



UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO  
CENTRO DE CIÊNCIAS AGRÁRIAS E ENGENHARIAS  
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIAS FLORESTAIS

**ANTONIO ALMEIDA DE BARROS JUNIOR**

**MODELAGEM DO PROBLEMA DE ROTEAMENTO NO PLANEJAMENTO DO  
INVENTÁRIO FLORESTAL**

JERÔNIMO MONTEIRO – ES  
2017

ANTONIO ALMEIDA DE BARROS JUNIOR

**MODELAGEM DO PROBLEMA DE ROTEAMENTO NO PLANEJAMENTO DO  
INVENTÁRIO FLORESTAL**

Tese apresentada ao Programa de Pós-Graduação em Ciências Florestais do Centro de Ciências Agrárias e Engenharias da Universidade Federal do Espírito Santo, como parte das exigências para obtenção do Título de Doutor em Ciências Florestais na Área de Concentração Ciências Florestais.  
Orientador: Dr. Gilson Fernandes da Silva  
Coorientador: Dr. Geraldo Regis Mauri

JERÔNIMO MONTEIRO – ES  
2017

Dados Internacionais de Catalogação-na-publicação (CIP)  
(Biblioteca Setorial de Ciências Agrárias, Universidade Federal do Espírito Santo, ES, Brasil)  
Bibliotecário: Felício Gomes Corteletti – CRB-6 ES-000646/O

---

B268m Barros Junior, Antonio Almeida de, 1978-  
Modelagem do problema de roteamento no planejamento do inventário florestal natural / Barros Junior, Antonio Almeida de. – 2017. 87 f. : il.

Orientador: Gilson Fernandes da Silva.

Coorientador: Geraldo Regis Mauri.

Tese (Doutorado em Ciências Florestais) – Universidade Federal do Espírito Santo, Centro de Ciências Agrárias e Engenharias.

1. Levantamentos de recursos naturais . 2. Programação linear. 3. Problema de roteamento de veículos. 4. Heurística. I. Silva, Gilson Fernandes da. II. Mauri, Geraldo Regis. III. Universidade Federal do Espírito Santo. Centro de Ciências Agrárias e Engenharias. IV. Título.

CDU: 630

---

MODELAGEM DO PROBLEMA DE ROTEAMENTO NO PLANEJAMENTO DO  
INVENTÁRIO FLORESTAL

Antonio Almeida de Barros Junior

Tese apresentada ao Programa de Pós-Graduação em Ciências Florestais do Centro de Ciências Agrárias e Engenharias da Universidade Federal do Espírito Santo, como parte das exigências para obtenção do Título de Doutor em Ciências Florestais na Área de Concentração Recursos Florestais.  
Orientador: Dr. Gilson Fernandes da Silva  
Coorientador: Dr. Geraldo Regis Mauri

Aprovada em 11 de setembro de 2017.



---

**Prof. Dr. Carlos Pedro Boechat Soares** (Examinador externo)  
Universidade Federal de Viçosa



---

**Prof. Dr. Helio Garcia Leite** (Examinador externo)  
Universidade Federal de Viçosa



---

**Adriano Ribeiro de Mendonça** (Examinador interno)  
Universidade Federal do Espírito Santo



---

**Geraldo Regis Mauri** (Coorientador)  
Universidade Federal do Espírito Santo



---

**Gilson Fernandes da Silva** (Orientador)  
Universidade Federal do Espírito Santo

*A Deus e aos meus pais Antonio e Annazilda.*  
**OFEREÇO**

*À minha esposa Michelha e a minha filha Estela. Amo vocês!*  
**DEDICO**

**"A fé e a razão caminham juntas, mas a fé vai mais longe."**  
Santo Agostinho

## AGRADECIMENTOS

Agradeço a Deus, que por intercessão de Nossa Senhora da Conceição Aparecida me concede forças para enfrentar as barreiras da vida e seguir em frente.

Aos meus pais Antonio e Annazilda, pela simplicidade, honestidade e exemplos de superação e amor. À minha tia Emília (*in memoriam*). Aos meus irmãos Maria José, Marco Antonio e Marconi, por serem presentes na minha vida e por terem me concedido, junto de seus cônjuges, sobrinhos que tanto amo: Nathália, Anna Laura, Esther, Arthur e Amanda. É um orgulho fazer parte desta família!

Ao meu orientador, Gilson Fernandes da Silva, pela oportunidade, ensinamentos e amizade. Seu profissionalismo é um exemplo a ser seguido e respeitado. Também agradeço ao meu coorientador Geraldo Regis Mauri, pela paciência, disponibilidade e auxílio com os métodos. Sem sua colaboração esta pesquisa não teria sido realizada.

Aos professores da Banca Examinadora, Helio Garcia Leite, Carlos Pedro Boechat Soares e Adriano Ribeiro de Mendonça, pelas contribuições fundamentais para a melhoria deste trabalho. Não poderia deixar de agradecer à Professora Mayra Luiza Marques da Silva e a Daniel Henrique Breda Binoti, pelas contribuições no exame de qualificação.

Aos meus colegas do Departamento de Computação pelo apoio, especialmente aos amigos Clayton e Helder, por terem embarcado junto comigo no desafio de cursar doutorado nas Ciências Florestais.

À Universidade Federal do Espírito Santo, ao Departamento de Ciências Florestais e da Madeira e ao Programa de Pós-graduação em Ciências Florestais, pela oportunidade. A todos os colegas do LaMFlor, pela amizade e apoio durante este período. Em especial aos doutorandos Giovanni, Jeangelis e Evandro.

Agradeço à empresa Fibria, nos nomes de Cristiane e Matheus, pela concessão dos dados e orientações a respeito do problema.

Agradeço a minha bela esposa Michelha, pelo companheirismo, pelo amor e por ter segurado as pontas em casa para que eu pudesse trabalhar nesta tese. Agradeço a minha filha Estela, a mais pura demonstração de amor e carinho. Todo o tempo dispensado neste trabalho, é para lhes proporcionar um futuro melhor. AMO MUITO vocês duas!

## **BIOGRAFIA**

**Antonio Almeida de Barros Junior**, filho de Antonio Almeida de Barros e Anazilda Pedrosa de Barros, brasileiro, nascido em 17 de março de 1978 na cidade de Carangola-MG. Casado com Michelha Vaz Pedrosa, e pai de Estela Pedrosa de Barros. Concluiu o ensino médio na Escola Estadual Santo Agostinho na cidade de São Francisco do Glória em 1995, ingressou no curso de Processamento de Dados na Escola Agrotécnica Federal de Barbacena em 1997, concluindo em 1999. Em agosto de 2000, ingressou no curso de Sistemas de Informação na Universidade Iguazu, concluindo o curso de graduação em 2004. Em março de 2007 iniciou o mestrado em Ciência da Computação na Universidade Federal de Viçosa (UFV), defendendo sua dissertação em fevereiro de 2010. Entre os anos 2000 e 2009, trabalhou como programador e analista de sistemas. No ano de 2006, iniciou a atividade de docência na Faculdade Santa Marcelina. Em 2010 foi aprovado no concurso para professor efetivo do Departamento de Computação do Centro de Ciências Agrárias da Universidade Federal do Espírito Santo onde iniciou suas atividades. Em março de 2014, iniciou o doutorado no Programa de Pós-graduação em Ciências Florestais da mesma universidade, defendendo sua tese em setembro de 2017.

## RESUMO

BARROS JUNIOR, Antonio Almeida de. **MODELAGEM DO PROBLEMA DE ROTEAMENTO NO PLANEJAMENTO DO INVENTÁRIO FLORESTAL**. 2017. Tese (Doutorado em Ciências Florestais) – Universidade Federal do Espírito Santo, Jerônimo Monteiro, ES. Orientador: Prof. Dr. Gilson Fernandes da Silva. Coorientador: Prof. Dr. Geraldo Regis Mauri.

Dentre as diversas atividades ligadas à produção da madeira, a atividade de inventário florestal se destaca por promover a coleta de dados para análise e tomada de decisão. Em empreendimentos florestais com grandes áreas de plantio, o número de parcelas instaladas torna-se muito grande, fazendo com que seja necessário um prévio planejamento e programação das atividades de inventário. O planejamento consiste em definir quais parcelas serão inventariadas em qual período do tempo. Neste contexto, esta tese apresenta uma nova abordagem para representar e obter soluções para o Problema de Roteamento de Veículos no planejamento de inventários florestais (PRV-IF). Para tal, foi proposto um modelo adaptado do Problema de Roteamento Periódico de Veículos com Janela de Tempo (PRPVJT) para o problema. Os métodos de solução utilizados foram o método exato *branch-and-cut* por meio do *solver* CPLEX e as meta-heurísticas *Iterated Local Search* (ILS) e *Simulated Annealing* (SA). Com a finalidade de analisar o desempenho dos métodos, foram criadas instâncias para o PRV-IF de diferentes tamanhos. Os resultados obtidos demonstram que os métodos foram satisfatórios na solução do problema, especialmente nas maiores instâncias. A meta-heurística *Simulated Annealing* obteve os melhores resultados quando comparada com os demais métodos.

**Palavras-chave:** Inventário florestal, Modelo de roteamento periódico de veículos com janela de tempo, Programação Linear.

## ABSTRACT

BARROS JUNIOR, Antonio Almeida de. **MODELING THE ROUTING PROBLEM FOR PLANNING THE FOREST INVENTORY**. 2017. Thesis (Doctor in Forest Sciences) – Federal University of Espírito Santo, Jerônimo Monteiro, ES. Advisor: Prof. Dr. Gilson Fernandes da Silva. Co-advisor: Prof. Dr. Geraldo Regis Mauri.

Among the various activities related to timber production, the forest inventory activity stands out for promoting the collection of data for analysis and decision making. In forest projects with large planting areas, the number of installed plots becomes very large, requiring prior planning and inventory activities programming. The planning consists of defining which plots will be inventoried in what period of time. In this context, this thesis presents a new approach to represent and obtain solutions for the Vehicle Routing Problem in the planning of forest inventories (PRV-IF). An adapted model of the Periodic Vehicle Routing Problem with Time Windows (PRPVJT) was proposed for the problem. The exact *branch-and-cut method* using the CPLEX solver and the *Iterated Local Search* (ILS) and *Simulated Annealing* (SA) metaheuristics were used as solution methods. The performance of the applied methods was analyzed by creating instances for the PRV-IF with different sizes. The results show that the methods were satisfactory in solving the problem, especially for the larger instances. The *Simulated Annealing* metaheuristic obtained the best results when compared to the other methods.

**Keywords:** Forest inventory, Periodic vehicle routing problem with time window, Linear Programming.

## LISTA DE TABELAS

Tabela 1. Flexibilidade de data para medição das atividades de inventário florestal	41
Tabela 2. Tipos de inventários considerados .....	46
Tabela 3. Períodos considerados para definição da Janela de Tempo.....	46
Tabela 4. Instâncias para o Problema de Roteamento de Veículos no Planejamento de Inventários Florestais .....	65
Tabela 5. Valores obtidos para os parâmetros dos métodos ILS e SA .....	66
Tabela 6. Resultados obtidos pelos métodos CPLEX, ILS e SA para as instâncias do PRV-IF .....	69
Tabela 7. Menor tempo em minutos para encontrar a melhor solução por método ..	71
Tabela 8. Resumo mensal dos valores obtidos pela meta-heurística SA para a instância PRV-IF_2016_252.....	77

## LISTA DE FIGURAS

Figura 1. Meta-heurística <i>Iterated Local Search</i> .....	32
Figura 2. Representação gráfica da meta-heurística ILS .....	32
Figura 3. Meta-heurística <i>Simulated Annealing</i> .....	34
Figura 4. Fluxograma da ferramenta desenvolvida por Marcatti (2013). (a) Implementação da metodologia de otimização do acesso a parcela. (b) Implementação da metodologia de otimização do caminhamento .....	35
Figura 5. Representação das variáveis envolvidas no PRV-IF.....	40
Figura 6. Representação de uma rota diária estabelecida para uma equipe .....	41
Figura 7. Diagrama metodológico para geração de rotas para atividades de inventário florestal .....	43
Figura 8. Mapa de localização da região estudada .....	44
Figura 9. Exemplo de arquivo de instância para um problema com 24 pontos e 6 dias de planejamento.....	48
Figura 10. Representação de uma solução.....	52
Figura 11. Pseudocódigo da construção da solução inicial.....	53
Figura 12. Movimento 1-Optimal intra-rota.....	54
Figura 13. Movimento Melhor Melhora.....	55
Figura 14. Movimento de remoção intra-rota.....	56
Figura 15. Movimento de troca intra-rota .....	56
Figura 16. Movimento de remoção entre rotas.....	57
Figura 17. Movimento de troca entre rotas.....	58
Figura 18. Movimento <i>shift</i> (2,0).....	58
Figura 19. Pseudocódigo do movimento de Correção .....	59
Figura 20. Movimento Correção .....	60
Figura 21. Pseudocódigo do movimento Equilíbrio .....	62
Figura 22. ILS para o PRV-IF .....	62

Figura 23. SA para o PRV-IF.....	63
Figura 24. Distância percentual entre meta-heurísticas e limitante para cada instância .....	70
Figura 25. Melhor resultado encontrado pela meta-heurística SA para os 3 primeiros dias da instância PRV-IF_2016_252 <sup>1</sup> .....	75
Figura 26. Melhor resultado encontrado pela meta-heurística ILS para os 3 primeiros dias da instância PRV-IF_2016_252 <sup>2</sup> .....	76

## LISTA DE ABREVIATURAS E SIGLAS

IFC - Inventário Florestal Contínuo

ILS - *Iterated Local Search*

JT - Janela de Tempo

JTD - Janela de Tempo Diária

PCC - Problema do Carteiro Chinês

PCV - Problema do Caixeiro Viajante

PI - Programação Inteira

PL - Programação Linear

PLI - Programação Linear Inteira

PLIM - Programação Linear Inteira Mista

PNL - Programação Não-linear

PRV-IF - Problema de Roteamento de Veículos no Planejamento de Inventários Florestais

PRPV - Problema de Roteamento Periódico de Veículos

PRPVJT - Problema de Roteamento Periódico de Veículos com Janela de Tempo

PRV - Problema de Roteamento de Veículos

PRVC - Problema de Roteamento de Veículos Capacitado

PRVJT - Problema de Roteamento de Veículos com Janela de Tempo

SA - *Simulated Annealing*

TTD - Tempo Total de Deslocamento

TTS - Tempo Total de Serviço

## SUMÁRIO

1. INTRODUÇÃO.....	16
1.1. Objetivos.....	17
1.1.1. Objetivo geral.....	17
1.1.2. Objetivos específicos.....	18
1.2. Estrutura da Tese.....	18
2. REVISÃO DE LITERATURA.....	19
2.1. Inventário Florestal.....	19
2.2. Programação Linear.....	20
2.3. Problema de Roteamento de Veículos.....	21
2.3.1. Problema de Roteamento de Veículos Capacitado - PRVC.....	25
2.3.2. Problema de Roteamento Periódico de Veículos - PRPV.....	26
2.3.3. Problema de Roteamento de Veículos com Janela de Tempo - PRVJT...28	
2.4. Meta-heurísticas como possibilidade de solução para problemas complexos.....	30
2.4.1. <i>Iterated Local Search</i> - ILS.....	31
2.4.2. <i>Simulated Annealing</i> - SA.....	33
2.5. Trabalhos relacionados ao planejamento de inventários florestais.....	34
2.5.1. Metodologia proposta por Marcatti (2013).....	34
2.5.2. Metodologia proposta por Haddad (2015).....	36
2.5.3. Metodologia proposta por Meneguzzi (2011).....	37
3. PLANEJAMENTO DE INVENTÁRIOS FLORESTAIS.....	39
4. MATERIAL E MÉTODOS.....	43
4.1. Base de Dados.....	44
4.2. Tratamento dos dados.....	45
4.3. Geração dos arquivos básicos.....	47
4.4. Otimização.....	49
4.4.1. Formulação do Modelo Matemático.....	49
4.4.2. Métodos de solução.....	52
4.4.2.1. Representação de uma solução.....	52
4.4.2.2. Construção da solução inicial.....	53
4.4.2.3. Estratégias de busca local.....	54
4.4.2.4. Estratégias de perturbação.....	55
4.4.2.5. <i>Iterated Local Search</i> para resolução do PRV-IF.....	62
4.4.2.6. <i>Simulated Annealing</i> para resolução do PRV-IF.....	63

5.	EXPERIMENTOS COMPUTACIONAIS.....	65
5.1.	Descrição das instâncias .....	65
5.2.	Parâmetros .....	66
6.	RESULTADOS E DISCUSSÃO .....	67
6.1.	Comparação dos métodos.....	67
6.2.	Ferramentas desenvolvidas .....	71
6.3.	Estudo de caso .....	72
7.	CONCLUSÕES E TRABALHOS FUTUROS .....	78
7.1.	Trabalhos futuros .....	78
8.	REFERÊNCIAS BIBLIOGRÁFICAS .....	80

## 1. INTRODUÇÃO

O planejamento florestal indica metas de produção para períodos de tempo definidos, considerando previsões do mercado (preço e demanda), estoque e recursos disponíveis (GOMIDE, 2009). De forma hierárquica, está organizado em níveis: estratégico (longo prazo), tático (médio prazo) e operacional (curto prazo) (RODRIGUES, 2001).

Dentre as atividades que fazem parte do planejamento tático e operacional de uma empresa florestal, o inventário se destaca por ser uma importante ferramenta para o manejo e o planejamento da produção, o qual pode ser realizado sob diferentes níveis de detalhamento e em diferentes pontos no tempo (SOARES et al., 2011). Equívocos nesse tipo de planejamento podem levar a grandes perdas. Neste sentido, essas empresas realizam seus inventários periodicamente para, entre outras coisas, construir modelos de crescimento e produção, realizar pesquisas e planejar a exploração.

Uma parte significativa do custo total de um inventário está relacionada à coleta de dados (CAMPOS e LEITE, 2017). A importância e os custos que envolvem a atividade de inventário florestal faz com que sua realização seja previamente planejada. Em empresas florestais de grande porte, existe uma dificuldade em realizar o planejamento desta atividade, dada a quantidade de parcelas instaladas em seus povoamentos. Para isso, realiza-se um planejamento anual, no qual são definidos quais povoamentos florestais (talhões) serão inventariados, e em quais períodos do ano.

Para Meneguzzi (2011), entre as dificuldades encontradas na realização do planejamento do inventário florestal estão: a limitação do número de equipes, o tempo de medição de cada parcela, o tempo de deslocamento para as unidades amostrais e a distância percorrida. Uma alternativa para auxiliar a redução de custos desta atividade, consiste em otimizar rotas e, conseqüentemente, o tempo de deslocamento das equipes de inventário entre os povoamentos florestais.

A otimização de rotas ou roteamento de veículos é um caso particular de programação linear (PL). Para Arenales et al. (2015), o Problema de Roteamento de Veículos (PRV) exerce uma função preponderante na área de gerenciamento da distribuição e logística. O PRV pode ser aplicado em situações práticas que ocorrem em indústrias, no comércio, no setor de serviços, na segurança, na saúde pública e no lazer.

Goldbarg e Luna (2005) destacam que os problemas de roteamento de veículos estão entre os mais complexos da área de otimização combinatória. Lenstra e Rinnooy Kan (1981) enquadraram o PRV na classe dos problemas NP-Difíceis. A solução de um problema NP-Difícil exigirá, em última análise, consumo exponencial de processamento em tempo ou em memória em função do tamanho da entrada de dados desses problemas (GOLDBARG et al., 2015). Em problemas de grande porte, onde há um número significativo de variáveis, nem sempre é possível a solução por métodos exatos.

A insuficiência da resposta dos métodos exatos e a alta complexidade dos problemas de roteamento de veículos faz dos procedimentos aproximativos (heurísticos) uma considerável alternativa quando é preciso encontrar soluções boas em um tempo limitado (GOLDBARG et al., 2015; VIEIRA, 2013). No contexto computacional, uma heurística é uma técnica capaz de encontrar uma solução aceitável para um dado problema utilizando um esforço computacional razoável.

Para direcionar o processo de construção de heurísticas, surgiu uma classe de estratégias denominadas meta-heurísticas. De acordo com Osman e Laporte (1996), meta-heurística pode ser compreendida como um processo iterativo que guia heurísticas subordinadas, combinando diferentes e inteligentes conceitos para explorar o espaço de busca. Para Cordeau et al. (2002), as meta-heurísticas executam uma busca muito mais ampla no espaço de solução quando comparadas com heurísticas clássicas.

Face ao exposto, esta pesquisa apresenta uma metodologia capaz de resolver um Problema de Roteamento de Veículos no Planejamento de Inventários Florestais (PRV-IF), de forma que o custo total da atividade seja minimizado. Para isso, são apresentados resultados obtidos por uso de dois métodos adaptados de meta-heurísticas clássicas da literatura: *Iterated Local Search* (ILS) e *Simulated Annealing* (SA).

## **1.1. Objetivos**

### **1.1.1. Objetivo geral**

O objetivo geral deste trabalho foi propor uma nova abordagem para resolução do Problema de Roteamento de Veículos no Planejamento de Inventários Florestais (PRV-IF).

### 1.1.2. Objetivos específicos

- Propor e validar um modelo de otimização para Problema de Roteamento de Veículos aplicado ao Planejamento de Inventários Florestais.
- Desenvolver um sistema informatizado capaz de gerar cenários (instâncias) para o problema proposto.
- Desenvolver um sistema de otimização capaz de encontrar soluções para a programação anual das atividades de inventário florestal, por meio do uso das meta-heurísticas ILS e SA.

### 1.2. Estrutura da Tese

No Capítulo 1 são descritas as principais motivações e objetivos do trabalho realizado e a organização do texto desta tese.

O Capítulo 2 expõe os principais fundamentos necessários à compreensão das abordagens apresentadas nos capítulos seguintes. Nesse capítulo, é exposta uma ideia geral sobre inventário florestal e programação linear. Também são apresentados alguns fundamentos sobre problemas de roteamento de veículos e meta-heurísticas e alguns trabalhos relacionados ao tema.

No Capítulo 3 é abordado o Problema de Roteamento de Veículos no Planejamento de Inventários Florestais (PRV-IF) de forma detalhada. São apresentadas as principais restrições do problema e suas implicações.

O Capítulo 4 apresenta a metodologia proposta nesta pesquisa subdividida nas seguintes etapas: base de dados, tratamento dos dados, geração de arquivos básicos e otimização. Estas etapas representam fases do desenvolvimento que vão desde a concessão dos dados à geração dos resultados por meio das meta-heurísticas *Iterated Local Search* e *Simulated Annealing*.

No Capítulo 5 são descritos detalhes dos experimentos computacionais, tais como: parâmetros e instâncias. No Capítulo 6 são descritos os resultados obtidos com os métodos propostos. É realizada uma breve descrição das ferramentas desenvolvidas e encerra-se relacionando os resultados com as necessidades de uma empresa do setor florestal. Por fim, o Capítulo 7 sumariza as conclusões e contribuições da pesquisa apresentada nesta tese, seguidas por algumas sugestões para sua continuação.

## 2. REVISÃO DE LITERATURA

Este capítulo apresenta uma breve descrição dos fundamentos necessários para a compreensão das abordagens e metodologia adotadas nesta pesquisa.

### 2.1. Inventário Florestal

O inventário florestal é uma atividade na qual, a partir da mensuração de árvores, associada a um conjunto de técnicas estatísticas de amostragem, obtêm-se informações qualitativas e quantitativas a respeito dos povoamentos (HUSCH et al., 2003; SCOLFORO; MELO, 2006; KANGAS; MALTAMO, 2006). Qualquer tomada de decisão envolvendo recursos florestais, como a avaliação de reflorestamento ou implantação de novas áreas, intervenções no manejo silvicultural, colheita da floresta, controle de incêndios, entre vários outros, precisa ser subsidiada com informações provenientes de inventários (SANQUETTA et al., 2009).

O principal objetivo desta atividade é estimar as características qualitativas e quantitativas de árvores individuais e dos povoamentos, como o estoque de madeira e sua distribuição entre as diversas espécies ou classes de diâmetro (AVERY; BURKHART, 1983; KANGAS; MANTANO, 2006).

Para Kangas e Maltamo (2006), um inventário poderia, em princípio, ser baseado em um censo, ou seja, avaliação de cada árvore em uma determinada área. Porém, realizar o censo torna-se impossível, dado as grandes áreas envolvidas. Desta forma, a aquisição de informações é tipicamente baseada na amostragem, em que apenas uma parte da população florestal (amostra) é avaliada. Assim, as inferências relativas de toda a população são baseadas nestas amostras.

Meneguzzi (2011) destaca que a atividade de inventário pode ser dividida no mapeamento da área, amostragem (fase de coleta de dados) e processamento, análise dos dados e geração de relatórios. O sucesso de um inventário se baseia, entre outras coisas, em um mapeamento acurado da área a ser amostrada. É de suma importância o conhecimento da área de cada compartimento, pois toda informação obtida com a amostra será extrapolada para todo o povoamento florestal. Uma informação diferente da realidade irá gerar sub ou superestimativa do estoque madeireiro. Outras informações importantes são as características do plantio, a idade, o genótipo plantado, o regime de manejo utilizado (alto fuste ou talhadia).

Uma classificação dos tipos de inventários (SOARES et al., 2011) é:

- a. inventário de pré-corte: realizado antes da exploração, com alta intensidade amostral, utilizando parcelas temporárias;
- b. inventário florestal convencional: realizado para obtenção do estoque de volume de madeira, utilizando parcelas temporárias;
- c. inventário florestal contínuo: realizado com o objetivo de verificar as mudanças ocorridas em um povoamento em determinado período de tempo, utilizando parcelas permanentes;
- d. inventário para planos de manejo: realizado com alto grau de detalhamento, chegando às estimativas por classe de diâmetro, por espécie.
- e. inventário de sobrevivência: realizado após o plantio, com o objetivo de verificar o percentual de falhas/sobrevivência das mudas no campo.

No Brasil, o Inventário Florestal Contínuo (IFC) é a atividade mais utilizada para acompanhar o crescimento da floresta. Nesse tipo de inventário, são utilizadas parcelas permanentes, instaladas quando os povoamentos atingem dois ou três anos e remeidas em períodos que variam de um a dois anos (KANEGAE JÚNIOR et al., 2006; OLIVEIRA, 2006; SCOLFORO; MELLO, 2006).

Em resumo, o IFC pode ser definido como um conjunto de unidades de amostra permanente estabelecido numa população florestal, isto é, todas as unidades amostrais instaladas e medidas no primeiro inventário são remeidas em todas as demais medições sucessivas (LOETSCH; HALLER, 1964). Esse tipo de inventário proporciona a geração de uma base de dados para construção de tabelas de produção e modelos de crescimento, que permitem fazer a prognose do crescimento e da produção.

## **2.2. Programação Linear**

Um modelo matemático é uma representação da realidade, que permite obter conclusões por meio de sua análise e extrapolar a realidade. Denomina-se Programação Matemática, os principais modelos de Pesquisa Operacional (PO), que são direcionados ao apoio da tomada de decisão no gerenciamento de sistemas de grande porte. O processo de modelagem matemática, em si, pouco varia; contudo, as técnicas de solução acabaram agrupadas em várias subáreas como: Programação Linear (PL), Programação Não-linear (PNL) e Programação Linear Inteira (PLI) (GOLDBARG; LUNA, 2005).

Andrade (2015) define PL como uma ferramenta utilizada para resolver problemas de otimização, atribuindo e distribuindo (planejando) os recursos da

melhor forma possível entre as tarefas a serem realizadas. Para Colin (2007), a PL tem como objetivo alocar de forma ótima esses recursos finitos.

De acordo com Bazaraa et al. (2010), um problema de PL consiste em minimizar ou maximizar uma função linear considerando um conjunto de restrições formado por equações e, ou, inequações lineares. Para Wolsey (1998), caso as variáveis de decisão do problema abordado sejam estritamente inteiras, a Programação Linear é chamada de Programação Linear Inteira (PLI), ou simplesmente Programação Inteira (PI). Portanto, a PI é um caso particular da PL e, quando apenas parte das variáveis decisórias são inteiras, denomina-se Problema de Programação Linear Inteira Mista (PLIM) (DANTZIG, 1963).

A PL tornou-se uma das técnicas mais utilizadas dentro da PO por se tratar de uma ferramenta padrão e muito difundida no meio da computação científica quando envolve o uso de computadores (HILLIER; LIEBERMAN, 2012). De acordo com Silva et al. (2017), o sucesso da PL deve-se principalmente pela simplicidade do modelo utilizado e pela facilidade de solução considerando técnicas programáveis em computador. Destaca-se, como exemplo de método de solução, o algoritmo Simplex proposto por G. Dantzig, em 1947, que é considerado o primeiro algoritmo eficiente na resolução de problemas de PL.

Com o desenvolvimento deste e outros algoritmos, a PL passou a ser capaz de resolver problemas cada vez maiores e mais complexos. Para Luenberger e Ye (2008), a ampla e contínua literatura sobre a PL mostra sua relevância para a formulação de diversos tipos de problemas do mundo real. Entre os tipos de problemas abordados pela PL, podem ser citados os problemas da dieta, da mochila, de produção e alocação de recursos, programação de tarefas e roteamento de veículos.

### **2.3. Problema de Roteamento de Veículos**

Um Sistema de Roteamento pode ser considerado com um conjunto organizado de meios que objetiva o atendimento de demandas localizadas nos arcos ou nos vértices (nós) de alguma rede de transportes (GOLDBARG e LUNA, 2005). De acordo com Bodin et al. (1983), a roteirização é, em primeiro lugar, um problema de ordem espacial, cujo objetivo é minimizar a distância percorrida. Contudo, em situações reais, o problema pode estar sujeito a aspectos espaciais e temporais, o que faz com que haja um tempo associado a cada atividade. Por isso, algumas vezes o problema também é referido como Roteirização e Programação de Veículos

(*Routing and Scheduling of Vehicles* - PRV). A origem deste problema está associada ao trabalho desenvolvido por Dantzig e Ramser (1959), derivado do clássico Problema do Caixeiro Viajante (*Traveling Salesman Problem* – PCV). Desde então, o PRV tem sido um dos problemas mais abordados nas áreas de Otimização Combinatória e Pesquisa Operacional. Isso se deve em parte ao grande desenvolvimento de métodos de resolução e da variedade de aplicações existentes.

Existem dois grupos que são tratados dentro da classe de problemas de roteamento: o problema de roteirização sem restrições, denominado Problema do Caixeiro Viajante, e o problema de roteirização com restrições, chamado de Problema de Roteamento de Veículos, no qual pode haver um ou mais depósitos e vários veículos de capacidade limitada (NOVAES, 1989).

O PCV é um dos mais tradicionais e conhecidos problemas de programação matemática (GOLDENBERG e LUNA, 2005). Trata-se de um problema que tenta determinar a menor rota para percorrer uma série de cidades (visitando uma única vez cada uma delas, retornando à cidade de origem).

O PRV consiste em projetar rotas de veículos de forma a minimizar o custo total, cada uma começando e terminando na garagem garantindo que cada local seja visitado exatamente uma vez e a demanda total de qualquer rota não exceda a capacidade do veículo que a atende (LAPORTE et al., 2000). De forma objetiva, consiste na determinação de rotas otimizadas e que serão usadas por uma frota de veículos situados em uma ou mais garagens, para atender a um conjunto de locais.

Essa categoria de problema é classificada como NP-Difícil, que são problemas que não possuem uma solução de fácil obtenção. A maioria dos pesquisadores da área de Ciência da Computação acredita que problemas desta categoria sejam intratáveis devido à ampla faixa de problemas estudados até hoje. Dentre as dificuldades encontradas no PRV, a geração de subciclos se destaca por necessitar de um expressivo conjunto de restrições para impedir que estes sejam gerados.

Para Goldberg e Luna (2005), os problemas de roteamento de veículos estão entre os mais complexos da área de otimização combinatória. Embora seja um dos mais estudados na literatura, é uma categoria de problema que apresenta um grande número de variações. Dado o grande número de variáveis, a diversidade de restrições e os objetivos apresentados, impõem-se sob este tipo de problema o exame de uma cuidadosa taxonomia para seu melhor entendimento.

Neste contexto, diversos trabalhos da literatura propõem taxonomias com o objetivo de classificar o PRV. Uma das taxonomias clássicas é a taxonomia proposta por Bodin e Golden (1981), cujos critérios são apresentados no Quadro 1.

Quadro 1. Taxonomia para o PRV segundo Bodin e Golden (1981)

<b>Critério</b>	<b>Especificações</b>
1. Tempo para atender um determinado nó ou arco	1.1 Tempo especificado e prefixado
	1.2 Janela de Tempo
2. Número de domicílios	2.1 Um domicílio
	2.2 Mais de um domicílio
3. Tamanho da frota	3.1 Um veículo
	3.2 Mais de um veículo
4. Tipo de frota disponível	4.1 Homogênea
	4.2 Heterogênea
5. Natureza da demanda e parâmetros	5.1 Determinística
	5.2 Estocástica
6. Localização da demanda	6.1 Nos vértices
	6.2 Nos arcos
7. Grafo de substrato	7.1 Direcionado
	7.2 Não direcionado
	7.3 Misto
8. Restrições na capacidade de veículos	8.1 Todos sujeitos às mesmas restrições
	8.2 Restrições diferentes
9. Tempo de roteamento	9.1 O mesmo para todos os veículos
	9.2 Tempos diversos
	9.3 Sem restrições de tempo
10. Custos	10.1 Variáveis (associados à rota escolhida)
	10.2 Fixos
11. Operação	11.1 De entrega
	11.2 De recolhimento
	11.3 Ambas
12. Objetivo	12.1 Minimizar custos fixos
	12.2 Minimizar custos de operação na rota
	12.3 Minimizar o número de veículos
13. Restrições na capacidade dos arcos	13.1 Imposta a todos os arcos
	13.2 Imposta a um subconjunto de arcos
	13.3 Sem restrições

Fonte: Adaptado de Bodin e Golden (1981).

De forma mais detalhada, Eksioglu et al. (2009) propuseram uma taxonomia com três níveis de subdivisão. O primeiro nível apresenta cinco categorias: (1) tipo de estudo; (2) características do cenário; (3) características físicas do problema; (4) características de informação; e (5) características dos dados. Na primeira categoria, uma publicação é classificada de acordo com seu conteúdo. Na segunda, listam-se os fatores que não fazem parte das restrições incorporadas na solução, mas parte do contexto dos problemas. A terceira abrange os fatores que afetam diretamente a solução do problema e trata-se de uma extensão do trabalho de Bodin e Golden (1981). A quarta categoria lida com a solução do PRV apresentada pela qualificação da natureza, acessibilidade e processamento da informação. A quinta e última categoria objetiva classificar os tipos de dado com base na sua origem. O Quadro 2 apresenta a classificação proposta pelos autores.

Quadro 2. Taxonomia para o PRV em três níveis segundo Eksioglu et al. (2009)

1. Tipo de Estudo	3.5 Número de ponto de instalações de carga/descarga (depósitos)
1.1 Teoria	3.5.1 Garage única
1.2.1 Métodos exatos	3.5.2 Múltiplas garagens
1.2.2 Heurísticas	3.6 Tipo de janela de tempo
1.2.3 Simulação	3.6.1 Restrição sobre os locais
1.2.4 Métodos de solução de tempo real	3.6.2 Restrições sobre as vias
2. Características do Cenário	3.6.3 Restrições sobre a garage
2.1 Número de paradas na rota	3.6.4 Restrições sobre os motoristas/veículos
2.1.1 Conhecida (determinística)	3.7 Número de veículos
2.1.2 Parcialmente conhecida, parcialmente probabilística	3.7.1 Exatamente $n$ veículos (TSP neste segmento)
2.2 Restrição de carregamento fracionado	3.7.2 Até $n$ veículos
2.2.1 Fracionamento permitido	3.7.3 Número ilimitado de veículos
2.2.2 Fracionamento não permitido	3.8 Consideração de capacidade
2.3 Quantidade demandada pelo local	3.8.1 Veículos capacitados
2.3.1 Determinística	3.8.2 Veículos não capacitados
2.3.2 Estocástica	3.9 Homogeneidade do veículo (capacidade)
2.3.3 Desconhecida <sup>1</sup>	3.9.1 Veículos similares
2.4 Tempo de solicitação de novos locais	3.9.2 Veículos com cargas específicas <sup>2</sup>
2.4.1 Determinística	3.9.3 Veículos heterogêneos
2.4.2 Estocástica	3.9.4 Veículos com locais específicos <sup>3</sup>
2.4.3 Desconhecida	3.10 Tempo de viagem
2.5 Tempo de espera/serviço no local	3.10.1 Determinístico
2.5.1 Determinístico	3.10.2 Função dependente (uma função do tempo atual)
2.5.2 Tempo dependente	3.10.3 Estocástico
2.5.3 Tipo de veículo dependente	3.10.4 Desconhecido
2.5.4 Estocástico	3.11 Custo de transporte
2.5.5 Desconhecido	3.11.1 Dependente do tempo de viagem
2.6 Estrutura de janela de tempo	3.11.2 Dependente da distância
2.6.1 Janelas de tempo flexíveis	3.11.3 Dependente do veículo <sup>4</sup>
2.6.2 Janelas de tempo rígidas	3.11.4 Dependente da operação
2.6.3 Mistura de ambas	3.11.5 Função do atraso
2.7 Horizonte de tempo	3.11.6 Relacionado com perigo/ risco implícito
2.7.1 Período único	4. Características de informação
2.7.2 Múltiplos períodos	4.1 Evolução da informação
2.8 <i>Backhauls</i>	4.1.1 Estática
2.8.1 Nós solicitam coleta e entrega simultânea	4.1.2 Parcialmente dinâmica
2.8.2 Nós solicitam tanto o serviço <i>linehaul</i> quanto <i>backhaul</i> , mas não ambos	5. Característica dos dados
2.9 Restrições de coberturas de nós/arcos	5.1 Dados usados
2.9.1 Restrições de ligação e precedência	5.1.1 Dados do mundo real
3.4 Número de pontos de origem	5.1.2 Dados sintéticos
3.4.1 Origem única	5.1.3 Dados reais e sintéticos
3.4.2 Múltiplas origens	5.2 Nenhum dado usado

Fonte: Eksioglu et al. (2009).

Aqui foram escolhidas, a critério do autor, três classes do PRV, com propósito de analisar as peculiaridades do planejamento de inventários florestais: Problema de Roteamento de Veículos Capacitado (PRVC), Problema de Roteamento Periódico

<sup>1</sup> Desconhecida refere-se ao caso no qual a informação é revelada em tempo real (ou seja, estudos dinâmicos e *fuzzy* caem nessa categoria).

<sup>2</sup> Cada veículo pode ser usado para lidar com um tipo específico de carga.

<sup>3</sup> Um local deve ser visitado por um tipo específico de veículo.

<sup>4</sup> Custo de operar um veículo não é negligenciado.

de Veículos (PRPV) e Problema de Roteamento de Veículos com Janela de Tempo (PRVJT).

### 2.3.1. Problema de Roteamento de Veículos Capacitado - PRVC

O PRVC é considerado a versão clássica do PRV. A definição formal do problema é como segue. Seja  $G = (V, E)$  um grafo completo com um conjunto de vértices  $V = \{0, \dots, L\}$ , em que o vértice 0 representa a garage e os demais vértices os locais. Cada  $\{i, j\} \in E$  tem um custo  $d_{i,j}$  não-negativo, e cada local  $i \in V' = V/\{0\}$  tem uma demanda  $q_i$ . Sendo  $K = \{1, \dots, k\}$  um conjunto de veículos homogêneos com capacidade  $Q$ . O PRVC consiste na construção de  $m$  conjuntos de rotas, de tal maneira que: (i) cada percurso começa e termina na garage; (ii) todas as restrições sejam atendidas; (iii) a capacidade do veículo não seja excedida; (iv) um local seja visitado por apenas um único veículo; e (v) a soma dos custos seja minimizada. Uma das formulações mais utilizadas do PRV como base em diversos métodos de solução é a formulação de Fisher e Jaikumar (1981).

**Minimizar**

$$\sum_{i \in L} \sum_{j \in L} \sum_{k \in K} d_{i,j} x_{i,j,k} \quad (1)$$

**sujeito a:**

$$\sum_{k \in K} y_{i,k} = 1 \quad \forall i = 1, \dots, L \quad (2)$$

$$\sum_{k \in K} y_{i,k} = K \quad \forall i = 0 \quad (3)$$

$$\sum_{i \in L} q_i y_{i,k} \leq Q_k \quad \forall k \in K \quad (4)$$

$$\sum_{j \in L} x_{i,j,k} = \sum_{j \in L} x_{j,i,k} = y_{i,k} \quad \forall i \in L; k \in K \quad (5)$$

$$\sum_{i,j \in S} x_{i,j,k} \leq |S| - 1 \quad \forall S \subseteq \{1, \dots, L\}; k \in K \quad (6)$$

$$y_{i,k} \in \{0,1\} \quad \forall i \in L; k \in K \quad (7)$$

$$x_{i,j,k} \in \{0,1\} \quad \forall i \in L; j \in L; k \in K \quad (8)$$

em que:

$x_{i,j,k}$  = variável binária que assume valor 1 quando o veículo  $k$  visita o local  $j$  imediatamente após o local  $i$ , e 0 caso contrário;

$y_{i,k}$  = variável binária que assume o valor 1 se o local  $i$  é visitado pelo veículo  $k$ , 0 em caso contrário;

$q_i$  = demanda do local  $i$ ;

$Q_k$  = capacidade do veículo  $k$ ;

$d_{i,j}$  = custo de deslocamento para percorrer o trecho entre o local  $i$  e  $j$ .

A função objetivo tem como finalidade minimizar os custos de viagem das rotas feitas por todos os veículos, enquanto que as restrições (2) asseguram que um veículo não visite um local mais de uma vez. As restrições (3) garantem que o número de veículos que deixam o depósito seja igual a  $K$ . As restrições (4) cuidam para que a capacidade dos veículos não seja ultrapassada. As restrições (5) garantem que um veículo não encerre sua rota em um local, obrigando-o sempre encerrá-la em uma garage. As restrições (6) eliminam os subciclos. As restrições (7) e (8) definem os valores que as variáveis  $y_{i,k}$  e  $x_{i,j,k}$  podem assumir para cada veículo  $k$ .

### 2.3.2. Problema de Roteamento Periódico de Veículos - PRPV

O Problema de Roteamento Periódico de Veículos (PRPV) é uma generalização do clássico PRV onde o horizonte de planejamento é de  $T$  dias e, para cada dia, deve-se desenvolver um PRV com os locais selecionados para serem visitados neste dia. Para este problema, considera-se  $L = \{1, 2, \dots, l\}$  o conjunto dos locais, e  $r_i$  ( $1 \leq r_i \leq t$ ) o número de visitas que devem ser realizadas a cada local  $i \in L$  durante o horizonte de planejamento. Além disso, cada local possui um conjunto de possíveis combinações de dias de visita  $V(i)$ , em que cada combinação é composta por  $t_i$  dias em que o local  $i$  deve ser visitado (GONÇALVES, 2005).

De forma ilustrativa, se um local  $j$  requer três visitas durante o período de  $T = 5$  dias e existindo duas possíveis combinações,  $V(j) = \{\{1, 3, 4\}, \{2, 3, 5\}\}$ , então, deve-se visitar o local  $j$  nos dias 1, 3 e 4 se a combinação  $\{1, 3, 4\}$  for selecionada ou nos dias 2, 3 e 5, caso a segunda combinação seja escolhida. Dessa forma, no PRVP deve-se simultaneamente selecionar uma combinação de visitas para cada local e determinar rotas para os veículos em cada dia de modo que a distância total percorrida no horizonte de planejamento seja minimizada.

Diferentemente do PRV Clássico, o PRPV é um modelo de roteamento de veículos onde os locais necessitam de visitas com uma determinada frequência, seja para reabastecimento de estoque ou coleta de produtos ou resíduos. Assim, há a necessidade de definir primeiramente quais locais serão atendidos em cada dia, para assim realizar o roteamento desses locais escolhidos, respeitando o número de veículos disponíveis por dia de planejamento e também a restrição de capacidade que cada veículo possui (MUNHOZ et al., 2012). Dessa forma, o problema

normalmente é abordado em duas fases: a primeira define quais os padrões de visita dos locais (Problema da Atribuição) e posteriormente realiza-se o roteamento com os locais já fixados em seus dias (MUNHOZ, 2012).

Cristofides e Beasley (1984) apresentaram a primeira formulação matemática para o PRPV. Eles definiram esse problema como um problema de designar um conjunto de rotas para cada dia de um período de planejamento de  $|T|$  dias, atendendo a frequência de visita de cada um dos locais. A formulação utiliza-se dos conjuntos de variáveis de decisão, um para destinar locais ao período de visitas (agendas), e outro para o roteamento de um dado veículo em um determinado dia. São utilizadas três variáveis de decisão:  $x_{i,j,k}^t$ ,  $z_i^s$  e a variável de decisão binária  $v_i^t$ , que recebe o valor 1 caso o local  $i \in L_t$  seja visitado no dia  $t \in T$ , e o valor 0, caso contrário. Esta formulação é apresentada a seguir.

**Minimizar**

$$\sum_{t \in T} \sum_{(i,j) \in A} \sum_{k \in K} d_{i,j} x_{i,j,k}^t \quad (1)$$

**sujeito a:**

$$\sum_{s \in S_i} z_i^s = 1 \quad \forall i \in L_t \quad (2)$$

$$v_i^t = \sum_{s \in S_i} z_i^s a_{s,d} \quad \forall t \in T; i \in L_t \quad (3)$$

$$\sum_{k \in K} x_{i,j,k}^t \leq \frac{v_i^t + v_j^t}{2} \quad \forall t \in T; i, j \in L_t (i \neq j) \quad (4)$$

$$\sum_{k \in K} x_{i,j,k}^t = \sum_{j \in L_t} x_{j,i,k}^t \quad \forall i \in L; k \in K; t \in T \quad (5)$$

$$\sum_{k \in K} \sum_{i \in L} x_{i,j,k}^t = \begin{cases} v_j^t & \forall j \in L_t \\ |K| & j = 0 \end{cases} \quad \forall t \in T \quad (6)$$

$$\sum_{i,j \in R} x_{i,j,k}^t \leq |R| - 1 \quad \forall R \subseteq L_t; k \in K; t \in T \quad (7)$$

$$\sum_{j \in L_t} x_{0,j,k}^t \leq 1 \quad \forall k \in K; t \in T \quad (8)$$

$$\sum_{i \in L_t} q_i \sum_{j \in N} x_{i,j,k}^t \leq Q \quad \forall k \in K; t \in T \quad (9)$$

$$\sum_{(i,j) \in A} d_{i,j} x_{i,j,k}^t \leq L \quad \forall k \in K; t \in T \quad (10)$$

$$z_i^s \in \{0,1\} \quad \forall i \in N_c; s \in S_i \quad (11)$$

$$x_{i,j,k}^t \in \{0,1\} \quad \forall (i,j) \in A; k \in K; t \in T \quad (12)$$

em que:

$d_{i,j}$  = custos de transporte entre os locais  $i$  e  $j$ ;

$x_{i,j,k}^t = 1$  caso o veículo  $k \in K$  percorre o arco  $(i, j)$  no dia  $t \in T$  e 0 caso contrário;

$z^s_i = 1$  se o local  $i \in L_t$  é visitado na agenda  $s \in S$  e 0 caso contrário;

$v^t_i = 1$  se a agenda for escolhida e 0 caso contrário;

$a_{s,t} = 1$  se o dia  $t \in T$  pertence à agenda  $s \in S$  e 0 caso contrário;

$q_i$  = demanda do local  $i$ ;

$T$  = conjunto dos dias de visita do período de planejamento;

$A$  = conjunto de arestas  $(i, j)$ ;

$K$  = conjunto de veículos disponíveis;

$L$  = conjunto de locais.

Cada local está associado a uma lista de dias possíveis (agendas) para visita. Essa lista especifica em quais dias do horizonte o local pode ser atendido. Cada cliente possui uma ou mais agendas de visita, sendo que exatamente uma deve ser atendida. Portanto, define-se o conjunto  $S$  como todas as possíveis agendas, indexadas por  $s \in S$ . Cada agenda em  $S$  é totalmente descrita por um vetor binário  $a_{s,t}$ . Além disso, no PRPV cada nó requer um número fixo de visitas durante um período de planejamento, definido por  $f_i$ . Por isso, para cada ponto  $i \in L_t$ , uma agenda dentre o subconjunto não vazio de possíveis agendas candidatas  $s_i \subseteq S$  deve ser escolhida. Esse subconjunto é definido como:

$$s_i = \left\{ s \in S : \sum_{t \in T} a_{s,t} = f_i \right\}$$

A função objetivo (1) minimiza os custos de deslocamento entre os locais. As restrições (2) garantem que uma agenda viável seja escolhida para cada ponto, enquanto as restrições definidas por (3) atribuem  $v^t_i$  para os dias da agenda escolhida. As restrições (4) permitem apenas a ligação de locais que foram alocados para o dia  $t \in T$ . As restrições (5) são responsáveis pela conservação de fluxo. As restrições (6) garantem que somente sejam incluídos locais em rotas para os dias dentro da sua agenda atribuída. As restrições (7) eliminam subciclos. As restrições (8) garantem que um veículo não seja utilizado mais que uma vez por dia. As restrições (9) e (10) são as restrições de capacidade do veículo e tamanho máximo da rota, respectivamente. E por último, as restrições (11) e (12) definem a integralidade das variáveis.

### 2.3.3. Problema de Roteamento de Veículos com Janela de Tempo - PRVJT

O Problema de Roteamento de Veículos com Janela de Tempo (PRVJT) pode ser definido como segue. Seja  $G = \{V, A\}$ , um grafo em que  $V = \{v_0, v_1, \dots, v_n\}$  é o

conjunto de vértices e  $A=\{(v_i,v_j) : v_i, v_j \in V, i \neq j\}$  o conjunto de arcos. O vértice  $v_0$  representa a garage, enquanto os demais  $n$  vértices correspondem aos locais, com  $A$  associado a uma matriz de tempo de viagens  $(d_{i,j})$ . Caso essas matrizes sejam simétricas, é padrão definir o PRVJT em um grafo não direcionado  $G=\{V, E\}$ , em que  $E=\{(v_i,v_j): v_i, v_j \in V, i \neq j\}$  é um conjunto de arestas. Cada local tem uma demanda não-negativa  $q_i$  e um tempo de atendimento  $ts_i$ . Uma frota de veículos idênticos de capacidade  $Q$  é atribuída à garage. O número de veículos pode ser conhecido com antecedência ou tratado como uma variável de decisão (GOMES JUNIOR et al., 2005).

Resumidamente, o PRVJT consiste em construir um conjunto de rotas para entrega e ou coleta, tal que: cada rota inicie e termine no  $v_0$ ; cada local seja visitado exatamente uma vez por exatamente um veículo; a demanda total de cada rota não exceda  $Q$ ; e a janela de tempo  $[e_i, f_i]$  seja respeitada, ou seja, um local não pode ser atendido antes de  $e_i$  e nem após o tempo  $f_i$  e o custo total da rota seja minimizado. A formulação matemática do problema, de acordo com Vieira (2013) é dada por:

**Minimizar**

$$\sum_{i \in L} \sum_{j \in L} \sum_{k \in K} d_{i,j} x_{i,j,k} \quad \forall i \neq j; i = 1, \dots, l; j = 0, \dots, L \quad (1)$$

**sujeito a:**

$$\sum_{k \in K} \sum_{i \in L} x_{0,j,k} \leq K \quad (2)$$

$$\sum_{j \in L} x_{0,j,k} = \sum_{j \in L} x_{j,0,k} \leq 1 \quad \forall k \in K \quad (3)$$

$$\sum_{k \in K} \sum_{j \in L} x_{i,j,k} = 1 \quad \forall i \in L \quad (4)$$

$$\sum_{i \in L} x_{i,j,k} - \sum_{j \in L} x_{j,i,k} = 0 \quad \forall k \in K; i \in L \quad (5)$$

$$\sum_{k \in K} \sum_{i \in L} \sum_{j \in L} x_{i,j,k} \leq |L| - v(L) \quad \forall L \subseteq v \setminus \{0\}, |L| \geq 2 \quad (6)$$

$$\sum_{i \in L} q_i \sum_{j \in L} x_{i,j,k} \leq Q \quad \forall i \neq j; j \in L \quad (7)$$

$$\sum_{k \in K} \sum_{i \in L} x_{ijk} (b_i + ts_i + d_{i,j}) \leq b_j \quad \forall j \in L \quad (8)$$

$$e_i \leq b_i \leq f_i \quad \forall i \in L \quad (9)$$

$$x_{ijk} \in \{0,1\} \quad \forall i \in L; j \in L; k \in K \quad (10)$$

em que:

$i$  = nó de origem;

$j$  = nó de destino;

$l$  = número total de locais;

$K$  = número total de veículos;

$d_{i,j}$  = tempo de deslocamento entre o local  $i$  e o local  $j$ ;

$x_{i,j,k}$  = 0 se não for utilizado o caminho do local  $i$  para o local  $j$ , e 1 caso contrário;

$e_i$  = início do período para chegada ao local  $i$ ;

$f_i$  = término do período para chegada ao local  $i$ ;

$ts_i$  = tempo de serviço no local  $i$ ;

$L$  = conjunto de locais;

$q_i$  = demanda;

$Q$  = capacidade do(s) veículo(s);

$b_i$  = instante de início do atendimento no local  $i$ ; e

$b_j$  = instante de início de atendimento no local  $j$ .

A função objetivo (1) visa minimizar os custos fixos e variáveis relacionados com a distância percorrida. As restrições (2) garantem que  $K$  veículos sairão da garage, enquanto (3) asseguram que cada rota tenha início e fim na garage. As restrições (4) e (5) garantem que cada local seja visitado exatamente uma vez, e que o veículo que chega em um local saia dele. As restrições (6) evitam que seja formado uma subciclo que não inclua a garage. Nesta restrição,  $v(L)$  representa o número mínimo de veículos necessários para atender o conjunto de locais  $L$ .

Por exigir que o número de veículos usados para atender os locais do conjunto  $L$  não seja inferior a  $v(L)$ , o mínimo necessário, as restrições (7) asseguram, indiretamente, que a capacidade dos veículos não seja violada.

As restrições (8) relacionam o instante de início do atendimento de dois locais visitados consecutivamente por um mesmo veículo. Ela evita que o intervalo entre os instantes de início do atendimento desses locais seja inferior à soma do tempo gasto no atendimento do primeiro local com o tempo consumido na viagem entre os dois. A folga das restrições (8) corresponde ao tempo de espera do veículo que atende o local  $j$ , ou seja, o tempo consumido entre o instante de chegada do veículo ao local e o início do atendimento deste.

A restrições (9) impedem que o início do atendimento do local  $i$  não ocorra fora de sua janela de tempo. E, finalmente, as restrições (10) garantem que o modelo seja binário.

## **2.4. Meta-heurísticas como possibilidade de solução para problemas complexos**

Meta-heurísticas, *a priori*, são utilizadas para resolver problemas que não podem ser tratados por métodos exatos, tentando reproduzir, muitas vezes, o que é

realizado manualmente. Para Gomes e Neto (2003), o objetivo principal de uma meta-heurística é encontrar soluções de boa qualidade, mas com um baixo custo computacional. Osman e Laporte (1996) definem meta-heurísticas como métodos aproximados, pois contam com um processo de geração iterativa de soluções. Esses métodos utilizam heurísticas subordinadas e combinam conceitos inteligentes de busca para encontrar de forma eficiente soluções aproximadas da solução ótima.

De acordo com Arroyo (2002), a diferença entre as meta-heurísticas está basicamente nos princípios e nas estratégias de busca utilizadas. Na maioria das vezes, esses métodos se diferenciam pelo critério de escolha de uma solução inicial, ou na definição da vizinhança de uma solução, ou ainda no critério de seleção de uma solução vizinha ou também no critério de término (COLLETTE; SIARRY, 2003).

Na literatura, são apresentadas diversas meta-heurísticas, cada qual destinada, em geral, à resolução de uma única variante do PRV. Dadas as dificuldades relatadas por Meneguzzi (2011) quanto ao uso de métodos exatos nesse tipo de problema, nesta pesquisa optou-se pelo uso de duas meta-heurísticas como alternativas para resolver o problema de roteamento do planejamento de inventários florestais: *Iterated Local Search* e *Simulated Annealing*. Esses algoritmos foram escolhidos devido à simplicidade, a velocidade computacional e a qualidade das soluções obtidas em trabalhos da literatura.

#### **2.4.1. *Iterated Local Search* - ILS**

A meta-heurística ILS, de acordo com Lourenço et al. (2003), é um método de busca iterativa que faz uso de perturbações em ótimos locais com o objetivo de diversificar e visitar outras regiões do espaço de soluções. O pseudocódigo da Figura 1 descreve os quatro principais componentes que definem o método ILS: geração da solução inicial, busca local, perturbação e critério de aceitação.

O primeiro passo do método é construir uma solução inicial  $S_0$ . Uma vez construída,  $S_0$  é submetida a uma busca local onde se obtém  $S^*$  (passos 3 e 4). Posteriormente, enquanto o critério de parada não for atendido, o método realiza iterativamente os seguintes passos: i) efetua uma perturbação na solução corrente  $S^*$  e obtém  $S'$  (passo 6); ii) aplica uma busca local em  $S'$  com a finalidade de melhorar a solução perturbada, obtendo assim  $S^{**}$  (passo 7); e iii) neste último passo, a solução obtida  $S^{**}$  é submetida ao critério de aceitação, caso aceite,  $S^{**}$  é retornada para uma iteração, caso não, a melhor solução conhecida até o momento  $S^*$  é retornada.

Figura 1. Meta-heurística *Iterated Local Search*

```

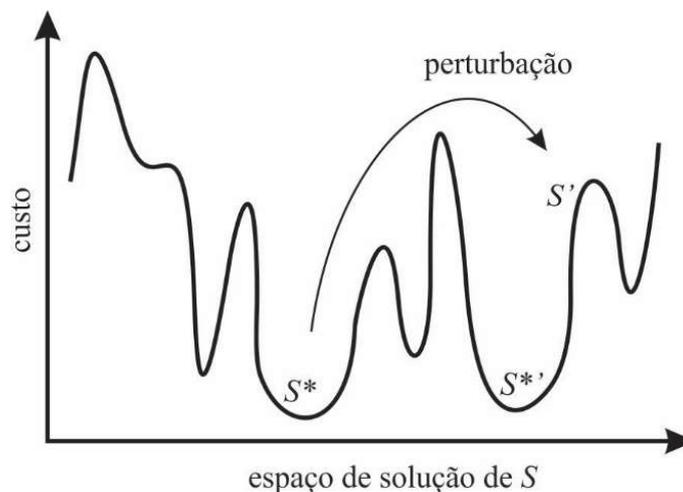
1. ILS(critério de parada)
2. INÍCIO
3.    $S_0 \leftarrow$  Construção;
4.    $S^* \leftarrow$  BuscaLocal( $S_0$ );
5.   REPITA
6.      $S' \leftarrow$  Perturbação( $S^*$ , Histórico);
7.      $S^{*'} \leftarrow$  BuscaLocal( $S'$ );
8.      $S^* \leftarrow$  CritérioDeAceitação( $S^*$ ,  $S^{*'}$ , Histórico)
9.   ATÉ critério de parada;
10. FIM.

```

Fonte: Lourenço et al. (2003).

Lourenço et al. (2003) destacam ainda que o sucesso do ILS está diretamente relacionado com as perturbações realizadas. Porém, as perturbações devem ser dosadas de tal maneira que as alterações sejam suficientes para escapar de ótimos locais e explorar diferentes regiões sem, contudo, causar a perda de características do ótimo local corrente. A Figura 2 representa graficamente o funcionamento do método ILS. Começa com um mínimo local  $S^*$ , que após aplicada a perturbação leva a uma solução  $S'$ . Após aplicada a busca local em  $S'$ , um novo mínimo local, que pode ser melhor que  $S^*$ , é encontrado ( $S^{*'}$ ).

Figura 2. Representação gráfica da meta-heurística ILS



Fonte: Lourenço et al. (2003).

Existem na literatura diversas aplicações da meta-heurística ILS em Problemas de Roteamento de Veículos. Dentre elas, pode-se citar algumas que, indiretamente motivaram a escolha deste método, devido aos bons resultados obtidos para este tipo de problema: Ribas (2011), Subramanian, 2012, Penna et al. (2013), Cattaruzza et al. (2014), Michallet et al. (2014) e Silva et al. (2015).

### 2.4.2. Simulated Annealing - SA

A meta-heurística *Simulated Annealing* (SA) fundamenta-se em uma analogia com a termodinâmica ao simular o resfriamento de um conjunto de átomos aquecidos, operação conhecida como recozimento. O SA foi proposto originalmente por Kirkpatrick et al. (1983) como um método de busca local que aceita movimentos de piora para escapar de ótimos locais. O método SA, representado na Figura 3, começa sua busca a partir de uma solução inicial qualquer (passo 3). O procedimento principal consiste em um *loop* que gera aleatoriamente, em cada iteração, um único vizinho  $S'$  da solução corrente  $S$ .

Considerando um problema de minimização, seja  $\Delta$  a variação de valor da função objetivo ao mover-se para uma solução vizinha candidata, isto é,  $\Delta = f(S') - f(S)$ . O método aceita o movimento e a solução vizinha passa a ser a nova solução corrente se  $\Delta < 0$ . Caso  $\Delta \geq 0$ , a solução vizinha candidata também poder ser aceita, neste caso, com uma probabilidade  $e^{-\Delta/T}$ , em que  $T$  é um parâmetro do método, chamado de temperatura e que regula a probabilidade de se aceitar soluções de pior custo.

O parâmetro de temperatura  $T$  possibilita, no início do procedimento, que soluções de pior custo tenham maior probabilidade de serem aceitas. Isso acontece porque  $T$  assume um valor elevado  $T_0$  (passo 5). Por conseguinte, após um número fixo de iterações (o qual representa o número de iterações necessárias para o sistema atingir o equilíbrio térmico em uma determinada temperatura), a temperatura é gradativamente diminuída por uma razão de resfriamento  $\alpha$ , tal que, em um instante  $t$ ,  $T_t \leftarrow \alpha * T_{t-1}$ , sendo  $\alpha$  um valor compreendido entre 0 e 1. Nesta etapa, o algoritmo comporta-se como um método de descida, uma vez que diminui a probabilidade de se aceitar movimentos de piora.

Quando a temperatura atinge o valor de congelamento ( $T_c$ ), o procedimento para e a melhor solução encontrada é retornada. Dessa forma, os parâmetros de controle do procedimento são a razão de resfriamento  $\alpha$ , o número de iterações para cada temperatura  $I_{eq}$ , a temperatura inicial  $T_0$  e a temperatura de congelamento  $T_c$ .

Algumas aplicações do método SA em Problemas de Roteamento de Veículos são apresentadas em Osman (1993), Chiang e Russel (1996), Tan et al. (2001), Czech e Czarnas (2002) e Tavakkoli-Moghaddam (2011).

Figura 3. Meta-heurística *Simulated Annealing*

```

1.  SA(Ieq, Tres, T0, Tc e S)
2.  INÍCIO
3.  S* ← S;
4.  IAtual ← 0;
5.  T ← T0;
6.  ENQUANTO T < Tc FAÇA
7.  ENQUANTO IAtual < Ieq FAÇA
8.  IAtual ← IAtual + 1;
9.  S' ← N(S); {Movimento de variação de S'}
10. Δ ← f(S') - f(S);
11. SE Δ < 0 ENTÃO
12. S ← S';
13. SE (f(S') < f(S*)) ENTÃO
14. S* ← S';
15. FIM-SE;
16. SENÃO
17. SE CriterioAceitacao ENTÃO
18. S ← S';
19. FIM-SE;
20. FIM-SE;
21. FIM-ENQUANTO;
22. T ← Tres * T;
23. IAtual ← 0;
24. FIM-ENQUANTO;
25. S ← S*;
26. FIM.

```

Fonte: MAURI (2008).

## 2.5. Trabalhos relacionados ao planejamento de inventários florestais

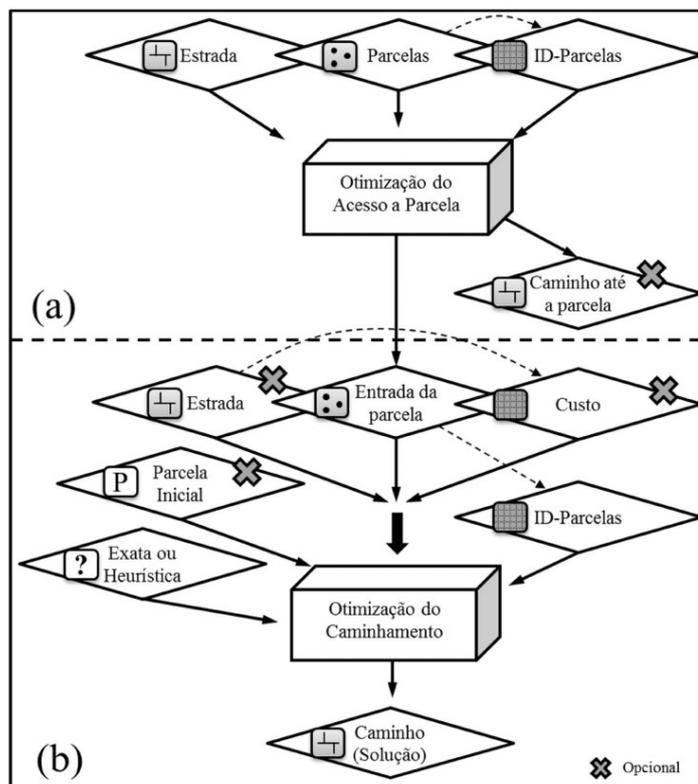
A seguir são descritos trabalhos relacionados ao planejamento de inventários florestais. Os trabalhos de Marcatti (2013) e Haddad (2015) contêm características restritas ao problema abordado nesta pesquisa. Em contrapartida, o trabalho proposto por Meneguzzi (2011) tem uma ampla relação com o problema.

### 2.5.1. Metodologia proposta por Marcatti (2013)

O objetivo do estudo de Marcatti (2013) foi desenvolver metodologias para otimização do acesso e caminhamento às parcelas de inventário florestal, bem como implementá-las em um ambiente com interface gráfica integradas a um sistema de informação geográficas. Para isso, foi proposto um método em duas etapas (Figura 4). A primeira etapa consiste na identificação da entrada e do caminho até a primeira parcela. Na segunda etapa, realiza-se a otimização do caminhamento entre parcelas.

A otimização do acesso à parcela (Figura 4) foi baseada no algoritmo do vizinho mais próximo. O algoritmo determina os pontos da estrada mais próximos a cada parcela. O problema do caminhamento entre as parcelas foi formulado como um problema do caixeiro-viajante (PCV), permitindo-se, contudo, a imposição da malha viária no processo de otimização.

Figura 4. Fluxograma da ferramenta desenvolvida por Marcatti (2013). (a) Implementação da metodologia de otimização do acesso a parcela. (b) Implementação da metodologia de otimização do caminhamento



Fonte: Marcatti (2013).

O problema foi resolvido via formulação de programação linear inteira e otimizado pelo algoritmo *cutting-plane*. Os resultados obtidos foram comparados com os resultados obtidos pelo método aproximativo do vizinho mais próximo, na tentativa de simular as atividades realizadas intuitivamente pelo executor em campo.

Para os experimentos, foram consideradas bases de dados espaciais compostas da localização de 80 parcelas de inventário distribuídas em 15 talhões e da rede de estradas local. A metodologia de acesso às parcelas otimiza o caminhamento entre a estrada e as parcelas de inventário florestal, e sua implementação computacional possibilita a automatização da tarefa. O autor destaca o aumento da eficiência tanto no planejamento do inventário quanto em sua execução. Em relação à metodologia de caminhamento entre as parcelas a serem inventariadas, os resultados mostraram a capacidade de otimização do modelo, visto que a solução exata se mostrou em média 17% superior à solução aproximada.

### 2.5.2. Metodologia proposta por Haddad (2015)

O trabalho de Haddad (2015) foi realizado com o objetivo de avaliar a viabilidade da aplicação de um modelo de roteamento via Programação Linear Inteira (PLI), visando melhorar o planejamento do inventário florestal urbano da cidade de Belo Horizonte de forma que minimizasse as distâncias percorridas pelas equipes de inventário. Para isso, vários fatores foram levados em consideração, tais como a quantidade e a distribuição espacial das árvores a serem inventariadas, as ruas e caminhos disponíveis para locomoção das equipes, o número de equipes e o número de membros por equipe, entre outros.

Nesse estudo, os veículos foram substituídos pelas equipes e os locais pelas ruas, onde estão presentes as árvores a serem inventariadas. O objetivo do modelo utilizado foi minimizar as distâncias a serem percorridas pelas equipes, de forma a aumentar o rendimento e diminuir o tempo gasto na realização do inventário em um lote e, desta forma, melhorar o planejamento e execução do inventário florestal urbano.

O modelo matemático empregado nesse estudo teve como base um modelo da literatura para o Problema do Carteiro Chinês (PCC). Na formulação, foi considerado o uso de variáveis de decisão inteiras, via programação linear inteira. O modelo descrito por Haddad (2015) é representado a seguir:

**Minimizar**

$$\sum_{i=1}^I \sum_{j \in C_i} d_{i,j} x_{i,j} + \sum_{i=1}^I \sum_{j \in C_i} d_{i,j} y_{i,j} \quad (1)$$

**sujeito a:**

$$x_{i,j} + x_{j,i} = 1 \quad j \in C_i; \forall i; \forall j \quad (2)$$

$$x_{i,j} - x_{j,i} + y_{i,j} - y_{j,i} = 0 \quad j \in C_i; \forall i; \forall j \quad (3)$$

$$y_{i,j} + y_{j,i} = 1 \quad j \in C_i; \forall i; \forall j \quad (4)$$

$$x_{i,j} = \{0,1\}, y_{i,j} = \{0,1\} \quad (5)$$

em que:

$d_{i,j}$  = custo de deslocamento entre os vértices  $i$  e o adjacente  $j$  em metros;

$x_{i,j} = 1$  a atividade (censo) foi executada, e 0 caso contrário;

$y_{i,j}$  = garante a continuidade do fluxo.

A função objetivo (1) visa minimizar o somatório do custo de deslocamento ( $d$ ) real e linear em metros, entre o local  $i$  e o adjacente  $j$ . As restrições (2) garantem a

passagem das equipes de inventário em todas as ruas uma única vez, independente do sentido de caminamento, realizando a medição das árvores. As restrições (3) garantem o fluxo de deslocamento entre os locais presentes no grafo, sendo os locais representados pelas esquinas das ruas. As restrições (4) garantem a união de grafos desconexos dentro de um lote, aplicada somente a alguns cenários. A restrição (5) contempla as variáveis de decisão do modelo de programação linear inteira, sendo  $x_{i,j}$  e  $y_{i,j}$  binárias, representando o deslocamento da equipe entre os locais adjacentes  $i$  e  $j$ .

Além do método exato, nesse trabalho foi proposta a meta-heurística *Simulated Annealing* para resolução do roteamento, na tentativa de boas soluções em um tempo inferior ao método exato. Com o objetivo de minimizar o custo de deslocamento entre os pontos, os resultados obtidos por meio do modelo de PLI proporcionou uma melhora de 6,36% em relação ao método tradicional realizado de forma empírica. A meta-heurística *Simulated Annealing* obteve um bom desempenho, já que seu tempo de processamento foi 7.200 vezes menor para obtenção de uma solução viável a 1,87% do valor encontrado pela PLI.

### 2.5.3. Metodologia proposta por Meneguzzi (2011)

O modelo de Roteamento de Veículos Aplicado ao Planejamento do Inventário Florestal proposto por Meneguzzi (2011) é baseado no modelo de Toth e Vigo (2002) utilizado no Problema de Roteamento de Veículos com Janela de Tempo. O veículo foi substituído por mês de trabalho e a formulação do modelo foi:

**Minimizar**

$$\sum_{i \in L} \sum_{j \in L} \sum_{t \in T} d_{i,j} x_{i,j}^t \quad (1)$$

**sujeito a**

$$\sum_{j \in V^t} x_{0,j}^t = 1 \quad \forall t = 1, \dots, T \quad (2)$$

$$\sum_{i \in V^t} x_{i,0}^t = 1 \quad \forall t = 1, \dots, T \quad (3)$$

$$\sum_{t \in T} \sum_{j \in V^t} x_{i,j}^t = 1 \quad \forall i \in V^t \quad (4)$$

$$\sum_{j \in V^t \cup \{0\}} x_{i,j}^t - \sum_{j \in V^t \cup \{0\}} x_{i,j}^t = 0 \quad \forall j \neq i; t \in T; i \in V^t \quad (5)$$

$$\sum_{i \in V^t \cup \{0\}} \sum_{j \in V^t \cup \{0\}} (e_{i,j} x_{i,j}^t) + E_t = 0 \quad \forall t = 1, \dots, T \quad (6)$$

$$\sum_{i \in S} \sum_{j \in S} x_{i,j}^t \leq |S| - 1 \quad \forall S \subset V^t, 2 \leq |S| \leq |V^t| - 1; t = 1, \dots, T \quad (7)$$

$$x_{i,j}^t \in \{0,1\} \quad \forall t = 1, \dots, T; i \in V^t \cup \{0\}; j \in V^t \cup \{0\}; j \neq i \quad (8)$$

em que:

$T$  = número de meses (rotas) do planejamento;

$V^t$  = conjunto com as medições (locais) que podem ser realizadas no mês  $t$ ;

$t$  = meses de trabalho representado;

$d_{i,j}$  = distância entre as medições  $i$  e  $j$  (incluindo o escritório – indicado por 0);

$e_{i,j}$  = número de equipes necessárias para atender a medição  $j$  após a medição  $i$ ;

$E_t$  = número de equipes disponíveis no mês  $t$  ( $t=1, \dots, T$ );

$x_{i,j}^t = 1$  se no mês  $t$  a medição  $j$  é atendida após a medição  $i$ , e 0 caso contrário.

A função objetivo (1) visa minimizar a distância total entre as medições de todas as rotas mensais, conforme programação anual da atividade de inventário florestal. As restrições (2) e (3) indicam que a rota deverá ser iniciada e finalizada no escritório apenas uma vez em cada mês. As restrições (4) garantem que cada medição seja selecionada para ser medida em um único mês, no período de planejamento e conforme datas flexíveis de medição, e as restrições (5) garantem a “contenção de fluxo”, ou seja, quando uma medição  $i$  é selecionada para ser medida, a próxima medição  $j$  selecionada deve sempre partir da medição  $i$ . As restrições (6) calculam a capacidade de recursos humanos (número de equipes) necessários para atender à empresa no período planejado. As restrições (7) impedem a geração de subciclos. Por fim, as restrições (8) garantem que as variáveis de decisão sejam binárias.

O objetivo dessa pesquisa foi de propor um modelo capaz de programar as atividades de inventário para um horizonte de um ano. Em cada atividade é necessário que seja obedecida a flexibilidade quanto a data de visita em relação ao plantio do povoamento. O povoamento é escolhido uma única vez no horizonte planejado para que seja possível obter uma melhor dispersão espacial diária. O modelo também permite que seja considerado o número de equipes disponíveis na empresa.

### 3. PLANEJAMENTO DE INVENTÁRIOS FLORESTAIS

Para melhor ilustrar o planejamento de inventários florestais, considere uma área florestal composta por povoamento equiâneos subdividido em diversas subpartes denominadas talhões. Um talhão é uma área onde são plantadas mudas com características semelhantes, tais como: data de plantio, espécie, clone, espaçamento e data de colheita. A idade de rotação dos talhões varia de 4 a 7 anos, e sua colheita ocorre de acordo com as necessidades da empresa.

Durante uma rotação, existe a necessidade de que sejam realizados inventários com o propósito de obter informações do maciço florestal. E para não medir todas as árvores de um talhão, as equipes instalam parcelas em determinados pontos. Assim, a coleta de dados é realizada por amostragem e posteriormente extrapolada para o todo com uso de modelos estatísticos.

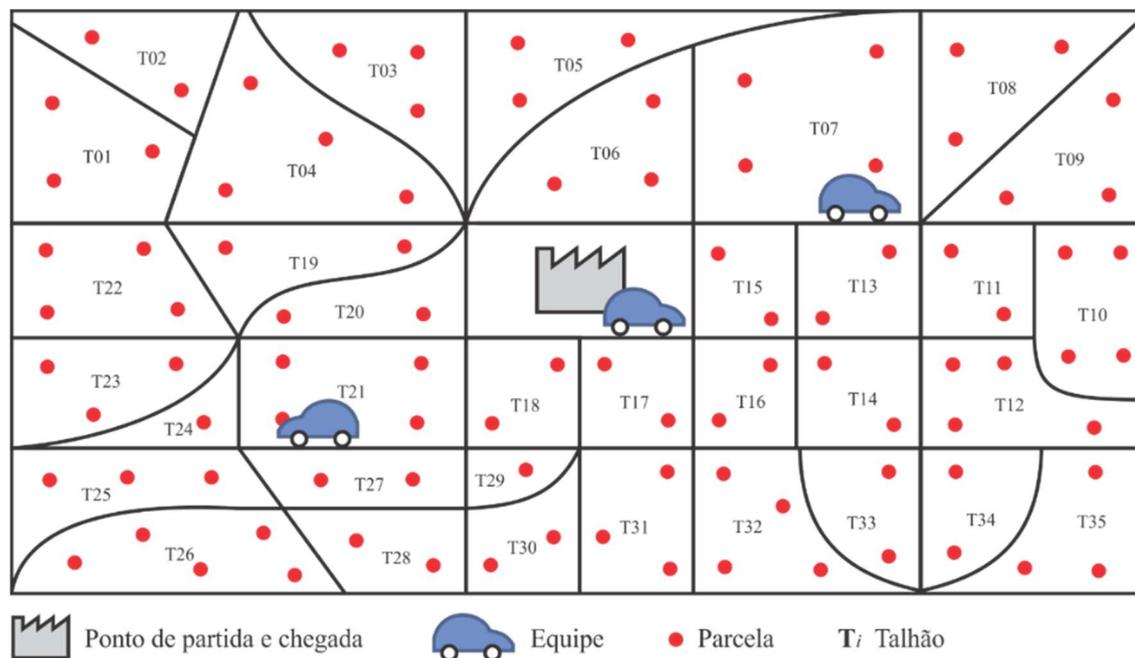
De acordo com Meneguzzi (2011), a necessidade de se obter informações precisas do inventário florestal torna importante que se defina a quantidade de parcelas necessárias para cada tipo de atividade a ser realizada em um talhão. Como há variabilidade no tamanho, em termos de área (hectare), dos talhões, o número de parcelas instaladas pode variar conforme essa característica.

A produção da matéria prima florestal, especialmente a madeira, exige normalmente o plantio de grandes áreas. Com isso, o número de parcelas instaladas torna-se muito grande, fazendo com que seja necessário um prévio planejamento e programação das atividades. A programação consiste em definir quais talhões serão visitados por qual equipe e em qual período do tempo. Assim, equipes responsáveis pelo inventário de parcelas distribuídas ao longo do plantio florestal devem se deslocar diariamente entre os talhões para realizar as medições e retornar ao final do dia ao escritório. Devido às distâncias entre os talhões e o número de equipes disponíveis, o PRV-IF consiste em gerar uma programação da atividade de maneira que reduza a distância total das rotas diárias percorridas pelas equipes, considerando as datas previstas para medição de cada parcela.

Esse problema foi apresentado por Meneguzzi (2011), que propôs o uso de um modelo de roteamento de veículos como forma de auxiliar no planejamento das atividades e redução das rotas das equipes. Na Figura 5, é apresentado um esquema relacionando as principais variáveis (ponto de partida, ponto de chegada, equipe, talhão e parcela) do problema de roteamento de equipes no planejamento de inventários florestais. No esquema ilustrado (Figura 5), existe um ponto de partida

e chegada das equipes, representado pela fábrica. Cada equipe, representada por um veículo, deve sair do ponto de origem, percorrer os talhões e retornar ao ponto de destino no final do dia.

Figura 5. Representação das variáveis envolvidas no PRV-IF

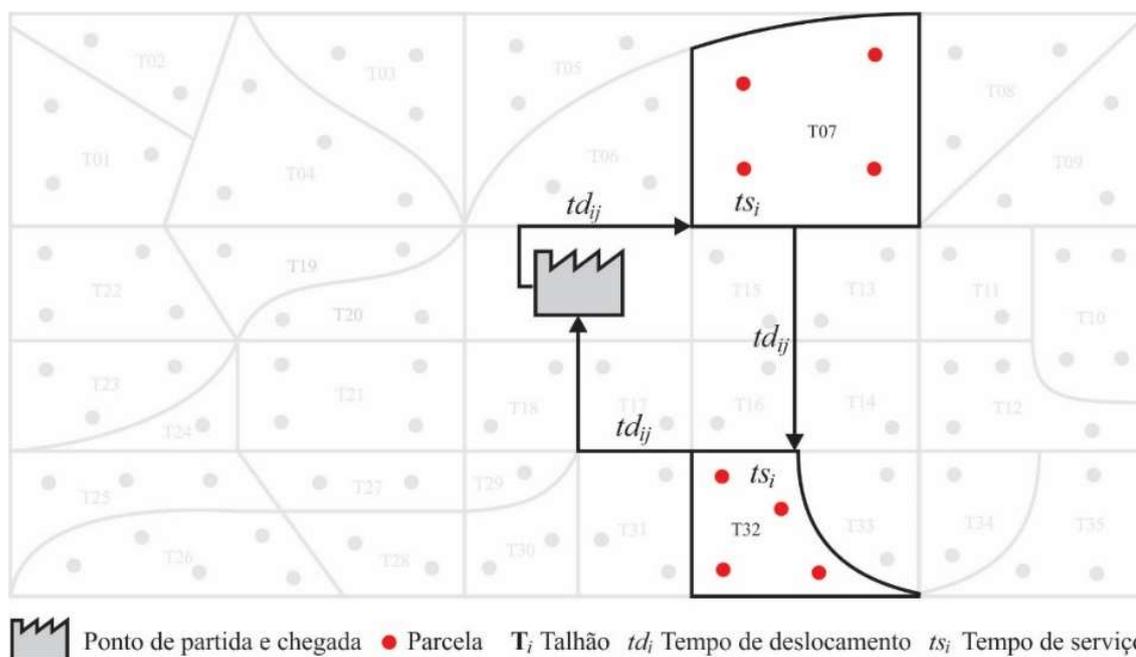


Fonte: O autor.

Na programação das atividades, deve-se considerar um horizonte  $H$  de  $n$  dias. Para cada dia, define-se uma sequência (rota) de talhões ( $T_i$ ) que deve ser inventariado por uma equipe  $k$ . O tempo de cada rota (Tempo Total de Serviço - TTS) deve ser inferior à Janela de Tempo Diária (JTD), considerada nesta pesquisa como 8 horas de trabalho diário, ou seja, 480 minutos. Desta forma, o Tempo Total de Serviço (TTS) de uma rota corresponde à soma do Tempo Total de Medição (TTM) com o Tempo Total de Deslocamento (TTD). Sendo TTD a soma dos deslocamentos entre  $i$  e  $j$  ( $td_{ij}$ ) e TTM a soma do tempo de medição de cada parcela ( $ts_i$ ). A Figura 6 ilustra a representação destas variáveis.

A definição de qual talhão incorporar na programação pode ser determinada por variáveis como: idade do plantio, idade de corte, data da última medição ou por uma solicitação independente das anteriores. Uma vez definido qual o talhão, define-se qual tipo de atividade deve ser realizada. De acordo com Meneguzzi (2011), essas atividades distinguem-se em inventário florestal qualitativo de 6 e 12 meses, inventário florestal contínuo de instalação e remedição, inventário pré-corte e *check-plots*.

Figura 6. Representação de uma rota diária estabelecida para uma equipe



Fonte: O autor.

Cada atividade apresenta uma flexibilidade no período de medição (Tabela 1), significando que cada talhão poderá ser medido dentro de um determinado período de meses. Essa flexibilidade possibilita um melhor arranjo da programação anual, pois permite que atividades localizadas próximas, mas que apresentam data de medição com diferença de poucos meses, possam ser programadas conjuntamente, evitando o retorno das equipes ao mesmo local em períodos subsequentes, reduzindo assim o custo com deslocamento.

Tabela 1. Flexibilidade de data para medição das atividades de inventário florestal

<b>Sigla</b>	<b>Atividade</b>				<b>Flexibilidade</b>
IFQ_6	Qualitativo 6 meses				Medido na quinzena em que completa 6 meses
IFQ_12	Qualitativo 12 meses				1 mês antes até 1 mês após completar 12 meses
IFC_I	Inventário	Florestal	Contínuo	–	1 ano após o IFQ12
	instalação				
IFC_Rn	Inventário	Florestal	Contínuo	– n <sup>a</sup>	1 ano após o IFCi ou 1 ano após o IFCrn
	remedição				
IPC	Inventário Pré-corte				Até 3 meses antes da data de corte
CP	Check-plots				Conforme solicitação do setor de pesquisa

Fonte: Adaptado de Meneguzzi (2011).

Uma rotação corresponde ao intervalo de tempo que decorre entre o plantio e o corte de um povoamento. Durante este período, as parcelas de um talhão devem

ser inventariadas várias vezes. O Quadro 3 representa um exemplo das atividades de inventário que deverão ocorrer em um plantio hipotético, instalado no mês de janeiro de um ano qualquer. A letra “X” no mês de janeiro do primeiro ano representa o plantio. Observa-se que, ao completar seis meses, o primeiro inventário é realizado (IFQ\_6). O inventário IFQ\_12 deve ocorrer entre o 11º e o 13º mês de vida do talhão. O período entre 11º e o 13º mês é o que denominamos Janela de Tempo (JT), isto é, período de flexibilidade para que o IFQ\_12 seja realizado. A data de realização do IFQ\_12 determina a provável data de realização do IFC\_I, que deverá ocorrer entre o 24º e 28º mês de plantio (Janela de Tempo para o caso do IFC\_I). A data de realização do IFC\_I determina a provável data de realização do IFC\_R1, e assim sucessivamente.

Quadro 3. Distribuição das parcelas de acordo com o número de medições e idade

ANO	JAN	FEV	MAR	ABR	MAI	JUN	JUL	AGO	SET	OUT	NOV	DEZ
1	X						IFQ_6				IFQ_12	IFQ_12
2	IFQ_12											IFC_I
3	IFC_I	IFC_I								IFC_R1	IFC_R1	IFC_R1
4	IFC_R1									IFC_R2	IFC_R2	IFC_R2
5	IFC_R2									IFC_R3	IFC_R3	IFC_R3
6	IFC_R3									IFC_R4	IFC_R4	IFC_R4
7	IFC_R4									IFC_R5	IFC_R5	IFC_R5

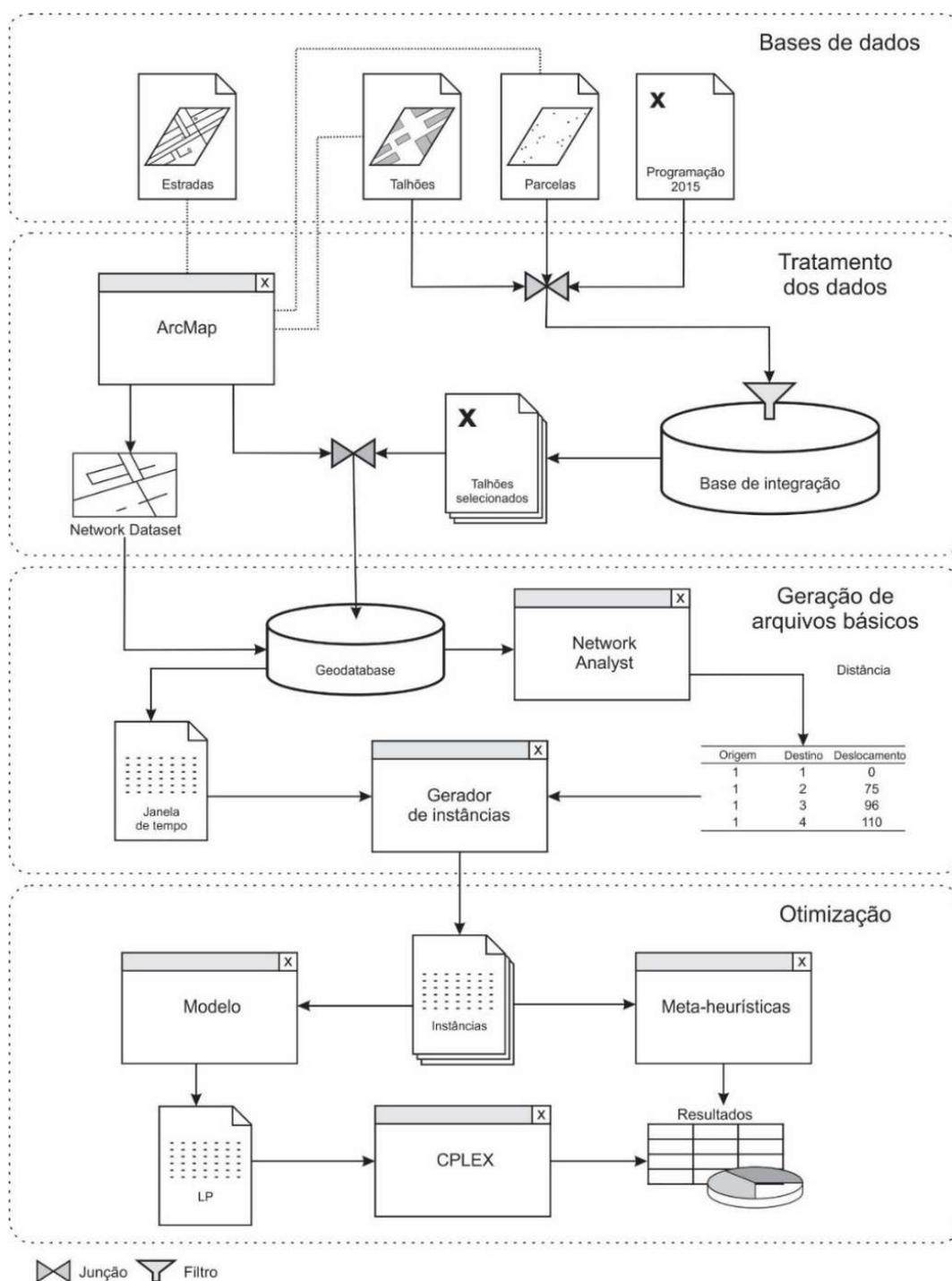
Fonte: O autor.

Em cada tipo de inventário são realizados procedimentos diversos. No Inventário Florestal Qualitativo (IFQ) é adotado um procedimento para avaliar o padrão de desenvolvimento e a qualidade da floresta no primeiro ano de vida. O Inventário Florestal Contínuo (IFC) é um procedimento utilizado para estimar o estoque presente (volume e número de árvores) e para monitorar o crescimento dos povoamentos. Compõe uma base de dados que pode ser utilizada tanto para análises qualitativas e quantitativas da floresta, quanto para ajuste de modelos de crescimento e produção. O Inventário Pré-corte (IPC) é um procedimento adotado para estimar o volume individual das árvores e o volume por unidade de área (hectare) dos povoamentos destinados à colheita. E por fim, o *Check-plots* (CP) é um procedimento utilizado para monitorar o crescimento dos povoamentos, essencialmente para fins de pesquisa.

#### 4. MATERIAL E MÉTODOS

Os passos metodológicos (Figura 7) necessários para realizar o planejamento do inventário florestal utilizando o modelo de roteamento de veículos foram: Bases de dados, Tratamento dos dados, Geração de arquivos básicos e Otimização.

Figura 7. Diagrama metodológico para geração de rotas para atividades de inventário florestal



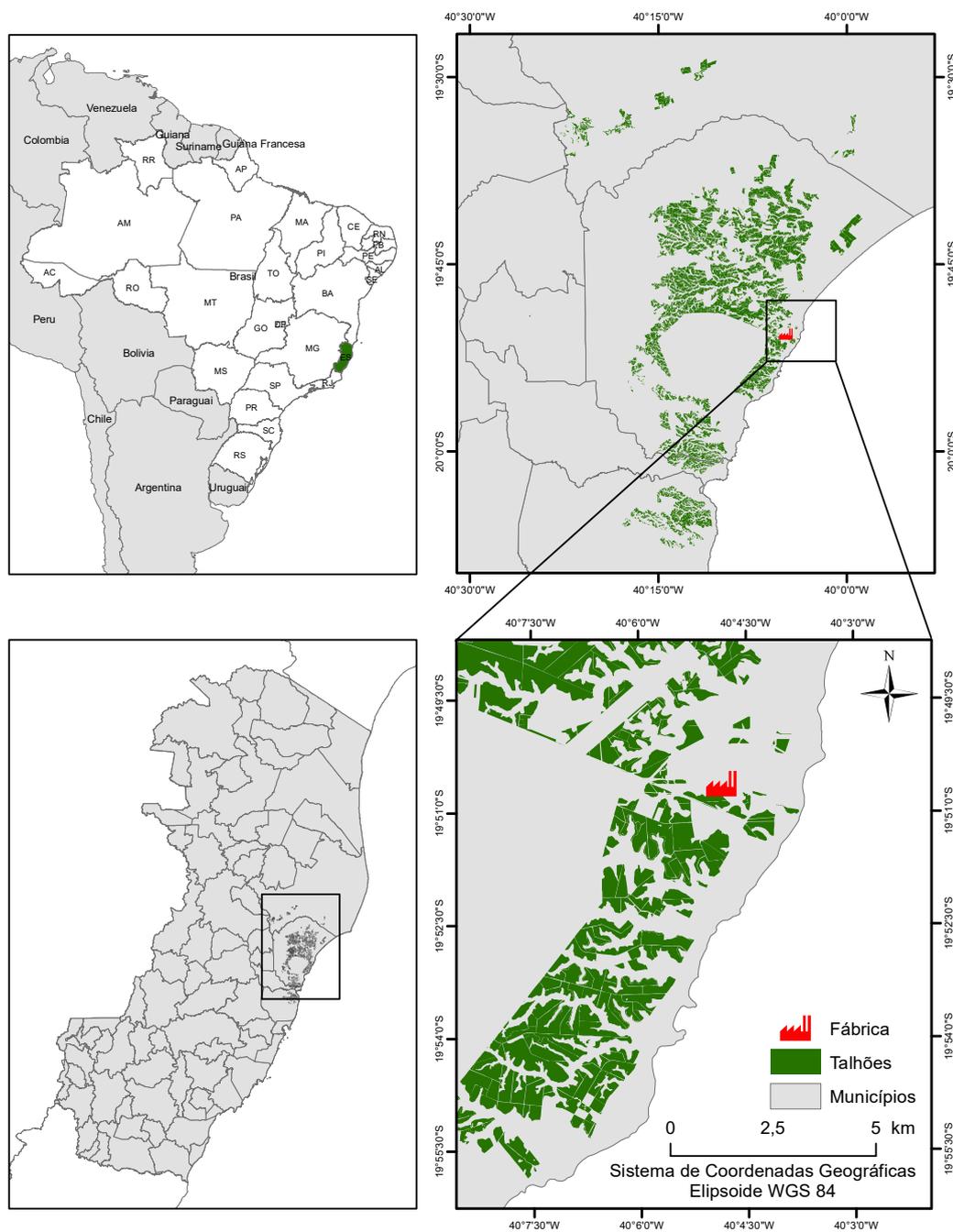
Fonte: O autor.

Na sequência são apresentados detalhes de cada um dos passos mencionados no fluxograma da Figura 7.

#### 4.1. Base de Dados

O presente estudo foi realizado com dados obtidos do planejamento do inventário florestal da empresa Fibria Celulose S.A., especificamente com informações da regional Aracruz (Figura 8).

Figura 8. Mapa de localização da região estudada



A regional faz parte da unidade Aracruz, a qual compreende os estados do Espírito Santo, Bahia e Minas Gerais. A regional Aracruz, situada no estado do Espírito Santo, corresponde a 15% do total de área florestal da unidade em questão, totalizando 31.513 hectares distribuídos entre os municípios de Aracruz, Fundão, Linhares, Ibatiba, Santa Teresa e Serra.

Fazem parte dos dados fornecidos, três arquivos, no formato *shapefile*, que contêm informações sobre estradas (linear), talhões (poligonais) e parcelas (pontos). Além destes arquivos, foram também fornecidos pela empresa dados no formato de planilha, que correspondem à programação do inventário florestal realizado no ano de 2015.

## 4.2. Tratamento dos dados

Considerando a base de dados descrita na Seção 4.1, foi realizado um pré-processamento para a composição de uma nova base de dados a ser empregada como *input* para o modelo de otimização. Essa base foi organizada considerando informações geográficas para gerar um novo conjunto de feições e de redes. Para modelar a rede de transportes, foi criado um *Network Dataset* que serve de base para o *ArcGIS Network Analyst*. Nessa etapa, utilizou-se o *software ArcGIS 10.1* (ESRI, 2016).

Os dados disponíveis nos arquivos *shapefile* (Talhões e Parcelas) foram integrados com dados obtidos na planilha de programação (Programação 2015). O objetivo desta junção foi estabelecer uma intercessão entre esses dados uma vez que eram necessárias informações da programação para definir os intervalos de janela de tempo (períodos em que a equipe pode realizar o inventário em um determinado talhão). Essa integração, bem como os filtros, foi realizada no *software MySQL 6.0*, denominada na Figura 7 de “Base de integração”.

A “Base de integração” possibilitou organizar os dados quanto ao tipo de inventário (Tabela 2), que neste trabalho foram considerados IFQ\_6, IFQ\_12, IFC\_I e IFQ\_R. Para definir o tipo de inventário, a considerar o planejamento de 2015, utilizou-se a data de plantio, disponível no arquivo Programação 2015.

Uma vez definidos os tipos de inventário, foram estabelecidas as Janelas de Tempo (JT) para cada talhão de acordo com a última medição realizada. Para o leitor compreender o que significa Janelas de Tempo, é útil consultar os exemplos apresentados no Capítulo 3. Os tamanhos das JTs considerados neste trabalho, informação esta fornecida pela empresa, são apresentados na Tabela 2. Admitindo-

se que, para se definir as Janelas de Tempo, é importante saber a data da última medição. No caso dos inventários IFQ\_6 e IFQ\_12, a data da última medição foi considerada como a data de plantio. Para o inventário IFC\_I, a data da última medição foi considerada como a data de realização do IFQ\_12. Por fim, a data da última medição considerada para o IFC\_R<sub>i</sub> foi a data de realização do IFC\_I ou a data de realização do IFC\_R<sub>i-1</sub> (Tabela 2).

Tabela 2. Tipos de inventários considerados

Tipo	Atividade	Flexibilidade
IFQ_6	Qualitativo 6 meses	Deve ser medido no mês em que completa 6 meses
IFQ_12	Qualitativo 12 meses	Até 1 mês antes até 1 mês após completar 12 meses
IFC_I	Contínuo implantação	Até 1 mês antes ou depois de completar 12 meses de realização do IFQ_12
IFC_R	Contínuo remedição	Até 1 mês antes ou depois de completar 12 meses de realização do IFC_I ou do (IFC_R <sub>i-1</sub> )*.

\*O IFC\_R<sub>i-1</sub> diz respeito ao fato que podem ocorrer várias remediações ao longo da rotação. Neste caso,  $i=2, 3, \dots, n-1$ , sendo  $n$  o momento da última remedição.

Para implementar o modelo, foi proposto um período de planejamento de um ano, escolhendo-se o ano de 2015 por sugestão da empresa. A unidade de tempo foi o dia, começando em 1º de janeiro de 2015 e encerrando em 31 de dezembro de 2015. Ocorre que, no 1º dia do horizonte de planejamento (1º de janeiro de 2015), a base de dados traz consigo povoamentos florestais que já existiam antes de 2015. Criou-se então uma classificação, com base na Tabela 3, para se definir qual o tipo de inventário seria realizado em cada povoamento pré-existente.

A título de exemplo e com base na Tabela 3, povoamentos que foram implantados entre 01/07/2014 e 30/06/2015 terão que se submeter ao inventário IFQ\_6. O mesmo raciocínio vale para os demais tipos de inventário da Tabela 3.

Tabela 3. Períodos considerados para definição da Janela de Tempo

Tipo	Período de Plantio		Nº de Registros
	Data Inicial	Data Final	
IFQ_6	01/07/2014	30/06/2015	715
IFQ_12	01/12/2013	31/01/2015	803
IFC_I	01/12/2012	30/11/2013	690
IFC_R	...	30/11/2012	924
<b>Total</b>			<b>3132</b>

Embora o arquivo de Programação dispusesse de 3132 medições para o ano de 2015, não necessariamente talhões, visto que os inventários IFQ\_6 e IFQ\_12 podem ocorrer em um mesmo ano, a intercessão dos talhões disponíveis nesta planilha com os talhões disponíveis nos arquivos do ArcGIS resultou em um

quantitativo de 2016 medições. A intercessão (junção) entre estes arquivos foi realizada por meio do código de uso do solo, que é comum em todas as bases.

Além da junção, também foram aplicados filtros na base de dados, como por exemplo, remoção de medições cujo código de uso de solo fosse repetido, exceto para IFQ\_6 e IFQ\_12. Também foram considerados apenas talhões com quantidade de parcelas superior ou igual a 0 (zero).

Uma vez definidas as medições, foi criado um atributo capaz de representar o tempo de serviço de cada medição. Neste, foi considerado o tempo médio de medição de cada parcela, isto é, para IFQ\_6 utilizou-se o tempo médio de 9 minutos e para os demais tipos de inventário, o tempo médio considerado foi de 18 minutos. O tempo médio de caminhamento entre parcelas foi de 7 minutos para qualquer tipo de inventário. Desta forma, para obter o tempo médio de cada medição, basta multiplicar a quantidade de parcelas pelo tempo médio de medição somado ao tempo médio de caminhamento.

De posse de todas as informações integradas na base de dados “Base de integração”, foram realizadas consultas, a fim de definir diferentes instâncias para o problema. Neste caso, foram selecionados oito arquivos contendo 24, 50, 100, 200, 400, 500, 1000 e 2016 medições. Para cada arquivo, foi realizada uma integração (junção) com os Talhões disponíveis no ArcGIS. O objetivo desta integração é exportá-los para um *Geodatabase* e posteriormente gerar as matrizes de deslocamento.

### **4.3. Geração dos arquivos básicos**

Nesta etapa, os arquivos foram novamente incorporados ao *ArqGIS*. Porém, o arquivo *shapfile* de talhões disponível passou a ter informações inerentes da programação, como período de medição e tempo de serviço. Talhões cuja programação previa duas visitas no mesmo ano foram duplicados considerando o tipo de inventário. Também foi adicionado a cada um dos arquivos o ponto que representa a Fábrica, que foi definida no modelo como garagem.

Considerando que cada talhão disponibiliza uma ou mais parcelas, e que foi necessário definir uma delas como ponto de chegada dos veículos, optou-se por escolher a parcela de menor código. Para cada uma destas parcelas, gerou-se um ponto vizinho na estrada, definido como ponto de acesso à parcela. Esse ponto foi fundamental para que fossem geradas as matrizes de deslocamento, visto que o ponto referente à parcela se localiza fora da estrada e dentro do polígono talhão.

Todos estes dados foram incorporados em um *Geodatabase*, que serviu de base para a ferramenta *Network Analyst* gerar as matrizes de deslocamento. Uma Matriz de Deslocamento contém para cada par origem (*i*) destino (*j*), o tempo de deslocamento, em minutos, entre *i* e *j*. O tempo médio de deslocamento estabelecido para as equipes foi de 30 km/h. Embora existam várias possibilidades de rotas entre os talhões, foi considerada a menor rota possível.

Uma vez geradas as matrizes de deslocamento para cada instância, também foram exportados do *ArcGIS* os arquivos referentes à janela de tempo. Este último contém para cada talhão o mês inicial e final em que a medição deve ser realizada. Tanto a matriz de deslocamento quanto o arquivo de janela de tempo servem como base para uma ferramenta desenvolvida para gerar as instâncias do problema. Um exemplo de instância para um problema com 24 pontos e 6 dias de planejamento é representado na Figura 9.

Figura 9. Exemplo de arquivo de instância para um problema com 24 pontos e 6 dias de planejamento

```

6
26
0.002.587.915.335.164.998.084.647.911.727.228.431.723.445.503.102.244.823.105.163.617.575.685.684.470.00
2.580.004.991.723.275.686.364.992.752.063.962.242.242.067.223.613.443.106.714.133.105.852.243.108.082.58
7.914.990.007.052.582.246.366.884.305.337.575.686.714.477.917.225.331.892.588.268.082.753.965.165.857.91
5.331.727.050.002.061.892.416.716.192.756.711.728.085.852.752.755.682.067.572.248.436.887.916.886.195.33
5.163.272.582.060.005.857.917.057.052.754.992.928.086.363.611.894.647.916.361.725.337.574.826.193.965.16
4.995.682.241.895.850.007.573.274.997.743.274.137.915.507.744.138.268.264.134.996.197.913.442.756.194.99
8.086.366.362.417.917.570.001.726.717.053.966.193.787.744.303.963.788.085.681.727.577.226.882.924.308.08
4.644.996.886.717.053.271.720.002.248.086.196.024.138.086.715.505.167.911.723.442.065.685.684.825.684.64
7.912.754.306.197.054.996.712.240.002.243.784.646.544.643.613.788.262.065.686.193.445.506.366.884.477.91
1.722.065.332.752.757.747.058.082.240.004.992.414.302.245.166.718.433.105.852.063.618.086.714.822.241.72
7.223.967.576.714.993.273.966.193.784.990.002.417.914.303.615.688.267.056.022.755.853.105.855.686.367.22
8.432.245.681.722.924.136.196.024.642.412.410.002.925.162.753.104.826.884.136.192.924.995.502.243.788.43
1.722.246.718.088.087.913.784.136.544.307.912.920.003.107.052.411.897.222.588.434.993.446.542.922.061.72
3.442.064.475.856.365.507.748.084.642.244.305.163.100.002.415.334.996.716.714.642.756.547.576.717.743.44
5.507.227.912.753.617.744.306.713.615.163.612.757.052.410.004.138.434.137.055.165.854.302.925.335.855.50
3.103.617.222.751.894.133.965.503.786.715.683.102.415.334.130.007.407.917.911.894.993.781.723.963.443.10
2.243.445.335.684.648.263.785.168.268.438.264.821.894.998.437.400.004.997.575.506.194.827.911.895.332.24
4.823.101.892.067.918.268.087.912.063.107.056.887.226.714.137.914.990.005.505.851.722.416.542.755.684.82
3.106.712.587.576.364.135.681.725.685.856.024.132.586.717.057.917.575.500.003.787.914.827.056.362.243.10
5.164.138.262.241.724.991.723.446.192.062.756.198.434.645.161.895.505.853.780.003.967.574.996.884.995.16
3.613.108.088.435.336.197.572.063.443.615.852.924.992.755.854.996.191.727.913.960.005.508.083.614.133.61
7.575.852.756.887.577.917.225.685.508.083.104.993.446.544.303.784.822.414.827.575.500.006.882.064.827.57
5.682.243.967.914.823.446.885.686.366.715.855.506.547.572.921.727.916.547.054.998.086.880.007.406.545.68
5.683.105.166.886.192.752.924.826.884.825.682.242.926.715.333.961.892.756.366.883.612.067.400.008.265.68
4.478.085.856.193.966.194.305.684.472.246.363.782.067.745.853.445.335.682.244.994.134.826.548.260.004.47
0.002.587.915.335.164.998.084.647.911.727.228.431.723.445.503.102.244.823.105.163.617.575.685.684.470.00
111111
000001
000010
000111
001111
011111
000001
111110
111100
000001
000001
000011
000100
000001
011000
011000
011111
000010
000010
010000
000111
001100
000110
001110
011111
111111
025251002520012510050125175501752517512525175250200200252252502500

```

No exemplo da Figura 9, o primeiro valor 6 representa o horizonte, ou seja, o tempo total em que as medições devem ser realizadas. O número seguinte, 26, representa a quantidade de medições (24) acrescida de dois pontos que representam as garagens. Embora seja uma garagem, considerou-se duas, visto que o veículo deve sair e retornar à garagem ao final do dia. A matriz de deslocamento vem em seguida ao número de medições. Trata-se de uma matriz quadrada com os tempos de deslocamentos entre os 26 pontos do problema. A matriz binária, abaixo da matriz de deslocamento, representa a janela de tempo. A linha que representa a garagem contém apenas valores iguais a um. Isso significa que os veículos devem visitar esse ponto todos os dias do planejamento. A segunda linha, que representa o ponto de número um, indica que apenas no sexto dia a janela está aberta para medição. Por último, a matriz de tempo de medição que, para o ponto zero (garagem) o tempo de medição é zero e para o primeiro ponto, o tempo de medição é de 25 minutos e assim sucessivamente.

#### **4.4. Otimização**

Os arquivos de instâncias gerados na etapa anterior servem de base para os sistemas desenvolvidos para otimização. Foram desenvolvidos dois sistemas com objetivo de avaliar o comportamento do modelo quando utilizado um método exato e quando utilizado uma meta-heurística. O sistema Modelo é capaz de gerar arquivos de entrada (LPs) para o *software* de otimização CPLEX que utiliza um método exato como mecanismo de encontrar uma solução. No sistema de meta-heurísticas, foram desenvolvidos o ILS e o SA, os quais têm como objetivo encontrar soluções viáveis para o problema de roteamento do planejamento de inventários florestais.

Nas subseções 4.4.1 e 4.4.2, são detalhados o modelo e os métodos de solução propostos nesta pesquisa.

##### **4.4.1. Formulação do Modelo Matemático**

Neste trabalho, utilizou-se um modelo adaptado do modelo proposto por Cordeau et al. (2001) e citado por Nguyen et al. (2014), para o Problema de Roteamento Periódico de Veículos com Janela de Tempo (PRPVJT). O PRPVJT é uma variação do PRVJT. Nesse modelo, foram realizadas adaptações quanto ao uso de uma única garagem (todos os veículos começam e terminam sua rota na mesma garagem), conforme o trabalho de Mauri (2008). Para melhor representar o

Planejamento de Inventários Florestais, os termos “local” e “veículo” são substituídos por “talhão” e “equipe”, respectivamente.

O modelo proposto para representar o problema de roteamento do planejamento de inventários florestais é descrito a seguir. Os nós  $Go$  e  $Gd$  representam, respectivamente, a base de saída e de chegada das equipes na fábrica (garagem). A cada um destes dois nós estão associados os tempos de medição nulos e as janelas de tempo que indicam os dias permitidos de saída e de chegada das equipes à fábrica.

### Minimizar

$$\sum_{t \in T} \sum_{k \in K} \sum_{i \in \{L_t \cup Go\}} \sum_{j \in \{L_t \cup Gd\}} d_{i,j} x_{i,j,k}^t \quad \forall i \neq j; i = Go \mid j \neq Gd; j = Gd \mid i \neq Go \quad (1)$$

### Sujeito a:

$$\sum_{t \in T} \sum_{k \in K} \sum_{j \in \{L_t \cup Gd\}} x_{i,j,k}^t = 1 \quad \forall i \neq j; i = Go \mid j \neq Gd; j = Gd \mid i \neq Go; i \in L \quad (2)$$

$$\sum_{j \in L_t} x_{Go,j,k}^t = 1 \quad \forall k \in K; t \in T \quad (3)$$

$$\sum_{j \in L_t} x_{j,Gd,k}^t = 1 \quad \forall k \in K; t \in T \quad (4)$$

$$\sum_{j \in \{L_t \cup Gd\}} x_{i,j,k}^t - \sum_{i \in L_t} x_{j,i,k}^t = 0 \quad \forall i \neq j; i = Go \mid j \neq Gd; j = Gd \mid i \neq Go; k \in K; t \in T \quad (5)$$

$$H_j \geq (H_{Go,k}^t + d_{Go,j}) + M(1 - x_{Go,j,k}^t) \quad \forall k \in K; t \in T; j \in L_t \quad (6)$$

$$H_j \geq (H_i + s_i + d_{i,j}) + M(1 - x_{i,j,k}^t) \quad \forall i \neq j; i = Go \mid j \neq Gd; j = Gd \mid i \neq Go; i \in L_t; j \in \{L_t \cup Gd\} \quad (7)$$

$$H_{Gd,k}^t \geq (H_i + s_i + d_{i,Gd}) x_{i,Gd,k}^t \quad \forall k \in K; t \in T; i \in L_t \quad (8)$$

$$H_{Go,k}^t = a \quad \forall k \in K; t \in T \quad (9)$$

$$H_{Gd,k}^t \leq b \quad \forall k \in K; t \in T \quad (10)$$

$$x_{i,j,k}^t \in \{0,1\} \quad \forall i \neq j; i = Go \mid j \neq Gd; j = Gd \mid i \neq Go; k \in K; t \in T; i \in \{L_t \cup Go\}; j \in \{L_t \cup Gd\} \quad (11)$$

em que:

$i$  = talhão de origem;

$j$  = talhão de destino;

$Go$  = garagem de origem (representa a fábrica);

$Gd$  = garagem de destino (representa a fábrica);

$d_{i,j}$  = tempo de deslocamento entre os talhões  $i$  e  $j$ ;

$d_{Go,j}$  = tempo de deslocamento entre a garagem e o primeiro talhão a ser visitado;

$d_{i,Gd}$  = tempo de deslocamento entre o último talhão visitado e a garagem;

$X_{i,j,k}^t = 1$  caso o arco  $(i,j)$  for escolhido para integrar a solução para a equipe  $k$  no período  $t$ , e 0 caso contrário;

$T$  = horizonte de planejamento;

$K$  = conjunto de equipes disponíveis;

$L_t$  = lista de talhões (locais) a serem visitados em um período  $t$ ;

$s_i$  = tempo de serviço no nó  $i$ ;

$H_j$  = instante de início do atendimento no talhão  $j$ ;

$H_i$  = instante de início do atendimento no talhão  $i$ ;

$H_{Go,k}^t$  = horário de saída do veículo  $k$  no período  $t$ ;

$H_{Gd,k}^t$  = horário de chegada do veículo  $k$  no período  $t$ ;

$a$  = início do período de atendimento; e

$b$  = término do período de atendimento.

A função objetivo (1) visa minimizar o deslocamento das equipes de inventário entre talhões. As restrições (2) garantem que cada talhão seja visitado uma única vez, por uma única equipe em um determinado período de tempo. As restrições (3) garantem que exatamente  $K$  equipes saíam da garagem, enquanto as restrições (4) asseguram que  $K$  equipes cheguem à garagem. Também chamada de “contenção de fluxo”, as restrições (5) garantem que toda equipe que chega a um talhão, deve partir dele para visitar o próximo. As restrições (6) calculam os horários de chegada para atendimento no primeiro talhão para cada equipe, após deixar a garagem. As restrições (7) garantem que o horário de início do atendimento do talhão  $j$  seja superior ao tempo de serviço no talhão  $i$  somado ao deslocamento entre  $i$  e  $j$ . As restrições (8) calculam os horários de retorno à garagem. As restrições (9) e (10) asseguram que os horários de atendimento ocorram dentro da janela de tempo diária, isto é, entre 0 e 480 minutos. Por fim, as restrições (11) asseguram que as variáveis de decisão sejam binárias.

Sempre que uma solução violar uma das restrições, utiliza-se a técnica de penalização na Função Objetivo a fim não permitir que soluções inviáveis sejam aceitas. Seja  $f(s)$  um valor de penalização  $p_i$  (para problemas de minimização) sempre que este violar um restrição  $r_i$ , sendo cada penalização  $p_i$  referente a uma restrição  $r_i$  violada. As penalizações  $p_i$  aplicáveis a  $f(s)$  são definidas por  $p_1$ ,  $p_2$  e  $p_3$ .

A penalização  $p_1$  é aplicada sempre que algum ponto presente na solução  $s$  for atendido fora de sua janela de tempo previamente estabelecida. Sempre que uma equipe não realizar atendimento em determinado dia, ou quando o tempo de deslocamento somado ao tempo de serviço da equipe for superior aos 480 minutos diários, atribuem-se a  $f(s)$  as penalizações  $p_2$  e  $p_3$ , respectivamente.

#### 4.4.2. Métodos de solução

Nesta pesquisa, são utilizadas as meta-heurísticas ILS e SA para resolver o modelo apresentado na seção 4.4.1. Nas subseções 4.4.2.1, 4.4.2.2, 4.4.2.3 e 4.4.2.4, são descritas informações comuns às meta-heurísticas implementadas.

##### 4.4.2.1. Representação de uma solução

A solução gerada pelas meta-heurísticas considera os seguintes elementos: horizonte de planejamento, número de equipes número de talhões visitados em um determinado dia. Desta forma, a solução é representada por uma matriz de  $n$  linhas por  $m$  colunas, sendo  $n = \text{número de dias } (H) \times \text{número de equipes } (nk)$  e  $m$ , o número máximo de talhões que podem ser visitados por uma equipe em um dia.

A Figura 10 ilustra um exemplo de representação para um problema com horizonte de 3 dias, 2 equipes e 24 talhões. Observa-se que, cada linha da matriz representa uma rota que uma equipe ( $k$ ) realiza em um dia ( $t$ ). Portanto, a sequência de talhões que deverão ser medidos no dia 5 pela equipe 1 é: 4 – 11 – 20. Os pontos de origem e de destino estão implícitos em cada uma das rotas. Desta forma, no dia 5 a equipe 1 realiza a seguinte rota: 0 – 4 – 20 – 0, em que 0, representa o ponto de origem e de destino final.

Figura 10. Representação de uma solução

$t$	$k$	rotas				
1	1	8				
1	2	7				
2	1	15	19			
2	2	24				
3	1	16				
3	2	23	21	14		
4	1	12				
4	2	5	22			
5	1	4	11	20		
5	2	17	2	18		
6	1	13	10	6	3	1
6	2	9				

Fonte: O autor.

Cada talhão está associado a um tempo de serviço ( $ts_i$ ) que corresponde ao produto entre o tempo médio de medição de uma parcela e o número de parcelas. O tempo de deslocamento ( $td_{ij}$ ) entre os pontos (talhões) é dado pela matriz de distância. Desta forma, na rota referente ao dia 5 da equipe 1 o tempo total é dado pela soma dos tempos de serviços realizados em 4 e 20, acrescido dos tempos de deslocamento entre 0 – 4, 4 – 20 e 20 – 0.

#### 4.4.2.2. Construção da solução inicial

Devido às restrições impostas ao PRV-IF, a solução inicial é construída de maneira aleatória e gulosa, sempre com objetivo de gerar uma solução viável. Porém, em virtude das dificuldades em construir uma solução viável, esta pode ser gerada de maneira inviável, visto que, para cada restrição violada, penaliza-se a Função Objetivo (FO). Este procedimento é apresentado na Figura 11.

Figura 11. Pseudocódigo da construção da solução inicial

```

1.  CONSTRUTIVA(s)
2.  INÍCIO
3.      CRIAR (uma lista T com todos os talhões a serem visitados);
4.      PARA (cada rota l, l = 1,2,...,n, de s) FAÇA
5.          REMOVER (um ponto p de T, que seja compatível com a rota l);
6.          INSERIR (o ponto p na rota l);
7.      FIM-PARA;
8.      PARA (cada ponto p restante de T) FAÇA
9.          REMOVER (um ponto p de T);
10.         CRIAR (uma lista L com todas as rotas possíveis para o ponto p);
11.         PontoIn ← Falso;
12.         PARA (cada rota l de L) FAÇA
13.             SE (pode inserir o ponto p na rota l)
14.                 INSERIR (o ponto p na rota l);
15.                 PontoIn ← Verdadeiro;
16.             FIM-SE;
17.         FIM-PARA;
18.         SE (PontoIn = Falso)
19.             SELECIONAR (aleatoriamente uma rota l em n);
20.             INSERIR (o ponto p na rota l);
21.         FIM-SE;
22.     FIM-PARA;
23. FIM.

```

Fonte: O autor.

No pseudocódigo apresentado na Figura 11, na primeira iteração (passos 4 a 7), é realizada a inserção de um ponto em cada rota disponível. Essa inserção é importante, visto que há uma restrição no modelo que determina que todos os dias, todas as equipes devem sair da garagem. Na segunda iteração (passos 8 a 22), os pontos restantes, que não foram distribuídos na primeira iteração, são inseridos em rotas escolhidas de maneira restritiva. Essa restrição refere-se ao fato de serem escolhidas apenas rotas em que o ponto pode ser inserido quanto à janela de

tempo, e que após inserido, não viole a restrição de janela de tempo diária (480 minutos). Essa verificação é realizada no passo 13 do algoritmo. Caso não seja encontrada uma rota possível, o ponto é inserido em uma rota selecionada aleatoriamente (passos 18 a 21). Neste último caso, a Função Objetivo da solução será penalizada.

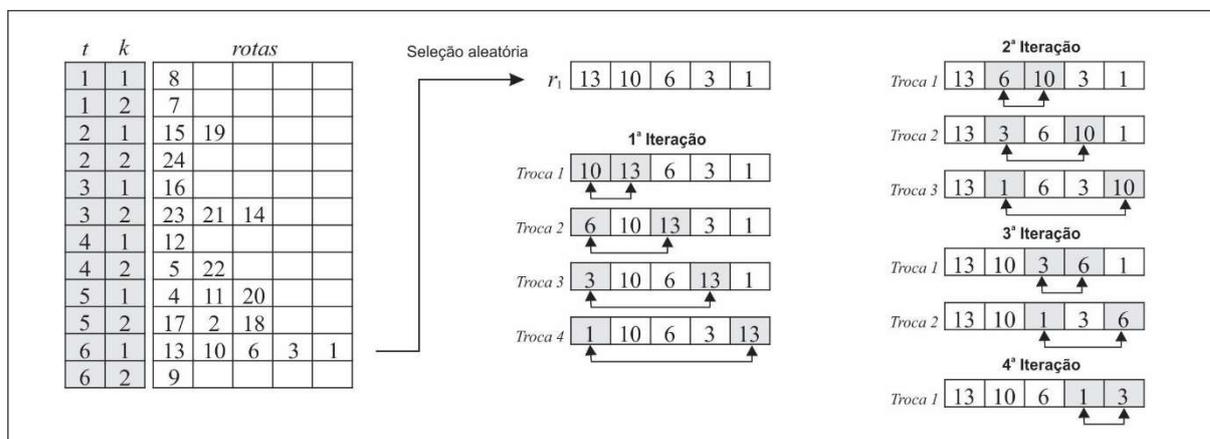
#### 4.4.2.3. Estratégias de busca local

As estratégias de busca local utilizadas foram: 1-optimal intra-rota (1-Opt) e Melhor Melhora (MM). Mais detalhes sobre estas estratégias são descritos a seguir.

##### 4.4.2.3.1.1-Optimal intra-rota

Este movimento seleciona aleatoriamente uma rota e realiza combinações entre os pontos vizinhos. Cada uma das combinações é gerada por meio de trocas entre dois pontos ( $i$  e  $j$ ). Desta forma, para cada ponto  $i$  são realizadas trocas entre os demais pontos correspondentes a  $j$ . A Figura 12 ilustra uma situação em que uma rota  $r_1$  é escolhida aleatoriamente, e nesta, constam os seguintes pontos a serem visitados: 13, 10, 6, 3 e 1. A cada iteração, um ponto  $i$  é fixado e a troca é realizada alternando os demais pontos aqui chamados de  $j$ . Na primeira iteração, o valor de  $i$  corresponde ao ponto 13 e este é trocado com os demais 10, 6, 3 e 1. Na segunda iteração, o valor fixado para  $i$  corresponde ao ponto 10, que também realiza a troca com os demais 6, 3 e 1. A cada troca, avalia-se o tempo de deslocamento da rota, e caso esta seja melhor que a rota inicial, substitui-se a melhor rota. Ao final de todas as iterações, permanece a melhor opção de rota para  $r_1$ .

Figura 12. Movimento 1-Optimal intra-rota

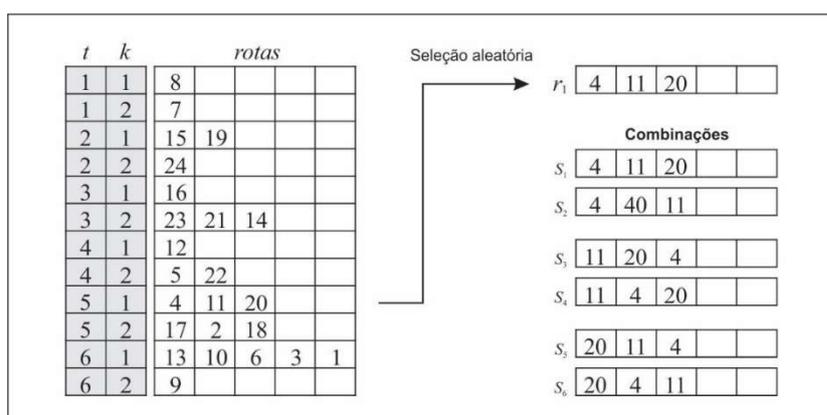


Fonte: O autor.

#### 4.4.2.3.2. Melhor Melhora (MM)

Este movimento seleciona aleatoriamente uma rota e realiza todas as combinações possíveis entre os pontos desta rota. No exemplo representado na Figura 13, após escolhida de maneira aleatória a rota  $r_1$ , outras várias opções de rotas são geradas a partir da combinação dos pontos de  $r_1$ . Uma vez verificada todas as combinações ( $ra_1, ra_2, \dots, ra_n$ ), o método escolhe aquela de menor tempo de deslocamento.

Figura 13. Movimento Melhor Melhora



Fonte: O autor.

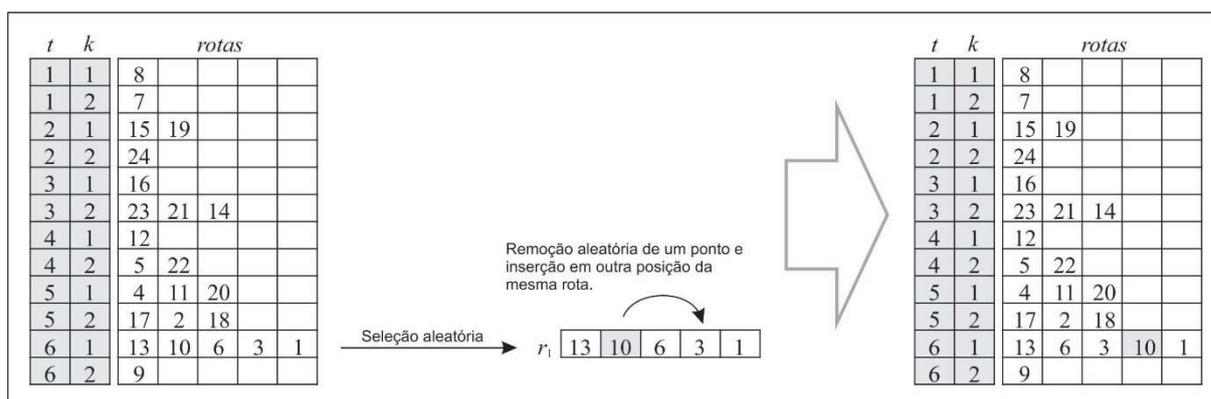
#### 4.4.2.4. Estratégias de perturbação

Com a finalidade de promover a fuga dos ótimos locais, nesta pesquisa foram implementados sete movimentos de perturbação classificados como movimentos intra-rota e entre-rotas.

##### 4.4.2.4.1. Remoção intra-rota

Neste movimento, o primeiro passo consiste em selecionar aleatoriamente uma rota. Na rota escolhida, seleciona-se um ponto para ser removido. O ponto removido é reinserido na mesma rota de maneira gulosa, ou seja, testando-o em cada posição para verificar o menor tempo de deslocamento. Na Figura 14, apresentado um exemplo deste tipo de movimento, em que, a rota selecionada  $r_1$ , dispõe dos seguintes pontos: 12, 10, 6, 3 e 1. Após a seleção da rota, o ponto 10 foi selecionado aleatoriamente e em seguida inserido em todas as posições da rota. O ponto permanece na posição que denotar o menor deslocamento para a rota.

Figura 14. Movimento de remoção intra-rota

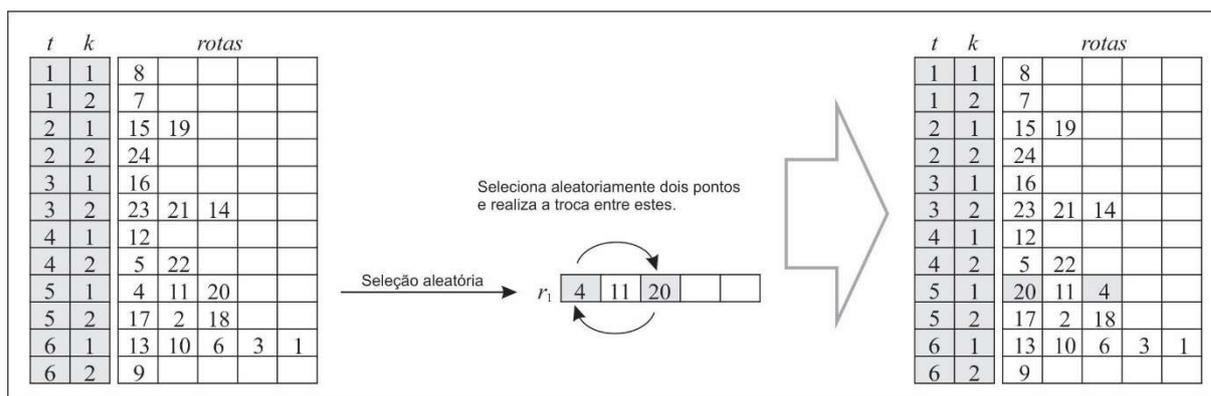


Fonte: O autor.

#### 4.4.2.4.2. Troca intra-rota

Neste tipo de movimento, primeiramente seleciona-se uma rota de maneira aleatória, e em seguida, seleciona-se dois pontos distintos, também de maneira aleatória. Após escolhidos as rotas e os pontos, realiza-se a troca entre os mesmos, conforme exemplificado na Figura 15. No exemplo, a rota  $r_1$  selecionada contém os pontos 4, 11 e 20. Posteriormente foram selecionados dois pontos (4 e 20), também de forma aleatória. Em seguida é realizada a troca destes pontos. Independente da melhoria no tempo de deslocamento, o método retorna a rota após realizada a perturbação.

Figura 15. Movimento de troca intra-rota



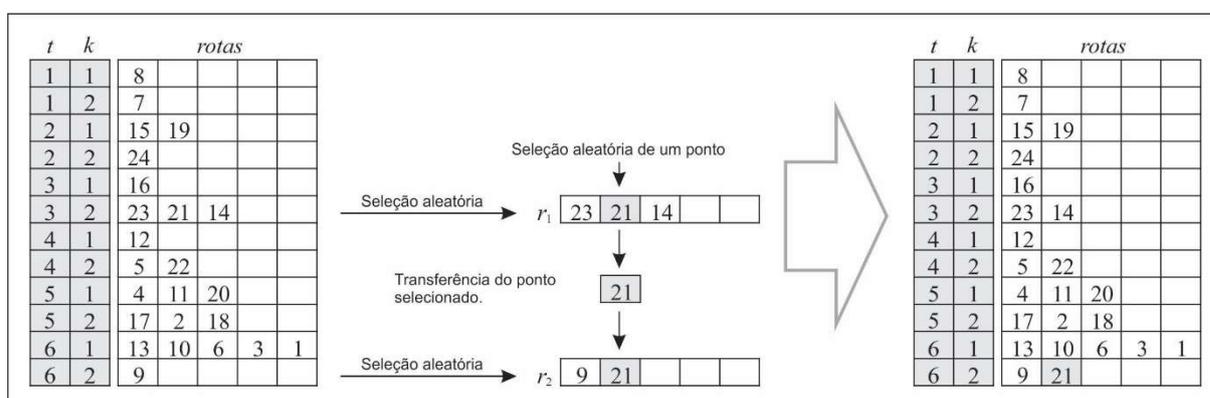
Fonte: O autor.

#### 4.4.2.4.3. Remoção entre rotas

Neste movimento, primeiramente seleciona-se aleatoriamente uma rota. Na rota escolhida, seleciona aleatoriamente um ponto para ser removido. O ponto removido é reinserido em uma rota escolhida de maneira aleatória, entre as rotas

possíveis. As rotas possíveis para um ponto são rotas que pertencem a períodos de janela de tempo estabelecidos para aquele ponto. A inserção deste ponto na nova rota também ocorre de maneira gulosa, ou seja, escolhe-se a melhor posição para sua inserção. No exemplo da Figura 16, o ponto 21 foi removido de maneira aleatória da rota  $r_1$ . Após a remoção, é escolhida uma nova rota para sua inserção, porém esta rota deve pertencer a um período de janela de tempo viável para o ponto 21. Suponhamos que a janela de tempo para o ponto 21 esteja compreendida entre os dias 3 e 6, o que possibilita a sua inserção em 8 rotas, visto que são 4 dias disponíveis ( $t$ ) e duas equipes ( $k$ ). Dentre as opções, o método escolheu aleatoriamente a rota  $r_2$  que corresponde ao dia 6, equipe 2. O ponto 21 é inserido nesta nova rota.

Figura 16. Movimento de remoção entre rotas

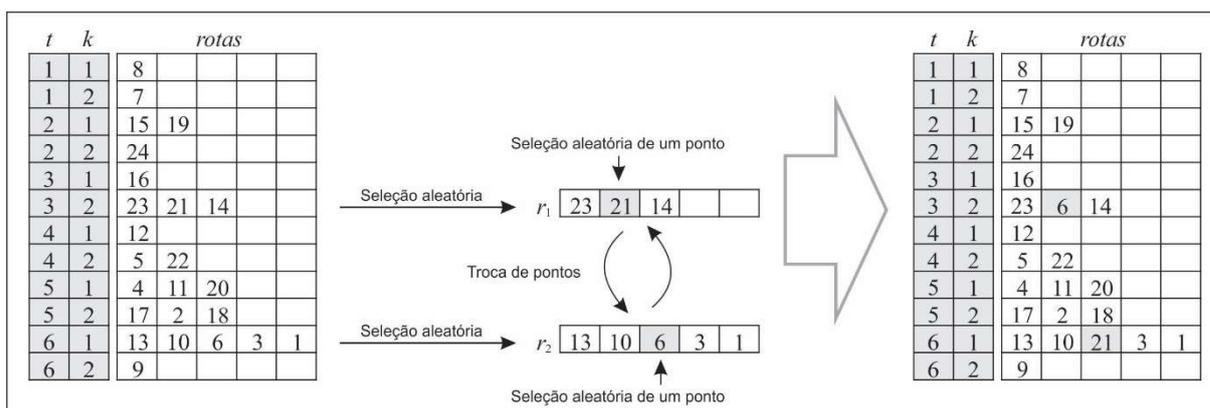


Fonte: O autor.

#### 4.4.2.4.4. Troca entre rotas

No movimento de troca entre rotas, duas rotas são selecionadas de forma aleatória e em cada uma das rotas seleciona-se um ponto também de forma aleatória. Os pontos selecionados são trocados entre estas rotas, como mostra a Figura 17. Observa-se que duas rotas são selecionadas:  $r_1$  e  $r_2$ . Na rota  $r_1$ , seleciona-se o ponto 21 e na rota  $r_2$ , seleciona-se o ponto 6. No passo seguinte, realiza-se a troca destes pontos entre as rotas.

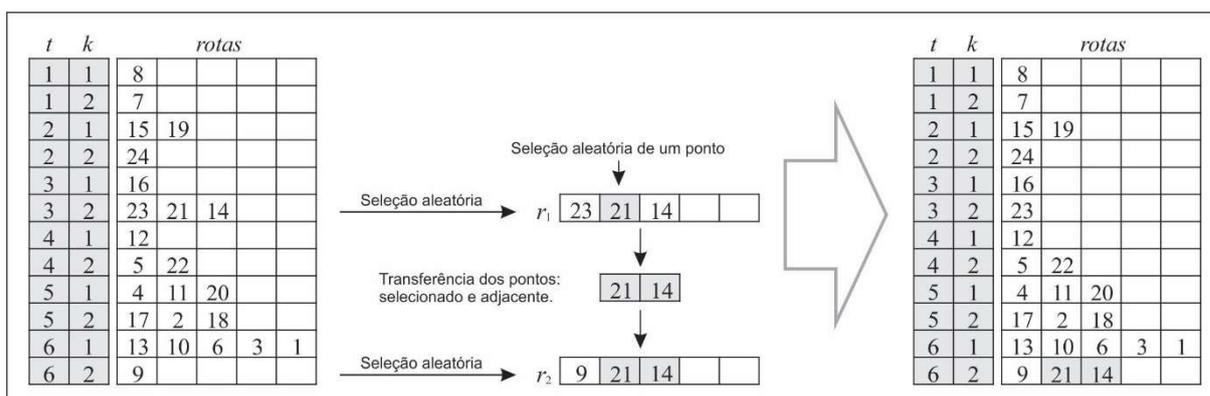
Figura 17. Movimento de troca entre rotas



Fonte: O autor.

#### 4.4.2.4.5. Shift (2,0)

Esse movimento consiste na transferência de dois pontos,  $p_1$  e  $p_2$ , de uma rota para outra. A Figura 18 ilustra este tipo de movimento, em que primeiramente, escolhe-se, de maneira aleatória, uma rota  $r_1$  e em seguida, também de forma aleatória, um ponto qualquer (21). São removidos dois pontos: o ponto escolhido (21) e seu adjacente da direita (14). Após a remoção, os pontos são reinseridos em outras rotas, também escolhidas de forma aleatória e que atenda a restrição de janela de tempo de cada ponto removido.

Figura 18. Movimento *shift* (2,0)

Fonte: O autor.

#### 4.4.2.4.6. Correção

Diferente dos movimentos descritos anteriormente, que são movimentos tradicionais do PRV, os movimentos de correção e de equilíbrio (descritos nas subseções 4.4.2.4.6 e 4.4.2.4.7) são movimentos desenvolvidos para o tipo de problema abordado nesta pesquisa. No primeiro movimento, denominado Correção,

o objetivo é tornar uma solução viável no que se refere à janela de tempo diária. Para que isso ocorra, pontos são removidos de rotas que violam a restrição de janela de tempo diária, e inseridos em rotas com tempo ocioso. A reinserção dos pontos é priorizada em rotas que não têm nenhum ponto, como forma de corrigir a restrição de que toda equipe deve sair da fábrica diariamente.

O pseudocódigo do movimento de correção é apresentado na Figura 19. Observa-se que, inicialmente são removidos pontos das rotas cujo tempo acumulado seja superior à janela de tempo diária (passos 4 ao 8). A janela de tempo diária é o tempo limite para desenvolvimento das atividades em um dia, que neste trabalho foi de 480 minutos. Os pontos são removidos por ordem de menor impacto, ou seja, pontos com menor tempo de medição.

Figura 19. Pseudocódigo do movimento de Correção

```

1.  CORREÇÃO (s)
2.  INÍCIO
3.      CRIAR (uma lista T com todos os talhões removidos);
4.      PARA (cada rota l, l = 1,2,...,n, de s) FAÇA
5.          ENQUANTO (tempo acumulado da rota l > janela diária) FAÇA
6.              REMOVER (um ponto p de menor impacto na rota l);
7.              INSERIR (o ponto p na lista T);
8.          FIM ENQUANTO;
9.      FIM-PARA;
10.     PARA (cada ponto p disponível em T > 0) FAÇA
11.         PontoIn ← Falso;
12.         PARA (cada rota l, l = 1,2,...,n, de s possível para p) FAÇA
13.             SE (tempo acumulado da rota l == 0)
14.                 REMOVER (o ponto p de T);
15.                 INSERIR (o ponto p na rota l);
16.                 PontoIn ← Verdadeiro;
17.             FIM-SE;
18.         FIM-PARA;
19.         SE (PontoIn = Falso)
20.             PARA (cada rota l, l = 1,2,...,n, de s possível para p) FAÇA
21.                 SE (tempo acumulado da rota l < tempo mínimo de ocupação)
22.                     REMOVER (o ponto p de T);
23.                     INSERIR (o ponto p na rota l);
24.                     PontoIn ← Verdadeiro;
25.                 FIM-SE;
26.             FIM-PARA;
27.         FIM-SE;
28.         SE (PontoIn = Falso)
29.             SELECIONAR (aleatoriamente uma rota l em n, possível para p);
30.             INSERIR (o ponto p na rota l);
31.         FIM-SE;
32.     FIM-PARA;
33. FIM.

```

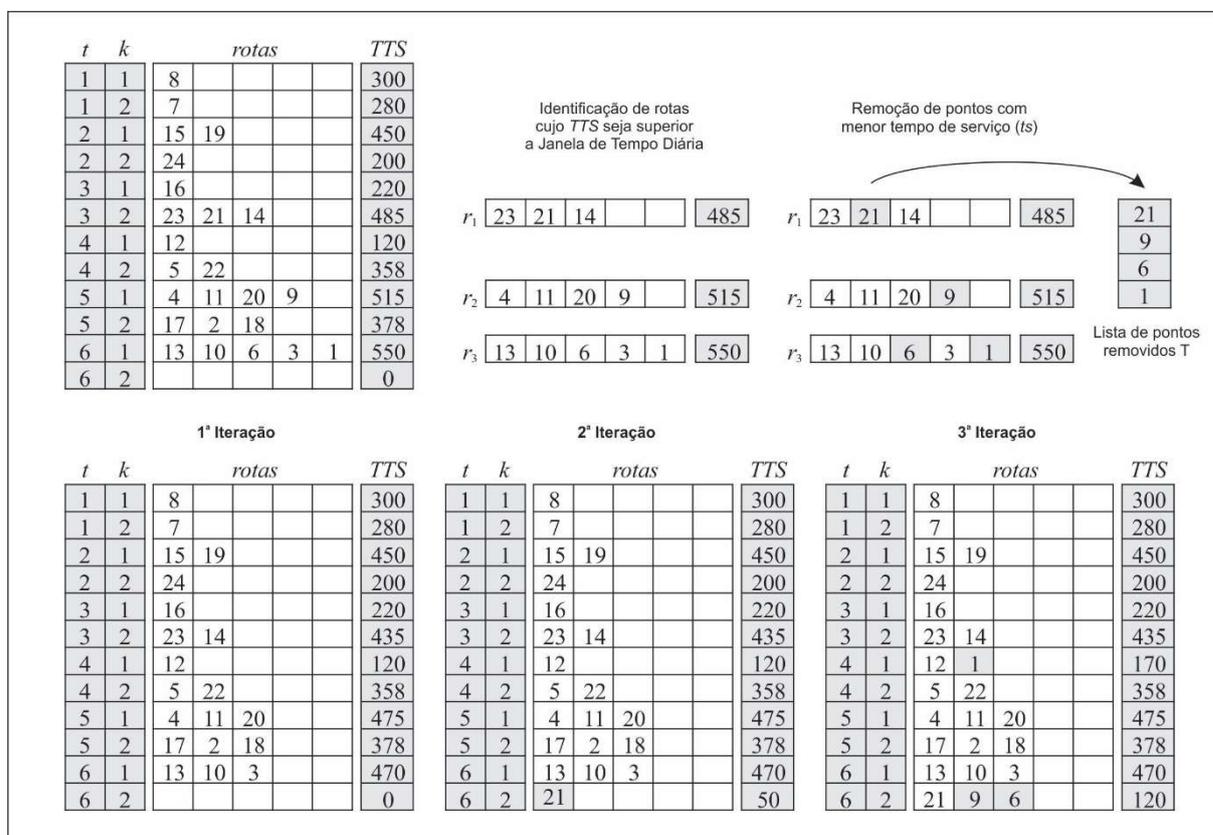
Fonte: O autor.

Após a remoção, deve-se redistribuir os pontos removidos (disponíveis na lista T) para outras rotas. Essa redistribuição é realizada para cada ponto de T e ocorre em três etapas. A primeira, compreende em redistribuir o ponto para uma das rotas vazias (passos 11 ao 17), como forma de corrigir estas rotas quanto a restrição

que determina que todo veículo deve sair da garagem diariamente. Outra restrição também observada é a restrição de janela de tempo (passo 11), visto que existe um período em que ponto deve ser visitado (vide Seção 4.2).

Caso não seja encontrada nenhuma rota vazia em que possa ser inserido o ponto, realiza-se uma busca por rotas com uma percentagem de tempo ocioso (passos 19 ao 25). Por fim, caso o ponto ainda não tenha sido inserido em uma das rotas, escolhe-se aleatoriamente uma rota e faz a inserção do ponto (passos 27 ao 30). A Figura 20 ilustra o movimento de correção, em que, a solução inicial é inviável em função do tempo total de serviço (TTS) ser superior à janela de tempo diária (JTD) estabelecida em 480 minutos.

Figura 20. Movimento Correção



Fonte: O autor.

O método seleciona as rotas que inviabilizam a solução ( $r_1$ ,  $r_2$  e  $r_3$ ). Para cada uma destas rotas, o movimento remove pontos de menor impacto, ou seja, pontos com menor tempo de serviço, até que o TTS da rota seja menor ou igual à JTD. No exemplo, os pontos 21, 9, 6 e 1 foram removidos das rotas  $r_1$ ,  $r_2$  e  $r_3$  respectivamente. Observa-se que na rota  $r_3$ , foi necessária a remoção de dois pontos para que o valor do TTS seja ajustado.

De posse da lista T de pontos removidos, o método reinsere estes pontos em outras rotas obedecendo ao seguinte critério: rotas com menor TTS tem prioridade. Assim, as rotas vazias (vide 1ª iteração e 2ª iteração) têm prioridade por terem seu TTS igual à zero. Isso faz com que a solução seja corrigida quanto à outra restrição do problema que determina que toda equipe deve sair da garagem todos os dias. No exemplo, o ponto 21 foi reinserido na rota que representa o dia 6 da equipe 2.

Inserido os pontos em rotas vazias, os demais são distribuídos nas demais rotas por ordem crescente do TTS. No exemplo, a 3ª iteração ilustra que os pontos 9, 6 e 1 foram inseridos nas rotas de menor TTS. Isso faz com que a restrição de Janela de Tempo Diária seja atendida, tornando a solução viável.

#### 4.4.2.4.7. Equilíbrio

O objetivo deste movimento é equilibrar as rotas quanto ao Tempo Total de Serviço (TTS). Para realizar este equilíbrio, as rotas são agrupadas em dois conjuntos aqui denominados de Conjunto Fechado e Conjunto Aberto. O Conjunto Fechado contém rotas que violam a restrição de Janela de Tempo Diária (JTD). O Conjunto Aberto, por sua vez, corresponde às rotas cujo Tempo Total de Medição (TTM) seja inferior à  $n\%$  da Janela de Tempo Diária (JTD), ou seja, rotas com  $n\%$  do tempo ocioso. Denomina-se tempo ocioso o tempo não trabalhado por uma equipe em um dia de planejamento. Nesse caso, considerando uma JTD igual a 480 minutos e  $n$  igual à 30%, o tempo ocioso seria de 144 minutos.

O pseudocódigo do movimento de equilíbrio é apresentado na Figura 21. Inicialmente é criado o Conjunto Fechado ( $cf$ ) com todas as rotas cujo Tempo Total de Serviço (TTS) seja superior à Janela de Tempo Diária (JTD) (passo 3). Uma vez definido o Conjunto Fechado, o procedimento percorre cada rota ( $cf_i$ ), e para cada rota, seleciona-se um ponto ( $cf_{ip}$ ) provável de remoção (passo 5). Provável, porque para cada ponto cria-se um Conjunto Aberto de possíveis rotas que podem receber este ponto em função da janela de tempo e do  $n\%$  de tempo ocioso (passo 6). Caso nenhuma rota se enquadre nestes critérios, escolhe-se outro ponto ( $cf_{ip}$ ), e uma nova tentativa de criar um Conjunto Aberto é realizada. Quando se forma um Conjunto Aberto (passo 7), o procedimento seleciona aleatoriamente uma rota deste conjunto (passo 8) para finalmente realizar a transferência do ponto ( $cf_{ip}$ ) para a nova rota do Conjunto Aberto (passos 9 e 10).

Figura 21. Pseudocódigo do movimento Equilíbrio

```

1.  EQUILIBRIO(s)
2.  INÍCIO
3.      CRIAR (conjunto de rotas cf com tempo superior à janela de tempo
         diária);
4.      PARA (cada rota l, l = 1,2,...,n, de cf) FAÇA
5.          PARA (cada ponto p disponível em l) FAÇA
6.              CRIAR (conjunto de rotas ca com tempo ocioso e possível para p);
7.              SE (conjunto ca > 0)
8.                  SELECIONAR (aleatoriamente uma rota r em l;
9.                  REMOVER (o ponto p de l);
10.                 INSERIR (o ponto p na rota r);
11.             FIM-SE;
12.         FIM-PARA;
13.     FIM-PARA;
32. FIM.

```

Fonte: O autor.

#### 4.4.2.5. Iterated Local Search para resolução do PRV-IF

A meta-heurística ILS proposta neste trabalho para resolução do problema de roteamento no planejamento de inventários florestais teve como base o algoritmo proposto por Lourenço et al. (2003), conforme descrito na Seção 2.4.1. O pseudocódigo da meta-heurística é apresentado na Figura 22.

Figura 22. ILS para o PRV-IF

```

1.  ILS(TempoMax, d e S)
2.  INÍCIO
3.      S0 ← Construção;
4.      SELECIONAR (aleatoriamente um movimento de Busca Local);
5.      S0 ← BuscaLocal(S0);
6.      S* ← S0;
7.      IncTemp ← 1;
8.      Temp ← TempInicial;
9.      ENQUANTO Tempo ≤ TempoMax FAÇA
10.         SE Temp < TempCongelamento FAÇA
11.             Temp ← TempInicial * IncTemp;
12.         FIM-SE;
13.         PARA (p = 1,2,...,d) FAÇA
14.             SELECIONAR (aleatoriamente um movimento de Perturbação);
15.             S' ← Perturbação(S0);
16.         FIM-PARA;
17.         SELECIONAR (aleatoriamente um movimento de Busca Local);
18.         S' ← BuscaLocal(S');
19.         SE S' < S0 FAÇA
20.             S0 ← S';
21.             SE f(S0) < f(S*) FAÇA
22.                 S* ← S0;
23.         FIM-SE;
24.         SENÃO
25.             SE CritérioAceitacao ENTÃO
26.                 S0 ← S';
27.             FIM-SE;
28.         FIM-SE;
29.         Temp ← Temp * txResfriamento;
30.         IncTemp ← IncTemp + 2;
31.     FIM-ENQUANTO;
32.     S ← S*;
33. FIM.

```

Fonte: O autor.

A principal diferença entre o algoritmo proposto e o algoritmo tradicional está na seleção aleatória das estratégias de Busca Local e de Perturbação (passos 4, 14 e 17), realizados a cada iteração do algoritmo.

Os movimentos de Busca Local utilizados foram: 1-Optimal intra-rota (Subseção 4.4.2.3.1) e Melhor Melhora (Subseção 4.4.2.3.2). Para a perturbação, foram utilizados os seguintes movimentos: Remoção Intra-rota (Subseção 4.4.2.4.1), Remoção entre rotas (Subseção 4.4.2.4.3), Troca entre rotas (Subseção 4.4.2.4.4) e Correção (Subseção 4.4.2.4.6).

#### 4.4.2.6. *Simulated Annealing* para resolução do PRV-IF

A meta-heurística SA proposta neste trabalho para resolução do PRV-IF teve como base o algoritmo proposto originalmente por Kirkpatrick et al. (1983), conforme descrito na Seção 2.4.2. No pseudocódigo da meta-heurística SA apresentado na Figura 23, pode-se observar que a principal diferença entre o algoritmo proposto e o algoritmo tradicional está na seleção aleatória das estratégias de Busca Local e de Perturbação (passos 4, 14 e 17), realizados a cada iteração do algoritmo.

Figura 23. SA para o PRV-IF

```

1.  SA(TempoMax, Ieq, Tres, T0, Tc e S)
2.  INÍCIO
3.    S0 ← Construção;
4.    S* ← S0;
5.    IncTemp ← 1;
6.    ENQUANTO Tempo ≤ TempoMax FAÇA
7.      Temp ← TempInicial * IncTemp;
8.      ENQUANTO T < Tc FAÇA
9.        ENQUANTO IAtual < Ieq FAÇA
10.         IAtual ← IAtual + 1;
11.         SELECIONAR (aleatoriamente um movimento de Perturbação);
12.         S' ← Perturbação(S0);
13.         Δ ← f(S') - f(S);
14.         SE Δ < 0 ENTÃO
15.           S ← S';
16.           SE (f(S') < f(S*)) ENTÃO
17.             S* ← S';
18.         FIM-SE;
19.       SENÃO
20.         SE CriterioAceitacao ENTÃO
21.           S ← S';
22.       FIM-SE;
23.     FIM-ENQUANTO;
24.   FIM-ENQUANTO;
25.   T ← Tres * T;
26.   IncTemp ← IncTemp + 2;
27.   IAtual ← 0;
28. FIM-ENQUANTO;
29. S ← S*;
30. FIM.

```

Fonte: Adaptado de MAURI (2008).

Os movimentos de Busca Local utilizados foram: 1-Optimal intra-rota (Subseção 4.4.2.3.1) e Melhor Melhora (Subseção 4.4.2.3.2). Para a perturbação, foram utilizados os seguintes movimentos: Remoção intra-rota (Subseção 4.4.2.4.1), Troca entre rotas (Subseção 4.4.2.4.4), Troca intra-rota (Subseção 4.4.2.4.2), Remoção entre rotas (Subseção 4.4.2.4.3), Shift (2,0) (Subseção 4.4.2.4.5), Correção (Subseção 4.4.2.4.6) e Equilíbrio (Subseção 4.4.2.4.7).

## 5. EXPERIMENTOS COMPUTACIONAIS

O sistema capaz de gerar arquivos de entrada (LPs) para o *software* de otimização CPLEX foi desenvolvido na linguagem de programação C++. Os testes exatos foram efetuados em uma máquina com microprocessador Intel Xeon 2.40 GHz com 32 GB de memória RAM utilizando o sistema operacional Windows 10.

As meta-heurísticas ILS e SA também foram desenvolvidas na linguagem de programação C++. Os testes foram efetuados em uma máquina com microprocessador Intel Core i5 750 2.67 GHz com 8 GB de memória RAM utilizando o sistema operacional Windows 7.

### 5.1. Descrição das instâncias

Os experimentos computacionais foram realizados em um conjunto de 9 instâncias elaboradas para esta finalidade. O maior problema contém 2016 pontos (talhões) que corresponde ao número máximo de talhões obtidos no processo de integração dos dados, conforme descrito na Seção 4.2. Para essa instância foi considerado um horizonte de 252 dias, que representa o número médio de dias úteis em um ano. A partir dessa instância foram geradas as demais conforme representado na Tabela 4. O objetivo foi obter diferentes tamanhos de instâncias para comparação dos métodos. Cada instância está associada a um número de equipes e a um tempo de processamento. Este último foi estabelecido com base no tempo gasto pelo método exato para encontrar uma solução ótima para a menor instância. O tempo de execução é fundamental na comparação entre as meta-heurísticas.

Tabela 4. Instâncias para o Problema de Roteamento de Veículos no Planejamento de Inventários Florestais

Instância	Nº de Pontos	Horizonte	Nº Equipes	Tempo
PRV-IF_0024_006	24	6 dias	2	120''
PRV-IF_0050_012	50	12 dias	2	180''
PRV-IF_0100_024	100	24 dias	2	240''
PRV-IF_0100_036	100	36 dias	2	240''
PRV-IF_0200_048	200	48 dias	2	480''
PRV-IF_0400_060	400	60 dias	2	960''
PRV-IF_0500_060	500	60 dias	3	1200''
PRV-IF_1000_120	1000	120 dias	3	2400''
PRV-IF_2016_252	2016	252 dias	3	4800''

## 5.2. Parâmetros

Para ajustar os parâmetros do método SA, foram escolhidas três instâncias ao acaso: PRV-IF\_0024\_006, PRV-IF\_0100\_024 e PRV-IF\_0200\_048. Foram realizadas combinações entre os parâmetros  $I_{eq}$ ,  $T_0$ ,  $T_{res}$  e  $T_c$  nas seguintes faixas de valores [500, 1000, 5000, 10000], [500, 1000, 5000, 10000], [0,775, 0,975] e [0,001, 0,01] respectivamente. Foram realizadas 5 execuções para cada instância com cada combinação.

Embora o ILS tenha apenas um parâmetro  $d$  que representa a taxa de destruição, foram obtidas três faixas de valores para instâncias acima de 500 pontos, entre 500 e 100 pontos e menor ou igual a 100 pontos. Para obtenção destes valores, utilizou-se a mesma metodologia aplicada no método SA. A faixa de valores utilizada para o parâmetro  $d$  foi [0,001, 0,025, 0,050]. Para o critério de aceitação do método ILS, utilizou-se os mesmos valores adotados pelo método SA. Os melhores valores encontrados para os parâmetros são apresentados na Tabela 5.

Tabela 5. Valores obtidos para os parâmetros dos métodos ILS e SA

Meta-heurística	Parâmetro	Significado	Melhor valor
ILS	$d$	Taxa de destruição > 500 pontos	0,001
	$d$	Taxa de destruição $\geq 100 \leq 500$ pontos	0,025
	$d$	Taxa de destruição < 100 pontos	0,050
SA	$I_{eq}$	Número de iterações	1000
	$T_0$	Temperatura inicial	1000
	$T_{res}$	Taxa de resfriamento	0,975
	$T_c$	Temperatura de congelamento	0,001

Os valores atribuídos às penalizações  $p_1$ ,  $p_2$  e  $p_3$  foram de 1000, 1000 e 100 respectivamente. Esses valores foram definidos empiricamente e por se tratar de um problema de minimização, estes valores foram propositalmente elevados com o objetivo de favorecer a aceitação de soluções viáveis.

## 6. RESULTADOS E DISCUSSÃO

Esta Seção descreve os principais resultados obtidos nesta pesquisa. Na Seção 6.1 é realizada uma avaliação comparativa entre os métodos propostos neste trabalho. Mais especificamente são discutidos os resultados das meta-heurísticas em comparação com os resultados do *software* CPLEX. Na Seção 6.2 são apresentadas as ferramentas desenvolvidas e na seção 6.3, os resultados são abordados como um estudo de caso para o setor florestal.

### 6.1. Comparação dos métodos

Nesta pesquisa foram propostas as meta-heurísticas ILS e SA como alternativa de solução do PIF. Para obter os resultados médios dessas meta-heurísticas, executou-se 30 vezes cada método para cada instância. Em todas as execuções foram encontradas soluções viáveis. Estes resultados são comparados aos resultados obtidos pelo *software* CPLEX.

A condição de parada das meta-heurísticas SA e ILS são estabelecidas na temperatura e no número de iterações, respectivamente. No caso da SA, quando a condição de parada é alcançada, reinicia-se o método a partir da melhor solução inicial obtida na primeira iteração. Esse processo se repete até que o método atinja o tempo limite. Para a ILS, a condição de parada que é realizada pelo número de iterações é apenas substituída pelo tempo. Os tempos de execução estabelecidos para cada instância são apresentados na Tabela 4.

Os experimentos computacionais são resumidos na Tabela 6. Para cada instância, são apresentados os valores obtidos pelos métodos CPLEX, ILS e SA. Quanto ao método exato, as colunas representam: a melhor solução encontrada (coluna FO), o limitante inferior (coluna Limitante), tempo de processamento (coluna Tempo) e o erro relativo percentual (coluna GAP). Para as meta-heurísticas, as colunas representam: a melhor solução encontrada (coluna FO), a solução média encontrada (coluna Média), o coeficiente de variação (coluna CV) e o tempo para encontrar a melhor solução (coluna Tempo). Os valores da coluna GAP são dados pela seguinte fórmula:

$$GAP = \frac{FO - Limitante}{Limitante} * 100\%$$

O GAP é mostrado pelo próprio *software* CPLEX e indica o erro relativo percentual da melhor solução viável encontrada e do melhor limitante inferior obtido

pelo algoritmo dentro do tempo limite estabelecido. Caso o GAP seja igual a 0,0%, significa que foi encontrada a solução ótima para o problema.

Com base na análise dos resultados da Tabela 6, pode-se observar que o método exato foi capaz de obter a solução ótima apenas para a menor instância (PRV-IF\_0024\_006). Para as instâncias PRV-IF\_0050\_012 e PRV-IF\_0100\_036, embora o CPLEX tenha encontrado soluções exatas, estas ficaram distantes da solução ótima em no máximo 5,88% e 8,32% respectivamente (vide coluna GAP). Nas instâncias de PRV-IF\_0100\_024 e PRV-IF\_0200\_048, o CPLEX não foi capaz de encontrar soluções ótimas, indicando apenas os limitantes inferiores. Nas demais instâncias, o CPLEX não foi capaz de encontrar soluções. O gráfico representado na Figura 24, ilustra a distância percentual entre as meta-heurísticas ILS e SA em relação aos limitantes obtidos pelo *solver* CPLEX para instâncias de até 200 pontos.

As meta-heurísticas ILS e SA foram capazes de encontrar soluções viáveis para 8 das 9 instâncias testadas. Apenas na instância PRV-IF\_2016\_252, as meta-heurísticas não foram capazes de encontrar solução viável no tempo estabelecido. Neste caso, a janela de tempo diária (JTD) desta instância foi flexibilizada na tentativa de encontrar soluções viáveis para o problema. Desta forma, utilizou-se a JTD em 490 minutos (instância PRV-IF\_2016\_252<sup>1</sup>), 500 minutos (instância PRV-IF\_2016\_252<sup>2</sup>) e 510 minutos (instância PRV-IF\_2016\_252<sup>3</sup>). Com essa flexibilização, os métodos foram capazes de encontrar soluções viáveis. Os melhores valores médios encontrados são apresentados em negrito.

A meta-heurística SA obteve os melhores resultados quando comparados com os demais métodos ILS e CPLEX. Apenas nos problemas da instância PRV-IF\_2016\_252 flexibilizada com 500 (PRV-IF\_2016\_252<sup>2</sup>) e 510 (PRV-IF\_2016\_252<sup>3</sup>) minutos a ILS foi superior ao SA. Deve-se considerar que, a flexibilização da JTD para 500 minutos, fazendo com que as equipes tenham que trabalhar vinte minutos a mais diariamente, possibilitou a ILS encontrar uma solução com FO inferior à FO da solução encontrada pela SA com a JTD de 490 minutos. Essas flexibilizações permitem a empresa avaliar o custeio das horas extras em relação ao ganho de tempo no desenvolvimento das atividades.

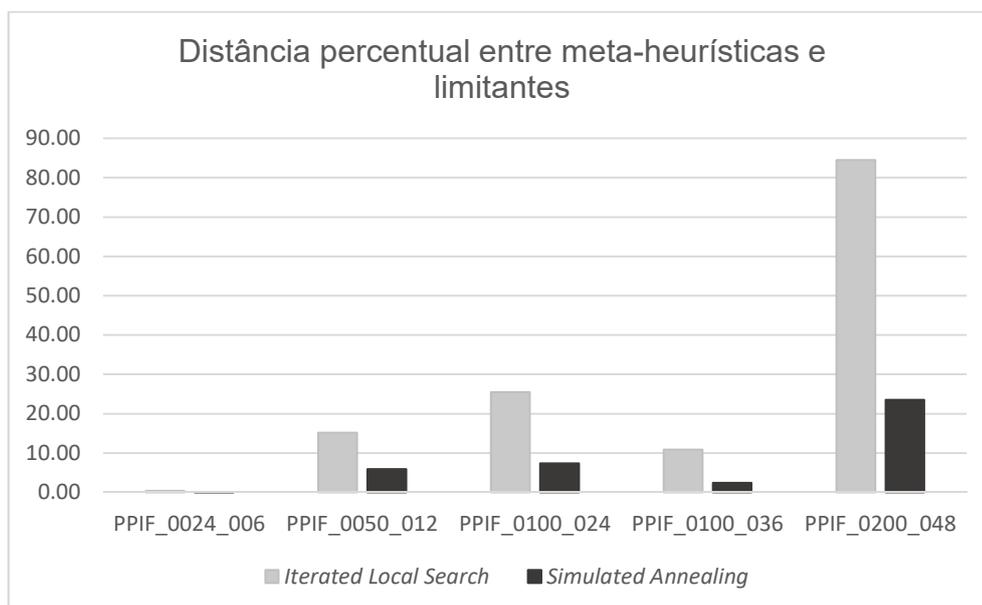
O bom desempenho destes métodos também pode ser verificado quando se avalia o coeficiente de variação (Tabela 6), visto que em apenas uma situação ele ultrapassou a margem de um por cento. Isso ocorreu com a meta-heurística ILS para a instância PRV-IF\_0050\_012.

Tabela 6. Resultados obtidos pelos métodos CPLEX, ILS e SA para as instâncias do PRV-IF

Instância	CPLEX				ILS				SA			
	FO	Limitante	Tempo(s)	GAP	FO	Média	CV(%)	Tempo(s)	FO	Média	CV(%)	Tempo(s)
PRV-IF_0024_006	132,61	132,61	43,78	0.00	132,61	133,12	0,36	2,60	132,61	<b>132,65</b>	0,10	17,14
PRV-IF_0050_012	2318,97	2190,03	1787,20	5.88	2418,33	2522,96	1,12	141,39	2318,95	<b>2319,56</b>	0,06	19,50
PRV-IF_0100_024	-	3722,80	357,22	∞	4606,25	4672,29	0,81	148,18	3985,29	<b>3997,40</b>	0,14	40,37
PRV-IF_0100_036	6446,30	5951,28	47103,37	8.32	6481,70	6596,90	0,72	16,99	6084,63	<b>6094,29</b>	0,13	49,68
PRV-IF_0200_048	-	6307,32	6,44	∞	11356,34	11634,67	0,91	234,75	7678,00	<b>7794,52</b>	0,60	160,30
PRV-IF_0400_060	-	-	-	-	23806,79	24137,14	0,59	581,73	14634,03	<b>14822,84</b>	0,81	581,95
PRV-IF_0500_060	-	-	-	-	24673,59	25192,68	0,86	283,34	14378,04	<b>14651,20</b>	0,82	1199,20
PRV-IF_1000_120	-	-	-	-	63245,32	64202,35	0,61	2045,55	63280,23	<b>63960,11</b>	0,47	2284,61
PRV-IF_2016_252 <sup>1</sup>	-	-	-	-	126627,15	128696,41	0,64	4766,06	125460,82	<b>126696,31</b>	0,48	4658,02
PRV-IF_2016_252 <sup>2</sup>	-	-	-	-	124472,54	<b>126013,76</b>	0,60	3189,07	126211,89	127524,40	0,53	4581,68
PRV-IF_2016_252 <sup>3</sup>	-	-	-	-	125023,11	<b>127110,10</b>	0,80	4468,10	126956,55	128633,28	0,66	4601,87

A coluna FO (melhor valor para a Função Objetivo encontrado) representa o tempo de deslocamento das equipes em minutos. A coluna Limitante representa o limitante inferior fornecido pelo *solver* CPLEX. A coluna Tempo representa o tempo em segundos para encontrar o melhor valor para a FO. A coluna GAP representa o erro relativo percentual disponibilizado pelo *solver* CPLEX. A coluna média representa o valor médio das 30 execuções dos métodos ILS e SA. A coluna CV representa o coeficiente de variação da média.

Figura 24. Distância percentual entre meta-heurísticas e limitante para cada instância



Fonte: O autor.

Na literatura, é comum que experimentos computacionais para PRVJT e PRVPJT clássicos utilizem instâncias com no máximo 100 locais Solomon (1987). Recentemente, novos conjuntos de instâncias para o PRVJT foram estendidos para problemas com 200, 400, 600, 800 e 1.000 locais. Para o PRVPJT, as instâncias clássicas disponíveis na literatura e introduzidas por Cordeau et al. (2001) variam de 48 a 288 locais, 3 a 20 veículos e horizonte de planejamento de 4 ou 6 dias. Nesse aspecto, o PRV-IF mostra-se um problema complexo e de difícil solução, dada a quantidade de talhões a serem visitados e a quantidade de dias atribuídos ao horizonte de planejamento.

Para tanto, os diferentes tamanhos de instâncias criadas para o problema proporcionam avaliar a eficácia e eficiência das meta-heurísticas quando comparadas com o CPLEX. Observa-se que para a menor instância PRV-IF\_0024\_006 (Tabela 6), as meta-heurísticas foram capazes de encontrar a solução ótima para o problema. Com o aumento da complexidade das instâncias, o método exato torna-se ineficaz e, até mesmo, ineficiente para as maiores instâncias.

Na Tabela 7 são apresentados os tempos em minutos gasto pelos métodos para encontrar as melhores soluções para os problemas. Observa-se que as meta-heurísticas foram capazes de encontrar as melhores soluções em um tempo inferior ao *solver* CPLEX para todas as instâncias testadas. Embora não seja garantido que

uma meta-heurística encontre a solução ótima para o problema, estes métodos mostraram-se eficazes por encontrar soluções para todas as instâncias independentemente da complexidade.

Tabela 7. Menor tempo em minutos para encontrar a melhor solução por método

Instância	Tempo (min)		
	CPLEX	ILS	SA
PRV-IF_0024_006	0,73	0,04	0,29
PRV-IF_0050_012	29,79	2,36	0,33
PRV-IF_0100_024	5,95	2,47	0,67
PRV-IF_0100_036	785,06	0,28	0,83
PRV-IF_0200_048	0,11	3,91	2,67
PRV-IF_0400_060	-	9,70	9,70
PRV-IF_0500_060	-	4,72	19,99
PRV-IF_1000_120	-	34,09	38,08
PRV-IF_2016_252 <sup>1</sup>	-	79,43	77,63
PRV-IF_2016_252 <sup>2</sup>	-	53,15	76,36
PRV-IF_2016_252 <sup>3</sup>	-	74,47	76,70

## 6.2. Ferramentas desenvolvidas

Na metodologia desta pesquisa, foram utilizadas diversas ferramentas como descrito no Capítulo 4. Dentre estas, as ferramentas “Gerador de instâncias”, “Modelo” e “Meta-heurísticas” representadas na Figura 4 foram desenvolvidas para esta tese. Embora não tenham sido desenvolvidas interfaces para estas ferramentas, todas as funcionalidades necessárias para o bom funcionamento foram implementadas e testadas.

A ferramenta “Gerador de Instâncias” é capaz de gerar instâncias (Figura 9) de diferentes tamanhos para o PRV-IF. Nesta, pode-se optar por gerar instâncias reais a partir de dados fornecidos por sistemas de SIG e ou a partir de dados aleatórios. A ferramenta “Modelo” foi desenvolvida com o objetivo de fazer uma interface com o *software* CPLEX. Essa ferramenta faz a leitura de arquivos de instâncias e gera novos arquivos que servem de *input* para o *software*. Essa também é capaz de atribuir uma solução inicial, obtida por uma das meta-heurísticas, ao CPLEX.

Por fim, a ferramenta “Meta-heurísticas” contém os algoritmos ILS e SA como descritos na metodologia desta pesquisa. Para obter um melhor desempenho destes métodos, utilizou-se no desenvolvimento dessa ferramenta estruturas estáticas. Diferente das estruturas de dados dinâmicas, as estruturas estáticas

necessitam de mais recursos computacionais, em especial a memória. Para contornar este problema, foram utilizados alguns recursos como de trabalhar com janela de tempo binária e armazenar a matriz de distâncias em variáveis do tipo inteiro, que exigem um menor uso da memória disponível no computador.

### 6.3. Estudo de caso

Nesta seção, os resultados apresentados são analisados face às necessidades executivas propostas por uma empresa do setor florestal. Embora o número de talhões previstos na programação de 2015 da empresa tenham sido de 2567, neste estudo não foi possível contemplar todos estes talhões, dada as dificuldades encontradas na integração das diferentes bases de dados.

Para que seja possível a integração de bases de dados distintas, é necessário que se tenha um identificador em comum nas bases envolvidas. O identificador utilizado nesta pesquisa foi o código de uso do solo, como descrito no Capítulo 4. Ao realizar a junção destas bases considerando o identificador citado, parte dos registros foi perdida em decorrência desta inconsistência.

Outra dificuldade encontrada no processamento dos dados está relacionada ao fato da programação do inventário não ser automatizada. Nessa atividade, a empresa em questão utiliza o recurso de planilhas eletrônicas. Esse tipo de recurso é passível de erros, porque não são realizadas validações de consistências automaticamente.

Vale ressaltar que, para um melhor desempenho da metodologia proposta nesta pesquisa, é importante que as bases de dados que servem de *input* sejam consistentes entre si. Deve-se existir uma preocupação, por parte da empresa, em validar estes dados e realizar o registro de outras informações como ponto de acesso às parcelas na estrada e tempo de deslocamento entre talhões nas mais diferentes rotas. Dessa forma, a instância PRV-IF\_2016\_252 é o problema mais próximo do real e que melhor pode representá-lo.

Para o problema PRV-IF\_2016\_252, a melhor solução encontrada foi da meta-heurística SA, que considerou 2016 pontos para 252 dias com 3 equipes trabalhando 490 minutos por dia (PRV-IF\_2016\_252<sup>1</sup>). Outra solução que merece destaque para esse mesmo problema, foi a solução obtida pela meta-heurística ILS que considerou 2016 pontos para 252 dias com 3 equipes trabalhando 500 minutos por dia (PRV-IF\_2016\_252<sup>2</sup>). Exemplos de rotas obtidas por estes resultados são ilustrados nas Figuras 25 e 26.

Na Figura 25, são apresentadas as rotas de cada uma das equipes para os 3 primeiros dias do planejamento para a instância PRV-IF\_2016\_252<sup>1</sup>. As cores vermelho, azul e verde identificam as equipes 1, 2 e 3, respectivamente. No primeiro dia de planejamento, a equipe 1 (vermelho) é responsável por realizar o inventário nos talhões 324 e 1413. Desta forma, a equipe sai da fábrica, desloca até o ponto 324, realiza o inventário nas parcelas destinadas a este talhão, em seguida a equipe se desloca para o ponto 1413, realiza o inventário nas parcelas e retorna para a fábrica no final. O mesmo ocorre no segundo dia em que esta mesma equipe deve deixar a fábrica e se deslocar entre os pontos 1273, 777 e 969. Após as atividades no último talhão, a equipe retorna para a fábrica. Não diferente dos dois primeiros dias, a equipe 1 no terceiro dia tem a responsabilidade de realizar inventários nas parcelas dos pontos 265, 1404 e 1581. Semelhante ao que fora apresentado, na Figura 25 e na Figura 26 são apresentadas as rotas de cada equipe para os três primeiros dias do planejamento para a instância PRV-IF\_2016\_252<sup>2</sup>, obtidas pela meta-heurística ILS.

Parte das soluções com a programação detalhada destas instâncias pode ser vista nos apêndices A e B. Na primeira linha, tem-se o valor referente ao número de pontos considerando a origem e o destino. O valor representado na segunda linha refere-se ao horizonte de planejamento. Na terceira linha, encontra-se o número de equipes. Os valores da quarta e quinta linhas são referentes ao valor da função objetivo e a viabilidade ou não da solução encontrada (um para sim e zero caso contrário). Por fim, as demais linhas representam as rotas de cada dia para cada equipe.

A Tabela 8 apresenta um resumo dos resultados obtidos pela meta-heurística SA para o problema PRV-IF\_2016\_252. Nesta, para cada mês previsto no planejamento, são apresentados o número total de parcelas inventariadas (coluna Nº de Parcelas), o número real de parcelas medidas pela empresa no período (coluna Nº de Parcelas Real), o tempo total de serviço em minutos, o tempo total de deslocamento e a distância percorrida pelas três equipes em km.

Pode-se observar que, embora o número de parcelas seja inferior ao número de parcelas real, por nesta pesquisa não terem sido considerados todos os talhões previstos na programação de 2015, como descrito no início desta seção, os resultados do modelo proposto mostram-se relevantes, visto que proporcionou uma melhor distribuição no número de parcelas inventariadas por mês. Observa-se que

neste período o coeficiente de variação reduziu significativamente com uso da metodologia proposta.

Figura 25. Melhor resultado encontrado pela meta-heurística SA para os 3 primeiros dias da instância PRV-IF\_2016\_252<sup>1</sup>

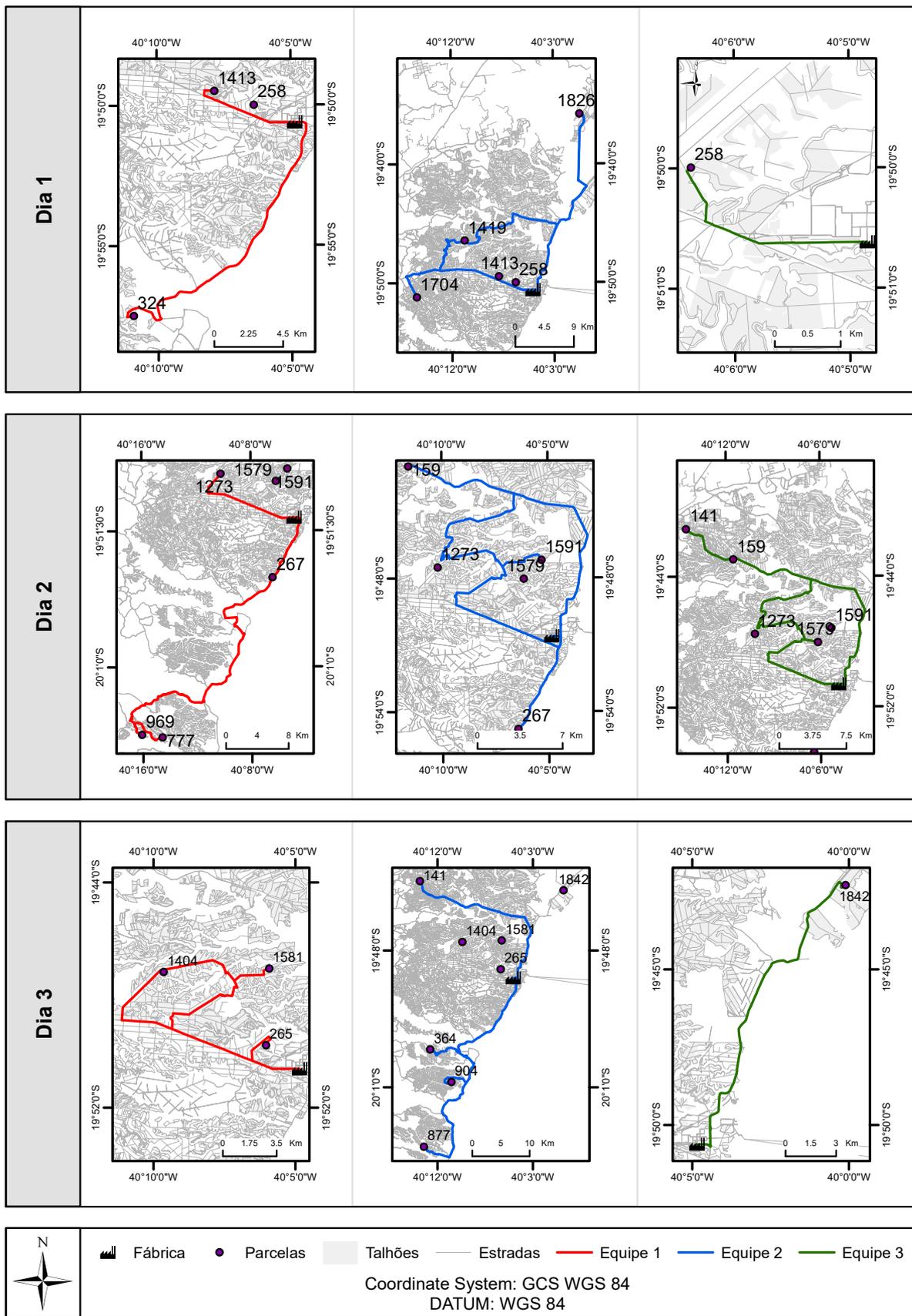


Figura 26. Melhor resultado encontrado pela meta-heurística ILS para os 3 primeiros dias da instância PRV-IF\_2016\_252<sup>2</sup>

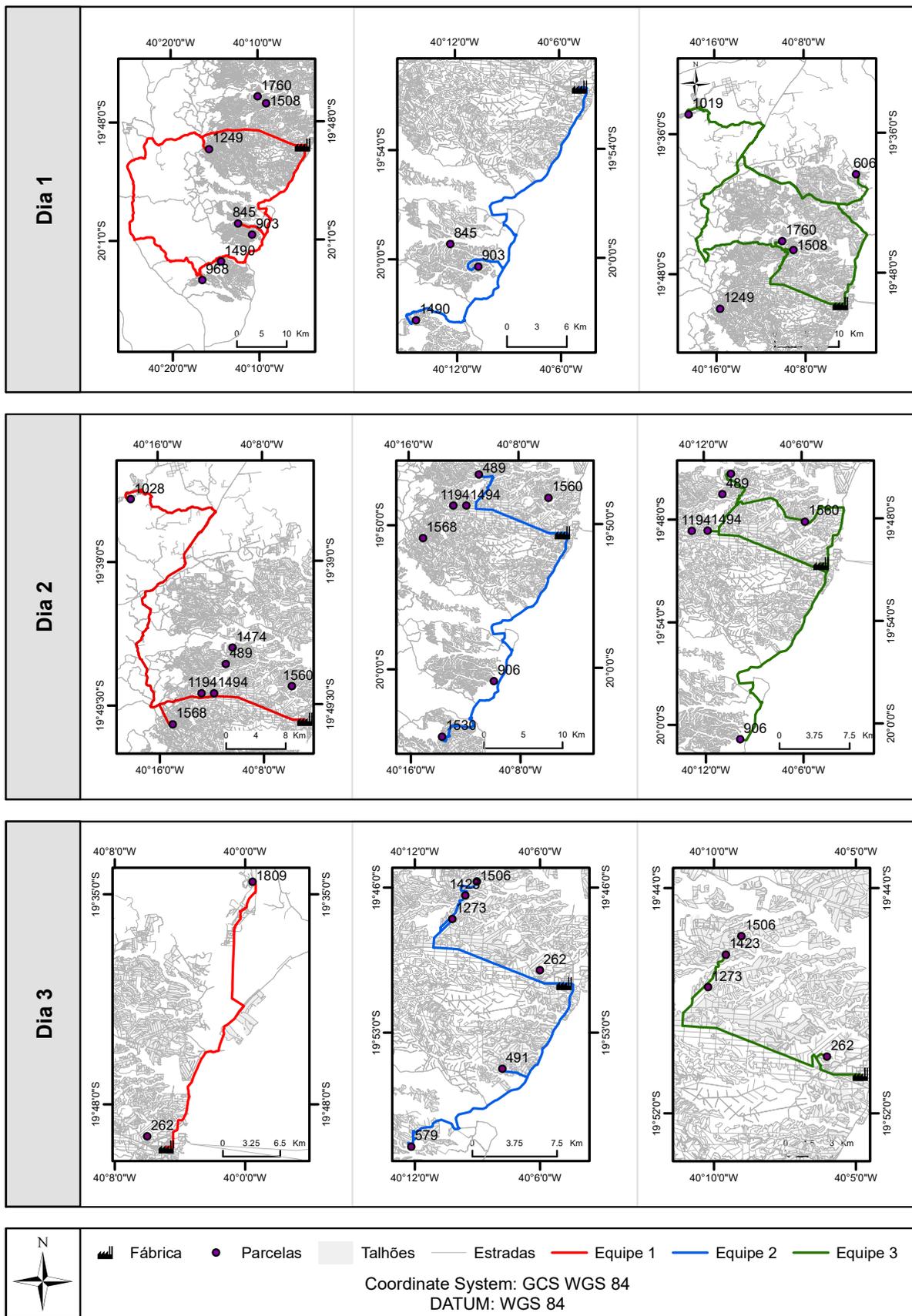


Tabela 8. Resumo mensal dos valores obtidos pela meta-heurística SA para a instância PRV-IF\_2016\_252

<b>Mês</b>	<b>Nº de Parcelas</b>	<b>Nº de Parcelas Real</b>	<b>Tempo de Serviço</b>	<b>Tempo de Deslocamento (min)</b>	<b>Distância Percorrida (km)</b>
Janeiro	480	759	10083	8615,20	5008,84
Fevereiro	475	692	10562	10068,09	5853,54
Março	586	820	13589	12029,04	6993,63
Abril	537	611	11976	11962,00	6954,65
Mai	605	418	13081	11232,35	6530,44
Junho	620	718	13664	10711,18	6227,43
Julho	575	735	12836	10003,95	5816,25
Agosto	507	429	11406	9768,37	5679,28
Setembro	484	899	11344	9932,90	5774,94
Outubro	432	354	10359	9958,90	5790,06
Novembro	482	761	10943	9393,18	5461,15
Dezembro	493	476	12067	11785,66	6852,13
<b>Total</b>	<b>6276</b>	<b>7672</b>	<b>141910</b>	<b>125460.82</b>	<b>72942,34</b>
<b>C.V.</b>	<b>11.50</b>	<b>27.88</b>			

## 7. CONCLUSÕES E TRABALHOS FUTUROS

Nesta pesquisa, buscou-se propor um método capaz de encontrar soluções para o problema de roteamento de equipes no planejamento de inventários florestais. Foi utilizado um modelo adaptado do Problema de Roteamento Periódico de Veículos com Janela de Tempo. Os métodos de solução utilizados foram o método exato por meio do *software* CPLEX e as meta-heurísticas ILS e SA.

Nas condições em que a pesquisa foi realizada, a considerar algumas dificuldades em relação à integração dos dados, a análise dos resultados permitiu apresentar as seguintes conclusões:

- A modelagem do problema contribuiu de forma eficaz na obtenção dos resultados, uma vez que toda a programação de rotas estabelecidas para cada equipe e em cada dia de planejamento foi obtida. Além da definição das rotas, também foram estabelecidos os horários de saída e chegada de cada equipe nos talhões e na fábrica.
- Foi desenvolvido um sistema para gerar cenários (instâncias) para o problema. O mesmo é capaz de gerar instâncias aleatórias e instâncias provenientes de dados reais de empresas do setor.
- Foi desenvolvido um sistema de otimização capaz de obter soluções para o problema por meio do *solver* CPLEX e das meta-heurísticas ILS e SA.
- Os experimentos computacionais mostraram um bom desempenho das meta-heurísticas em relação aos resultados encontrados pelo *solver* CPLEX. Dentre as meta-heurísticas, a SA mostrou-se como um método mais eficaz em relação a ILS na busca por soluções viáveis. O SA foi superior em 9 dos 11 problemas testes.
- Foram obtidos avanços na metodologia em relação ao trabalho proposto por Meneguzzi (2011), em especial na obtenção de rotas diárias e uso de métodos capazes de obter soluções viáveis para problemas em escalas muito superiores.

### 7.1. Trabalhos futuros

Embora os resultados apresentados sejam satisfatórios, esta pesquisa pode ser complementada, principalmente no sentido de investigar novos métodos de solução para o problema. Com esse intuito, sugere-se:

- Desenvolver um estudo relacionado à organização e a integração de dados com a finalidade de aproximar os resultados obtidos dos resultados reais observados em uma empresa florestal.
- Investigar técnicas de identificação do melhor ponto de acesso às parcelas, como por exemplo, uma integração desta pesquisa com a pesquisa desenvolvida por Marcatti (2013).
- Integrar as meta-heurísticas implementadas (ILS e SA) com o *solver* CPLEX, de forma que os métodos se comportem como heurísticas de perturbação e o CPLEX como mecanismo de intensificação da solução.

## 8. REFERÊNCIAS BIBLIOGRÁFICAS

ANDRADE, E. L. de. **Introdução à pesquisa operacional: métodos e modelos para a análise de decisão**. 5. ed. Rio de Janeiro: LTC, 2015. 220 p.

ARENALES, M. et al. **Pesquisa Operacional para os Cursos de Engenharia**. 2. ed. Rio de Janeiro: Elsevier, 2015. 564 p.

ARROYO, J. E. C. **Heurísticas e metaheurísticas para otimização combinatória multiobjetivo**. 2002. 225 f. Tese (Doutorado em Engenharia Elétrica) – Curso de Pós-Graduação em Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, SP, 2002.

AVERY, T. E.; BURKHART, H. E. **Forest measurements**. 3. ed. New York: McGraw-Hill, 1983. 331 p.

BAZARAA, M. S.; JARVIS, J. J.; SHERALI, H. D. **Linear programming and network flows**. 4. ed. New Jersey: John Wiley & Sons, 2010. 748 p.

BODIN, L.; GOLDEN, B. Classification in vehicle routing and scheduling. **Networks**, v. 11, n. 2, p. 97-108, 1981.

BODIN, L. D et al. Routing and scheduling of vehicles and crews: The state of the art. **Computers & Operations Research**, v.10, n.2, p. 63-211, 1983.

CAMPOS, J. C. C.; LEITE, H. G. **Mensuração florestal: perguntas e respostas**. 5. ed. Viçosa: UFV, 2017. 632 p.

CATTARUZZA, D. et al. An iterated local search for the multi-commodity multi-trip vehicle routing problem with time windows. **Computers & Operations Research**, v. 51, p. 257–267, 2014.

CHIANG, W-C.; RUSSELL, R. A. Simulated annealing metaheuristics for the vehicle routing problem with time windows. **Annals of Operations Research**, v. 63, p. 3-27, 1996.

COLIN, E. C. **Pesquisa operacional: 170 aplicações em estratégia, finanças, logística, produção, marketing e vendas**. Rio de Janeiro: LTC, 2007. 501 p.

COLLETTE, Y.; SIARRY, P. **Multiobjective optimization**: principles and case studies. Paris: Springer, 2003. 293 p.

CORDEAU, J. -F.; LAPORTE, G.; MERCIER, A. A unified tabu search heuristic for vehicle routing problems with time windows. **Journal of the Operational Research Society**. v. 52, n. 8, p. 928–936, 2001.

CORDEAU, J. -F. et al. A guide to vehicle routing heuristics. **Journal of the Operational Research Society**. v. 53, n. 5, p. 512–522, 2002.

CRISTOFIDES, N.; BEASLEY, J. E. The period routing problem. **Networks**, v. 14, p. 237-256, 1984.

CZECH, Z.; CZARNAS, P. Parallel simulated annealing for the vehicle routing problem with time windows. In: EUROMICRO WORKSHOP ON PARALLEL DISTRIBUTED AND NETWORK-BASED PROCESSING, 10., 2002, Canary Islands. **Proceedings...** Canary Islands, Spain: IEEE, 2002. p. 376-383.

DANTZIG, G. B.; RAMSER, J. H. The truck dispatching problem. **Management Science**, v., n. 6, p. 80-91, 1959.

DANTZIG, G. B. **Linear programming and extensions**. Princeton, N.J.: Princeton University Press, 1963. 617 p.

EKSIOGLU, B.; VURAL, A. V.; REISMAN, A. The vehicle routing problem: A taxonomic review. **Computers & Industrial Engineering**, v. 57, n. 4, p. 1472-1483, 2009.

ENVIRONMENTAL SYSTEMS RESEARCH INSTITUTE (ESRI, 2016). Help What is a network dataset. Disponível em: ArcGis 10.1 2016.

FICHER, M. L.; JAIKUMAR, R. A generalized assignment heuristic for vehicle routing. **Networks**, v. 11, n. 2, p. 109-124, 1981.

GOLDBARG, M. C.; LUNA, H. P. L. **Otimização Combinatória e Programação Linear**: Modelos e Algoritmos. 2. ed. Rio de Janeiro: Campus, 2005. 518 p.

GOLDBARG, E.; GOLDBARG, M. C.; LUNA, H. P. L. **Otimização Combinatória e Meta-heurísticas**: Algoritmos e Aplicações, Rio de Janeiro: Campus, 2015. 416 p.

GOMES, H. A. S.; NETO, J. F. B. Utilização de metaheurística na programação de escala de pessoal em empresas de transporte coletivo por ônibus. In: SIMPÓSIO BRASILEIRO DE PESQUISA OPERACIONAL, 35., 2003, Natal, RN. **Anais...** Natal, 2003. p. 894-905.

GOMES JÚNIOR, A. C.; SOUZA, M. J. F.; MARTINS, A. X. Simulated annealing aplicado à resolução do problema de roteamento de veículos com janela de tempo. **Revista Transportes**, v. 13, n. 2, p. 5-20, 2005.

GOMIDE, L. R. **Planejamento florestal espacial**. 2009. 180 f. Teste (Doutorado em Engenharia Florestal) – Universidade Federal do Paraná, Curitiba, 2009.

GONÇALVES, L. B. **Heurísticas GRASP para um Problema de Roteamento Periódicos de Veículos**. 2005, 62 f. Dissertação (Mestrado em Computação) – Programa de Pós-Graduação em Computação, Universidade Federal Fluminense, Niterói, 2005.

HADDAD, H. M. D. **Roteamento otimizado no inventário florestal das árvores de Belo Horizonte**. 2015, 63 f. Dissertação (Doutorado em Engenharia Florestal) – Programa de Pós-Graduação em Engenharia Florestal, Universidade Federal de Lavras, Lavras, 2015.

HILLIER, F. S.; LIEBERMAN, G. J. Introdução à pesquisa operacional. 9. ed. Porto Alegre: AMGH, 2012. 1028p.

HUSCH, B.; BEERS, T. W.; KERSHAW JR., J. A. **Forest mensuration**. 4. ed. New Jersey: John Wiley & Sons, 2003.

KANEGAE JÚNIOR, H. et al. Avaliação de interpoladores estatísticos e determinísticos como instrumento de estratificação de povoamentos clonais de Eucalyptus sp. **Cerne**. Lavras, v. 12. P. 123-136, abr./jun. 2006.

KANGAS, A.; MALTANO, M. **Forest inventory: methodology and applications**. New York: Springer, 2006. 363 p.

KIRKPATRICK, S.; GELLAT, D. C.; VECCHI, M. P. Optimization by simulated annealing. **Science**, v. 220, n 4598, p. 671-680, 1983.

LAPORTE, G. et al. Classical and modern heuristics for the vehicle routing problem. **International Transactions in Operational Research**, v. 7, n 4-5, p. 285-300, 2000.

LENSTRA, J.; RINNOOY KAN, A. Complexity of vehicle routing and scheduling problems. **Networks**, v. 11, p. 221-227, 1981.

LOURENÇO, H. R.; MARTIN, O. C.; STUTZLE, T. Iterated local search. In: GLOVER, F.; KOCHENBERGER, G. (Ed.). **Handbook of Metaheuristics**. Boston: Kluwer Academic Publishers, 2003.

LOETSCH, F.; HALLER, K. E. **Forest inventory**. Munich, BLV, 1964. v. 1, 436 p.

LUENBERGER, D. G.; YE, Y. **Linear and Nonlinear Programming**. 3. ed. CA: Springer. 2008. 546 p.

MAURI, G. R. **Novas abordagens para representação e obtenção de limitantes e soluções para alguns problemas de otimização combinatória**. 2008, 241 f. Tese (Doutorado em Computação Aplicada) – Programa de Pós-Graduação em Computação Aplicada, Instituto Nacional de Pesquisas Espaciais, São José dos Campos, 2008.

MARCATTI, G. E. **Caminhamento ótimo para acesso às parcelas de inventário florestal**. 2013, 32f. Dissertação (Mestrado em Ciência Florestal) – Programa de Pós-Graduação em Ciência Florestal, Universidade Federal de Viçosa, Viçosa, 2013.

MENEGUZZI, C. C. **Modelo de roteamento de veículos aplicado ao planejamento do inventário florestal**. 2011, 94 f. Dissertação (Mestrado em Ciências Florestais) – Programa de Pós-Graduação em Ciências Florestais, Universidade Federal do Espírito Santo, Jerônimo Monteiro, 2011.

MICHALLET, J. et al. Multi-start iterated local search for the periodic vehicle routing problem with time windows and time spread constraints on services. **Computers & Operations Research**, v. 41, p. 196–207, 2014.

MUNHOZ, P. L. A. **Um algoritmo baseado em iterated local search para o problema de roteamento de veículos periódico**. 2012, 50 f. Dissertação (Mestrado em Computação) - Programa de Pós-Graduação em Computação, Universidade Federal Fluminense, Niterói, 2012.

MUNHOZ, P. L. A.; OCHI, L. S.; SOUZA, M. J. F. Um algoritmo baseado em iterated local search para o problema de roteamento de veículos periódico. In: Encontro Nacional de Engenharia de Produção, 32., 2012, Bento Gonçalves, RS. **Anais...** Bento Gonçalves, 2012.

NGUYEN, P. K.; CRAINIC, T. G.; TOULOUSE, M. A hybrid generational genetic algorithm for the periodic vehicle routing problem with time windows. **Journal of Heuristics**, v. 20, n. 4, p. 383–416, 2014.

NOVAES, A. G. **Sistemas Logísticos: Transporte, Armazenagem e Distribuição de Produtos**. São Paulo: Edgard Bluncher, 1989.

OLIVEIRA, A. C. **Geotecnologia na estratificação de povoadamentos de *Eucalyptus ssp.*** 2006. 100 f. Dissertação (Mestrado em Engenharia Florestal) – Universidade Federal de Lavras, 2006.

OSMAN, I. H. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. **Annals of Operations Research**, v. 41, n. 4, p. 421–451, 1993.

OSMAN, I. H.; LAPORTE, G. Metaheuristics: a bibliography. **Annals of Operations Research**. v. 63, p. 511-623, 1996.

PENNA, P. H. V.; SUBRAMANIAN, A.; OCHI, L. S. An Iterated Local Search heuristic for the Heterogeneous Fleet Vehicle Routing Problem. **Journal of Heuristics**, v. 19, n. 2, p. 201–232, 2013.

RIBAS, S. **Um algoritmo híbrido para o problema de roteamento de veículos com janelas de tempo**. 2011, 59 f. Dissertação (Mestrado em Computação) – Programa de Pós-Graduação em Computação, Universidade Federal Fluminense, Niterói, 2011.

RODRIGUES, F. L. **Metaheurística e sistema de suporte à decisão no gerenciamento de recursos florestais**. 2001. 25 f. Tese (Doutorado em Engenharia Florestal) – Curso de Pós-graduação em Engenharia Florestal, Universidade Federal de Viçosa, Viçosa, MG, 2001.

SANQUETTA, C. R. et al. **Inventários florestais: planejamento e execução**. 2. ed. Curitiba: Multigraphics, 2009. 307 p.

SCOLFORO, J. R. S.; MELLO, J. M. **Inventário florestal**. Lavras: UFLA – FAEPE, 2006. 561 p.

SILVA, E. M. et al. **Pesquisa Operacional: programação linear e simulação**. 5. ed. São Paulo: Atlas, 2017.

SILVA, M. M.; SUBRAMANIAN, A.; OCHI, L. S. An iterated local search heuristic for the split delivery vehicle routing problem. **Computers & Operations Research**, v. 53, p. 234–249, 2015.

SOARES, C. P. B. et al. **Dendrometria e inventário florestal**. Viçosa: UFV, 2011, 272 p.

SOLOMON, M. M. Algorithms for the vehicle routing and scheduling problems with time window constraints. **Operations research**, v. 35, n. 2, p. 254-265, 1987.

SUBRAMANIAN, A. **Heuristic, exact and hybrid approaches for vehicle routing problems**. 2012. 152 f. Tese (Doutorado em Computação) – Curso de Pós-Graduação em Computação, Universidade Federal Fluminense, Niterói, RJ, 2012.

TAN, K. C.; LEE, L. H.; ZHU, Q. L.; OU, K. Heuristic methods for vehicle routing problem with time windows. **Artificial Intelligence in Engineering**. v. 15, p. 281-295, 2001.

TAVAKKOLI-MOGDDAM, R. et al. A new mathematical model for a competitive vehicle routing problem with time windows solved by simulated annealing. **Journal of Manufacturing Systems**. v. 30, p. 83-92, 2011.

TOTH, P.; VIGO, D. Metaheuristics for the VRP. In: TOTH, P.; VIGO, D. (Ed.). **The vehicle routing problem**. Philadelphia: SIAM, 2002, p. 129-154.

VIEIRA, H. P. **Metheurística para a solução de problemas de roteamento de veículos com janela de tempo**. Dissertação (Mestrado em Matemática Aplicada) – Instituto de Matemática, Estatística e Computação Científica - UNICAMP, Campinas, 2013.

WOLSEY, L. A. **Integer programming**. New Jersey: John Wiley & Sons, 1998. 288 p.

## APENDICES

### Apêndice A – Programação das atividades de inventário florestal obtida pela meta-heurística SA para a instância PRV-IF\_2016\_252<sup>1</sup>.

2018					
252					
3					
125460.8					
1					
0	324	1413	0		
0	1704	1419	1826	0	
0	258	0			
0	1273	777	969	0	
0	159	1591	267	0	
0	1579	143	0		
0	265	1404	1581	0	
0	364	877	904	141	0
0	1842	0			
0	1490	1611	630	602	0
0	270	0			
0	1022	1464	0		
0	1822	612	0		
0	1530	1212	0		
0	906	875	1484	1401	0
0	260	0			
0	1423	491	0		
0	1569	0			
0	1562	0			
0	1492	287	0		
0	1783	0			
0	847	0			
0	1973	0			
0	733	0			
0	1252	1025	1137	1474	0
0	1427	780	0		
0	255	1641	1176	0	
0	901	0			
0	746	0			
0	1809	712	0		
0	1126	0			
0	842	1587	0		
0	1476	0			
0	327	0			
0	2001	489	0		
0	1184	0			
0	1266	0			
0	1902	1839	0		

**Apêndice B – Programação das atividades de inventário florestal obtida pela meta-heurística ILS para a instância PRV-IF\_2016\_252<sup>2</sup>.**

2018					
252					
3					
124472.5					
1					
0	845	968	1249	0	
0	1490	903	0		
0	1508	1760	1019	606	0
0	1194	1568	1028	0	
0	489	1530	0		
0	1560	1474	1494	906	0
0	1809	0			
0	491	579	1506	1273	0
0	262	1423	0		
0	1641	1419	1478	0	
0	1839	0			
0	1205	2007	0		
0	1973	0			
0	1126	0			
0	1992	0			
0	1427	1826	0		
0	1415	0			
0	91	0			
0	1476	92	0		
0	1756	1562	0		
0	260	0			
0	413	272	0		
0	1842	0			
0	144	920	1306	0	
0	361	0			
0	258	0			
0	1962	1704	0		
0	1569	0			
0	1137	1182	1413	0	
0	1581	0			
0	95	0			
0	743	0			
0	1401	1573	160	1902	0
0	1421	0			
0	1025	265	0		
0	343	0			
0	139	142	0		
0	1404	0			
0	412	145	0		