

Universidade Federal de Minas Gerais
Escola de Engenharia
Curso de Graduação em Engenharia de Controle e Automação

Planejamento da Colheita de Eucalipto
Uma Abordagem Via Metaheurística

Anilton Cardoso Junior

Orientador: Prof. Lucas Batista, Dr.
Supervisor: Prof. Ronaldo Sampaio, Dr.

Belo Horizonte, Novembro de 2015

Monografia

Planejamento da Colheita de Eucalipto: Uma Abordagem Via Metaheurística

Monografia submetida à banca examinadora designada pelo Colegiado Didático do Curso de Graduação em Engenharia de Controle e Automação da Universidade Federal de Minas Gerais, como parte dos requisitos para aprovação na disciplina Projeto Final de Curso II.

Belo Horizonte, Novembro de 2015

Resumo

RESUMO DO PROJETO

Agradecimentos

Aqui vai o texto dos agradecimentos.

Sumário

Resumo	i
Agradecimentos	iii
Lista de Figuras	vii
Lista de Tabelas	ix
1 Introdução	1
1.1 Motivação e Justificativa	1
1.1.1 A VFL e Demanda de Carvão Vegetal nos Alto-Fornos	1
1.2 Objetivos do Projeto	3
1.3 Contribuições do Projeto	3
1.4 Setores Envolvidos no Projeto	3
1.5 Estrutura da Monografia	4
2 Revisão Bibliográfica	5
2.1 Técnicas de Otimização	5
2.1.1 Dimensionamento de Lotes e Sequenciamento	6
2.1.2 Meta-heurísticas	7
2.1.3 Otimização Multiobjetivo	9
2.1.4 Otimização Robusta	11
2.2 Planejamento Florestal	13
2.3 Conclusão	13
3 Metodologia	14
3.1 Modelagem Matemática do Planejamento de Colheita	14
3.1.1 Características do Problema de Otimização	14
3.1.2 Funções Objetivo	16
3.1.3 Restrições	17
3.2 Algoritmos de Otimização	19
3.2.1 GRASP	19
3.2.2 Análise de Robustez	24
3.2.3 Abordagem Escalar Via Método Epsilon-restrito e GRASP	24
3.3 Estrutura de Testes	25
4 Resultados	26

5 Conclusões	27
Referências Bibliográficas	28

Lista de Figuras

1.1	Disposição das Regionais da VFL	2
2.1	Espaço de Decisão e de Objetivos	9
2.2	Método ϵ -restrito para problema biobjetivo	10
2.3	Exemplo de atualização dos parâmetros ϵ para dois objetivos	11
3.1	Estrutura da Solução	15
3.2	Exemplo de Sequenciamento	15
3.3	Estrutura da Solução considerando plantas de carbonização	16
3.4	Sequenciamento de talhões: prioridade por direção	20
3.5	N_1 – Trocar talhões	22
3.6	N_1 – Trocar Fazendas	22
3.7	N_1 – Troca de períodos	22
3.8	Exemplo de operações de troca para períodos diferentes	23
3.9	N_2 – Deslocar talhão	23
3.10	N_3 – Inverter talhões	23
3.11	N_4 – Inserir talhão	24
3.12	N_5 – Deletar talhão	24
3.13	N_6 – Trocar destino do talhão	24

Lista de Tabelas

2.1	Dimensionamento de Lotes: Variáveis de Decisão	6
2.2	Dimensionamento de Lotes: Parâmetros	6

Capítulo 1

Introdução

Em um mercado cada vez mais competitivo, as empresas necessitam inovar e investir na qualidade e melhoria de seus processos de produção. Nesse quesito, graças ao aumento do poder computacional nas últimas décadas, os algoritmos de otimização ganharam espaço nas indústrias por serem capazes de lidar com modelos que incorporam e tratam um grande número de variáveis, representando, assim, os processos de forma mais fiel e gerando resultados práticos atrativos [33]. Este trabalho visa a aplicação de algoritmos de otimização para aprimorar o sequenciamento do corte de eucalipto da Vallourec Florestal (VFL), empresa produtora de carvão vegetal para uso siderúrgico.

1.1 Motivação e Justificativa

1.1.1 A VFL e Demanda de Carvão Vegetal nos Alto-Fornos

O grupo Vallourec no Brasil produz tubos de aço sem costura para atender o mercado nacional e internacional de soluções tubulares *Premium*. Suas duas usinas no país contam com três alto-fornos para produção de ferro gusa, a matéria prima do aço mais importante para o grupo no Brasil. Esses alto-fornos operam à base energética de carvão vegetal e suas demandas variam dinamicamente com a quantidade solicitada de ferro gusa, operação do forno, carga de minério utilizada e qualidade do próprio carvão.

O fornecedor de carvão para os alto-fornos é a subsidiária do grupo, Vallourec Florestal, que deve estar atenta as demandas de seu cliente, realizando um planejamento efetivo de seus recursos. Contando com mais de 100 mil hectares de floresta de eucalipto divididos em 23 fazendas, a VFL necessita de sistemas de suporte à decisão e de profissionais experientes para conseguir gerir seu complexo processo de produção e suprir toda demanda das usinas.

Introdução ao Problema Gerencial

A VFL é composta de 23 fazendas situadas em três regionais. Juntas possuem mais de 5.000 unidades de floresta conhecidas como talhões. Os talhões são as unidades que serão efetivamente colhidas. Cada um possui, essencialmente, uma área e um volume de madeira de eucalipto em certa idade de crescimento. A disposição das regionais pode ser vista na Figura 1.1.



Figura 1.1: Disposição das Regionais da VFL

Em algumas fazendas, há uma unidade de carbonização – planta de produção de carvão vegetal utilizando madeira seca – que deve atender as demandas mensais de carvão vegetal dos alto-fornos. Por conseguinte, devem-se alocar as máquinas de corte disponíveis para colherem talhões em sequência de forma a atingir as demandas de todas as fazendas. Nesse processo, é necessário respeitar, a princípio, duas restrições. Primeiramente, a capacidade da máquina que consiste em número de horas disponíveis de operação em um mês. Em segundo lugar, deve-se respeitar o tempo de estocagem necessário para a madeira atingir a umidade ideal. Estimativas do tempo de estocagem podem ser obtidas, fundamentalmente, em função do clone de eucalipto e das condições de plantio. Outra característica importante a ser analisada é a homogeneização do diâmetro médio das toras de madeira enviadas à carbonização, que é considerado pela VFL um fator essencial para o bom rendimento de produção de carvão. Além disso, tem-se o aumento no custo de transporte em épocas de chuva, pois todo o tráfego de máquinas é feito em estradas de terra que se deterioram consideravelmente em período chuvoso.

O Planejamento da Colheita

No momento presente, o planejamento da colheita é feito de forma manual e requer cerca de 200 homens-hora semanais. Com frequência, o tempo de estocagem da madeira foge do ideal, além de não se considerar estimativas do diâmetro médio e da umidade das toras enviadas à carbonização, o que resulta em uma heterogeneidade da madeira, comprometendo a qualidade da produção de carvão. Recentemente, tentou-se abordar o problema para instâncias pequenas, apenas 120 talhões, utilizando otimização linear clássica [30]. Porém, o tempo de computação foi demasiadamente longo e o algoritmo não convergiu para o ótimo. Por se tratar de um problema de otimização combinatória NP-completo, ainda não se conhece métodos capazes de encontrar a solução ótima em tempo polinomial. Entretanto, sugere-se que, para esses tipos de problema, os algoritmos meta-heurísticos possam ser mais eficientes, gerando soluções mais atrativas com menor tempo computacional [20]. Dessa forma, propõe-se neste trabalho modelar o processo de sequenciamento da colheita e otimizá-lo de forma satisfatória por meio de meta-heurísticas.

Um bom planejamento da colheita, ao utilizar algoritmos mais eficientes, tem considerável impacto no custo total da VFL, visto que a colheita representa 19% do custo total de produção do carvão vegetal. Além disso, tem repercussão direta no processo de carbonização que necessita de maior homogeneidade da madeira cortada para produção de carvão de melhor qualidade e representa 38% do custo total. Dessa forma, otimizar o sequenciamento do corte não se limita apenas a reduzir custos do processo de colheita em si, mas também da carbonização e aprimora a qualidade do produto final.

1.2 Objetivos do Projeto

Os objetivos do projeto são:

- a. Descrever e modelar o processo de sequenciamento da colheita de eucalipto, relacionando as variáveis mais importantes envolvidas;
- b. Desenvolver meta-heurísticas eficientes capazes de realizar adequadamente o planejamento e sequenciamento da colheita, garantindo a minimização dos custos envolvidos e melhor homogeneização da madeira enviada para as unidades de carbonização.

1.3 Contribuições do Projeto

Algumas possíveis e esperadas contribuições do projeto para com a VFL são:

- a. Elaboração de um modelo matemático mais representativo do processo de colheita, que reflita o contexto prático vivenciado pela empresa;
- b. Implementação de um arcabouço de técnicas de otimização que possibilite encontrar uma solução prática eficiente, com baixo custo computacional;
- c. Redução expressiva do número de homens-horas semanais envolvido no processo de planejamento, e melhor aproveitamento desse recurso em outras áreas;
- d. Redução de custos da colheita por meio de um planejamento eficaz realizado pelos algoritmos desenvolvidos.

1.4 Setores Envolvidos no Projeto

A VFL é subdividida em gerências responsáveis por cada etapa de seu processo de produção de carvão vegetal. Este projeto se destinou ao setor de colheita em si. Entretanto, o planejamento deve examinar as etapas adjacentes em uma visão integrada do processo, ou seja, também são essenciais para o projeto as considerações do setor de silvicultura e carbonização.

1.5 Estrutura da Monografia

Este trabalho está dividido em seis capítulos. Este primeiro aborda o contexto no qual o projeto está inserido. Em seguida, o capítulo dois apresenta uma revisão bibliográfica dos temas fundamentais acerca deste trabalho, incluindo planejamento florestal, a modelagem matemática do problema de otimização e dos algoritmos implementados, estendendo-se por todos conceitos necessários para entendimento do projeto. O capítulo três aborda a metodologia de desenvolvimento, seguido pela implementação do método de otimização e planejamento da colheita, além dos critérios de análises que serão realizados. O capítulo quatro apresenta os resultados obtidos pelo método proposto e pelas análises feitas e o capítulo cinco os discute e compara criticamente com o estado da arte apresentado no capítulo dois. Por fim, o capítulo seis apresenta a conclusão do trabalho, um resumo das contribuições feitas por este projeto e sugestão para trabalhos futuros.

Capítulo 2

Revisão Bibliográfica

Como introduzido na Seção 1.1.1, o planejamento de colheita consiste em escolher os melhores talhões a serem colhidos e sequenciar máquinas de corte de forma a atender as demandas de madeira seca, respeitando as restrições envolvidas. Para solução desse tipo de problema, vasto conhecimento científico foi desenvolvido nas últimas décadas. Dentre esse conhecimento, abrangem-se modelagem matemática de processos e algoritmos de otimização especialistas ou genéricos para diversos casos vivenciados em contextos práticos de indústrias e empresas.

As seguintes seções apresentam os conceitos e o estado-da-arte de estratégias de otimização apropriadas para o tipo de problema proposto e suas aplicações no planejamento florestal.

2.1 Técnicas de Otimização

Otimizar é o ato de obter o melhor resultado sob certas circunstâncias. Matematicamente pode ser definido como o processo de encontrar as condições que maximizam ou minimizam o valor de certa função, definida como função objetivo [33].

A formulação geral dos problemas de otimização, descrita em [3], é dada como:

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimizar}} && \mathbf{f}(\mathbf{x}) \in \mathbb{R}^m \\ & \text{s.a. :} && \begin{cases} g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, p. \\ h_i(\mathbf{x}) = 0, \quad i = 1, \dots, q. \\ \mathbf{x} \in \mathcal{X}. \end{cases} \end{aligned} \quad (2.1)$$

Onde $\mathbf{f}(\mathbf{x})$ são as funções objetivo, $\mathbf{x} \in \mathbb{R}^n$ são as variáveis do problema, $\mathbf{g}(\mathbf{x}) \in \mathbb{R}^p$ são restrições de desigualdade, $\mathbf{h}(\mathbf{x}) \in \mathbb{R}^q$ são restrições de igualdade e \mathcal{X} é o espaço de busca definido pelos limites inferiores e superiores das variáveis de otimização. O problema pode ser mono-objetivo, $m = 1$, ou multiobjetivo, $m > 1$. Técnicas de otimização, então, devem minimizar \mathbf{f} para as variáveis x_1, \dots, x_n , respeitando as restrições de igualdade e desigualdade.

Problemas de sequenciamento são caracterizados como combinatórios [2], i.e. as variáveis de otimização \mathbf{x} assumem valores discretos. Técnicas de otimização combinatória para sequenciamento são descritas a seguir.

2.1.1 Dimensionamento de Lotes e Sequenciamento

No contexto prático de planejamento da produção, alguns problemas podem ser modelados por meio de uma abordagem de dimensionamento de lotes seguida de sequenciamento, utilizando programação linear. O dimensionamento de lotes consiste em transformar a demanda do cliente em lotes, i.e. ordens de produção, com objetivo de minimizar custos envolvendo máquinas, produtos e estoques [14], [19].

Tabela 2.1: Dimensionamento de Lotes: Variáveis de Decisão

Variável	Definição
X_t	Quantidade produzida no período t
I_t	Quantidade de estoque no final do período t
Y_t	Variável binária que indica se ocorreu preparação de máquina no período t

O dimensionamento de lotes para um único produto e sem restrições de capacidade é formalmente definido como um problema de otimização, sendo as variáveis de decisão definidas na Tabela 2.1 e os parâmetros na Tabela 2.2, pela Equação 2.2 [9].

Tabela 2.2: Dimensionamento de Lotes: Parâmetros

Parâmetro	Definição
d_t	Demanda do produto no período t
p_t	Custo de produção de uma unidade no período t
s_t	Custo de preparação de máquina no período t
h_t	Custo de estoque no período t

$$\text{Minimizar} \quad \sum_{t=1}^T (s_t Y_t + p_t X_t + h_t I_t) \quad (2.2)$$

s.a.:

$$I_{t-1} + X_t = d_t + I_t \quad \forall t, \quad (2.3a)$$

$$X_t \leq Y_t d_{tT} \quad \forall t, \quad (2.3b)$$

$$X_t, I_t \geq 0, Y_t \in \{0, 1\} \quad \forall t. \quad (2.3c)$$

Sendo o período de planejamento $t = 1, \dots, T$, $d_{tT} = \sum_{k=t}^T d_k$, o somatório de todas demandas futuras em relação ao período atual t . A Função Objetivo 2.2 contabiliza os custos de produção, estoque e preparação de máquina. A Restrição 2.3a realiza o balanço de estoque, considerando o estoque no período anterior $t - 1$ e a produção atual X_t para atenderem a demanda d_t . O restante é armazenado no estoque do fim do período atual I_t . A Restrição 2.3b limita a produção atual X_t e a relaciona com a variável binária Y_t . Finalmente, a Restrição 2.3c define X_t e I_t como não-negativos e Y_t como binário.

Após o dimensionamento de lotes, tem-se a etapa de sequenciamento de n jobs, i.e. tarefas, [10] sujeito a restrições de tempo e objetivando a minimização dos custos envolvidos. Além disso, a função objetivo pode priorizar tarefas e penalizar atrasos caso essas tenham prazo para serem processadas.

Em [31], diversas variantes de problemas de sequenciamento são descritas. Tem-se para máquinas paralelas não-relacionadas, isto é, máquinas com diferentes velocidades de processamento para cada tarefa, o seguinte modelo.

$$\text{Minimizar} \quad \sum_{i=1}^m \sum_{j=1}^n \sum_{k=1}^n k p_{ij} x_{ikj} \quad (2.4)$$

s.a.:

$$\sum_{i=1}^m \sum_{k=1}^n x_{ikj} = 1 \quad \forall j, \quad (2.5a)$$

$$\sum_{j=1}^n x_{ikj} \leq 1 \quad \forall i, k, \quad (2.5b)$$

$$x_{ikj} \in \{0, 1\}. \quad (2.5c)$$

Sendo $i = 1, \dots, m$ as máquinas paralelas, $j = 1, \dots, n$ as tarefas a serem processadas, x_{ikj} uma variável binária que assume valor 1 se a tarefa j é a k -ésima tarefa processada na máquina i e $k p_{ij}$ é o tempo de processamento da tarefa j na máquina i . A Função Objetivo 2.4 contabiliza o tempo total do sequenciamento. A Restrição 2.5a garante que cada tarefa é processada somente uma vez. Já a Restrição 2.5b assegura que cada máquina não processe tarefas simultaneamente. Finalmente, a Restrição 2.5c define x_{ikj} como variável binária.

Outras abordagens utilizando heurísticas, que consideram m máquinas paralelas, são propostas em [28] e [23]. Este também cita a ineficácia de técnicas de programação inteira e métodos *branch-and-bound* para instâncias maiores, em virtude de muitos problemas de sequenciamento serem caracterizados como NP-completo.

Um modelo de dimensionamento de lotes, considerando restrições de capacidade e máquinas paralelas, foi proposto em [32] e apresentado em [30] como um método capaz de escolher os melhores talhões a serem colhidos em um horizonte de planejamento. Este também apresenta um método exato integrando dimensionamento de lotes e sequenciamento implementado para VFL. Entretanto, o método desenvolvido não é utilizado na prática por demandar tempo excessivo de computação, inclusive para instâncias pequenas de 120 talhões.

2.1.2 Meta-heurísticas

Problemas combinatórios e NP-completos, como o sequenciamento de colheita [39], são improváveis de serem resolvidos em tempo ‘razoável’ por métodos exatos [8]. Dessa forma, heurísticas foram desenvolvidas para encontrar soluções próximas do ótimo global com menor tempo de computação. Entretanto, estas são especializadas a tipos específicos de problemas. Para contornar esta desvantagem, as meta-heurísticas agregam qualidades de ambas abordagens, sendo genéricas para problemas discretos e fornecendo soluções razoáveis com baixo tempo computacional, devido a sua natureza, em geral, estocástica e também pela incorporação de outras heurísticas [13].

Em [37] é possível notar o aumento no uso de meta-heurísticas nos últimos anos, graças a eficiência e flexibilidade dessas estratégias. Dentre esses métodos, pode-se citar os métodos *multi-start* como o *Greedy Randomized Adaptive Search Procedure* (GRASP). Proposto em [16] e [17], o GRASP é uma meta-heurística bastante utilizada em sequenciamento [18].

GRASP

O GRASP é basicamente dividido em duas etapas, sendo a primeira de construção, seguida pela etapa de busca local. A estrutura geral do método é apresentada na Algoritmo 1.

Algoritmo 1 Meta-heurística GRASP

- 1: **repete**
 - 2: Construir solução factível utilizando heurística gulosa
 - 3: Aplicar busca local à solução construída
 - 4: **até** Condição de parada satisfeita
 - 5: **devolve** Melhor solução
-

Na fase de construção do GRASP, uma solução é construída iterativamente. A cada passo, elementos são incorporados até que a solução esteja completa e factível, i.e. nenhuma restrição é violada. Para se adicionar um elemento, uma lista de possíveis candidatos (RCL, *restricted candidate list*) é construída e ordenada de acordo com alguma função gulosa. O elemento a ser inserido na solução parcialmente construída é selecionado aleatoriamente da RCL. A RCL pode ser limitada pelo número de elementos ou pelo valor da função gulosa. Neste último caso, apenas elementos com custo menor do que $c_{\min} + \alpha(c_{\max} - c_{\min})$ fazem parte da RCL, sendo c_{\min} e c_{\max} os valores mínimo e máximo da função gulosa, respectivamente. Finalmente, α representa um compromisso entre diversidade e intensificação da construção.

Na etapa de busca local, outras meta-heurísticas podem ser utilizadas [5] ou até mesmo algoritmos exatos híbridos chamados de *matheuristics* [36]. Uma possível meta-heurística, utilizada como busca local, é o *Variable Neighborhood Search* (VNS) [35]. O VNS tem sido amplamente utilizado em sequenciamento e apresenta resultados satisfatórios [12].

Essa meta-heurística consiste em explorar dinamicamente a vizinhança de uma solução [8]. Inicialmente, são definidas estruturas de vizinhança $(N_1, \dots, N_{n_{\max}})$ que exploram vizinhos cada vez mais distantes em seu ciclo. Esse ciclo consiste em 3 fases. A primeira, chamada de *shake*, é uma perturbação, utilizando uma estrutura de vizinhança N_n , aplicada à solução. Em seguida, ocorre a fase de busca local que se utiliza de qualquer estrutura de vizinhança para encontrar um ótimo local. Se a solução encontrada é melhor do que a inicial, então n passa a ser igual a 1 e o processo se reinicia. Senão o algoritmo move para a próxima estrutura de vizinhança $n + 1$. Essa, então, é a terceira fase do VNS. A estrutura básica desse método pode ser vista no Algoritmo 2.

O GRASP ainda pode ser aprimorado aplicando diversas outras técnicas. Dentre elas, pode-se citar a Reconexão por Caminhos que se resume em utilizar informação das melhores soluções encontradas para criar uma nova superior e o GRASP Reativo que iterativamente adapta o parâmetro α da construção gulosa com base na qualidade das soluções construídas [7].

Em [21] podem ser vistos diversos artigos com as aplicabilidades do GRASP e do VNS em problemas de roteamento de veículos, sequenciamento e outros.

Algoritmo 2 Meta-heurística VNS

```

1: procedimento VNS( $x, k_{max}$ )
2:    $k \leftarrow 1$ 
3:   repete
4:      $\hat{x} \leftarrow \text{SHAKE}(x, k)$ 
5:      $\hat{\hat{x}} \leftarrow \text{BUSCALOCAL}(\hat{x})$ 
6:     se  $f(\hat{\hat{x}}) < f(x)$  então
7:        $k \leftarrow 1$ 
8:        $x \leftarrow \hat{\hat{x}}$ 
9:     senão
10:       $k \leftarrow k + 1$ 
11:    fim se
12:  até  $k = k_{max}$ 
13: fim procedimento

```

2.1.3 Otimização Multiobjetivo

Problemas reais, em geral, possuem mais de um objetivo, e.g. reduzir custos e maximizar capacidade de carga em projetos de aeronaves ou minimizar distância percorrida e o número de veículos em problemas de roteamento. Nessa categoria de problemas, tem-se m funções objetivo, em geral contraditórias, que devem ser minimizadas/maximizadas simultaneamente. Dessa maneira, tem-se um conjunto de soluções de compromisso, *tradeoff* entre os objetivos, para esse tipo de problema que é, normalmente, grande e em alguns casos, infinito.

Para se comparar soluções, dados muitos objetivos, define-se as relações de Dominância Pareto. À vista disso, uma solução \mathbf{x}_1 Pareto domina \mathbf{x}_2 , ou simplesmente $\mathbf{x}_1 \prec \mathbf{x}_2$, se $\mathbf{f}(\mathbf{x}_1) \leq \mathbf{f}(\mathbf{x}_2)$ e $\mathbf{f}(\mathbf{x}_1) \neq \mathbf{f}(\mathbf{x}_2)$. Por outro lado, soluções são não dominadas entre si, ou incomparáveis, quando $\mathbf{x}_1 \not\prec \mathbf{x}_2$ e $\mathbf{x}_2 \not\prec \mathbf{x}_1$ [11].

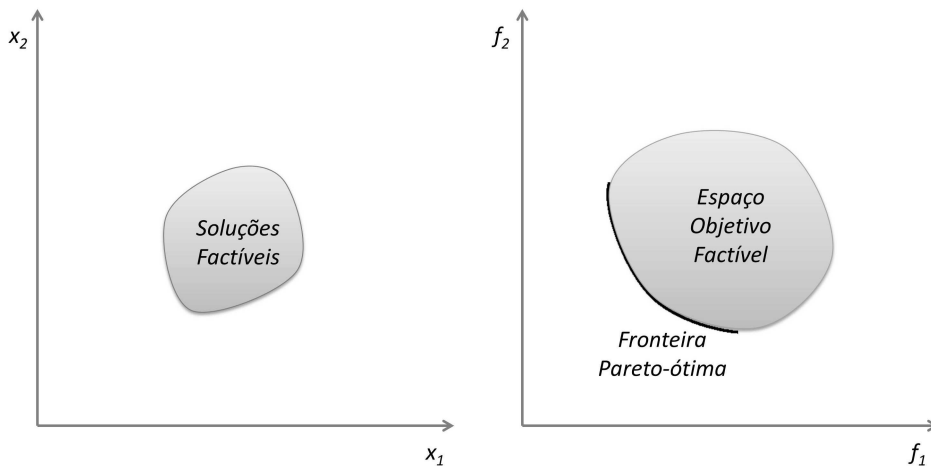


Figura 2.1: Espaço de Decisão e de Objetivos

Dadas as relações de dominância, pode-se definir o conjunto de soluções Pareto-ótimo, i.e. o conjunto de soluções não dominadas, como $\mathcal{P} = \{\mathbf{x}^* \in \mathcal{F} \mid \nexists \mathbf{x} \in \mathcal{F} \text{ tal que } \mathbf{f}(\mathbf{x}) \prec \mathbf{f}(\mathbf{x}^*)\}$. Por consequência à definição anterior, tem-se o conjunto Fronteira Pareto-ótima \mathcal{S}

que se refere a imagem do conjunto Pareto-ótimo global, sendo $\mathcal{S} = \{\mathbf{y} = \mathbf{f}(\mathbf{x}) : \mathbf{x} \in \mathcal{P}\}$. Para apenas dois objetivos e duas variáveis de decisão, pode-se ver um exemplo de um conjunto de soluções factíveis e da Fronteira Pareto-ótima na Figura 2.1.

Entretanto, a relação de dominância Pareto não oferece flexibilidade para inclusão de preferências do tomador de decisão, aquele que deve escolher, dentre as soluções Pareto-ótimas, qual de fato será implementada, ou determinar a resolução desejada para o conjunto Pareto-ótimo. Para contornar esse problema, pode-se utilizar a relação ϵ -dominância, uma simples modificação da dominância Pareto [26]. Dessa maneira, \mathbf{x}_1 ϵ -domina \mathbf{x}_2 se $f_i(\mathbf{x}_1) - \epsilon_i \leq f_i(\mathbf{x}_2) \quad \forall i \in 1, \dots, m$ e $\epsilon_i > 0$. Essa relação é escrita como $\mathbf{x}_1 \prec_{\epsilon} \mathbf{x}_2$ e permite regular a resolução do conjunto Pareto estimado e, simultaneamente, a convergência e diversidade das soluções encontradas.

Método ϵ -restrito

Uma técnica comum para otimização multiobjetivo é o método ϵ -restrito. Por esse método, o problema multiobjetivo é transformado em mono-objetivo com restrições adicionais [15].

Basicamente, o ϵ -restrito consiste em minimizar um objetivo, sendo os demais considerados como restrições de desigualdade. Assim, tem-se o problema de otimização modificado dado pela Formulação 2.6. Para o caso biobjetivo, representado pela Figura 2.2, tanto soluções Pareto-ótimas globais quanto locais podem ser encontradas.

$$\begin{aligned} & \underset{\mathbf{x}}{\text{minimizar}} && f_k(\mathbf{x}) \in \mathbb{R} \\ & \text{s.a. :} && \begin{cases} f_i(\mathbf{x}) \leq \epsilon_i, \quad i = 1, \dots, m \text{ e } i \neq k. \\ g_i(\mathbf{x}) \leq 0, \quad i = 1, \dots, p. \\ h_i(\mathbf{x}) = 0, \quad i = 1, \dots, q. \\ \mathbf{x} \in X. \end{cases} \end{aligned} \quad (2.6)$$

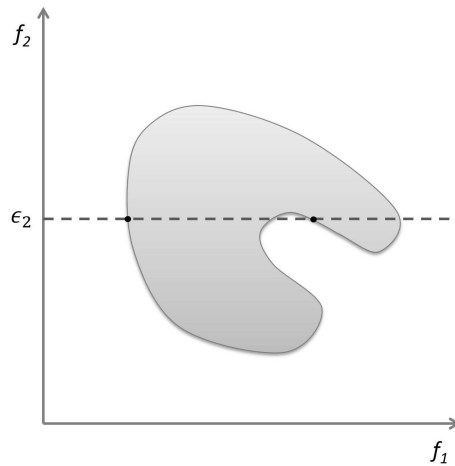


Figura 2.2: Método ϵ -restrito para problema biobjetivo

Pelo método, é possível encontrar qualquer solução do conjunto Pareto-ótimo. Sendo assim, ao variar sistematicamente os parâmetros ϵ_i , pode-se obter um conjunto representativo dessas soluções e mapear a Fronteira Pareto-ótima. Portanto, a variação dos parâmetros deve ser precisa o suficiente para não se perder soluções desse conjunto. Entretanto, isso

pode resultar em um grande número de iterações redundantes ao obter soluções que não são significativamente diferentes. Em [27], são descritas metodologias eficientes para variação dos parâmetros do método ϵ -restrito, utilizando informação das soluções já encontradas e, à vista disso, obter a Fronteira Pareto-ótima mais rapidamente.

Nessa metodologia, as funções objetivo são transformadas em restrições de desigualdade do tipo $\epsilon_i \leq f_i(\mathbf{x}) \leq \hat{\epsilon}_i$. É preciso, então, definir limites inferiores e superiores para todos $m - 1$ objetivos. Apesar de possuir o dobro de parâmetros que a formulação original, a metodologia referida garante convergência mais eficiente para encontrar o conjunto Pareto-ótimo. Inicialmente, todas as restrições são definidas no intervalo $(-\infty, \infty)$ e o conjunto Pareto está vazio $\mathcal{P} = \emptyset$. A medida que soluções são adicionadas ao conjunto Pareto, os limites inferiores e superiores das restrições são atualizados. A Figura 2.3 exemplifica a metodologia para um problema biobjetivo, sendo f_2 transformada em restrição, com duas soluções Pareto já encontradas. Essas soluções agora delimitam os novos limites inferiores e superiores para f_2 , sendo $(\epsilon_{2,1}, \infty)$, $(\epsilon_{2,2}, \epsilon_{2,1})$ e $(-\infty, \epsilon_{2,2})$ nos blocos 1, 2 e 3, respectivamente. Ao adicionar uma nova solução x_3 ao conjunto Pareto, novos blocos são criados e, portanto, o algoritmo somente finalizará sua execução quando percorrer todos os blocos criados e não encontrar mais nenhuma solução não dominada.

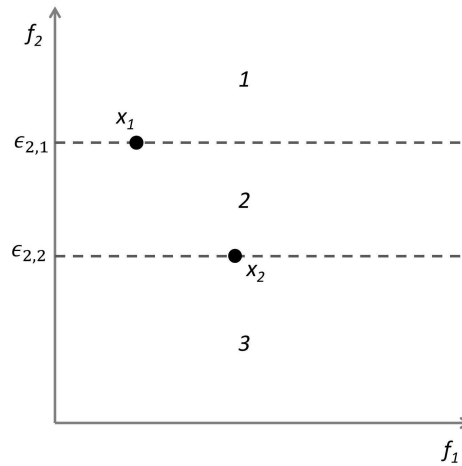


Figura 2.3: Exemplo de atualização dos parâmetros ϵ para dois objetivos

2.1.4 Otimização Robusta

No âmbito prático, sistemas reais possuem diversas fontes de incerteza, como comentado em [29] para problemas de planejamento florestal. Em [24] são discutidas diferentes categorias de incertezas em computação evolucionária e respectivas abordagens para contornar esses problemas.

Em primeira análise, a robustez de uma solução pode ser definida em termos da sensibilidade da função objetivo em relação à perturbações:

- a. Das variáveis de otimização;
- b. Dos parâmetros;
- c. Da própria função objetivo.

Considerando a definição acima, em [24] são apresentadas metodologias para busca de soluções robustas. Em uma possível abordagem, os algoritmos otimizam o valor esperado da função objetivo, dada uma distribuição de probabilidade de perturbações, como expresso pela Equação 2.7.

$$F(x) = \int_{-\infty}^{\infty} f(x + \delta)p(\delta)d\delta \quad (2.7)$$

Sendo f a função objetivo original do problema, F a nova função objetivo a ser considerada nos algoritmos de otimização, $p(\delta)$ a distribuição de probabilidade dos possíveis distúrbios δ .

Em outro caso, considera-se uma abordagem multiobjetivo quando a Equação 2.7 não se mostra suficiente para encontrar soluções robustas, pois variações positivas e negativas da função objetivo podem se cancelar em certos cenários. Consequentemente, soluções com função objetivo de alta variância são erroneamente consideradas robustas. Portanto, o valor esperado, descrito anteriormente, a média e o desvio padrão da função objetivo podem, também, ser considerados objetivos diferentes [34], resultando em soluções de compromisso entre desempenho e robustez.

Em [38], é descrito um *framework* para se encontrar soluções robustas utilizando meta-heurísticas. Este define funções de avaliação de robustez como a Equação 2.8.

$$\hat{f}(\xi) = \sum_{i=1}^{n_e} \omega_i f(S_i(\xi), S'_i(\pi)) \quad (2.8)$$

Sendo $f(\xi, \pi)$ a função objetivo, ξ é uma solução com variáveis de otimização estocásticas, π são os parâmetros estocásticos de f , S_i e S'_i são funções de amostragem de ξ e π , respectivamente, n_e é a quantidade de testes a serem realizados em f , ω são pesos para realizar a média ponderada de f frente aos diferentes cenários.

Algoritmo 3 Obter Soluções Robustas

- 1: Gerar soluções diversas utilizando métodos de otimização
 - 2: Avaliar soluções com a Função de Avaliação de Robustez 2.8
 - 3: **devolve** Solução mais robusta
-

Soluções robustas podem ser encontradas seguindo o Algoritmo 3. Uma alternativa à Equação 2.8, é avaliar o valor da função objetivo para todos os testes n_e e considerar apenas o pior caso. Assim, a solução robusta escolhida seria aquela com melhor desempenho no pior caso.

Além de robustez em relação a variação da função objetivo, pode-se também analisar a robustez de uma solução relacionada a sua factibilidade, i.e. uma solução não viola restrições frente as incertezas do sistema que podem tanto alterar a solução quanto as restrições do modelo.

Uma outra importante característica de uma solução é sua flexibilidade. Em [38], define-se a flexibilidade de uma solução como sua relativa facilidade em ser adaptada, por uma rotina de reparo, devido a mudanças dos requisitos do problema de otimização. Dessa forma, uma solução pode ser flexível e robusta se a solução original e a reparada estão ‘próximas’ no espaço de soluções, i.e. as duas têm alto grau de similaridade.

2.2 Planejamento Florestal

A aplicação de técnicas de otimização no planejamento florestal tem sido amplamente estudada nos últimos anos. Em [2], são descritos métodos e abordagens comuns com restrições e variáveis espaciais para planejamento florestal, como custos envolvendo construção e manutenção de estradas, restrições de adjacência entre áreas colhidas e restrições ambientais. Esse artigo também se refere a ineficácia em otimizar planejamento florestal e sequenciamento de colheita utilizando métodos exatos, devido à incorporação de novas restrições, resultando em problemas maiores, e não-linearidades provindas de problemas cada vez mais complexos.

Em [25], o planejamento florestal é discriminado em estratégico, tático e operacional sendo planejamentos de longo, médio e curto prazo, respectivamente. No artigo também é descrito o planejamento anual – operacional – de colheita, similar ao problema proposto neste trabalho, envolvendo alta complexidade como múltiplas máquinas, fatores climáticos, custos de estradas, estoque e entrega da madeira para unidades de processamento. Além desse artigo, muitos outros foram escritos para planejamento de colheita, como a aplicação de modelo linear inteiro misto [1] e algoritmo genético [22].

Problemas de robustez no planejamento de colheita podem ser vistos em [6] e [29]. Além disso, uma metodologia para validação de heurísticas aplicadas ao planejamento florestal é descrita em [4].

2.3 Conclusão

Tendo em vista a alta complexidade e o grande número de variáveis envolvidos no planejamento de colheita descrito em [25], os métodos exatos demandam excessivo tempo de computação para convergirem, o que os tornam inviáveis para planejamento de curto prazo. Dessa forma, meta-heurísticas são mais apropriadas, apesar de não garantirem otimização global, por serem eficientes e flexíveis para incorporar diversas restrições, e.g. restrições ambientais, espaciais e de capacidade e outros objetivos, como a qualidade da madeira cortada e a robustez das soluções.

As meta-heurísticas escolhidas para solução do problema proposto foram o GRASP e VNS descritos na Seção 2.1.2. Esses métodos já foram validados em problemas de sequenciamento e são expansíveis com simplicidade para casos multiobjetivo. Além disso, são versáteis para agregar outras heurísticas, que os tornam especialistas no problema proposto, e escaláveis caso, futuramente, deseje-se acrescentar novas variáveis.

Capítulo 3

Metodologia

Neste capítulo, discute-se a abordagem escolhida para solucionar o problema proposto. Primeiramente, é descrita toda modelagem matemática necessária do planejamento de colheita da VFL. A seguir, o algoritmo de otimização é exposto, juntamente com todas as técnicas utilizadas. Finalmente, todos os testes e avaliações necessárias para validação da metodologia são apresentados.

3.1 Modelagem Matemática do Planejamento de Colheita

O planejamento de colheita pode ser modelado como dimensionamento de lotes seguido de sequenciamento, como apresentado na Seção 2.1.1. Sendo assim, são consideradas características de ambos modelos para uma melhor descrição do problema proposto. A modelagem, então, é apresentada nas próximas seções.

3.1.1 Características do Problema de Otimização

O planejamento de colheita consiste em sequenciar M máquinas para percorrerem T talhões e, por conseguinte, atender a quantidade de madeira requerida pelas plantas de carbonização que, por sua vez, devem atender as demandas de carvão vegetal dos alto-fornos. Cada máquina m está sujeita a uma limitação de capacidade C_m^p em cada período p . Os planejadores de colheita, então, devem distribuir as máquinas entre as R regionais e sequenciá-las em um plano de corte de curto prazo.

Sendo C a quantidade de plantas de carbonização, cada uma possui demanda D_c^p de madeira no período p . O conjunto de fazendas F de todas regionais R , então, deve fornecer a madeira suficiente para atender o conjunto de demandas D^p de todas plantas de carbonização em todos os períodos $1, \dots, P$.

À vista disso, a estrutura adotada para representar a solução deste problema de otimização é apresentada na Figura 3.1. As setas indicam o sentido de movimento da máquina. Além disso, o índice i , em cada talhão T_i , indica a posição do talhão no sequenciamento. Para cada máquina m , portanto, tem-se uma lista de talhões a ser percorrida.

Um exemplo de sequenciamento entre fazendas diferentes pode ser visto na Figura 3.2. Neste caso, pode-se constatar que a fazenda F_2 não possui floresta disponível para corte e, logo, as fazendas mais próximas devem fornecer madeira para sua planta de carbonização. Portanto, em cada talhão deve ser especificada para qual planta de carbonização este se

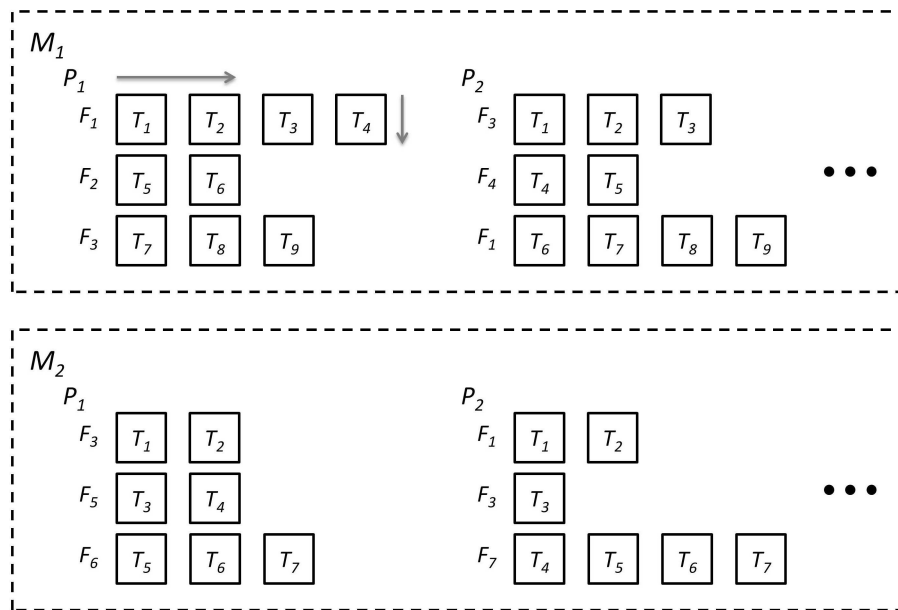


Figura 3.1: Estrutura da Solução

destina e a estrutura de solução deve, também, representar as plantas como mostra a Figura 3.3.

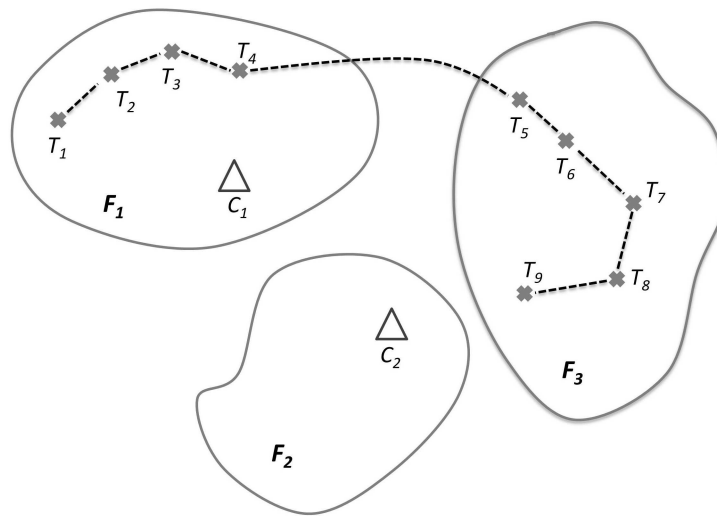


Figura 3.2: Exemplo de Sequenciamento

Após o corte, a madeira deve permanecer em estoque para secagem. Este tempo depende essencialmente do clone do eucalipto e das condições climática enfrentadas pelos talhões. Deste modo, é importante destacar que os talhões cortados no período p serão apenas transportados para a planta de carbonização após sua permanência em estoque. Além disso, um talhão pode ser transportado para a planta em múltiplos períodos consecutivos.

Sendo assim, o método de otimização aplicado deve ser capaz de obter a melhor configuração de talhões – sua ordem e, também, o destino – para cada máquina, respeitando todas

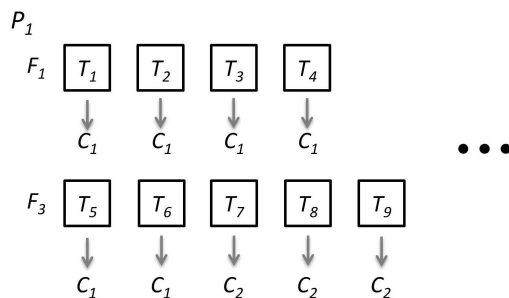


Figura 3.3: Estrutura da Solução considerando plantas de carbonização

restrições envolvidas.

3.1.2 Funções Objetivo

Nesta seção, são definidas funções matemáticas, ou funções objetivo, que têm por finalidade analisar e ranquear soluções quantitativamente. Sendo o problema de planejamento da colheita naturalmente multiobjetivo, são definidas mais de uma função objetivo segundo os critérios mais importantes para colheita.

Custos de Máquina

Em primeira análise, tem-se o custo total de uso das máquinas, para percorrerem os talhões sequenciados, definido pela Equação 3.1. Como a finalidade deste objetivo é quantificar o custo relacionado à transição das máquinas, são desconsiderados custos envolvendo o corte dos talhões em si, apesar de existirem máquinas com rendimentos e custos de corte diferentes.

$$F_1([T^1, \dots, T^M]) = \sum_{m=1}^M \sum_{p=1}^P \sum_{i=1}^{I^{m,p}-1} [d(T_i^{m,p}, T_{i+1}^{m,p}) K^{m,p}] \quad (3.1)$$

Onde T^i, \dots, T^M são vetores de talhões alocados para cada máquina para todos períodos, $I^{m,p}$ é a quantidade de talhões alocados para máquina m no período p , $d(T_i^{m,p}, T_{i+1}^{m,p})$ é a distância Euclidiana em metros entre talhões $T_i^{m,p}$ e $T_{i+1}^{m,p}$ percorridos pela máquina m no período p , $K^{m,p}$ é o custo por metro percorrido para cada máquina m no período p . Logo, a função objetivo F_1 resulta em custos em reais para transição de máquinas.

Distância em Relação às Plantas de Carbonização

O transporte de madeira para as plantas de carbonização é precário em período de chuvas, pois grande parte das estradas disponíveis são de terra. Portanto, o custo de transporte da madeira nesses meses é mais caro em relação ao período de seca. Para quantificar essa característica do problema de planejamento de colheita, tem-se a Equação 3.2.

$$F_2([T^1, \dots, T^M]) = \sum_{m=1}^M \sum_{p=1}^P \sum_{i=1}^{I^{m,p}} [d(T_i^{m,p}, C_{T_i^{m,p}}) F_K(\mathcal{L}(T_i^{m,p}))] \quad (3.2)$$

Onde $d(T_i^{m,p}, C_{T_i^{m,p}})$ é a distância Euclidiana em metros entre o talhão $T_i^{m,p}$ e a planta de carbonização $C_{T_i^{m,p}}$ a qual se destina sua madeira, $\mathcal{L}(T_i^{m,p})$ é uma função que retorna as quantidades de madeira do correspondente talhão que serão transportadas em cada período para a planta de carbonização, $F_K(\cdot)$ é o custo por metro de se transportar a madeira para a planta de carbonização baseando-se nos períodos e quantidades de madeira transportadas, e não no período de corte p . Assim, F_2 representa o custo total em reais de transporte de madeira para as plantas de carbonização.

Homogeneização da Madeira

A homogeneidade da madeira enviada à carbonização é considerada um dos fatores mais importantes para eficiência do processo. Madeira com distribuição de diâmetros dispersa, i.e. com alto desvio padrão, causa atrasos no processo de carbonização, gera maior quantidade de perdas e resulta em carvão vegetal de baixa qualidade. Assim, é importante para a VFL que o setor de colheita forneça madeira homogênea para as plantas de carbonização. Entretanto, isso não é considerado, atualmente, ao realizar um planejamento de corte. A Equação 3.3 quantifica a homogeneidade da madeira ao computar o desvio padrão dos diâmetros em todos os períodos.

$$F_3([T^1, \dots, T^M]) = \sum_{c=1}^C \sum_{m=1}^M \sum_{p=1}^P \sigma(T^m, p, c) \quad (3.3)$$

Onde C é a quantidade total de plantas de carbonização e $\sigma(T^m, p, c)$ é uma função que retorna o desvio padrão da madeira colhida pela máquina m que será entregue a planta c no período p . Note-se que a variável de talhões sequenciados T^m para máquina m não está indexada pelo período p . Sendo assim, p não significa o período em que os talhões foram cortados, mas sim o período em que eles serão transportados para a planta de carbonização. Logo, F_3 retorna a soma de todos desvios padrão de madeira enviadas à carbonização.

3.1.3 Restrições

Uma vez definida todas as funções objetivo, os métodos de otimização devem minimizá-las, porém respeitando algumas restrições impostas. Essas restrições são reais limitações enfrentadas pela equipe de planejamento da colheita e, assim, garantem que o modelo resulte em soluções que possam ser utilizadas na prática.

Capacidade

As máquinas de corte possuem limitação de horas de trabalho em um mês, dependendo de diversos fatores estocásticos como, por exemplo, o tempo gasto em manutenção. Assim, a equipe de colheita define dois parâmetros principais para mensurarem o tempo total disponível das máquinas por mês. Primeiramente, tem-se a eficiência operacional, $\mathcal{EO} \in \{0, 1\}$, que mede a fração de tempo em que a máquina pode estar em trabalho, considerando horário de almoço de funcionários, tempo de estagnação em troca de turno, atrasos e outros. Em segundo lugar, tem-se a disponibilidade mecânica, $\mathcal{DM} \in \{0, 1\}$, que se refere a fração

de tempo em que a máquina não está em manutenção. Esses dois parâmetros são estipulados, pela equipe de planejamento, de acordo com dados históricos e, dessa forma, oferecem estimativas médias.

Dessa forma, a capacidade média $\bar{C}^{m,p}$ de uma máquina m no período p pode ser calculada pela equação 3.4.

$$\bar{C}^{m,p} = C_{\max}^{m,p} \mathcal{EO}^{m,p} \mathcal{DM}^{m,p} \quad (3.4)$$

Onde $C_{\max}^{m,p}$ é a quantidade total de horas de trabalho da máquina m no período p que depende do número de turnos trabalhos, a quantidade de horas por turno e do número de dias de trabalho no período p , $\mathcal{EO}^{m,p}$ e $\mathcal{DM}^{m,p}$ são a eficiência operacional e a disponibilidade mecânica da máquina m no período p , respectivamente.

Porém, como os parâmetros são valores médios e estimados, é melhor definir a capacidade de máquina disponível para trabalho em um valor em torno da média calculada na Equação 3.4. Por conseguinte, define-se a restrição de capacidade como um intervalo dado pela Equação 3.5.

$$\bar{C}^{m,p}(1 - \delta_c) \leq C^{m,p} \leq \bar{C}^{m,p}(1 + \delta_c) \quad (3.5)$$

Onde $C^{m,p}$ é a capacidade utilizada pela máquina m no período p e $\delta_c \in [0, 1]$ é uma estimativa da dispersão que a capacidade disponível tem em torno de sua média.

Demanda

Cada planta de carbonização possui, fundamentalmente, demandas mensais de madeira a ser carbonizada e, portanto, o plano de corte deve garantir que todas essas sejam atingidas. Sendo assim, tem-se a restrição de demanda dada pela Equação 3.6.

$$g_d(\mathbf{T}) = \sum_{m=1}^M \mathcal{V}(\mathbf{T}^m, p, c) = \mathcal{D}^{c,p} \quad (3.6)$$

Sendo $\mathcal{D}^{c,p}$ a demanda de madeira da planta c no período p , $\mathcal{V}(\mathbf{T}^m, p, c)$ uma função que retorna a quantidade de madeira colhida pela máquina m que será transportada no período p para a planta de carbonização c .

Estoque

A madeira, após colhida, deve permanecer um período de tempo em estoque para que se seque até o nível ideal de umidade, a fim de ser carbonizada mais eficientemente. O tempo mínimo de estoque é definido pelas características do clone de eucalipto e das condições climáticas, sendo estimado para cada talhão. Há também necessidade de se definir um tempo máximo em estoque, quando a madeira passa a apodrecer e perder sua qualidade para a carbonização. Logo, tem-se a restrição de estoque dada pelas Equações 3.7a, 3.7b.

$$\min_p(\mathcal{L}(T_i^{m,p})) \geq \mathcal{E}_{\min}(T_i^{m,p}) \quad \forall i, m, p \quad (3.7a)$$

$$\max_p(\mathcal{L}(T_i^{m,p})) \leq \mathcal{E}_{\max}(T_i^{m,p}) \quad \forall i, m, p \quad (3.7b)$$

Onde $\mathcal{E}_{\min}(T_i^{m,p})$ e $\mathcal{E}_{\max}(T_i^{m,p})$ são funções que, para cada talhão cortado $T_i^{m,p}$, retornam o tempo mínimo e máximo de estoque, respectivamente, $\mathcal{L}(T_i^{m,p})$ a função que retorna as quantidades de madeira do correspondente talhão transportadas para a planta de carbonização em cada período e as funções \min_p e \max_p retornam o primeiro e último período em que a madeira será enviada para as plantas, respectivamente.

Sabe-se, também, que não há tempo de estoque menor do que quatro meses para todas as fazendas da VFL. Portanto, para se atender as demandas de madeira no período p , o corte deve ocorrer no máximo até $p - 4$, porém o tempo exato de estoque será determinado pela Equação 3.7a.

3.2 Algoritmos de Otimização

Tendo em vista a formulação descrita na Seção 3.1, escolheu-se o GRASP em conjunto com VNS como método de otimização, sabendo que essas meta-heurísticas foram bem sucedidas para problemas semelhantes na literatura e que possuem confluência com a estrutura de solução adotada.

3.2.1 GRASP

Em relação ao GRASP é importante notar sua generalidade e facilidade para lidar com diversos tipos de problemas de otimização. É possível, também, incluir características específicas dos problemas por meio de heurísticas de construção e de busca local, o deixando especialista e mais eficiente no problema a ser resolvido. Portanto, é importante detalhar cada uma dessas etapas a fim de que seja possível retratar os aspectos mais relevantes da metodologia desenvolvida.

Construção Gulosa

A etapa de construção gulosa, como já descrita, consiste em adicionar iterativamente elementos à solução até que esta se torne factível. Dessa forma, talhões são acrescentados à solução até que as restrições de capacidade (3.1.3), demanda (3.1.3) e estoque (3.1.3) sejam respeitadas. A escolha desses talhões é feita baseando-se numa lista de candidatos, RCL, que é construída por meio de uma simples adaptação das funções objetivo definidas pelas Equações 3.1, 3.2 e 3.3.

Inicialmente, a heurística de construção detecta, para cada período, quais são as fazendas mais críticas – fazendas que não possuem estoque suficiente para atender o quarto mês após o período de corte, pois, como discutido na Seção 3.1.3, o menor tempo possível para estoque são quatro meses. Define-se, também, a distribuição de máquinas entre as regionais para que as demandas de todas as plantas de carbonização possam ser atendidas. A seguir, então, é feito um sequenciamento dessas fazendas e de talhões em cada uma das regionais. Dessa maneira, o sequenciamento é feito em dois níveis: primeiramente as fazendas e depois os talhões. As fazendas são sequenciadas numa abordagem gulosa baseada na distância Euclidiana entre seus centros de massa. O centro de massa, calculado para uma fazenda f , é dado pela Equação 3.8.

$$\mathbf{x}_{\text{CM}}^f = \frac{\sum_{i=1}^{I_f} \mathbf{x}_i^f v_i^f}{\sum_{i=1}^{I_f} v_i^f} \quad (3.8)$$

Onde \mathbf{x}_{CM}^f é o vetor de coordenadas do centro de massa para fazenda f , I_f é o número de talhões disponíveis – talhões que ainda não foram sequenciados – para a fazenda f , \mathbf{x}_i^f é o vetor de coordenadas do talhão i da fazenda f e v_i^f é a massa de madeira do respectivo talhão. Os centros de massa devem ser re-calculados a cada novo sequenciamento de fazendas, pois a cada período alguns talhões são sequenciados e, assim, o centro de massa não estará condizente com a nova configuração de talhões.

A seguir, os talhões são sequenciados, seguindo a ordem pré-definida de fazendas. À vista disso, constrói-se a RCL, para a escolha dos talhões a constituírem a solução, baseando-se, primordialmente, na distância entre estes e nas outras funções objetivo definidas na Seção 3.1.2. A RCL também considera a próxima fazenda a ser visitada e, dessa maneira, prioriza os talhões escolhidos em direção a ela. Considerando o transporte de madeira para as plantas de carbonização em meses de chuva, os talhões com maior chance de serem escolhidos estarão em uma região determinada pela direção à próxima fazenda e pela direção à planta de carbonização. A Figura 3.4 representa um exemplo de escolha de talhões quando transporte será em meses de chuva. Sendo assim, nota-se que a região em destaque, entre as direções à planta de carbonização e à próxima fazenda, define os talhões a serem priorizados.

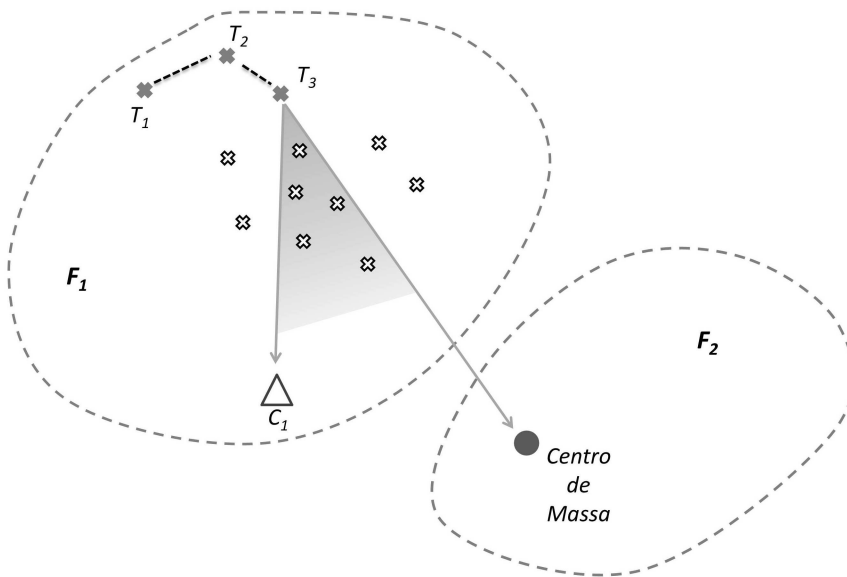


Figura 3.4: Sequenciamento de talhões: prioridade por direção

Neste processo de construção, também são definidas quais plantas de carbonização serão atendidas por quais talhões e os períodos em que a madeira será transportada após secagem.

A estrutura geral da construção é apresentada no Algoritmo 4, sendo os conjuntos R de regionais, F de fazendas, T de talhões, C de plantas de carbonização e M de máquinas e P de períodos do horizonte de planejamento.

Algoritmo 4 Construção Gulosa**requer** R, F, T, C, M, P

```

1: para  $p \leftarrow 1$  até  $P$  faça
2:    $\hat{M} = \text{ALOCARMÁQUINASEMREGIONAIS}(R, M, C, p) \triangleright$  Relacionar máquinas as
   fazendas de cada regional
3:   para todo  $m \in \hat{M}$  faça
4:      $\hat{F} \leftarrow \text{IDENTIFICARFAZENDASCRÍTICAS}(F, C, m, p)$ 
5:      $F_{\text{seq}} \leftarrow \text{SEQUENCIARFAZENDAS}(\hat{F})$ 
6:     para todo  $f \in F_{\text{seq}}$  faça
7:       repete
8:          $RCL \leftarrow \text{CRIARRCL}(f, T, C, p)$ 
9:          $\hat{T} = \text{SELECIONARCANDIDATO}(RCL)$ 
10:         $\text{ASSOCIARTALHÃOAPLANTA}(\hat{T}, C)$ 
11:         $S \leftarrow \text{ADICIONARTALHÃO NASOLUÇÃO}(S, p, m, f, \hat{T})$ 
12:      até Capacidade alocada para  $f$ 
13:    fim para
14:  fim para
15: fim para
16: devolve Nova solução  $S$ 

```

Primeiramente, alocam-se máquinas às fazendas de cada regional de forma a atender todas as demandas críticas. A demanda crítica é a demanda do quarto mês após o período de corte e deve ser prioritariamente atendida, pois não há período de secagem menor do que quatro meses no conjunto de fazendas da VFL. A função `IDENTIFICARFAZENDASCRÍTICAS(.)` analisa todas as fazendas relacionadas para a máquina m e divide a capacidade desta somente entre as consideradas críticas. Após identificá-las, as fazendas são sequenciadas e para cada uma, talhões são sequenciados até que se utilize toda capacidade alocada. Cada talhão é associado a uma planta de carbonização para receber sua madeira e adicionado à solução em construção.

Busca Local

Após a construção, o VNS (Algoritmo 2) é utilizado para busca local nas novas soluções a fim de que se encontre ótimos locais. Nessa etapa, é importante definir as estruturas de vizinhança que mais se adequam à modelagem proposta. Assim, é essencial utilizar conhecimento específico do problema para que essas estruturas sejam mais apropriadas, resultando em um método de busca local mais eficiente.

As estruturas de vizinhas foram criadas para alterar as soluções desde o nível mais baixo, alterar sequência de talhões, até os mais altos, fazendas e períodos. As estruturas são semelhantes para cada um desses níveis, portanto, a seguir, são detalhadas mais especificamente para talhões. As estruturas de vizinhança, ou operações, definidas para o VNS são:

- N_1 – Troca de talhões, fazendas ou períodos;
- N_2 – Deslocamento de talhão, fazenda ou período;
- N_3 – Inversão de talhões, fazendas ou períodos;
- N_4 – Inserção de talhão ou fazenda;

N_5 – Deleção de talhão ou fazenda;

N_6 – Troca de destino do talhão.

Primeiramente, têm-se as operações de troca N_1 que consistem em simples permutações de elementos. Na Figura 3.5, pode-se ver a operação de troca de talhões: um primeiro talhão t é escolhido aleatoriamente. A seguir, o segundo deve ser escolhido até uma certa distância δ do primeiro.

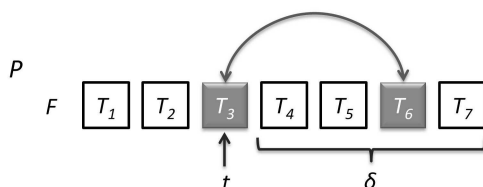


Figura 3.5: N_1 – Trocar talhões

Pode-se, também, permutar fazendas e períodos (Figuras 3.6 e 3.7, respectivamente), sendo similar à troca de talhões: escolhe-se o primeiro elemento (f ou p) aleatoriamente e o segundo a uma distância máxima δ .

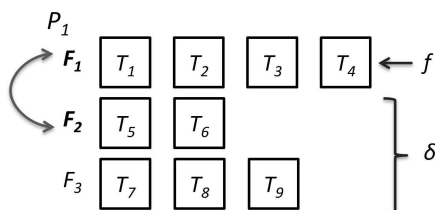


Figura 3.6: N_1 – Trocar Fazendas

A operação N_1 também permite a troca de talhões e fazendas entre períodos diferentes (Figura 3.8). Além disso, ainda é possível realizar a permutação entre talhões que foram sequenciados e aqueles que não foram. Essas trocas, entretanto, geralmente inviabilizam a solução, violando as restrições de capacidade e/ou demanda devido a diferenças de quantidade de madeira entre os elementos. Portanto, é aplicada, após essa operação, uma rotina

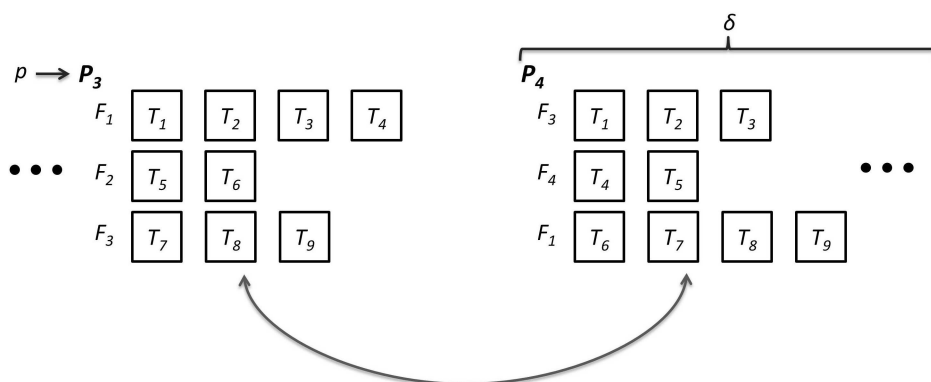


Figura 3.7: N_1 – Troca de períodos

de reparo que analisa todos períodos e fazendas a fim de identificar violações de restrições e viabilizar novamente a solução.

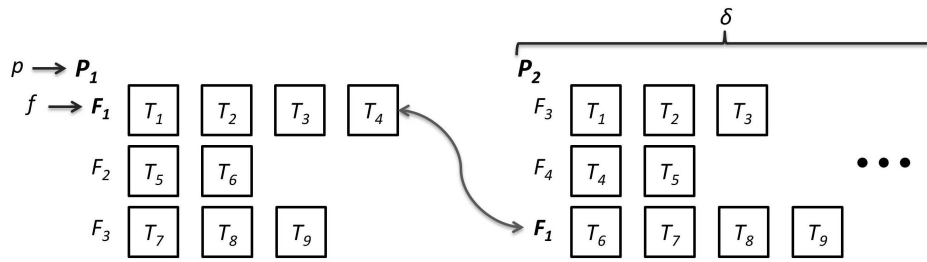


Figura 3.8: Exemplo de operações de troca para períodos diferentes

A segunda operação N_2 realiza o deslocamento de elementos (talhões, fazendas e períodos). Para talhões, um esquema é apresentado na Figura 3.9. No deslocamento, escolhe-se aleatoriamente o talhão t a ser deslocado e sua nova posição até uma distância máxima δ entre as posições final e inicial. Essa operação, ao deslocar talhões e fazendas entre períodos diferentes, também inviabiliza a solução, sendo necessário utilizar a rotina de reparo de solução.

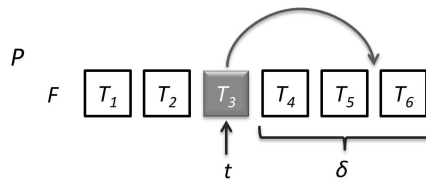


Figura 3.9: N_2 – Deslocar talhão

Pode-se, também, inverter elementos pela estrutura de vizinhança N_3 , como mostra a Figura 3.10 para inversão de talhões em um mesmo período. A operação N_3 também é limitada por uma quantidade máxima de elementos $\delta + 1$, sendo o primeiro da sequência escolhido aleatoriamente.

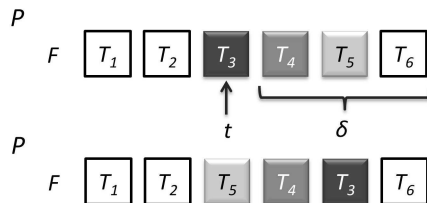
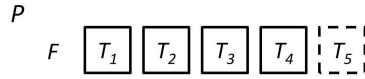
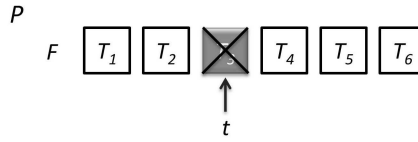


Figura 3.10: N_3 – Inverter talhões

A próxima operação N_4 (Figura 3.11) insere um talhão, que não foi sequenciado, em um período, fazenda e posição escolhidos aleatoriamente. Já a operação N_5 (Figura 3.12) realiza a deleção de talhões. Essas operações também podem realizar inserção e deleção de fazendas e também requer o uso de rotinas de reparo.

Finalmente, a operação N_6 (Figura 3.13) altera o destino da madeira dos talhões entre as plantas de carbonização que estão relativamente próximas da fazenda em análise, sendo necessário o reparo após a operação.

Figura 3.11: N_4 – Inserir talhãoFigura 3.12: N_5 – Deletar talhão

Nota-se que, após cada operação, as funções objetivo não são re-calculadas em toda solução. Porém, são calculadas somente na região alterada, aumentando, então, a eficiência, em termos de tempo de processamento, do algoritmo de busca local. Além disso, deve-se garantir que as soluções continuem viáveis posteriormente às operações.

3.2.2 Análise de Robustez

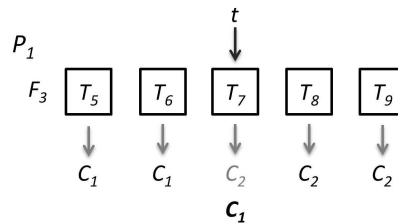
Como discutido na Seção 2.1.4, é importante analisar soluções de um problema de otimização também pela sua robustez frente a perturbações para que essas continuem práticas e sem alterações significativas em sua qualidade.

Dessa maneira, a busca por soluções robustas utilizada por essa metodologia será baseada no Algoritmo 3, onde o GRASP irá salvar as s_n melhores soluções encontradas, aplicar perturbações e, então, selecionar como melhor solução aquela que obtiver melhor desempenho dado pela Equação 2.8, considerando os parâmetros π constantes e a função de amostragem S_i para perturbar a solução por meio da aplicação das operações N_1, \dots, N_6 aleatoriamente.

É importante destacar que as operações N_4 e N_5 são consideradas as perturbações mais frequentes na prática de planejamento da colheita na VFL. Uma vez que talhões podem ser removidos do plano por causa de eventuais queimadas e, assim, devem ser substituídos. Além disso, alguns talhões podem ser adicionados ao plano por meio de requisições do setor de pesquisa da VFL.

3.2.3 Abordagem Escalar Via Método Epsilon-restrito e GRASP

Assim como descrito na Seção 2.1.3, o método ϵ -restrito tem relativa facilidade em ser implementado em estruturas de otimização mono-objetivo ao minimizar/maximizar um único

Figura 3.13: N_6 – Trocar destino do talhão

objetivo, transformando os demais em restrições. Assim, na metodologia desenvolvida, combina-se o GRASP com o método ϵ -restrito para obter uma aproximação do conjunto Pareto-ótimo. Além disso, utilizam-se as relações de ϵ -dominância para regular a resolução do conjunto Pareto e convergência do método de otimização e a metodologia de variação dos parâmetros ϵ descrita em [27].

Inicialmente, escolhe-se o objetivo F_1 (Equação 3.1) para ser minimizado e os demais são transformados em restrições. As novas restrições devem ser consideradas no algoritmo de construção gulosa e no VNS que deve implementar, também, rotinas de reparo para essas.

Além disso, o GRASP deve também salvar o conjunto de soluções pertencentes ao conjunto Pareto estimado e atualizá-lo sempre que houver uma nova solução ϵ -dominante.

3.3 Estrutura de Testes

A fim de validar a metodologia desenvolvida, o método de otimização foi testado para realizar planejamento de colheita anual, assim como é feito pela VFL, utilizando bases de dados dos planos de colheita desde 2011. Cada uma das bases de dados possui diferentes talhões disponíveis para corte, demandas de madeira das plantas de carbonização e configurações de máquinas. A seguir, as soluções encontradas pelos métodos deste trabalho são comparadas com os planejamentos de colheita realizados pela VFL e pelo algoritmo linear clássico desenvolvido em [30].

Além disso, são utilizadas metodologias citadas em [4] como auto-validação. Nessa análise, os principais pontos considerados são os piores e melhores casos, desempenho médio e desvio padrão dos testes em cada uma das bases de dados com objetivo avaliar a repetibilidade dos métodos.

Ademais, são realizados testes de robustez nas soluções encontradas e testes de sensibilidade da convergência do método em relação a variação de seus parâmetros, como o parâmetro α da construção gulosa, o número de iterações do VNS e GRASP, os parâmetros δ das operações, os parâmetros ϵ das relações ϵ -dominância e a quantidade de perturbações n_e no teste de robustez.

Capítulo 4

Resultados

Capítulo 5

Conclusões

Referências Bibliográficas

- [1] José Rodrigo Banhara, Luiz Carlos Estraviz Rodriguez, Fernando Seixas, José Mauro Magalhães Ávila Paz Moreira, Lana Mirian Santos da Silva, Silvana Ribeiro Nobre, and Andrew Cogswell. Agendamento otimizado da colheita de madeira de eucaliptos sob restrições operacionais, espaciais e climáticas. *Scientia Florestalis*, 38(85):85—95, 2010.
- [2] Emin Zeki Baskent and Sedat Keles. Spatial forest planning: A review. *Ecological Modelling*, 188(2-4):145–173, 2005.
- [3] Mokhtar S Bazaraa, Hanif D Sherali, and Chitharanjan Marakada Shetty. *Nonlinear programming: theory and algorithms*. John Wiley & Sons, 2013.
- [4] Pete Bettinger, John Sessions, and Kevin Boston. A review of the status and use of validation procedures for heuristics used in forest planning. *Mathematical and Computational Forestry & Natural-Resource Sciences (MCFNS)*, 1(1):Pages—26, 2009.
- [5] Christian Blum, Jakob Puchinger, Günther R Raidl, and Andrea Roli. Hybrid metaheuristics in combinatorial optimization: A survey. *Applied Soft Computing*, 11(6):4135–4151, 2011.
- [6] Carlos Bohle, Sergio Maturana, and Jorge Vera. A robust optimization approach to wine grape harvesting scheduling. *European Journal of Operational Research*, 200(1):245–252, 2010.
- [7] Mourad Boudia, Mohamed Aly Ould Louly, and Christian Prins. A reactive GRASP and path relinking for a combined production–distribution problem. *Computers & Operations Research*, 34(11):3402–3419, 2007.
- [8] Ilhem Boussaïd, Julien Lepagnot, and Patrick Siarry. A survey on optimization metaheuristics. *Information Sciences*, 237:82–117, 2013.
- [9] Nadjib Brahimi, Stéphane Dauzere-Peres, Najib M Najid, and Atle Nordli. Single item lot sizing problems. *European Journal of Operational Research*, 168(1):1–16, 2006.
- [10] Jacques Carlier. The one-machine sequencing problem. *European Journal of Operational Research*, 11(1):42–47, 1982.
- [11] Yann Collette and Patrick Siarry. *Multiobjective optimization: principles and case studies*. Springer Science & Business Media, 2003.
- [12] Mateus Rocha De Paula, Martín Gómez Ravetti, Geraldo Robson Mateus, and Panos M Pardalos. Solving parallel machines scheduling problems with sequence-dependent setup times using variable neighbourhood search. *IMA Journal of Management Mathematics*, 18(2):101–115, 2007.
- [13] Johann Dréo. *Metaheuristics for hard optimization: methods and case studies*. Springer Science & Business Media, 2006.

- [14] Andreas Drexl and Alf Kimms. Lot sizing and scheduling - survey and extensions. *European Journal of operational research*, 99(2):221–235, 1997.
- [15] Matthias Ehrgott. *Multicriteria optimization*. Springer Science & Business Media, 2006.
- [16] Thomas A Feo and Mauricio G C Resende. A probabilistic heuristic for a computationally difficult set covering problem. *Operations research letters*, 8(2):67–71, 1989.
- [17] Thomas A Feo and Mauricio G C Resende. Greedy randomized adaptive search procedures. *Journal of global optimization*, 6(2):109–133, 1995.
- [18] Paola Festa and Mauricio G C Resende. An annotated bibliography of GRASP–Part II: Applications. *International Transactions in Operational Research*, 16(2):131–172, 2009.
- [19] Bernhard Fleischmann and Herbert Meyr. The general lotsizing and scheduling problem. *Operations-Research-Spektrum*, 19(1):11–21, 1997.
- [20] M Gen and R Cheng. *Genetic Algorithms and Engineering Optimization*. A Wiley-Interscience publication. Wiley, 2000.
- [21] Michel Gendreau and Jean-Yves Potvin. *Handbook of metaheuristics*, volume 2. Springer, 2010.
- [22] Lucas Rezende Gomide, Julio Eduardo Arce, and Arinei Carlos Lindbeck da Silva. Uso do algoritmo genético no planejamento florestal considerando seus operadores de seleção. *Cerne*, 15(4):460, 2009.
- [23] Johnny C Ho and Yih-Long Chang. A new heuristic for the n-job, M-machine flow-shop problem. *European Journal of Operational Research*, 52(2):194–202, 1991.
- [24] Yaochu Jin and Jürgen Branke. Evolutionary optimization in uncertain environments-a survey. *Evolutionary Computation, IEEE Transactions on*, 9(3):303–317, 2005.
- [25] Jenny Karlsson, Mikael Rönnqvist, and Johan Bergström. An optimization model for annual harvest planning. *Canadian Journal of Forest Research*, 34(8):1747–1754, 2004.
- [26] Marco Laumanns, Lothar Thiele, Kalyanmoy Deb, and Eckart Zitzler. Combining convergence and diversity in evolutionary multiobjective optimization. *Evolutionary computation*, 10(3):263–282, 2002.
- [27] Marco Laumanns, Lothar Thiele, and Eckart Zitzler. An efficient, adaptive parameter variation scheme for metaheuristics based on the epsilon-constraint method. *European Journal of Operational Research*, 169(3):932–942, 2006.
- [28] Muhammad Nawaz, E Emory Enscore, and Inyong Ham. A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *Omega*, 11(1):91–95, 1983.
- [29] Cristian D Palma and John D Nelson. A robust optimization approach protected harvest scheduling decisions against uncertainty. *Canadian journal of forest research*, 39(2):342–355, 2009.
- [30] Fábio Pavinato and Guilherme Souza. *Desenvolvimento de um Modelo Matemático para Otimização do Planejamento do Setor de Colheita em uma Empresa do Ramo Florestal*. Trabalho de graduação, Universidade Federal de Minas Gerais, 2013.
- [31] Michael L Pinedo. *Scheduling: theory, algorithms, and systems*. Springer Science & Business Media, 2012.

- [32] Daniel Quadt and Heinrich Kuhn. Production planning in semiconductor assembly. In *Proceedings of the Fourth Aegean International Conference on Analysis of Manufacturing Systems*, pages 181–189, 2003.
- [33] S S Rao. *Engineering Optimization: Theory and Practice*. Wiley, 2009.
- [34] Tapabrata Ray. Constrained robust optimal design using a multiobjective evolutionary algorithm. In *Evolutionary Computation, 2002. CEC'02. Proceedings of the 2002 Congress on*, volume 1, pages 419–424. IEEE, 2002.
- [35] Amir Salehipour, Kenneth Sörensen, Peter Goos, and Olli Bräysy. Efficient GRASP+ VND and GRASP+ VNS metaheuristics for the traveling repairman problem. *4OR*, 9(2):189–209, 2011.
- [36] Patrick Schittekat, Joris Kinable, Kenneth Sörensen, Marc Sevaux, Frits Spieksma, and Johan Springael. A metaheuristic for the school bus routing problem with bus stop selection. *European Journal of Operational Research*, 229(2):518–528, 2013.
- [37] Ozlem Senvar, Ebru Turanoglu, and Cengiz Kahraman. *Meta-Heuristics Optimization Algorithms in Engineering, Business, Economics, and Finance*. IGI Global, 2013.
- [38] Kenneth Sörensen. A framework for robust and flexible optimisation using metaheuristics. *Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, 1(4):341–345, 2003.
- [39] Jeffrey D Ullman. NP-complete scheduling problems. *Journal of Computer and System sciences*, 10(3):384–393, 1975.