11 - Veículo v1.....	62
Tabela 12 - Exemplo de Compartimentalização.....	63
Tabela 13 - Exemplo de Compartimentalização – linhas 1 a 3.....	65
Tabela 14 - Exemplo de Compartimentalização – linhas 4 a 8.....	66
Tabela 15 - Exemplo de Compartimentalização – linhas 9 a 12.....	66
Tabela 16 - Exemplo de Compartimentalização – linhas 13 a 31.....	67
Tabela 17 - Exemplo de Compartimentalização – linhas 32 a 36.....	68
Tabela 18 - Ordens de entrega o1 a o9.....	72
Tabela 19 - Lista de viagens compartimentalizadas factíveis.....	74
Tabela 20 - Exemplo de alocação.....	75
Tabela 21 - Exemplo de alocação – linhas 1 a 4.....	77
Tabela 22 - Exemplo de alocação – linhas 5 a 16.....	78
Tabela 23 - Exemplo de alocação – linhas 17 a 23.....	79
Tabela 24 - Exemplo de alocação – linhas 24 a 33.....	80
Tabela 25 - Módulos de programação.....	81
Tabela 26 - Dados gerais do problema por dia de operação.....	85
Tabela 27 - Clientes.....	87
Tabela 28 - Terminais de carregamento.....	88
Tabela 29 - Ordens de entrega (pedidos).....	88
Tabela 30 - Veículos disponíveis.....	89
Tabela 31 - Custo dos veículos disponíveis.....	89
Tabela 32 - Dados dos compartimentos dos veículos.....	90
Tabela 33 - Distâncias percorridas.....	91
Tabela 34 - Disponibilidade dos veículos.....	92
Tabela 35 - Zona de tráfego dos veículos.....	92
Tabela 36 - Produtos.....	92
Tabela 37 - Disponibilidade de produto.....	92
Tabela 38 - Divisão dos cenários.....	93
Tabela 39 - Número de alterações da data de entrega.....	94
Tabela 40 - Análise de demanda nos cenários 2 e 5.....	95
Tabela 41 - Custo final da operação (kUS\$).....	96
Tabela 42 - Tempo de processamento médio (hh:mm:ss).....	101
Tabela 43 - Unidades de cliente.....	116

Lista de Abreviaturas e Siglas

BL	Veículos com carregamento <i>Bottom Loading</i> (carregam por baixo);
CCGAP	<i>Collectively Capacitated Generalized Assignment Problem</i> (problema de atribuição de capacidade coletiva);
ERP	<i>Enterprise Resource Planning</i> (planejamento de recursos empresariais);
FPT	Frota própria terceirizada;
FT	Frota terceirizada;
GAP	<i>Generalized Assignment Problem</i> (problema generalizado de atribuição);
LP	<i>Linear Programming</i> (programação linear);
MILP	<i>Mixed Integer Linear Programming</i> (programação linear inteira mista);
MINLP	<i>Mixed Integer non-Linear Programming</i> (programação não linear inteira mista);
PO	Pesquisa Operacional;
SQL	<i>Structured Query Language</i> ;
STD	<i>Supply, Transformation and Distribution</i> (Transformação, fornecimento e distribuição);
TL	Veículos com carregamento <i>Top Loading</i> (carregam por cima);
UGL	Unidade de volume para Galões;
VB	<i>Visual Basic</i> .

SUMÁRIO

RESUMO.....	3
ABSTRACT.....	4
LISTA DE FIGURAS.....	5
LISTA DE TABELAS.....	6
LISTA DE ABREVIATURAS E SIGLAS.....	7
1 Introdução.....	10
1.1 Objetivo.....	11
1.1.1 Objetivos específicos.....	12
1.2 Justificativa.....	13
2 Revisão da Literatura.....	14
2.1 Abordagens de Pesquisa Operacional.....	14
2.2 Representação de Problemas de Pesquisa Operacional.....	15
2.3 Busca Tabu.....	16
2.4 Incertezas.....	19
2.5 Literatura sobre o problema proposto.....	21
3 Descrição do Problema.....	24
3.1 Níveis de Planejamento.....	24
3.1.1 Nível estratégico.....	25
3.1.2 Nível tático.....	25
3.1.3 Nível Operacional.....	25
3.1.4 Âmbito do problema.....	26
3.2 Clientes.....	30
3.2.1 Demanda dos clientes.....	31
3.3 Terminais.....	32
3.3.1 Incerteza na demanda - disponibilidade de produto.....	34
3.4 Veículos de transporte.....	36
3.4.1 Restrições de segurança de carregamento.....	37
3.4.2 Compartimentação dos veículos.....	38
3.4.3 Restrição de necessidade de bomba no cliente.....	40
3.4.4 Incerteza de disponibilidade de veículo.....	40
3.5 Modelo de Pagamento de Frete.....	41
3.5.1 Exemplo de diferentes formas de pagamento.....	43
i) FPT - Frota Própria Terceirizada.....	43
ii) FT - Frota Terceirizada.....	43
3.5.1.1 Comparação dos tipos de pagamentos de frete citados.....	43
3.6 <i>Distância e tempo de viagem</i>	45
3.6.1 Incerteza no tempo de execução.....	46
3.7 <i>Incerteza na demanda</i>	51
4 Metodologia.....	54
4.1 Metodologia utilizada.....	54
4.1.1 Etapa 1 - Compartimentalização.....	58
4.1.2 Etapa 2 - Alocador de viagens.....	69
4.2 Arquitetura da implementação.....	80
4.3 Considerações técnicas.....	82
5 Resultados experimentais.....	84
5.1 Dados do problema.....	84
5.1.1 Caso do dia 15 de dezembro de 2011.....	85
5.2 Geração de Cenários.....	93
5.3 Resultados Obtidos.....	93
5.3.1 Demanda.....	94
5.3.2 Custo.....	95
5.3.3 Tempo de processamento.....	100

5.3.4	Apontamentos técnicos	102
6	Conclusão.....	104
7	REFERÊNCIAS BIBLIOGRÁFICAS	106
	APÊNDICES.....	112
	APÊNDICE A – FUNÇÃO DE HASH	113
	APÊNDICE B – VARIÁVEIS VETORIAIS	115
	APÊNDICE C – UNIDADES DE TRABALHO	116
	APÊNDICE D – ROTINA PRINCIPAL DO MÓDULO MCOMPART_TABU.....	117
	APÊNDICE E – ROTINA PRINCIPAL DO MÓDULO MSCHEDULE_TABU	118
	APÊNDICE F – DEMANDA DIÁRIA POR CENÁRIO	119
	APÊNDICE G – CUSTO DIÁRIO POR CENÁRIO	120
	APÊNDICE H – NÚMERO DE PEDIDOS DIÁRIOS POR CENÁRIO	121
	APÊNDICE I – NÚMERO DE ALTERAÇÕES DE PEDIDOS DIÁRIOS POR CENÁRIO	122
	APÊNDICE J – INDISPONIBILIDADE DIÁRIA DE VEÍCULOS POR CENÁRIO.....	123

1 Introdução

Grande parte dos trabalhos realizados na área de pesquisa operacional (PO) busca encontrar soluções cada vez melhores para problemas, também, cada vez mais complexos, conforme revisão literária apresentada no capítulo 2. Entretanto, em nível prático, no momento em que as empresas executam estas soluções, o resultado alcançado nem sempre atinge a eficiência dos resultados teóricos calculados, devido a vários fatores que comumente influenciam a sequência da produção, tais como: variações de demanda, atrasos no tempo de execução, interrupções de trabalho, dentre outros. Estes aspectos criam, assim, uma lacuna entre a solução “ótima” calculada e a executada. O presente trabalho explora o impacto das incertezas nas soluções calculadas, gerando desta forma soluções “mais factíveis” de serem aplicados no nível prático.

O trabalho propõe a utilização de uma modelagem robusta na alocação de recursos heterogêneos de produção e logística considerando a presença de incerteza em três diferentes partes do problema: tempo de execução, disponibilidade de veículo e disponibilidade de produto. Propõe-se, a princípio, a utilização de pesquisa operacional e métodos heurísticos para a resolução do problema.

O caso escolhido para estudo foi uma rede de distribuição logística de uma empresa multinacional, com atuação em diferentes regiões, tendo como foco a minimização dos custos de operação ao selecionar veículos em uma frota heterogênea, considerar a consolidação das cargas de acordo com cada cliente e o tipo de pagamento de frete utilizado. O problema foi solucionado levando em consideração diferentes níveis de decisão e diferenciando entre eles apenas o número de variáveis a serem encontradas.

Segundo Arnold (1996) e Bateman e Snell (1998), na seção 3.1, os problemas de produção e de logística podem ser classificados em três níveis de decisão: estratégico, tático e operacional. O presente trabalho analisa os resultados apenas nos níveis operacional e tático.

No nível operacional o problema consiste na determinação do sequenciamento e alocação de tarefas de uma rede de distribuição logística, em que os produtos são transportados em compartimentos ou lotes, em uma frota

heterogênea, com janelas de tempo para a execução das entregas e com possibilidades de entregas fracionadas. Do mesmo modo, tal problema pode ser aplicado sobre a ótica da manufatura, visto que a abordagem proposta possibilita a simulação do modelo em problemas de alocação de recursos, como linhas ou células de manufatura.

No nível tático são abordadas decisões como a normalização da demanda versus a forma de pagamento do frete. O transporte de produtos foi dividido em diferentes formas de pagamento de frete sendo eles: de acordo com os pagamentos por tarifas mensais, quilômetro rodado e número de carregamentos. Na seção 3.6 é apresentada uma descrição detalhada destas opções. Portanto, como os custos incidentes em cada opção variam, discute-se no nível tático quantos caminhões são necessários para satisfazer a operação restringida pelas diferentes formas de pagamento de frete.

1.1 Objetivo

Dada a demanda dos clientes em um determinado dia, o objetivo do modelo é decidir como alocar os veículos aos clientes, dado que a consolidação de cargas pode ser de entregas completas ou fracionadas (vários clientes); e determinar a quantidade a ser entregue de cada veículo e seus respectivos compartimentos para cada cliente.

Em vista das diversas possibilidades de soluções proporcionadas pela presença de incertezas, tem-se também como objetivo a comparação de diferentes abordagens de otimização, na presença e na ausência destas incertezas. Para tanto, a fim de comparar os resultados, é proposto um estudo de caso. Adicionalmente, para a análise dos resultados do problema são comparados dias distintos de operação considerando possíveis flutuações nas incertezas, já que, num caso real, há diferentes expectativas de mercado e cenários.

A figura 1 mostra uma representação gráfica em nível tático e com valores fictícios do objetivo do trabalho, sendo L1 a diferença entre o resultado de custo da solução prática contra o resultado de custo da solução teórica na ausência de incerteza e L2 é a diferença entre a solução prática contra a solução na presença de incertezas.

Comparando as soluções obtidas, será possível analisar não somente o custo final de cada solução, mas também a confiabilidade e potencial do uso das abordagens propostas na busca de soluções factíveis de serem implementadas num nível prático. Desta forma, pretende-se obter, com detrimento da minimização do custo final obtido, uma maximização da confiabilidade e factibilidade da solução encontrada com a presença de incertezas.

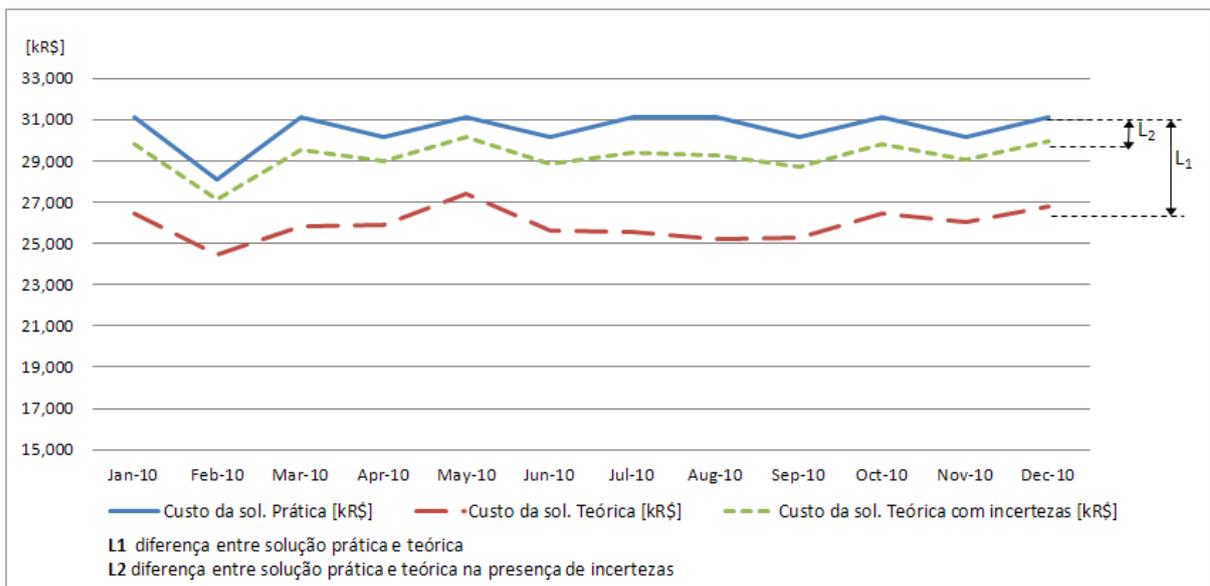


Figura 1 - Lacuna entre soluções,

1.1.1 Objetivos específicos

São propostos como objetivos específicos deste trabalho:

- Implementação de uma abordagem de otimização para a solução do problema proposto com e sem a consideração de incertezas;
- Desenvolvimento da implementação através de programação de computador sem o auxílio de "pacotes comerciais" de otimização;
- Testar as abordagens implementadas para diferentes cenários, comparando as soluções teóricas obtidas com as realizadas na prática sobre critérios de qualidade e custo;

1.2 Justificativa

Segundo a revisão bibliográfica, no capítulo 2, pode-se dizer que a maioria dos esforços realizados na área de otimização, no sequenciamento e na alocação de recursos, está voltada a tratar a complexidade do problema, principalmente, decorrente do aumento do número de restrições, para assim encontrar soluções factíveis, próximas de serem realizadas no nível prático.

O presente trabalho busca explorar uma abordagem ainda pouco investigada na área de manufatura e logística, que é o uso de incertezas no sequenciamento e na alocação de recursos para cenários complexos, deixando assim um espaço de melhoria a ser discutido entre: soluções reais aplicáveis na prática e soluções ótimas complexas encontradas matematicamente, de modo que no decorrer deste estudo haja um maior e mais profundo entendimento a respeito desta lacuna.

Outro ponto adicionado ao estudo é a inserção de diferentes possibilidades de pagamentos na busca de soluções ótimas, permitindo que o trabalho seja estendido para diferentes abordagens no nível operacional. Por último, na área de pesquisa operacional, o tema proposto também visa contribuir com a literatura na busca das soluções utilizando métodos heurísticos.

2 Revisão da Literatura

O problema de alocação de recursos já é bem estudado na literatura (NSAKANDA e PRICE, 2006; CAUX, BRUNIAUX e PIERRAVAL, 2000; MULVEY, VANDERBEI e ZENIOS, 1995; LEVNER, KOGAN e LEVIN, 1995; STERN e VITNER, 1990; RAMAN e RACHAMADUGU, 1986). Foi encontrada literatura combinando a alocação de recursos sujeito a incertezas (XIAOXIA, STACY e CHRISTODOULOS, 2004; TOKTAS e YEN, 2004; SAHINIDIS, 2004; WALLER, SCHOFE e ZILIASKOPOULOS, 2001; ESCUDERO, QUINTANA e SALMAREÓN, 1999; ORÇUN, ALTINEL e HORTAÇSU, 1996). No entanto, o presente trabalho aborda a alocação de recursos sujeito a incertezas, porém de uma forma original.

A revisão da literatura foi dividida em cinco partes: *i)* abordagens na área de PO voltadas para resolução do problema; *ii)* representação de problemas em PO; *iii)* Busca Tabu; *iv)* incertezas; *v)* literatura acerca do problema proposto.

2.1 Abordagens de Pesquisa Operacional

As primeiras contribuições na pesquisa operacional foram extensões dos problemas das máquinas em *flow-shop*, originalmente resolvido por Johnson (1954). O pesquisador tentou resolver este problema considerando o manuseio de materiais, a disponibilidade de equipamentos e a capacidade para *buffers*. Mais tarde, dentre outros autores que contribuíram para a PO, podemos citar Maggu e Kumar (1981), Stern e Vitner (1990) e Levner, Kogan e Levin (1995), os quais realizaram estudos acerca do problema para a minimização do *makespan*. Em paralelo, um problema de tamanho maior de *flow-shop* foi analisado por Raman e Rachamadugu (1986) em um sistema flexível de manufatura.

Todavia, quando se fala de métodos heurísticos para o encontro de soluções, grande parte da literatura encontrada faz uso de métodos baseados em algoritmos genéticos, como por exemplo, Fonseca e Fleming (1991) que discute a utilização do método para diferentes abordagens. Na área de métodos exatos, em contrapartida, há um vasto campo já explorado e que utiliza, como principal método, a busca em árvore (*branch-and-bound*) por Ross e Terry (1974), ou o MILP (*Mixed-integer linear programming*) por Heeman *at al.* (1996).

Em geral observa-se um amplo número de métodos exatos e heurísticos como principal proposta na resolução de grande parte dos problemas relacionados à formação de células, sequenciamento, restrição de capacidade, bem como ao problema aqui proposto. É possível identificar o uso de tal método nos artigos de Irani (1999), sobre o sistema de célula de manufatura, algoritmos genéticos híbridos para resolução de formação de células com múltiplas rotas; de Nsakanda e Price (2006), sobre a formação de células com planos alternativos de processo e restrições de capacidade de máquina; de Caux, Bruniaux e Pierraval (2000) e ainda por Nagi, Harhalakis e Proth (1990), os quais utilizam métodos heurísticos com programação linear para a resolução de problema de formação de células. Outras abordagens analisadas com mais detalhes podem ser encontrados na seção 2.3.

2.2 Representação de Problemas de Pesquisa Operacional

A subárea da PO denominada programação matemática emprega “símbolos matemáticos” para, a partir da idealização da realidade, representar as variáveis do sistema real (PUCCINI, 1990).

Vale ressaltar que o presente trabalho aborda tanto citações em programação matemática, quanto em linguagem de programação que será denominada de programação computacional. Assim o termo programação matemática não deve ser confundido com a expressão correlata presente na área computacional, referente à programação em linguagem de computador.

Em um modelo de programação matemática são incluídos três conjuntos principais de elementos, conforme Lisboa (2002): *i*) variáveis de decisão e parâmetros; *ii*) restrições; e *iii*) função objetivo.

As variáveis de decisão são as incógnitas a serem determinadas pela solução do modelo, enquanto os parâmetros se referem aos dados fixos de entrada. As restrições são um conjunto de equações ou inequações que limitam possíveis valores das variáveis. E por fim, a função objetivo é uma função matemática que expressa o intuito de maximizar ou minimizar o resultado do modelo.

A formulação de modelos matemáticos depende fundamentalmente do sistema a ser simulado. Nele a função objetivo de restrições pode ser linear ou não linear, as variáveis de decisão contínuas ou discretas e os parâmetros determinísticos ou

probabilísticos. Como exemplo um caso de programação linear por Bradley, Hax e Magnanti (1997), apresentado através da equação 1, onde x é um vetor coluna n dimensional de variáveis do modelo, c é um vetor linha n dimensional de fatores de ponderação, b é um vetor coluna m dimensional de parâmetros e A é uma matriz $m \times n$. Vale destacar que m é o número de equações do modelo.

$$\begin{array}{ll} \text{Função Objetivo} & Cx \\ \text{Sujeito a} & Ax = b \\ & X \geq 0 \end{array} \quad (1)$$

No estudo de caso abordado a função objetivo é linear com a presença de variáveis contínuas e discretas. Os parâmetros determinísticos e probabilísticos, no qual os probabilísticos englobam as três variáveis de incerteza propostas. A metodologia utilizada na resolução do problema é apresentada na seção 2.3.

2.3 Busca Tabu

A forma de resolução do estudo de caso escolhido será através do auxílio de métodos heurísticos, com a utilização de Busca Tabu. Este método, introduzido por Glover (1986), tem sido utilizado com sucesso na resolução de diversos problemas reais de otimização. Glover (1990) também apresentou algumas sofisticacões na forma da metodologia, consolidando ingredientes básicos necessários à utilização do método.

A Busca Tabu se caracteriza como uma heurística de busca na vizinhança da solução vigente. A partir de uma solução buscam-se soluções melhores na vizinhança desta solução vigente, empregando restrições para inibir certos movimentos, por um determinado número de iterações (GLOVER, 1997).

Basicamente realiza-se uma busca em uma vizinhança tomando como nova solução aquela com melhor valor da função objetivo. A vizinhança é um conjunto de soluções possíveis de gerar a partir de uma solução dada. Se não existem movimentos que melhorem a solução vigente, a Busca Tabu escolhe o movimento que “piora menos” o valor da função objetivo da solução anterior. Esta lógica é ilustrada de maneira geral através do fluxograma apresentado na figura 2.

No método usado, originalmente proposto por Glover (1990), é adicionado à metodologia um contador de número de iterações $Cesc$ para calcular o número de iterações executadas sem o melhoramento do resultado da solução objetivo. Quando $Cesc$ atinge um determinado número de iterações $Nesc$, o código regressa ao estado inicial de geração de uma solução inicial. Esta medida evita com que a Busca Tabu fique presa em uma determinada região de solução e não analise outras possibilidades.

Considerando que N é o número máximo de iterações que a Busca Tabu irá executar e i a iteração executada no momento, são dados os seguintes passos:

Passo 1. Carregar dados do problema e contadores;

Passo 2. Gerar solução inicial ($i = 1$ e $Cesc = 0$);

Passo 3. Intensificação - explorar a vizinhança da última solução gerada;

Passo 4. Verificar solução encontrada em relação à função objetivo;

Passo 5. Contar variável $Cesc$, quando aplicável;

Passo 6. Contar variável $i = i + 1$;

Passo 7. Se $Nesc > Cesc$, voltar para o passo 3; senão ir para passo 8;

Passo 8. Se $N = i$: finalizar execução. Caso contrário, ir para o passo 9;

Passo 9. Diversificação da solução (voltar para o passo 2).

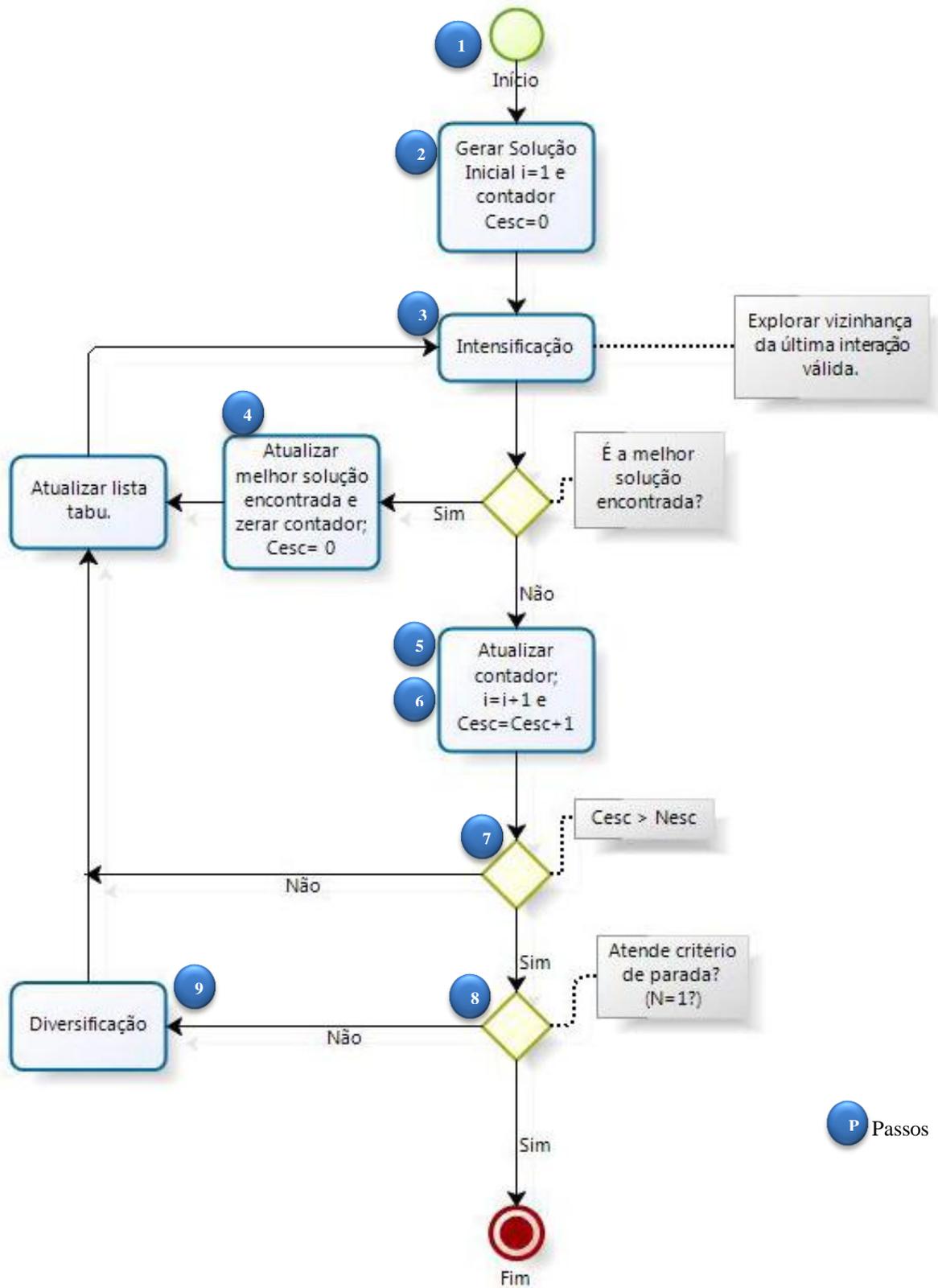


Figura 2 - Fluxograma Busca Tabu, adaptado de Glover (1990)

2.4 Incertezas

Um grande número de problemas em planejamento de produção e sequenciamento, alocação, transporte, finanças e engenharia requerem decisões a serem tomadas na presença de incertezas. Estas influenciam, como exemplo, os preços de *commodities*, disponibilidade de energia e demanda de um produto qualquer. Segundo Sahinidis (2004), a principal dificuldade na otimização envolvendo incertezas está na “negociação” das barreiras das incertezas, que podem ser vastas e, frequentemente, leva a modelos de otimização de grande escala.

Para Xiaoxia, Stacy e Christodoulos (2004), a questão da robustez na programação sobre incertezas tem recebido pouca atenção, apesar de importância e ao fato de que houve uma quantidade “substancial” de trabalhos para resolver o problema de projeto e operação de plantas com produção por bateladas sobre incerteza. A maioria dos trabalhos existentes tem seguido a mesma abordagem, na qual a incerteza é modelada por meio do uso de uma série de cenários, usando distribuições de probabilidade discretas ou a discretização das funções de distribuição contínua de probabilidade e a expectativa de um determinado critério de desempenho, tais como o lucro esperado, que é otimizado com relação à programação de variáveis de decisão.

Bassett, Pekny e Reklaitis (1997), exploram as incertezas no processo e processamento das flutuações do tempo, da confiabilidade do equipamento / disponibilidade, demanda e recursos humanos. Os autores usaram o método Monte Carlo de amostragem para gerar instâncias aleatórias, determinando um cronograma para cada instância, e gerando uma distribuição de agregados para induzir as propriedades políticas operacionais. Ierapetritou e Floudas (1998), focaram no agendamento de uma única fase, plantas multicelulares e multiprodutos com uma única linha de produção, em cada etapa que a demanda do produto estava envolvida. Eles usaram integração da quadratura de Gauss para avaliar o lucro esperado e formularam modelos programação linear inteira mista (MILP).

Ierapetritou, Hené e Floudas (1999) introduziram várias métricas para avaliar a robustez de um cronograma e propuseram um modelo de programação multiperíodos usando pontos extremos da faixa de demanda como cenários para

gerar uma única sequência de tarefas com o mínimo tempo de processamento médio de todos os cenários.

Balasubramanian e Grossmann (2002) usaram um modelo MILP multiperíodo para o agendamento de vários estágios de plantas flowshop com tempos de processamento incertos. Eles minimizaram o tempo de processamento esperado e desenvolveram um algoritmo de “ramo” especial e amarrado com um modelo de probabilidade agregado. As abordagens baseadas em cenários fornecem uma simples maneira implícita em incorporar incertezas. No entanto, eles inevitavelmente aumentam o tamanho do problema de forma significativa, já que o número de cenários aumentava exponencialmente com o número de parâmetros incertos. Este principal inconveniente limita a aplicação dessas abordagens para a resolução prática de problemas com um grande número de parâmetros incertos.

Sanmarti, Espuña e Puigjaner (1997) apresentaram uma abordagem diferente para a programação de produção e tarefas de manutenção em plantas de lote multipropósito com incerteza de falha de equipamento. Eles estimaram um índice de confiabilidade para cada unidade e para cada tarefa agendada e formularam um modelo de programação não linear inteira mista (MINLP) convexo para maximizar a confiabilidade do cronograma geral. Devido à dificuldade significativa na solução rigorosa do problema resultante, um método heurístico foi desenvolvido para encontrar soluções que melhorassem a robustez de um calendário existente.

Foi tentando também uma transformação de um modelo estocástico para um modelo direto de representação determinística equivalente que Orçun, Altinel e Hortaçsu (1996) consideraram uma transformação em tempos incertos em processos batelada e empregaram restrições de chance para dar conta do risco de violação de restrições de tempo, sobre certas condições, tais como funções de distribuição uniforme. Uma abordagem alternativa para a programação sobre incerteza está na programação reativa. É realizada para ajustar um cronograma, que normalmente é obtida de forma determinística, na realização dos parâmetros incertos ou na ocorrência de eventos inesperados. Devido ao "on-line" da natureza do sequenciamento reativo, é necessário gerar listas atualizadas temporariamente e, muitas vezes, são desenvolvidas abordagens heurísticas para modificações de horário.

Xiaoxia, Stacy e Christodoulos (2004) propôs uma abordagem de otimização robusta para resolver o problema de agendamento em limitada incerteza. A estrutura é baseada numa metodologia de otimização introduzindo a programação linear (LP) por problemas de Ben-Tal e Nemirovski (2000) e estendido no trabalho para problemas MILP. A abordagem produz "robustas" soluções que estão em um sentido imune contra as incertezas em ambos os coeficientes e parâmetros das restrições de desigualdade. A abordagem pode ser aplicada para resolver o problema de programação da produção com incerteza de tempos de processamento, exigências do mercado e / ou preços de produtos.

2.5 Literatura sobre o problema proposto

Considerando o estado da arte atual das ferramentas na área da pesquisa operacional (PO), para a área de otimização de problemas em logística da programação, grandes dificuldades não são comumente encontradas para resolver tais tipos de problema. No entanto, tem sido por muito tempo reconhecido (Beale, 1955; Dantzig, 1955) que a otimização determinística tradicional não é adequada para capturar o comportamento dinâmico verdadeiramente da maioria das aplicações do mundo real.

Encontrar as ferramentas de modelagem corretas é um dos problemas mais tecnologicamente desafiante que operadores e administradores enfrentam hoje. Diversas abordagens, com base em diferentes métodos matemáticos, são propostas com a mesma finalidade de aperfeiçoar parte do problema ou da programação completa de logística, ou parte dela. A otimização estocástica por meio de análise de cenários é uma metodologia também estudada e que pode ser ampliada para resolver problemas mais complexos, como do tipo *Supply, Transformation and Distribution* (STD) (LOUVEAUX e BIRGE, 1997; MULVEY, VANDERBEI e ZENIOS, 1995; ROCKAFELLAR e WETS, 1991; GASSMANN, 1990; BIRGE, 1985; VAN SLYKE e WETS, 1969; entre outros). Entretanto, os estudos encontrados que mais se aproximam do presente trabalho foram os que utilizam a resolução de problema generalizado de atribuição, *generalized assignment problem* (GAP) (WOODCOCK e ANDRE, 2010; MAJUMDAR e BHUNIA, 2005; TOKTAS e YEN, 2004; HIGGINS, 1999).

O GAP é um problema conhecido de otimização combinatória que envolve a atribuição de n tarefas para m agentes de modo que cada tarefa deve ser alocada a um único agente. Vale ressaltar que todas as tarefas devem estar atribuídas a algum agente, estes sujeitos às restrições. O objetivo é maximizar o lucro ou minimizar os custos de alocação de tarefas dos agentes.

Existem diversas aplicações para o GAP, como alocação de recursos nas áreas de logística e manufatura, e como já citado, pode ser aplicado ao problema do presente estudo. Representados matematicamente pelas equações 2, 3a, 3b e 3c. Aonde x_{ij} é a variável de decisão, sendo $x_{ij}=1$ se a tarefa i é atribuída ao agente j , caso contrário 0, p_{ij} é o custo da atribuição da tarefa i ao agente j , a_{ij} é a capacidade requerida para a atribuição da tarefa i ao agente j , e b_{ij} é a capacidade do agente j .

$$\text{Maximizar } Z = \sum_{i=1}^n \sum_{j=1}^m p_{ij} x_{ij} \quad (2)$$

Sujeito a

$$\sum_{i=1}^n a_{ij} x_{ij} \leq b_{ij}, \quad j \in \{1, \dots, m\}, \quad (3a)$$

$$\sum_{j=1}^m x_{ij} \leq 1, \quad i \in \{1, \dots, m\}, \quad (3b)$$

$$x_{ij} \in \{0,1\}, \quad i \in \{1, \dots, n\}, \quad j \in \{1, \dots, m\} \quad (3c)$$

Grande parte dos estudos são conduzidos sem considerar os impactos da incerteza. Waller, Schofer e Ziliaskopoulos (2001), no entanto, observou que a avaliação do desempenho da uma rede, ao se utilizar incerteza na espera da demanda, tende a superestimar podendo levar a resultados errôneos. Desta maneira, várias ações potenciais foram discutidas para lidar com o problema, como inflação de demanda, que favorece não só os rendimentos selecionados, mas melhorias que são esperadas com menor viagem no tempo total do sistema, bem como uma redução significativa na variância associada a estas medidas. Yang e Bell (1998) sugerem a necessidade de selecionar um nível de inflação apropriado para se obter uma teoria bem definida. Já Waller, Schofer e Ziliaskopoulos (2001) aplicaram a inflação de demanda em um problema com dinâmica no design da rede para atender à demanda incerta. Eles também sugerem a formulação de dois estágios estocásticos com recursos para o problema.

Alguns estudos recentes têm aplicado o conceito de confiabilidade baseados em projetos para resolver o problema de rede em condições de incerteza. O objetivo é obter um design caracterizado por uma baixa probabilidade de falha. As abordagens são feitas para maximizar e medir a confiabilidade ao introduzir restrições de confiabilidade com chance de maximizar a eficiência. Exemplos da abordagem incluem Yin (2002) para maximização de confiabilidade e satisfação; Chootinan e Recker (2005) para a maximização da capacidade de confiabilidade; e Sumalee, Connors e Watling (2006) para a viagem no tempo total da confiabilidade de maximização. Lo e Tung (2003) é exemplo desta última abordagem em que a capacidade de rede reservada é maximizada e sujeitas a restrições de confiabilidade em redes.

Vale ressaltar que dos trabalhos já encontrados, o que mais se aproxima deste estudo é do autor Escudero, Quintana e Salmareón (1999), acerca da solução de um STD com foco na redução de custo de *Supply* (situado na área de incertezas da demanda). Porém os autores focam pouco na redução de custo do sequenciamento logístico de entrega do produto e restrições durante o processo de entrega. As demais alternativas encontradas que se assemelham a esta situação podem resolver a clássica versão determinística do problema completo para um determinado cenário, ou uma versão estocástica envolvendo ainda apenas cenários determinísticos.

3 Descrição do Problema

O problema escolhido pertence à área de transporte, tendo como finalidade a tomada de decisões de seleção de veículos, consolidação das cargas para cada cliente e escolha do tipo de frete utilizado, a partir da possibilidade de um ou mais centros de distribuição, e que atenda às restrições de abastecimento dos clientes com menor custo possível. Assim, as principais características do problema são:

- A frota de veículos é heterogênea, em tamanho, tempos de execução, tipos de compartimentos e outras particularidades;
- Existe mais de uma possibilidade de pagamento de frete, impactando o custo final;
- A quantidade de veículos disponíveis diariamente pode ser afetada por incertezas;
- O tempo de transporte é uma incerteza, pois pode variar conforme o tráfego do horário de execução e o peso transportado por trecho;
- A demanda dos clientes pode ser maior que a capacidade/disponibilidade dos veículos.

3.1 Níveis de Planejamento

Se a tomada de decisões puder ser definida em diferentes níveis de planejamento, de acordo com Bateman e Snell (1998), as decisões organizacionais podem ser divididas em três níveis: estratégico, tático e operacional, conforme mostra a figura 3. Divididas de acordo com o tipo de trabalho que é desenvolvido por cada nível.



Figura 3 - Níveis de Planejamento

3.1.1 Nível estratégico

O nível estratégico compreende os altos executivos da organização, responsáveis pela definição dos objetivos, planos da empresa e a tomada de decisões quanto às questões de longo prazo da empresa, como: sua sobrevivência, crescimento e eficácia geral.

3.1.2 Nível tático

O planejamento, no nível tático, é utilizado para traduzir os objetivos gerais e as estratégias da alta diretoria em objetivos e atividades mais específicos. Neste nível são tomadas decisões que impactam o nível operacional em um horizonte mais amplo. No estudo de caso este horizonte está em torno de pelo menos 3 a 6 meses de operação, pois envolve decisões como o número de veículos e os tipos de contrato de frete.

3.1.3 Nível Operacional

No planejamento operacional, o processo é de uma menor amplitude em termos de horizonte de tempo. Ele aplica as decisões específicas definidas no planejamento tático e implementa tarefas mais relacionadas com o trabalho do dia a dia, como a programação de entrega diária.

3.1.4 Âmbito do problema

O presente trabalho está baseado exclusivamente nos níveis de decisão operacional e tático, já que, conforme a definição de Bateman e Snell (1998) o problema não chega a um nível de decisão dos altos executivos da empresa: o estratégico.

Os resultados de alocação dos veículos aos clientes e consolidação de cargas são os principais objetivos da análise no nível operacional e as decisões do número de veículos utilizados e a forma de pagamento de frete as principais decisões para o nível tático.

A ilustração completa dos parâmetros do problema está apresentada na figura 4, que apresenta a “arquitetura do problema”. Ou seja, apresenta-se a estrutura das informações e decisões associadas ao problema, que foram agrupadas por quadros e suas relações, e que são detalhadas nas demais seções do presente capítulo.

Para todo quadro contendo informações sobre o problema é dado um nome, iniciando por “Set_”, seguido do complemento do nome do agrupamento da tabela. Por exemplo, acrescenta-se “Vehicle” (Set_Vehicle), para o agrupamento dos veículos. Definiram-se assim grupos de quadros que contemplam todos os parâmetros utilizados pelo modelo. Como exemplo, na figura 4, para o grupo com parâmetros de veículos são encontrados os seguintes quadros: Set_Vehicle, Set_Vehicle_Cost_Setup, Set_Vehicle_Compartment e Set_Vehicle_Driver. As tabelas 1 e 2 apresentam, respectivamente, todos os índices e parâmetros associados ao problema, e que foram indicados na figura 4.

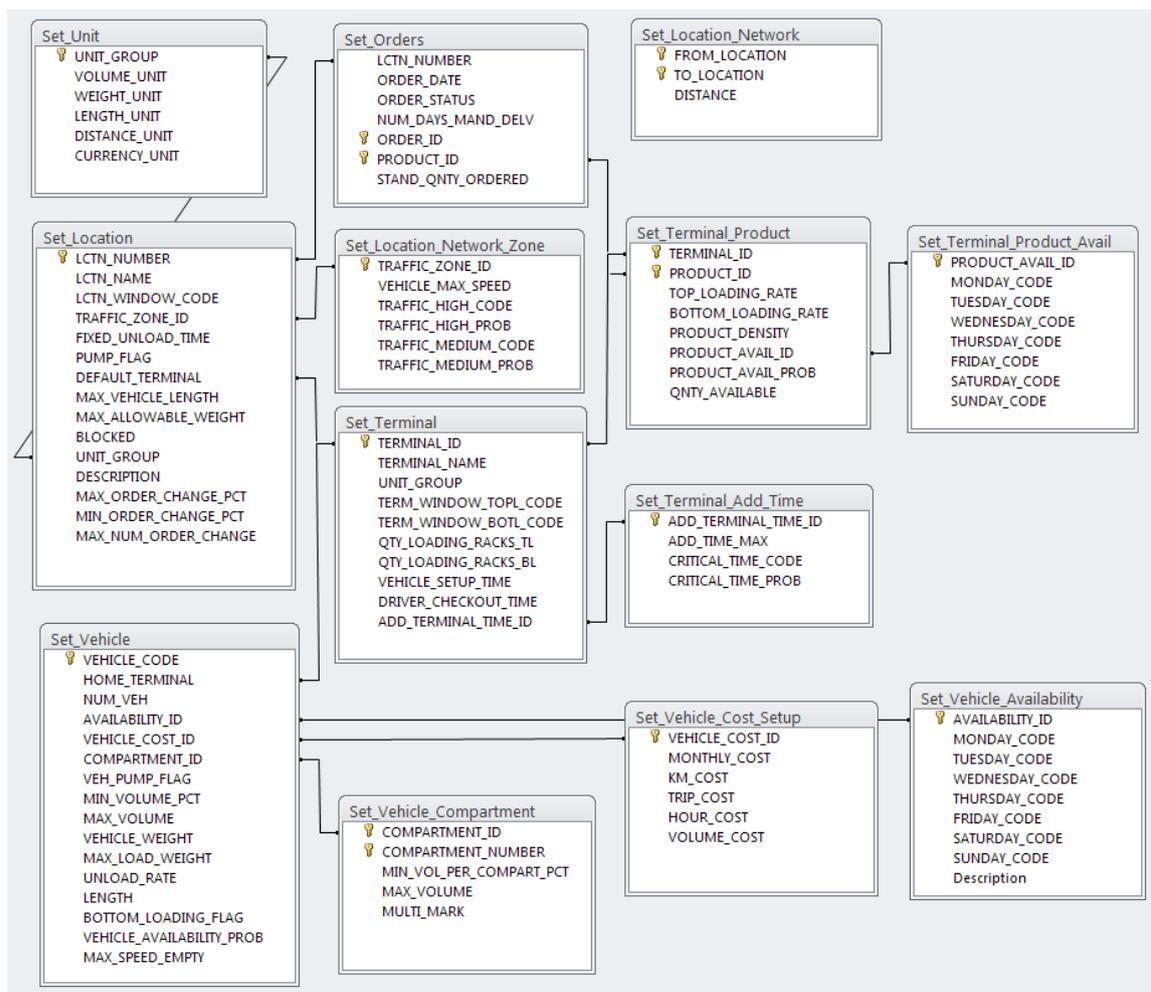


Figura 4 - Arquitetura do problema na implementação

Tabela 1 - Índices

Índices	Descrição	Nome utilizado na implementação
<i>l</i>	Índice <i>l</i> que define o código dos clientes. Exemplo <i>l1, l2, l3 .. etc.</i>	LCTN_NUMBER
<i>t</i>	Terminais <i>t</i> de carregamento. <i>t</i> igual a <i>t1, t2, t3 ... etc</i>	TERMINAL_ID
<i>v</i>	Índice identificador dos veículos. Exemplo <i>v1, v2, v3 ..., etc</i>	VEHICLE_CODE
<i>p</i>	Índice <i>p</i> que define o código dos produtos. Exemplo <i>p1, p2, p3, p4, ..., etc</i>	PRODUCT_ID
<i>o</i>	Ordem de pedido <i>o</i> solicitada pelo cliente <i>l</i> . $o \in l$.	ORDER_ID
<i>d</i>	Código que define a disponibilidade de produto <i>p</i> em terminal <i>t</i> . $d \in t, l$.	PRODUCT_AVAIL_ID
<i>c</i>	Código que define o tipo de pagamento de um veículo <i>v</i> , podendo ser <i>c</i> igual <i>FPT</i> ou <i>FT</i> . $c \in v$.	VEHICLE_COST_ID
<i>cpi</i>	Índice que define o número do compartimento <i>cpi</i> em um veículo <i>v</i> . $cpi \in v$. Exemplo veículo <i>v1</i> possui 4 compartimentos, sendo eles <i>cpi1, cpi2, cpi3</i> e <i>cpi4</i> .	
<i>h</i>	Horas do dia, sendo <i>h</i> igual de 0 a 23, referente à 00:00h. à 23:00h. respectivamente.	
<i>n</i>	Referente a um trecho <i>n</i> que o veículo <i>v</i> irá percorrer de um local (<i>t</i> ou <i>l</i>) para outro (<i>t</i> ou <i>l</i>). $n \in v, t, l$.	FROM_LOCATION e TO_LOCATION
<i>vg</i>	Índice que identifica o número de viagem. Exemplo <i>vg1, vg2, vg3, etc.</i>	
<i>z</i>	Identifica o tipo de velocidade de transporte permitida em uma determinada região <i>z</i> . $z \in l$.	TRAFFIC_ZONE_ID

Tabela 2 - Parâmetros

Índices	Descrição	Nome utilizado na implementação
Parâmetros referentes a custo		
CTkm(<i>c,v</i>)	Custo por quilômetro rodado de um veículo <i>v</i> do pagamento de frete tipo <i>c</i> .	KM_COST
CTmonth(<i>c,v</i>)	Custo mensal pago integralmente pela utilização de um veículo <i>v</i> do pagamento de frete tipo <i>c</i> .	MONTHLY_COST
CTtrip(<i>c,v</i>)	Custo por viagem ou carregamento de um veículo <i>v</i> do pagamento de frete tipo <i>c</i> .	TRIP_COST
Parâmetros de distância		
Dist(<i>n</i>)	Distância referente ao um trecho <i>n</i> .	DISTANCE
Parâmetros relacionados aos clientes		
Lmaxl(<i>l,v</i>)	Tamanho máximo em comprimento que um cliente <i>l</i> pode receber.	MAX_VEHICLE_LENGTH
Lmaxn(<i>l</i>)	Número máximo de vezes que um cliente <i>l</i> pode aceitar alteração de seus pedidos em um período de um mês. (Alterações podem ser de volume ou data de entrega)	MAX_NUM_ORDER_CHANGE
Lmaxo(<i>l,o</i>)	Porcentagem de variação para mais do volume total de um pedido <i>o</i> de um cliente <i>l</i> .	MAX_ORDER_CHANGE_PCT
Lmaxw(<i>l,v</i>)	Peso máximo de um veículo <i>v</i> que um cliente <i>l</i> pode aceitar	MAX_ALLOWABLE_WEIGHT
Lmino(<i>l,o</i>)	Porcentagem de variação para menos do volume total de um pedido <i>o</i> de um cliente <i>l</i> .	MIN_ORDER_CHANGE_PCT
Lplant(<i>l,t</i>)	Terminal <i>t</i> padrão de carregamento do cliente <i>l</i> .	DEFAULT_TERMINAL
Ltfix(<i>l</i>)	Tempo Fixo de descarga no cliente <i>l</i> .	FIXED_UNLOAD_TIME
Lwin(<i>l</i>)	Janela horária de trabalho do cliente <i>l</i> .	LCTN_WINDOW_CODE
Lz(<i>l,z</i>)	Zona de tráfego <i>z</i> que o cliente <i>l</i> pertence.	TRAFFIC_ZONE_ID
LBLK	Parâmetro binário que define se o cliente está ou não bloqueado por motivos de crédito. LBLK = 1 para Bloqueado e 0 = Desbloqueado.	BLOCKED
LPUMP	Define se é necessário bomba para entregar ao cliente. LPUMP = 1 para sim e = 0 para não.	PUMP_FLAG
Parâmetros de tráfego		
Nhigh(<i>z</i>)	Parâmetro que define os horários de alta intensidade de tráfego na zona <i>z</i> .	TRAFFIC_HIGH_CODE
Nhighp(<i>z</i>)	Probabilidade de ocorrência de tráfego de alta intensidade na zona <i>z</i> .	TRAFFIC_HIGH_PROB
Nmed(<i>z</i>)	Parâmetro que define os horários de média intensidade de tráfego na zona <i>z</i> .	TRAFFIC_MEDIUM_CODE
Nmedp(<i>z</i>)	Probabilidade de ocorrência de tráfego de média intensidade na zona <i>z</i> .	TRAFFIC_MEDIUM_PROB
Nspeed(<i>z</i>)	Velocidade máxima alcançada na zona de tráfego <i>z</i> , em situações de baixa intensidade de tráfego.	VEHICLE_MAX_SPEED
Parâmetros referentes às ordens de pedido		
Pdate(<i>o</i>)	Data de entrega do pedido <i>o</i> , no formato de dd/mm/aaaa.	ORDER_DATE
PdateIn	Data inicial de entrega do período que será simulado. Formato: dd/mm/aaaa.	[Dado de entrada]
PdateOut	Data final de entrega do período que será simulado. Formato: dd/mm/aaaa.	[Dado de entrada]
Ploc(<i>o,l</i>)	Cliente <i>l</i> do pedido <i>o</i> .	LCTN_NUMBER
Pvol(<i>o,p</i>)	Quantidade de volume do produto <i>p</i> no pedido <i>o</i> a ser entregue, sendo Pprod(<i>o,p</i>) o parâmetro que define o código do produto a ser entregue.	STAND_QNTY_ORDERED

Parâmetros referentes a terminais e produtos

TCT(t)	Tempo médio de espera para sair do terminal t após o carregamento de uma viagem.	DRIVER_CHECKOUT_TIME	
TPAcod(d,p,t)	Parâmetro que define os horários de disponibilidade d do produto p no terminal t .	MONDAY_CODE SUNDAY_CODE	a
TPAprb(p,t)	Probabilidade de indisponibilidade de produto p , no tipo de disponibilidade d , em um terminal t .	MONDAY_PROB SUNDAY_PROB	a
TPBL(t,p)	Vazão de escoamento do produto p para carregamento do tipo BL no terminal t .	BOTTOM_LOADING_RATE	
TPde(t,p)	Densidade do produto p no terminal t	PRODUCT_DENSITY	
TPT(t)	Tempo de carregamento médio de um veículo v no terminal t .	VEHICLE_SETUP_TIME	
TPTL(t,p)	Vazão de escoamento do produto p para carregamento do tipo TL no terminal t .	TOP_LOADING_RATE	
TPvol(t,p)	Volume de produto p disponível no terminal t .	QNTY_AVAILABLE	
TQBL(t)	Número de postos de carregamento do tipo BL no terminal t .	QTY_LOADING_RACKS_BL	
TQTL(t)	Número de postos de carregamento do tipo TL no terminal t .	QTY_LOADING_RACKS_TL	
TWBL(t)	Parâmetro que define a janela horária de carregamento do tipo BL no terminal t	TERM_WINDOW_BOTL_CODE	
TWTL(t)	Parâmetro que define a Janela horária de carregamento do tipo TL no terminal t .	TERM_WINDOW_TOPL_CODE	

Parâmetros relacionados a veículos de transporte

Vavp(v)	Probabilidade de um veículo v estar disponível para utilização	VEHICLE_AVAILABILITY_PROB	
VCmark(v,cpi)	Valor volumétrico no qual o compartimento cpi pode ser gradualmente carregado no veículo v .	MULTI_MARK	
VCmax(v,cpi)	Volume máximo que cada compartimento cpi pode carregar no veículo v .	MAX_VOLUME	
VCminp(v,cpi)	Porcentagem mínima que cada compartimento cpi pode ser carregado no veículo v . VCmin padrão igual a 85%.	MIN_VOL_PER_COMPART_PCT	
Vcost(v,c)	Código que identifica o tipo de pagamento de frete c , do um dado veículo v .	VEHICLE_COST_ID	
VCmaxw(v,cpi)	Peso máximo de produto que o compartimento cpi pode carregar no veículo v .	MAX_WEIGHT	
Vlen(v)	Comprimento total do veículo v .	LENGTH	
Vmax(v)	Volume máximo de carregamento do veículo v .	MAX_VOLUME	
Vmaxw(v)	Peso máximo que um veículo v pode ser carregado.	MAX_LOAD_WEIGHT	
Vminp(v)	Porcentagem mínima que o veículo v pode ser totalmente carregado. Vminp padrão igual a 85%.	MIN_VOLUME_PCT	
Vnv(v)	Número de veículos do tipo v disponíveis	NUM_VEH	
Vplant(v, t)	Terminal t padrão, aonde o veículo v é utilizado	HOME_TERMINAL	
Vspeed(v)	Velocidade máxima que o veículo v alcança percorrer sem produto carregado e congestionamento	MAX_SPEED_EMPTY	
Vunload(v,p)	Vazão de escoamento na descarga do produto p , no veículo v .	UNLOAD_RATE	
Vw(v)	Peso total do veículo v sem produto carregado. Também chamado de tara.	VEHICLE_WEIGHT	
VBL	Parâmetro binário que define se o veículo possui ou não opção de carregamento BL. VBL = 1 para Possui e 0 = Não possui.	BOTTOM_LOADING_FLAG	
VPUMP	Define se um veículo v possui bomba de descarga. VPUMP = 1 possui e VPUMP = 0 Não possui.	VEH_PUMP_FLAG	

3.2 Clientes

No estudo de caso proposto são dados diferentes tipos de clientes I que devem ser abastecidos pela rede logística conforme a demanda solicitada pelo próprio cliente. Cada cliente I possui distintos parâmetros que o caracterizam, definidos quanto a sua capacidade física, localização e comportamento. Ilustrado conforme figura 5.



Figura 5 - Parâmetros dos clientes

Os parâmetros que definem a capacidade física são $L_{max}(I,v)$ e $L_{maxw}(I,v)$, constituindo respectivamente no comprimento e peso máximo que um veículo v pode ter para entregar para o cliente I . Adicionalmente $L_{fix}(I)$ define o tempo médio gasto para realizar a entrega no cliente I , sendo relacionado apenas com o tempo utilizado por um veículo qualquer estacionar no cliente I e esperar pela documentação.

Com relação à localização do cliente é dado o parâmetro $L_z(I,z)$ definindo a zona de tráfego z que o cliente I pertence e $L_{plant}(I,t)$ o terminal t padrão que usualmente o cliente I recebe seus pedidos.

E por último os parâmetros de comportamento, tais como $L_{maxn}(I)$, $L_{maxo}(I,o)$ e $L_{mino}(I,o)$ estabelecendo respectivamente o número máximo de vezes que o cliente I aceita alteração da data de entrega ou volume de seus pedidos e a variação em porcentagem para mais e para menos que seus pedidos podem ser alterados sem serem contabilizados por $L_{maxn}(I)$. E $L_{win}(I)$ que expressa a janela horária que um cliente I fica disponível para receber pedidos.

A figura 6 apresenta o quadro Set_Location utilizado na implementação, contendo os parâmetros dos clientes.

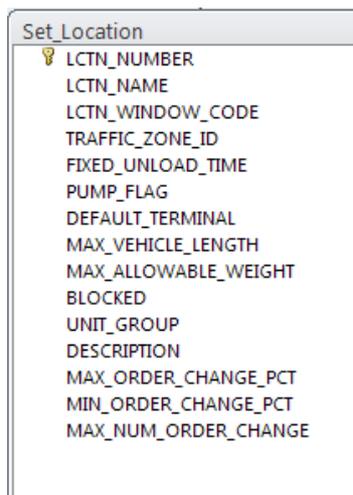


Figura 6 - Parâmetros dos clientes

3.2.1 Demanda dos clientes

A demanda dos clientes é gerada pela soma da quantidade de volume $Pvol(o,p)$ das ordens de entrega o para cada produto p . Os parâmetros $PdateIn$ e $PdateOut$ são os parâmetros de entrada para informar respectivamente a data inicial e final de entrega do período que se deseja simular no estudo de caso, sendo que $PdateOut$ é maior e igual a $PdateIn$. Este período filtra as ordens através do parâmetro $Pdate(o)$, criando um grupo de ordens a serem entregues.

O volume informado em $Pvol(o,p)$ é considerado volume padrão devido a sua flexibilidade de alteração do volume original da ordem de entrega para a acomodação do pedido no veículo v a ser utilizado. A variação do volume padrão para mais ou para menos é definido pela configuração dos parâmetros de clientes já mencionados $Lmaxo(l)$, $Lmino(l)$ e $Lmaxn(l)$. E por fim o parâmetro $Ploc(o,l)$ é quem define o cliente l que esta solicitando a ordem de pedido o .

A figura 7 representa a tabela de pedidos utilizada na implementação.

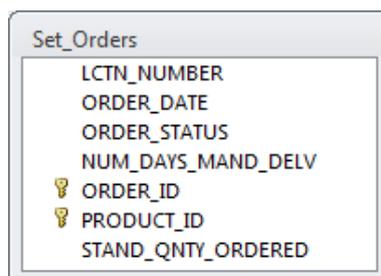


Figura 7 - Ordens de serviço

3.3 Terminais

Terminais t de entregas são os pontos onde os veículos de transporte carregam os pedidos solicitados o . O presente estudo de caso assume que um veículo que carrega um pedido no terminal $t1$, após a entrega do pedido, deve retornar ao terminal $t1$ novamente.

A figura 8 corresponde ao agrupamento de quadros associadas às informações de terminais utilizados na implementação da solução. Nele é aberta a presença não apenas de índices, variáveis e parâmetros relacionados aos terminais, mas também aos produtos. Já que são os terminais t que disponibilizam os diferentes tipos de produtos, ambos os índices estão diretamente relacionados entre si, assim como seus parâmetros e variáveis.

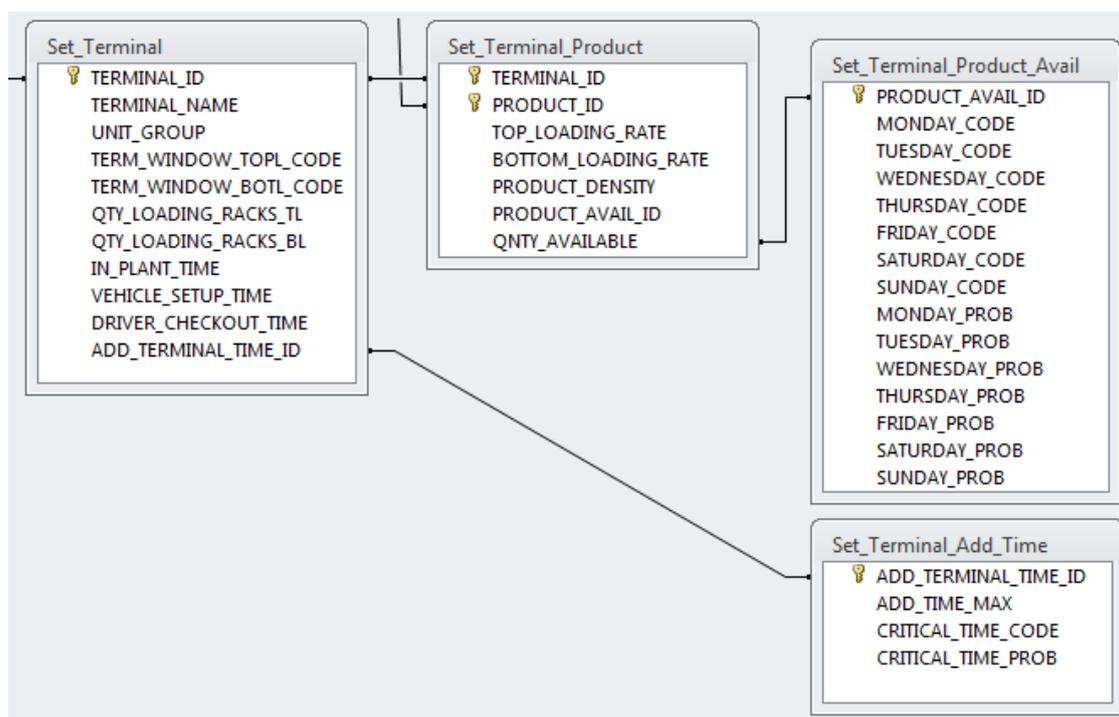


Figura 8 - Quadros relacionados a terminais e produtos na implementação

Existem diferentes tipos de carregamento que o terminal oferece para carregar os veículos, no exemplo proposto os veículos podem ser carregados de duas diferentes maneiras, variando conforme o tipo de fabricação do veículo, no caso: Top Loading (TL) e Bottom Loading (BL).

A princípio os veículos já são de fábrica *BL*, visto que no momento da carga do veículo o produto é escoado pela parte superior dos tanques com a utilização da força da gravidade, enquanto para os *TL* faz-se o uso de uma bomba que possui uma vazão maior do que a alcançada pela escoação, utilizando a força da gravidade. Desta forma os veículos TL apresentam uma vazão $TPTL(t,p)$ e podem ser carregados mais rapidamente que veículos BL que apresentam uma vazão $TPBL(t,p)$. Os tipos de carregamento Top Loading (*TL*) e Bottom Loading (*BL*) são ilustrados respectivamente pelas figuras 9 e 10.



Figura 9 - Carregamento do tipo Top Loading (TL)



Figura 10 - Carregamento do tipo Bottom Loading (BL)

Como estes são processos de diferentes postos de carregamento em um terminal t , os parâmetros $TQBL(t)$ e $TQTL(t)$ definem o número de postos de carregamento e os $TWBL(t)$ e $TWTL(t)$ os horários de disponibilidade do tipo de

carregamento BL e TL respectivamente. Entretanto é importante mencionar que os veículos TL são também BL de fábrica, desta forma os caminhões TL possuem vantagem de disponibilidade de horas de carga no terminal, pois podem carregar de ambas as formas.

A soma da disponibilidade total TL e BL condiz com a capacidade total de carregamento do terminal em determinado momento. Quando toda esta capacidade está sendo ocupada por um determinado espaço de tempo (capacidade total de carregamento igual a zero), é criada uma janela de indisponibilidade de tempo, até que se libere um posto de carregamento dando espaço para o carregamento da próxima entrega.

A soma dos parâmetros fixos TPT(t) e TCT(t) são referentes ao tempo médio gasto por um veículo no terminal no carregamento. TPT(t) é o tempo de espera do motorista no terminal para carregar o veículo mais o tempo gasto para o veículo estacionar e preparar as mangueiras para carregar o produto, enquanto TCT(t) referente ao tempo médio gasto para desacoplar as mangueiras do caminhão e retirar o veículo do terminal com sua documentação.

Quanto às janelas de disponibilidade por produto, estas são referentes principalmente às variáveis de incerteza que o problema pode enfrentar na disponibilidade de produto, desta forma estas são expressadas em tempos definidos pelos parâmetros $TPAcod(d,p,t)$ e $TPAprb(p,t)$, detalhados na seção 3.3.1. O índice d é um conjunto de códigos que define os horários de disponibilidade de produto. Como informação adicional, para a implementação utiliza-se de código de Hash perfeito mínimo para expressar este índice, conforme mostra o apêndice A.

3.3.1 Incerteza na demanda - disponibilidade de produto

A indisponibilidade de produto nos terminais é ocasionada por diversos fatores e imprevistos que são difíceis de serem listados devido à ampla possibilidade de eventos. Eles podem ocorrer pela falta de produto por motivos de problemas no bombeamento ou na refinaria que abastece o terminal, se estendendo a problemas como falta de energia elétrica, manutenção e até casos mais incomuns como roubo de equipamentos dos terminais. Assim estes e outros

fatores que podem ocorrer são expressos no estudo de caso proposto como incerteza na disponibilidade de produto.

$TPAprb(p,t)$ expressa a probabilidade de ocorrência destes fatores no terminal t e por produto p , sendo eles determinados pelo histórico de ocorrência no último ano registrado pela empresa escolhida para o estudo de caso proposto. O parâmetro $TPAcod(d,p,t)$ informa a janela original de disponibilidade do produto p no terminal t e nos horários d , sem a presença de incerteza.

Como exemplo, para expressar uma janela de indisponibilidade $TPAcod(d_1,p,t)$, de um terminal t que possui disponibilidade de produto p na janela horária d_1 , sendo d_1 igual à janela das 02:00 às 20:00h. Dados que o histórico de falta de produto $TPAprb(p,t)$ é equivalente a 5%, é dada a seguinte sequência lógica:

Passo 1. Para cada hora do dia definida por h , gerar um número randômico $rnd_{(h)}$ correspondente;

Passo 2. Se $rnd_{(h)}$ for menor que $TPAprb(p,t)$, a hora h antes disponível passa a ficar indisponível;

Passo 3. Descontar todas as horas que ficaram indisponíveis pelo passo 2 do parâmetro $TPAcod(d_1,p,t)$;

Passo 4. Disponibilidade final no terminal t para o produto p igual ao resultado final do passo 3.

O exemplo é ilustrado na tabela 3, mostrando tanto a lógica quanto aos dados do exemplo. A ordem de leitura da tabela 3 é de cima para baixo, seguindo a referência de passos descritos, até a obtenção do valor final de disponibilidade de produto final $TPAcod(d_1,p,t)$. Os valores destacados na tabela representam as horas vigentes do passo indicado e suas respectivas alterações.

Assim a tabela 3 inicialmente apresenta o valor inicial de $TPAcod(d_1,p,t)$ igual a 1. Logo após é gerado um valor randômico $rnd(h)$ para cada hora vigente de $TPAcod(d_1,p,t)$ inicial e comparado com $TPAprb(p,t)$. Como apenas para a hora 02:00 do dia o valor randômico $rnd(2)$ ficou menor que $TPAprb(p,t)$, esta hora é descontada do valor original $TPAcod(d_1,p,t)$, gerando o valor final de $TPAcod(d_1,p,t)$ conforme última linha da tabela 3.

Tabela 3 - Cálculo de disponibilidade de produto.

Passo \ h	0:00	1:00	2:00	3:00	4:00	5:00	6:00	7:00	8:00	9:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00
Horário original TPAcod2(d1)	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
<i>md(h) gerados</i>			0.01	0.32	0.30	0.50	0.12	0.56	0.91	0.77	0.53	0.88	0.67	0.23	0.11	0.45	0.63	0.42	0.63	0.91				
TPAprb2(d1)			0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05	0.05				
Se $md(h) <$ TPAprb2(d1)			-1																					
Disponibilidade real Disp2(d1)	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0

O algoritmo de disponibilidade de produto é executado por dia de execução do problema. É importante ressaltar que caso o programa seja executado novamente, mesmo que os parâmetros de entrada do sistema sejam iguais, a disponibilidade de produto final pode variar devido à utilização de variáveis randômicas. O método de variáveis randômicas utilizado é referente à proposta de Ben-Tal e Nemirovski (2000).

3.4 Veículos de transporte

Os produtos são distribuídos por uma frota de veículos heterogênea e limitados que podem estar ou não disponíveis, sendo esta uma das variáveis de incertezas do problema. Assim é pré-determinado um grupo de veículos por contrato com a transportadora ou com uma frota própria que muitas vezes podem sofrer indisponibilidade devido à quebra e/ou acidentes. É importante ressaltar que para o presente estudo de caso o número de motoristas utilizados para cada veículo v não será dado como parâmetro de entrada, uma vez que o número de motoristas disponíveis por veículo é de responsabilidade da transportadora.

Geralmente estas incertezas de disponibilidade podem acarretar ou na substituição por outro caminhão, que em muitas vezes é menos eficiente, ou no aumento de tarifa e, por último, na diminuição da capacidade de entrega da frota. A forma na qual a incerteza de disponibilidade dos veículos é tratada será mostrada na seção 3.4.4.

Na figura 11, é apresentada a disposição em quadros de todos os parâmetros e índices relacionados às restrições dos veículos de transporte utilizadas na implementação.

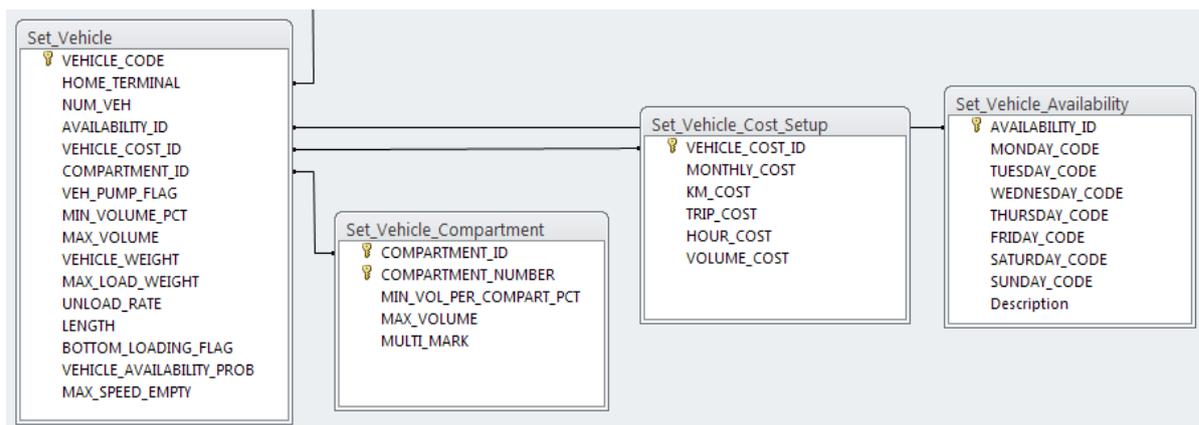


Figura 11 - Quadros de veículos de transporte

Os parâmetros que definem as características principais dos veículos são dados por $Vlen(v)$, $Vmax(v)$, $Vw(v)$ e $Vmaxw(v)$ informando respectivamente o tamanho em comprimento, volume máximo, tara e peso máximo do veículo. $Vplant(v,t)$ informa o terminal t padrão onde o veículo v geralmente carrega. No parâmetro $Vnv(v)$ é dado o número de veículos disponíveis para o determinado tipo de veículo v . E por fim, $Vcost(v,c)$ indicando a forma de pagamento de frete c do veículo v , que é mostrado com mais detalhes na seção 3.5.

3.4.1 Restrições de segurança de carregamento

Adicionalmente aos parâmetros principais no ramo de transporte de combustíveis existem restrições de carga mínima para que um veículo v carregado possa sair do terminal. São restrições de segurança devido o transporte de produtos com capacidade explosiva, devendo assim haver uma quantidade mínima a ser carregada por caminhão e por compartimento. Tais regulamentos podem ser encontrados em normas de segurança do país que se esta operando ou ainda pela empresa que está transportando o produto. No Brasil pode ser encontrado em regulamento técnico de qualidade 7c INMETRO (2009).

O presente trabalho segue as seguintes restrições de segurança:

- i) Máximo um compartimento vazio por veículo;
- ii) Volume mínimo que um compartimento cpi pode carregar igual a 85%, definido por $VCminp(v,cpi)$.

3.4.2 Compartimentação dos veículos

Os veículos, já de fabrica, possuem vagões de carga seccionados por compartimentos, que fracionam a capacidade máxima do caminhão em partes, muitas vezes heterogêneas. Conforme apresentado pelas figuras 12 e 13.



Figura 12 - Frota de veículos heterogênea – vagões de carga

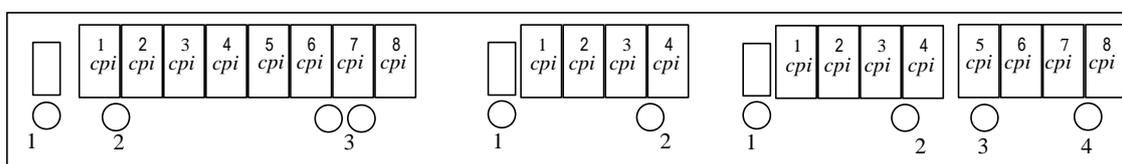


Figura 13 - Frota de veículos heterogênea – compartimentos

Além de obedecer à restrição mínima de carregamento sobre o volume total do caminhão, é ainda necessário atender a essa mesma regra de volume mínimo por compartimento cpi , definido por $VC_{min}(v,cpi)$, tendo como valor padrão também igual a 85%.

$VC_{max}(v,cpi)$ e $VC_{maxw}(v,cpi)$ são os parâmetros que informam respectivamente o volume e peso máximo que cada compartimento cpi pode receber para carregamento de produto no veículo v . Levando em conta estes critérios é apresentada a tabela 4 como exemplo de compartimentalização de dois veículos, $v1$ com quatro compartimentos e $v2$ com cinco compartimentos.

Tabela 4 - Volume e peso máximo por compartimento

Caminhão v	Compartimento cpi	Volume $V_{Cmax}(kUGL^1)$	Peso $VC_{maxw}(kg^2)$
V_1	1	1	4000
V_1	2	1	4000
V_1	3	1	4000
V_1	4	1	4000

¹kUGL: representa mil galões

²kg: representa quilos

V_2	1	2	8000
V_2	2	1	4000
V_2	3	1	4000
V_2	4	1	4000
V_2	5	1	4000

Adicionalmente, para que cada compartimento cpi de um veículo possa ser preenchido gradativamente com produto, é necessário o parâmetro $VCmark(v,cpi)$ informando a graduação volumétrica do compartimento. Como exemplo, para um veículo com compartimento $cpi1$ de volume máximo $VCmax(cp,cpi1)$ igual a 1kUGL e $VCmark(cp,cpi1)$ igual a 0,1 kUGL. Tem-se que este compartimento pode ser carregado com volume igual a 1 kUGL, 0,9 kUGL, 0,8 kUGL, 0,7 kUGL e assim por diante. Entretanto, como o parâmetro $VCminp(v,cpi1)$ permite, por segurança, apenas carregamentos de no mínimo 85% do volume total do compartimento, o compartimento $cpi1$ pode ser preenchido apenas com 1kUGL ou 0,9kUGL.

A figura 14 ilustra um segundo exemplo de multimarcas para um compartimento $VCmax(cp,cpi4)$ de 3kUGL, $VCmark(cp,cpi4)$ igual a 0,1 kUGL e $VCminp(v,cpi4)$ de 85%.

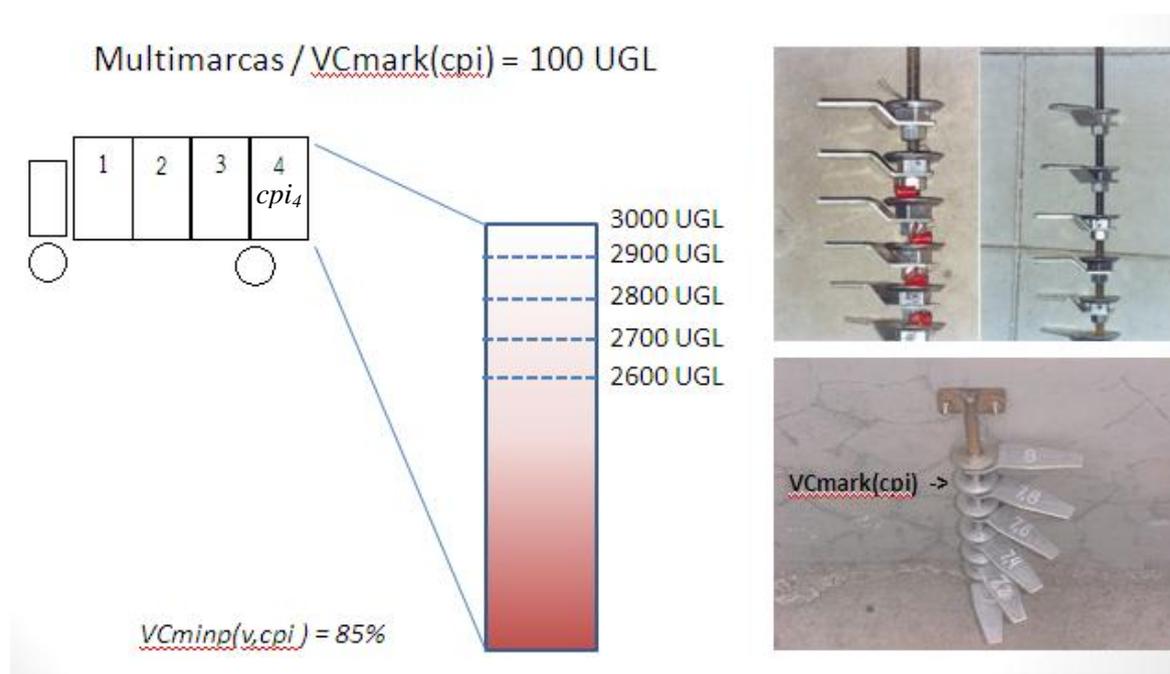


Figura 14 - Exemplo de um compartimento de 3000 UGL com multimarcas de 100 UGL, podendo ser carregado com até 2600 UGL devido restrição de segurança de 85%.

3.4.3 Restrição de necessidade de bomba no cliente

O fator que define se um cliente / irá receber ou não veículos que possuem bomba é definido pelos parâmetros LPUMP e VPUMP, sendo LPUMP para cliente que necessita bomba e VPUMP para veículos que possuem bomba.

Caso um cliente possua tanques subterrâneos, a descarga do produto é feita utilizando a própria força da gravidade. Como exemplo, nos postos de combustíveis, LPUMP será igual a “falso”. Porém, quando o cliente possui tanques aéreos e não possui uma bomba própria em suas instalações, o caminhão que irá entregar o produto deverá conter uma bomba para poder fazer a vazão do produto mecanicamente, no caso LPUMP e VPUMP devem ser iguais a “verdadeiro”. Esta segunda classe de clientes são normalmente indústrias.

É importante ressaltar que clientes que não precisam de bomba para receber a entrega do produto possuem uma vantagem em relação à disponibilidade de caminhões, já que nem todos os caminhões possuem bomba, ajudando a flexibilizar a operação. Desta forma, o número de caminhões disponíveis para clientes que não precisam de bomba para receber um produto é sempre maior ou igual ao número de caminhões disponíveis para clientes que precisam de bomba.

3.4.4 Incerteza de disponibilidade de veículo

Na prática, a ocorrência de indisponibilidade de veículos de transporte, geralmente, pode acarretar na substituição por outro veículo, que em muitas vezes é menos eficiente, seguido por um aumento de tarifa e, por último, na diminuição da capacidade de entrega da frota.

A forma na qual é representada a presença de indisponibilidade de um veículo no problema é similar à utilizada para a disponibilidade de produto nos terminais, ou seja, com o uso de variáveis randômicas versus uma probabilidade de ocorrência $V_{avp}(v)$.

No caso é dado um valor para $V_{avp}(v)$ entre 0 e 1, indicando a probabilidade de ocorrência para o tipo de veículo v indicado. Se o valor da variável randômica, de 0 a 1, executada no momento da verificação for menor do

que o valor indicado em $V_{vp}(v)$ é descontado um veículo da variável $V_{nv}(v)$ que indica o total de veículos do tipo v que está sendo analisado.

Os valores utilizados em $V_{vp}(v)$ foram também retirados de dados de indisponibilidade de veículo do último ano na empresa do estudo de caso. Porém, devido aos baixos valores de probabilidade para $V_{vp}(v)$ provenientes do campo, aproximadamente 0,5% ao mês para alguns veículos, é estipulado no presente trabalho que o cálculo de indisponibilidade de veículos ocorre no máximo uma vez para cada tipo v , penalizando assim em apenas um veículo por tipo. Ou seja, para veículos $v1$ com $V_{nv}(v1)$ igual a 3, a probabilidade de indisponibilidade pode minimizar o número de caminhões para no máximo igual a 2. Entretanto, esta abordagem mesmo que simplista, implica uma total reestruturação da alocação de viagens, pois caso apenas 1 veículo não esteja disponível, já é necessário realocar toda uma cadeia de viagens para poder compensar a diminuição da capacidade de entrega.

3.5 Modelo de Pagamento de Frete

O problema estudado admite a possibilidade de diferentes formas de pagamento de frete c , sendo este tipo de custo atribuído para cada veículo v . A variável c pode representar duas diferentes possibilidades de custo: frota própria terceirizada (*FPT*) ou frota terceirizada (*FT*), que são detalhadas na seção 3.5.1.

As formas de custo de frete são baseadas em três parâmetros de entrada $CT_{month}(c,v)$, $CT_{km}(c,v)$ e $CT_{trip}(c,v)$. O parâmetro $CT_{month}(c,v)$ corresponde ao valor a ser pago pela utilização do veículo v no mês. Caso o veículo seja utilizado por apenas 15 dias ou menos, a tarifa $CT_{month}(c,v)$ por contrato, pode ser cobrada apenas pela metade do seu valor, porém não menos do que isso, mesmo que o veículo não seja utilizado. E finalizando, $CT_{km}(c,v)$ refere-se ao valor a ser cobrado por cada quilômetro rodado e $CT_{trip}(c,v)$ por cada viagem do veículo v .

A equação 4 apresenta a formulação utilizada para o cálculo do custo de entrega por viagem. O índice vg é utilizado para definir o número da viagem que esta sendo executada. Assim na equação 4 é possível observar a tarifa $CT_{km}(c,v)$ sendo multiplicada por $Distância(vg)$ que representa o número de quilômetros percorridos pela viagem vg e logo após sendo somado apenas uma

vez à tarifa $CT_{trip}(c,v)$ por estar se tratando de apenas uma viagem do veículo. A equação 4 também pode ser entendida como uma equação de nível operacional, pois se trata de uma viagem pontual na programação de um período mais extenso e que será abordada pela equação 6.

$$CustoViagemOp_{(vg,c,v)} = (Distância_{vg} CT_{km}(c,v) + CT_{trip}(c,v)) \quad (4)$$

Para se obter uma expressão matemática similar à equação 4 para um período mais extenso, será necessário adicionar o custo mensal do veículo $CT_{month}(c,v)$ à equação 4, gerando assim a equação 6 que será apresentada a seguir. Entretanto para adicionar o custo mensal a um período exclusivo de apenas uma única viagem vg é necessário primeiramente ratear o custo mensal em uma única viagem.

Na equação 5 é apresentada a forma de rateio proposta, apresentando o critério de rateio do custo mensal a ser utilizado. O critério nada mais é que a média ponderada do volume Vol_{vg} e Horas $DHrs_{vg}$ da viagem vg em relação a todas as outras viagens executadas no terminal t . Ou seja, em um dado terminal t foram executadas 99 viagens, vg_{99} , e deseja-se calcular a tarifa mensal CT_{month} a ser cobrada para a viagem de número 5, vg_5 , é dada a equação 5.

$$CustoViagemM_{(t,vg_5,c,v)} = \left(\frac{Vol_{vg_5} \cdot DHrs_{vg_5}}{\sum_{n=vg_1}^{vg_{99}} [Vol_n \cdot Hrs_n]} \right) \cdot CT_{month}(c,v) \quad n \in \{t\} \quad (5)$$

Após o cálculo do custo por viagem operacional e mensal, nas equações 4 e 5 respectivamente, os valores finais são somados para se obter o custo total por viagem, conforme a equação 6.

$$CustoTotalViagem_{(t,vg,c,v)} = (CustoViagemOp_{(vg,c)} + CustoViagemM_{(t,vg,c,v)}) \quad (6)$$

A fim de exemplificar com mais detalhes a citada formulação de custo de frete, a seguinte seção 3.5.1 apresenta com mais detalhes a relação dos tipos de

frota e os três parâmetros de custo, CTmonth, CTkm e CTtrip abordados no estudo de caso.

3.5.1 Exemplo de diferentes formas de pagamento

Baseando-se nos parâmetros de custo de frete é possível contextualiza-los com duas diferentes formas de frota: frota própria terceirizada *FPT* e frota terceirizada *FT*. Conforme apresentadas a seguir:

i) FPT - Frota Própria Terceirizada

A empresa possui o veículo, porém contrata um transportador para administrar, aonde geralmente as tarifas negociadas são CTmonth, CTkm e CTtrip. Entretanto vale ressaltar que o fato de a forma de contrato possuir três tarifas não exclui a possibilidade de eventualmente alguma delas ser nula.

Neste tipo de contrato a transportadora terceirizada assume todos os custos e riscos administrativos, como o pagamento de motoristas e de segurança no transporte, por isso a necessidade de uma tarifa mensal que seria um valor equivalente a estes custos assumidos.

ii) FT - Frota Terceirizada

Ao contrário da tarifa *FPT*, a *FT* apresenta somente duas tarifas, CTkm e CTtrip, já que os caminhões contratados como *FT* não são de exclusividade do contratante, ou seja, não há compromisso do contratante nem do contratado, de utilizar ou disponibilizar estes caminhões, pois tanto os caminhões como o serviço prestado são disponibilizados por terceiros.

Por não existir nenhum vínculo entre as partes, geralmente as tarifas *FT* são mais caras que as *FPT*, mas isso não é obrigatório.

3.5.1.1 Comparação dos tipos de pagamentos de frete citados

A presente seção expõe uma comparação aritmética dos diferentes modelos citados. Vale lembrar que a abordagem utilizada a seguir é apenas explicativa e que o trabalho propõe outros métodos de resolução mais robustos

baseados em PO. A tabela 5 expõe resumidamente a relação dos parâmetros e índices por cada forma de pagamento de frete.

Tabela 5 - Relação dos índices e parâmetros por forma de pagamento

Nome da Tarifa	c = FPT	c = FT
Tarifa Mensal	$CT_{month}(fpt,v)$	-
Tarifa por Km	$CT_{km}(fpt,v)$	$CT_{km}(ft,v)$
Tarifa por carregamento	$CT_{trip}(fpt,v)$	$CT_{trip}(ft,v)$

Como as tarifas da *FPT* e *FT* são calculadas com os mesmos parâmetros, é possível formular um ponto de equilíbrio onde as duas tarifas são equivalentes financeiramente e afirmar que acima deste ponto de utilização da tarifa *FPT* é a mais viável e abaixo deste ponto a tarifa *FT* é a mais econômica. O ponto de equilíbrio entre *FPT* e *FT* é apresentado pela equação 7, sendo $ToTalKm_v$ e $Totalvg_v$ o número total de quilômetros e viagens que um veículo v pode percorrer no período de um mês.

$$CT_{month}(fpt,v) + (CT_{km}(fpt,v) \cdot ToTalKm_v) + (CT_{trip}(fpt,v) \cdot Totalvg_v) = (CT_{km}(ft,v) \cdot ToTalKm_v) + (CT_{trip}(ft,v) \cdot Totalvg_v) \quad (7)$$

A equação 7 é ilustrada pela figura 15 com valores fictícios, apenas para compreensão do conceito. No eixo horizontal, a porcentagem de utilização de um veículo no mês e o vertical o custo resultante para cada porcentagem de utilização.

Na figura 15 é observada a existência de um ponto de equilíbrio definido pela intersecção das retas *FPT* e *FT*. A partir deste ponto pode-se afirmar que a utilização de veículos com tarifa *FPT* é mais barata e abaixo deste ponto a utilização da tarifa *FT* é a mais viável.

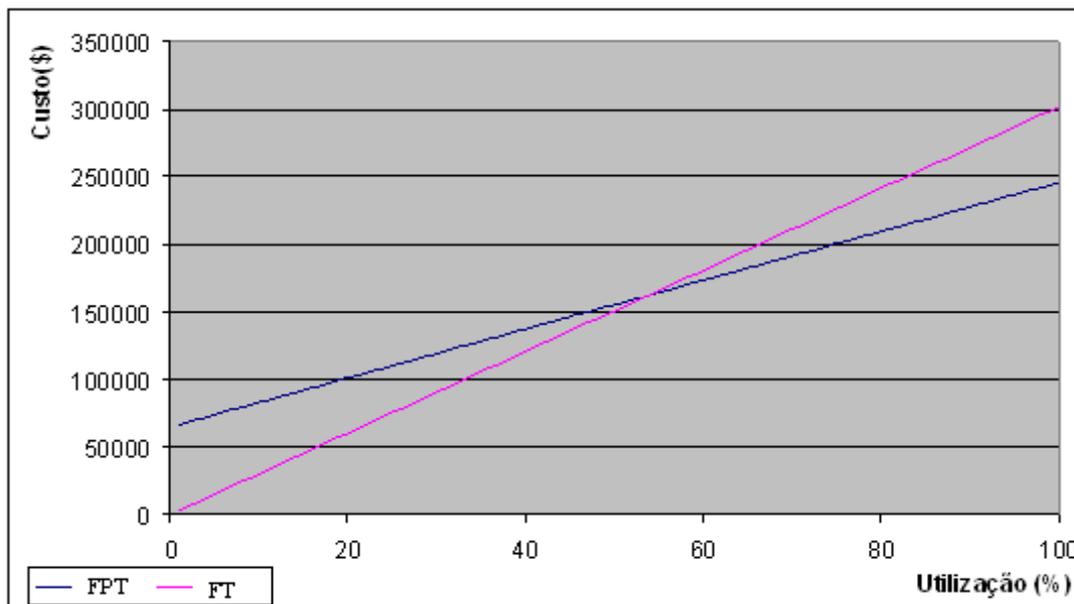


Figura 15 - Tarifas FPT versus FT

3.6 Distância e tempo de viagem

As distâncias utilizadas no estudo de caso são dadas por trechos n no qual são sempre já definidas, já que a rota no qual o veículo percorre é uma rota previamente definida e aprovada por segurança, tendo em vista que o transporte de combustível é um transporte de carga perigosa. Esta é a principal causa do porque o estudo de caso escolhido não trata de um problema de roteirização. Assim a distância é expressa através do parâmetro $Dist(n)$, e representada através do quadro da figura 16 na implementação.

Como exemplo, para uma viagem $vg1$ saindo do terminal $t1$, entregando produto para os clientes $l1$ e $l2$ e regressando ao terminal $t1$, tem-se a seguinte equação 8, onde são dados três trechos n : “ $t1$ para $l1$ ”, “ $l1$ para $l2$ ” e “ $l2$ para $t1$ ”, para completar o percurso total da viagem $vg1$.

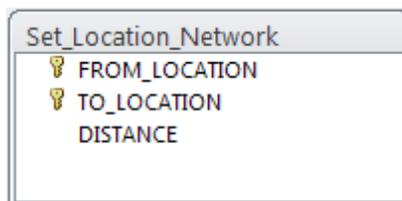


Figura 16 - Quadro de trechos e distâncias utilizado na implementação

$$Distancia_{vg1} = Dist_{(t1\ para\ l1)} + Dist_{(l1\ para\ l2)} + Dist_{(l2\ para\ t1)} \quad (8)$$

Logo depois de encontrada a distância da viagem a ser executada, inicia-se o cálculo do tempo de viagem. A figura 17 ilustra a estrutura da tabela utilizada na implementação para o cálculo do tempo de viagem, onde TRAFFIC_ZONE_ID ou z é o principal índice para definir a velocidade a ser percorrida para cada trecho n do percurso total da viagem. O índice z tem como objetivo mapear regiões e classifica-las para posteriormente serem utilizadas no cálculo do tempo de viagem.

No caso cada cliente l pertence a uma zona de tráfego z , definida pelo parâmetro de entrada $Lz(l,z)$. A forma de cálculo utilizada para estimar o tempo de viagem é apresentada com mais detalhes na seguinte sessão 3.6.1, por se tratar de uma variável de incerteza.

Set_Location_Network_Zone	
🔑	TRAFFIC_ZONE_ID
	VEHICLE_MAX_SPEED
	TRAFFIC_HIGH_CODE
	TRAFFIC_HIGH_PROB
	TRAFFIC_MEDIUM_CODE
	TRAFFIC_MEDIUM_PROB

Figura 17 - Quadro de velocidades utilizado na implementação

3.6.1 Incerteza no tempo de execução

A área de transporte em geral está diretamente relacionada ao ambiente externo ao qual se está sendo utilizado como meio de condução. Este ambiente sofre constantes alterações, causando possíveis variações no tempo de execução das tarefas e é mapeado no presente trabalho através do índice z . O presente estudo de caso trata estes impactos de duas diferentes formas, por intensidade de tráfego no momento da viagem e por peso do veículo.

Pode-se observar que diferente da distância percorrida pelo veículo, o tempo de execução de uma viagem pode variar conforme a hora em que ela irá ser executada. Como exemplo uma viagem executada pelo período da madrugada tende a ter um tempo de duração menor do que caso fosse executada no período da tarde, pois a ocorrência de congestionamento é mais frequente pela tarde do que pela madrugada.

O fator peso do veículo carregado é definido por medidas de segurança da operação, na qual veículos carregados devem trafegar a uma velocidade menor do que veículos vazios. Essa diferença faz com que o tempo de regresso do veículo ao terminal seja menor do que o de ida ao cliente.

O parâmetro $N_{speed}(z)$ define a velocidade máxima possível de ser trafegada na zona z , aonde ambos os fatores apresentados impacta na sua redução. Ou seja, tanto a intensidade de tráfego, como o peso do veículo por trecho percorrido tem como finalidade reduzir o valor $N_{speed}(z)$ inicial para um valor mais próximo do cenário do trecho n que esta sendo percorrido.

A forma na qual os dois fatores são calculados são dados da seguinte forma. Quanto ao parâmetro de penalidade por peso do veículo, este é proveniente das normas de segurança da empresa. Segundo informações da operação, a diferença de velocidade de um veículo v trafegar cheio ou vazio está em torno de 25%. Tendo em vista que $N_{speed}(z)$ é a velocidade máxima que um veículo v pode trafegar em uma viagem vg , é dada a seguinte equação 9 para refletir a penalidade de 25%.

$$Velocidade\ peso_{(z,vg,v)} = N_{speed}(z) * \left[1 - \left(\frac{[V_{w_v} + PesoProduto_{vg}]/4}{V_{maxw_v}} \right) \right] \quad (9)$$

Na equação 9, " $V_{w_v} + PesoProduto_{(vg)}$ " é igual ao peso real do veículo carregado para a viagem vg , em relação ao peso máximo V_{maxw_v} do veículo v ; porém, o peso real do veículo na viagem é dividido por 4 para fazer referência aos 25% de penalidade.

Diferente do fator peso, que é tratado de uma forma aritmética, o fator intensidade de tráfego é tratado sobre uma ótica de incerteza, no qual já mencionado pelo trabalho de Ben-Tal e Nemirovski (2000), é gerado um número randômico para confrontar com uma probabilidade ou de ocorrência de tráfego. Desta forma, são dados como parâmetros de entrada as possíveis horas do dia onde existe a ocorrência de congestionamento do tipo médio $N_{med}(z)$ ou alto $N_{high}(z)$. O estudo de caso define como padrão a ausência dos dois tipos de congestionamento, tomando como base um congestionamento do tipo baixo e que não afeta $N_{speed}(z)$. Para $N_{med}(z)$ e $N_{high}(z)$ existem $N_{medp}(z)$ e $N_{highp}(z)$

respectivamente, indicando a probabilidade de ocorrência de cada um dos tipos de congestionamento. Conforme exemplo ilustrado pela figura 18.

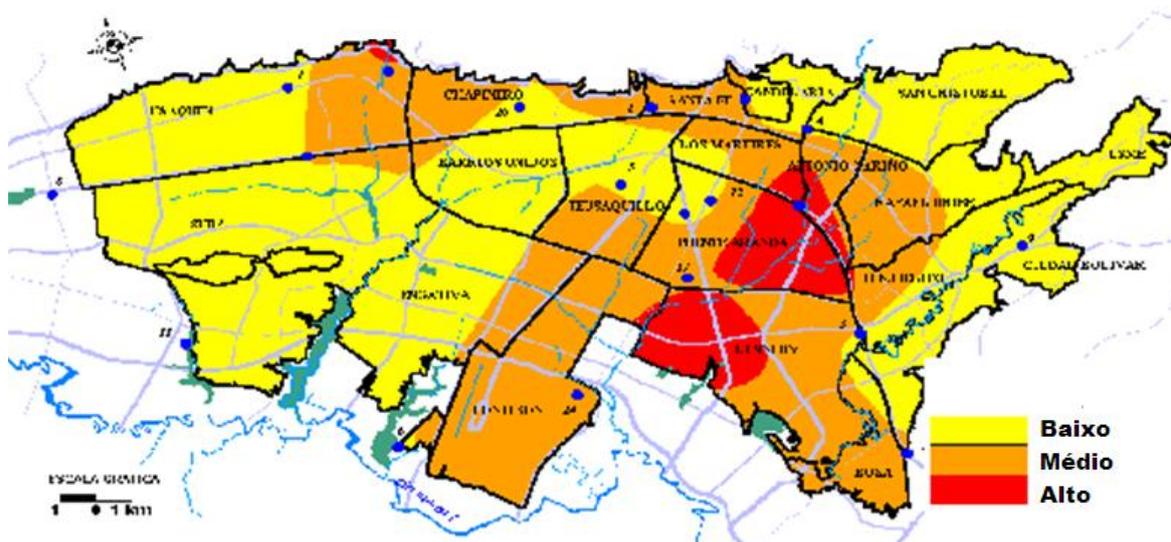


Figura 18 - Exemplo de zonas z de alto, médio e baixo tráfego

Assim como para o cálculo de disponibilidade de produto, é gerada uma variável randômica $rnd(h)$ para cada hora do dia e comparado com as variáveis $Nmedp(z)$ e $Nhighp(z)$, vigentes por $Nmed(z)$ e $Nhigh(z)$. Caso o valor da variável randômica $rnd(h)$ seja menor do que indicado por $Nmedp(z)$ é descontado 20% da velocidade $Nspeed(z)$ e caso esta mesma variável randômica seja também menor do que $Nhighp(z)$ é descontado 35% da velocidade $Nspeed(z)$ original.

Os valores de penalidade de 20% para congestionamento médio e 35% para congestionamento alto foram retirados como referência da pesquisa IPEA e ANTP (1997). No estudo citado foi levantado o valor de retardamento, mencionado da seguinte maneira, "os retardamentos estão na maioria dentro da faixa de 25% a 30% do tempo total de percurso. Estes valores devem ser vistos dentro de uma certa margem de tolerância estatística, uma vez que o levantamento de dados foi feito apenas em um dia e considerando que as oscilações dos valores medidos são normalmente abruptas, devido à variação das condições de tráfego". Tomando este comentário como base foi adicionada uma margem de 5% de variação para menos e para mais, para os níveis de congestionamento médio e alto respectivamente.

Depois de gerados os dois fatores que penalizam $N_{speed}(z)$, o valor final da velocidade do veículo em um trecho n , a uma determinada hora do dia, será o menor valor dentre eles encontrado.

A fim de esclarecer o método utilizado são dadas a seguir as tabelas de 6 a 9, exemplificando passo a passo o processo de cálculo utilizado. O exemplo contempla o cálculo do tempo de execução de uma viagem $vg1$, saindo do terminal $t1$ e entregando para o cliente $l1$ que pertence à zona de tráfego $z1$. Os seguintes parâmetros de entrada são dados:

- $N_{speed}(z1)$ é igual a 50km/h de velocidade máxima, alcançada em tráfego considerado baixo, comparado ao padrão;
- $N_{med}(z1)$ é a janela horária, que será igual das 04h00 às 17h00, para tráfego médio;
- $N_{high}(z1)$ é igual das 01h00 às 04h00 e 17h00 às 19h00 para tráfego alto;
- $N_{medp}(z1)$ é igual a 50% de probabilidade de ocorrência de tráfego médio;
- $N_{highp}(z1)$ é igual a 80% de probabilidade de ocorrência de tráfego alto.

Iniciando pela tabela 6 é apresentado o cálculo de incerteza para estimar a velocidade a ser utilizada a cada hora do dia na viagem $vg1$. A leitura da tabela pode ser interpretada de cima para baixo, considerando o valor de velocidade padrão $N_{speed}(z1)$ e um número randômico $rnd(h)$ para cada hora h do dia.

Logo após para as horas h aonde se encontrado o tráfego de média intensidade $N_{med}(z)$ é considerado a probabilidade de ocorrência $N_{medp}(z1)$ e comparada com os valores $rnd(h)$ referentes para cada hora. Para os valores $rnd(h)$ menores que $N_{medp}(z1)$ é encontrada uma nova velocidade para cada hora h do dia, tendo como resultado uma velocidade 20% menor que $N_{speed}(z1)$ original e igual a 40km/h.

O mesmo passo é repedido para o tráfego de alta intensidade $N_{high}(z1)$ com probabilidade $N_{highp}(z1)$, porém com uma redução de 35% do valor original de $N_{speed}(z1)$ e resultado igual a 33km/h.

E, por fim, na tabela 6 é selecionado como valor de velocidade final o menor valor encontrado para cada hora h entre $N_{speed}(z1)$ original, $N_{speed}(z1)$ penalizado por tráfego médio e $N_{speed}(z1)$ penalizado por tráfego alto.

Tabela 6 - Cálculo de velocidade de tráfego

Passo \ h	0:00	1:00	2:00	3:00	4:00	5:00	6:00	7:00	8:00	9:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00	
Velocidade Padrão Nspeed(z1)	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
rnd(h) (gerados pelo sistema)	0.32	0.44	0.49	0.32	0.30	0.60	0.12	0.56	0.91	0.27	0.15	0.84	0.65	0.31	0.27	0.14	0.51	0.98	0.07	0.18	0.96	0.83	0.35	0.07	
Tráfego médio	Nmed(z1)	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	
	Nmedp(z1)					0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50	0.50							
	Se rnd(h) < Nhighp(z1) então Nspeed(z1) - 20%					40	50	40	50	50	40	40	50	50	40	40	40	50	50						
Tráfego alto	Nhigh(z1)	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0
	Nhighp(z1)			0.80	0.80	0.80													0.80	0.80	0.80				
	Se rnd(h) < Nmedp(z1) então Nspeed(z1) - 35%			33	33	33													50	33	33				
Velocidade de tráfego final (z1, h) = menor valor entre (Nspeed(z1); Nspeed(z1, h) alto; Nspeed(z1, h) médio)	50	50	33	33	33	50	40	50	50	40	40	50	50	40	40	40	50	50	33	33	50	50	50	50	

Conforme a tabela 7, o próximo passo será o cálculo da velocidade penalizada pelo peso do veículo carregado. Utilizando-se a equação 9 é possível observar que o sistema penaliza em 25% a velocidade Nspeed(z1) original para o veículo carregado na ida e 0% na volta. Obtendo uma velocidade estimada de 38km/h para o veículo quando carregado.

Na tabela 7 também é possível observar que o sistema já assume um tempo de viagem inicial para a estimativa do cálculo por peso. Este tempo inicial é calculado com base em Nspeed(z1) original.

Tabela 7 - Cálculo de velocidade peso

Passo \ h	0:00	1:00	2:00	3:00	4:00	5:00	6:00	7:00	8:00	9:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00
Velocidade Padrão Nspeed(z1)	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
Proposta de horário da viagem vg1 s/ restrições, com velocidade Nspeed(z1)	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Período da viagem vg1 no veículo v1					Ida (Veículo Carregado)		Volta (Veículo descarregado)																	
Velocidade peso (vg1, v)					38	38	38	38	50	50	50													

Para o período de tempo estimado da viagem vg1, utilizando Nspeed(z1), o sistema irá tomar como velocidade final para a hora h , o menor valor encontrado entre a restrição por intensidade de tráfego e peso. Após a identificação destas velocidades finais por hora h , é calculada uma velocidade média final para toda a viagem. Conforme apresentando pela tabela 8.

Tabela 8 - Cálculo de velocidade média

Passo \ h	0:00	1:00	2:00	3:00	4:00	5:00	6:00	7:00	8:00	9:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00
Velocidade Padrão Nspeed(z1)	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
Velocidade de trafego final (z1,h)	50	50	33	33	33	50	40	50	50	40	40	50	50	40	40	40	50	50	33	33	50	50	50	50
Velocidade peso (vg1,v)					38	38	38	38	50	50	50													
Velocidade final = menor valor entre (Velocidade de trafego e peso)					33	38	38	38	50	40	40													
Velocidade média final a ser utilizada								39.3																

Tendo em vista que o cálculo do tempo de viagem está sujeito a diversas incertezas, foi estipulado que a velocidade média final 39,3km/h será a utilizada para calcular o tempo final da viagem *vg1*. A tabela 9 apresenta a diferença entre o tempo de viagem original baseado em Nspeed(z1) igual a 50km/h e o final igual a 39km/h. Nesta tabela também é possível observar que a presença de incerteza causou um aumento de duas horas no tempo total de viagem previsto.

Tabela 9 - Cálculo final do tempo de viagem

Passo \ h	0:00	1:00	2:00	3:00	4:00	5:00	6:00	7:00	8:00	9:00	10:00	11:00	12:00	13:00	14:00	15:00	16:00	17:00	18:00	19:00	20:00	21:00	22:00	23:00
Velocidade Padrão Nspeed(z1)	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
Proposta de horário da viagem <i>vg1</i> s/ restrições, com velocidade Nspeed(z1) = 50km/h	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Horário final da viagem <i>vg1</i> , com velocidade = 39.3km/h	0	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0

3.7 Incerteza na demanda

O trabalho de Toktas e Yen (2004) propõe duas diferentes abordagens para a determinação de incertezas na capacidade da demanda. A primeira utilizando a aproximação da capacidade de um mesmo recurso. A segunda encontrando soluções determinísticas sobre um conjunto de recursos e utilizando soluções intermediárias durante a construção de uma solução única.

Para ambas as abordagens foram utilizadas uma variação da metodologia tipo GAP, chamada de CCGAP, *collectively capacitated generalized assignment problem*, diferenciado principalmente pela possibilidade de alocação de tarefas para mais de um agente, conforme a equação 10.

Observa-se uma similaridade com a equação 2 de GAP apresentada anteriormente, porém sujeita a uma diferente equação 11a comparada à 3a, no qual a_{ijr} representa a capacidade usada de um grupo de recurso r quando a tarefa i é alocada a um agente j .

$$\text{Minimizar } Z = \sum_{i=1}^n \sum_{j=1}^m p_{ij} x_{ij} \quad (10)$$

Sujeito a

$$\sum_{i=1}^n \sum_{j=1}^m a_{ijr} x_{ij} \leq b_{ij}, \quad j \in \{1, \dots, m\}, \quad (11a)$$

$$\sum_{j=1}^m x_{ij} \leq 1, \quad i \in \{1, \dots, m\}, \quad (11b)$$

$$x_{ij} \in \{0,1\}, \quad i \in \{1, \dots, n\}, \quad j \in \{1, \dots, m\} \quad (11c)$$

Traçando um paralelo entre a idéia de coletividade proposta por Toktas (2004) para a geração de incerteza na capacidade de entrega, é possível considerar que, no presente caso estudado, cada dia de operação pode ser considerado uma demanda individual. É possível recriar este mesmo efeito de grupos de trabalho proposto, com a criação de vários cenários de mais de um dia de entrega, fazendo com que os pedidos dos clientes possam ser entregues em dias diferentes da data solicitada, porém com certa tolerância. Os parâmetros já citados que definem estas características são dados pelos campos $L_{\max}(l)$ e $P_{\max}(o,l)$, respectivamente. $L_{\max}(l)$ é o número máximo de vezes que é possível alterar o pedido do cliente por execução do programa e $P_{\max}(o,l)$ o valor em dias da variação que o pedido do cliente pode variar para mais e para menos, definido como padrão igual a 1 para cenários com a presença de incerteza e 0 na ausência de incerteza.

Com a geração de cenários onde $P_{\max}(o,l)$ é igual a 0, indica que a demanda a ser entregue naquele dia é determinada e caso sobre ou falte capacidade de entrega o resultado do problema continua o mesmo.

Para $P_{\max}(o,l)$ igual a 1 o cenário adquire a capacidade de adiantar pedidos dos clientes ou de atrasar em caso de falta de capacidade de entrega. Por esta razão é necessário executar os cenários, com presença de incerteza, em grupos de pelo menos 2 em 2 dias consecutivamente. É necessário excluir os pedidos que já foram entregues com antecedência ou incluir os pedidos atrasados do dia anterior.

Concluindo a descrição de incerteza na demanda, vale ressaltar que adicionalmente à metodologia apresentada, a incerteza de demanda também ocorre pela indisponibilidade de produto conforme apresentado na seção 3.3.1. É importante considerar que ao mencionar nos seguintes capítulos a presença ou ausência de incerteza na demanda o trabalho está se referindo aos dois itens apresentados, incerteza de capacidade da demanda e indisponibilidade de produto.

4 Metodologia

O estudo de caso proposto neste trabalho é identificado como um problema de pesquisa operacional complexo, que exige um grande nível de processamento de informações. Existem diferentes metodologias aplicadas para a solução do problema, tanto de forma heurística como determinística. Desta forma, é proposta a solução do problema através da Busca Tabu, para posterior comparação entre as soluções encontradas.

A busca da solução do problema é fracionada em diferentes etapas de execução, detalhadas através da seção 4.1. Na seção 4.2, a arquitetura utilizada na implementação é apresentada.

Visando aprofundar ainda mais o entendimento dos impactos do estudo realizado, o presente trabalho inclui o desenvolvimento em linguagem de programação da implementação da solução, sem o auxílio de *softwares* de otimização. Considerações adicionais sobre a parte técnica são feitas na seção 4.3.

Logo após as descrições do presente capítulo é iniciada a fase de testes e a execução da ferramenta, apresentada no capítulo 5, que contém as informações dos dados que foram utilizados e cenários gerados para análise, assim como seus respectivos resultados.

4.1 Metodologia utilizada

O objetivo geral do problema é a minimização dos custos de operação na seleção de veículos, consolidação das cargas para cada cliente e do tipo de pagamento de frete utilizado, seja no nível operacional ou tático.

Dentro do estudo de caso proposto foram identificadas duas etapas importantes de cálculo, que demandam um grande número de iterações para a resolução. A primeira, chamada de Compartmentalizador, é o cálculo de consolidação das cargas nos compartimentos dos veículos. A segunda, denominada de Alocador, é referente à alocação das viagens já definidas como factíveis pelo resultado da primeira etapa.

Tendo em vista que o Alocador necessita do resultado do Compartmentalizador para fazer a otimização da programação de entregas, o

primeiro passo do Alocador é a leitura dos dados do Compartimentalizador. Assim, o Compartimentalizador dá origem a uma lista de viagens factíveis de serem executadas (compartimentalizadas). Posteriormente, esta lista é utilizada pelo Alocador para gerar a programação final de entrega, com até três pedidos de entregas por viagem. O processo é apresentado de forma sucinta pela figura 19. Nela é possível observar os quadros 2 e 4 representando etapas do processo de busca e os quadros 1, 3 e 5, que representam dados de entrada e saída entre as etapas.

Iniciando pelo primeiro quadro da figura 19 são encontrados todos os parâmetros e variáveis descritos no capítulo 3, como dados de entrada para a resolução do problema. No quadro número 2, a primeira etapa do processo de busca é executada, referente à consolidação dos pedidos aos veículos, apresentado com mais detalhes na seção 4.1.1. O objetivo principal desta primeira etapa é gerar uma lista de viagens factíveis de serem compartimentalizadas. O resultado desta primeira etapa é representado pelo quadro 3.

Ainda na figura 19, após o sistema finalizar esta primeira parte do processo, ele dá início à segunda etapa, representada pelo quadro 4, chamado de Alocador de viagens. A etapa 2 do processo tem como objetivo arranjar todos os pedidos solicitados inicialmente, de forma que satisfaçam as restrições do problema. Para tal objetivo a etapa 2 utiliza o resultado da etapa 1 como base de dados para fazer a alocação final das viagens aos veículos. O processo referente à etapa 2 é apresentado com mais detalhes na seção 4.1.2 e o resultado desta etapa é apresentado pelo quadro 5, sendo este o resultado final da programação de entregas.

Para um melhor entendimento dos processos das etapas 1 e 2, é apresentada a figura 20, ilustrando na forma de fluxograma a iteração entre as duas etapas.

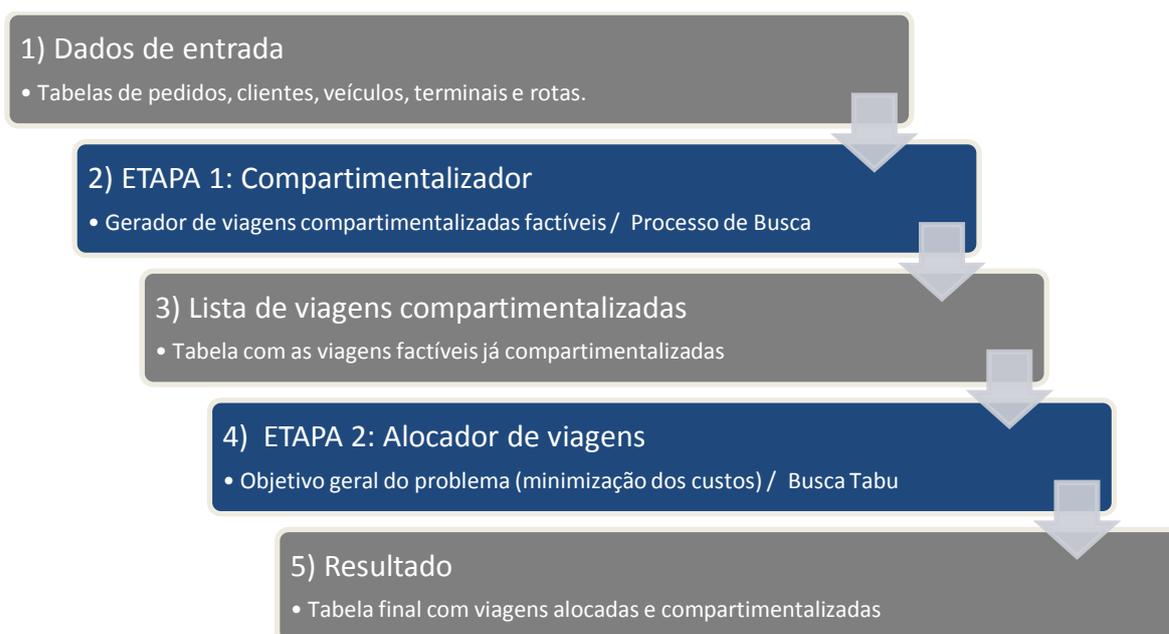


Figura 19 - Processo de busca

A etapa 1 da figura 20 tem como objetivo a geração de uma lista de viagens compartimentalizadas factíveis, onde é necessário executar o algoritmo de compartimentalização inúmeras vezes para que a busca das viagens possa ser executada. Desta forma, o Compartimentalizador executa sua busca inúmeras vezes, gerando uma lista das viagens factíveis, conforme a demanda. Ao final de cada procedimento de Intensificação da busca, os resultados factíveis do Compartimentalizador são incluídos à lista de viagens compartimentalizadas.

Lembrando que a lógica do algoritmo de compartimentalização é apresentada na seção 4.1.1. A lógica apresentada na figura 20 é a de um procedimento de busca executado no Compartimentalizador. Na primeira etapa, o procedimento de busca é encerrado apenas quando a lista de pedidos solicitados cessa. A busca executada pelo Compartimentalizador analisa a possível combinação de até três pedidos por cada veículo v .

Finalizado o processo da etapa 1, a etapa 2 é iniciada utilizando o resultado da primeira etapa como base de dados. Já a etapa 2 é executada para a alocação das viagens com base em Busca Tabu, que é apresentada com mais detalhes na seção 4.1.2.

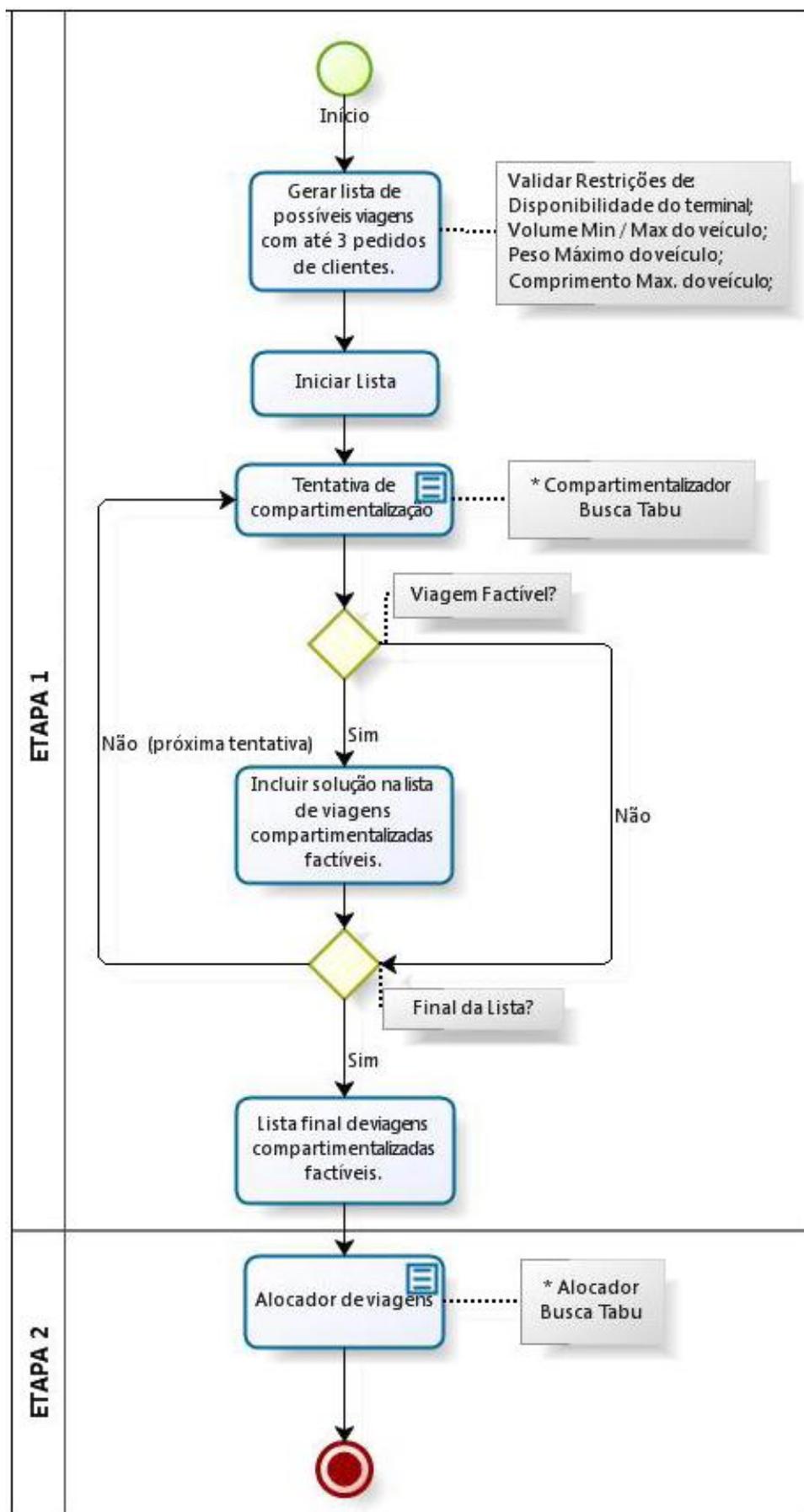


Figura 20 - Etapa 1 – Procedimento de busca

4.1.1 Etapa 1 - Compartimentalização

A forma na qual o Compartimentalizador consolida as cargas dos pedidos é através da utilização de um método de busca. Lembrando que, como o objetivo da primeira etapa é gerar uma lista de viagens compartimentalizadas factíveis, o método repete várias vezes a busca por novas soluções factíveis. Ou seja, a busca “chama” inúmeras vezes o método para que ela tente alocar um dado pedido em um veículo v , consistindo na tentativa de acomodação de até três pedidos de clientes em uma viagem vg .

Nesta etapa do processo são analisadas as seguintes restrições do estudo de caso proposto:

- i) A frota de veículos é heterogênea e cada veículo apresenta restrições de volume e peso máximos definidos por compartimento;
- ii) Cada veículo possui diferentes possibilidades de configurações de compartimentos, podendo ser carregados somente dentro das restrições de segurança e seguindo uma métrica de dosagem definida por compartimento.
- iii) A forma de descarregamento de produtos dos veículos, nos clientes, é com a utilização da força da gravidade, via escoamento, porém existindo alguns veículos que apresentam bomba para forçar a vazão do produto;
- iv) Os clientes possuem restrições quanto ao recebimento do veículo por tamanho, volume e presença de bomba para descarregamento;
- v) A forma de carregamento de produtos dos veículos, nos terminais, é por padrão TL (*Top Loading*), existindo para alguns caminhões uma segunda opção via BL (*Bottom Loading*).
- vi) A demanda dos clientes por produto é incerta e pode ser maior que a capacidade dos veículos, fazendo com que o código altere o volume original do pedido dentro de uma tolerância;
- vii) Para casos com incerteza na demanda são geradas adicionalmente à lista, novas possibilidades de entrega, para datas dentro do período de

tolerância de um dia para mais e para menos em relação à data original.

A figura 21 descreve em fluxograma a forma na qual o Compartimentalizador opera. É importante ressaltar que ele segue uma lógica inspirado na Busca Tabu já apresentada pela figura 2 na seção 2.3 do presente trabalho. Para o método de busca apresentada pela figura 21, em relação à figura 2, foram acrescentados itens que customizam a Busca Tabu para compartimentalizar os pedidos aos veículos gerando assim o método de busca apresentado.

Cesc é um contador do número de iterações para registrar o número de iterações executadas pelo procedimento de “intensificação” sem o melhoramento do resultado da melhor solução já encontrada. Quando *Cesc* atinge um determinado número de iterações *Nesc*, o código dá início a uma nova etapa da “intensificação”, onde tenta-se distribuir as cargas excedentes num compartimento para outro compartimento vazio. Esta medida evita que o método fique preso em uma determinada região de solução e não sejam analisadas outras possibilidades (ou regiões de busca). Considerando que *N* é o número máximo de iterações que a busca irá executar e *i* a iteração executada no momento, são dados os seguintes passos:

Passo 1. Carregar dados do problema e contadores → “Entrar” com as ordens dos pedidos *o* e o código do veículo *v*, para que os produtos das ordens sejam distribuídos nos compartimentos *cpi* do veículo *v* selecionado;

Passo 2. Gerar solução Inicial ($i = 1$ e $Cesc = 0$) → Distribuir nos compartimentos do veículo o volume total por produto, respeitando restrições de peso máximo permitido. A restrição de peso é satisfeita no passo 2, deixando assim a restrição de volume por compartimento a ser explorada e satisfeita para os seguintes passos;

Passo 3. Intensificação - explorar a vizinhança da última solução gerada → Explorar compartimentos que estão abaixo ou acima do permitido por volume. Inicialmente, se $Nesc > Cesc$, realizar o deslocamento do conteúdo de um compartimento cheio para um vazio. Se $Nesc \leq Cesc$, retirar volume de um compartimento com excedente e mover para

compartimento com déficit. Ou seja, fazer movimentações de volume ainda não exploradas, como combinar o volume utilizado em dois compartimentos em apenas um, ou trocar entre dois compartimentos o volume e produto utilizados;

Passo 4. Verificar solução encontrada em relação à função objetivo;

Passo 5. Atualizar contador $Cesc$, se a melhor solução (encontrada na iteração i) for inferior à melhor solução encontrada desde a geração da solução inicial (ou parada). Caso seja encontrada a melhor solução na iteração i , o contador $Cesc$ será zerado;

Passo 6. Atualizar contador de iterações: $i = i + 1$;

Passo 7. Se $Nesc > Cesc$, voltar para o passo 3, usando o procedimento inicial de intensificação. Senão, altera o procedimento de intensificação (fazendo movimentações de volume ainda não exploradas) e segue para o passo 8;

Passo 8. Se $N = i$ (critério de parada foi atingido), finaliza a busca e salva a solução na lista de viagens compartimentalizadas factíveis. Caso contrário, ir para o passo 4 (atualiza lista de agrupamento), antes de reiniciar a busca para outra combinação de acomodação do pedido no veículo (a partir do passo 3).

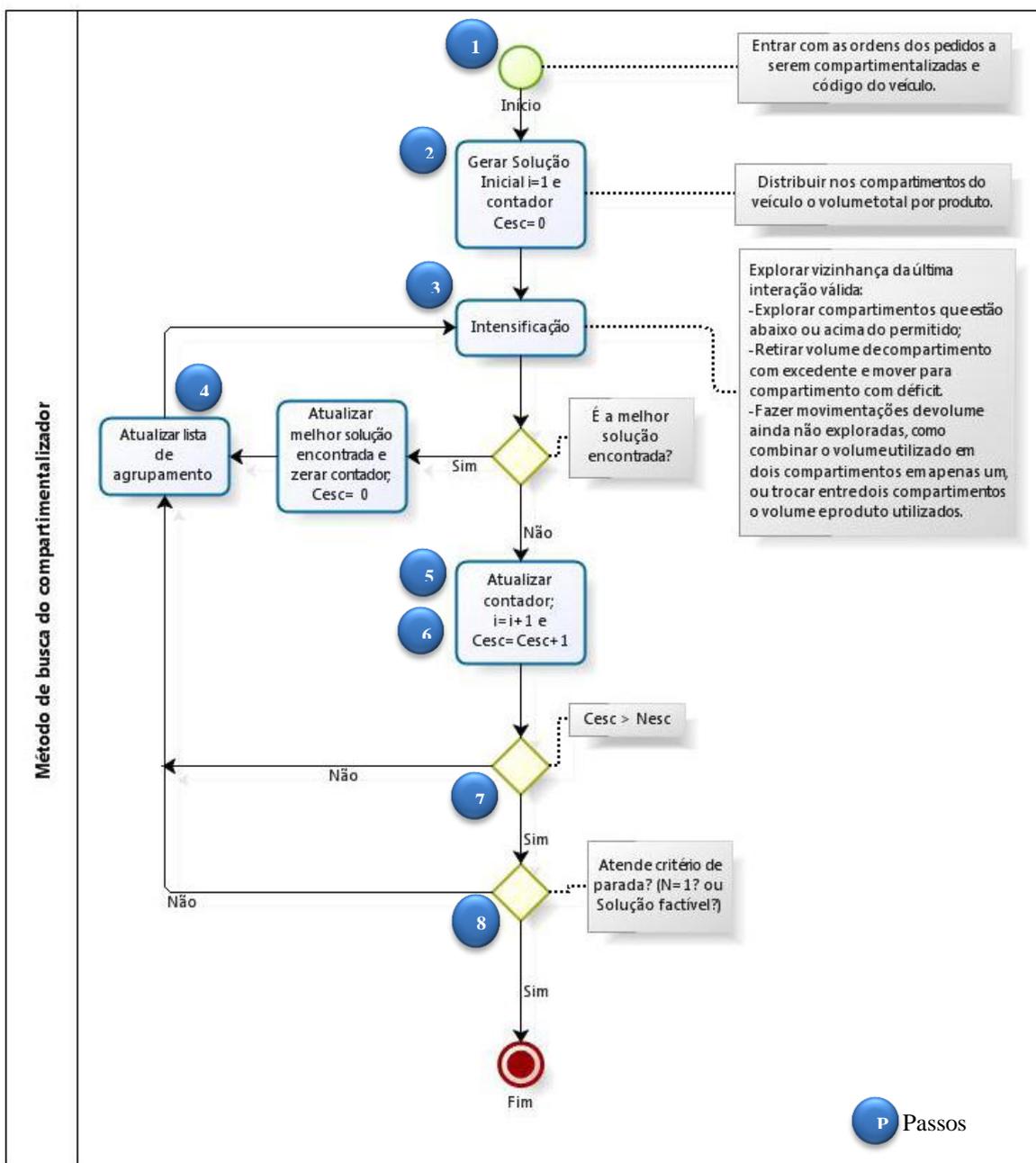


Figura 21 - Lógica do método de busca do Compartimentalizador

• Exemplo de Funcionamento do Compartimentalizador

A fim de esclarecer o processo de compartimentalização de um veículo apresentado no fluxograma da figura 21, é dado o exemplo a seguir. Duas ordens de entrega ($o1$ e $o2$) de dois clientes ($I1$ e $I2$), com seus respectivos volumes solicitados $Pvol(o,p)$ são recebidas, sendo p o índice que define o código do produto, conforme os dados mostrados na tabela 10. As ordens devem ser

compartimentalizadas em um veículo v_1 com quatro compartimentos (cpi_1 , cpi_2 , cpi_3 e cpi_4), com o volume máximo por compartimento $VC_{max}(cpi)$, que são apresentados na tabela 11. O peso máximo de carregamento de produto no veículo (VC_{maxw}) é igual a 28200kg.

Tabela 10 - Ordens o_1 e o_2

Ordem o	Cliente l	Produto p	Volume $P_{vol}(o,p)$ (UGL ³)	Peso por produto (kg ⁴)
o_1	l_1	p_1	3000	8371,83
o_1	l_1	p_2	3000	9627,81
o_2	l_2	p_1	4000	11356,24

Tabela 11 - Veículo v_1

Veículo v	Compartimento cpi	Volume $V_{max}(v)$ (UGL)	Marcação volumétrica $VC_{mark}(cpi)$ (UGL)
v_1	cpi_1	3000	100
v_1	cpi_2	2000	100
v_1	cpi_3	2000	100
v_1	cpi_4	3000	100

Após o carregamento dos dados das tabelas 10 e 11, inicia-se o procedimento de busca indicado na figura 20. Posteriormente, o Compartimentador descrito na figura 21, é acionado para dar início ao processo de compartimentalização, apresentado na tabela 12. Caso o resultado final seja factível, a solução gerada alimenta a lista de viagens compartimentalizadas factíveis.

É importante mencionar que a função objetivo utilizada nesta Busca Tabu é maximizar o volume total factível, ou seja, a função objetivo é igual ao volume total já designado ao compartimento e que esteja dentro dos limites de volume máximo VC_{max} do mesmo. Outro ponto foi a adição de um critério de parada à Busca Tabu. Caso o procedimento de busca já tenha testado todas as combinações de pedidos possíveis, a lista de soluções geradas é tomada como resultado, finalizando a busca. Para facilitar a leitura da tabela 12, ela é descrita e dividida entre as tabelas de 13 a 15.

³UGL: representa galões

⁴kg: representa quilos

A tabela 12 deve ser lida de cima para baixo e acompanhando os comentários explicativos situados à direita. A primeira coluna identifica as linhas (apenas como referência), enquanto a segunda refere-se ao número da iteração i que está sendo executada. A terceira coluna identifica o passo referente ao processo na descrição do fluxograma da figura 21. A quarta e quinta colunas são, respectivamente, os resultados da função objetivo e o peso total do veículo com produto carregado. E, finalmente, nas colunas de cpi1 a cpi4, são apresentados os compartimentos nos quais os produtos devem ser alocados, com seus respectivos volumes VCmax disponíveis para cada compartimento.

Neste exemplo, o valor de Nesc é 2 e o número de iterações igual a 1, os valores que apresentam asterisco na tabela 19 simbolizam ser “Tabu” e não podem ser movimentados por 1 iteração.

Tabela 12 - Exemplo de Compartimentalização

Linha	Iteração	Passo	Função Objetivo	Peso Total (Kg)	Cvmax e ordens por produto já alocadas nos compartimentos (cpi1, cpi2, cpi3, cpi4)				Descrição
					cpi1 3000 -	cpi2 2000 -	cpi3 2000 -	cpi4 3000 -	
1	1	1	3000	8517,18	3000 o1 / p1	-	-	-	
2	1	1	5000	18145	3000 o1 / p1	3000 o1 / p2	-	-	Distribuir Volume das ordens o1 e o2 por compartimento.
3	1	1	7000	29501,2	3000 o1 / p1	3000 o1 / p2	4000 o2 / p1	-	
4	1	2	6900	29217,3	2900 o1 / p1	3000 o1 / p2	4000 o2 / p1	-	
5	1	2	6800	28933,4	2800 o1 / p1	3000 o1 / p2	4000 o2 / p1	-	
6	1	2	6700	28649,5	2700 o1 / p1	3000 o1 / p2	4000 o2 / p1	-	Ajustar Volume com o peso máximo Vcmaxw = 28200kg permitido pelo veículo v. (Peso Total deve ser <= Vcmaxw)
7	1	2	6700	28365,6	2700 o1 / p1	3000 o1 / p2	3900 o2 / p1	-	
8	1	2	6700	28081,7	2700 o1 / p1	3000 o1 / p2	3800 o2 / p1	-	
9	1	3	7700	28081,7	2700 o1 / p1	3000 o1 / p2	-	3800 o2 / p1	
10	1	3	7700	28081,7	2700 o1 / p1	-	3800 o2 / p1	3000 o1 / p2	Movimento (intensificação): explorar movimentação dos compartimentos cheios para o vazio cpi4 .
11	1	3	6700	28081,7	-	3000 o1 / p2	3800 o2 / p1	2700 o1 / p1	
12	1	4	7700	28081,7	2700 o1 / p1	3000 o1 / p2	-	3800 o2 / p1	Atualizar melhor solução encontrada. Movimento da linha 9 escolhido, de cpi3 para cpi4 .
13	2	3	7000	28081,7	-	3000 o1 / p2	2700 o1 / p1	3800 o2 / p1	
14	2	3	7700	28081,7	2700 o1 / p1	-	3000 o1 / p2	3800 o2 / p1	Movimento (intensificação): explorar movimentação dos compartimentos cheios para o vazio cpi3 .
15	2	3	6700	28081,7	2700 o1 / p1	3000 o1 / p2	3800 o2 / p1	-	
16	2	4	7700	28081,7	2700 o1 / p1	-	3000 o1 / p2	3800 o2 / p1	Atualizar melhor solução encontrada. Movimento da linha 14 escolhido, de cpi2 para cpi3 .

17	3	3	7000	28081,7	-	2700	3000	3800	
					-	o1 / p1	o1 / p2	o2 / p1	Movimento (intensificação): <u>explorar</u>
18	3	3	7700	28081,7	2700	3000	-	3800	movimentação dos compartimentos
					o1 / p1	o1 / p2	-	o2 / p1	cheios para o vazio cp12 .
19	3	3	6700	28081,7	2700	3800	3000	-	
					o1 / p1	o2 / p1	o1 / p2	-	
20	3	4	7700	28081,7	2700	3000	-	3800	Atualizar melhor solução encontrada.
					o1 / p1	o1 / p1	-	o2 / p1	Movimento da linha 18 escolhido, de cp13
									para cp12 .
21	4	3	7000	28081,7	-	3000	2700	3800	
					-	o1 / p2	o1 / p1	o2 / p1	Movimento (intensificação): <u>explorar</u>
22	4	3	7700	28081,7	2700	-	3000	3800	movimentação dos compartimentos
					o1 / p1	-	o1 / p2	o2 / p1	cheios para o vazio cp13 .
23	4	3	6700	28081,7	2700	3000	3800	-	
					o1 / p1	o1 / p2	o2 / p1	-	
24	4	4	7700	28081,7	2700	-	3000	3800	Atualizar melhor solução encontrada.
					o1 / p1	-	o1 / p2	o2 / p1	Movimento da linha 22 escolhido, de cp12
									para cp13 .
25	5	3	7000	28081,7	-	2700	3000	3800	
					-	o1 / p1	o1 / p2	o2 / p1	Movimento (intensificação): <u>explorar</u>
26	5	3	7700	28081,7	2700	3000	-	3800	movimentação dos compartimentos
					o1 / p1	o1 / p2	-	o2 / p1	cheios para o vazio cp12 . (Obs: Porém,
									todos estes movimentos já foram testados
27	5	3	6700	28081,7	2700	3800	3000	-	anteriormente)
					o1 / p1	o2 / p1	o1 / p2	-	
28	5	3	8400	28081,7	2700	1800	3000	2000	
					o1 / p1	o2 / p1	o1 / p2	o2 / p1	Movimento: <u>dividir</u> produto do
29	5	3	8400	28081,7	2700	1900	3000	1900	compartimento mais cheio cp14 para o
					o1 / p1	o2 / p1	o1 / p2	o2 / p1	mais vazio cp12 .
30	5	3	8400	28081,7	2700	2000	3000	1800	VCmark(v, cp1) = 100 e VCminp(v, cp1) = 85%
					o1 / p1	o2 / p1	o1 / p2	o2 / p1	
31	5	4	8400	28081,7	2700	1800	3000	2000	Atualizar melhor solução encontrada.
					o1 / p1	o2 / p1	o1 / p2	o2 / p1	Movimento da linha 28 escolhido, de cp14
									parcial para cp12 .
32	6	3	8800	28081,7	3000	1800	2700	2000	
					o1 / p2	o2 / p1	o1 / p1	o2 / p1	Movimento (intensificação): <u>explorar</u>
33	6	3	8500	28081,7	2700	3000	1800	2000	movimentação do compartimento mais
					o1 / p1	o1 / p2	o2 / p1	o2 / p1	cheio cp13 para os demais
34	6	3	9200	28081,7	2700	1800	2000	3000	compartimentos. (permuta)
					o1 / p1	o2 / p1	o2 / p1	o1 / p2	
35	6	4	9200	28081,7	2700	1800	2000	3000	Atualizar melhor solução encontrada.
					o1 / p1	o2 / p1	o2 / p1	o1 / p2	Movimento da linha 34 escolhido,
									permuta entre cp13 e cp14 .
36	7	Fim	9200	28081,7	2700	1800	2000	3000	Compartimentalização factível
					o1 / p1	o2 / p1	o2 / p1	o1 / p2	

Iniciando o processo pelo passo 1, linhas de 1 a 3 mostrado na tabela 13, as ordens por produto listados na tabela 10 são designadas sequencialmente por compartimento cp1 a cp4, gerando a solução inicial. Neste passo não há a necessidade do volume e peso designado por compartimento ser ainda factível, pois para este passo o objetivo é verificar se o número de compartimentos do veículo suporta a quantidade de diferentes combinações de produtos por cliente, solicitados na ordem de entrega.

Tabela 13 - Exemplo de Compartimentalização – linhas 1 a 3

Linha	Interação	Passo	Função Objetivo	Peso Total (Kg)	Cvmax e ordens por produto já alocadas nos compartimentos (cpi1, cpi2, cpi3, cpi4)				Descrição
					cpi1 3000 -	cpi2 2000 -	cpi3 2000 -	cpi4 3000 -	
1	1	1	3000	8517.18	3000 o1 / p1				
2	1	1	5000	18145	3000 o1 / p1	3000 o1 / p2			Distribuir Volume das ordens o1 e o2 por compartimento.
3	1	1	7000	29501.2	3000 o1 / p1	3000 o1 / p2	4000 o2 / p1		

Como segundo passo, ilustrado na tabela 14 pelas linhas de 4 a 8, o objetivo é adequar o peso total dos produtos ao peso máximo suportado pelo veículo. Como o parâmetro $V_{Cmark}(cpi)$ é igual a 100UGL, os volumes são ajustados de 100 em 100UGL para adequar-se ao peso permitido do veículo. Dando início pela linha 4, o volume a ser entregue de 3000UGL é alterado para 2900UGL, baixando o peso total do veículo de 29501,2Kg para 29217,3Kg. Porém, como o peso máximo permitido V_{cmaxw} é igual a 28200kg, este processo deve repetir-se até ser menor ou igual ao peso permitido. Seguindo esta lógica, o compartimento $cpi1$ é alterado até 2700UGL de produto (conforme mostra a linha 6); porém, não mais que 2700UGL por motivos de segurança. Vale lembrar que o volume mínimo carregado permitido por compartimento é igual a 85% do volume total do compartimento. Após o sistema finalizar a alteração de um produto da primeira ordem de entrega, no caso $o1$, ele passa a alterar a próxima ordem, no caso $o2$, conforme linhas de 6 a 8. Caso não haja uma segunda ou terceira ordem de pedido a ser alterada, a busca continuará “penalizando” a ordem vigente; no caso, $o1$.

Quando o peso total do veículo estiver ajustado ao peso máximo permitido, V_{cmaxw} , as ordens designadas por compartimento estão prontas para serem redistribuídas em outras possíveis combinações. Sem a necessidade de reavaliar o peso permitido em futuros passos da busca.

Tabela 14 - Exemplo de Compartimentalização – linhas 4 a 8

Linha	Interação	Passo	Função Objetivo	Peso Total (Kg)	Cvmax e ordens por produto já alocadas nos compartimentos (cpi1, cpi2, cpi3, cpi4)				Descrição
					cpi1 3000	cpi2 2000	cpi3 2000	cpi4 3000	
4	1	2	6900	29217,3	2900 o1/p1	3000 o1/p2	4000 o2/p1	-	Ajustar Volume com o peso máximo Vcmaxw = 28200kg permitido pelo veículo v. (Peso Total deve ser <= Vcmaxw)
5	1	2	6800	28933,4	2800 o1/p1	3000 o1/p2	4000 o2/p1	-	
6	1	2	6700	28649,5	2700 o1/p1	3000 o1/p2	4000 o2/p1	-	
7	1	2	6700	28365,6	2700 o1/p1	3000 o1/p2	3900 o2/p1	-	
8	1	2	6700	28081,7	2700 o1/p1	3000 o1/p2	3800 o2/p1	-	

A partir da linha 9, o método de busca dá início ao processo de exploração de combinações para acomodar as ordens aos compartimentos, mostrado pela tabela 15. Entre as linhas 9 e 11, o Compartimentalizador explora a possibilidade de preencher o compartimento cpi4 com produto de algum outro compartimento já designado, correspondente ao passo 3, gerando assim três possibilidades para designar cpi4, conforme as linhas de 9 a 11. Através do passo 4 é selecionada a melhor movimentação encontrada, buscando esta através do maior valor da função objetivo gerada entre as linhas 9 e 11. No caso, a melhor solução encontrada foi a da linha 9, com função objetivo igual a 7700, sendo esta a solução apresentada na linha 12.

Tabela 15 - Exemplo de Compartimentalização – linhas 9 a 12

Linha	Interação	Passo	Função Objetivo	Peso Total (Kg)	Cvmax e ordens por produto já alocadas nos compartimentos (cpi1, cpi2, cpi3, cpi4)				Descrição
					cpi1 3000	cpi2 2000	cpi3 2000	cpi4 3000	
9	1	3	7700	28081,7	2700 o1/p1	3000 o1/p2	-	3800 o2/p1	Movimento (intensificação): <u>explorar</u> movimentação dos compartimentos cheios para o vazio cpi4 .
10	1	3	7700	28081,7	2700 o1/p1	-	3800 o2/p1	3000 o1/p2	
11	1	3	6700	28081,7	-	3000 o1/p2	3800 o2/p1	2700 o1/p1	
12	1	4	7700	28081,7	2700 o1/p1	3000 o1/p2	-	3800* o2/p1	Atualizar melhor solução encontrada. Movimento da linha 9 escolhido, de cpi3 para cpi4 .

Entre as linhas 13 e 16 é feito o mesmo processo de exploração do passo 3; porém, agora para o compartimento vazio cpi3. Encontrando como resultado do passo 4 da linha 16 a linha 14 como melhor solução escolhida. Esta sequência de exploração se repete entre as linhas 17 e 20 e, depois, novamente de 21 a 24.

Porém, caso não haja uma melhoria da função objetivo por 3 iterações consecutivas, foi definido como parâmetro interno da busca que o sistema iniciará na próxima iteração outro tipo de movimento, no caso a quebra de volume. Desta forma, para a próxima iteração de número 5 (linhas de 25 a 31) o método de busca além de tentar a exploração do compartimento vazio cpi3 entre as linhas 25 a 27, o método também explora a quebra do volume do compartimento mais cheio cpi4 para o mais vazio cpi3. Entretanto esta quebra também se dá pouco a pouco através do parâmetro VCmark(cpi) igual a 100UGL. Ou seja, retirando 100UGL do compartimento cpi4 e adicionando 100UGL para o cpi3, conforme mostrado nas linhas de 28 a 30. Vale ressaltar novamente que, por segurança, o valor mínimo de preenchimento de um compartimento é igual a 85%, e por este motivo o compartimento cpi2 começa a ser preenchido com 1800UGL e não com 100UGL.

Dado este passo adicional de exploração, a solução encontrada com maior valor na função objetivo é o da linha 28, sendo esta selecionada pelo passo 4 na linha 31 como solução encontrada. O processo descrito entre as linhas 13 e 31 é mostrado na tabela 16.

Tabela 16 - Exemplo de Compartimentalização – linhas 13 a 31

Linha	Interação	Passo	Função Objetivo	Peso Total (Kg)	Cvmax e ordens por produto já alocadas nos compartimentos (cpi1, cpi2, cpi3, cpi4)				Descrição
					cpi1 3000	cpi2 2000	cpi3 2000	cpi4 3000	
					-	-	-	-	
13	2	3	7000	28081,7	-	3000	2700	3800	Movimento (intensificação): <u>explorar</u> movimentação dos compartimentos cheios para o vazio cpi3 .
					-	o1 / p2	o1 / p1	o2 / p1	
14	2	3	7700	28081,7	2700	-	3000	3800	
					o1 / p1	-	o1 / p2	o2 / p1	
15	2	3	6700	28081,7	2700	3000	3800	-	Atualizar melhor solução encontrada. Movimento da linha 14 escolhido, de cpi2 para cpi3 .
					o1 / p1	o1 / p2	o2 / p1	-	
16	2	4	7700	28081,7	2700	-	3000	3800	
					o1 / p1	-	o1 / p2	o2 / p1	
17	3	3	7000	28081,7	-	2700	3000	3800	Movimento (intensificação): <u>explorar</u> movimentação dos compartimentos cheios para o vazio cpi2 .
					-	o1 / p1	o1 / p2	o2 / p1	
18	3	3	7700	28081,7	2700	3000	-	3800	
					o1 / p1	o1 / p2	-	o2 / p1	
19	3	3	6700	28081,7	2700	3800	3000	-	Atualizar melhor solução encontrada. Movimento da linha 18 escolhido, de cpi3 para cpi2 .
					o1 / p1	o2 / p1	o1 / p2	-	
20	3	4	7700	28081,7	2700	3000	-	3800	
					o1 / p1	o1 / p1	-	o2 / p1	

21	4	3	7000	28081,7	-	3000	2700	3800	
					-	o1 / p2	o1 / p1	o2 / p1	Movimento (intensificação): <u>explorar</u>
22	4	3	7700	28081,7	2700	-	3000	3800	movimentação dos compartimentos
					o1 / p1	-	o1 / p2	o2 / p1	cheios para o vazio cp13 .
23	4	3	6700	28081,7	2700	3000	3800	-	
					o1 / p1	o1 / p2	o2 / p1	-	
24	4	4	7700	28081,7	2700	-	3000	3800	Atualizar melhor solução encontrada.
					o1 / p1	-	o1 / p2	o2 / p1	Movimento da linha 22 escolhido, de cp12
									para cp13 .
25	5	3	7000	28081,7	-	2700	3000	3800	Movimento (intensificação): <u>explorar</u>
					-	o1 / p1	o1 / p2	o2 / p1	movimentação dos compartimentos
26	5	3	7700	28081,7	2700	3000	-	3800	cheios para o vazio cp12 . (Obs: Porém,
					o1 / p1	o1 / p2	-	o2 / p1	todos estes movimentos já foram testados
27	5	3	6700	28081,7	2700	3800	3000	-	anteriormente)
					o1 / p1	o2 / p1	o1 / p2	-	
28	5	3	8400	28081,7	2700	1800	3000	2000	Movimento: dividir produto do
					o1 / p1	o2 / p1	o1 / p2	o2 / p1	compartimento mais cheio cp14 para o
29	5	3	8400	28081,7	2700	1900	3000	1900	mais vazio cp12 .
					o1 / p1	o2 / p1	o1 / p2	o2 / p1	VCmark(v, cp1) = 100 e VCminp(v, cp1) = 85%
30	5	3	8400	28081,7	2700	2000	3000	1800	
					o1 / p1	o2 / p1	o1 / p2	o2 / p1	
31	5	4	8400	28081,7	2700	1800	3000	2000	Atualizar melhor solução encontrada.
					o1 / p1	o2 / p1	o1 / p2	o2 / p1	Movimento da linha 28 escolhido, de cp14
									parcial para cp12 .

Continuando com o processo de exploração, a iteração 6 entre as linhas 32 e 35, conforme mostra a tabela 17, começa uma exploração do compartimento mais cheio cp12 para os demais compartimentos. Contudo, como não há mais compartimentos vazios, o processo de exploração é realizado com a permuta de cp12 com cp1 na linha 32, cp13 na linha 33 e cp14 na linha 34. Encontrando como resultado a linha 34 com o maior valor da função objetivo, selecionado pelo passo 4 na linha 35.

Por fim, tendo o sistema realizado uma alteração que também satisfaz as restrições de volumes máximos e mínimos por compartimento permitidos, a solução encontrada na linha 35 é dada como a final pela linha 36, finalizando a Busca Tabu e adicionando o resultado da linha 36 à lista de viagem compartimentalizadas factíveis.

Tabela 17 - Exemplo de Compartimentalização – linhas 32 a 36

Linha	Interação	Passo	Função Objetivo	Peso Total (Kg)	Cvmax e ordens por produto já alocadas nos compartimentos (cp1, cp2, cp3, cp4)				Descrição
					cp1 3000	cp2 2000	cp3 2000	cp4 3000	
					-	-	-	-	
32	6	3	8500	28081,7	2700	3000	1800*	2000	Movimento (intensificação): <u>explorar</u>
					o1 / p1	o1 / p2	o2 / p1	o2 / p1	movimentação do compartimento mais cheio
33	6	3	8800	28081,7	3000	2700	1500*	2000	cp12 para os demais compartimentos. (permuta)
					o1 / p2	o1 / p1	o2 / p1	o2 / p1	
34	6	3	9500	28081,7	3000	2000	1800*	2700	
					o1 / p2	o2 / p1	o2 / p1	o1 / p1	
35	6	4	9500	28081,7	3000	2000*	1800	2700	Atualizar melhor solução encontrada.
					o1 / p2	o2 / p1	o2 / p1	o1 / p1	Movimento da linha 34 escolhido, permuta entre
36	7	Fim	9500	28081,7	3000	2000	1800*	2700	Compartimentalização factível
					o1 / p2	o2 / p1	o2 / p1	o1 / p1	

Após um dado número de iterações (M), realiza-se o procedimento de parada, que gera uma nova solução inicial, retornando a busca ao passo 2. São realizadas duas paradas para cada combinação de pedidos, antes de reiniciar a busca para outra combinação de pedidos (a partir do passo 2). Caso o procedimento de busca já tenha testado todas as combinações de pedidos possíveis, a lista de soluções geradas é tomada como resultado, finalizando a busca.

4.1.2 Etapa 2 - Alocador de viagens

Após a geração da tabela com a lista de viagens compartmentalizadas factíveis, é iniciada a segunda etapa do processo. O Alocador de viagens, baseado em Busca Tabu, visa designar as viagens presentes na lista gerada na etapa 1 (Compartimentalizador), conforme ilustrado pela figura 22.

A segunda etapa da metodologia utilizada deve seguir as seguintes restrições do estudo de caso:

- i) Todos os veículos iniciam e terminam seu trajeto no mesmo terminal.
- ii) Serão consideradas duas possíveis formas de pagamento de frete FPT e FT, conforme apresentado na seção 3.5.1, aonde os custos incidentes em cada opção variam em função da quilometragem rodada, do número de carregamentos realizados e do custo de disponibilidade do caminhão;
- iii) As entregas dos produtos também devem obedecer às janelas horárias de recebimento dos clientes;
- iv) Existe uma disponibilidade mínima e máxima de produto nos terminais apresentando janelas de desabastecimento controladas por variáveis de incerteza;
- v) O tempo total da viagem, sendo carregamento, transporte e descarregamento, pode apresentar variações pela presença de uma variável de incerteza no tempo de transporte;
- vi) A disponibilidade de veículos também é regida por variáveis de incerteza;

- vii) As rotas possuem restrição de janelas de tempo de permissão de rodagem dos veículos.

A figura 22 descreve em fluxograma a forma que o Alocador de Viagens procura pela solução ótima. É importante ressaltar que ela é baseada na Busca Tabu, já apresentada pela figura 2, na seção 2.3 do presente trabalho. Onde para a Busca Tabu apresentada pela figura 22 foram acrescentados à figura 2 itens que customizam a Busca Tabu para alocar as viagens já compartimentalizadas.

Cesc é um contador de número de iterações para calcular o número de iterações executadas sem a melhoria do resultado da solução objetivo. Quando *Cesc* atinge um determinado número de iterações *Nesc*, o código regressa ao estado inicial de geração de uma solução inicial. Esta medida evita que a Busca Tabu fique presa em uma determinada região de solução e não analise outras possibilidades. Considerando que *N* é o número máximo de iterações que a Busca Tabu executará e *i* é a iteração executada no momento, são dados os seguintes passos:

Passo 1. Carregar dados do problema e contadores → Lista de viagens compartimentalizadas factíveis;

Passo 2. Gerar solução inicial ($i = 1$ e $Cesc = 0$) → Alocar o máximo número de viagens possíveis nos veículos disponíveis;

Passo 3. Intensificação - explorar a vizinhança da última solução gerada → Tentativa de adição de uma nova viagem à solução; Movimentação de uma viagem já alocada em um veículo para outro; e Exclusão de uma viagem já alocada;

Passo 4. Verificar solução encontrada em relação à função objetivo;

Passo 5. Contar variável *Cesc*, quando aplicável;

Passo 6. Contar variável $i = i + 1$;

Passo 7. Se $Nesc > Cesc$, voltar para o passo 3; senão ir para passo 8;

Passo 8. Se $N = i$: finalizar execução. Caso contrário, ir para o passo 9;

Passo 9. Diversificação da solução (voltar para o passo 2).

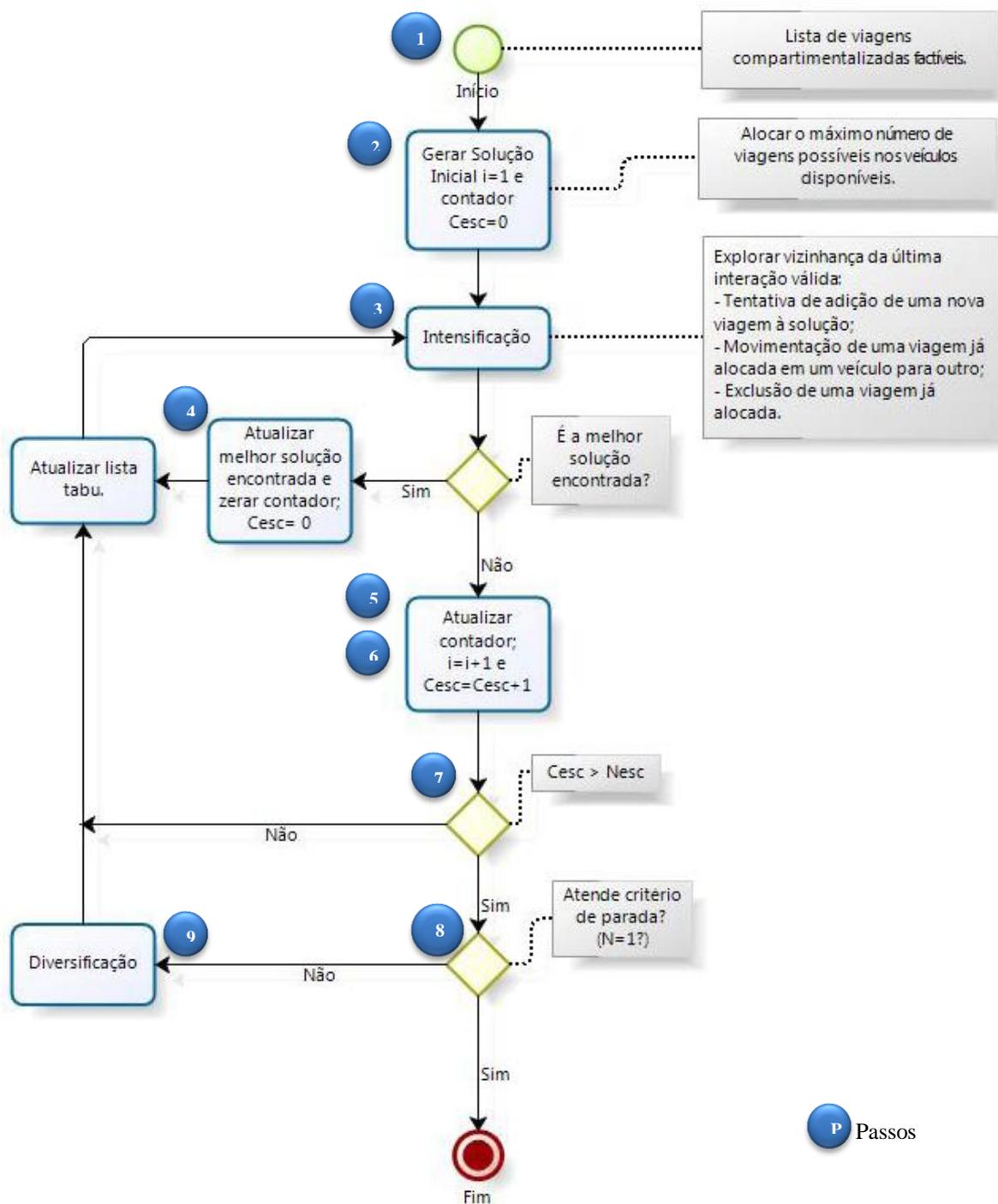


Figura 22 - Lógica da Busca Tabu do Alocador de viagens

• Exemplo do uso do Alocador

Com a finalidade de esclarecer o processo de alocação de viagens, apresentado no fluxograma da figura 22, é dado o exemplo descrito a seguir.

Nove ordens de entrega o_1 a o_9 , dos clientes l_1 a l_9 , respectivamente, devem ser entregues através do terminal t_1 e que podem conter dois tipos de produtos diferentes, p_1 e p_3 . O produto p_1 representa a gasolina e p_3 representa o diesel, valendo ressaltar que a densidade $TPde(t_1, p_1)$ da gasolina é menor do que a densidade $TPde(t_1, p_3)$ do diesel, impactando no resultado final da primeira etapa de compartimentalização, conforme mostra a tabela 18.

Para simplificar o exemplo proposto, a frota é homogênea, sendo os veículos de v_1 a v_3 com o mesmo número de horas disponíveis $TPAcod(d, p, t)$ (igual a 20 horas disponíveis), volume máximo de carregamento $VCmax$ e compartimentos $VCmax(v, cpi)$, conforme apresentado pela tabela 11.

Tabela 18 - Ordens de entrega o_1 a o_9

Ordem o	Cliente l	Volume $Pvol(o, p)$ (UGL ⁵)	Produtos	Descrição dos produtos
o_1	l_1	10000	p_1	p_1 Gasolina
o_2	l_2	10000	p_1	p_1 Gasolina
o_3	l_3	6000	p_1	p_1 Gasolina
o_3	l_3	4000	p_3	p_3 Diesel
o_4	l_4	10000	p_3	p_3 Diesel
o_5	l_5	10000	p_3	p_3 Diesel
o_6	l_6	4000	p_1	p_1 Gasolina
o_6	l_6	5000	p_3	p_3 Diesel
o_7	l_7	2000	p_3	p_3 Diesel
o_8	l_8	8000	p_1	p_1 Gasolina
o_9	l_9	5000	p_1	p_1 Gasolina

Como indicado no exemplo anterior da seção 4.1.1, na primeira etapa os dados das tabelas 11 e 18 são carregados e executados através de um procedimento de busca, conforme figuras 20 e 21. Finalizada a etapa de compartimentalização, uma lista de viagens compartimentalizadas factíveis é gerada para dar início à segunda etapa do processo: a alocação de viagens.

A tabela 19 apresenta as viagens compartimentalizadas factíveis, geradas através das tabelas 11 e 18 da primeira etapa. Nelas são encontradas 12 viagens que foram geradas a partir das 9 ordens, sendo importante notar que:

⁵UGL: representa galões

- i) O volume $Pvol(o,p)$ apresentado na tabela refere-se ao volume total do pedido o , já compartimentalizado e para ambos os produtos $p1$ e $p3$;
- ii) As viagens $vg6$, $vg7$, $vg8$, $vg10$, $vg11$ e $vg12$ são compartilhadas entre duas ordens de entrega;
- iii) Para a viagem $vg6$, o pedido $o1$ foi significativamente alterado para a acomodação do pedido $o7$ ao veículo. Este efeito também ocorre nas viagens $vg7$ e $vg3$. Caso essas viagens sejam utilizadas pelo Alocador, estas são contabilizadas pelo parâmetro $Lmaxn(l)$ ⁶ mencionado na seção 3.2. Vale ressaltar que estas alterações estão fora dos limites de $Lmaxo(l,o)$ ⁷ e $Lmino(l,o)$ ⁸ e, por este motivo, podem ser contabilizadas por $Lmaxn(l)$;
- iv) Ordens de entregas, como $o1$, $o2$, $o3$, $o4$ e $o5$, foram solicitadas com a mesma quantidade de produto, 10 KUGL, porém compartimentalizadas com volumes iguais ou inferiores ao original, devido à densidade do produto;
- v) No momento da alocação de uma viagem, esta pode inviabilizar a utilização de outra viagem. Como exemplo, a alocação da $vg6$ exclui a possibilidade de utilização das viagens $vg7$, $vg8$, $vg11$ e $vg12$ por possuírem o pedido de entrega $o7$ em comum.

A tabela 20, indica passo a passo a forma na qual o Alocador de viagens encontra a solução utilizando Busca Tabu. A função objetivo utilizada na Busca Tabu do Alocador de viagens é minimizar o custo total, devendo seguir a principal restrição de programar todas as ordens de pedidos solicitadas. Tendo em vista que o objetivo do exemplo é apresentar a forma na qual as viagens são alocadas na busca da solução ótima, o resultado da função objetivo apresentado na tabela 20 é apenas o cálculo de horas utilizadas pelos veículos.

⁶ $Lmaxn(l)$: Número máximo de vezes que um cliente l pode aceitar alteração de seus pedidos em um período de um mês. (Alterações podem ser de volume ou data de entrega)

⁷ $Lmaxo(l,o)$: Porcentagem de variação para mais do volume total de um pedido o de um cliente l .

⁸ $Lmino(l,o)$: Porcentagem de variação para menos do volume total de um pedido o de um cliente l .

Tabela 19 - Lista de viagens compartimentalizadas factíveis

Viagem vg	Ordem o	Volume Pvol(o,p) (kUGL)	Tempo de duração da viagem (h.)
vg_1	O_1	10000	5
vg_2	O_2	10000	5
vg_3	O_3	9800	8
vg_4	O_4	9200	6
vg_5	O_5	9200	6
vg_6	O_1	7000	9
vg_6	O_7	1900	
vg_7	O_2	7000	12
vg_7	O_7	1900	
vg_8	O_3	6900	9
vg_8	O_7	1700	
vg_9	O_6	9000	6
vg_{10}	O_9	4000	7
vg_{10}	O_8	6000	
vg_{11}	O_7	2000	10
vg_{11}	O_8	7700	
vg_{12}	O_7	2000	14
vg_{12}	O_9	5400	

Neste exemplo, os valores de Nesc e do número de iterações Tabu igual a 2, os valores que apresentam asterisco na tabela 21 simbolizam ser “Tabu” e não podem ser movimentados por 2 iterações.

A tabela 20 deve ser lida de cima para baixo e acompanhando os comentários explicativos situados à direita. A primeira coluna é identificada por linha (apenas como referência), enquanto a segunda refere-se ao número da iteração i que está sendo executada. A terceira coluna é referente ao passo do processo de busca, que foram apresentados na descrição do fluxograma da figura 22. Para facilitar a leitura da tabela 20, ela é descrita e dividida entre as tabelas de 21 a 23.

Tabela 20 - Exemplo de alocação

Linha	Interação	Passo	Lista de pedidos ainda não alocados	Lista de viagens factíveis não alocadas	Função Objetivo	Horas disponíveis e viagens já alocadas dos veículos (v1, v2, v3)			Descrição
						v1 20hrs	v2 20hrs	v3 20hrs	
1	1	1	o1, o2, o3, o4, o5, o6, o7, o8, o9	vg1, vg2, vg3, vg4, vg5, vg6, vg7, vg8, vg9, vg10, vg11, vg12	0	20	20	20	Carregar lista de viagens compartimentalizadas factíveis
2	1	2	o4, o5, o6, o7, o8, o9	vg4, vg5, vg9, vg10, vg11, vg12	18	15	15	12	Geração da solução inicial sequencial (que não é aleatória). Alocar o máximo número de viagens possíveis nos veículos disponíveis
3	1	2	o7, o8, o9	vg10, vg11, vg12	36	9	9	6	
4	1	2	o7		43	2	9	6	
5	1	3	o7		43	2	9	6	
6	1	3	o7		43	9	2	6	Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg10 (escolha aleatória da viagem).
7	1	3	o7		42	9	9	-1	
8	1	4	o7		43	9	2	6	Atualizar melhor solução encontrada. Movimento escolhido linha 6 , de vg10 em v2 . (Cesc = Cesc +1) Cesc = 1
9	2	3	o7		43	3	2	12	Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg9 (escolha aleatória da viagem).
10	2	3	o7		43	9	-4	12	
11	2	3	o7		43	9	2	6	
12	2	4	o7		43	3	2	12	Atualizar melhor solução encontrada. Movimento escolhido linha 9 , de vg9 em v1 . (Cesc = Cesc +1) Cesc = 2
13	3	3	o7		43	3	2	12	Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg1 (escolha aleatória da viagem).
14	3	3	o7		43	8	-3	12	
15	3	3	o7		43	8	2	7	
16	3	4	o7		43	8	2	7	Atualizar melhor solução encontrada. Movimento escolhido linha 15 , de vg1 em v3 . (Cesc = Cesc +1) Cesc = 3
17	3	8	o2, o7	vg2*, vg7	38	8	7	7	[Cesc > Nesc] = Sim, Então: Movimento exclusão aleatória (Retirar uma viagem aleatória). Viagem escolhida: vg2

18	4	3	o2, o7	vg2*, vg7	38	8 vg4, vg9*	7 vg5, vg10	7 vg1*, vg3	Movimento (intensificação): <u>Alocar</u> viagem aleatória da lista de viagens disponíveis, com exceção de vg2 (Tabu).
19	4	3	o2, o7	vg2*, vg7	38	8 vg4, vg9*	7 vg5, vg10	7 vg1*, vg3	Atualizar melhor solução encontrada: Porém como a única viagem que permite alocação é vg7 e não tempo disponível (12hrs) então: Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg4 (escolha aleatória da viagem).
20	4	3	o2, o7	vg2*, vg7	38	14 vg9*	1 vg4, vg5, vg10	7 vg1*, vg3	
21	4	3	o2, o7	vg2*, vg7	38	14 vg9*	7 vg5, vg10	1 vg4, vg1*, vg3	
22	4	4	o2, o7	vg2*, vg7	38	14 vg9	1 vg4* , vg5, vg10	7 vg1*, vg3	Atualizar melhor solução encontrada. Movimento escolhido linha 20 , de vg4 em v2 . (Cesc = Cesc +1) Cesc = 1
23	5	3			50	2 vg9, vg7*	1 vg4*, vg5, vg10	7 vg1*, vg3	Movimento (intensificação): <u>Alocar</u> viagem aleatória da lista de viagens disponíveis, com exceção de vg2 (Tabu). Escolhida: vg7
24	6	3			50	2 vg9, vg7*	1 vg4*, vg5, vg10	7 vg1*, vg3	Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg9 (escolha aleatória da viagem).
25	6	3			50	8 vg7*	-5 vg4*, vg5, vg9, vg10	7 vg1*, vg3	
26	6	3			50	8 vg7*	1 vg4*, vg5, vg10	1 vg1*, vg3, vg9	
27	6	4			50	8 vg7*	1 vg4*, vg5, vg10	1 vg1, vg3, vg9*	
28	7	3			50	8 vg7*	1 vg4*, vg5, vg10	1 vg1, vg3, vg9*	Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg3 (escolha aleatória da viagem).
29	7	3			50	8 vg7*	-7 vg3, vg4*, vg5, vg10	9 vg1, vg9*	
30	7	3			50	0 vg3, vg7*	1 vg4*, vg5, vg10	9 vg1, vg9*	
31	7	4			50	0 vg3*, vg7*	1 vg4, vg5, vg10	9 vg1, vg9*	Atualizar melhor solução encontrada. Movimento escolhido linha 26 , de vg9 em v3 . (Cesc = Cesc +1) Cesc = 3
32	8	8	o4	vg4*	44	0 vg3*, vg7*	7 vg5 , vg10	9 vg1, vg9*	[Cesc > Nesc] = Sim , Então: Movimento exclusivo aleatória (Retirar uma viagem aleatória). Viagem escolhida: vg4
33	Fim				50	0 vg3, vg7	1 vg4, vg5, vg10	9 vg1, vg9	Alocação de viagens é factível (satisfaz a entrega de todos os pedidos) Última melhor solução encontrada: Linha 32

A quarta coluna indica a lista de pedidos ainda não alocados, sendo inicialmente proveniente da tabela 19. A quinta é referente à lista de viagens factíveis não alocadas, representada inicialmente pela tabela 20. A sexta coluna apresenta apenas o resultado da função objetivo. E, finalmente, as colunas de v1 a v3 indicam os veículos (nos quais as viagens devem ser alocadas) com suas respectivas informações de horas disponíveis e a lista de viagens programadas para cada veículo.

É importante mencionar que, ao movimentar uma viagem da lista de viagens factíveis não alocadas para algum veículo, esta viagem deve

desaparecer da lista na iteração seguinte. Adicionalmente, os pedidos referentes à viagem alocada devem desaparecer da lista de pedidos ainda não alocados, assim como as viagens atreladas a estes pedidos da lista de viagens factíveis não alocadas. Ou seja, caso a viagem *vg1* seja utilizada, as viagens *vg1* e *vg7* devem desaparecer da lista de viagens factíveis não alocadas e o pedido *o1* desaparecerá da lista de pedidos ainda não alocados.

Iniciando o processo pelo passo 1, na linha 1, os dados do problema são carregados, conforme mostra tabela 21. No passo 2, linha de 2 a 4, é gerada sequencialmente uma solução inicial. Neste passo é alocada uma viagem por veículo na sequência onde, viagens e veículos, foram carregados. É importante ressaltar o evento da linha 3, no qual (após alocar as viagens *vg1* a *vg5*) a viagem *vg9* foi alocada em *v3*. Este fato se deve ao efeito mencionado anteriormente de que uma viagem pode excluir outra(s) da lista de viagens factíveis não alocadas. No caso, as viagens *vg6*, *vg7* e *vg8* foram eliminadas desta lista.

Tabela 21 - Exemplo de alocação – linhas 1 a 4

Linha	Interação	Passo	Lista de pedidos ainda não alocados	Lista de viagens factíveis não alocadas	Função Objetivo	Horas disponíveis e viagens já alocadas dos veículos (v1, v2, v3)			Descrição
						v1 20hrs	v2 20hrs	v3 20hrs	
1	1	1	o1, o2, o3, o4, o5, o6, o7, o8, o9	<i>vg1, vg2, vg3, vg4, vg5, vg6, vg7, vg8, vg9, vg10, vg11, vg12</i>	0	20	20	20	Carregar lista de viagens compartimentalizadas factíveis
2	1	2	o4, o5, o6, o7, o8, o9	<i>vg4, vg5, vg9, vg10, vg11, vg12</i>	18	15	15	12	Geração da solução inicial sequencial (que não é aleatória). Alocar o máximo número de viagens possíveis nos veículos disponíveis
3	1	2	o7, o8, o9	<i>vg10, vg11, vg12</i>	36	9	9	6	
4	1	2	o7		43	2	9	6	
						<i>vg1, vg4, vg10</i>	<i>vg2, vg5</i>	<i>vg3, vg9</i>	

Na linha 4 da tabela 22 é possível observar que a solução inicial não conseguiu ainda alocar todos os pedidos dos clientes, devido à ausência da ordem *o7*. Porém, não há viagem disponível entre as iterações de 1 a 3 para fazer com que ela seja utilizada. Neste caso, a Busca Tabu começa a mover viagens de um veículo para outro (desde que Cesc não atinja um valor maior do que 2 e cause uma exclusão aleatória de alguma viagem já alocada e que não seja Tabu).

Tabela 22 - Exemplo de alocação – linhas 5 a 16

Linha	Interação	Passo	Lista de pedidos ainda não alocados	Lista de viagens factíveis não alocadas	Função Objetivo	Horas disponíveis e viagens já alocadas dos veículos (v1, v2, v3)			Descrição
						v1 20hrs	v2 20hrs	v3 20hrs	
5	1	3	o7		43	2 vg1, vg4, vg10	9 vg2, vg5	6 vg3, vg9	Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg10 (escolha aleatória da viagem).
6	1	3	o7		43	9 vg1, vg4	2 vg2, vg5, vg10	6 vg3, vg9	
7	1	3	o7		42	9 vg1, vg4	9 vg2, vg5	-1 vg3, vg9, vg10	
8	1	4	o7		43	9 vg1, vg4	2 vg2, vg5, vg10*	6 vg3, vg9	Atualizar melhor solução encontrada. Movimento escolhido linha 6 , de vg10 em v2 . (Cesc = Cesc +1) Cesc = 1
9	2	3	o7		43	3 vg1, vg4, vg9	2 vg2, vg5, vg10*	12 vg3	Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg9 (escolha aleatória da viagem).
10	2	3	o7		43	9 vg1, vg4	-4 vg2, vg5, vg10*, vg9	12 vg3	
11	2	3	o7		43	9 vg1, vg4	2 vg2, vg5, vg10*	6 vg3, vg9	
12	2	4	o7		43	3 vg1 , vg4, vg9*	2 vg2, vg5, vg10*	12 vg3	Atualizar melhor solução encontrada. Movimento escolhido linha 9 , de vg9 em v1 . (Cesc = Cesc +1) Cesc = 2
13	3	3	o7		43	3 vg1, vg4, vg9*	2 vg2, vg5, vg10*	12 vg3	Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg1 (escolha aleatória da viagem).
14	3	3	o7		43	8 vg4, vg9*	-3 vg1, vg2, vg5, vg10*	12 vg3	
15	3	3	o7		43	8 vg4, vg9*	2 vg2, vg5, vg10*	7 vg1, vg3	
16	3	4	o7		43	8 vg4, vg9*	2 vg2, vg5, vg10*	7 vg1* , vg3	Atualizar melhor solução encontrada. Movimento escolhido linha 15 , de vg1 em v3 . (Cesc = Cesc +1) Cesc = 3

Na linha 17, no final da interação 3, a viagem vg2 é excluída aleatoriamente liberando as viagens vg2 e vg7 para serem utilizadas, porém, vg2 ainda permanece Tabu por mais duas iterações. Entretanto, na interação 4 a Busca Tabu não consegue alocar a viagem vg7 agora disponível, por não haver nenhum veículo com 12 horas disponíveis para alocar esta viagem. Assim, a Busca Tabu movimenta a viagem vg4 do veículo v1 para v2. Contudo, este último movimento executado pela interação 4 abre uma maior janela horária para o veículo v1, permitindo assim a alocação da viagem vg7 ao veículo v1, conforme mostra a linha 23 (iteração 5) na tabela 23.

Tabela 23 - Exemplo de alocação – linhas 17 a 23

Linha	Interação	Passo	Lista de pedidos ainda não alocados	Lista de viagens factíveis não alocadas	Função Objetivo	Horas disponíveis e viagens já alocadas dos veículos (v1, v2, v3)			Descrição
						v1 20hrs	v2 20hrs	v3 20hrs	
17	3	8	o2, o7	vg2*, vg7	38	8 vg4, vg9*	7 vg5, vg10	7 vg1*, vg3	[Cesc > Nesc] = Sim, Então: Movimento exclusão aleatória (Retirar uma viagem aleatória). Viagem escolhida: vg2
18	4	3	o2, o7	vg2*, vg7	38	8 vg4, vg9*	7 vg5, vg10	7 vg1*, vg3	Movimento (intensificação): Alocar viagem aleatória da lista de viagens disponíveis, com exceção de vg2 (Tabu).
19	4	3	o2, o7	vg2*, vg7	38	8 vg4, vg9*	7 vg5, vg10	7 vg1*, vg3	Atualizar melhor solução encontrada: Porém como a única viagem que permite alocação é vg7 e não tempo disponível (12hrs) então:
20	4	3	o2, o7	vg2*, vg7	38	14 vg9*	1 vg4, vg5, vg10	7 vg1*, vg3	Movimento (intensificação): Explorar movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg4 (escolha aleatória da viagem).
21	4	3	o2, o7	vg2*, vg7	38	14 vg9*	7 vg5, vg10	1 vg4, vg1*, vg3	Atualizar melhor solução encontrada. Movimento escolhido linha 20, de vg4 em v2 . (Cesc = Cesc +1) Cesc = 1
22	4	4	o2, o7	vg2*, vg7	38	14 vg9	1 vg4*, vg5, vg10	7 vg1*, vg3	Movimento (intensificação): Alocar viagem aleatória da lista de viagens disponíveis, com exceção de vg2 (Tabu). Escolhida: vg7
23	5	3			50	2 vg9, vg7*	1 vg4*, vg5, vg10	7 vg1*, vg3	

Para as iterações 6 e 7, linhas 24 a 31, a Busca Tabu acaba apenas fazendo movimentações de viagens entre veículos sem a melhoria da função objetivo. Consecutivamente, estas movimentações acabam por aumentar o valor de Cesc, acarretando em mais uma exclusão de viagem aleatória e diminuindo o valor da função objetivo, conforme mostra linha 32 na tabela 24.

No exemplo apresentado a última melhor solução encontrada foi da linha 32, iteração 7, sendo esta utilizada como resultado da Busca Tabu e do exemplo. Entretanto, como a metodologia heurística não é determinística, caso a Busca Tabu continue a ser executada por mais iterações, ela ainda pode encontrar outras soluções melhores. Isso é possível graças ao movimento de diversificação, que gera uma nova solução inicial caso a melhor solução encontrada não melhore após um número limite de iterações.

Tabela 24 - Exemplo de alocação – linhas 24 a 33

Linha	Interação	Passo	Lista de pedidos ainda não alocados	Lista de viagens factíveis não alocadas	Função Objetivo	Horas disponíveis e viagens já alocadas dos veículos (v1, v2, v3)			Descrição
						v1 20hrs	v2 20hrs	v3 20hrs	
24	6	3			50	2 vg9, vg7*	1 vg4*, vg5, vg10	7 vg1*, vg3	Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg9 (escolha aleatória da viagem).
25	6	3			50	8 vg7*	-5 vg4*, vg5, vg9, vg10	7 vg1*, vg3	
26	6	3			50	8 vg7*	1 vg4*, vg5, vg10	1 vg1*, vg3, vg9	
27	6	4			50	8 vg7*	1 vg4*, vg5, vg10	1 vg1, vg3, vg9*	Atualizar melhor solução encontrada. Movimento escolhido linha 26 , de vg9 em v3 . (Cesc = Cesc +1) Cesc = 2
28	7	3			50	8 vg7*	1 vg4*, vg5, vg10	1 vg1, vg3, vg9*	Movimento (intensificação): <u>Explorar</u> movimentação de uma viagem já alocada de um veículo para outro. Viagem escolhida vg3 (escolha aleatória da viagem).
29	7	3			50	8 vg7*	-7 vg3, vg4*, vg5, vg10	9 vg1, vg9*	
30	7	3			50	0 vg3, vg7*	1 vg4*, vg5, vg10	9 vg1, vg9*	
31	7	4			50	0 vg3* , vg7*	1 vg4, vg5, vg10	9 vg1, vg9*	Atualizar melhor solução encontrada. Movimento escolhido linha 26 , de vg9 em v3 . (Cesc = Cesc +1) Cesc = 3
32	8	8	o4	vg4*	44	0 vg3* , vg7*	7 vg5, vg10	9 vg1, vg9*	[Cesc > Nesc] = Sim , Então: Movimento exclusão aleatória (Retirar uma viagem aleatória). Viagem escolhida: vg4
33		Fim			50	0 vg3, vg7	1 vg4, vg5, vg10	9 vg1, vg9	Alocação de viagens é factível (satisfaz a entrega de todos os pedidos) Última melhor solução encontrada: Linha 32

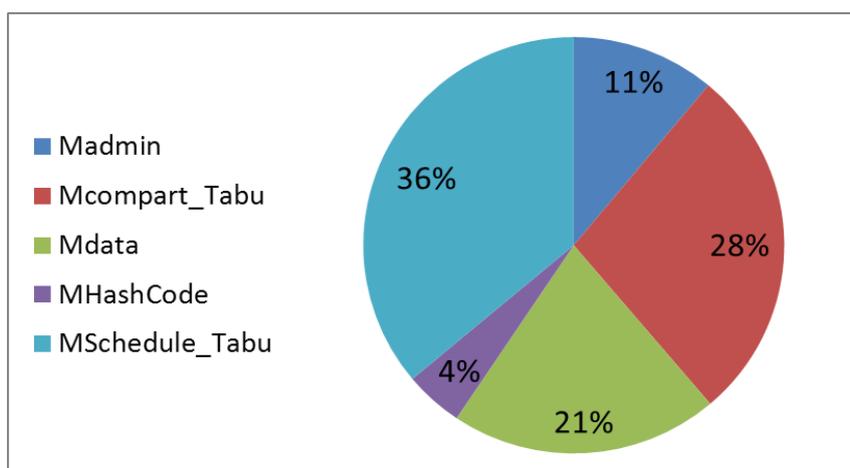
4.2 Arquitetura da implementação

A linguagem de programação escolhida para o desenvolvimento da implementação foi Visual Basic (VB), devido à experiência do autor com a linguagem. O software Microsoft Access 2010 é utilizado como base para a criação do programa, já que este possui todas as funcionalidades de uma plataforma de desenvolvimento em VB e SQL.

O programa foi dividido em seis módulos de programação, totalizando 2463 linhas de código. Aproximadamente igual a 82 páginas de código, considerando 30 linhas de código por página, conforme apresentado pela tabela 25 e figura 23.

Tabela 25 - Módulos de programação

Módulo	Nº de linhas	Nº de Páginas	Função
Madmin	272	9	Gerador de viagens
Mcompart_Tabu	683	23	Compartimentalizador
Mdata	508	17	Restrições do alocador de viagens
MHashCode	111	4	CODEC do código de Hash
MSchedule_Tabu	889	30	Alocador Busca Tabu
Total:	2463	82	

**Figura 23 -** Distribuição percentual dos módulos

Na primeira parte de cálculo do programa, o gerador de viagens compartimentalizadas factíveis, pertence aos módulos Madmin e Mcompart_Tabu e equivale a 29% do programa. Os demais módulos, Mdata, MhashCode e MSchedule_Tabu fazem parte da segunda etapa do programa, o Alocador de viagens, totalizando 71% do programa.

O módulo Madmin “controla” o Compartimentalizador, Mcompart_Tabu. O módulo MSchedule_Tabu de Busca Tabu administra o cálculo de alocação de viagens. Ele utiliza os módulos Mdata, com as restrições, e MhashCode para codificar e decodificar as informações horárias do programa em código de Hash.

Adicionalmente as rotinas principais de Mcompart_Tabu e MSchedule_Tabu referentes às etapas 1 e 2, podem ser visualizados em linguagem de programação nos apêndices D e E.

4.3 Considerações técnicas

Para a implementação da solução, algumas considerações técnicas devem ser consideradas, tais como: tempo de processamento, origem dos dados, recursos utilizados e metodologias utilizadas na programação. Todos os testes foram realizados em um Computador Intel Core i5, CPU M560 2.6GHz / 2.6GHz, Memória RAM 4GB, sistema operacional Windows 7 32-bit.

Com relação ao tempo de processamento é importante ressaltar que ele não faz parte dos objetivos específicos, porém este é exposto na seção 5.3.3 de resultados. A justificativa é o fato de ele ser utilizado como um indicador relevante para futuros desenvolvimentos de trabalhos na área. Ou seja, ele permite um aprofundamento do entendimento da metodologia utilizada e pode apontar regiões de otimização de gargalos no processamento ou na forma com que o código foi escrito.

A respeito dos dados utilizados no programa, estes são provenientes de duas diferentes fontes. A primeira do sistema ERP da empresa, como ordens de pedido e informações de cadastro dos veículos e clientes. A segunda proveniente do campo, no caso, de dados operacionais, como tempo de carregamento, horários críticos de tráfego e indisponibilidade de veículo. No entanto, é importante afirmar que os dados utilizados são apenas estimados a partir dos dados reais da empresa e não iguais, para assim manter a integridade e sigilo das informações da companhia.

Por último, com relação às metodologias de programação utilizadas, é necessário discorrer um pouco sobre a etapa de desenvolvimento do programa. Nesta etapa foram encontrados vários desafios para a conclusão do desenvolvimento do código do programa. Dentre as medidas utilizadas, foram identificadas duas abordagens chave para a viabilização da implementação do código. A primeira corresponde ao passo 1 de ambas as Busca Tabu, referente ao carregamento dos dados, citados nas seções 4.1.1 e 4.1.2, onde a implementação carrega inicialmente todos os dados do problema em variáveis compostas, também chamadas de vetoriais, ao invés de fazer com que o código manipule diretamente os dados das tabelas. E, somente ao final do cálculo o programa descarrega novamente os resultados nas tabelas. Para maiores informações sobre variáveis compostas, consultar o apêndice B.

O segundo desafio foi quanto à forma na qual o programa trata as janelas horárias, sendo utilizado um código Hash para tal finalidade, como indicado no apêndice A. A decisão foi tomada ao identificar que o uso de código Hash “perfeito mínimo” apresenta operações matemáticas básicas como adição e subtração para um grande volume de informação, que facilitariam a resolução do problema.

5 Resultados experimentais

Tendo em vista que o estudo de caso proposto é um problema complexo, conforme visto na seção 3, o objetivo do trabalho a comparação entre diferentes abordagens com e sem a presença de incertezas. Os dados do problema serão apresentados na seção 5.1, que são depois separados em cenários na seção 5.2 e são apresentadas suas respectivas análises e resultados na seção 5.3.

5.1 Dados do problema

A janela de dados utilizada para testes foi entre os dias 11 a 24 de dezembro de 2011, equivalentes a 14 dias ininterruptos de operação, apresentando grandes variações na demanda diária, conforme ilustra a demanda em volume na figura 24.

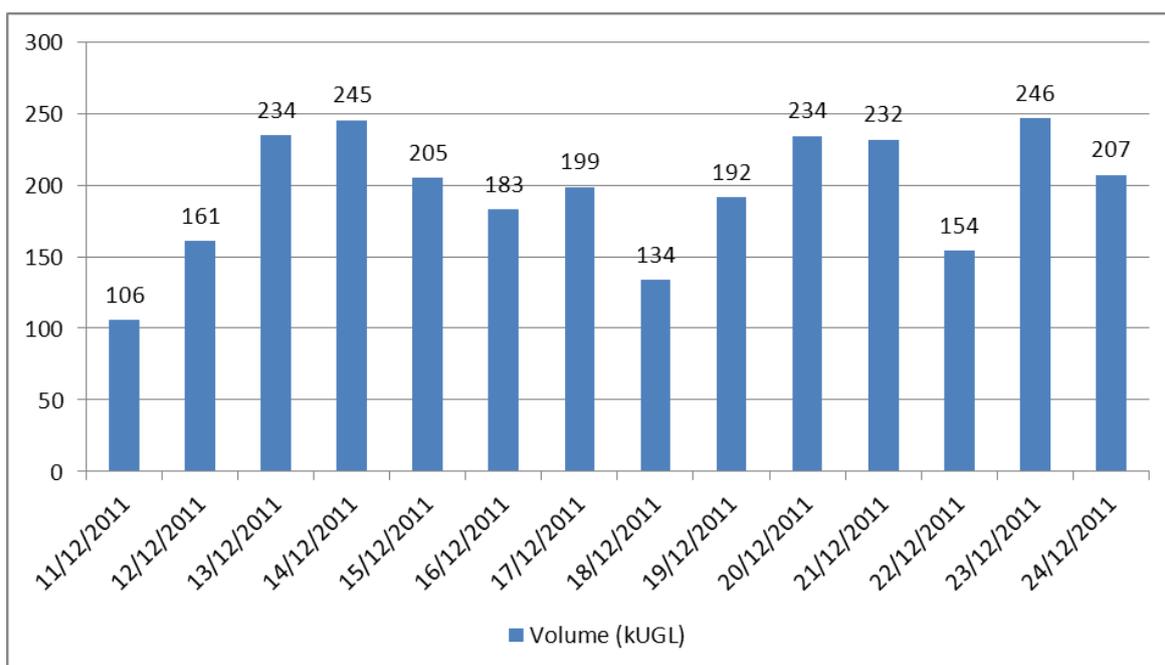


Figura 24 - Distribuição de demanda do estudo de caso

Na tabela 26 são encontradas informações da demanda para cada dia de operação, assim como o custo e número de entregas provenientes da operação executada.

Tabela 26 - Dados gerais do problema por dia de operação

Data	Volume (kUGL)	Custo (US\$)	Número de Pedidos	Número de entregas
11/12/2011	106	2,79	14	11
12/12/2011	161	4,62	22	21
13/12/2011	234	7,46	33	31
14/12/2011	245	7,48	33	31
15/12/2011	205	6,46	27	26
16/12/2011	183	5,45	24	24
17/12/2011	199	6,29	28	26
18/12/2011	134	3,65	16	15
19/12/2011	192	5,99	27	25
20/12/2011	234	7,23	34	30
21/12/2011	232	6,85	30	29
22/12/2011	154	4,94	22	21
23/12/2011	246	8,12	37	32
24/12/2011	207	5,94	28	26
Total:	2732	83,26	375	348

É importante mencionar que todos os pedidos devem ser atendidos na execução da operação exibida na tabela 26. Porém, o número de entregas é menor ou igual ao número de pedidos, já que o problema proposto aceita entregas parciais, com mais de um pedido por entrega.

Com relação ao custo, como o período de dias analisados é aproximadamente igual à metade do número de dias do mês, os veículos que possuem tarifa mensal no custo tiveram seu valor dividido por dois para compensar a outra metade do mês.

Como a quantidade de dados de entrada utilizada para o cálculo dos 14 dias analisados é muito grande, será tomado o dia 15 de dezembro como exemplo. Este caso é apresentado com detalhes na seção 5.1.1, para exemplificação do estudo de caso e, posteriormente, utilização na seção 5.3 para entendimento de alguns resultados obtidos.

5.1.1 Caso do dia 15 de dezembro de 2011

O dia 15 de dezembro de 2011 foi escolhido como exemplo por apresentar uma demanda aproximadamente média em relação aos outros dias de operação. O cenário possui um total de 19 clientes, no qual 18 podem receber tanto veículos

com ou sem bomba, conforme mostra a tabela 27. Dos 19 clientes apenas 1 deve ter entrega pelo terminal T02 e os demais pelo terminal T01. Os dados de terminais são apresentados pela tabela 28.

Dado o grupo de clientes, são solicitadas as ordens de entrega expostas pela tabela 29. Estas ordens podem ser entregues tanto como carga parcial ou completa de um veículo, conforme lista de veículos na tabela 30.

A tabela 30 de veículos é complementada pelas tabelas de custo, disponibilidade e compartimentos, identificados por códigos nos campos “ID de custo”, “ID de disponibilidade” e “ID de compartimentos”, respectivamente. A tabela 31 apresenta os códigos para o custo de pagamento de frete com seus valores, enquanto a tabela 34 apresenta os códigos de disponibilidade e a tabela 32 apresenta os tipos de compartimentos que os veículos possuem.

Para o cálculo de tempo de viagem é apresentado na tabela 33 o conjunto das distâncias dos trechos que o sistema pode utilizar para calcular a distância total da viagem. Na tabela 35 encontra-se a especificação do tipo de zona de tráfego que o cliente pertence.

Demais especificações dos terminais são dadas pelas tabelas 36 e 37, respectivamente referentes às especificações de produtos dos terminais e suas disponibilidades.

Para todas as tabelas apresentadas na presente seção, os parâmetros referentes à presença de incerteza estão totalmente carregados; ou seja, como em um cenário no qual todas as incertezas, citadas no capítulo 3, estariam sendo aplicadas ao problema a ser resolvido.

Tabela 27 - Clientes

Código do Cliente	Janela Horária do cliente (hh:mm)	Zone Id	Tempo fixo de carga (hh:mm)	Necessidade de bomba	Terminal de carregamento	Tamanho máximo de caminhão (m)	Peso máximo do caminhão (kg)	% de alteração para mais	% de alteração para menos	Máximo de pedidos alteradas por volume	Máximo de pedidos alteradas por data
C01	00h00 - 23h00	TRAF00	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C02	00h00 - 23h00	TRAF00	00:30	Sim	T01	10	30000	10,0%	10,0%	1	1
C03	00h00 - 23h00	TRAF00	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C04	00h00 - 23h00	TRAF01	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C05	00h00 - 23h00	TRAF01	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C06	08h00 - 19h00	TRAF01	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C07	00h00 - 23h00	TRAF01	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C08	00h00 - 23h00	TRAF01	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C09	00h00 - 23h00	TRAF00	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C10	05h00 - 19h00	TRAF00	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C11	00h00 - 23h00	TRAF01	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C12	00h00 - 23h00	TRAF00	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C13	00h00 - 23h00	TRAF00	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C14	00h00 - 23h00	TRAF00	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C15	05h00 - 19h00	TRAF00	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C16	08h00 - 19h00	TRAF01	00:30	Não	T01	10	30000	10,0%	10,0%	1	1
C17	00h00 - 23h00	TRAF02	00:30	Não	T02	10	30000	10,0%	10,0%	1	1
C18	00h00 - 23h00	TRAF00	00:20	Não	T01	10	30000	10,0%	10,0%	1	1
C19	08h00 - 19h00	TRAF01	00:30	Não	T01	10	30000	10,0%	10,0%	1	1

Tabela 30 - Veículos disponíveis

Veículo	Número de Veículos	ID de disponibilidade do veículo	ID do custo do veículo	ID dos compartimentos do veículo	Caminhão com Bomba	Volume Máximo (UGL)	Peso do caminhão (kg)	Peso máximo a ser carregado (Kg)	Vazão de descarga (UGL/min)	Veículo BL	Probabilidade de indisponibilidade	Velocidade máxima quando vazio (km/h)
V02	2	AVA	COSTV02	COMPV02	Não	4,4	23760	13530	0,45	Não	0,02	70
V03	1	AVA	COSTV03	COMPV03	Não	10	42800	30320	0,45	Não	0,02	70
V04	2	AVA	COSTV04	COMPV04	Não	7,1	38780	25980	0,45	Sim	0,02	70
V07	1	AVA	COSTV07	COMPV07	Sim	9,5	49400	36920	0,45	Não	0,02	70
V08	1	AVA	COSTV08	COMPV08	Não	9	38780	26260	0,45	Não	0,02	70
V01	1	AVR	COSTV01	COMPV01	Não	10	42940	29470	0,45	Sim	0,02	70
V05	4	AVR	COSTV05	COMPV05	Não	10	42940	28700	0,45	Sim	0,02	70
V06	1	AVR	COSTV06	COMPV06	Não	7	38780	25510	0,45	Não	0,02	70
V09	1	AVR	COSTV09	COMPV09	Não	4	23100	12680	0,45	Não	0,02	70
V10	1	AVR	COSTV10	COMPV10	Não	10	42940	29440	0,45	Não	0,02	70

Tabela 31 - Custo dos veículos disponíveis

ID do custo do veículo	Custo Mensal (\$)	Custo por Km (\$)	Custo por viagem (\$)
COSTV01	0,0	0,37	77,8
COSTV02	1758,9	0,30	68,4
COSTV03	2345,8	0,44	85,7
COSTV04	2341,1	0,39	82,9
COSTV05	2241,0	0,37	77,8
COSTV06	0,0	0,56	96,5
COSTV07	0,0	0,57	109,2
COSTV09	0,0	0,40	74,7
COSTV10	0,0	0,56	96,5

Tabela 32 - Dados dos compartimentos dos veículos

ID dos compartimentos do veículo	Número do Compartimento	% mínima de carregamento	Volume (UGL)	Multi-Marca (UGL)
COMPV01	1	85%	3	0,1
COMPV01	2	85%	2	0,1
COMPV01	3	85%	2	0,1
COMPV01	4	85%	3	0,1
COMPV02	1	85%	1,2	0,1
COMPV02	2	85%	0	0,1
COMPV02	3	85%	2,2	0,1
COMPV02	4	85%	1	0,1
COMPV03	1	85%	3	0,1
COMPV03	2	85%	2	0,1
COMPV03	3	85%	2	0,1
COMPV03	4	85%	3	0,1
COMPV04	1	85%	2	0,1
COMPV04	2	85%	1	0,1
COMPV04	3	85%	1	0,1
COMPV04	4	85%	1	0,1
COMPV04	5	85%	2,1	0,1
COMPV05	1	85%	2,5	0,1
COMPV05	2	85%	2	0,1
COMPV05	3	85%	1	0,1
COMPV05	4	85%	2	0,1
COMPV05	5	85%	2,5	0,1
COMPV06	1	85%	7	0,1
COMPV07	1	85%	3,1	0,1
COMPV07	2	85%	2,1	0,1
COMPV07	3	85%	1,3	0,1
COMPV07	4	85%	3	0,1
COMPV08	1	85%	3	0,1
COMPV08	2	85%	2,1	0,1
COMPV08	3	85%	1,2	0,1
COMPV08	4	85%	2,7	0,1
COMPV09	1	85%	1	0,1
COMPV09	2	85%	1	0,1
COMPV09	3	85%	1	0,1
COMPV09	4	85%	1	0,1
COMPV10	1	85%	2,5	0,1
COMPV10	2	85%	2	0,1
COMPV10	3	85%	1,5	0,1
COMPV10	4	85%	1,5	0,1
COMPV10	5	85%	2,5	0,1

Tabela 33 - Distâncias percorridas

Do local:	Para o Local:	Distância (km)
OSRF	C14	39,7
OSWE	C09	42,9
OSWE	C04	39,5
OSWE	C05	14,7
OSWE	C01	16,2
OSWE	C12	20,0
OSWE	C03	20,1
OSWE	C08	39,5
OSWE	C02	13,7
OSWE	C13	10,4
OSWE	C19	19,8
OSWE	C04	21,7
OSWE	C05	10,4
OSWE	C01	13,8
OSWE	C06	15,4
OSWE	C12	20,1
OSWE	C13	13,8
OSWE	C12	13,0
OSWE	C13	21,5
OSWE	C19	21,7
C01	C04	17,9
C01	C05	24,9
C01	C12	10,8
C01	C03	7,9
C01	C08	2,1
C01	C02	19,0
C01	C13	13,2
C02	C04	7,1
C02	C05	7,9
C02	C01	19,0
C02	C12	3,8
C02	C03	6,3
C02	C13	19,0
C02	C19	13,1
C03	C04	6,9
C03	C12	8,3
C03	C02	17,5
C03	C19	17,4
C03	C09	16,1
C05	C01	13,0

C05	C03	16,5
C05	C08	15,7
C05	C02	15,8
C05	C13	13,0
C07	C04	40,0
C07	C05	16,6
C07	C08	31,4
C07	C02	38,4
C07	C09	24,3
C08	C04	21,4
C08	C03	13,5
C08	C02	22,2
C08	C13	20,3
C12	C04	39,0
C12	C05	10,5
C12	C06	12,4
C12	C13	8,1
C12	C19	20,9
C13	C03	12,4
C13	C08	10,3
C13	C02	20,3
C13	C19	19,1
C19	C01	10,3
C19	C03	18,8
C19	C02	9,7
C19	C14	6,0

5.2 Geração de Cenários

A análise dos resultados do problema foram divididas em 8 cenários de teste, sendo executados os 14 dias de operação para cada um dos cenários gerados. Para o primeiro cenário não houve o uso de incertezas caracterizando assim um problema normal de otimização. Do segundo ao quarto cenário, uma variável de incerteza foi adicionada por cenário, consistindo em incerteza na demanda, disponibilidade de veículo e tempo de execução, respectivamente.

Nos cenários de 5 a 7 tem-se a presença de duas variáveis de incerteza e no cenário 8 tem-se a combinação das três variáveis de incerteza citadas, conforme ilustrado na tabela 38.

Tabela 38 - Divisão dos cenários

Nº de variáveis de incerteza:		1 variável			2 variáveis			3 variáveis	
Método	Real	Sem Incerteza	Incerteza na Demanda	Incerteza Disp. de veículo	Incerteza Tempo de viagem	Demanda +Disp.	Disp. +Tempo	Demanda +Tempo	Demanda +Disp. +Tempo
Teste	-	cenário 1	cenário 2	cenário 3	cenário 4	cenário 5	cenário 6	cenário 7	cenário 8

As tabelas completas com os resultados de cada cenário por dia de operação são encontradas nos apêndices de F à J.

5.3 Resultados Obtidos

A análise dos resultados está dividida em quatro seções: por demanda, custo, tempo de processamento e apontamentos técnicos. O método utilizado para análise das três primeiras seções (a seguir) é por profundidade; ou seja, inicia-se a análise dos resultados de uma forma geral e depois vai-se aprofundando em mais detalhes do problema para a sondagem das causas.

O início da análise é dado com uma comparação entre os resultados de todos os cenários gerados. Logo após, passa-se para uma verificação de um cenário específico com 14 dias de operação e, por último, analisa-se mais detalhadamente o cenário de apenas um dia. Vale ressaltar que para a última análise foi escolhido utilizar sempre o dia 15 de dezembro de 2011, conforme citado pela seção 5.1.1.

5.3.1 Demanda

Comparando a forma na qual os testes foram gerados, houve uma diferenciação para os cenários 2, 5, 7 e 8 por possuírem a presença da variável de incerteza na demanda. Para estes cenários foi necessário executar o programa de dois em dois dias de operação escalonadamente. Desta forma, o código pôde compensar os pedidos adiantados e pendentes decorrentes da incerteza na demanda, enquanto que, para os demais cenários, as soluções foram geradas dia a dia, desde que não exista iteração entre os dias de operação.

Na tabela 39 são dados os números de alterações encontradas para cada um dos cenários e, posteriormente, identificados pelo número de pedidos que foram adiantados ou pendentes. O número total de alterações na data de entrega aumentou de 18 (no cenário 7) para 19 (no cenário 8) com a adição da incerteza na disponibilidade de veículo, incrementando em um o número de alterações de pedidos pendentes.

Tabela 39 - Número de alterações da data de entrega

Nº de variáveis de incerteza:		1 variável			2 variáveis			3 variáveis	
Método	Real	Sem Incerteza	Incerteza na Demanda	Incerteza Disp. de veículo	Incerteza Tempo de viagem	Demanda +Disp.	Disp. +Tempo	Demanda +Tempo	Demanda +Disp. +Tempo
Teste	-	cenário 1	cenário 2	cenário 3	cenário 4	cenário 5	cenário 6	cenário 7	cenário 8
Nº alterações da data de entrega	-	0	18	0	0	19	0	18	19
Nº pedidos adiantados	-	0	8	0	0	8	0	8	8
Nº pedidos pendentes	-	0	10	0	0	11	0	10	11

Foram escolhidos os cenários 2 e 5 com o objetivo de fazer uma comparação do número de alterações da data de entrega versus o número de veículos que ficaram indisponíveis por dia de operação, conforme apresentado na tabela 40.

No cenário 2 é observado que o dia 20 de dezembro apresentou uma viagem pendente para ser realocada para o dia posterior. Entretanto, para este mesmo dia, no cenário 5, houve a ocorrência de indisponibilidade de um veículo, impactando no aumento de mais uma ordem pendente em relação ao cenário 2.

A presença de indisponibilidade de veículo também ocorreu para outros dias do cenário 5, porém estes não impactaram em mais alterações na demanda por dois motivos encontrados. O primeiro foi a baixa demanda no dia e o segundo motivo foi devido à indisponibilidade de veículos que não estavam sendo utilizados na programação de entregas, pois eram caminhões do tipo FT. Lembrando que veículos com pagamento de frete do tipo FT apresentam, em geral, um custo mais elevado do que do tipo FPT.

Tabela 40 - Análise de demanda nos cenários 2 e 5

Data	Cenário 2			Cenário 5		
	Nº pedidos adiantados	Nº pedidos pendetes	Nº de veículos pendetes	Nº pedidos adiantados	Nº pedidos pendetes	Nº de veículos pendetes
11/12/2011	0	0	0	0	0	0
12/12/2011	2	0	0	2	0	1
13/12/2011	1	2	0	1	2	0
14/12/2011	1	1	0	1	1	0
15/12/2011	0	1	0	0	1	0
16/12/2011	0	0	0	0	0	1
17/12/2011	0	1	0	0	1	0
18/12/2011	0	0	0	0	0	0
19/12/2011	2	0	0	2	0	2
20/12/2011	0	1	0	0	2	1
21/12/2011	0	3	0	0	3	0
22/12/2011	0	0	0	0	0	1
23/12/2011	2	1	0	2	1	0
24/12/2011	0	0	0	0	0	0
Total	8	10	0	8	11	6

5.3.2 Custo

O resultado final do custo obtido de cada um dos cenários gerados é apresentado pela tabela 41. O cenário 1, sem a presença de incerteza, é o cenário base das demais soluções dos cenários de 2 a 8. Nele foi obtido um resultado de custo menor se comparado ao real da operação executada.

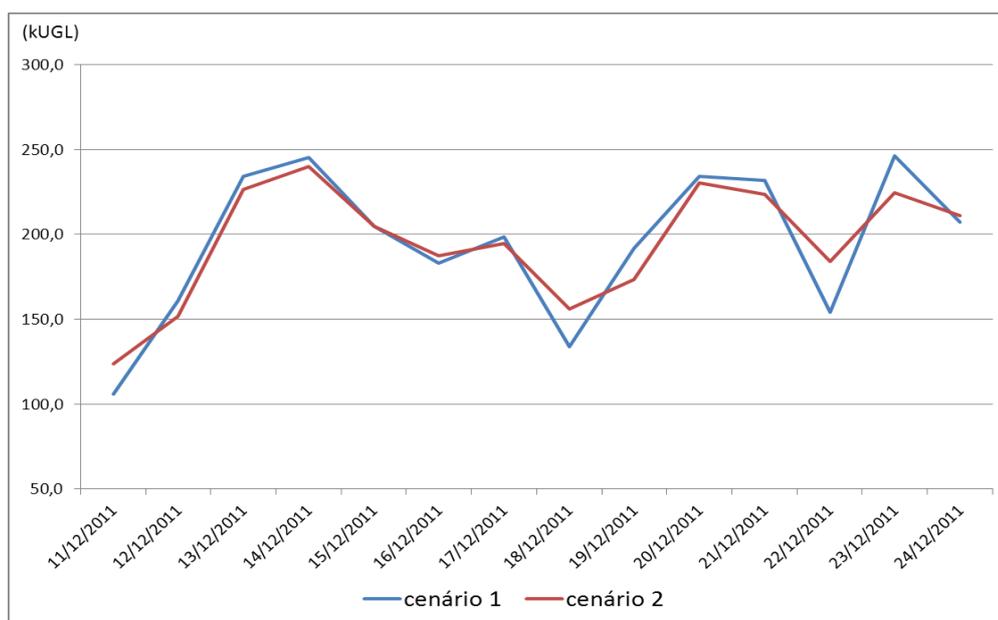
É importante ressaltar que na empresa, na qual o estudo de caso foi realizado, já existe um *software* para a alocação das viagens. Apesar da obtenção de um custo menor de operação no cenário 1, pelo sistema proposto, isso ainda não significa que o programa desenvolvido seja superior ao que já está sendo utilizado na empresa. Tal explicação é dada pelo fato do resultado da operação real já estar considerando variações reais na solução proposta pelo programa da empresa.

Tabela 41 - Custo final da operação (kUS\$)

Nº de variáveis de incerteza:			1 variável			2 variáveis			3 variáveis
Método	real	Sem Incerteza	Incerteza na Demanda	Incerteza Disp. de veículo	Incerteza Tempo de viagem	Demanda +Disp.	Disp. +Tempo	Demanda +Tempo	Demanda +Disp. +Tempo
Teste	-	cenário 1	cenário 2	cenário 3	cenário 4	cenário 5	cenário 6	cenário 7	cenário 8
Custo (kU\$)	83,26	81,77	78,13	82,88	81,87	80,43	82,30	79,93	80,96

O menor custo encontrado foi no cenário 2. Nele observa-se o impacto da flexibilização da demanda de um dia para outro, conforme comentado na seção 5.3.1, já que a existência da possibilidade do programa encontrar melhores soluções na alteração da data de entrega minimiza a utilização de veículos com tarifas mais caras.

O efeito de flexibilização gerado pela incerteza na demanda é visualizado através da comparação da flutuação da demanda dos cenários 1 e 2, conforme mostra a figura 25, minimizando o desvio padrão da demanda do cenário 1 para o cenário 2 de 41,9kUGL para 33,2kUGL, em relação a uma média de demanda diária igual a 195,1kUGL. Vale ressaltar que, apesar deste efeito minimizar diretamente o custo final da solução, sua utilização foi restringida pelos parâmetros $L_{maxn}(l)$ e $P_{maxd}(o,l)$ citados na tabela 2.

**Figura 25 - Distribuição da demanda, cenários 1 e 2.**

Ao contrário do efeito do cenário 2, no cenário 3 é obtido o maior impacto no aumento do custo final da solução, refletindo-se este aumento no incremento do custo dos demais cenários 5, 6 e 8, que também possuem a presença da incerteza na disponibilidade de veículos.

A indisponibilidade de veículos aumentou o custo da solução final do cenário 3 em 1,4% em relação ao cenário 1, ficando 0,5% abaixo da solução real executada. Entretanto, ao agregar, no resultado do cenário 3, o cenário 4 (com incerteza no tempo de viagem), que está 0,1% acima do resultado da solução do cenário 1, faz com que o cenário 6 alcance uma diferença de apenas 0,3% abaixo da solução real executada. Cabe observar que o cenário 6 é a combinação dos cenários 3 e 4. Desta forma é importante observar que o cenário mais próximo do resultado da operação real executada é o cenário de número 6, sendo este cenário o mais recomendado para o desenvolvimento de soluções em nível operacional, já que ele gera soluções mais próximas da real que são executadas diariamente, conforme pode ser visualizado através da figura 26.

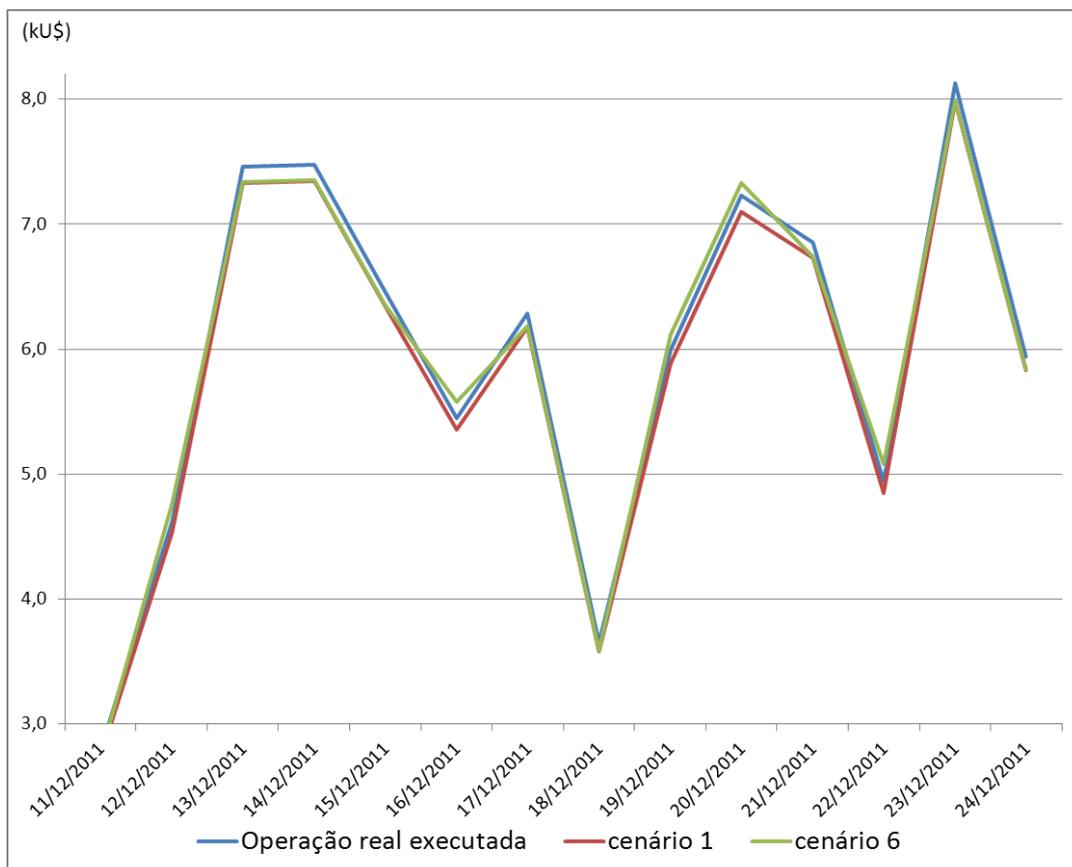


Figura 26 - Custo diário da operação real executada versus cenários 1 e 6.

Para os demais cenários, 5 e 7, que apresentam duas variáveis de incerteza, ocorreu uma diminuição do custo final pela presença de incertezas na demanda, assim como para a solução final do cenário 8.

Analisando os cenários, não houve a possibilidade de redução de veículos do tipo FPT, pois o fato de todos apresentarem tarifa mensal, CT_{month} , fez com que a utilização destes fosse priorizada na alocação de recursos. Porém, para os veículos do tipo FT, a análise de uso destes veículos não chega a ser viável, pois não existe um contrato fixo neste tipo de pagamento de frete. Neste contexto, foi observado que os cenários que apresentam incerteza na demanda apresentaram uma melhor utilização dos veículos do tipo FPT, gerando as reduções de custo observadas nos cenários 2, 5, 7 e 8. Conclui-se que, para análise de operação, o cenário 8 é o mais apropriado, pois nele está presente o efeito do cenário 6 de incerteza de disponibilidade de veículo e tempo de execução, porém com o adicional da incerteza na demanda.

Uma análise interessante para detalhar é a diferença diária entre os cenários 6 e 8, conforme ilustrado pela figura 20, já que estes foram os eleitos para a utilização no nível operacional e tático de produção, respectivamente.

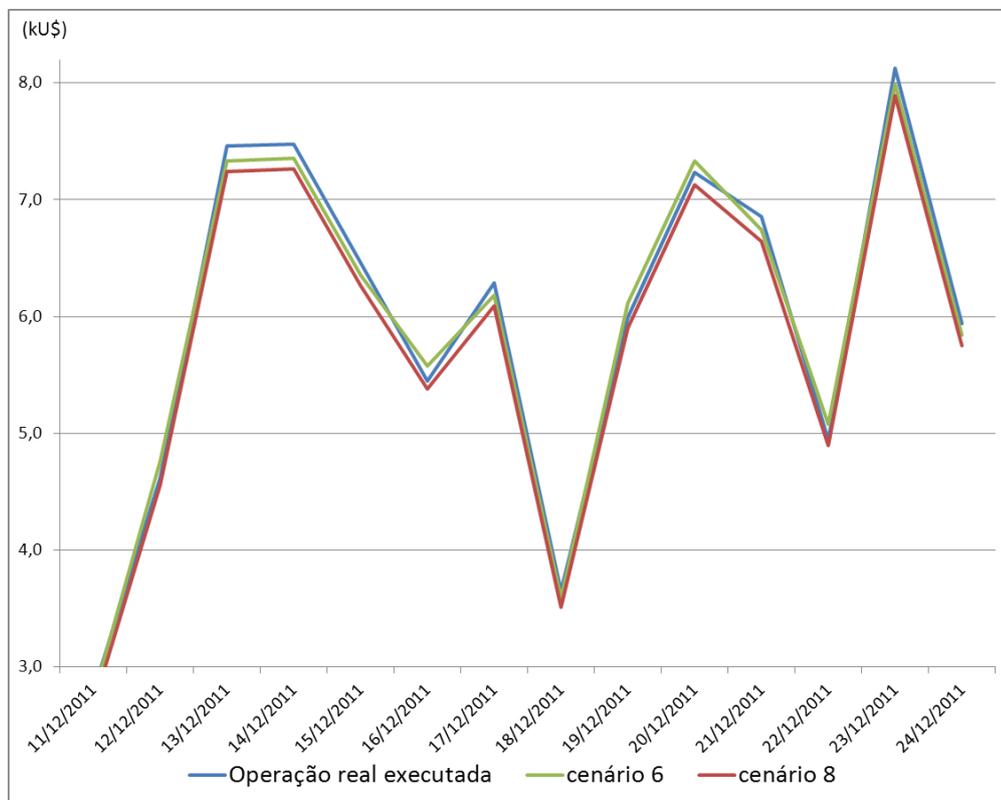


Figura 27 - Custo diário da operação real executada versus cenários 6 e 8.

Também é válido analisar os efeitos criados pelo cenário 4, onde além do aumento do custo em relação ao cenário 1, causado pelo aumento do tempo de execução, também apresentou outro efeito possível de ser observado apenas na comparação diária da programação de entregas. Analisando os resultados do dia 15 de dezembro dos cenários 1 e 4 (figura 28), foi possível observar no cenário 4 (em relação ao 1) a realocação de quatro viagens que se alternaram entre si. Ou seja, duas viagens de maior distância ao cliente foram realocadas de um horário de maior tráfego para substituir outras duas viagens de menor distância ao cliente em um período de menor tráfego. Nesta comparação, foi observado um número maior de viagens mais distantes em horários de baixo e médio tráfego. Tendo em vista que o tempo total de viagem é a soma do tempo de viagem mais o tempo gasto no terminal e cliente, isso fez com que o aumento do tráfego gerasse um aumento do tempo de viagem dos clientes mais distantes ao terminal, comparado aos clientes mais próximos. Isso fez com que os veículos permanecessem mais tempo nos terminais e clientes em horários de alto tráfego e menos tempo nas estradas com alto tráfego. Este efeito é ilustrado pela figura 28. Vale ressaltar que tal efeito foi positivo na otimização do tempo total de viagem da programação de entregas do cenário 4, porém, como a discretização horária utilizada no programa foi de uma em uma hora, acredita-se que esta discretização “freou” o impacto do efeito observado.

O efeito criado pela discretização do tempo em uma hora também pode ser interpretado como uma segurança nos resultados de tempo de viagem. Ou seja, o aumento da discretização minimiza o erro de possíveis entregas atrasadas, por superestimar tempos de viagem e garantir a entrega dentro do período de tempo estimado. Na perspectiva do presente estudo de caso, ela também pode ser apontada como incerteza no tempo de execução da viagem, pois, flexibiliza o esperado horário de entrega de uma viagem.

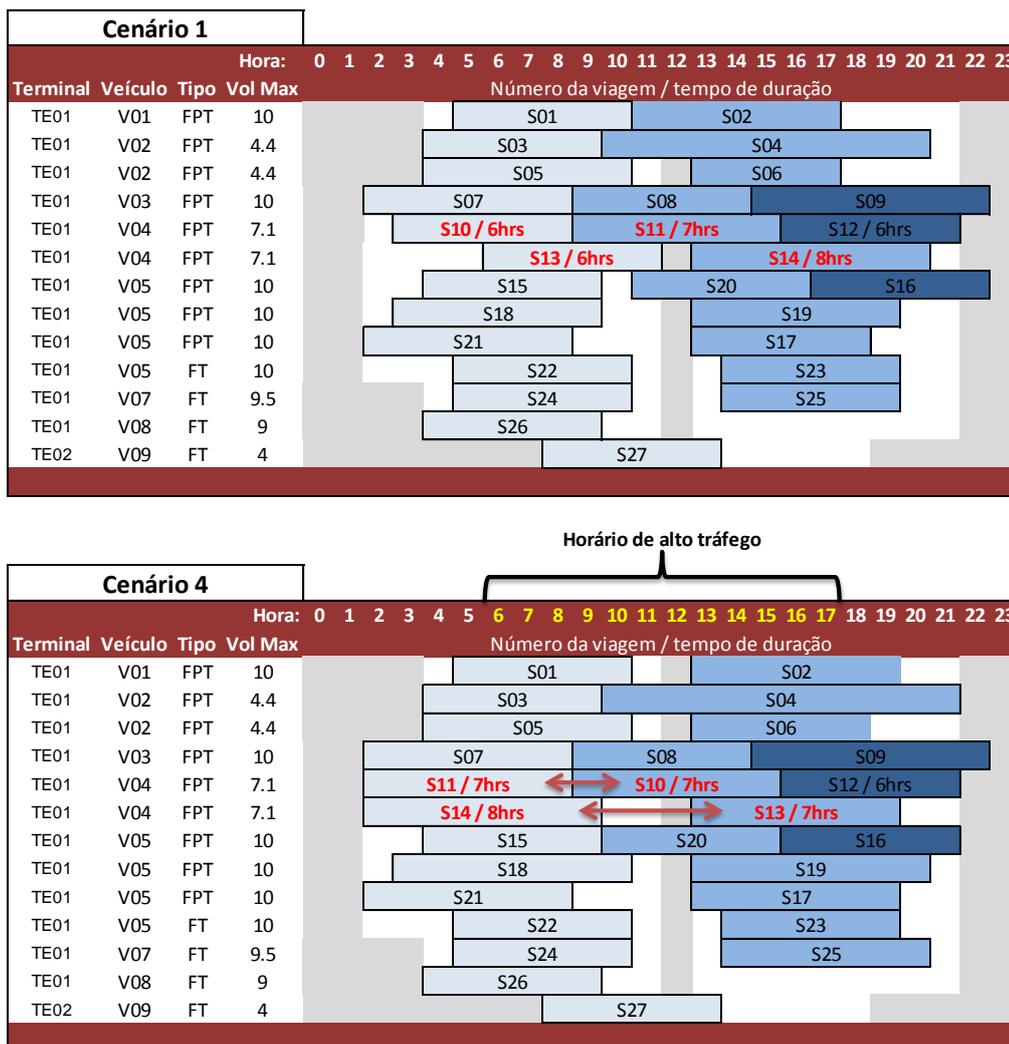


Figura 28 - Programação de entregas do dia 15 de dezembro para os cenários 1 e 4.

5.3.3 Tempo de processamento

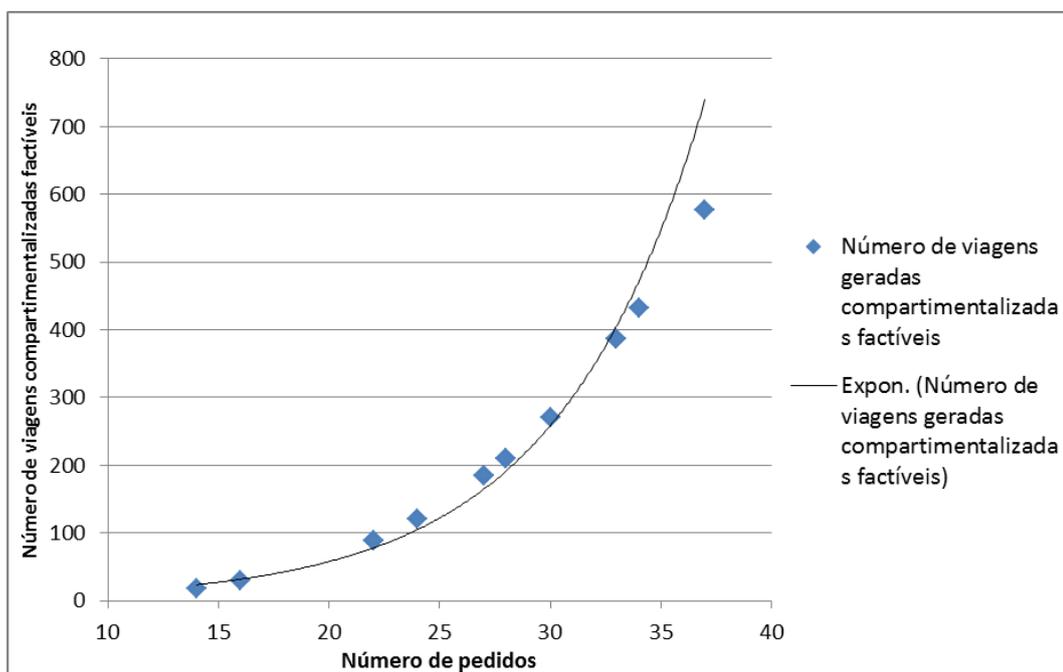
Embora o tempo de processamento não faça parte dos objetivos específicos do presente estudo, este é apresentado por contribuir nos apontamentos técnicos do desenvolvimento do programa.

Os valores apresentados na tabela 42 apresentam o tempo médio de processamento por dia de operação, separados pelo gerador de viagens compartimentalizadas e pelo alocador de viagens. Todos os testes foram realizados em um Computador Intel Core i5, CPU M560 2.6GHz / 2.6GHz, Memória RAM 4GB, sistema operacional Windows 7 32-bit, com banco de dados do software Microsoft Access 2010.

Tabela 42 - Tempo de processamento médio (hh:mm:ss)

Número de variáveis de incerteza:			1 variável			2 variáveis			3 variáveis
Método	Real	Sem Incerteza	Incerteza na Demanda	Incerteza Disponibilidade de veículo	Incerteza Tempo de viagem	Demanda +Disponibilidade	Disponibilidade +Tempo	Demanda +Tempo	+Disponibilidade +Tempo
Teste	-	cenário 1	cenário 2	cenário 3	cenário 4	cenário 5	cenário 6	cenário 7	cenário 8
Compartimentalizador	-	01:24:51	01:10:14	01:24:51	01:24:51	01:10:14	01:24:51	01:10:14	01:10:14
Alocador	-	00:06:18	00:05:40	00:07:00	00:06:37	00:06:55	00:06:41	00:06:05	00:06:33
Total	-	01:31:09	01:15:54	01:31:51	01:31:28	01:17:09	01:31:32	01:16:19	01:16:47

Analisando os dados, observa-se um alto valor gasto no tempo de processamento do compartimentalizador, consumindo mais de 90% do tempo total de processamento utilizado pelo programa. Ao verificar sua causa foi identificado que o código que o controla testa todas as possibilidades possíveis de compartimentalização, gerando um número exponencial de possibilidades de acordo com o número de pedidos a ser alocado, conforme figura 29.

**Figura 29 -** Número de viagens compartimentalizadas versus número de pedidos

Nos primeiros testes realizados com a ferramenta, ao executar o programa não foi possível chegar a um resultado final para muitos dias de programação,

devido ao alto tempo de processamento. Assim, foram tomadas duas medidas para a redução do tempo de processamento no gerador de viagens compartimentalizadas. Como primeira medida, houve a limitação do número de pedidos combinados por viagem em no máximo três e a segunda medida foi a adição de restrições do problema antes da tentativa de compartimentalização. Entretanto, o tempo de processamento do compartimentalizador continuou expressivo, conforme resultados indicados pela tabela 42. Eles sugerem que a arquitetura do programa proposta poderia ser mais eficiente utilizando o compartimentalizador controlado pelo alocador de viagens. Ou seja, ao invés do compartimentalizador gerar uma lista de possíveis viagens factíveis, ele seria controlado pelo alocador de viagens e executado somente quando solicitado. Adicionalmente, seria válida a criação de uma lista com o histórico de viagens já compartimentalizadas, evitando desta forma o reprocessamento.

5.3.4 Apontamentos técnicos

Após a análise dos resultados obtidos, três pontos relevantes foram identificados, sendo dois relacionados ao desenvolvimento da solução e o terceiro com o desempenho das variáveis de incerteza. Os dois pontos relacionados com o desenvolvimento já foram previamente apresentados na seção 4.3, consistindo no uso de código Hash para o cálculo horário e do carregamento dos dados utilizando variáveis vetoriais. O terceiro ponto está relacionado à utilização de variáveis randômicas na probabilidade de ocorrência dos eventos de incerteza.

i) Código Hash

Foi observado que a utilização do código Hash, além de facilitar o desenvolvimento do programa quanto à linguagem de programação, também gerou uma economia no volume de dados utilizados, já que tabelas horárias de um dia inteiro puderam ser compactadas em apenas um registro de informação.

Como já comentado na seção 5.3.2, o grau de precisão de 1 hora, para mensurar as janelas horárias, pode ter restringido o impacto da incerteza de indisponibilidade de veículo. Assim, futuros desenvolvimentos no aumento da precisão das janelas horárias fazem com que o uso de código Hash seja ainda mais

expressivo. Ele faria com que o volume de dados do sistema continuasse relativamente baixo, contudo, aumentando o número de informações necessárias para a programação mais precisa. Outro ponto interessante seria a facilidade de manuseio, pois ele permite a adaptação do programa sem grandes alterações no código.

É importante apontar que o uso da abordagem proposta poderia ser utilizada não apenas para auxiliar a implementação em linguagem de programação, mas também para a resolução de problemas em programação matemática. Abrindo possibilidades de futuros estudos na área de pesquisa operacional, como por exemplo, o impacto do uso de código Hash na formulação matemática para a otimização de problemas com um alto volume de informações.

ii) Carregamento dos dados

Nas seções 4.1.1 e 4.1.2 foram descritos os passos utilizados para a elaboração do código em Busca Tabu, onde o carregamento dos dados do problema é incluído como o primeiro passo da resolução do problema, visto que este foi determinado como um passo importante na elaboração do código.

Os principais dados utilizados para a geração da solução (tais como pedidos dos clientes, entregas e veículos disponíveis) são transferidos do formato de tabela do banco de dados para parâmetros compostas de três ou quatro dimensões, conforme apêndice B. Isso facilita o manuseio da informação na elaboração do código e aumenta a velocidade de processamento do programa.

iii) Parâmetros randômicos

Um ponto relevante observado é quanto à forma na qual os parâmetros randômicos são gerados para a utilização no problema, pois a distribuição dos valores aleatórios originados nem sempre reflete uma distribuição realista do parâmetro específico do problema analisado. Desta forma, apesar dos resultados obtidos com o uso de incertezas serem satisfatórios, acredita-se que soluções mais refinadas podem ser encontradas através do aperfeiçoamento das variáveis randômicas. Como exemplo, é possível sugerir o uso de métodos estocásticos na sua geração. Vale ressaltar que este ponto é também mencionado por Ben-Tal e Nemirovski (2000).

6 Conclusão

O objetivo deste estudo é decidir como alocar os veículos aos clientes, dado que a consolidação de cargas pode ser de entregas completas ou fracionadas, determinando a quantidade a ser entregue com cada veículo e seus respectivos compartimentos para cada cliente. Nos primeiros quadrimestres da pós-graduação foi dedicado a entender o maior número possível de obras literárias e formas de resolução do problema, abrangendo a utilização de métodos exatos, estocásticos e heurísticos, assim como outras abordagens não muito exploradas na área, que dão um maior nível de complexidade ao problema, como a presença de incertezas. Após a revisão da literatura, os demais quadrimestres foram dedicados à implementação da solução através do desenvolvimento em linguagem computacional de uma ferramenta de otimização baseada em métodos heurísticos, com a finalidade de ser utilizada como bancada de teste para a análise dos vários cenários gerados com e sem a presença de incertezas.

Adicionalmente, uma importante contribuição no desenvolvimento do trabalho foi o conhecimento do autor na área de sequenciamento e modelamento de frota, pois já atua no ramo por seis anos em uma empresa multinacional, contribuindo com a exatidão no detalhamento das restrições e conhecimento prévio de onde se encontra as maiores incertezas do processo.

No campo acadêmico, o presente estudo contribui com a literatura nas áreas correlatas de manufatura e logística, pela complexidade do problema proposto com seu número de restrições e a abordagem do uso de incertezas. Foi explorada a lacuna existente entre soluções reais e teóricas que as propostas matemáticas usualmente oferecem para as duas áreas de estudo.

Os resultados apontam que para o estudo de caso proposto o custo das soluções teóricas encontradas estão em torno de 1,8% defasadas em relação aos das soluções práticas. Observou-se que esta lacuna pode ser minimizada em 74% pela incerteza de disponibilidade de veículo, em 7% pela incerteza no tempo de execução das tarefas e combinadas produzem uma redução de 36% da lacuna. Já a utilização de incerteza na demanda reduziu a solução teórica encontrada, aumentando a lacuna de 1,8% para 6,2%, e que entretanto, ao combinar a incerteza

de demanda com disponibilidade de veículo e tempo de execução o aumento da lacuna é reduzido para 4%.

Concluiu-se que a lacuna existente entre soluções matemáticas e a prática é minimizada no nível operacional com a utilização de incertezas no tempo de execução da operação e na disponibilidade de recursos utilizado. Onde, caso uma empresa do ramo implementa-se um programa de flexibilização de demanda, esta poderia reduzir ainda mais seu custo de entrega a ponto de compensar as incertezas geradas pelo tempo de execução e disponibilidade de recursos.

Para a área de pesquisa operacional, este estudo oferece uma literatura complementar no campo de métodos heurísticos, pois testa a viabilidade do uso da metodologia em diferentes cenários com diferentes formas de pagamento de frete e utilização de variáveis de incerteza. Entretanto como houve uma variação no método Busca Tabu utilizado, propõe-se como trabalho futuro a comparação do método utilizado no presente trabalho versus o método de Busca Tabu clássico.

Na área computacional, a implementação da solução diretamente em linguagem de programação permitiu apontar a possibilidade do uso de código Hash na mensuração das janelas horárias. E com o uso de variáveis randômicas na geração das incertezas, constatou-se a necessidade de uma análise estatística com um maior número de execuções do programa para cada cenário gerado, garantindo uma maior confiabilidade dos resultados obtidos.

7 REFERÊNCIAS BIBLIOGRÁFICAS

ARNOLD J. R. TONY, **Introduction To Material Management**, 2nd Edition, 1996.

BALASUBRAMANIAN, J., & GROSSMANN, I. E. **A novel branch and bound algorithm for scheduling flowshop plants with uncertain processing times.** Computers and chemical engineering, 26, 41–57 2002.

BASSETT, M. H., PEKNY, J. F., REKLAITIS, G. V..**Using detailed scheduling to obtain realistic operating policies for a batch processing facility.** Industrial engineering and chemical research, 36, 1717–1726 1997.

BATEMAN THOMAS S., SNELL, SCOTT A. **Administração: construindo vantagem competitiva.** São Paulo: atlas, 1998.

BEALE, E.M.L.. **On minimizing a convex function subject to linear inequalities.**Journal of royal statistics society 17b, 173-184 1955.

BEN-TAL A., NEMIROVSKI, **A robust solutions of linear programming problems contaminated with uncertain data.** Mathematicalprogramming, 88, 411–424 2000.

BOTELHO F. C.. **Estudo comparativo do uso de hashing perfeito mínimo.** Dissertação de mestrado, dep. Ciência da computação da ufmg. Novembro, 2004.

BIRGE, J.R..**Decomposition and partitioning methods for multistage linear programs.** Operations research 33, 1089±1107, 1985.

BRADLEY, S. P., HAX A. C., MAGNANTI, T. L.. **Applied Mathematical Programming.** Addison-Wesley publishing Company, Inc, Massachusetts, EUA, (1977).

CAUX, C., BRUNIAUX, R., PIERRAVAL, H.. **Cell formation with alternative process plans and machine capacity constraints: a new combined approach.** Int. J. Production economics64, 279-284, 2000.

CICHELLI R. J. **Minimal perfect hash functions made simple**. Communications of the acm, 27 (4): 384-387, December, 1984.

CHOOTINAN CHEN A. P., RECKER, W.W.. **Examining the quality of synthetic origin-destination trip table estimated by path flow estimator**. Journal of transportation engineering, 131, 506-513, 2005.

DANTZIG, G.B..**Linear programming under uncertainty**. Management science 1, 197±206, 1955.

ESCUDERO LAUREANO F., QUINTANA FRANCISCO J., SALMAREÓN JAVIER CORO. **A Modelling And An Algorithmic Framework For Oil Supply, Transformation And Distribution Optimization Under Uncertainty**, European Journal Of Operational Research Volume 114, Issue 3, 638-656, 1999.

FONSECA, CARLOS M., FLEMING PETER J.**Genetic algorithms for multobjective optimization: formulation, discussion and generalization**, dept. Automatic control and system eng. Sheffield s1 4du, uk, 1991.

HEEMAN LEE, PINTO M. JOSE, IGNACIO E. GROSSMANN, SUNWON PARK, **Mixed-Integer Linear Programming Model For Refinery Short-Term Scheduling Of Crude Oil Unloading With Inventory Management**, Department Of Chemical Engineering, 305-701, 1996.

HIGGINS, A.J. **A dynamic Tabu search for large-scale generalised assignment problems**. Computers & operations research 28, 1039-1048, 1999.

GASSMANN, H.I.. **MSLIP a computer code for the multiusage stochastic linear programming problem**. Mathematical programming 4, 407-423, 1990.

GLOVER F.. **Future paths for integer programming and links to artificial intelligence**. Computer and operations research 13, 33-159, 1986.

GLOVER F.. **Tabu search part ii**. Orsa journal on computing 2, 4-32., 1990.

GLOVER F. e Laguna, M.. **Tabu search**. Kluwer Academic Publishers, Boston, 1997.

IERAPETRITOU, M. G., FLOUDAS, C. A.. **Effective continuous-time formulation for short-term scheduling: 1. Multipurpose batch processes**. Industrial engineering and chemical research, 37, 4341–4359, 1998.

IERAPETRITOU, M. G., HENÉ, T.S., FLOUDAS, C. A.. **Effective continuous-time formulation for short-term scheduling.3. Multiple intermediate due dates**. Industrial engineering and chemical research, 38, 3446–3461, 1999.

INMETRO. **Regulamento técnico da qualidade 7c - inspeção na construção de equipamentos com pressão máxima de trabalho admissível de 690 kpa para o transporte rodoviário de produtos perigosos a granel – líquidos**, 2009.

IPEA E ANTP. **Redução Das Deseconomias Urbanas Pela Melhoria Do Transporte Público**; Instituto De Pesquisa Econômica Aplicada - Ipea, Em Parceria Com A Associação Nacional De Transporte Público –Antp, 1997.

IRANI, S.A.. **Handbook Of Cellular Manufacturing Systems**, John Wiley, New York, 1999.

JOHNSON, S.M.. **Optimal two and three stage production schedules with setup times included**. Naval research logistics quarterly, 1, 61-67,1954.

LEVNER E., KOGAN K., LEVIN, I. **Scheduling a two-machine robotic cell: a solvable case**. Annals of operations research, 57, 217-232, 1995.

LISBOA, ÉRICO. **Pesquisa operacional**, Apostila da disciplina. Rio de Janeiro – RJ, (2002).

LO HK, TUNG Y-K. **Network with degradable links: capacity analysis and design**. Transp res 37b:345–363, 2003.

LOURIVAL, F. COELHO, **Algoritmos**, Departamento De Informática E Matemática AplicadaUfrn, 2011.

LOUVEAUX, F.V., BIRGE, J.. **Introduction to stochastic programming**. Springer, berlin, 1997.

MAGGU, P. L., DAS, G. E KUMAR, R. **On equivalent-job for job-block in 2xn sequencing problem with transportation-times**. Journal of the operations research society of japan, 24, 136-146, 1981.

MAJUMDAR, J. E BHUNIA, A.K.. **Elitist genetic algorithm for assignment problem with imprecise goal**. European journal of operational research 177, 684-692, 2005.

MULVEY, J.M., VANDERBEI R.J., ZENIOS, S.A.. **Robust optimization of large-scale systems: general modeling framework and computations**. Operations research 43, 244±281, 1995.

NAGI, R., HARHALAKIS, G., PROTH, J.M.. **Multiple routing and capacity considerations in group technology applications**. Int j. Production research 28, 2243-2257, 1990.

NSAKANDA, A.L., DIABY, M., PRICE, W.L.. **Hybrid genetic approach for solving large scale capacitated cell formation problems with multiple routings**. European j. Of operational research 171, 1051 – 1070, 2006.

PUCCINI, A. L.; PIZZOLATO, N. D.. **Programação linear**, 2nd edição, Rio de Janeiro, RJ, (1990).

ORÇUN, S., ALTINEL, I. K., HORTAÇSU, Ö. **Scheduling of batch processes with operational uncertainties**. Computers and chemical engineering, 20, s1191–s1196, 1996.

RAMAN N. TALBOT, F. B. E RACHAMADUGU, R. V. **Simultaneous scheduling of machines and material handling devices in automated manufacturing**. Second conference on flexible manufacturing systems, 321-332. El Sevier science publishersb.v., Amsterdam, 1986.

ROCKAFELLAR, R.T., WETS, R.J.. **Scenario and policy aggregation in optimization under uncertainty**. Mathematics of operations research 16, 119-147, 1991.

ROSS, G. TERRY, SOLAND RICHARD M., **A Branch And Bound Algorithm For The Generalized Assignment Problem**, Mathematical Programming, Volume 8, Number 1, 91-103, 1974.

SAHINIDIS, NIKOLAOS V.. **Optimization Under Uncertainty: State-Of-Art And Opportunities**, El Selvier, Computers And Chemical Engineering 28, 971-983, 2004.

SANMARTI E., ESPUÑA, A., PUIGJANER, L.. **Batch production and preventive maintenance scheduling under equipment failure uncertainty**, 1997.

SPRUGNOLI R.. **Perfect hashing functions: a single probe retrieving method for static sets**. Communications of the acm, 20 (11): 841-850, November, 1977.

STERN, H. L., E VITNER, G.. **Scheduling parts in a combined production-transportation work cell**. Journal of the operational research society 625-632, 1990.

SUMALEE, A., CONNORS, R., WATLING, D. **An optimal toll design problem with improved behavioral equilibrium model: the case of the probit model**. Mathematical and computational models for congestion charging. Springer, 2006.

TOKTAS, B., J. YEN E Z. B. ZAMBINSKY. **Addressing capacity uncertainty in resource-constrained assignment problems**. Computers & operations research 33, 724-745, 2004.

VAN SLYKE, R., WETS, R.J.-B.. **L-shaped linear programs with applications to optimal control and stochastic programming**. Siamjournal on applied mathematics 17, 638±663, 1969.

WALLER, S.T., J.L. SCHOFER, A.K. ZILIASKOPOULOS. **Evaluation with traffic assignment under demand uncertainty**. Transportation research record: journal of the transportation research board 177169–74, 2001.

WOODCOCK, ANDRE J. E WILSON, JOHN M. **A hybrid Tabu search/branch & bound approach to solving the generalized assignment problem.** European journal of operational research 207, 566-578, 2010.

XIAOXIA LIN, STACY L. JANAK, CHRISTODOULOS A. FLOUDAS. **A new robust optimization approach for scheduling under uncertainty:** i. Bounded uncertainty, el selvier, computers and chemical engineering 28 1067-1085, 2004.

YANG, H., M.G.H. BELL.. **Models and algorithms for road network design: a review and some new developments,** transport reviews 18 (3), pp.257-278, 1998.

YIN, ROBERT K. **Estudo de caso: planejamento e métodos.** 2 ed. Porto alegre bookman, 2002.

APÊNDICES

APÊNDICE A – FUNÇÃO DE HASH

Segundo Botelho (2004), “para conjuntos estáticos, é possível computar uma função $h(x)$ para encontrar qualquer chave na tabela em uma única tentativa, sem a ocorrência de colisão. Esta função é chamada função hash perfeita. Uma função hash perfeita que preserva a ordem previamente estabelecida entre as chaves é chamada função hash perfeita em ordem preservada. Uma função hash perfeita que mapeia um conjunto de chaves de tamanho n em endereços de uma tabela hash de igual tamanho é chamada de função hash perfeita mínima. Uma função perfeita mínima pode evitar totalmente o problema de desperdício de espaço e de tempo.”

Em outras palavras pode-se dizer que uma função hash é a transformação de uma grande quantidade de informação em uma pequena quantidade de informação, em uma sequência que busca identificar um arquivo ou informação unicamente. Ou seja, um método para transformar dados de tal forma que o resultado seja exclusivo, como no caso da função hash perfeita mínima. A função domínio $h(x)$ recebe um valor de um determinado tipo e retorna um código de endereçamento, enquanto a função contradomínio $h^{-1}(x)$ retorna o dado inicial codificado.

Diferentes tipos de algoritmos de função hash perfeita mínima estão disponíveis na literatura, alguns exponenciais no tamanho do conjunto de chaves, como em Cichelli (1984) e Sprugnoli(1977), no qual o presente trabalho utiliza como base para codificar as janelas horarias de funcionamento em um único código representativo.

Foi estabelecido que a precisão do sistema é de 1 hora e desta forma tem-se que a função hash de entrada, domínio, é dada pela equação 12.

$$h(x_{0...23}) = \sum_{n=0}^{23} 2^{n*x} \quad \therefore x_{0...23} \in \{0,1\} \quad (12)$$

A variável é x booleana e considera ou não a hora informada como válida. Ou seja, para uma janela horária das “00:00 horas” às “18:00 horas” de um dia a variável $x=1$ para $n = 0...18$ e $x=0$ para $n = 19...23$, gerando um código de hash perfeito mínimo igual a 524287. A figura 30 apresenta o código utilizado para gerar a equação 12.

```

'Função Hash h(x)
tb24.MoveFirst
TimeCodeCalc = 0
For I = 0 To 23
    If tb24(I).Value = 1 Then TimeCodeCalc = TimeCodeCalc + 2 ^ I
Next

```

Figura 30 - Função Hash domínio

Já a função hash de saída, contradomínio, é dada pela equação 13 e representada pelo código na figura 31.

$$h^{-1}(x_{0...23}) = \text{Int} \left\{ \frac{\log(h(x) - 2^{n \cdot x})}{\log 2} \right\} \quad \text{para } n = 1 \dots 23 \quad (13)$$

```

TimeCodeTemp = 0
Do Until TimeCodeIN - TimeCodeTemp = 0
    'Função Hash h-1(x)
    'Foi usado a hr(n)>> 2^n e depois somados para distinguir as hrs do dia.
    TimeCodeCalc = Int(Log(TimeCodeCalc - TimeCodeTemp) / Log(2))
    tb24(TimeCodeCalc).Value = "1"
    TimeCodeTemp = TimeCodeTemp + 2 ^ TimeCodeCalc
Loop

```

Figura 31 - Função Hash contradomínio

APÊNDICE B – VARIÁVEIS VETORIAIS

Segundo Lourival (2001) “Quando uma determinada Estrutura de dados é composta de variáveis com o mesmo tipo básico, temos um conjunto homogêneo de dados. Variáveis compostas homogêneas, também chamadas de variáveis indexadas, correspondem a um conjunto de variáveis do mesmo tipo, referenciáveis pelo mesmo nome e individualizadas entre si através de sua posição dentro desse conjunto (os índices). Uma variável indexada pode ser definida contendo um ou mais índices. Quando possui um único índice a variável é chamada de vetor, quando possui dois índices a variável é chamada de matriz e com três ou mais índices não recebe nome especial, também, na prática, sua ocorrência é pouco frequente. Ao número de índices necessários à localização de um componente dentro da variável indexada dá-se o nome de dimensão.”

Assim o presente estudo utiliza de variáveis compostas homogêneas de três e quatro dimensões para a manipulação de dados, conforme exemplo ilustrado pela figura 32. Na figura é possível notar a criação de três variáveis compostas, sendo duas de três dimensões e 1 de quatro dimensões.

```
'Order and Shipment buffer:
'O-Best 1-Current 2-Last / 3000.ShipmentBuffer
'/ 0.ShipmentNumber, 1.Used {TABU} 2.ShipmentDate, 3.Vehicle, 4.Terminal, 5.DateChange, 6.VolChange, 7.TABU string
Dim Shipment(2, 3000, 7) As String
'O-Best 1-Current 2-Last/ 3000.Order / 0.OrderNumber, 1.Used, 2.OrderDate,
Dim Order(2, 3000, 2) As String
'Allocate Solution:
'O-Best 1-Current 2-Last / 199.Vehicle / xx00=TotalVehicle per day 01-99yy=Shipment=TripSequence per day
'/ 0.VehicleID 1.ShipmentDate 2.WeekdayHrCODE 3.AvailableHrCODE, 4.FirstHrAv, 5.Shipment, 6.StartTime, 7.EndTime,
' 8.TripTime, 9.TripDist, 10.Cost 11.ZCost, 12.Tabu 13.CounterTABU
Dim AllocateSol(2, 199, 3300, 13) As String
```

Figura 32 - Declaração de variáveis composta

APÊNDICE C – UNIDADES DE TRABALHO

A forma na qual o programa desenvolvido organiza as unidades dimensionais de trabalho utilizadas no sistema é através da criação de grupos de dimensões, definidos pelos índices u . Em cada grupo u são determinadas as unidades de medida dos grupos de dados que estão sendo trabalhados, como kg para quilogramas, km para quilômetros, UGL para a medida de volume em galões, dentre outros. Especificado conforme figura 33 e denotação da tabela 43.

As unidades são classificadas em cinco diferentes categorias: por unidade de volume, peso, comprimento, distância de percurso e moeda, respectivamente dada pelos campos VOLUME_UNIT, WEIGHT_UNIT, LENGTH_UNIT, DISTANCE_UNIT e CURRENCY_UNIT e classificada pelo campo UNIT_GROUP.

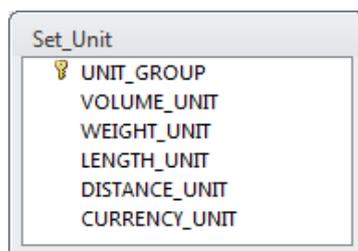


Figura 33 - Unidade de trabalho

Tabela 43 - Unidades de cliente

Índice		
u	Conjunto das unidades de trabalho, $u = \text{unit1, unit2 ... , unit10}$	UNIT_GROUP
Parâmetros		
$UV_{(u)}$	Unidade de volume referente ao índice u ex.: UGL, m3.	VOLUME_UNIT
$UW_{(u)}$	Unidade de peso referente ao índice u , ex.: kg, TON.	WEIGHT_UNIT
$UL_{(u)}$	Unidade de comprimento referente ao índice u , ex.: m, km.	LENGTH_UNIT
$UD_{(u)}$	Unidade de distância dos trechos referente ao índice u , ex.: km, milhas.	DISTANCE_UNIT
$UC_{(u)}$	Unidade monetária referente ao índice u , ex.: R\$, \$.	CURRENCY_UNIT

APÊNDICE D – ROTINA PRINCIPAL DO MÓDULO MCOMPART_TABU

```

Option Compare Database
'Order and Shipment buffer:
'O-Best 1-Current 2-Last / 3000.ShipmentBuffer
'/ 0.ShipmentNumber, 1.Used (TABU) 2.ShipmentDate, 3.Vehicle, 4.Terminal, 5.DateChange, 6.VolChange, 7.TABU string
Dim Shipment(2, 3000, 7) As String
'O-Best 1-Current 2-Last/ 3000.Order / 0.OrderNumber, 1.Used, 2.OrderDate,
Dim Order(2, 3000, 2) As String
'Allocate Solution:
'O-Best 1-Current 2-Last / 199.Vehicle / xx00=TotalVehicle per day 01-99yy=Shipment=TripSequence per day
'/ 0.VehicleID 1.ShipmentDate 2.WeekdayHrCODE 3.AvailableHrCODE, 4.FirstHrAv, 5.Shipment, 6.StartTime, 7.EndTime,
' 8.TripTime, 9.TripDist, 10.Cost 11.ZCost, 12.Tabu 13.CounterTABU
Dim AllocateSol(2, 199, 3300, 13) As String
Dim CNT0 As Integer, CNT1 As Integer, CNT2 As Integer, CNT3 As Integer, CNT4 As Integer
Dim MNCOST As Integer, MAXSH As Integer, MAXORD As Integer, MAXVEH As Integer

Function fMain_BuscaTABU(ScenarioADM As String) As String
'Function to allocate shipments via TABU Search
Dim DB As Database
Dim tbTabu As Recordset
Dim CountInter As Single
Dim MaxRep As Single
Dim MaxRepNStep As Single
Dim MaxInter As Single
Dim MaxInterTabu As Single
Dim MaxInterESC As Single
Dim ZSelect As Double
Dim ZBest As Double
Dim ZEscape As Single
Dim WDATA As Double, WVOL As Double

ZEscape = 0 'Counter to Max Number of Interactions before Escape
CountInter = 0 'Interaction Counter
ZSelect = 0 'Best Candidate Solution
ZBest = 0 'Best Found Solution

Set DB = CurrentDb()
'Set Tabu Logic Compartment Tables
Set tbTabu = DB.OpenRecordset("Set_Setup_TABU")
tbTabu.Index = "PrimaryKey"
tbTabu.Seek "=", "Schedule"
MaxRep = tbTabu("MaxRepetition").Value 'NOT IN USE* 9999 Max numero de repetições para tentar realocar 1 compartimento
MaxRepNStep = tbTabu("MaxRepNextStep").Value '9999 Max numero de repetições (Max Iteration for fEscapeSol_Allocate generate
MaxInter = tbTabu("MaxIteration").Value '9999 Max numero de Interações
MaxInterESC = tbTabu("MaxInterToEscape").Value '9999 Max numero de interações sem achar uma solução para criar um Escape novame
MaxInterTabu = tbTabu("MaxIterationTabu").Value '99 Max numero de Interações TABU

If fEscapeSol_Allocate(ScenarioADM, MaxRepNStep, MaxInterTabu) <> False Then
Do While CountInter <> MaxInter
'Choose and Insert BestMove (NextStep) in Calc_TABU_Compart_Movement
Call fGenSol_Allocate(1, MaxRep, MaxInterTabu)
'Solution to be evaluate (Min Cost / Vol change / Data change)
WDATA = (AllocateSol(1, 0, 0, 11) / Shipment(1, 0, 1)) * 0.05 'Increase cost 5% per shipment with Day Changed
WVOL = (AllocateSol(1, 0, 0, 11) / Shipment(1, 0, 1)) * 0.01 'Increase cost 1% per shipment with Vol Change
ZSelect = AllocateSol(1, 0, 0, 11) + WDATA * Val(Shipment(1, 0, 5)) + WVOL * Val(Shipment(1, 0, 6))
'Check Best Solution
If ZEscape <= MaxInterESC Then
If Order(1, 0, 1) > Val(Order(0, 0, 1)) Then 'Max N# Orders Scheduled
ZBest = ZSelect
Call fGetAllocation(1, 0) 'Save Solution
Call fGetShipment(1, 0)
Call fGetOrder(1, 0)
AllocateSol(0, 0, 0, 0) = CountInter
ElseIf Order(1, 0, 1) = Order(0, 0, 1) Then
If ZSelect < ZBest Then 'Min Cost / Vol change / Data change
ZBest = ZSelect
Call fGetAllocation(1, 0) 'Save Solution
Call fGetShipment(1, 0)
Call fGetOrder(1, 0)
AllocateSol(0, 0, 0, 0) = CountInter
End If
End If
ZEscape = ZEscape + 1
Else
ZEscape = 0
Call fEscapeSol_Allocate(ScenarioADM, MaxRepNStep, MaxInterTabu)
End If
'Next Interaction - Counter
CountInter = CountInter + 1
If CountInter = 110 Then 'For Check only
CountInter = CountInter
End If
Loop
Else
ZBest = 0
End If

tbTabu.Edit
If ZBest = 0 Then
tbTabu("TotalCost").Value = 0
tbTabu("BestSol").Value = 0
tbTabu("Solver_Comment").Value = "Error"
fMain_BuscaTABU = 0 'Error
Else
tbTabu("TotalCost").Value = AllocateSol(0, 0, 0, 11)
tbTabu("BestSol").Value = AllocateSol(0, 0, 0, 0)
tbTabu("Solver_Comment").Value = fSaveSol
fMain_BuscaTABU = AllocateSol(0, 0, 0, 11)
End If
tbTabu.Update

End Function

```

APÊNDICE E – ROTINA PRINCIPAL DO MÓDULO MSCHEDULE_TABU

```

Option Compare Database
'Compartment Solution:
'9999.Iteration- MaxRepetition / 0=PerCompartment & 1=Total: Value / 99.Compartment
'/ 0.MaxCapacity 1.CompID 2.Prod, 3.Customer, 4.VolDel, 5.Weight, 6.Feasible, 7.ZWeight, 8.MinVol, 9.Mult.Mark
Dim CompartSol(5000, 1, 99, 9) As String
'9999.Main Iteration - MaxIteration / 0=PerCompartment & 1=Total: Value / 99.Compartment
'/ 0.MaxCapacity 1.CompID 2.Prod, 3.Customer, 4.VolDel, 5.Weight, 6.Feasible, 7.ZWeight 8.Tabu 9.CounterTABU
Dim CompartTABU(5000, 1, 99, 9) As String
Dim CCOUNT1 As Integer, CCOUNT2 As Integer, CCOUNT3 As Integer, CCOUNT4 As Integer, CCOUNT5 As Integer, CCOUNT6 As Integer
Dim MAXCOMP As Integer
Dim MAXWEIGHT As Double

```

```

Function fCompartmentTABU(VehicleID As String, ShipmentID As String) As String
'Function to allocate the order in the vehicle compartments
Dim DB As Database
Dim tbTabu As Recordset
Dim CountInter As Single
Dim MaxRep As Single
Dim MaxRepNStep As Single
Dim MaxInter As Single
Dim MaxInterTabu As Single
Dim MaxInterESC As Single
Dim ZSelect As Double
Dim ZBest As Double
Dim ZBestCode As Single
Dim ZEscape As Single
'ReDim CompartSol(9999, 1, 99, 6) As String
'ReDim CompartTABU(9999, 1, 99, 6) As String

ZEscape = 0 'Counter to Max Number of Interactions before Escape
CountInter = 1 'Interaction Counter
ZSelect = 0 'Best Candidate Solution
ZBest = 0 'Best Found Solution
ZBestCode = 0 'Save the best CountInter Solution

Set DB = CurrentDb()
'Set Tabu Logic Compartment Tables
Set tbTabu = DB.OpenRecordset("Set_Setup_TABU")
tbTabu.Index = "PrimaryKey"
tbTabu.Seek "=", "Compartment"
MaxRep = tbTabu("MaxRepetition").Value '**NOT IN USE* 9999 Max numero de repetições para tentar realocar 1 compartimento /
MaxRepNStep = tbTabu("MaxRepNextStep").Value '9999 Max numero de repetições (Max Iteration for fEscapeSol_Compartment generate a r
MaxInter = tbTabu("MaxIteration").Value '9999 Max numero de Interações
MaxInterESC = tbTabu("MaxInterToEscape").Value '9999 Max numero de interações sem achar uma solução para criar um Escape novament
MaxInterTabu = tbTabu("MaxIterationTabu").Value '99 Max numero de Interações TABU

If fEscapeSol_Compartment(MaxRepNStep, VehicleID, ShipmentID) Then
Do While CountInter <> MaxInter
'Choose and Insert BestMove (NextStep) in Calc_TABU_Compartment_Movement
Call fGenSol_Compartment(CountInter, MaxRep, MaxInterTabu)
'Check Best Solution
'ZSelect is defined by ZWeight and Feasible var
ZSelect = CompartTABU(CountInter, 1, MAXCOMP, 7)
'Check #Repetitions
If ZEscape <= MaxInterESC Then
If ZSelect > ZBest Then
ZBest = ZSelect
ZBestCode = CountInter
If CompartTABU(CountInter, 1, MAXCOMP, 6) = "True" Then Exit Do
End If
ZEscape = ZEscape + 1
Else
ZEscape = 0
Call fEscapeSol_Compartment(MaxRepNStep, VehicleID, ShipmentID)
End If
'Next Interaction - Counter
CountInter = CountInter + 1
Loop
Else
ZBestCode = 0
End If

'Save the Best Solution found in Buffer Tables
If CompartTABU(ZBestCode, 1, MAXCOMP, 6) = "True" Then Call SaveFinalSolution(VehicleID, ShipmentID, ZBestCode)

If ZBestCode = 0 Then
fCompartmentTABU = 0 '**Error
Else
fCompartmentTABU = CompartTABU(ZBestCode, 1, MAXCOMP, 6) 'False/True
End If
'MsgBox ZBestCode, vbOKOnly, ZBestCode

End Function

```


APÊNDICE G – CUSTO DIÁRIO POR CENÁRIO

Custo (kU\$)									
Data	-	cenário 1	cenário 2	cenário 3	cenário 4	cenário 5	cenário 6	cenário 7	cenário 8
11/12/2011	2,8	2,7	2,5	2,7	2,7	2,6	2,7	2,6	2,7
12/12/2011	4,6	4,5	4,3	4,8	4,5	4,5	4,8	4,4	4,6
13/12/2011	7,5	7,3	7,1	7,3	7,3	7,2	7,3	7,2	7,2
14/12/2011	7,5	7,3	7,1	7,3	7,4	7,2	7,4	7,2	7,3
15/12/2011	6,5	6,3	6,1	6,3	6,4	6,2	6,4	6,2	6,3
16/12/2011	5,4	5,4	5,1	5,6	5,4	5,3	5,6	5,2	5,4
17/12/2011	6,3	6,2	5,9	6,2	6,2	6,0	6,2	6,0	6,1
18/12/2011	3,6	3,6	3,4	3,6	3,6	3,5	3,6	3,5	3,5
19/12/2011	6,0	5,9	5,6	6,1	5,9	5,8	6,1	5,7	5,9
20/12/2011	7,2	7,1	6,8	7,3	7,1	7,1	7,3	7,0	7,1
21/12/2011	6,9	6,7	6,5	6,7	6,7	6,6	6,7	6,6	6,6
22/12/2011	4,9	4,9	4,6	5,1	4,9	4,8	5,1	4,7	4,9
23/12/2011	8,1	8,0	7,7	8,0	8,0	7,8	8,0	7,8	7,9
24/12/2011	5,9	5,8	5,6	5,8	5,8	5,7	5,8	5,7	5,8
Total:	83,26	81,77	78,13	82,88	81,87	80,43	82,98	79,93	81,19

APÊNDICE I – NÚMERO DE ALTERAÇÕES DE PEDIDOS DIÁRIOS POR CENÁRIO

Número de pedidos pendentes

Data	-	cenário 1	cenário 2	cenário 3	cenário 4	cenário 5	cenário 6	cenário 7	cenário 8
11/12/2011	0	0	0	0	0	0	0	0	0
12/12/2011	0	0	0	0	0	0	0	0	0
13/12/2011	0	0	2	0	0	2	0	2	2
14/12/2011	0	0	1	0	0	1	0	1	1
15/12/2011	0	0	1	0	0	1	0	1	1
16/12/2011	0	0	0	0	0	0	0	0	0
17/12/2011	0	0	1	0	0	1	0	1	1
18/12/2011	0	0	0	0	0	0	0	0	0
19/12/2011	0	0	0	0	0	0	0	0	0
20/12/2011	0	0	1	0	0	2	0	1	2
21/12/2011	0	0	3	0	0	3	0	3	3
22/12/2011	0	0	0	0	0	0	0	0	0
23/12/2011	0	0	1	0	0	1	0	1	1
24/12/2011	0	0	0	0	0	0	0	0	0
Total:	0	0	10	0	0	11	0	10	11

Número de pedidos adiantados

Data	-	cenário 1	cenário 2	cenário 3	cenário 4	cenário 5	cenário 6	cenário 7	cenário 8
11/12/2011	0	0	0	0	0	0	0	0	0
12/12/2011	0	0	2	0	0	2	0	2	2
13/12/2011	0	0	1	0	0	1	0	1	1
14/12/2011	0	0	1	0	0	1	0	1	1
15/12/2011	0	0	0	0	0	0	0	0	0
16/12/2011	0	0	0	0	0	0	0	0	0
17/12/2011	0	0	0	0	0	0	0	0	0
18/12/2011	0	0	0	0	0	0	0	0	0
19/12/2011	0	0	2	0	0	2	0	2	2
20/12/2011	0	0	0	0	0	0	0	0	0
21/12/2011	0	0	0	0	0	0	0	0	0
22/12/2011	0	0	0	0	0	0	0	0	0
23/12/2011	0	0	2	0	0	2	0	2	2
24/12/2011	0	0	0	0	0	0	0	0	0
Total:	0	0	8	0	0	8	0	8	8

APÊNDICE J – INDISPONIBILIDADE DIÁRIA DE VEÍCULOS POR CENÁRIO

Número de veículos indisponíveis

Data	-	cenário 1	cenário 2	cenário 3	cenário 4	cenário 5	cenário 6	cenário 7	cenário 8
11/12/2011	0	0	0	0	0	0	0	0	0
12/12/2011	0	0	0	1	0	1	1	0	1
13/12/2011	0	0	0	0	0	0	0	0	0
14/12/2011	0	0	0	0	0	0	0	0	0
15/12/2011	0	0	0	0	0	0	0	0	0
16/12/2011	0	0	0	1	0	1	1	0	1
17/12/2011	0	0	0	0	0	0	0	0	0
18/12/2011	0	0	0	0	0	0	0	0	0
19/12/2011	0	0	0	1	0	2	2	0	2
20/12/2011	0	0	0	1	0	1	1	0	1
21/12/2011	0	0	0	0	0	0	0	0	0
22/12/2011	0	0	0	1	0	1	0	0	1
23/12/2011	0	0	0	0	0	0	0	0	0
24/12/2011	0	0	0	0	0	0	0	0	0
Total:	0	0	0	5	0	6	5	0	6

