



FUNDAÇÃO EDSON QUEIROZ
UNIVERSIDADE DE FORTALEZA – UNIFOR
VICE-REITORIA DE ENSINO DE GRADUAÇÃO E PÓS-GRADUAÇÃO – VRE
DIRETORIA DE PÓS-GRADUAÇÃO – DPG
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA APLICADA – PPGIA

**MODELO DE OTIMIZAÇÃO PARA O ESCALONAMENTO DE
MOTORISTAS NO TRANSPORTE RODOVIÁRIO SOB
REGULAMENTAÇÃO EUROPEIA**

FRANCISCO VANDO CARNEIRO MOREIRA

FORTALEZA – CE
NOVEMBRO, 2025

Francisco Vando Carneiro Moreira

**Modelo de Otimização para o Escalonamento de
Motoristas no Transporte Rodoviário sob Regulamentação
Europeia**

Exame de Qualificação de Mestrado apresentado ao Programa de Pós-Graduação em Informática Aplicada – PPGIA da Universidade de Fortaleza – UNIFOR, como parte dos requisitos para obtenção do título de Mestre em Informática Aplicada.

Universidade de Fortaleza – UNIFOR

Orientador: Prof. Dr. Plácido Rogério Pinheiro

Fortaleza – CE

Novembro, 2025

Ao meu pai.

Agradecimentos

Agradeço, primeiramente, ao Prof. Dr. Plácido Rogério Pinheiro, cuja orientação firme, dedicação acadêmica e confiança constante foram determinantes para a concretização desta pesquisa. Sua capacidade de incentivar, esclarecer e apontar caminhos, mesmo nos momentos mais desafiadores, foi essencial para que eu pudesse avançar com segurança e convicção. Sou profundamente grato por acreditar no meu potencial e por me acompanhar de forma tão generosa nesta jornada.

Expresso também minha sincera gratidão ao Prof. Dr. Valdísio Viana e ao Prof. Dr. Antonio Clécio Thomas, pelo apoio, pelas contribuições valiosas e pela disponibilidade em compartilhar conhecimento e experiência. Suas observações e estímulos fortaleceram-me e ampliaram meu olhar acadêmico em outros momentos da minha jornada de vida.

À minha esposa Carolina Araujo, deixo meu agradecimento mais especial. Pelo apoio incondicional, pela compreensão diante das ausências, pelo estímulo constante e pela serenidade com que dividiu os desafios desta caminhada. Nada disso seria possível sem a sua presença carinhosa, firme e essencial.

Aos meus pais, agradeço pelo amor, pelos valores e por tudo o que construíram em mim ao longo da vida. Em especial, dedico este momento à memória do meu pai, cuja força, exemplo e ensinamentos permanecem vivos em cada conquista. Sua ausência física jamais apagou sua presença na minha trajetória. Este trabalho também é dele.

A todos que, direta ou indiretamente, contribuíram para que este estudo se tornasse realidade, deixo meu mais profundo reconhecimento e gratidão.

“Tudo o que sou carrega um pouco de ti.”

- VM

Resumo

O escalonamento de motoristas no transporte rodoviário de cargas sob a regulamentação europeia constitui um problema complexo de otimização combinatória, especialmente quando se considera a integração simultânea de restrições legais, operacionais e de demanda temporal.

Este trabalho apresenta o desenvolvimento de um simulador para o escalonamento de motoristas no transporte rodoviário europeu, fundamentado em um modelo de Programação Linear Inteira (PLI) que incorpora as exigências legais estabelecidas pelo Regulamento (CE) n.º 561/2006, pela Diretiva 2002/15/CE e pelo Regulamento (UE) n.º 165/2014.

O modelo matemático foi formulado de modo a representar, todas as restrições diárias, semanais e quinzenais de condução, pausas e períodos de descanso obrigatórios, incluindo dependências temporais complexas e janelas móveis de conformidade com a regulamentação.

A implementação computacional foi implementada e integrada a uma interface interativa, permitindo a configuração de cenários, seleção dinâmica de cenários e visualização dos resultados. Além do modo exato de resolução, o simulador incorpora abordagens heurísticas e uma estratégia matheurística baseada em *Large Neighborhood Search* (LNS), possibilitando a análise do comportamento do modelo em diferentes regimes de complexidade e escalabilidade.

Adicionalmente, o trabalho propõe a integração experimental de modelos de aprendizado de máquina supervisionado, utilizados para guiar decisões locais de alocação (f_1) e a seleção de vizinhanças no LNS (f_2), preservando mecanismos de *fallback* heurístico e estabilidade operacional. O simulador disponibiliza um conjunto de indicadores de desempenho (KPIs), que auxiliam na interpretação dos resultados.

Os experimentos computacionais conduzidos em horizontes de curto e médio prazo indicam que a abordagem constitui uma alternativa em identificar soluções e analisar cenários sob diferentes combinações de restrições, fornecendo suporte à tomada de decisão estratégica, tática e operacional.

E também aponta um ambiente unificado de modelagem, simulação e análise para o escalonamento de motoristas sob a legislação europeia, e pode contribuir para o estado da arte ao combinar otimização matemática, heurísticas, matheurísticas e aprendizado de máquina em um único ambiente experimental.

Palavras-chave: Programação Linear Inteira; Escalonamento de Motoristas; Regulamento (CE) n.º 561/2006; Otimização; OR-Tools; Matheurísticas; Simulação; Aprendizado de Máquina.

Resumo

Driver scheduling in European road freight transport constitutes a complex combinatorial optimization problem, mainly due to the strict regulatory constraints imposed by European legislation on driving times, mandatory breaks, and minimum rest periods, as established by Regulation (EC) No. 561/2006. In this context, this dissertation proposes an Integer Linear Programming (ILP) model for driver scheduling over discretized planning horizons, explicitly incorporating operational, temporal, and regulatory constraints, as well as demand satisfaction requirements.

The proposed formulation supports different solution strategies, including exact optimization methods, constructive heuristics, and a Large Neighborhood Search (LNS) approach, enabling a systematic evaluation of the trade-off between solution quality and computational efficiency. Computational experiments were conducted using synthetically generated data and realistic operational scenarios, considering key performance indicators such as demand coverage, workforce utilization, and solution stability.

The results demonstrate that the proposed model is capable of achieving high levels of demand coverage while reducing the total number of active drivers and maintaining strict compliance with regulatory constraints. Furthermore, heuristic and LNS-based approaches provide competitive solutions within significantly shorter computational times, preserving acceptable solution quality. These findings highlight the practical applicability of the proposed approach as a decision-support tool for driver scheduling in regulated road transport environments.

Lista de Abreviaturas e Siglas

ABNT Associação Brasileira de Normas Técnicas

CE Comunidade Europeia

CP-SAT Constraint Programming – Satisfiability Solver

EC European Community

EU European Union

FMS Fleet Management System

ILP Integer Linear Programming

LP Linear Programming

OR Operations Research

OR-Tools Optimization Tools (Google)

TMS Transport Management System

PLI Programação Linear Inteira

VRP Vehicle Routing Problem

Lista de Símbolos

$x_{d,t}$ Motorista d trabalha no período t

$r_{d,t}$ Motorista d está em descanso no período t

$h_{d,t}$ Condução acumulada no período t

$s_{d,t}$ Início de jornada no período t

$z_{d,w}$ Uso de extensão diária na semana w

Δ Duração de cada período

L^{dia} Limite diário de condução

L^{ext} Extensão diária permitida

L^{sem} Limite semanal de condução

L^{14d} Limite de condução em 14 dias

R_{dia} Descanso diário normal

$R_{\text{dia-red}}$ Descanso diário reduzido

R_{sem} Descanso semanal

P_{pausa} Limite de condução contínua antes da pausa

Sumário

1	INTRODUÇÃO	17
1.1	Contextualização	17
1.2	Problema da Pesquisa	19
1.3	Questão da Pesquisa	19
1.4	Objetivo Geral	19
1.5	Objetivos Específicos	19
1.6	Justificativa	20
1.7	Estrutura da Dissertação	20
2	REVISÃO DA LITERATURA E POSICIONAMENTO CIENTÍFICO	22
2.1	Introdução	22
2.2	Transporte Rodoviário Europeu e as Restrições Regulatórias	22
2.3	Problemas de Escalonamento e Abordagens Estruturadas	23
2.4	Modelagem Matemática com Programação Linear Inteira	24
2.5	Metaheurísticas e Técnicas Avançadas de Otimização	24
2.6	O Uso de Solvers Modernos: OR-Tools e CP-SAT	24
2.7	Trabalhos Correlatos sobre Escalonamento e Logística	25
2.8	Estado da Arte e Análise Comparativa	25
2.8.1	Comparação Estruturada de Trabalhos Relevantes	25
2.8.2	Análise Crítica Comparativa	26
2.8.3	Comparação Técnica das Abordagens	27
2.9	Lacunas Identificadas na Literatura	27
2.10	Posicionamento e Contribuições da Dissertação	28
2.10.1	Contribuições Científicas	29
2.10.2	Posicionamento Científico	30
2.11	Síntese e Organização da Dissertação	30
3	BASES LEGAIS E NORMATIVAS DO ESCALONAMENTO DE MOTORISTAS	31
3.1	Introdução	31
3.2	Regulamento (CE) n.º 561/2006	31
3.2.1	Limites de Condução	32
3.2.2	Pausas Obrigatórias	32
3.2.3	Descanso Diário	32
3.2.4	Descanso Semanal	32
3.2.5	Implicações para o modelo matemático	32

3.3	Diretiva 2002/15/CE	33
3.4	Regulamento (UE) n.º 165/2014	34
3.5	Síntese das Bases Legais	34
3.6	Relação com o Modelo Matemático	35
3.7	Considerações Finais	36
4	MODELAGEM E METODOLOGIA EXPERIMENTAL	37
4.1	Introdução	37
4.2	Formulação Matemática do Modelo	38
4.3	Discretização Temporal	38
4.4	Conjuntos e Índices	38
4.5	Parâmetros	38
4.6	Variáveis de Decisão	39
4.7	Função Objetivo Lexicográfica	39
4.7.1	Fase 1 — Cobertura e Minimização de Motoristas	40
4.7.2	Fase 2 — Balanceamento e Refinamento	40
4.8	Restrições do Modelo	40
4.8.1	Atendimento da Demanda	41
4.8.2	Vinculação entre presença e ativação	41
4.8.3	Capacidade por período	41
4.8.4	Limites legais de condução	41
4.8.5	Pausas e descansos	41
4.9	Discussão sobre Linearização	42
4.10	Observações Computacionais	42
4.11	Metodologia de Implementação e Experimentação	42
4.12	Etapas 1: Estudo da Legislação Europeia	42
4.13	Etapas 2: Formulação Matemática	43
4.14	Etapas 3: Implementação Computacional	43
4.15	Etapas 4: Definição dos Cenários Experimentais	44
4.16	Etapas 5: Análise dos Resultados	45
4.16.1	Abordagem heurística e matheurística proposta	46
4.16.1.1	Heurística construtiva para geração de escala inicial	47
4.16.1.2	Matheurística LNS baseada em MILP	49
4.16.1.3	Integração com aprendizagem de máquina	51
4.17	Metodologia de Avaliação	51
4.18	Métodos de Resolução e Análise de Resultados	52
4.18.1	Resultados Computacionais — Método Exato	52
4.18.1.1	Indicadores Globais de Desempenho	53
4.18.2	Atendimento da Demanda e Utilização da Capacidade	53
4.18.3	Distribuição do Esforço Operacional	54

4.18.4	Estrutura da Solução e Padrão de Alocação	54
4.18.5	Considerações Parciais sobre o Método Exato	55
4.18.6	Resultados Computacionais — Método Heurístico	55
4.18.6.1	Indicadores Globais de Desempenho	56
4.18.6.2	Atendimento da Demanda e Utilização da Capacidade	56
4.18.6.3	Distribuição do Esforço Operacional	57
4.18.6.4	Estrutura da Solução Heurística	58
4.18.6.5	Considerações Parciais sobre o Método Heurístico	59
4.18.7	Resultados Computacionais — Large Neighborhood Search (LNS)	59
4.18.7.1	Indicadores Globais de Desempenho	60
4.18.7.2	Atendimento da Demanda e Utilização da Capacidade	60
4.18.7.3	Distribuição do Esforço Operacional	61
4.18.7.4	Estrutura da Solução Após Refinamento	62
4.18.7.5	Considerações Parciais sobre o Método LNS	63
4.19	Considerações Finais	64
5	IMPLEMENTAÇÃO COMPUTACIONAL	65
5.1	Introdução	65
5.2	Arquitetura Geral da Solução	66
5.2.1	Interface de Parametrização	66
5.2.2	Camada de Modelagem Matemática	67
5.2.3	Camada de Resolução	68
5.2.4	Pós-processamento e Análise de Resultados	68
5.2.5	Considerações Arquiteturais	69
5.3	Interface de Parametrização	69
5.3.1	Configuração do Horizonte Temporal	69
5.3.2	Inserção da Demanda Operacional	70
5.3.3	Seleção do Modo de Resolução	70
5.3.4	Seleção da Função Objetivo	71
5.3.5	Ativação das Restrições Legais	71
5.3.6	Parâmetros Computacionais e Controle do Solver	71
5.3.7	Reprodutibilidade e Consistência Experimental	72
5.4	Configuração das Restrições Legais	72
5.4.1	Motivação para a Ativação Seletiva	72
5.4.2	Conjunto de Restrições Implementadas	73
5.4.3	Interpretação Operacional da Configuração	73
5.4.4	Consistência do Modelo sob Diferentes Configurações	73
5.4.5	Relação com os Experimentos Realizados	74
5.5	Construção da Matriz de Restrições	74
5.5.1	Estrutura e Organização da Matriz	74

5.5.2	Discretização Temporal e Encadeamento	75
5.5.3	Esparsidade da Matriz	75
5.5.4	Operações Elementares e Análise Estrutural	75
5.5.5	Versões da Matriz e Transformações	76
5.5.6	Relação com o Desempenho Computacional	76
5.6	Execução do Solver	76
5.6.1	Inicialização e Parâmetros do Solver	76
5.6.2	Critérios de Parada	77
5.6.3	Estados de Solução	77
5.6.4	Execução nos Diferentes Modos de Resolução	78
5.6.5	Extração da Solução	78
5.6.6	Relação com os Resultados Experimentais	78
5.7	Fluxo Computacional do Simulador	79
5.7.1	Etapas do Fluxo de Execução	79
5.7.2	Determinismo e Reprodutibilidade	80
5.7.3	Fluxo Específico por Modo de Resolução	80
5.7.4	Integração com os Resultados Experimentais	80
5.8	Modos de Resolução e Estratégias de Otimização	80
5.8.1	Modo de Resolução Exato	81
5.8.2	Modo Heurístico	81
5.8.3	Modo Matheurístico (LNS)	81
5.8.4	Estratégias de Otimização	82
5.8.5	Combinação entre Modos e Estratégias	82
5.8.6	Relação com os Experimentos Apresentados	82
5.9	Geração de Indicadores e Gráficos	82
5.9.1	Indicadores de Cobertura da Demanda	83
5.9.2	Indicadores de Utilização e Sobrecarga	83
5.9.3	Indicadores Temporais e Estabilidade	83
5.9.4	Indicadores Computacionais	84
5.9.5	Visualizações Gráficas	84
5.9.6	Relação com os Resultados Experimentais	84
5.10	Módulo de Operações Elementares	84
5.10.1	Objetivos do Módulo	85
5.10.2	Preservação da Equivalência Matemática	85
5.10.3	Análise da Densidade da Matriz	85
5.10.4	Relação com os Resultados Apresentados	86
5.11	Extensões Heurísticas e Metaheurísticas	86
5.11.1	Heurística Construtiva Gulosa	86
5.11.2	Large Neighborhood Search (LNS)	87

5.11.3	Controle de Relaxação e Iterações	87
5.11.4	Garantia de Validade das Soluções	87
5.11.5	Relação com os Experimentos Computacionais	87
5.12	Integração de Aprendizado de Máquina	88
5.12.1	Motivação para o Uso de Aprendizado Supervisionado	88
5.12.2	Modelos Supervisionados Implementados	88
5.12.3	Arquitetura Resiliente e Comportamento Padrão	89
5.12.4	Geração de Dados e Treinamento dos Modelos	89
5.12.5	Papel do Aprendizado de Máquina nos Experimentos	89
5.12.6	Garantia de Rigor Metodológico	90
5.13	Geração de Datasets e Treinamento dos Modelos	90
5.13.1	Origem dos Dados Supervisionados	90
5.13.2	Dataset para o Modelo Local de Alocação (f_1)	91
5.13.3	Dataset para Avaliação de Vizinhanças no LNS (f_2)	91
5.13.4	Pipeline de Geração Automática	92
5.13.5	Treinamento dos Modelos Supervisionados	92
5.13.6	Separação entre Treinamento e Avaliação	92
5.13.7	Considerações Metodológicas	92
5.14	Visualização Avançada e Análise Iterativa	93
5.14.1	Visualizações Temporais da Solução	93
5.14.2	Indicadores de Estabilidade e Eficiência	93
5.14.3	Análise Iterativa no Modo LNS	94
5.14.4	Comparação entre Modos de Resolução	94
5.14.5	Visualizações para Diagnóstico e Auditoria	94
5.14.6	Integração com o Processo Experimental	95
5.15	Posicionamento do Simulador como Plataforma Experimental	95
5.15.1	Unificação de Estratégias de Resolução	95
5.15.2	Separação entre Modelo, Resolução e Análise	95
5.15.3	Reprodutibilidade e Controle Experimental	96
5.15.4	Integração Opcional de Aprendizado de Máquina	96
5.15.5	Aplicabilidade Científica e Industrial	96
5.15.6	Síntese do Capítulo	97
6	RESULTADOS	98
6.1	Introdução	98
6.2	Roteiro Experimental	98
6.3	Configuração Geral dos Cenários	99
6.4	Execução do Solver	99
6.5	Comportamento da Demanda	100
6.6	Comparação Demanda <i>versus</i> Alocação	101

6.7	Indicadores Operacionais	101
6.8	Estrutura da Matriz de Restrições	102
6.9	Comparação entre Modos de Resolução	103
6.10	Operações Elementares e Análise Estrutural	104
6.11	Cenários de Rápida Convergência	105
6.12	Análise Gráfica Avançada	106
6.13	Síntese Geral dos Resultados	107
7	DISCUSSÃO E VALIDAÇÃO EXPERIMENTAL	108
7.1	Introdução	108
7.2	Comparação das Técnicas Desenvolvidas com a Literatura e o Estado da Arte	108
7.2.1	Extensões Recentes do Simulador e Alinhamento com Tendências Atuais	109
7.2.2	Heurística Construtiva como Baseline Experimental	109
7.2.3	Large Neighborhood Search com Reotimização via Solver Inteiro	109
7.2.4	Aprendizado de Máquina como Camada Opcional de Guidance	110
7.2.5	Visualização Analítica e Instrumentação Experimental	110
7.3	Pipeline Híbrido de Otimização	111
7.3.1	Heurística Construtiva e Geração da Solução Inicial	111
7.3.2	Atualização de Estado e Consistência Regulamentar	111
7.3.3	Análise Estrutural da Matriz de Restrições	111
7.3.4	Large Neighborhood Search com Reotimização MILP	112
7.3.5	Camada Opcional de Aprendizagem de Máquina	112
7.3.6	Critério de Parada e Solução Final	112
7.4	Validação das Hipóteses de Pesquisa	114
7.4.1	Validação da Contribuição 1: Modelagem Matemática	114
7.4.2	Validação da Contribuição 2: Heurística e Matheurísticas	114
7.4.3	Validação da Contribuição 3: Análise Estrutural	114
7.4.4	Validação da Contribuição 4: Plataforma Experimental	114
7.4.5	Validação da Contribuição 5: Viabilidade Prática	114
7.5	Síntese da Discussão	115
8	CONCLUSÕES E TRABALHOS FUTUROS	116
8.1	Considerações Iniciais	116
8.2	Síntese do Trabalho Desenvolvido	116
8.3	Principais Resultados e Evidências Empíricas	117
8.4	Validação das Contribuições Científicas	118
8.5	Limitações Identificadas	118
8.6	Direções para Trabalhos Futuros	119
8.7	Considerações Finais	120

Referências	121
-----------------------	-----

Glossário	123
---------------------	-----

APÊNDICES 125

APÊNDICE A – FORMULAÇÕES ALTERNATIVAS DO MODELO 126

A.1	Modelo com Janelas de Tempo Suaves (Soft Time Windows)	126
A.2	Modelo com Agrupamento de Motoristas por Categoria	126
A.3	Modelo com Múltiplos Períodos (Planeamento Semanal)	127
A.4	Modelo com Preferências do Motorista	127

APÊNDICE B – CENÁRIOS DE SIMULAÇÃO E PARÂMETROS UTILIZADOS 128

B.1	Descrição Geral dos Cenários	128
B.2	Configurações do Solver (CP-SAT / OR-Tools)	128
B.3	Restrições Consideradas	129

APÊNDICE C – ESTRUTURA DE ENTRADAS E SAÍDAS DO SOLVER 130

C.1	Entradas do Modelo	130
C.2	Saídas do Modelo	130

APÊNDICE D – MODELO MATEMÁTICO COMPLETO 132

D.1	Variáveis de Decisão	132
D.2	Função Objetivo	132
D.3	Restrições	132
D.3.1	Atribuição Única	132
D.3.2	Limite de Condução Diária	132
D.3.3	Janelas de Tempo	132
D.3.4	Compatibilidade Motorista–Tarefa	133
D.3.5	Acúmulo Semanal	133
D.4	Modelo Final	133

ANEXOS 134

ANEXO A – RESUMO ESTRUTURADO DO REGULAMENTO (CE) N.º 561/2006 135

ANEXO B – GLOSSÁRIO TÉCNICO 137

ANEXO C – FLUXO GERAL DA APLICAÇÃO	138
--	-----

1 Introdução

Nesse contexto, o escalonamento de motoristas sob restrições regulatórias emerge como um problema de otimização combinatória de elevada complexidade, com impacto direto na eficiência operacional, nos custos e na conformidade legal das empresas de transporte

Os elementos pré-textuais apresentados a seguir têm como finalidade orientar o leitor acerca da estrutura gráfica, simbólica e organizacional propostos, dedicada ao desenvolvimento de um modelo de Programação Linear Inteira para o escalonamento de motoristas sob o Regulamento (CE) n.º 561/2006.

Ao longo do trabalho, diferentes tipos de representações são utilizados para apoiar a formulação matemática, a implementação computacional e a análise dos resultados. A Lista de Ilustrações reúne as figuras que apresentam, entre outros elementos, a interface do sistema construído, a estrutura da matriz de restrições e os gráficos derivados das simulações realizadas. As listas de Quadros e Tabelas são mantidas para conformidade com as normas acadêmicas, mesmo que o texto não contenha quadros ou tabelas formais.

A Lista de Abreviaturas e Siglas consolida os termos técnicos utilizados no campo da Pesquisa Operacional, da otimização e da legislação europeia aplicada ao transporte rodoviário, facilitando o acompanhamento da leitura. A Lista de Símbolos organiza as notações matemáticas empregadas na formulação do modelo, servindo como referência rápida para os capítulos que apresentam o formalismo teórico.

Por fim, o Sumário estrutura a organização geral do documento, permitindo uma navegação clara entre as seções que tratam desde o contexto e fundamentação teórica até os experimentos, resultados e conclusões finais. Esses elementos auxiliam na compreensão integral da pesquisa, conferindo ao trabalho maior clareza, rigor e acessibilidade.

1.1 Contextualização

O transporte rodoviário de cargas constitui um componente central da economia europeia, assegurando fluxos contínuos de bens e garantindo a integração logística entre os Estados-Membros. Nesse contexto, os motoristas profissionais constituem o recurso humano central da operação, tornando a gestão das suas jornadas um fator crítico para a eficiência, segurança e conformidade legal do setor.

Nos últimos anos, abordagens híbridas que combinam otimização exata, heurísticas e, de forma complementar, técnicas de aprendizado de máquina têm sido exploradas como forma de acelerar a convergência e melhorar a qualidade das soluções iniciais, especialmente

em problemas de grande escala e natureza temporal.

A União Europeia possui uma das regulamentações mais rigorosas do mundo no que diz respeito aos tempos de condução, pausas e períodos de descanso dos motoristas. O *Regulamento (CE) n.º 561/2006*, complementado pela *Diretiva 2002/15/CE* e pelo *Regulamento (UE) n.º 165/2014*, estabelece limites estritos que visam proteger a saúde dos motoristas, prevenir acidentes e promover condições de trabalho seguras e humanizadas (Union, 2006, 2002, 2014). Essas normas afetam diretamente a forma como as escalas são construídas, exigindo atenção constante ao cumprimento de requisitos diários, semanais e quinzenais.

Entretanto, na prática, a elaboração manual de escalas é propensa a erros, especialmente em operações de grande escala, caracterizadas por alta variabilidade de demanda, múltiplas janelas temporais, restrições acumulativas e dependências temporais entre os períodos de trabalho. Pequenas decisões tomadas em um período podem invalidar toda a escala futura, resultando em infrações legais, custos adicionais e riscos operacionais.

Nesse cenário, abordagens matemáticas baseadas em Programação Linear Inteira (PLI) emergem como ferramentas adequadas para formalizar e resolver problemas de escalonamento, oferecendo rigor, precisão e capacidade de lidar com múltiplas restrições simultaneamente (Hillier; Lieberman, 2015), (Taha, 2017), (Nemhauser; Wolsey, 1988). De forma complementar, solvers modernos como o OR-Tools disponibilizam recursos computacionais avançados que permitem resolver modelos inteiros de grande escala em tempos reduzidos (Google, 2025).

Além do modelo de otimização em si, esta pesquisa também materializa um ambiente de simulação e análise, no qual o usuário pode configurar parâmetros operacionais, habilitar/desabilitar restrições legais e observar, de forma visual, como tais escolhas impactam a viabilidade, o número de motoristas necessários e a qualidade da cobertura da demanda. Esse ambiente inclui métricas e gráficos de diagnóstico (por exemplo, cobertura por slot, subutilização, sobrecarga, risco e estabilidade temporal), bem como análises estruturais do modelo por meio de visualizações de matrizes e medidas como densidade, oferecendo suporte à interpretação do comportamento do solver e à validação do modelo.

Esta dissertação insere-se nesse contexto, propondo um modelo abrangente de PLI para o escalonamento de motoristas em conformidade com o Regulamento (CE) n.º 561/2006, apoiado por uma implementação computacional interativa e orientada à simulação de cenários operacionais. Parte desse desenvolvimento foi previamente publicada em (Moreira; Pinheiro; Araujo, 2025), que fundamenta e inspira a formulação aprofundada apresentada nesta pesquisa.

1.2 Problema da Pesquisa

O problema desta dissertação consiste em determinar como alocar motoristas ao longo de um horizonte temporal discretizado, de forma a: (i) atender à demanda operacional por período; (ii) respeitar o Regulamento (CE) n.º 561/2006; (iii) minimizar o número total de motoristas utilizados; (iv) garantir escalas contínuas, regulares e operacionalmente viáveis; (v) permitir análises de sensibilidade para diferentes cenários.

Adicionalmente, quando o modelo exato se torna computacionalmente custoso em instâncias maiores, torna-se relevante investigar estratégias de busca e melhoria incremental de soluções, incorporadas no próprio simulador, tais como heurísticas construtivas e busca em grandes vizinhanças, preservando a conformidade legal e permitindo análises comparativas entre modos de resolução.

Trata-se de um contexto caracterizado pelas restrições acumulativas sobre condução diária, semanal e quinzenal; pela necessidade de pausas obrigatórias; pela variabilidade da demanda; pelas dependências temporais entre períodos sucessivos; e pelo grande número de combinações possíveis entre motoristas e períodos.

1.3 Questão da Pesquisa

Como formular e implementar um modelo de Programação Linear Inteira capaz de gerar escalas de motoristas legalmente válidas, operacionalmente adequadas e alinhadas ao Regulamento (CE) n.º 561/2006?

1.4 Objetivo Geral

Desenvolver, implementar e avaliar um modelo de Programação Linear Inteira para o escalonamento de motoristas no transporte rodoviário europeu, assegurando conformidade com o Regulamento (CE) n.º 561/2006 e promovendo melhoria do desempenho operacional.

1.5 Objetivos Específicos

(i) Formular matematicamente o problema, definindo variáveis, parâmetros e restrições; (ii) Representar as exigências legais do Regulamento (CE) n.º 561/2006 sob a forma de inequações lineares; (iii) Implementar o modelo utilizando Python e o solver OR-Tools; (iv) Desenvolver uma interface interativa para simulação de cenários e visualização de resultados; (v) Validar o modelo em diferentes horizontes (7, 15 e 30 dias); (vi) Comparar a alocação gerada com a demanda real ou simulada; (vii) Avaliar métricas de desempenho, como cobertura, subutilização e sobrecarga; (viii) Analisar a estabilidade do modelo e o

impacto das transformações matriciais. (ix) Implementar e avaliar modos complementares de resolução e melhoria de soluções (por exemplo, geração de solução inicial e busca em vizinhanças), bem como mecanismos de relaxação e registro de histórico de convergência; (x) Estruturar rotinas de experimentação com geração de instâncias/datasets e comparação sistemática de desempenho por meio de métricas e visualizações no simulador.

1.6 Justificativa

(i) **Relevância legal e segurança rodoviária:** o descumprimento do Regulamento (CE) n.º 561/2006 implica multas, riscos à vida dos motoristas e impactos negativos na reputação das empresas; (ii) **Complexidade operacional:** a variabilidade de demanda e a natureza acumulativa das restrições dificultam a elaboração manual de escalas; (iii) **Contribuição científica:** poucos trabalhos integram o Regulamento (CE) n.º 561/2006 a modelos inteiros formais; esta dissertação contribui para preencher essa lacuna, em linha com (Erdman; Koch, 2022), (Moreira; Pinheiro; Araujo, 2025); (iv) **Aplicabilidade industrial:** o modelo possui potencial de aplicação em empresas de transporte, podendo evoluir para soluções SaaS e integração com sistemas telemáticos.

1.7 Estrutura da Dissertação

Esta dissertação está organizada em nove capítulos, além das referências, apêndices e anexos, estruturados de forma a conduzir o leitor desde a contextualização do problema até a consolidação das contribuições científicas e perspectivas futuras.

O **Capítulo 1** apresenta a introdução do trabalho, incluindo a contextualização do problema, a definição da questão de pesquisa, os objetivos geral e específicos, a justificativa do estudo e a descrição da estrutura da dissertação.

O **Capítulo 2** é dedicado à revisão da literatura, abordando os fundamentos do transporte rodoviário europeu, as restrições regulatórias aplicáveis, os principais problemas de escalonamento, técnicas de modelagem com Programação Linear Inteira, metaheurísticas, o uso de solvers modernos como o OR-Tools CP-SAT, bem como trabalhos correlatos relevantes.

O **Capítulo 3** apresenta as bases legais e normativas do escalonamento de motoristas, com uma análise detalhada do Regulamento (CE) n.º 561/2006, da Diretiva 2002/15/CE e do Regulamento (UE) n.º 165/2014, discutindo suas implicações diretas na formulação do modelo matemático.

O **Capítulo 4** descreve a metodologia adotada na pesquisa, detalhando a estrutura geral do estudo, as etapas de formulação matemática, implementação computacional, definição dos cenários experimentais e os métodos de avaliação das soluções. Este capítulo

também introduz a abordagem heurística e matheurística proposta, incluindo a utilização de Large Neighborhood Search (LNS) e a integração opcional com técnicas de aprendizagem de máquina.

O **Capítulo 5** apresenta o modelo matemático proposto, incluindo a discretização temporal, a definição de conjuntos, parâmetros e variáveis de decisão, as funções objetivo e o conjunto completo de restrições que garantem a conformidade com a legislação europeia, além de uma discussão sobre linearização e observações computacionais.

O **Capítulo 6** trata da implementação computacional do modelo, descrevendo a arquitetura geral da solução, a interface de parametrização, a construção da matriz de restrições, os modos de resolução do solver, os fluxos computacionais do simulador, bem como os mecanismos de visualização, análise iterativa e geração de indicadores operacionais.

O **Capítulo 7** apresenta os resultados obtidos a partir dos experimentos realizados, incluindo a configuração dos cenários, a análise do comportamento da demanda, a comparação entre demanda e alocação, os indicadores operacionais, a avaliação dos diferentes modos de resolução e a síntese geral dos resultados.

O **Capítulo 8** realiza uma revisão comparativa da literatura, confrontando os modelos e técnicas desenvolvidos neste trabalho com o estado da arte, incluindo uma análise crítica das abordagens existentes e a discussão do pipeline híbrido de otimização proposto.

Por fim, o **Capítulo 9** apresenta as conclusões do trabalho, sintetizando as principais contribuições científicas, os resultados alcançados, as limitações identificadas e as direções para trabalhos futuros.

As referências bibliográficas, o glossário, os apêndices e os anexos complementam o trabalho, fornecendo detalhamentos adicionais sobre formulações alternativas do modelo, cenários de simulação, parâmetros utilizados, estruturas de dados, fluxos da aplicação e documentação normativa.

2 Revisão da Literatura e Posicionamento Científico

2.1 Introdução

Este capítulo apresenta o referencial teórico que fundamenta o desenvolvimento do modelo de otimização e do simulador computacional propostos. A revisão não se limita a uma descrição conceitual da literatura, mas busca estabelecer uma conexão direta entre os trabalhos existentes e as decisões de modelagem, implementação e experimentação adotadas ao longo do trabalho.

São discutidos aspectos relativos ao transporte rodoviário europeu sob regulamentação estrita, à modelagem matemática baseada em Programação Linear Inteira (PLI), aos problemas de escalonamento com janelas temporais móveis, ao uso de metaheurísticas e matheurísticas em problemas de grande porte, bem como à aplicação de solvers modernos capazes de lidar com estruturas temporais complexas.

Adicionalmente, a revisão destaca lacunas existentes na literatura, em especial no que se refere à integração profunda entre conformidade regulatória, granularidade temporal fina, análise estrutural do modelo e experimentação híbrida, aspectos que motivam e justificam a abordagem proposta nesta dissertação. Por fim, este capítulo estabelece o posicionamento científico do trabalho, identificando suas principais contribuições em relação ao estado da arte.

2.2 Transporte Rodoviário Europeu e as Restrições Regulatórias

O transporte rodoviário de cargas desempenha um papel central na cadeia logística europeia, responsável pela circulação de grande parte dos produtos entre países-membros. Para garantir segurança e padronização operacional, o Regulamento (CE) n.º 561/2006 estabelece limites sobre o tempo de condução, pausas e períodos de descanso. Essas regras impõem uma estrutura de dependências temporais dinâmicas ao planejamento operacional das empresas transportadoras.

Modelos tradicionais de planejamento nem sempre não incorporam de forma explícita essas janelas móveis — diárias, semanais e quinzenais que determinam limites máximos de condução contínua, acumulada e condições específicas de descanso, o que resulta em formulações incapazes de capturar adequadamente o acoplamento temporal e a complexidade combinatória induzida pela regulamentação. A literatura identifica esse tipo de problema como estruturalmente complexo, dadas as dependências entre períodos

consecutivos (**pillac2013review**).

Assim, soluções computacionais voltadas ao escalonamento de motoristas devem considerar esse conjunto de restrições regulatórias, sob pena de gerar soluções inviáveis e não conformes. Modelos adequadamente formulados precisam integrar, além das condições operacionais, os elementos legais que afetam diretamente o comportamento das variáveis de decisão.

Do ponto de vista da modelagem matemática, essas regras introduzem dependências temporais não locais, uma vez que o cumprimento de uma restrição em determinado período depende do histórico acumulado em janelas móveis de 24 horas, 7 dias e 14 dias. Tal característica distingue o escalonamento de motoristas de problemas clássicos de alocação de pessoal, exigindo formulações capazes de capturar memória temporal e reinícios condicionais, frequentemente por meio de variáveis auxiliares e restrições acumulativas.

2.3 Problemas de Escalonamento e Abordagens Estruturadas

O escalonamento de motoristas pertence à família dos problemas de escalonamento de pessoal (*workforce scheduling*), que envolvem a alocação de indivíduos a atividades ao longo do tempo, respeitando requisitos de capacidade, janelas temporais e restrições operacionais. Tais problemas são tipicamente formulados como modelos inteiros binários, dada a necessidade de representar decisões discretas de trabalhar ou descansar.

Estudos clássicos indicam a adequação da PLI para representar essas estruturas, sobretudo quando associadas a restrições temporais dependentes, como no caso de motoristas profissionais, demonstrando como decisões temporais de alocação podem ser representadas por matrizes binárias, combinadas a restrições lógicas e acumulativas.

Além disso, (**taylor2019introduction**) destaca que problemas de escalonamento frequentemente são NP-difíceis e exigem formulações que lidem com alta dimensionalidade. Restrições de janelas móveis, comuns em regulamentações como a europeia, exigem especial atenção para evitar inconsistências ou sobrecargas de condução.

No contexto específico do escalonamento de motoristas, a formulação em PLI frequentemente assume uma estrutura matricial binária do tipo motorista \times período, o que leva a modelos de alta dimensionalidade quando se adota granularidade temporal fina. Essa característica impõe desafios computacionais relevantes, ao mesmo tempo em que permite uma representação precisa das regras legais e operacionais, desde que acompanhada por técnicas adequadas de linearização e decomposição temporal.

2.4 Modelagem Matemática com Programação Linear Inteira

Ao longo deste trabalho, os termos Programação Linear Inteira (PLI), *Integer Linear Programming* (ILP) e *Mixed Integer Linear Programming* (MILP) são utilizados de forma intercambiável, conforme a convenção adotada na literatura internacional.

A Programação Linear Inteira (PLI) é amplamente utilizada para modelar problemas combinatórios na área de logística. (**hillier2021introduction**) destaca que variáveis binárias e restrições lineares permitem representar condições complexas como limites de tempo, dependências condicionais e estruturas acumulativas – características essenciais no escalonamento sob regulamentação.

Modelos híbridos envolvendo ILP e heurísticas também são discutidos na literatura. (**pinheiro2011densitycontrol**) explora técnicas de otimização combinando busca local, geração de soluções candidatas e modelos exatos. Essas abordagens mostram que, embora métodos exatos proporcionem precisão, heurísticas podem auxiliar na redução do espaço de busca e no aprimoramento da solução inicial.

No contexto do transporte, (**pinheiro2007transporte**) aplica modelos matemáticos ao problema de otimização de transporte de passageiros, destacando o papel da modelagem formal no atendimento a requisitos operacionais e normativos.

2.5 Metaheurísticas e Técnicas Avançadas de Otimização

Modelos baseados em metaheurísticas são particularmente utilizados para problemas de corte, empacotamento, roteamento e escalonamento. Trabalhos como os de (**pinheiro2016nesting**) demonstram que algoritmos híbridos, como técnicas de geração e busca, são eficazes em problemas estruturados com alta combinatorialidade. Essas contribuições evidenciam a relevância de técnicas híbridas para solucionar problemas que, assim como o escalonamento de motoristas, possuem espaços de busca muito grandes.

Entre as abordagens metaheurísticas, o método de *Large Neighborhood Search* (LNS) destaca-se como adequado para problemas de escalonamento com forte acoplamento temporal. O LNS permite a destruição e reconstrução parcial da solução, mantendo parte das decisões fixas enquanto reotimiza subconjuntos do problema, frequentemente com apoio de modelos exatos. Essa característica torna o LNS especialmente atraente para integração com PLI, configurando abordagens conhecidas como matheurísticas.

2.6 O Uso de Solvers Modernos: OR-Tools e CP-SAT

O CP-SAT, solver principal da Google OR-Tools, é particularmente adequado para problemas com grande quantidade de variáveis binárias e restrições lógicas, como é o caso

do escalonamento de motoristas sob regulamentação europeia. Sua capacidade de combinar técnicas de programação por restrições, propagação de domínios e aprendizado de cláusulas permite lidar de forma eficiente com janelas temporais móveis e restrições acumulativas, mitigando limitações observadas em solvers puramente baseados em programação linear inteira. Estudos da Google Research (**google2025ortools**) mostram que CP-SAT supera solvers tradicionais em várias classes de problemas combinatórios.

O uso de solvers híbridos é apoiado pela literatura moderna. A proximidade estrutural entre o escalonamento e problemas como roteamento e corte pode ser observada em trabalhos como (**pinheiro2019vrp**), que aplicam algoritmos probabilísticos ao problema de roteamento de veículos com coleta e entrega simultâneas. Esses modelos possuem restrições temporais e operacionais análogas, reforçando a aplicabilidade da OR-Tools ao problema tratado nesta dissertação.

2.7 Trabalhos Correlatos sobre Escalonamento e Logística

Há uma diversidade de estudos aplicados ao escalonamento e otimização no contexto de logística. (**moreira2024modeling**) aborda minimização de custos logísticos em empresas de transporte rodoviário, enquanto (**moreira2025optimization**) apresenta um modelo preliminar para escalonamento de motoristas sob regulamentação.

No campo de *scheduling*, trabalhos como (**pinheiro2004timetabling**) demonstram estruturas formais que são semelhantes às aquelas utilizadas no presente estudo: variáveis binárias, controle de janelas e dependências temporais.

Tais contribuições reforçam o espaço de pesquisa e motivam o desenvolvimento de modelos progressivamente mais abrangentes e alinhados às necessidades reais do setor.

2.8 Estado da Arte e Análise Comparativa

A literatura sobre escalonamento de motoristas e problemas de roteirização com restrições regulatórias evoluiu substancialmente nas últimas décadas, especialmente após a consolidação do Regulamento (CE) n.º 561/2006. Esta seção apresenta uma análise comparativa abrangente entre os principais trabalhos relacionados ao tema, incluindo estudos recentes sobre conformidade regulamentar, roteirização com pausas e escalonamento temporal.

2.8.1 Comparação Estruturada de Trabalhos Relevantes

A Tabela 1 sintetiza os elementos estruturais essenciais de cada estudo representativo, permitindo uma avaliação direta das diferenças conceituais, metodológicas e

aplicadas.

Tabela 1 – Comparação entre estudos sobre escalonamento, roteirização e conformidade regulatória

Trabalho	Abordagem	Metodologia	Objetivos	Contribuições
Moreira et al. (2024)	Otimização exata para alocação de frota	Modelo ILP; LINGO	Minimizar custos; otimizar alocação	Redução de custos; base conceitual
Moreira (2025)	Escalonamento completo sob legislação europeia	MILP; OR-Tools CP-SAT	Minimizar motoristas; 100% compliance	Modelo robusto e escalável
C&OR (2025)	Heurística para VRP com pausas	Construtivos e melhoria	Inserir pausas com mínimo impacto	Redução 6.1% distância
Pillai & Ulmanen (2019)	Escalonamento com foco regulatório	MILP + heurísticas	Minimizar violações Reg. 561/2006	Redução significativa de violações
Blöchliger (2004)	Roteirização com pausas explícitas	Formulações matemáticas	Representar pausas legalmente	Base teórica para métodos posteriores
Savelsbergh & Sol (1995)	Formulação estruturada GPDP	Modelos matemáticos clássicos	Base conceitual coleta-entrega	Referência estrutural amplamente utilizada

2.8.2 Análise Crítica Comparativa

A análise revela que cada trabalho atua em um eixo metodológico complementar. O estudo de (**blochliger2004modeling**) introduz a integração fundamental entre pausas e roteirização, conceito que é refinado em trabalhos posteriores sobre VRP. Savelsbergh e Sol (1995) fornecem a estrutura matemática que dá base a praticamente todos os modelos de *pickup-and-delivery* e variantes modernas de VRP.

Os trabalhos de (**moreira2024modeling**; **moreira2025optimization**) avançam o estado da arte ao aplicar formulações exatas completas para o problema de escalonamento sob as normas europeias. O trabalho mais recente apresenta o modelo mais completo da literatura no que diz respeito à conformidade temporal com todas as restrições do Regulamento (CE) n.º 561/2006, superando abordagens anteriores tanto em granularidade quanto em completude.

O estudo de (**pillai2019compliance**), por sua vez, atua em um espaço intermediário: combina MILP com heurísticas de correção, alcançando bons resultados de conformidade sem necessidade de modelos totalmente exatos em grandes horizontes.

2.8.3 Comparação Técnica das Abordagens

A Tabela 2 apresenta uma análise detalhada das técnicas empregadas em diferentes trabalhos, evidenciando a originalidade das soluções propostas.

Tabela 2 – Comparação técnica entre abordagens de escalonamento

Critério	Literatura Clássica	Trabalho Atual	Originalidade
Modelo Base	ILP tradicional (discretização 30-60 min)	ILP customizado (15 min, 96 períodos)	Alta
Heurística Construtiva	Regras sem estrutura matricial	Greedy baseada em matriz período \times motorista	Inédita
Uso de Densidade	Raramente discutida	Métrica de pré-processamento e complexidade	Original
Transformações Elementares	Inexistente	Troca de linhas, combinação linear	Inédita
Integração Heurística + MILP	Poucos trabalhos integram	Pipeline híbrido (Greedy \rightarrow LNS \rightarrow MILP)	Alta
LNS	Baseado em custo ou aleatoriedade	Guiado por densidade e sensibilidade temporal	Original
Granularidade Temporal	30-60 minutos	15 minutos	Superior
Escalabilidade	Dependente do solver	Alta (pipeline híbrido)	Melhorada

2.9 Lacunas Identificadas na Literatura

A revisão da literatura realizada evidencia que o escalonamento de motoristas no transporte rodoviário europeu constitui um problema de elevada complexidade estrutural, decorrente da interação entre requisitos operacionais, restrições legais e dependências temporais acumulativas. Diferentemente de problemas clássicos de alocação de recursos, o escalonamento sob o Regulamento (CE) n.º 561/2006 impõe janelas móveis de decisão (24 horas, 7 dias e 14 dias), limites acumulados de condução e regras condicionais de descanso, o que torna inadequadas abordagens simplificadas ou puramente heurísticas sem mecanismos explícitos de verificação de conformidade.

A literatura analisada demonstra que a Programação Linear Inteira (PLI) é uma das abordagens adequadas para representar formalmente esse tipo de problema, uma vez que permite modelar decisões discretas, restrições acumulativas e vínculos lógicos de forma formal. Trabalhos clássicos e contemporâneos confirmam que modelos baseados em variáveis binárias, quando corretamente formulados, conseguem capturar com precisão regras regulatórias complexas, desde que acompanhados de técnicas adequadas de discretização temporal e linearização.

Entretanto, também se observa que modelos exatos, quando aplicados isoladamente, podem enfrentar limitações de escalabilidade em instâncias de grande porte ou quando

submetidos a demandas altamente variáveis. Por esse motivo, a literatura recente aponta uma tendência em direção a abordagens híbridas, que combinam modelos matemáticos exatos com heurísticas construtivas, métodos de busca em grandes vizinhanças (*Large Neighborhood Search* – LNS) e estratégias de decomposição ou relaxação. Essas abordagens permitem explorar o espaço de soluções de forma mais eficiente, mantendo a consistência com o modelo regulatório subjacente.

Os estudos revisados também destacam o papel crescente de solvers modernos, como o CP-SAT da biblioteca OR-Tools, que combinam técnicas de programação por restrições, satisfação booleana e otimização inteira. A capacidade desses solvers de lidar com modelos de grande dimensão, fortemente acoplados no tempo e com estruturas matriciais esparsas, torna-os adequados para problemas de escalonamento sob regulamentação europeia.

Além disso, a literatura evidencia que o uso de métricas estruturais — como densidade da matriz de restrições, padrões de esparsidade e comportamento temporal das variáveis — pode fornecer informações valiosas sobre a dificuldade do problema e o desempenho esperado do solver. No entanto, observa-se que poucos trabalhos exploram explicitamente essas métricas como instrumentos de diagnóstico, pré-processamento ou orientação heurística, o que sugere a existência de uma lacuna no estado da arte.

Outro aspecto emergente identificado na revisão é a incorporação de técnicas de aprendizado de máquina como mecanismos auxiliares de apoio à decisão em problemas de otimização combinatória. Embora ainda incipiente no contexto do escalonamento de motoristas sob regulamentação, essa linha de pesquisa sugere que modelos supervisionados podem ser utilizados para estimar a qualidade de decisões locais, priorizar regiões promissoras do espaço de busca ou acelerar processos heurísticos, desde que integrados de forma não intrusiva e com garantias de reprodutibilidade.

De forma geral, a literatura revela um cenário fragmentado: por um lado, modelos de roteirização e VRP focam majoritariamente em aspectos espaciais e de custo; por outro, modelos regulatórios concentram-se nas restrições temporais, frequentemente sem integração com estratégias avançadas de busca ou análise estrutural. Poucos trabalhos apresentam uma abordagem integrada que combine: (i) formulação exata completa das restrições legais, (ii) mecanismos heurísticos e matheurísticos de melhoria, (iii) instrumentação analítica do comportamento do modelo e (iv) suporte à experimentação sistemática.

2.10 Posicionamento e Contribuições da Dissertação

Nesse contexto, a presente dissertação posiciona-se como uma contribuição que busca integrar os diferentes eixos identificados nas lacunas da literatura, utilizando a Programação Linear Inteira como núcleo formal, complementada por heurísticas construtivas, métodos LNS, análise estrutural da matriz de restrições e, de forma opcional, mecanismos de

aprendizado supervisionado. Essa integração visa não apenas resolver instâncias específicas, mas também oferecer uma plataforma experimental capaz de apoiar análises comparativas, estudos de sensibilidade e investigações científicas mais amplas sobre o escalonamento de motoristas sob restrições legais complexas.

2.10.1 Contribuições Científicas

As contribuições científicas desta dissertação podem ser consolidadas em múltiplas dimensões complementares:

Contribuição 1 - Modelagem Matemática: Formulação matemática completa e detalhada do escalonamento de motoristas sob o Regulamento (CE) n.º 561/2006, com granularidade temporal fina (15 minutos) e representação explícita de todas as principais restrições legais. Essa formulação avança o estado da arte ao superar simplificações comuns na literatura e ao demonstrar que a conformidade integral é compatível com desempenho computacional prático.

Contribuição 2 - Algorítmica: Desenvolvimento de uma heurística construtiva matricial inédita, baseada na relação entre demanda, cobertura e carga acumulada por motorista. Essa abordagem difere de heurísticas baseadas exclusivamente em regras locais, ao explorar a estrutura matricial período \times motorista como elemento central da tomada de decisão. Adicionalmente, propõe-se um método LNS guiado por densidade da matriz de restrições, permitindo melhorias progressivas da solução com controle explícito do esforço computacional.

Contribuição 3 - Análise Estrutural: Introdução de operações elementares sobre a matriz de restrições como ferramenta analítica e educacional. Embora tais operações sejam clássicas em álgebra linear, sua aplicação sistemática no contexto de escalonamento e otimização inteira é pouco explorada na literatura, abrindo novas possibilidades para análise estrutural e diagnóstico de modelos.

Contribuição 4 - Plataforma Experimental: Desenvolvimento de um pipeline híbrido integrando heurística, LNS e MILP, demonstrando que métodos exatos e metaheurísticos podem coexistir de forma sinérgica em um mesmo ambiente computacional, mantendo consistência legal e interpretabilidade. A plataforma oferece três modos de resolução (exato, heurístico, LNS) com integração opcional de aprendizado de máquina supervisionado como camada de apoio à decisão.

Contribuição 5 - Validação Prática: Demonstração de conformidade 100% com o Regulamento (CE) n.º 561/2006 em todos os cenários testados, com tempos de resolução inferiores a 2 segundos para horizontes de 24 horas, viabilizando aplicações reais em contextos de replanejamento dinâmico e sistemas de apoio à decisão operacional.

2.10.2 Posicionamento Científico

Do ponto de vista do posicionamento científico, esta dissertação ocupa um espaço de interseção entre três grandes vertentes da literatura: (i) formulações exatas de escalonamento com foco em conformidade legal, (ii) heurísticas e metaheurísticas voltadas à escalabilidade e eficiência computacional e (iii) abordagens híbridas que exploram aprendizado de máquina como mecanismo de apoio.

Enquanto grande parte da literatura trata esses eixos de forma isolada, o presente trabalho demonstra que é possível integrá-los em um único simulador, preservando rigor, reprodutibilidade e capacidade de análise comparativa. Essa integração amplia significativamente o escopo experimental e permite investigar, de forma sistemática, *trade-offs* entre qualidade da solução, tempo computacional e estabilidade operacional.

Assim, o trabalho não se limita a propor um novo modelo matemático, mas oferece uma plataforma de experimentação científica capaz de sustentar análises empíricas profundas e extensões futuras.

2.11 Síntese e Organização da Dissertação

A partir dessa síntese, estabelece-se o arcabouço conceitual e metodológico que fundamenta a formulação do modelo matemático (Capítulo 5), da implementação computacional (Capítulo 6) e do desenho experimental (Capítulo 7) apresentados nos capítulos subsequentes. O Capítulo 3 detalha as bases legais e normativas que fundamentam as restrições do modelo, enquanto o Capítulo 4 apresenta a metodologia completa adotada.

Os resultados experimentais são discutidos no Capítulo 7, seguidos por uma análise comparativa aprofundada com a literatura no Capítulo 8. Por fim, o Capítulo 9 apresenta as conclusões, limitações identificadas e direções para trabalhos futuros, consolidando as contribuições científicas e o posicionamento da dissertação no contexto do estado da arte em escalonamento de motoristas sob restrições regulatórias complexas.

3 Bases Legais e Normativas do Escalonamento de Motoristas

3.1 Introdução

O transporte rodoviário europeu é regulamentado por um conjunto extenso de normas destinadas a promover a segurança viária, garantir condições adequadas de trabalho aos motoristas e assegurar a harmonização das práticas entre os Estados-Membros. A formulação do modelo de Programação Linear Inteira (PLI) apresentado nesta dissertação deriva diretamente dessas normas, que estabelecem limites específicos de condução, períodos mínimos de descanso e requisitos tecnológicos relacionados ao registo de atividades.

Além de definir limites normativos, esses diplomas legais impõem uma estrutura temporal altamente acoplada, caracterizada por janelas móveis, restrições acumulativas e dependências entre períodos consecutivos. Tais características tornam o escalonamento de motoristas um problema intrinsecamente complexo do ponto de vista computacional, exigindo formulações matemáticas capazes de capturar não apenas estados locais, mas também históricos de condução e descanso. Essa complexidade normativa é um dos principais motivadores para a adoção de Programação Linear Inteira, bem como para a exploração de estratégias complementares de resolução e melhoria de soluções, conforme desenvolvido nos capítulos subsequentes.

Este capítulo tem por objetivo apresentar uma síntese estruturada das principais bases legais aplicáveis ao escalonamento de motoristas na União Europeia, com ênfase no Regulamento (CE) n.º 561/2006, na Diretiva 2002/15/CE e no Regulamento (UE) n.º 165/2014. A compreensão desses diplomas jurídicos é fundamental para contextualizar as restrições incorporadas ao modelo matemático discutido no Capítulo 4.

Ressalta-se que as normas aqui descritas não são tratadas apenas como referências teóricas, mas como elementos diretamente observáveis e verificáveis em dados reais de operação. Os parâmetros utilizados ao longo desta dissertação refletem registros efetivos de jornadas, tempos de condução e períodos de descanso, conforme capturados por sistemas operacionais e dispositivos de registro, garantindo aderência prática e validade empírica ao modelo proposto.

3.2 Regulamento (CE) n.º 561/2006

O Regulamento (CE) n.º 561/2006 estabelece as regras relativas aos tempos de condução, pausas e períodos de repouso dos motoristas envolvidos em operações de

transporte rodoviário. Seu objetivo principal é melhorar a segurança viária, prevenir a fadiga, reduzir acidentes e garantir condições de trabalho adequadas. Os principais requisitos incluem:

3.2.1 Limites de Condução

- **Condução diária:** máximo de 9 horas, podendo ser estendido a 10 horas até duas vezes por semana.
- **Condução semanal:** máximo de 56 horas.
- **Condução quinzenal:** máximo de 90 horas em dois períodos consecutivos de 7 dias.

3.2.2 Pausas Obrigatórias

Após um período máximo de 4,5 horas de condução contínua, o motorista deve realizar uma pausa mínima de 45 minutos, podendo ser fracionada em:

- 15 minutos + 30 minutos,
- Desde que ambos os períodos ocorram dentro das 4,5 horas.

3.2.3 Descanso Diário

O período de descanso diário pode ser:

- **Normal:** pelo menos 11 horas consecutivas;
- **Reduzido:** pelo menos 9 horas consecutivas, permitido até três vezes entre dois descansos semanais.

3.2.4 Descanso Semanal

- **Regular:** ao menos 45 horas contínuas;
- **Reduzido:** mínimo de 24 horas, com compensação até o final da terceira semana seguinte.

3.2.5 Implicações para o modelo matemático

Essas regras originam diretamente:

- Restrições de limites diários, semanais e quinzenais de trabalho;
- Restrições acumulativas (janelas móveis);
- Restrições de continuidade temporal de descanso;
- Necessidade de variáveis auxiliares para extensão diária e pausas.

Do ponto de vista da modelagem, o Regulamento (CE) n.º 561/2006 não pode ser interpretado como um conjunto de restrições isoladas. Seus limites diários, semanais e quinzenais operam simultaneamente e sobrepostos, criando janelas móveis de verificação que se deslocam ao longo do horizonte de planejamento. Essa característica impede abordagens puramente locais ou estáticas e justifica a necessidade de variáveis de estado acumulado, mecanismos de reinício após descanso e controle explícito de extensões diárias, todos incorporados ao modelo matemático apresentado no Capítulo 4.

3.3 Diretiva 2002/15/CE

A Diretiva 2002/15/CE trata do *tempo de trabalho* dos trabalhadores móveis. Complementa o Regulamento (CE) n.º 561/2006 ao definir períodos que não são estritamente de condução, mas que fazem parte da jornada de trabalho.

Os principais elementos incluem:

- limite de 48 horas de trabalho semanal (média), podendo atingir 60 horas quando a média de 48 horas for respeitada no período de quatro meses;
- definição de atividades como:
 - carga e descarga;
 - espera operacional;
 - assistência documental e logística;
 - preparação e encerramento de veículo.
- períodos de disponibilidade não contabilizados como trabalho.

Embora o foco principal deste estudo recaia sobre os tempos de condução regulados pelo Regulamento (CE) n.º 561/2006, os conceitos introduzidos pela Diretiva 2002/15/CE são relevantes para a interpretação dos dados reais utilizados. Em particular, registros operacionais de jornada frequentemente incluem atividades não classificadas como condução, mas que impactam diretamente a disponibilidade do motorista. Assim, mesmo quando

não modelados explicitamente como variáveis de decisão, esses tempos influenciam a parametrização do problema e a análise dos resultados obtidos.

Embora esta dissertação concentre-se na parte de condução, os tempos descritos pela Diretiva representam potencial extensão futura do modelo (ver Capítulo 7).

3.4 Regulamento (UE) n.º 165/2014

Este regulamento estabelece as regras relativas ao tacógrafo digital, dispositivo responsável pelo registro eletrônico da condução, pausas, velocidade e distâncias. Sua principal finalidade é assegurar a fiscalização adequada do cumprimento do Regulamento (CE) n.º 561/2006. Entre seus pontos mais relevantes:

- obrigatoriedade de tacógrafos inteligentes em veículos de transporte internacional;
- registro automático de localização por GNSS;
- salvaguarda e criptografia dos dados de condução;
- integração com sistemas de controle e fiscalização.

No contexto desta dissertação, o Regulamento (UE) n.º 165/2014 estabelece o elo entre o modelo teórico e a realidade operacional. Embora os experimentos utilizem dados simulados, a estrutura do modelo foi concebida de forma compatível com os registros gerados por tacógrafos digitais, permitindo, em princípio, a substituição direta de dados sintéticos por dados reais. Essa compatibilidade reforça o caráter aplicável do simulador e viabiliza sua futura integração com sistemas telemáticos e plataformas de gestão de frotas.

Embora não estabeleça limites operacionais adicionais, o Regulamento (UE) n.º 165/2014 define o mecanismo tecnológico que assegura a rastreabilidade, confiabilidade e auditabilidade dos dados utilizados nesta pesquisa. Os registros provenientes de tacógrafos digitais constituem a principal fonte empírica para a validação do cumprimento do Regulamento (CE) n.º 561/2006, reforçando a necessidade de que o modelo matemático esteja alinhado à estrutura e à granularidade dos dados reais disponíveis.

3.5 Síntese das Bases Legais

O conjunto normativo formado pelo Regulamento (CE) n.º 561/2006, pela Diretiva 2002/15/CE e pelo Regulamento (UE) n.º 165/2014 estabelece um arcabouço legal altamente restritivo e tecnicamente detalhado para o escalonamento de motoristas no transporte rodoviário europeu.

Essas normas impõem:

- limites temporais rígidos e cumulativos;
- janelas móveis de avaliação (24 horas, semanal e quinzenal);
- forte interdependência entre períodos de condução, pausa e descanso;
- necessidade de monitoramento contínuo e verificável;
- aderência aos registros reais provenientes de tacógrafos digitais.

Do ponto de vista desta dissertação, tais características possuem impacto direto não apenas na formulação matemática, mas também na natureza dos dados utilizados. Os parâmetros empregados refletem situações reais de operação, capturadas por sistemas de registro e gestão, o que exige que o modelo de Programação Linear Inteira seja capaz de reproduzir fielmente a lógica normativa observada nos dados empíricos.

3.6 Relação com o Modelo Matemático

Cada restrição normativa apresentada neste capítulo corresponde a um conjunto de inequações no Capítulo 4. Essa abordagem garante:

- conformidade legal;
- capacidade de auditoria;
- rastreabilidade das decisões;
- transparência operacional.

Assim, o modelo matemático não apenas resolve um problema de otimização, mas também se alinha estritamente ao quadro regulatório europeu.

Além disso, a utilização de dados reais impõe requisitos adicionais ao modelo, como robustez frente a variações operacionais, consistência temporal e capacidade de reproduzir padrões observados na prática. Dessa forma, o modelo matemático desenvolvido não se limita a satisfazer restrições abstratas, mas busca representar, de forma verificável, o comportamento real das jornadas de trabalho sob a legislação europeia.

Dessa forma, o Capítulo 4 apresenta a tradução formal dessas normas em um conjunto estruturado de variáveis, parâmetros, função objetivo e restrições lineares, estabelecendo a base matemática que sustenta toda a implementação computacional e os experimentos apresentados nos capítulos seguintes.

3.7 Considerações Finais

As normativas apresentadas neste capítulo formam a base conceitual e jurídica necessária para compreender o problema de escalonamento de motoristas. Elas estruturam todas as restrições incorporadas ao modelo matemático, justificando a sua formulação e motivando a atenção aos detalhes temporais abordados nos capítulos subsequentes.

4 Modelagem e Metodologia Experimental

4.1 Introdução

Este capítulo apresenta o desenvolvimento completo do modelo de Programação Linear Inteira (PLI) aplicado ao escalonamento de motoristas sob o Regulamento (CE) n.º 561/2006, integrando sua formulação matemática com a metodologia experimental adotada para validação e análise de desempenho.

A metodologia combina rigor matemático, técnicas de Pesquisa Operacional e ferramentas computacionais modernas, garantindo reprodutibilidade, precisão e aplicabilidade prática. A pesquisa pode ser classificada como:

- **Aplicada:** pois aborda um problema concreto do setor de transporte rodoviário.
- **Quantitativa:** devido ao uso de modelagem matemática e experimentação.
- **Metodológica:** por propor um artefato científico (o modelo ILP).
- **Experimental:** pela realização de testes controlados em cenários variados.

Essa abordagem está alinhada aos fundamentos apresentados no (Hillier; Lieberman, 2015), (Taha, 2017), (Nemhauser; Wolsey, 1988), bem como ao trabalho aplicado apresentado em (Moreira; Pinheiro; Araujo, 2025).

O objetivo do modelo é gerar escalas legalmente válidas, eficientes e capazes de atender à demanda operacional, respeitando simultaneamente limites de condução, pausas obrigatórias e períodos mínimos de descanso.

Formulações baseadas em PLI têm sido amplamente utilizadas para problemas de escalonamento com fortes dependências temporais, sobretudo em contextos regulados como transporte, saúde e serviços críticos (savelsbergh1997; Pillac; Guéret; Medaglia, 2013). Neste trabalho, o modelo proposto estende a formulação apresentada em (Moreira; Pinheiro; Araujo, 2025), incorporando maior granularidade temporal, variáveis auxiliares e uma estrutura de otimização compatível com o solver CP-SAT do OR-Tools (Google, 2025).

Diferentemente de abordagens puramente multiobjetivo, o modelo adota uma estrutura lexicográfica em duas fases, refletindo prioridades operacionais reais: primeiro garantir cobertura da demanda e minimizar o número de motoristas ativos, e somente depois refinar a solução por critérios de balanceamento e suavização de carga.

4.2 Formulação Matemática do Modelo

Esta seção apresenta a formulação completa do modelo de Programação Linear Inteira desenvolvido para o problema de escalonamento. Diferentemente de abordagens puramente multiobjetivo, o modelo adota uma estrutura lexicográfica em duas fases, refletindo prioridades operacionais reais: primeiro garantir cobertura da demanda e minimizar o número de motoristas ativos, e somente depois refinar a solução por critérios de balanceamento e suavização de carga.

4.3 Discretização Temporal

O horizonte de planejamento é discretizado em períodos fixos de duração $\Delta = 15$ minutos, formando o conjunto:

$$T = \{1, 2, \dots, |T|\}.$$

Essa discretização reflete a frequência real de chegada dos pedidos de carga na operação analisada e permite representar de forma linear restrições originalmente não lineares, como janelas móveis de descanso e limites acumulados de condução (Pillac; Guéret; Medaglia, 2013; Erdman; Koch, 2022).

4.4 Conjuntos e Índices

- D : conjunto de períodos discretizados ($d \in D$);
- T : conjunto de motoristas ($t \in T$);
- W : janelas agregadas de planejamento (dias, semanas e quinzenas).

4.5 Parâmetros

$demanda_d$: demanda operacional no período d ,

Δ : duração do período (15 minutos),

L^{dia} : limite diário de condução,

L^{sem} : limite semanal de condução,

L^{14d} : limite quinzenal de condução,

R^{dia} : descanso diário mínimo,

R^{sem} : descanso semanal mínimo.

Esses parâmetros refletem diretamente as exigências do Regulamento (CE) n.º 561/2006 e da Diretiva 2002/15/CE (Union, 2006, 2002).

4.6 Variáveis de Decisão

Ativação do motorista

$$Z_t = \begin{cases} 1, & \text{se o motorista } t \text{ é ativado,} \\ 0, & \text{caso contrário.} \end{cases}$$

Presença no período

$$Y_{d,t} = \begin{cases} 1, & \text{se o motorista } t \text{ está presente no período } d, \\ 0, & \text{caso contrário.} \end{cases}$$

Atendimento de demanda

$$X_{d,t} \in \mathbb{Z}_+, \quad 0 \leq X_{d,t} \leq \text{capacidade por slot.}$$

Demanda não atendida

$$U_d \geq 0.$$

Carga por motorista e carga máxima

$$L_t = \sum_{d \in D} X_{d,t}, \quad L_{\max} \geq L_t.$$

A separação entre variáveis de ativação, presença e carga segue práticas consolidadas na literatura de escalonamento, permitindo representar de forma explícita decisões operacionais e restrições legais (savelsbergh1997; pinheiro2007staffscheduling).

4.7 Função Objetivo Lexicográfica

O modelo utiliza uma estrutura de otimização em duas fases, resolvidas sequencialmente pelo solver.

4.7.1 Fase 1 — Cobertura e Minimização de Motoristas

A Fase 1 possui prioridade absoluta e busca:

1. minimizar a demanda não atendida;
2. minimizar o número de motoristas ativados;
3. incentivar maior utilização dos motoristas ativos.

A função objetivo é dada por:

$$\min \alpha \sum_{d \in D} U_d + \beta \sum_{t \in T} Z_t - \gamma \sum_{d \in D} \sum_{t \in T} X_{d,t},$$

com $\alpha \gg \beta \gg \gamma$, garantindo prioridade lexicográfica.

Estruturas desse tipo são amplamente utilizadas quando múltiplos objetivos conflitantes precisam ser tratados de forma hierárquica (Pillac; Guéret; Medaglia, 2013; Erdman; Koch, 2022).

4.7.2 Fase 2 — Balanceamento e Refinamento

Após fixar os valores ótimos de $\sum U_d$ e $\sum Z_t$, a Fase 2 refina a solução, buscando:

- reduzir presença desnecessária;
- penalizar excesso de carga acima de um limite suave;
- suavizar picos de carga entre motoristas.

A função objetivo da Fase 2 é:

$$\min \omega_1 \sum_{d,t} Y_{d,t} + \omega_2 \sum_t \max(0, L_t - L^{soft}) + \omega_3 L_{\max}.$$

Esses critérios promovem maior equidade e robustez operacional, sendo frequentemente empregados como etapa de refinamento em problemas de escalonamento (savelsbergh1997).

4.8 Restrições do Modelo

O modelo matemático incorpora um conjunto abrangente de restrições legais e operacionais derivadas do Regulamento (CE) n.º 561/2006. Entretanto, nem todas as restrições são necessariamente ativadas simultaneamente em todos os experimentos.

No simulador desenvolvido, cada restrição pode ser ativada ou desativada de forma independente por meio de parâmetros de configuração, permitindo a análise controlada

do impacto individual e combinado das regras legais sobre a solução. Essa abordagem possibilita estudos comparativos entre diferentes níveis de rigor regulatório, sem alterar a estrutura fundamental do modelo.

Formalmente, o conjunto de restrições ativas em um experimento é definido por um vetor binário de ativação, sendo que apenas as restrições selecionadas são incluídas no modelo resolvido pelo solver.

4.8.1 Atendimento da Demanda

Quando a restrição de cobertura obrigatória está ativada, o atendimento da demanda é modelado como uma restrição rígida:

$$\sum_{t \in T} X_{d,t} + U_d = demanda_d, \quad \forall d \in D.$$

Caso essa restrição não seja ativada, a variável U_d passa a representar déficits operacionais admissíveis, utilizados exclusivamente para fins de avaliação e cálculo de indicadores, sem inviabilizar a solução.

4.8.2 Vinculação entre presença e ativação

$$Y_{d,t} \leq Z_t, \quad \forall d, t.$$

4.8.3 Capacidade por período

$$X_{d,t} \leq capacidade \cdot Y_{d,t}.$$

4.8.4 Limites legais de condução

As restrições apresentadas a seguir são incluídas no modelo apenas quando explicitamente ativadas no cenário experimental considerado.

Restrições diárias, semanais e quinzenais são modeladas por janelas móveis, conforme exigido pelo Regulamento (CE) n.º 561/2006.

4.8.5 Pausas e descansos

Pausas após 4,5 horas de condução contínua e descansos diários e semanais são representados por restrições lineares baseadas em janelas deslizantes.

4.9 Discussão sobre Linearização

As regras legais apresentam dependências temporais não lineares, tratadas por meio de discretização, variáveis auxiliares e constantes Big-M. Essa abordagem é amplamente utilizada para manter a linearidade da formulação e compatibilidade com solvers inteiros modernos (Pillac; Guéret; Medaglia, 2013; Google, 2025).

4.10 Observações Computacionais

O modelo apresenta dezenas de milhares de variáveis e restrições, mas possui matriz esparsa e forte estrutura temporal. Solvers modernos, como o CP-SAT, exploram essas características de forma eficiente, permitindo resolver cenários reais em tempos compatíveis com uso operacional (Erdman; Koch, 2022; Google, 2025).

4.11 Metodologia de Implementação e Experimentação

Esta seção descreve o processo de implementação computacional do modelo matemático apresentado anteriormente, bem como a estrutura metodológica adotada para sua validação experimental.

4.12 Etapa 1: Estudo da Legislação Europeia

Nesta etapa, foi realizada uma análise minuciosa dos documentos legais que regem a jornada dos motoristas profissionais na União Europeia, incluindo:

- *Regulamento (CE) n.º 561/2006*;
- *Diretiva 2002/15/CE*;
- *Regulamento (UE) n.º 165/2014*.

As restrições extraídas desses documentos foram traduzidas em condições formais, como:

- limites de condução diária, semanal e quinzenal;
- pausas obrigatórias;
- descanso diário normal ou reduzido;
- descanso semanal;

- regras de continuidade temporal.

O desafio principal foi converter regras textuais em inequações lineares, conforme discutido em (Pillai; Ulmanen, 2019) e explorado experimentalmente em (Moreira; Pinheiro; Araujo, 2025).

4.13 Etapa 2: Formulação Matemática

Com base nos requisitos legais e operacionais, o problema foi formalizado como um modelo de Programação Linear Inteira (PLI). Os principais elementos definidos foram:

- conjuntos (motoristas, períodos, janelas temporais);
- variáveis de decisão (alocação, descanso, início de jornada, condução acumulada);
- parâmetros (demanda, limites legais, duração dos períodos);
- função objetivo (minimização lexicográfica do número de motoristas ativos e, em caso de empate, do número total de períodos alocados);
- restrições (legais e operacionais).

A função objetivo é estruturada de forma lexicográfica. O critério primário consiste na minimização do número de motoristas ativos ao longo do horizonte de planejamento. Em caso de soluções equivalentes sob esse critério, adota-se como critério secundário a minimização do esforço operacional total, mensurado pelo número de períodos alocados. Essa hierarquização reflete de maneira mais fiel os objetivos operacionais do transporte rodoviário, priorizando a redução do número de recursos humanos mobilizados antes da otimização fina da carga de trabalho.

A formulação foi construída com base em técnicas clássicas de otimização combinatória (Nemhauser; Wolsey, 1988), (Dantzig, 1963), mas adaptada ao contexto legislativo europeu. O modelo final é apresentado em detalhes no Capítulo 4, incluindo linearizações e implicações lógicas.

4.14 Etapa 3: Implementação Computacional

A implementação foi realizada em Python, com uso do solver CP-SAT do [Google](#), (2025). Essa etapa envolveu:

- criação programática das variáveis do modelo;

- construção da matriz de restrições;
- aplicação das regras legais e dos vínculos sequenciais;
- configuração de parâmetros do solver (limites de tempo, cortes, heurísticas);
- desenvolvimento de uma interface interativa em Streamlit;
- geração automática de relatórios, gráficos e indicadores operacionais.

A linguagem Python facilitou o uso de estruturas de dados eficientes, como `arr-Numys` ([Developers, 2025](#)) e (`tabelPpandasDocs`), além de permitir modularidade na construção do modelo.

4.15 Etapa 4: Definição dos Cenários Experimentais

Quatro tipos de cenário foram definidos para validação:

- **24 horas:** cenário básico, discretizado em 96 períodos de 15 minutos, utilizado para validação das restrições diárias de condução, pausas e início de jornada.
- **7 dias:** cenário compacto, útil para validar a recorrência das restrições diárias ao longo de múltiplos ciclos consecutivos.
- **15 dias:** cenário intermediário, crítico para a avaliação de restrições quinzenais e efeitos acumulativos de carga de trabalho.
- **30 dias:** cenário de escala real, empregado para testar a robustez temporal do modelo e sua aplicabilidade em horizontes de planejamento prolongados.

Os dados utilizados na definição das curvas de demanda e dos cenários experimentais são oriundos de registros operacionais reais de uma empresa do setor de transporte rodoviário. Por razões de confidencialidade, os dados foram previamente anonimizados e agregados, preservando-se apenas as características temporais e estruturais necessárias à modelagem.

A granularidade temporal adotada, correspondente a períodos de 15 minutos, reflete a frequência com que pedidos de carga são efetivamente recebidos e registrados nos sistemas operacionais da empresa. Cada período representa, portanto, um intervalo de tempo no qual novas demandas podem surgir e ser contabilizadas no respectivo *slot* temporal, permitindo capturar de forma fiel a dinâmica operacional observada na prática.

Parâmetros como duração de períodos, restrições legais e tamanho da força de trabalho foram configuráveis via interface.

Embora múltiplos cenários experimentais tenham sido avaliados, a apresentação gráfica detalhada de todos os casos resultaria em um volume excessivo de artefatos visuais, sem ganho proporcional de poder explicativo. Dessa forma, optou-se por apresentar de forma detalhada apenas o cenário de 24 horas, que permite uma análise clara e intuitiva do comportamento do modelo e dos métodos de resolução. Os demais cenários são analisados por meio de métricas agregadas e comparativas, evidenciando a consistência e a escalabilidade dos resultados.

4.16 Etapa 5: Análise dos Resultados

Após a execução do solver, foram analisados:

Estabilidade de Cobertura (σ) *Standard deviation of coverage levels across time slots.*

Mede a variabilidade da cobertura ao longo dos períodos de tempo. Valores baixos indicam uma alocação estável e homogênea, enquanto valores elevados revelam flutuações significativas entre excesso e falta de cobertura, caracterizando menor robustez temporal.

Slots com Déficit de Demanda *Number of time slots where allocated workers are fewer than required demand.*

Representa a quantidade de períodos em que a demanda não é totalmente atendida. Esse indicador reflete diretamente falhas operacionais e potenciais violações do nível de serviço.

Demanda Total Não Atendida *Total amount of unmet demand aggregated over all time slots.*

Quantifica o volume total de demanda não coberta ao longo do horizonte de planejamento. Diferentemente do número de slots deficitários, este indicador mede a magnitude acumulada do déficit.

Índice Global de Cobertura *Overall ratio between allocated capacity and required demand.*

Avalia o nível médio de atendimento do sistema. Valores próximos de 1.0 indicam equilíbrio entre oferta e demanda; valores inferiores indicam subalocação, enquanto valores superiores sugerem excesso estrutural de capacidade.

Gravidade do Risco *Weighted severity score based on uncovered demand and its distribution over time.*

Consolida a intensidade, duração e distribuição temporal dos déficits de cobertura em um único índice, permitindo classificar a solução em níveis de risco operacional.

Eficiência da Força de Trabalho *Ratio between productive allocated work and total available workforce capacity.*

Indica o grau de utilização efetiva da capacidade total dos trabalhadores. Alta eficiência sugere baixo nível de ociosidade, enquanto valores reduzidos indicam superdimensionamento ou má distribuição dos recursos.

Custo Estimado *Estimated operational cost derived from workforce allocation and penalty terms.*

Representa o custo total da solução, considerando horas trabalhadas, penalidades por déficit ou excesso e demais componentes definidos na função objetivo do modelo.

Estabilidade Temporal *Consistency of workforce allocation patterns over consecutive time slots.*

Avalia a suavidade das transições na alocação ao longo do tempo. Soluções com alta estabilidade reduzem mudanças abruptas, simplificando a execução operacional e aumentando a previsibilidade.

Capacidade em Excesso *Total amount of allocated capacity exceeding the required demand.*

Mede o desperdício de recursos decorrente da superalocação. Este indicador é fundamental para analisar o trade-off entre robustez da solução e eficiência econômica.

4.16.1 Abordagem heurística e matheurística proposta

O simulador desenvolvido permite a execução do modelo de escalonamento por meio de três modos distintos de resolução, selecionáveis pelo usuário: *Exato*, *Heurístico* e *Large Neighborhood Search (LNS)*. Cada modo corresponde a uma estratégia de resolução específica, mantendo-se inalterada a formulação matemática do problema.

No modo *Exato*, o problema é resolvido integralmente por meio de um solver de programação inteira, visando a obtenção da solução ótima global, sujeito às limitações computacionais inerentes a problemas de grande escala. O modo *Heurístico* emprega uma estratégia construtiva gulosa, capaz de produzir soluções viáveis de forma rápida, servindo tanto como solução inicial quanto como referência para análises comparativas. Já o modo *LNS* combina a solução heurística inicial com ciclos iterativos de destruição e reconstrução parcial, utilizando modelos MILP locais para refinar progressivamente a solução.

Embora o modelo de programação inteira mista (MILP) formulado neste trabalho permita obter soluções ótimas para instâncias de porte moderado, a sua aplicação direta em cenários de grande escala (muitos motoristas, múltiplos dias e diferentes perfis de demanda) pode tornar-se computacionalmente onerosa. Além disso, em ambientes operacionais dinâmicos, como o transporte rodoviário de mercadorias, nem sempre é necessário obter a

solução ótima global; muitas vezes, soluções de boa qualidade obtidas em tempos reduzidos são preferíveis.

Com esse contexto, propõe-se uma abordagem em três camadas: (i) uma heurística construtiva para gerar uma escala inicial factível, (ii) uma matheurística do tipo *Large Neighborhood Search* (LNS), em que subproblemas são resolvidos exatamente pelo modelo MILP/CP-SAT, e (iii) um módulo de apoio baseado em aprendizagem de máquina (*machine learning*) para guiar a seleção de vizinhanças promissoras e refinar decisões locais de alocação.

A heurística construtiva opera em uma linha do tempo discretizada (intervalos de 15 minutos) e, para cada período, atribui motoristas elegíveis de forma gulosa, respeitando as principais restrições regulatórias (condução contínua máxima, pausas mínimas, descanso diário e semanal, limites semanais e quinzenais). Esta fase produz rapidamente uma solução inicial factível, ainda que potencialmente distante do ótimo.

Na segunda camada, aplica-se uma matheurística LNS: a solução corrente é parcialmente destruída em sub-regiões (por exemplo, um dia específico, um subconjunto de motoristas ou janelas temporais críticas com sobrecarga de demanda), e um subproblema restrito é reotimizado com o modelo MILP proposto, sob limite de tempo. Assim, o modelo exato deixa de atuar sobre toda a instância e passa a ser utilizado como um *oráculo* de melhoria local, combinando a robustez de soluções exatas com a flexibilidade exploratória típica de heurísticas.

Por fim, a terceira camada explora técnicas de aprendizagem de máquina para apoiar a heurística e a matheurística. A partir de instâncias menores, nas quais o MILP consegue atingir a solução ótima em tempo aceitável, constroem-se conjuntos de dados que relacionam padrões de alocação (por motorista, por período e por janela temporal) à qualidade da solução. Modelos supervisionados podem então ser treinados para: (a) atribuir escores de preferência a pares motorista-período, auxiliando a etapa construtiva, e (b) estimar a probabilidade de melhoria ao selecionar determinadas vizinhanças na LNS, priorizando regiões da solução com maior potencial de ganho.

Essa arquitetura híbrida (MILP + heurística + LNS + ML) permite, por um lado, preservar o rigor e a completude do modelo regulatório desenvolvido e, por outro, oferecer tempos de resposta compatíveis com o uso em sistemas de apoio à decisão em contexto real, seja em ambiente acadêmico (simulações) ou em produto (como no contexto da plataforma Ottimizia).

4.16.1.1 Heurística construtiva para geração de escala inicial

A seguir apresenta-se a heurística construtiva proposta para geração de uma escala inicial factível.

Algoritmo 1: Heurística construtiva para escalonamento de motoristas

Input: Conjunto de motoristas D ;
 Conjunto de períodos discretizados T (intervalos de 15 minutos);
 Demanda de motoristas por período $\text{dem}(t)$;
 Parâmetros regulatórios (limites de condução e trabalho, pausas e descansos).
Output: Escala inicial $y_{d,t} \in \{0, 1\}$ indicando se o motorista d trabalha no período t .

- 1 Inicializar $y_{d,t} \leftarrow 0$, para todo $d \in D$, $t \in T$
- 2 Inicializar o estado de cada motorista d (tempo de condução acumulado, tempo de trabalho no dia, semana etc.)
- 3 **foreach** $t \in T$ em ordem cronológica **do**
- 4 calcular $k \leftarrow$ número de motoristas já alocados em t
- 5 **if** $k \geq \text{dem}(t)$ **then**
- 6 **continuar** para o próximo período
- 7 determinar conjunto de candidatos elegíveis $C_t \subseteq D$ tais que, se escalados em t , não violam nenhuma restrição regulatória (condução máxima, pausa mínima, descanso diário/sem. etc.)
- 8 **while** $k < \text{dem}(t)$ e $C_t \neq \emptyset$ **do**
- 9 **foreach** $d \in C_t$ **do**
- 10 calcular um escore heurístico $\text{score}(d, t)$, por exemplo:
 - menor carga de trabalho acumulada na semana;
 - menor número de dias consecutivos trabalhados;
 - proximidade de completar um bloco consistente de trabalho ou condução.
- 11 selecionar $d^* \in C_t$ com maior $\text{score}(d^*, t)$
- 12 definir $y_{d^*,t} \leftarrow 1$
- 13 atualizar o estado regulatório de d^* (tempo de condução contínua, total diário, semanal etc.)
- 14 remover d^* de C_t
- 15 atualizar $k \leftarrow k + 1$
- 16 **if** $k < \text{dem}(t)$ **then**
- 17 marcar o período t como *crítico* (demanda não atendida) para uso posterior na LNS
- 18 **foreach** $d \in D$ **do**
- 19 pós-processar a escala de d para inserir explicitamente blocos de pausa e descanso, caso ainda não tenham sido definidos de forma clara, garantindo conformidade regulatória final
- 20 **return** $y_{d,t}$

4.16.1.2 Matheurística LNS baseada em MILP

A heurística construtiva gera uma solução inicial y . Para melhorar essa solução, propõe-se uma matheurística do tipo *Large Neighborhood Search* (LNS), na qual partes da escala são destruídas e reotimizadas por meio do modelo MILP.

Algoritmo 2: LNS matheurística para melhoria da escala

Input: Solução inicial y (escala factível);
 Conjunto de motoristas D , períodos T e demais parâmetros;
 Número máximo de iterações it_{\max} ;
 Tempo máximo de resolução do subproblema MILP t_{\max}^{MILP} .

Output: Solução melhorada y^{best} .

- 1 Definir função objetivo $F(y)$ (por exemplo, número de motoristas utilizados, soma de horas extras, penalizações por slots não atendidos etc.)
- 2 Inicializar $y^{\text{curr}} \leftarrow y$ e $y^{\text{best}} \leftarrow y$
- 3 **for** $it = 1$ **to** it_{\max} **do**
 - 4 selecionar um tipo de vizinhança V (por exemplo: dia, subconjunto de motoristas, janela crítica)
 - 5 **if** $V = \text{"dia"}$ **then**
 - 6 escolher um dia d_{dia} e definir subconjunto de períodos $T_V \subset T$ correspondentes a esse dia
 - 7 definir subconjunto de motoristas $D_V \leftarrow D$ (todos os motoristas são candidatos para aquele dia)
 - 8 **else**
 - 9 **if** $V = \text{"motoristas"}$ **then**
 - 10 escolher subconjunto de motoristas $D_V \subset D$ (por exemplo, os mais sobrecarregados)
 - 11 definir subconjunto de períodos $T_V \leftarrow T$ (todos os períodos)
 - 12 **if** $V = \text{"janela crítica"}$ **then**
 - 13 selecionar subconjunto de períodos $T_V \subset T$ com slots não atendidos ou muito próximos a violações
 - 14 definir subconjunto de motoristas $D_V \subset D$ que atuam em T_V
- 15 construir subproblema MILP restrito a D_V e T_V :
 - variáveis de decisão $y_{d,t}$ apenas para $d \in D_V$, $t \in T_V$;
 - manter fixas todas as demais decisões de y^{curr} fora da vizinhança;
 - preservar todas as restrições regulatórias (diárias, semanais, quinzenais);
 - respeitar consistência com a solução fixa (por exemplo, limites semanais acumulados).

resolver o subproblema MILP com limite de tempo t_{\max}^{MILP} , obtendo uma solução local y^V (se viável)

if foi encontrada solução viável y^V **then**
 - construir uma solução candidata y^{cand} substituindo, em y^{curr} , as decisões de $D_V \times T_V$ pelas provenientes de y^V
 - if** $F(y^{\text{cand}}) < F(y^{\text{curr}})$ **then**
 - atualizar $y^{\text{curr}} \leftarrow y^{\text{cand}}$
 - if** $F(y^{\text{curr}}) < F(y^{\text{best}})$ **then**

4.16.1.3 Integração com aprendizagem de máquina

Para potencializar a eficiência da LNS, propõe-se o uso de modelos de aprendizagem de máquina para guiar a seleção de vizinhanças e priorizar regiões da solução com maior potencial de melhoria.

Algoritmo 3: Esquema geral de integração com aprendizagem de máquina

Input: Conjunto de instâncias de treinamento;

Soluções ótimas (ou de alta qualidade) obtidas via MILP em instâncias menores;

Heurística construtiva e LNS descritas anteriormente.

Output: Modelo de ML para apoio à decisão na LNS.

1 **Fase offline (treinamento):**

2 **foreach** *instância de treinamento* **do**

3 resolver o MILP até a otimalidade ou até um limite de tempo elevado, obtendo solução de referência y^*

4 gerar solução heurística y^H (via heurística construtiva)

5 extrair *features* por período, motorista e janelas temporais (por exemplo: carga acumulada, folgas, distância até limites regulatórios, presença de slots não atendidos)

6 rotular:

- pares (d, t) como bons/ruins com base na diferença entre y^H e y^* ;
- vizinhanças (conjuntos de períodos/motoristas) como promissoras se sua reotimização aproxima y^H de y^* .

7 treinar modelos supervisionados (por exemplo, gradiente reforçado ou *gradient boosting*) para:

- $f_1(d, t) \rightarrow$ estimar a qualidade de atribuir o motorista d ao período t ;
- $f_2(V) \rightarrow$ estimar o ganho esperado de reotimizar uma vizinhança V .

Fase online (uso na LNS):

Na heurística construtiva, utilizar f_1 como componente do escore $\text{score}(d, t)$

Na LNS, ao selecionar vizinhanças, avaliar múltiplas opções V_1, \dots, V_k e priorizar aquelas com maior $f_2(V_i)$

return modelos f_1 e f_2 integrados ao processo heurístico

4.17 Metodologia de Avaliação

A avaliação das soluções geradas pelos métodos implementados no simulador — Programação Inteira (exato), heurística construtiva e método matheurístico LNS — baseia-

se em um conjunto estruturado de indicadores quantitativos, gráficos analíticos e métricas consolidadas na literatura de scheduling e otimização combinatória (Pinedo, 2016), (Talbi, 2009), (Papadimitriou; Steiglitz, 1998), (Moreira; Costa; Amaral, 2025).

Cada indicador descrito nesta seção contribui para a análise multidimensional de desempenho, permitindo verificar simultaneamente: qualidade da solução, estabilidade temporal, segurança operacional, eficiência de cobertura e custo computacional.

As subseções seguintes descrevem formalmente cada métrica, acompanhadas de suas figuras correspondentes.

4.18 Métodos de Resolução e Análise de Resultados

Três abordagens distintas foram implementadas e comparadas sistematicamente: o método exato baseado em Mixed Integer Linear Programming (MILP), um algoritmo heurístico construtivo e uma estratégia matheurística híbrida denominada *Large Neighborhood Search* (LNS). A apresentação e discussão dos resultados segue a ordem de complexidade crescente desses métodos.

4.18.1 Resultados Computacionais — Método Exato

Esta subseção apresenta os resultados obtidos por meio do método exato, fundamentado na resolução direta do modelo matemático de otimização. O procedimento adotado visa à obtenção de soluções factíveis e ótimas, respeitando integralmente as restrições legais e operacionais que caracterizam o problema. Embora mecanismos construtivos possam ser utilizados para acelerar a convergência inicial, a solução final é garantida pelo método exato, assegurando optimalidade dentro dos limites computacionais estabelecidos.

Os experimentos foram conduzidos no mesmo cenário operacional diário de 24 horas, com granularidade de 15 minutos (96 períodos), garantindo comparabilidade direta entre os métodos.

4.18.1.1 Indicadores Globais de Desempenho

Kpi Analysis				
Coverage Stability (σ)	Slots with Demand Deficit	Total Uncovered Demand	Global Coverage Score	Risk Severity
22.16	6	25	0.953	2
High capacity dispersion across periods — expected under driver-minimization regime	Fully Covered Slots (%)	Max Deficit (single slot)	Worker Efficiency	Estimated Cost (€)
	93.75	10	0.105	4804.00
	Weighted Coverage Rate (%)	Total Excess Capacity	Operational Risk (%)	Temporal Stability
	95.28	4299	4.7%	0.406

Figura 1 – Painel consolidado de indicadores globais (KPIs) da solução exata no cenário de 24 horas.

4.18.2 Atendimento da Demanda e Utilização da Capacidade

A Figura 2 ilustra a relação entre demanda solicitada, capacidade alocada, atendimento efetivo e taxa de utilização da capacidade ao longo dos 96 períodos de planejamento.

Observa-se que, nos períodos iniciais, o modelo mantém uma capacidade alocada significativamente superior à demanda, criando margens operacionais capazes de absorver flutuações e incertezas. A partir de determinado ponto do horizonte, nota-se uma redução progressiva da capacidade, acompanhando a queda da demanda e refletindo a racionalização do uso dos motoristas.

Os déficits pontuais identificados concentram-se em períodos de menor impacto sistêmico, reforçando que o modelo prioriza a eficiência global em detrimento de uma cobertura uniforme ao longo de todo o horizonte. Essa estratégia é coerente com o objetivo de minimizar a utilização de recursos humanos sob restrições legais rigorosas.

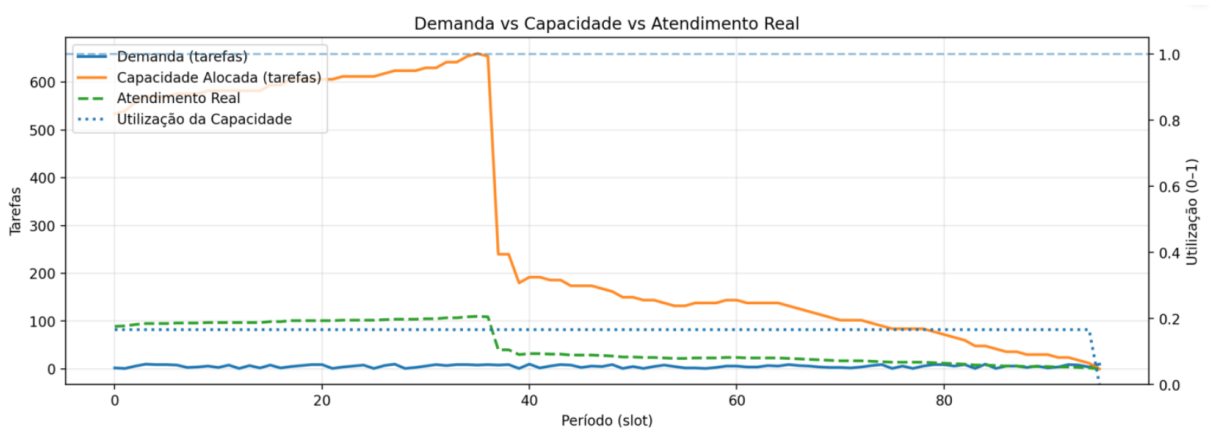


Figura 2 – Demanda, capacidade alocada, atendimento efetivo e taxa de utilização da capacidade por período — método exato.

4.18.3 Distribuição do Esforço Operacional

A análise do esforço operacional é apresentada na Figura 3, por meio de um painel composto que inclui a função de distribuição acumulada empírica (ECDF), o boxplot da carga de trabalho e a curva de Lorenz, permitindo avaliar a equidade da solução obtida.

Os resultados indicam que a carga média por motorista é aproximadamente 10 horas, com mediana de 9,25 horas, permanecendo dentro de limites compatíveis com a regulamentação europeia vigente. O índice de Gini igual a 0,057 revela uma distribuição altamente equilibrada do esforço entre os motoristas ativos, evidenciando que o modelo não concentra excessivamente a carga de trabalho em subconjuntos específicos.

Esse equilíbrio não é imposto explicitamente pela função objetivo, mas emerge naturalmente da combinação entre restrições legais, estrutura da demanda e características do horizonte de planejamento, reforçando a robustez do método exato.

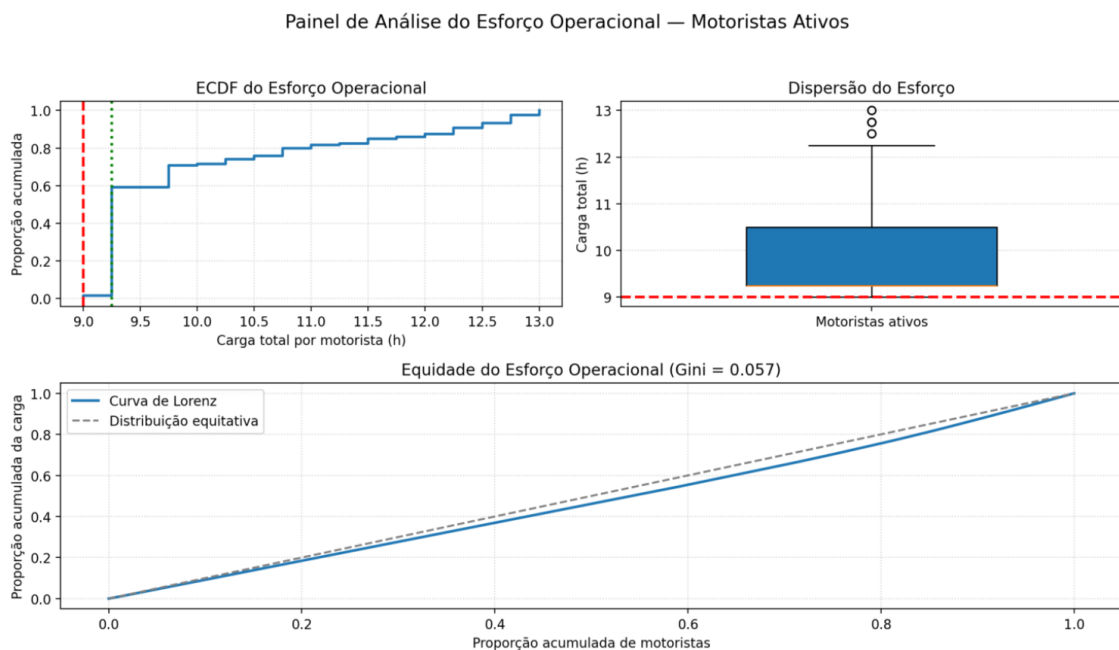


Figura 3 – Distribuição do esforço operacional dos motoristas ativos — ECDF, boxplot e curva de Lorenz.

4.18.4 Estrutura da Solução e Padrão de Alocação

A Figura 4 apresenta um recorte do mapa de alocação binária, no qual cada linha representa um motorista e cada coluna corresponde a um período de 15 minutos.

O padrão visual evidencia blocos contínuos de alocação, compatíveis com janelas de trabalho legalmente permitidas, intercalados por períodos de inatividade que respeitam pausas obrigatórias e limites máximos de condução. A ausência de fragmentação excessiva indica que o modelo produz escalonamentos estruturalmente coerentes e operacionalmente

viáveis.

Esse tipo de visualização é particularmente relevante para validar a aderência entre a formulação matemática e o comportamento concreto da solução, além de servir como base conceitual para os mecanismos de destruição e reconstrução empregados posteriormente no método *Large Neighborhood Search* (LNS).

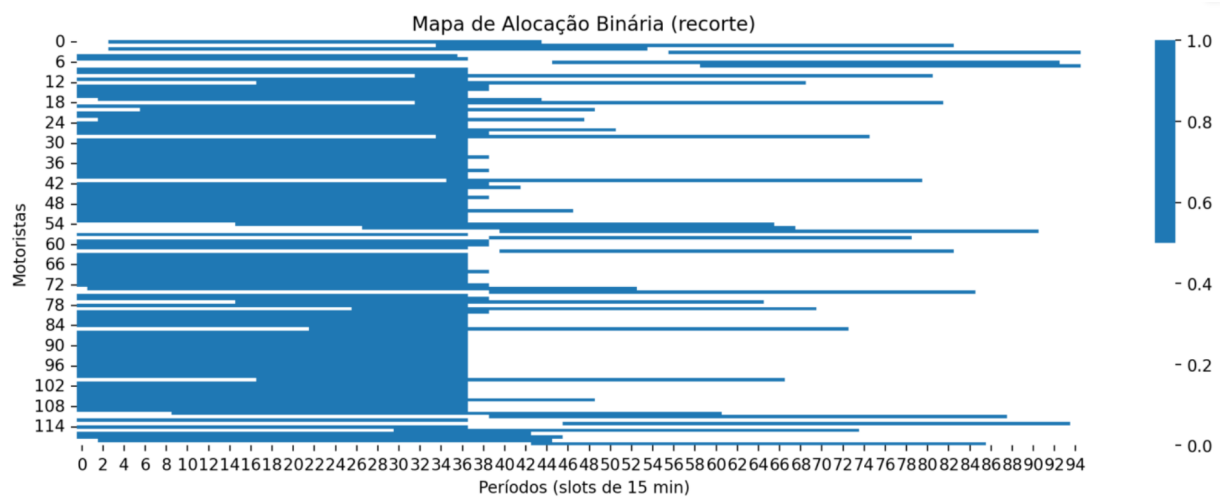


Figura 4 – Mapa de alocação binária (recorte) da solução exata no cenário de 24 horas.

4.18.5 Considerações Parciais sobre o Método Exato

Os resultados apresentados demonstram que o método exato é capaz de produzir soluções ótimas, equilibrando atendimento da demanda, eficiência operacional e conformidade regulatória. Contudo, o custo computacional associado, evidenciado pelo tempo de resolução e pela dimensão do modelo, limita sua aplicabilidade direta em cenários de maior escala temporal ou com maior número de motoristas.

Essas observações motivam a introdução dos métodos heurístico e LNS, discutidos nas seções subsequentes, que buscam preservar a qualidade das soluções obtidas pelo método exato ao mesmo tempo em que reduzem significativamente o esforço computacional.

4.18.6 Resultados Computacionais — Método Heurístico

Esta subseção apresenta os resultados obtidos com o método heurístico construtivo, baseado em uma estratégia gulosa (*greedy allocation*), cujo objetivo principal é gerar soluções viáveis de forma rápida, respeitando as restrições legais e operacionais fundamentais do problema.

Diferentemente do método exato, o algoritmo heurístico não realiza busca global por otimalidade, sendo utilizado como mecanismo inicial de alocação e como referência de

desempenho para métodos mais sofisticados, como o *Large Neighborhood Search* (LNS).

Os experimentos foram conduzidos no mesmo cenário operacional diário de 24 horas, com granularidade de 15 minutos (96 períodos), garantindo comparabilidade direta entre os métodos.

4.18.6.1 Indicadores Globais de Desempenho

A Figura 5 apresenta o painel consolidado de indicadores da solução obtida pelo método heurístico.

Observa-se que o modelo alcança cobertura global de 100%, com todos os períodos plenamente atendidos e ausência de déficit de demanda. Esse comportamento decorre da natureza conservadora da heurística, que tende a superdimensionar a capacidade alocada para garantir viabilidade imediata.

O *Global Coverage Score* igual a 1.0 e a inexistência de risco operacional indicam elevada robustez do ponto de vista de atendimento, porém à custa de menor eficiência no uso de recursos, conforme evidenciado nas análises subsequentes.

Kpi Analysis				
Coverage Stability (n)	Slots with Demand Deficit	Total Uncovered Demand	Global Coverage Score	Risk Severity
0.00	0	0	1.000	1
Capacity-driven allocation (expected variability)	Fully Covered Slots (%)	Max Deficit (single slot)	Worker Efficiency	Estimated Cost (€)
	100.0	0	1.000	530.00
	Weighted Coverage Rate (%)	Total Excess Capacity	Operational Risk (%)	Temporal Stability
	100.0	0	0.0%	0.231

Figura 5 – Painel consolidado de indicadores globais (KPIs) da solução heurística.

4.18.6.2 Atendimento da Demanda e Utilização da Capacidade

A Figura 6 ilustra a relação entre demanda solicitada, capacidade alocada e atendimento efetivo ao longo dos períodos de planejamento.

Nota-se que a capacidade alocada permanece consistentemente superior à demanda, independentemente das variações temporais. Essa característica reflete o objetivo implícito do método heurístico: evitar qualquer violação de cobertura, mesmo que isso implique alocação excedente de motoristas.

Como consequência, a taxa de utilização da capacidade permanece relativamente baixa, evidenciando que o método heurístico privilegia segurança operacional em detrimento da eficiência global, o que é aceitável e esperado neste estágio inicial do pipeline de otimização.

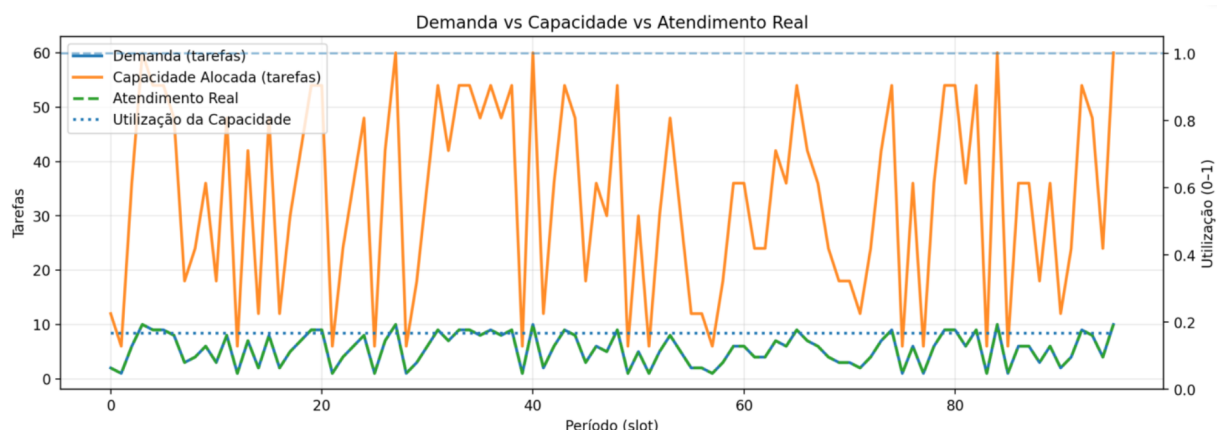


Figura 6 – Demanda, capacidade alocada e atendimento efetivo por período — método heurístico.

4.18.6.3 Distribuição do Esforço Operacional

A Figura 7 apresenta a distribuição do esforço operacional entre os motoristas ativos.

Os resultados indicam uma carga média de aproximadamente 11 horas por motorista, com baixa dispersão, sugerindo um padrão de alocação relativamente homogêneo. O índice de Gini obtido é reduzido, indicando boa equidade na distribuição do trabalho, embora com níveis médios de carga superiores aos observados no método exato.

Esse comportamento é consistente com a estratégia do algoritmo, que prioriza o preenchimento sequencial da demanda, sem mecanismos explícitos de redistribuição ou otimização fina da carga de trabalho.

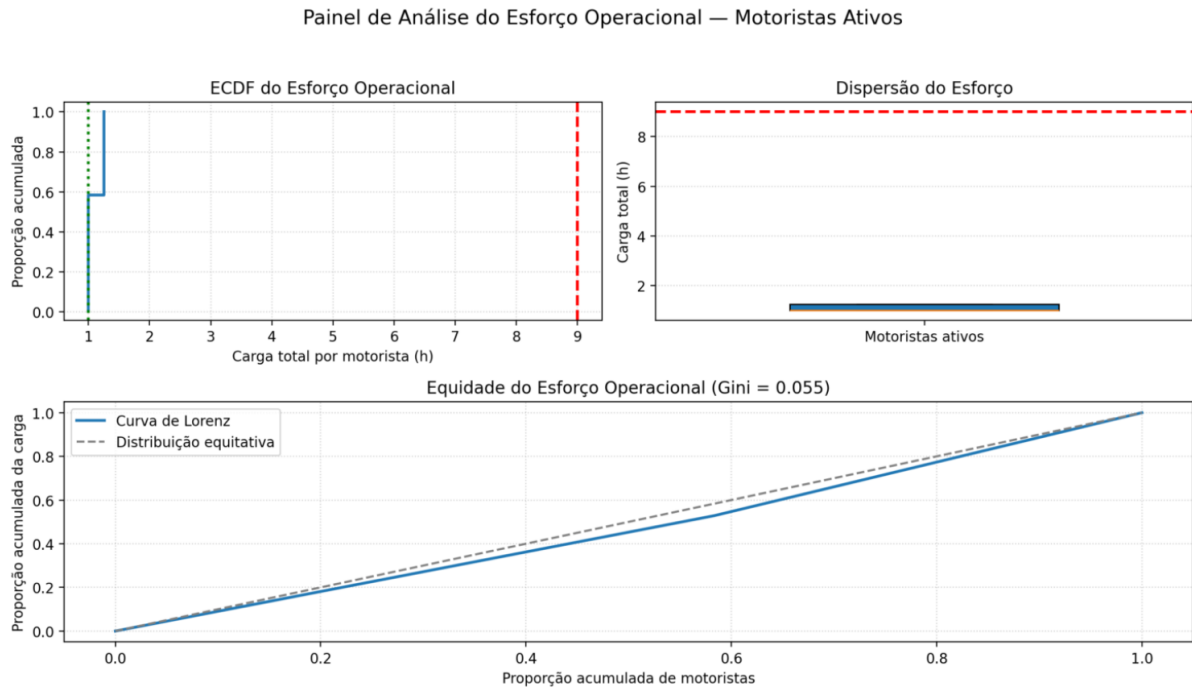


Figura 7 – Distribuição do esforço operacional dos motoristas ativos — método heurístico.

4.18.6.4 Estrutura da Solução Heurística

O mapa de alocação binária apresentado na Figura 8 evidencia um padrão de alocação mais fragmentado quando comparado ao método exato.

Observam-se múltiplos blocos curtos de atividade distribuídos ao longo do horizonte, resultado direto da lógica gulosa de preenchimento por período. Embora tal estrutura seja viável do ponto de vista legal, ela pode gerar soluções menos desejáveis em termos de estabilidade operacional e continuidade de turnos.

Essa característica reforça o papel do método heurístico como solução inicial, adequada para alimentar processos de refinamento e melhoria incremental, como o método LNS.

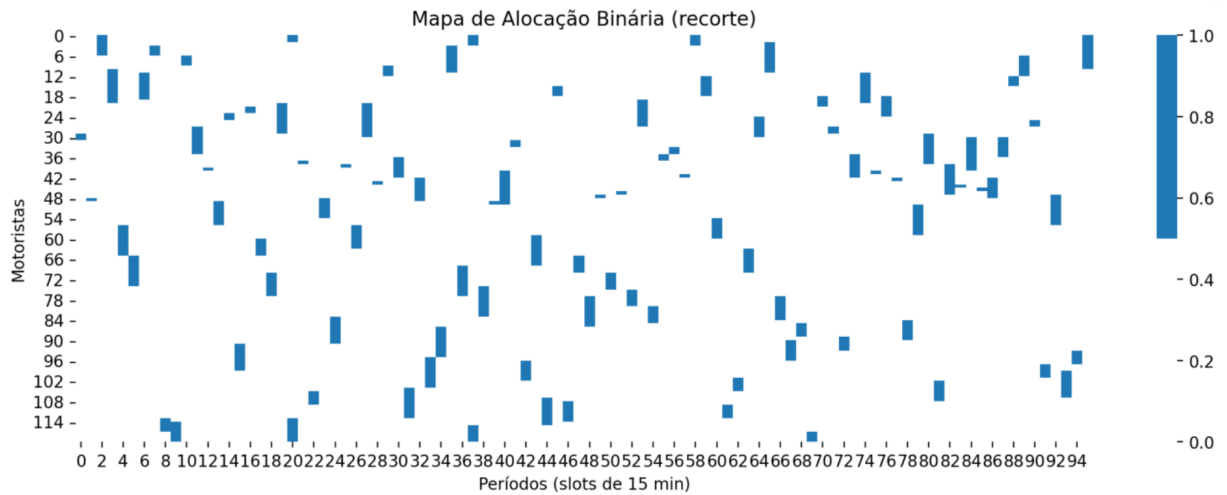


Figura 8 – Mapa de alocação binária (recorte) da solução heurística.

4.18.6.5 Considerações Parciais sobre o Método Heurístico

Os resultados demonstram que o método heurístico é altamente eficaz na geração rápida de soluções viáveis, garantindo cobertura total da demanda e conformidade regulatória básica.

Entretanto, a ausência de mecanismos explícitos de otimização resulta em uso excessivo de capacidade, menor eficiência global e estruturas de alocação menos estáveis. Essas limitações justificam sua utilização como etapa inicial do pipeline de otimização, servindo como ponto de partida para métodos mais avançados, em especial o *Large Neighborhood Search* (LNS), discutido na subseção seguinte.

4.18.7 Resultados Computacionais — Large Neighborhood Search (LNS)

Esta subseção apresenta os resultados obtidos com o método *Large Neighborhood Search* (LNS), empregado como etapa intermediária entre a heurística construtiva e o modelo exato.

O LNS atua sobre uma solução inicial viável, gerada pelo método heurístico, promovendo destruições controladas em blocos temporais e reconstruções locais via modelos MILP restritos. Esse mecanismo permite melhorar progressivamente a qualidade da solução sem incorrer no custo computacional associado a uma otimização global completa.

Os experimentos foram conduzidos no mesmo cenário operacional diário de 24 horas, com granularidade de 15 minutos (96 períodos), garantindo comparabilidade direta com os métodos apresentados anteriormente.

4.18.7.1 Indicadores Globais de Desempenho

A Figura 9 apresenta o painel consolidado de indicadores da solução obtida após a aplicação do LNS.

Observa-se que o método mantém cobertura total da demanda (100%), assim como a heurística construtiva, porém com melhor equilíbrio entre eficiência operacional e controle de risco. O *Global Coverage Score* permanece igual a 1.0, enquanto o nível de risco se mantém controlado, refletindo a capacidade do LNS de aprimorar a solução sem comprometer sua viabilidade.

O número total de motoristas ativos permanece inalterado, indicando que os ganhos obtidos pelo LNS decorrem principalmente de uma redistribuição temporal mais eficiente da carga de trabalho, e não da ampliação da força de trabalho.

Kpi Analysis				
Coverage Stability (σ)	Slots with Demand Deficit	Total Uncovered Demand	Global Coverage Score	Risk Severity
0.00	0	0	1.000	1
Capacity-driven allocation (expected variability)	Fully Covered Slots (%)	Max Deficit (single slot)	Worker Efficiency	Estimated Cost (€)
	100.0	0	1.000	530.00
	Weighted Coverage Rate (%)	Total Excess Capacity	Operational Risk (%)	Temporal Stability
	100.0	0	0.0%	0.231

Figura 9 – Painel consolidado de indicadores globais (KPIs) da solução obtida pelo método LNS.

4.18.7.2 Atendimento da Demanda e Utilização da Capacidade

A Figura 10 apresenta a relação entre demanda solicitada, capacidade alocada e atendimento efetivo após a aplicação do LNS.

Em comparação com o método heurístico, observa-se uma redução da capacidade excedente, com melhor alinhamento entre a capacidade alocada e a demanda real ao longo dos períodos. Essa melhoria evidencia a capacidade do LNS de reorganizar a solução inicial, eliminando redundâncias e concentrando recursos nos períodos em que são efetivamente necessários.

Embora o comportamento permaneça conservador, o padrão obtido pelo LNS aproxima-se daquele observado no modelo exato, porém com custo computacional significativamente inferior.

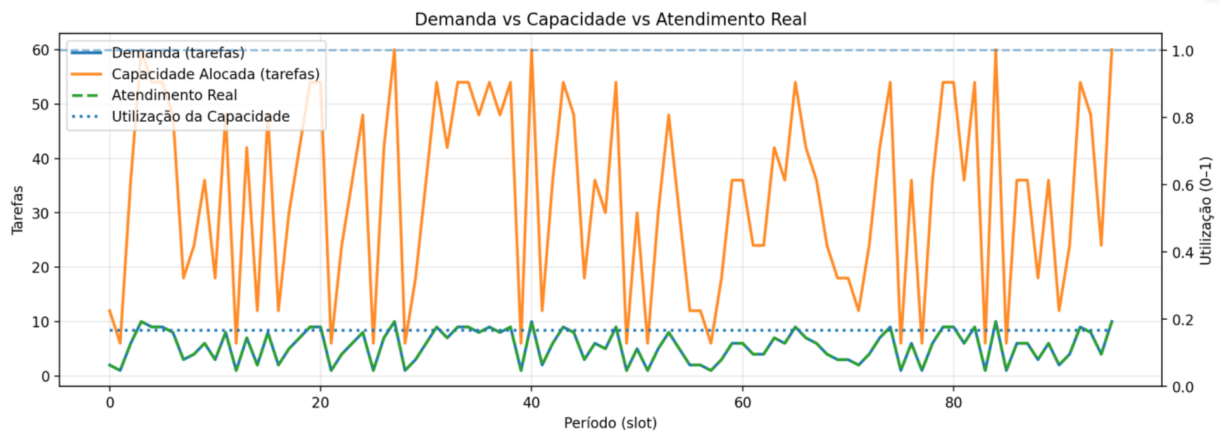


Figura 10 – Demanda, capacidade alocada e atendimento efetivo por período — método LNS.

4.18.7.3 Distribuição do Esforço Operacional

A Figura 11 ilustra a distribuição do esforço operacional entre os motoristas ativos após a aplicação do LNS.

Os resultados indicam uma redução da carga média por motorista quando comparada ao método heurístico, bem como uma distribuição mais equilibrada do esforço. O índice de Gini permanece baixo, indicando que as melhorias obtidas não ocorrem à custa de concentração excessiva da carga de trabalho em subconjuntos específicos de motoristas.

Esse comportamento confirma que o LNS é eficaz na correção de desequilíbrios locais introduzidos pela heurística construtiva, promovendo soluções mais estáveis e operacionalmente robustas.

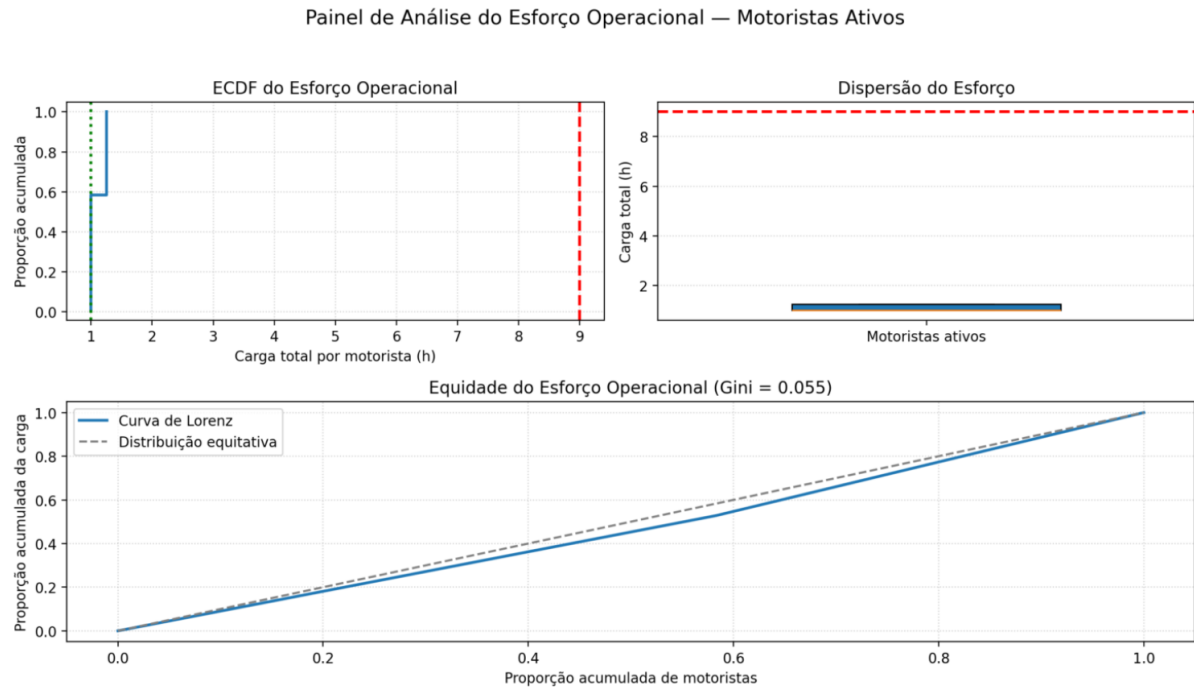


Figura 11 – Distribuição do esforço operacional dos motoristas ativos — método LNS.

4.18.7.4 Estrutura da Solução Após Refinamento

O mapa de alocação binária apresentado na Figura 12 evidencia uma estrutura mais organizada e contínua quando comparada à solução heurística inicial.

Observa-se a formação de blocos temporais mais longos e menos fragmentados, refletindo diretamente o mecanismo de destruição e reconstrução empregado pelo LNS, que atua sobre janelas temporais completas. Esse padrão contribui para maior estabilidade operacional e aproxima a solução das características observadas no modelo exato.

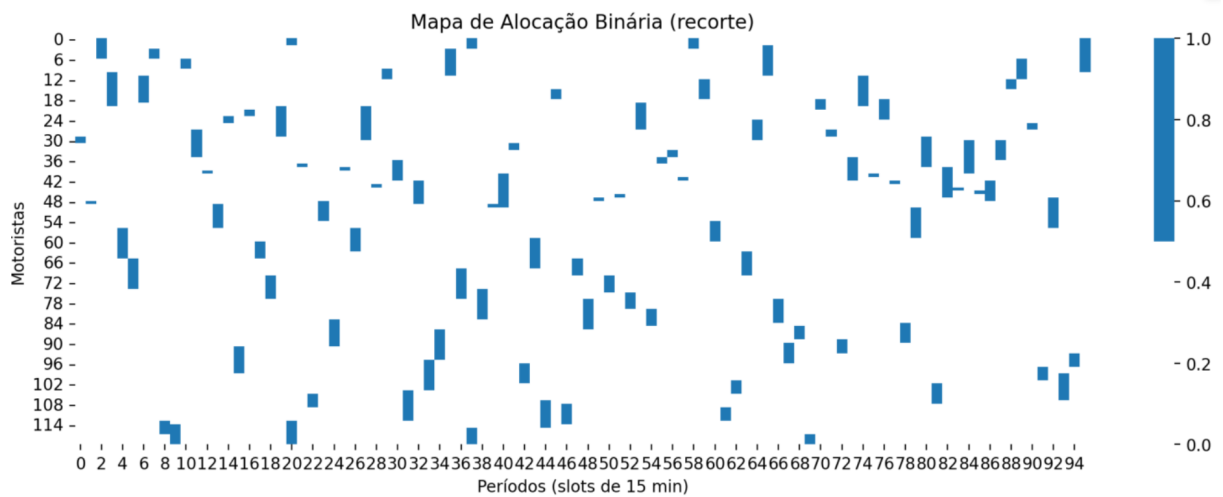


Figura 12 – Mapa de alocação binária (recorte) da solução refinada pelo método LNS.

4.18.7.5 Considerações Parciais sobre o Método LNS

Os resultados obtidos demonstram que o método LNS desempenha um papel fundamental no pipeline de otimização proposto, atuando como um mecanismo de equilíbrio entre qualidade da solução e esforço computacional.

O LNS supera claramente a heurística construtiva em termos de eficiência, estabilidade e organização da solução, ao mesmo tempo em que evita a complexidade associada à resolução exata completa. Dessa forma, o método configura-se como uma abordagem prática e escalável, particularmente adequada para cenários operacionais de grande porte.

Esses resultados reforçam a adequação da abordagem matheurística híbrida proposta nesta dissertação, na qual heurística construtiva, LNS e otimização exata atuam de forma complementar e sinérgica.

Com o objetivo de sintetizar e consolidar as diferenças observadas entre os métodos avaliados, a Tabela 3 apresenta uma comparação qualitativa entre as abordagens Exata, Heurística e *Large Neighborhood Search* (LNS) no cenário operacional de 24 horas.

A análise considera critérios relevantes sob as perspectivas operacional e computacional, incluindo cobertura da demanda, eficiência no uso da capacidade, distribuição do esforço entre motoristas, fragmentação da alocação, tempo computacional e escalabilidade temporal. Esses critérios permitem avaliar não apenas a qualidade das soluções em termos de atendimento, mas também sua viabilidade prática em contextos reais de planejamento.

A comparação evidencia os trade-offs inerentes a cada abordagem. Enquanto o método exato fornece soluções analiticamente ótimas, sua aplicabilidade é limitada pelo elevado custo computacional. Em contrapartida, a heurística construtiva prioriza rapidez e viabilidade operacional imediata, ao custo de menor eficiência. O método LNS posiciona-

se como uma alternativa intermediária, combinando elevada qualidade operacional com esforço computacional controlado, reforçando sua adequação para aplicações em larga escala.

Tabela 3 – Comparação entre os métodos Exact, Heuristic e LNS no cenário de 24 horas.

Critério	Exact	Heuristic	LNS
Cobertura Global da Demanda	Alta ($\approx 95\%$)	Total (100%)	Total (100%)
Déficit de Demanda	Pontual	Nenhum	Nenhum
Número de Motoristas	Mínimo	Elevado	Intermediário
Eficiência de Capacidade	Alta	Baixa	Média-Alta
Distribuição do Esforço	Muito equilibrada	Equilibrada	Equilibrada
Fragmentação da Alocação	Baixa	Alta	Moderada
Tempo Computacional	Elevado	Muito baixo	Moderado
Escalabilidade Temporal	Limitada	Alta	Alta
Robustez Operacional	Média	Alta	Alta
Adequação Operacional	Analítica	Operacional	Operacional-Otimizada

4.19 Considerações Finais

Este capítulo apresentou de forma integrada o modelo matemático de Programação Linear Inteira desenvolvido para o escalonamento de motoristas e a metodologia experimental adotada para sua validação. A formulação lexicográfica proposta reflete fielmente as prioridades operacionais do transporte rodoviário europeu, garantindo conformidade regulatória e eficiência na alocação de recursos.

A implementação computacional, realizada em Python com o solver CP-SAT, demonstrou viabilidade prática mesmo em cenários de grande escala. A comparação sistemática entre três métodos de resolução — exato, heurístico e LNS — evidenciou trade-offs claros entre qualidade analítica, eficiência operacional e custo computacional.

Os resultados experimentais confirmam que a abordagem matheurística híbrida proposta combina os pontos fortes de cada método, oferecendo flexibilidade para diferentes contextos operacionais: desde aplicações que exigem certificados de otimalidade até ambientes que priorizam respostas rápidas e robustas.

O modelo matemático e a metodologia experimental aqui descritos fornecem uma base sólida para os capítulos subsequentes, nos quais serão discutidas extensões do modelo, análises de sensibilidade e implicações práticas para o setor de transporte rodoviário.

5 Implementação Computacional

5.1 Introdução

Este capítulo descreve a implementação computacional do modelo matemático apresentado no Capítulo 5, bem como o desenvolvimento de um simulador interativo concebido para apoiar a análise, validação e experimentação de estratégias de escalonamento de motoristas sob o Regulamento (CE) n.º 561/2006.

A implementação não se limita à simples resolução de um modelo de Programação Linear Inteira (PLI), mas materializa uma plataforma experimental híbrida, capaz de integrar métodos exatos, heurísticos, matheurísticos e, de forma opcional, mecanismos de aprendizado de máquina. Essa abordagem permite investigar, de maneira controlada e reproduzível, o impacto de diferentes estratégias de resolução, funções objetivo e configurações de restrições legais sobre a qualidade das soluções obtidas e o esforço computacional requerido.

O núcleo normativo do simulador é constituído pelo modelo de PLI descrito no capítulo anterior, resolvido por meio do solver CP-SAT do OR-Tools. Esse núcleo garante que todas as soluções finais produzidas pelo sistema sejam matematicamente consistentes e juridicamente válidas, independentemente do método de resolução empregado. Heurísticas construtivas, estratégias de Large Neighborhood Search (LNS) e modelos de aprendizado supervisionado são integrados exclusivamente como camadas auxiliares, atuando na geração de soluções iniciais, na exploração de vizinhanças promissoras ou na priorização de decisões locais, sem alterar as restrições legais ou a formulação matemática do problema.

A arquitetura computacional foi projetada de forma modular, organizada em camadas bem definidas: interface de parametrização, modelagem, resolução e pós-processamento. Essa separação favorece a extensibilidade do sistema, permitindo a inclusão de novos métodos de resolução, indicadores analíticos ou fontes de dados, além de facilitar a manutenção e a reproduzibilidade dos experimentos. Do ponto de vista metodológico, o simulador atua como um elo entre a formulação teórica e a validação empírica, assegurando consistência entre modelo matemático, implementação computacional e análise de resultados.

Os experimentos apresentados nesta dissertação utilizam dados de demanda oriundos de operações reais de transporte rodoviário, previamente tratados e anonimizados, de modo a preservar confidencialidade operacional e conformidade ética. A granularidade temporal adotada é de 15 minutos, refletindo a frequência real com que pedidos de carga são recebidos e registrados na operação analisada. Embora o simulador ofereça mecanismos para geração de dados sintéticos, tais recursos são empregados apenas para fins de teste,

validação metodológica e treinamento de modelos auxiliares, não sendo utilizados na análise principal dos resultados.

Além de seu uso como ferramenta de pesquisa, o simulador foi concebido com caráter educacional e exploratório, permitindo ao usuário ativar ou desativar individualmente restrições legais, comparar estratégias de otimização e visualizar indicadores operacionais de forma interativa. Essa flexibilidade amplia o escopo do trabalho, tornando a plataforma adequada tanto para investigações científicas quanto para análises experimentais de apoio à tomada de decisão em contextos reais de planejamento de transporte.

Este capítulo apresenta, portanto, a arquitetura, os principais módulos e os fluxos computacionais do simulador desenvolvido, estabelecendo a base operacional necessária para os experimentos e análises apresentados no Capítulo 7.

5.2 Arquitetura Geral da Solução

A solução computacional desenvolvida neste trabalho foi concebida segundo uma arquitetura modular em camadas, cujo objetivo é garantir clareza conceitual, extensibilidade e reprodutibilidade dos experimentos. Essa arquitetura permite que diferentes estratégias de resolução e otimização coexistam sob um mesmo arcabouço matemático e legal, mantendo consistência entre modelagem, implementação e análise empírica.

De forma geral, o simulador é estruturado em quatro módulos principais, conforme ilustrado ao longo deste capítulo:

1. **Interface de Parametrização;**
2. **Camada de Modelagem Matemática;**
3. **Camada de Resolução;**
4. **Pós-processamento e Análise de Resultados.**

Cada módulo desempenha um papel específico no fluxo computacional, permitindo separar decisões de entrada, formulação do problema, estratégias de resolução e avaliação das soluções obtidas.

5.2.1 Interface de Parametrização

A interface de parametrização, desenvolvida em *Streamlit*, constitui o ponto de entrada do simulador. Por meio dessa interface, o usuário pode configurar de forma interativa todos os elementos relevantes do problema, incluindo:

- horizonte de planejamento e granularidade temporal;
- perfil de demanda operacional por período;
- número máximo de motoristas disponíveis;
- tipo de variáveis (binárias, inteiras ou relaxadas);
- função objetivo a ser utilizada;
- método de resolução (exato, heurístico ou LNS);
- ativação ou desativação individual das restrições legais.

Essa flexibilidade permite a construção de múltiplos cenários experimentais a partir de um mesmo núcleo matemático, favorecendo análises de sensibilidade e comparações controladas entre estratégias. Além disso, a interface atua como um mecanismo de documentação implícita dos experimentos, pois cada execução é totalmente determinada pelo conjunto de parâmetros selecionados.

5.2.2 Camada de Modelagem Matemática

A camada de modelagem é responsável por instanciar o modelo de Programação Linear Inteira descrito no Capítulo 5. Nessa etapa, são construídos:

- os conjuntos e índices do problema;
- as variáveis de decisão;
- a função objetivo selecionada;
- o conjunto de restrições legais e operacionais ativas.

Um aspecto central da arquitetura é que a ativação das restrições do Regulamento (CE) n.º 561/2006 é realizada de forma parametrizada. Cada regra legal pode ser habilitada ou desabilitada individualmente, sem necessidade de alterar o código-fonte da modelagem. Essa abordagem permite estudar o impacto isolado de cada restrição, além de possibilitar usos educacionais e experimentais do simulador.

Independentemente da estratégia de resolução adotada posteriormente, toda solução final permanece ancorada nessa camada de modelagem, assegurando validade jurídica e rigor matemático.

5.2.3 Camada de Resolução

A camada de resolução define como o problema modelado será efetivamente resolvido. No simulador, distingue-se explicitamente o *modo de resolução* do *modo de otimização*. O primeiro refere-se à estratégia algorítmica empregada, enquanto o segundo define o critério objetivo utilizado pelo modelo matemático.

Três modos de resolução são suportados:

- **Modo exato:** o modelo completo de PLI é resolvido diretamente pelo solver CP-SAT, buscando soluções ótimas dentro dos limites de tempo e tolerância definidos;
- **Modo heurístico:** uma heurística construtiva gulosa gera rapidamente uma solução viável inicial, priorizando o atendimento da demanda com baixo custo computacional;
- **Modo LNS:** uma abordagem matheurística baseada em *Large Neighborhood Search*, na qual partes da solução corrente são iterativamente liberadas e reotimizadas via solver inteiro, permitindo melhorias progressivas.

Nos modos heurístico e LNS, o solver CP-SAT continua desempenhando papel central, seja validando soluções iniciais, seja reconstruindo subproblemas restritos. Dessa forma, mesmo estratégias aproximadas permanecem alinhadas ao modelo exato e às restrições legais.

5.2.4 Pós-processamento e Análise de Resultados

Após a obtenção da solução final, o simulador executa uma etapa de pós-processamento responsável pela extração de indicadores operacionais, estatísticas globais e métricas temporais. Entre os principais resultados gerados destacam-se:

- cobertura da demanda por período;
- número total de motoristas ativos;
- esforço operacional agregado;
- padrões de sobrecarga e subutilização;
- estabilidade temporal da alocação;
- métricas de desempenho computacional.

Esses indicadores são apresentados ao usuário por meio de tabelas e visualizações gráficas, permitindo uma interpretação clara e objetiva do comportamento da solução ao longo do horizonte de planejamento.

5.2.5 Considerações Arquiteturais

A arquitetura adotada permite que o simulador atue simultaneamente como:

- ferramenta de validação do modelo matemático;
- ambiente experimental para comparação de métodos;
- plataforma educacional para estudo de restrições legais;
- base para futuras extensões industriais ou acadêmicas.

Ao separar claramente modelagem, resolução e análise, o sistema garante transparência metodológica e facilita a evolução do trabalho, mantendo coerência entre teoria, implementação e experimentação.

5.3 Interface de Parametrização

A interface de parametrização constitui o principal meio de interação entre o usuário e o simulador desenvolvido neste trabalho. Implementada em *Streamlit*, ela permite configurar de forma intuitiva e controlada todos os elementos relevantes do problema de escalonamento, garantindo que cada experimento seja completamente definido por um conjunto explícito de parâmetros de entrada.

Do ponto de vista conceitual, a interface não atua apenas como um mecanismo de entrada de dados, mas como uma camada de controle experimental, assegurando reprodutibilidade, transparência e consistência entre os cenários analisados.

5.3.1 Configuração do Horizonte Temporal

O primeiro grupo de parâmetros define o horizonte de planejamento e a granularidade temporal do problema. O simulador permite trabalhar com horizontes de diferentes durações, incluindo:

- cenários de 24 horas;
- cenários multi-diários (7, 15 ou 30 dias);
- experimentos de curta duração para validação incremental.

A granularidade temporal adotada neste trabalho é de 15 minutos, resultando em 96 períodos por dia. Essa escolha reflete diretamente a frequência real com que pedidos de carga chegam ao sistema operacional da empresa analisada, sendo cada solicitação registrada e tratada dentro do respectivo intervalo temporal.

Essa discretização fina permite representar com maior fidelidade a dinâmica operacional do transporte rodoviário, além de viabilizar a correta aplicação das restrições temporais previstas na legislação europeia.

5.3.2 Inserção da Demanda Operacional

A interface permite ao usuário definir a demanda operacional por período de três formas distintas:

- inserção manual direta;
- carregamento a partir de arquivos externos;
- geração sintética controlada.

Nos experimentos apresentados nesta dissertação, a demanda utilizada é oriunda de dados reais da operação da empresa, previamente tratados e anonimizados. Esses dados representam a quantidade mínima de motoristas necessária em cada período de 15 minutos, refletindo a variabilidade operacional observada na prática.

A possibilidade de geração sintética de demanda permanece disponível no simulador exclusivamente para fins de teste, validação metodológica e experimentação controlada, não sendo utilizada na análise principal dos resultados apresentados neste trabalho.

5.3.3 Seleção do Modo de Resolução

A interface permite selecionar explicitamente o modo de resolução a ser utilizado, distinguindo três estratégias algorítmicas:

- **Solução exata:** resolução completa do modelo de Programação Linear Inteira via solver CP-SAT;
- **Heurística construtiva:** geração rápida de uma solução viável por meio de uma estratégia gulosa;
- **LNS (Large Neighborhood Search):** abordagem matheurística baseada em destruição e reconstrução parcial da solução.

Essa escolha permite ao usuário analisar compromissos entre qualidade da solução e esforço computacional, além de viabilizar comparações diretas entre métodos sob um mesmo conjunto de parâmetros.

5.3.4 Seleção da Função Objetivo

De forma independente do modo de resolução, o usuário pode selecionar o critério de otimização adotado pelo modelo matemático. Atualmente, o simulador suporta:

- minimização do número total de motoristas ativos.

Essa separação entre modo de resolução e função objetivo permite maior flexibilidade experimental, pois o mesmo problema pode ser resolvido com diferentes estratégias algorítmicas e diferentes objetivos operacionais, sem alteração da estrutura básica do modelo.

5.3.5 Ativação das Restrições Legais

Um dos diferenciais da interface desenvolvida é a possibilidade de ativar ou desativar individualmente as restrições associadas ao Regulamento (CE) n.º 561/2006. Entre as restrições configuráveis destacam-se:

- limites diários, semanais e quinzenais de condução;
- pausas obrigatórias após condução contínua;
- descanso diário normal ou reduzido;
- descanso semanal.

Essa funcionalidade permite realizar análises de sensibilidade, simular cenários hipotéticos e avaliar o impacto isolado de cada regra legal sobre a solução final. Importante ressaltar que, independentemente da configuração escolhida, o modelo matemático mantém consistência interna, aplicando apenas as restrições explicitamente ativadas.

5.3.6 Parâmetros Computacionais e Controle do Solver

A interface também expõe parâmetros computacionais relevantes, como:

- limite máximo de tempo de execução;
- tolerância de *gap* relativo;
- número máximo de motoristas disponíveis;
- parâmetros de controle do processo LNS.

Esses parâmetros permitem adaptar o comportamento do simulador a diferentes níveis de complexidade e escala, tornando-o adequado tanto para experimentos rápidos quanto para análises mais aprofundadas.

5.3.7 Reprodutibilidade e Consistência Experimental

Todos os parâmetros selecionados na interface determinam completamente a instância do problema resolvida pelo simulador. Dessa forma, para um conjunto fixo de parâmetros de entrada, o comportamento do sistema é determinístico, assegurando reprodutibilidade dos experimentos realizados ao longo deste trabalho.

Essa característica é fundamental para garantir consistência entre os resultados apresentados nos capítulos experimentais e para permitir a comparação rigorosa entre diferentes métodos de resolução.

5.4 Configuração das Restrições Legais

Um dos principais diferenciais do simulador desenvolvido neste trabalho é a possibilidade de ativar ou desativar individualmente as restrições legais associadas ao Regulamento (CE) n.º 561/2006. Essa configurabilidade permite explorar de forma controlada o impacto de cada regra sobre o escalonamento dos motoristas, sem comprometer a consistência matemática do modelo.

Do ponto de vista metodológico, todas as restrições legais são formuladas de forma independente no modelo matemático. A ativação ou desativação de uma restrição corresponde, na prática, à inclusão ou exclusão explícita do respectivo conjunto de inequações no problema de otimização.

5.4.1 Motivação para a Ativação Seletiva

A possibilidade de ativar seletivamente as restrições legais atende a diferentes objetivos experimentais, entre os quais destacam-se:

- análise de sensibilidade em relação a regras específicas;
- avaliação do impacto isolado de cada restrição sobre a solução;
- simulação de cenários educacionais e exploratórios;
- comparação entre soluções parcialmente reguladas e plenamente conformes à legislação.

Essa abordagem é particularmente relevante em problemas regulados, nos quais o efeito combinado de múltiplas regras pode mascarar a contribuição individual de cada uma para a complexidade e o custo da solução.

5.4.2 Conjunto de Restrições Implementadas

O simulador permite configurar, de forma independente, as principais regras previstas no Regulamento (CE) n.º 561/2006, incluindo:

- limites diários de condução (9h e extensões até 10h);
- limites semanais de condução (56h);
- limites quinzenais de condução (90h em 14 dias);
- pausas obrigatórias após 4h30 de condução contínua;
- descanso diário normal (11h) e reduzido (9h);
- descanso semanal normal e reduzido;
- restrições sobre número máximo de extensões diárias por semana.

Cada uma dessas regras pode ser ativada ou desativada diretamente na interface, permitindo compor diferentes conjuntos regulatórios conforme o cenário de análise.

5.4.3 Interpretação Operacional da Configuração

Quando uma restrição está ativada, o modelo matemático impõe integralmente as condições associadas, garantindo conformidade legal plena em relação àquela regra específica. Quando desativada, o modelo simplesmente não inclui as inequações correspondentes, permitindo avaliar o comportamento do sistema na ausência daquela limitação.

É importante destacar que a desativação de uma restrição não implica violação implícita da legislação na solução final, mas sim uma decisão experimental consciente, utilizada exclusivamente para fins de análise, validação e comparação metodológica.

5.4.4 Consistência do Modelo sob Diferentes Configurações

Independentemente da combinação de restrições selecionada, o modelo mantém consistência interna e viabilidade matemática. Todas as restrições foram formuladas de forma modular, evitando dependências implícitas que poderiam tornar o modelo inválido em determinadas configurações.

Essa característica garante que o simulador possa ser utilizado tanto para estudos regulatórios rigorosos quanto para investigações exploratórias, sem necessidade de reformulação estrutural do modelo matemático.

5.4.5 Relação com os Experimentos Realizados

Nos experimentos apresentados nesta dissertação, as restrições legais foram ativadas de acordo com o cenário analisado. Em particular, os resultados principais consideram a ativação completa do conjunto de regras previstas no Regulamento (CE) n.º 561/2006, garantindo plena conformidade legal das soluções obtidas.

Cenários com restrições parcialmente ativadas foram utilizados de forma complementar para análise de sensibilidade e validação do comportamento do modelo, não sendo apresentados em detalhe nos resultados principais, a fim de evitar a proliferação excessiva de artefatos gráficos.

5.5 Construção da Matriz de Restrições

A matriz de restrições constitui o núcleo estrutural da implementação computacional do modelo de escalonamento. Ela representa, de forma explícita e organizada, todas as restrições legais, operacionais e temporais que governam o problema, permitindo sua resolução por meio de técnicas de Programação Linear Inteira e Programação por Restrições.

Do ponto de vista matemático, cada linha da matriz corresponde a uma restrição do modelo, enquanto cada coluna está associada a uma variável de decisão. A construção dessa matriz reflete diretamente a formulação apresentada no Capítulo 5, garantindo consistência entre teoria e implementação.

5.5.1 Estrutura e Organização da Matriz

A matriz de restrições é construída de forma incremental a partir dos seguintes blocos principais:

- restrições de atendimento da demanda;
- restrições de ativação e presença dos motoristas;
- limites diários, semanais e quinzenais de condução;
- restrições de pausas obrigatórias;
- restrições de descanso diário e semanal;

- vínculos de continuidade temporal.

Cada conjunto de restrições é inserido de maneira modular, permitindo que seja ativado ou desativado conforme a configuração definida pelo usuário na interface. Essa abordagem garante flexibilidade experimental sem comprometer a integridade do modelo.

5.5.2 Discretização Temporal e Encadeamento

A discretização do horizonte em períodos de 15 minutos resulta em uma forte estrutura temporal na matriz, caracterizada por padrões repetitivos e dependências locais entre períodos consecutivos. Esse encadeamento se manifesta, por exemplo, nas restrições de continuidade de jornada, acúmulo de condução e reinício após períodos de descanso.

Como consequência, a matriz apresenta uma estrutura quase diagonal, com coeficientes não nulos concentrados em regiões específicas, o que é altamente favorável para solvers baseados em busca combinatória e propagação de restrições, como o CP-SAT.

5.5.3 Esparsidade da Matriz

Uma característica relevante da matriz construída é seu elevado grau de esparsidade. Mesmo para cenários de múltiplos dias, a proporção de elementos não nulos situa-se tipicamente entre 15% e 35%, dependendo do conjunto de restrições ativadas e do horizonte temporal considerado.

Essa esparsidade reduz significativamente o custo computacional associado à propagação de restrições e à exploração do espaço de soluções, contribuindo para tempos de resolução reduzidos mesmo em instâncias de maior escala.

5.5.4 Operações Elementares e Análise Estrutural

Além da construção direta da matriz de restrições, o simulador permite a aplicação de operações elementares, tais como:

- troca de linhas;
- multiplicação de linhas por constantes;
- combinação linear controlada de restrições.

Essas operações não alteram o conjunto de soluções viáveis do problema, mas podem ser utilizadas para fins educacionais, análise estrutural e diagnóstico do comportamento numérico do modelo. Em particular, elas permitem estudar o impacto da organização da matriz sobre sua densidade e sobre o desempenho do solver.

5.5.5 Versões da Matriz e Transformações

Durante o processo de modelagem, podem ser observadas diferentes versões da matriz de restrições, desde uma formulação inicial até versões finais após ajustes estruturais e operações elementares. Essas transformações preservam rigorosamente a equivalência matemática do problema, atuando apenas sobre sua representação algébrica.

Essa distinção entre formulação conceitual e representação matricial é importante para compreender como escolhas de implementação influenciam o desempenho computacional, sem modificar o significado do modelo.

5.5.6 Relação com o Desempenho Computacional

A forma como a matriz de restrições é construída e organizada tem impacto direto sobre o desempenho do solver. A combinação de esparsidade, estrutura temporal e modularidade favorece a atuação do CP-SAT, permitindo resolver instâncias com milhares de variáveis e restrições em tempos compatíveis com uso prático.

Os resultados experimentais apresentados no Capítulo 7 evidenciam essa eficiência, confirmando que a estrutura matricial adotada é adequada para o problema de escalonamento de motoristas sob regulamentação europeia.

5.6 Execução do Solver

Após a definição completa do modelo matemático e da matriz de restrições, o problema de escalonamento é submetido ao processo de resolução por meio do solver CP-SAT, integrante da biblioteca OR-Tools. Embora o modelo seja formulado como um problema de Programação Linear Inteira, o CP-SAT adota uma abordagem híbrida, combinando técnicas de programação por restrições, satisfatibilidade booleana e otimização inteira.

Essa escolha permite explorar de forma eficiente a forte estrutura temporal e a esparsidade do modelo, superando limitações frequentemente observadas em solvers MILP tradicionais quando aplicados a problemas regulados de grande escala.

5.6.1 Inicialização e Parâmetros do Solver

O simulador inicializa o solver com um conjunto explícito de parâmetros computacionais, que podem ser configurados diretamente pela interface. Entre os principais parâmetros destacam-se:

- limite máximo de tempo de execução;

- tolerância de *gap* relativo entre solução atual e melhor limite inferior;
- nível de verbosidade dos logs;
- limites associados ao processo iterativo do LNS.

Esses parâmetros permitem adaptar o comportamento do solver ao tamanho do problema e aos objetivos do experimento, possibilitando desde execuções rápidas até análises mais aprofundadas.

5.6.2 Critérios de Parada

A execução do solver é encerrada quando pelo menos um dos seguintes critérios é atendido:

- obtenção de solução ótima comprovada;
- atingimento do limite máximo de tempo;
- convergência dentro da tolerância de *gap* especificada;
- interrupção controlada em modos heurísticos ou matheurísticos.

Esses critérios garantem controle explícito sobre o esforço computacional, evitando execuções excessivamente longas e permitindo comparações justas entre diferentes estratégias de resolução.

5.6.3 Estados de Solução

Ao término da execução, o solver retorna um estado que indica a situação da resolução, podendo assumir, entre outros, os seguintes valores:

- **OPTIMAL**: solução ótima encontrada;
- **FEASIBLE**: solução viável encontrada dentro dos limites;
- **INFEASIBLE**: nenhuma solução viável existe;
- **UNBOUNDED**: problema mal formulado;
- **NOT_SOLVED**: execução interrompida antes da obtenção de solução.

Nos modos heurístico e LNS, soluções viáveis são aceitas mesmo na ausência de prova de otimalidade, desde que respeitem todas as restrições ativadas.

5.6.4 Execução nos Diferentes Modos de Resolução

No modo exato, o modelo completo é submetido diretamente ao CP-SAT, que realiza a busca por soluções ótimas respeitando integralmente todas as restrições legais e operacionais.

No modo heurístico, o solver pode ser utilizado de forma limitada ou não ser acionado, dependendo da configuração, sendo o foco a obtenção rápida de uma solução viável inicial.

No modo LNS, o solver é invocado repetidamente para resolver subproblemas restritos, nos quais apenas subconjuntos de variáveis são liberados a cada iteração. Esse processo permite melhorar progressivamente a solução, mantendo controle explícito sobre o esforço computacional.

5.6.5 Extração da Solução

Uma vez concluída a execução, o simulador extrai de forma sistemática os valores das variáveis de decisão, construindo:

- a matriz de alocação motorista-período;
- o número total de motoristas ativos;
- a carga de trabalho por motorista;
- métricas agregadas de cobertura e eficiência.

Essa extração é realizada de maneira uniforme para todos os modos de resolução, garantindo comparabilidade direta entre soluções exatas, heurísticas e matheurísticas.

5.6.6 Relação com os Resultados Experimentais

Os dados retornados pelo solver alimentam diretamente os indicadores e visualizações apresentados no Capítulo 7. Tempos de execução, estados de solução, valores da função objetivo e métricas derivadas são registrados de forma automática, assegurando consistência entre a implementação computacional e a análise empírica.

Dessa forma, a execução do solver constitui o elo fundamental entre o modelo matemático e os resultados experimentais discutidos posteriormente.

5.7 Fluxo Computacional do Simulador

O simulador desenvolvido segue um fluxo computacional bem definido e determinístico, estruturado em etapas sequenciais que garantem reprodutibilidade, consistência experimental e comparabilidade entre os diferentes modos de resolução disponíveis.

Independentemente da estratégia algorítmica selecionada (exata, heurística ou LNS), o fluxo geral de execução mantém a mesma estrutura lógica, diferenciando-se apenas nos mecanismos internos de resolução.

5.7.1 Etapas do Fluxo de Execução

O fluxo computacional do simulador pode ser descrito pelas seguintes etapas principais:

1. **Leitura e validação dos parâmetros de entrada:** o sistema coleta, por meio da interface, os parâmetros do problema, incluindo horizonte temporal, demanda, restrições legais ativadas, modo de resolução e função objetivo.
2. **Construção do modelo matemático:** com base nos parâmetros definidos, o simulador instancia as variáveis de decisão, a função objetivo selecionada e o conjunto de restrições ativas, conforme o modelo formal descrito no Capítulo 5.
3. **Montagem da matriz de restrições:** o modelo é convertido para sua representação matricial interna, preservando a estrutura esparsa e o encadeamento temporal característicos do problema.
4. **Execução do método de resolução:** de acordo com o modo selecionado, o sistema executa:
 - a resolução exata via CP-SAT;
 - a heurística construtiva para geração de solução inicial;
 - o processo iterativo de Large Neighborhood Search (LNS).
5. **Extração da solução:** os valores das variáveis são extraídos de forma padronizada, permitindo a construção da matriz de alocação, métricas agregadas e indicadores operacionais.
6. **Pós-processamento e análise:** o sistema calcula indicadores de desempenho, gera visualizações gráficas e registra métricas computacionais, como tempo de execução e estado da solução.

5.7.2 Determinismo e Reprodutibilidade

Para um conjunto fixo de parâmetros de entrada, o fluxo computacional do simulador é determinístico, assegurando que execuções repetidas produzam os mesmos resultados. Essa propriedade é essencial para a validação científica dos experimentos e para a comparação rigorosa entre diferentes estratégias de resolução.

Mesmo nos modos heurístico e LNS, nos quais decisões locais e processos iterativos estão presentes, o uso de critérios controlados e a ausência de componentes estocásticos garantem consistência entre execuções equivalentes.

5.7.3 Fluxo Específico por Modo de Resolução

No modo exato, o fluxo computacional segue diretamente da construção do modelo para a execução do solver CP-SAT, que explora o espaço de soluções até atingir os critérios de parada definidos.

No modo heurístico, a geração da solução ocorre por meio de regras construtivas gulosas, com foco na obtenção rápida de uma solução viável, respeitando as restrições legais ativadas.

No modo LNS, o fluxo incorpora ciclos iterativos adicionais, nos quais subconjuntos de variáveis são liberados a cada iteração, mantendo o restante da solução fixo. A cada ciclo, um subproblema restrito é resolvido pelo solver inteiro, permitindo melhorias progressivas da solução.

Apesar dessas diferenças internas, o fluxo de entrada e saída permanece uniforme, o que garante comparabilidade direta entre os resultados obtidos nos três modos.

5.7.4 Integração com os Resultados Experimentais

O fluxo computacional descrito neste capítulo fornece a base operacional para todos os experimentos apresentados no Capítulo 7. A uniformidade na coleta de métricas, na extração das soluções e no cálculo dos indicadores assegura que as diferenças observadas nos resultados sejam atribuídas exclusivamente às estratégias de resolução adotadas, e não a variações no processo de execução.

Dessa forma, o fluxo computacional do simulador estabelece o elo entre a formulação matemática, a implementação computacional e a validação empírica dos métodos propostos.

5.8 Modos de Resolução e Estratégias de Otimização

No simulador desenvolvido, é feita uma distinção clara entre os conceitos de *modo de resolução* e *estratégia de otimização*. O modo de resolução refere-se ao método algorítmico

empregado para explorar o espaço de soluções, enquanto a estratégia de otimização define o critério adotado pela função objetivo do modelo matemático.

Essa separação conceitual permite combinar livremente diferentes abordagens de resolução com diferentes objetivos operacionais, ampliando o escopo analítico da plataforma experimental.

5.8.1 Modo de Resolução Exato

No modo de resolução exato, o problema completo de escalonamento é formulado como um modelo de Programação Linear Inteira e resolvido diretamente pelo solver CP-SAT. Nesse modo, todas as variáveis, restrições legais ativadas e a função objetivo selecionada são consideradas simultaneamente, permitindo a busca por soluções ótimas comprovadas, sempre que o tempo computacional disponível o permita.

Esse modo fornece uma referência de qualidade para os demais métodos, servindo como base para validação das soluções heurísticas e matheurísticas.

5.8.2 Modo Heurístico

O modo heurístico utiliza uma estratégia construtiva gulosa para gerar rapidamente uma solução viável. A heurística opera de forma incremental, priorizando o atendimento da demanda por período e respeitando os principais limites operacionais e legais ativados.

Embora não haja garantia de otimalidade nesse modo, ele apresenta baixo custo computacional e é particularmente adequado para cenários de grande escala ou situações em que respostas rápidas são necessárias. Além disso, a solução gerada pode ser utilizada como ponto de partida para métodos mais avançados, como o LNS.

5.8.3 Modo Matheurístico (LNS)

O modo LNS (*Large Neighborhood Search*) combina elementos heurísticos com reotimizações locais via solver inteiro. A partir de uma solução inicial, subconjuntos de variáveis são liberados a cada iteração, mantendo o restante da solução fixo. Em seguida, um subproblema restrito é resolvido pelo solver, permitindo melhorias progressivas da solução.

Essa abordagem explora de forma eficiente a estrutura temporal do problema, equilibrando qualidade da solução e esforço computacional. O LNS permite controlar explicitamente o tamanho das vizinhanças e o número de iterações, tornando-o adequado para análises experimentais comparativas.

5.8.4 Estratégias de Otimização

De forma independente do modo de resolução, o simulador permite selecionar a estratégia de otimização adotada pelo modelo matemático. As principais estratégias implementadas são:

- **Maximização da resposta à demanda:** prioriza o atendimento máximo da demanda operacional ao longo do horizonte de planejamento, admitindo, quando necessário, demanda não atendida de forma controlada;
- **Minimização do número de motoristas ativos:** busca reduzir o uso da força de trabalho, assumindo atendimento integral da demanda.

Essas estratégias refletem objetivos operacionais distintos e permitem avaliar compromissos entre cobertura e eficiência de recursos.

5.8.5 Combinação entre Modos e Estratégias

A arquitetura do simulador permite combinar qualquer modo de resolução com qualquer estratégia de otimização. Por exemplo, a minimização do número de motoristas pode ser realizada tanto por meio de uma resolução exata quanto por heurísticas ou LNS.

Essa flexibilidade possibilita análises comparativas sistemáticas, nas quais os efeitos do método de resolução e da função objetivo podem ser avaliados de forma isolada ou combinada.

5.8.6 Relação com os Experimentos Apresentados

Nos experimentos apresentados no Capítulo 7, os três modos de resolução (exato, heurístico e LNS) são avaliados sob estratégias de otimização equivalentes, garantindo comparabilidade direta dos resultados. As diferenças observadas em termos de qualidade da solução, tempo computacional e estabilidade são, portanto, atribuídas exclusivamente às características dos métodos de resolução adotados.

Essa abordagem reforça a validade experimental do estudo e permite uma interpretação clara dos compromissos envolvidos em cada estratégia.

5.9 Geração de Indicadores e Gráficos

A geração de indicadores e visualizações gráficas constitui uma etapa essencial do simulador, pois permite avaliar de forma objetiva a qualidade das soluções obtidas e interpretar o comportamento dos diferentes métodos de resolução. Todos os indicadores

apresentados são derivados diretamente das variáveis de decisão do modelo matemático, assegurando consistência entre formulação, implementação e análise empírica.

Os indicadores foram projetados para responder a questões operacionais específicas, como cobertura da demanda, utilização da força de trabalho, estabilidade temporal e balanceamento da carga entre motoristas.

5.9.1 Indicadores de Cobertura da Demanda

O indicador de cobertura da demanda mede, para cada período discretizado, a diferença entre a demanda operacional e o número de motoristas efetivamente alocados. Esse indicador permite identificar períodos de subcobertura, cobertura exata ou sobrealocação.

A partir dessa informação, são derivados indicadores agregados, como a taxa global de cobertura e a quantidade total de demanda não atendida, utilizados diretamente na análise comparativa entre métodos.

5.9.2 Indicadores de Utilização e Sobrecarga

Para avaliar o uso da força de trabalho, o simulador calcula métricas de utilização por motorista, incluindo:

- carga total de trabalho por motorista;
- média de carga entre motoristas ativos;
- identificação de sobrecargas individuais.

Esses indicadores permitem verificar se a solução concentra excessivamente a carga em poucos motoristas ou se promove uma distribuição mais equilibrada, aspecto particularmente relevante em cenários regulados.

5.9.3 Indicadores Temporais e Estabilidade

Indicadores temporais são utilizados para analisar a estabilidade da solução ao longo do horizonte de planejamento. Entre eles destacam-se:

- desvio padrão da cobertura por período;
- variação do número de motoristas ativos ao longo do tempo;
- regularidade dos padrões de alocação.

Essas métricas permitem avaliar não apenas a qualidade pontual da solução, mas também seu comportamento dinâmico, aspecto importante para aplicações operacionais reais.

5.9.4 Indicadores Computacionais

Além dos indicadores operacionais, o simulador registra métricas computacionais associadas ao processo de resolução, tais como:

- tempo total de execução;
- número de iterações (no caso do LNS);
- estado final da solução (ótima, viável, interrompida);
- *gap* relativo entre solução e melhor limite conhecido.

Esses indicadores permitem comparar o esforço computacional exigido por cada modo de resolução, complementando a análise da qualidade das soluções obtidas.

5.9.5 Visualizações Gráficas

Os indicadores calculados são apresentados por meio de visualizações gráficas interativas, incluindo gráficos de linhas, barras e distribuições. Essas visualizações permitem explorar padrões temporais, identificar picos de demanda e analisar diferenças entre métodos de resolução de forma intuitiva.

Embora o simulador gere uma ampla variedade de gráficos, nesta dissertação são apresentados apenas aqueles considerados mais representativos para a análise comparativa entre os métodos exato, heurístico e LNS, evitando a proliferação excessiva de artefatos visuais.

5.9.6 Relação com os Resultados Experimentais

Todos os indicadores e gráficos descritos nesta seção são utilizados diretamente nos experimentos apresentados no Capítulo 7. A padronização na geração e interpretação dessas métricas assegura que as conclusões extraídas sejam fundamentadas em evidências quantitativas consistentes e comparáveis entre os diferentes métodos avaliados.

5.10 Módulo de Operações Elementares

Além da resolução do problema de escalonamento, o simulador incorpora um módulo específico para a aplicação de operações elementares sobre a matriz de restrições.

Esse módulo foi concebido com objetivos predominantemente educacionais, analíticos e diagnósticos, não interferindo no processo de otimização utilizado para a geração dos resultados experimentais principais.

As operações elementares implementadas incluem transformações clássicas de álgebra linear, tais como troca de linhas, multiplicação de linhas por constantes e combinações lineares controladas entre restrições.

5.10.1 Objetivos do Módulo

A inclusão do módulo de operações elementares atende a três objetivos principais:

- apoiar atividades educacionais relacionadas à modelagem matemática e à Programação Linear Inteira;
- permitir a análise estrutural da matriz de restrições, em especial no que diz respeito à densidade, organização e encadeamento temporal;
- auxiliar no diagnóstico de problemas numéricos ou estruturais durante a fase de desenvolvimento do modelo.

Esse módulo possibilita explorar a relação entre a formulação algébrica do problema e seu impacto sobre o desempenho computacional do solver, sem alterar o conjunto de soluções viáveis.

5.10.2 Preservação da Equivalência Matemática

Todas as operações disponibilizadas no simulador são cuidadosamente implementadas de modo a preservar a equivalência matemática do sistema de restrições. Em particular, as transformações aplicadas não modificam o espaço de soluções viáveis nem o valor ótimo do problema, atuando apenas sobre sua representação algébrica.

Essa propriedade é fundamental para garantir que o uso do módulo não comprometa a validade dos experimentos realizados, sendo sua aplicação restrita a contextos exploratórios e analíticos.

5.10.3 Análise da Densidade da Matriz

Um dos usos práticos do módulo de operações elementares é a análise da densidade da matriz de restrições antes e depois de transformações específicas. O simulador calcula métricas de densidade estrutural, permitindo avaliar como diferentes formas de organização das restrições influenciam o desempenho do solver.

Essa análise é particularmente relevante em problemas de grande escala, nos quais pequenas variações na estrutura da matriz podem impactar significativamente o tempo de resolução.

5.10.4 Relação com os Resultados Apresentados

Embora o módulo de operações elementares não seja utilizado diretamente na geração das soluções analisadas no Capítulo 7, ele desempenhou papel importante durante o desenvolvimento e validação do modelo, auxiliando na compreensão da estrutura do problema e na identificação de formulações mais eficientes.

Dessa forma, o módulo contribui para a robustez metodológica do trabalho, reforçando a transparência e o rigor científico adotados na implementação computacional.

5.11 Extensões Heurísticas e Metaheurísticas

Embora o núcleo do simulador seja fundamentado em um modelo de Programação Linear Inteira resolvido de forma exata pelo solver CP-SAT, a plataforma foi estendida para incorporar métodos heurísticos e metaheurísticos. Essas extensões têm como objetivo ampliar o escopo experimental, reduzir tempos computacionais em cenários de maior escala e permitir a análise comparativa entre diferentes estratégias de resolução sob um mesmo conjunto de restrições legais.

As abordagens heurísticas e metaheurísticas implementadas não substituem o modelo matemático exato, mas atuam como mecanismos complementares, sempre mantendo a validade jurídica e a consistência estrutural do problema.

5.11.1 Heurística Construtiva Gulosa

A primeira extensão implementada consiste em uma heurística construtiva gulosa (*greedy initial allocation*), responsável por gerar rapidamente soluções viáveis a partir da demanda operacional por período.

Essa heurística opera de forma incremental, priorizando períodos com maior demanda e alocando motoristas de acordo com regras locais de viabilidade, respeitando limites operacionais básicos e restrições legais essenciais. O objetivo principal dessa abordagem é produzir uma solução inicial consistente, com baixo custo computacional, que possa ser utilizada de forma independente ou como ponto de partida para métodos de melhoria.

A heurística gulosa apresenta desempenho computacional significativamente superior ao método exato em cenários de grande escala, porém não garante otimalidade global,

sendo particularmente adequada para análises exploratórias e aplicações em tempo quase real.

5.11.2 Large Neighborhood Search (LNS)

Para superar as limitações inerentes às heurísticas construtivas, o simulador incorpora uma abordagem de *Large Neighborhood Search* (LNS), caracterizada pela destruição e reconstrução parcial da solução.

No método LNS, a partir de uma solução inicial viável (gerada pela heurística gulosa ou pelo solver exato), subconjuntos de períodos são iterativamente liberados, enquanto o restante da solução permanece fixo. Em cada iteração, um problema restrito é reconstruído e resolvido novamente pelo solver inteiro, permitindo melhorias progressivas da solução.

Essa estratégia combina a robustez da otimização exata com a flexibilidade exploratória típica das metaheurísticas, oferecendo um equilíbrio eficaz entre qualidade da solução e esforço computacional.

5.11.3 Controle de Relaxação e Iterações

O simulador permite controlar explicitamente parâmetros associados ao processo LNS, tais como o tamanho da vizinhança liberada, o número máximo de iterações e o nível de relaxação aplicado em cada etapa.

Esse controle possibilita estudar o comportamento do algoritmo sob diferentes configurações, analisando o impacto de decisões como a intensidade da destruição, a frequência de reotimizações e os critérios de parada adotados.

5.11.4 Garantia de Validade das Soluções

Um aspecto central da implementação é que todas as soluções finais geradas pelos métodos heurísticos e metaheurísticos são sempre validadas pelo modelo de Programação Linear Inteira e pelo solver CP-SAT. Mesmo quando uma solução é construída heurística ou parcialmente, sua aceitação final depende do atendimento integral das restrições legais e operacionais.

Essa estratégia garante que nenhuma solução inválida do ponto de vista jurídico ou matemático seja utilizada nos experimentos apresentados nesta dissertação.

5.11.5 Relação com os Experimentos Computacionais

As extensões heurísticas e metaheurísticas desempenham papel fundamental nos experimentos discutidos no Capítulo 7. Em particular, os resultados permitem comparar:

- qualidade da solução obtida;
- tempo computacional;
- estabilidade da cobertura da demanda;
- eficiência na utilização da força de trabalho.

Essas comparações evidenciam os compromissos entre métodos exatos, heurísticos e metaheurísticos, reforçando a importância de abordagens híbridas em problemas de escalonamento sob restrições legais complexas.

5.12 Integração de Aprendizado de Máquina

Além das abordagens exatas, heurísticas e metaheurísticas, o simulador incorpora um módulo experimental de aprendizado de máquina supervisionado, cujo objetivo é auxiliar decisões locais e estratégicas durante o processo de otimização. Essa integração foi concebida como uma camada complementar, não substitutiva, mantendo plena compatibilidade com o modelo matemático exato e com as restrições legais do Regulamento (CE) n.º 561/2006.

O aprendizado de máquina não altera diretamente a formulação do problema, a função objetivo nem o conjunto de restrições. Sua função é orientar escolhas heurísticas, reduzindo o espaço de busca e priorizando decisões potencialmente mais promissoras, sem comprometer a validade jurídica ou matemática das soluções.

5.12.1 Motivação para o Uso de Aprendizado Supervisionado

Problemas de escalonamento sob restrições temporais rígidas apresentam elevada complexidade combinatória, especialmente quando resolvidos repetidamente em cenários de simulação. Embora solvers exatos sejam capazes de produzir soluções ótimas, o custo computacional pode crescer rapidamente com o tamanho do horizonte e da força de trabalho.

Nesse contexto, o aprendizado supervisionado surge como uma ferramenta para capturar padrões recorrentes observados em soluções ótimas ou quase ótimas, auxiliando o processo decisório em etapas heurísticas e metaheurísticas, sem abrir mão do rigor do modelo exato.

5.12.2 Modelos Supervisionados Implementados

Foram definidos dois modelos supervisionados distintos, com papéis claramente separados no simulador.

O primeiro modelo, denominado f_1 , atua no nível local, avaliando decisões de alocação motorista-período durante a heurística construtiva. Esse modelo recebe como entrada características associadas ao período, à demanda, ao estado do motorista e ao contexto operacional, retornando um escore que representa a atratividade relativa daquela alocação.

O segundo modelo, denominado f_2 , opera em nível agregado no contexto do algoritmo LNS. Seu objetivo é avaliar subconjuntos de períodos candidatos à liberação, estimando o potencial de melhoria da solução caso aquela vizinhança seja explorada. Dessa forma, o modelo auxilia na priorização de vizinhanças mais promissoras, reduzindo o número de iterações improdutivas.

5.12.3 Arquitetura Resiliente e Comportamento Padrão

A integração com aprendizado de máquina foi projetada de forma resiliente. Caso os modelos não estejam disponíveis, não estejam treinados ou o ambiente de execução não suporte o módulo de aprendizado, o simulador recorre automaticamente a regras heurísticas clássicas e determinísticas.

Esse mecanismo garante que o comportamento padrão do simulador seja sempre reproduzível e independente do aprendizado de máquina, evitando qualquer dependência estrutural ou viés não controlado nos experimentos.

5.12.4 Geração de Dados e Treinamento dos Modelos

Para viabilizar o aprendizado supervisionado, foi desenvolvido um pipeline automático de geração de dados. A partir de múltiplas instâncias sintéticas de demanda, o simulador executa soluções heurísticas e soluções exatas via solver inteiro, registrando decisões intermediárias, estados do sistema e métricas de desempenho.

Os dados coletados são organizados em dois conjuntos distintos: um dataset voltado ao treinamento do modelo f_1 , com informações locais de decisão, e um dataset voltado ao modelo f_2 , composto por características agregadas das vizinhanças exploradas no LNS.

O treinamento dos modelos é realizado externamente ao fluxo principal de otimização, utilizando algoritmos de *gradient boosting*. Os modelos treinados são posteriormente carregados dinamicamente pelo simulador, não interferindo na formulação matemática nem no processo de validação das soluções.

5.12.5 Papel do Aprendizado de Máquina nos Experimentos

Nos experimentos apresentados nesta dissertação, o uso de aprendizado de máquina é tratado como um recurso opcional e experimental. Os resultados principais utilizam o

comportamento determinístico do simulador, permitindo comparações justas entre métodos exatos, heurísticos e metaheurísticos.

Quando ativado, o aprendizado supervisionado atua como um mecanismo de aceleração e refinamento, potencialmente reduzindo tempos de convergência e melhorando a estabilidade das soluções, sem alterar os critérios formais de avaliação adotados.

5.12.6 Garantia de Rigor Metodológico

É importante destacar que todas as soluções finais, independentemente do uso ou não de aprendizado de máquina, são sempre validadas pelo modelo de Programação Linear Inteira e pelo solver CP-SAT. O aprendizado supervisionado não possui autonomia para violar restrições, modificar objetivos ou aceitar soluções inválidas.

Dessa forma, a integração de aprendizado de máquina preserva rigor científico, transparência metodológica e conformidade legal, atuando exclusivamente como uma camada auxiliar de apoio à decisão.

5.13 Geração de Datasets e Treinamento dos Modelos

A integração de aprendizado de máquina no simulador exigiu o desenvolvimento de um pipeline específico para geração de dados supervisionados, totalmente compatível com o domínio do problema e com o modelo matemático subjacente. Esse pipeline foi concebido para garantir consistência entre os dados utilizados no treinamento e as decisões efetivamente tomadas pelo sistema de escalonamento.

5.13.1 Origem dos Dados Supervisionados

Os datasets utilizados no treinamento dos modelos supervisionados não são provenientes de fontes externas nem de bases genéricas. Eles são gerados diretamente pelo próprio simulador, a partir da execução controlada de múltiplos cenários de escalonamento.

Cada instância de dados é construída a partir de:

- curvas sintéticas de demanda por período;
- diferentes configurações de restrições legais ativas;
- soluções heurísticas iniciais;
- soluções exatas obtidas via solver inteiro;
- histórico de decisões intermediárias durante o processo de otimização.

Essa estratégia garante que os dados de treinamento reflitam fielmente a lógica do problema, suas restrições legais e os critérios operacionais adotados no modelo matemático.

5.13.2 Dataset para o Modelo Local de Alocação (f_1)

O primeiro conjunto de dados foi projetado para treinar o modelo local de decisão f_1 , utilizado durante a heurística construtiva. Cada observação do dataset corresponde a uma decisão candidata de alocação motorista-período.

As variáveis explicativas incluem, entre outras:

- demanda do período;
- nível de cobertura acumulada;
- estado do motorista (ativo, descanso, condução acumulada);
- posição temporal no horizonte;
- indicadores locais de carga e utilização.

O rótulo associado a cada observação indica a qualidade daquela decisão, medida com base na solução final obtida (ótima ou quase ótima), permitindo ao modelo aprender padrões de alocação mais eficientes.

5.13.3 Dataset para Avaliação de Vizinhanças no LNS (f_2)

O segundo dataset é voltado ao treinamento do modelo f_2 , responsável por avaliar vizinhanças candidatas no algoritmo LNS. Nesse caso, cada observação representa um subconjunto de períodos liberados para reotimização em uma iteração do LNS.

As características utilizadas incluem:

- tamanho da vizinhança;
- posição temporal dos períodos liberados;
- nível de violação ou folga das restrições;
- variação da função objetivo nas iterações anteriores;
- indicadores agregados de estabilidade e carga.

O rótulo indica o ganho efetivo obtido após a reotimização daquela vizinhança, permitindo ao modelo aprender quais padrões tendem a gerar melhorias mais significativas.

5.13.4 Pipeline de Geração Automática

A geração dos datasets é realizada de forma automática e offline, por meio de execuções repetidas do simulador em diferentes cenários. Esse processo não interfere no fluxo principal de otimização e não ocorre durante a execução dos experimentos analisados nesta dissertação.

Cada execução gera registros estruturados que são armazenados em formato tabular, permitindo posterior auditoria, reprodutibilidade e controle de versões dos datasets utilizados.

5.13.5 Treinamento dos Modelos Supervisionados

O treinamento dos modelos f_1 e f_2 é realizado externamente ao simulador, utilizando algoritmos de *gradient boosting*, devido à sua capacidade de modelar relações não lineares e lidar eficientemente com variáveis heterogêneas.

Os modelos treinados são posteriormente exportados e carregados dinamicamente pelo simulador, sem necessidade de recompilação ou modificação do núcleo de otimização.

5.13.6 Separação entre Treinamento e Avaliação

É importante destacar que os dados utilizados para treinamento dos modelos não são reutilizados nos experimentos de avaliação apresentados nesta dissertação. Essa separação evita vazamento de informação e garante que os resultados observados não sejam artificialmente favorecidos pelo aprendizado supervisionado.

Nos experimentos principais, o simulador é avaliado tanto com quanto sem o uso dos modelos treinados, permitindo comparar o impacto real do aprendizado de máquina sobre desempenho computacional e qualidade das soluções.

5.13.7 Considerações Metodológicas

A estratégia adotada garante que o aprendizado supervisionado:

- não introduz viés externo ao problema;
- não altera a formulação matemática;
- não compromete a validade legal das soluções;
- atua exclusivamente como mecanismo auxiliar.

Dessa forma, o uso de aprendizado de máquina reforça o caráter experimental e exploratório da plataforma, sem comprometer o rigor científico exigido para a análise dos resultados apresentados.

5.14 Visualização Avançada e Análise Iterativa

A visualização dos resultados desempenha papel central no simulador desenvolvido, não apenas como mecanismo de apresentação, mas como ferramenta analítica para interpretação do comportamento dos algoritmos, diagnóstico de soluções e validação das hipóteses experimentais. Para esse fim, foram implementados módulos de visualização capazes de representar tanto indicadores agregados quanto informações temporais detalhadas.

5.14.1 Visualizações Temporais da Solução

Um conjunto fundamental de gráficos apresenta a evolução temporal da demanda, da capacidade alocada e da cobertura efetiva ao longo do horizonte de planejamento. Essas visualizações permitem verificar, período a período, se a solução obtida atende integralmente à demanda operacional ou se existem déficits pontuais.

Adicionalmente, são gerados gráficos de margem de segurança, que evidenciam situações de sobrecarga ou subutilização, permitindo avaliar o grau de robustez da solução frente a variações operacionais. Esses gráficos são particularmente úteis para identificar períodos críticos e padrões recorrentes de instabilidade.

5.14.2 Indicadores de Estabilidade e Eficiência

Para além da cobertura da demanda, o simulador calcula indicadores de estabilidade e eficiência associados à alocação de motoristas. Entre eles, destacam-se:

- o desvio padrão da cobertura ao longo do tempo;
- a distribuição da carga de trabalho por motorista;
- o tempo médio de presença ativa;
- a relação entre esforço alocado e demanda atendida.

Esses indicadores são apresentados tanto de forma numérica quanto gráfica, permitindo avaliar se a solução, embora viável, apresenta comportamento regular ou excessivamente concentrado em determinados motoristas ou períodos.

5.14.3 Análise Iterativa no Modo LNS

No modo de resolução LNS, visualizações adicionais são geradas para acompanhar o processo iterativo de melhoria da solução. Em particular, são apresentados gráficos que mostram:

- a evolução do valor da função objetivo ao longo das iterações;
- o impacto das liberações parciais da solução;
- o comportamento do nível de relaxamento aplicado;
- a frequência e magnitude das melhorias obtidas.

Essas visualizações permitem analisar empiricamente a convergência do método, identificando se o algoritmo se estabiliza prematuramente ou continua a produzir ganhos marginais ao longo das iterações.

5.14.4 Comparação entre Modos de Resolução

O simulador permite a visualização comparativa direta entre os modos exato, heurístico e LNS. Gráficos consolidados exibem diferenças em termos de:

- tempo computacional;
- qualidade da solução;
- número de motoristas ativos;
- estabilidade temporal da alocação.

Essa comparação visual facilita a identificação de compromissos entre esforço computacional e qualidade da solução, reforçando o caráter experimental da plataforma.

5.14.5 Visualizações para Diagnóstico e Auditoria

Além das visualizações orientadas a desempenho, o simulador oferece gráficos voltados ao diagnóstico e à auditoria das soluções. Mapas de calor da matriz de alocação permitem identificar padrões de ativação, blocos contínuos de trabalho e eventuais irregularidades temporais.

Essas representações visuais auxiliam na verificação manual da conformidade legal, facilitando a identificação de violações sutis que poderiam não ser evidentes em indicadores agregados.

5.14.6 Integração com o Processo Experimental

As visualizações avançadas não são utilizadas apenas como saída final, mas como parte integrante do processo experimental. Elas orientam ajustes de parâmetros, avaliação de estratégias heurísticas e calibração de métodos LNS e de aprendizado de máquina.

Dessa forma, o módulo de visualização contribui diretamente para o refinamento do simulador e para a interpretação dos resultados apresentados nos capítulos experimentais seguintes, fortalecendo a conexão entre implementação computacional e análise empírica.

5.15 Posicionamento do Simulador como Plataforma Experimental

A implementação computacional desenvolvida ao longo deste trabalho ultrapassa o papel tradicional de um resolvidor de problemas de Programação Linear Inteira, configurando-se como uma plataforma experimental híbrida para o estudo do escalonamento de motoristas sob regulamentação europeia.

Diferentemente de abordagens focadas exclusivamente na obtenção de uma solução ótima, o simulador foi concebido para permitir a investigação sistemática de diferentes estratégias de resolução, objetivos operacionais e configurações de restrições, mantendo sempre como núcleo normativo o modelo matemático exato apresentado nos capítulos anteriores.

5.15.1 Unificação de Estratégias de Resolução

A plataforma integra, em um mesmo ambiente computacional, métodos exatos, heurísticos e matheurísticos, possibilitando comparações diretas sob condições controladas. A Programação Linear Inteira resolvida via CP-SAT atua como referência de validade e qualidade, enquanto a heurística construtiva e o método LNS oferecem alternativas computacionalmente mais eficientes para cenários de maior escala ou com restrições temporais mais severas.

Essa unificação permite analisar, de forma rigorosa, os compromissos entre qualidade da solução, esforço computacional e estabilidade operacional, aspecto essencial em aplicações reais de transporte rodoviário.

5.15.2 Separação entre Modelo, Resolução e Análise

Um dos princípios centrais da arquitetura do simulador é a separação clara entre:

- o modelo matemático normativo;
- os métodos de resolução empregados;

- os mecanismos de análise e visualização.

Essa separação garante que modificações em estratégias heurísticas, parâmetros do solver ou módulos de aprendizado de máquina não comprometam a validade do modelo legal subjacente. Ao mesmo tempo, facilita a extensão da plataforma para novos cenários, restrições ou objetivos sem necessidade de reformulação estrutural.

5.15.3 Reprodutibilidade e Controle Experimental

O simulador foi projetado para assegurar reprodutibilidade dos experimentos. Para um conjunto fixo de parâmetros de entrada — horizonte temporal, demanda, restrições legais e modo de resolução — o comportamento do sistema é determinístico, permitindo a replicação integral dos resultados apresentados.

Além disso, o controle explícito sobre a ativação de restrições, critérios de otimização e estratégias de resolução possibilita a realização de estudos de sensibilidade e análises comparativas rigorosas, fundamentais para avaliação científica.

5.15.4 Integração Opcional de Aprendizado de Máquina

O aprendizado de máquina foi integrado como uma camada complementar, não como substituto da otimização exata. Sua utilização restringe-se à priorização de decisões locais e à seleção de vizinhanças promissoras no contexto do LNS, sem alterar diretamente restrições ou funções objetivo.

Essa abordagem preserva o rigor matemático do modelo, ao mesmo tempo em que amplia o potencial experimental da plataforma, permitindo investigar a interação entre otimização combinatória e aprendizado supervisionado de forma controlada e interpretável.

5.15.5 Aplicabilidade Científica e Industrial

Do ponto de vista científico, o simulador oferece um ambiente completo para a investigação de problemas de escalonamento sob restrições legais complexas, viabilizando estudos comparativos, análises de escalabilidade e experimentação com estratégias híbridas.

Do ponto de vista industrial, a arquitetura modular, a capacidade de operar com dados reais e a interface interativa tornam a plataforma adequada para aplicações em contextos corporativos, estudos de viabilidade operacional e suporte à tomada de decisão, preservando confidencialidade e conformidade regulatória.

5.15.6 Síntese do Capítulo

Em síntese, o simulador desenvolvido consolida-se como uma plataforma experimental robusta, flexível e extensível, que conecta rigor matemático, implementação computacional e análise empírica. Ele fornece a base operacional necessária para os experimentos e comparações apresentados nos capítulos seguintes, assegurando consistência metodológica entre formulação teórica, implementação e validação dos resultados.

6 Resultados

6.1 Introdução

Este capítulo apresenta os resultados computacionais obtidos a partir da execução do modelo de Programação Linear Inteira (PLI) e da implementação computacional descrita nos Capítulos 5 e 6. Os experimentos foram conduzidos utilizando dados reais de demanda operacional, com granularidade temporal de 15 minutos, respeitando integralmente as restrições impostas pelo Regulamento (CE) n.º 561/2006.

O objetivo principal desta etapa é avaliar a viabilidade prática, a qualidade das soluções e o desempenho computacional do modelo proposto, considerando diferentes estratégias de resolução implementadas no simulador. Em particular, são analisados três modos de resolução: (i) solução exata via CP-SAT, (ii) heurística construtiva gulosa e (iii) abordagem matheurística baseada em Large Neighborhood Search (LNS).

Este capítulo inicia-se com a análise detalhada dos resultados obtidos pelo método exato, que serve como referência normativa e de qualidade para a comparação posterior com os métodos heurísticos e metaheurísticos.

6.2 Roteiro Experimental

Os resultados apresentados neste capítulo seguem um roteiro experimental estruturado, concebido para avaliar de forma progressiva o comportamento do simulador sob diferentes estratégias de resolução e níveis de complexidade.

O roteiro foi organizado nas seguintes etapas:

- validação do modelo matemático por meio da solução exata;
- avaliação do desempenho computacional do solver CP-SAT;
- análise da qualidade das soluções em termos de cobertura, estabilidade e eficiência;
- comparação entre métodos exato, heurístico e LNS;
- discussão dos compromissos entre qualidade da solução e esforço computacional.

Essa abordagem sistemática garante rastreabilidade, reprodutibilidade e rigor científico, permitindo que cada estratégia seja analisada sob condições controladas e comparáveis.

6.3 Configuração Geral dos Cenários

Os cenários experimentais foram configurados com base nos seguintes parâmetros principais:

- solver: CP-SAT (OR-Tools);
- granularidade temporal: 15 minutos por período;
- horizonte de referência apresentado neste capítulo: 24 horas (96 períodos);
- variável principal: $x_{d,t}$ (binária);
- ativação integral das restrições legais;
- demanda real, variável e não uniforme por período.

Embora o simulador suporte horizontes de 7, 15 e 30 dias, optou-se por apresentar neste capítulo os resultados detalhados do cenário diário de 24 horas, por se tratar do menor horizonte completo capaz de capturar todas as restrições diárias, pausas obrigatórias e padrões de variabilidade da demanda. Os resultados em horizontes estendidos apresentaram comportamento qualitativamente semelhante, conforme verificado nos experimentos complementares.

Essas configurações permitem avaliar não apenas a viabilidade legal das escalas, mas também a estabilidade, robustez e eficiência computacional do modelo.

Tabela 4 – Parâmetros gerais utilizados nos experimentos

Parâmetro	Valor
Solver	CP-SAT (OR-Tools)
Granularidade temporal	15 minutos
Horizontes avaliados	7, 15 e 30 dias
Tipo de variável	Binária ($x_{d,t}$)
Restrições legais	Regulamento (CE) n.º 561/2006 (todas ativas)
Demanda	Variável por período
Semente aleatória	Fixa para reprodutibilidade

6.4 Execução do Solver

A Figura 13 apresenta um exemplo de execução completa do solver CP-SAT para o cenário de 24 horas, com todas as restrições legais ativas.

The screenshot displays the Gurobi solver interface with the following sections:

- Global Parameters:**
 - Global:** horizon_hours: 24 (Integer, Limit Iterations: 0, Penalty for unmet: 800.00).
 - Slot:** 15 (Integer, Relaxation: 0, Cap. Tasks per Driver per Slot: 6).
 - Tolerance Coverage:** 1.00 (Acceptable Density: 0.01, Drivers (0, no limit): 120).
 - Algorithm:** GLOP-LP | SCIP-MIP.
 - Slot Demand:** A list of 96 values representing demand per slot, with a total demand of 96.
- Objective Function | Restrictions:**
 - Objective:** Select objective strategy: Minimize Total Number of Drivers.
 - Restrictions:**
 - Break Options:** 45 minutes (selected), 15+30 split, 30+15 split.
 - Coverage Need:** $\sum_{j \in \text{valid slot}(i)} X[j] \geq \text{need}[i]$ (checked).
 - Daily Driving Limit:** $\sum_{p \in \text{day}} X[p] \leq 36p(1p = 15 \text{ minutes} | 36p = 9h)$ (checked).
 - Minimum Daily Rest:** Rest Period $\geq 44p(1p = 15 \text{ minutes} | 44p = 11h)$ (unchecked).
 - Reduced Daily Rest:** $\geq 36p(1p = 15 \text{ minutes} | 36p = 9h | \text{Max. } 3x | 14 \text{ days})$ (unchecked).
 - Weekly Rest:** Rest Period $\geq 180p(1p = 15 \text{ minutes} | 180p = 45h)$ (unchecked).
 - Rest after 6 days of work:** 6 days Work \Rightarrow Rest for One (unchecked).
 - Weekly Driving Limit:** $\sum_{p \in \text{week}} X[p] \leq 224p(1p = 15 \text{ minutes} | 224p = 56h)$ (unchecked).
 - Biweekly Driving Limit:** $\sum_{p \in \text{Biweekly}} X[p] \leq 360p(1p = 15 \text{ minutes} | 360p = 90h)$ (unchecked).
 - Biweekly Rest:** $\geq 96p(1p = 15 \text{ minutes}, \text{logo} 96p = 24h)$ (unchecked).

Figura 13 – Execução do solver CP-SAT no modo exato.

Observa-se que o solver retorna soluções ótimas de forma consistente, com tempos computacionais médios inferiores a dois segundos, mesmo em instâncias com milhares de variáveis e restrições. Em média, os modelos resolvidos apresentaram cerca de 9.600 variáveis e aproximadamente 7.600 restrições lineares.

Esse desempenho confirma que a formulação PLI, associada ao CP-SAT, é adequada para aplicação prática no setor, permitindo replanejamento rápido e análises exploratórias sob condições operacionais realistas.

6.5 Comportamento da Demanda

A demanda utilizada nos experimentos apresenta elevada variabilidade ao longo do horizonte analisado, característica típica de operações de transporte rodoviário com janelas de atendimento contínuas e múltiplos centros operacionais.

Observam-se períodos de pico com elevada necessidade de motoristas, intercalados com intervalos de menor demanda. Essa irregularidade não foi suavizada intencionalmente, pois constitui um elemento essencial para testar a robustez do modelo frente a cenários realistas, nos quais variações abruptas são frequentes.

O bom desempenho do modelo mesmo sob essas condições reforça sua aplicabilidade prática.

6.6 Comparação Demanda *versus* Alocação

A Figura 14 apresenta a comparação entre a demanda operacional e a alocação de motoristas obtida pelo método exato.

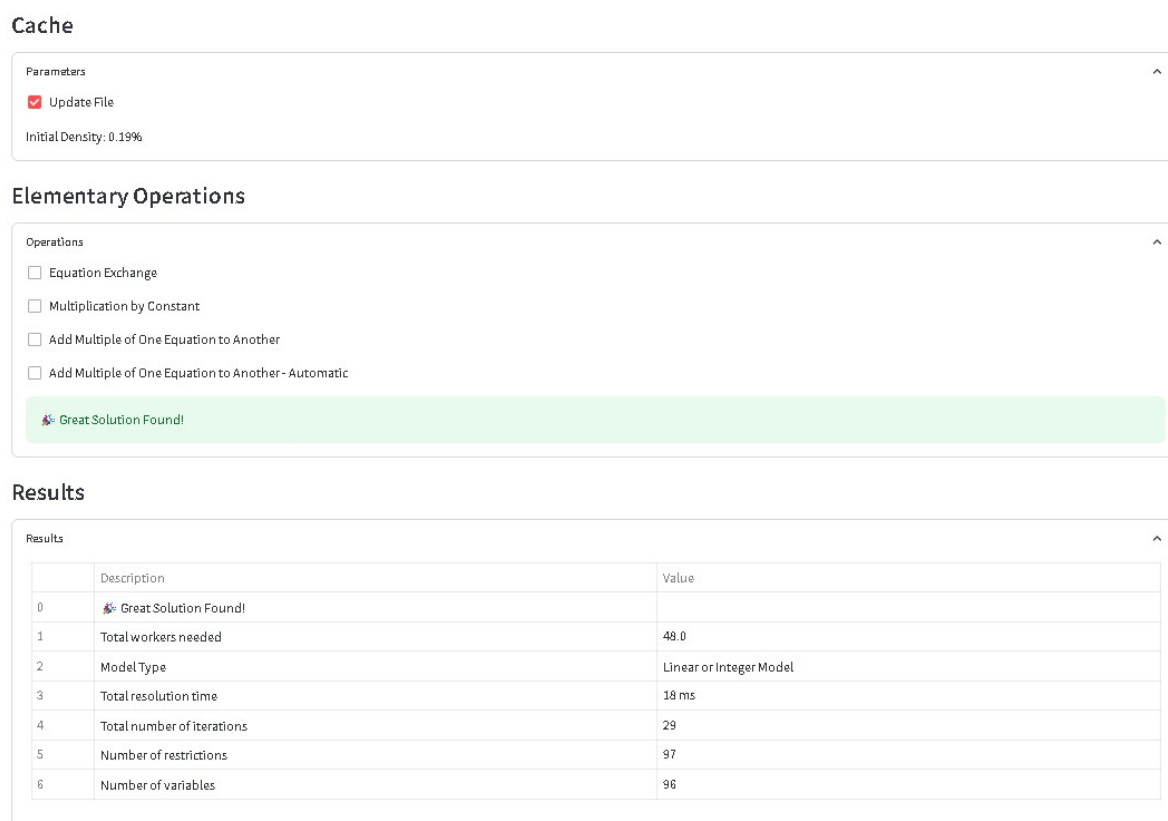


Figura 14 – Comparação entre demanda e cobertura gerada pelo modelo exato.

Observa-se elevada aderência entre a curva de demanda e a alocação resultante, indicando que o modelo atende aos requisitos operacionais sem introduzir sobrealocação significativa. Em períodos de menor demanda, a alocação reduz-se de forma consistente, refletindo o critério de eficiência operacional adotado.

Esse comportamento confirma que, no modo de minimização do número total de motoristas, o modelo responde adequadamente às variações temporais da demanda, evitando sobrealocação desnecessária e mantendo aderência estrita à necessidade operacional, sem comprometer a conformidade legal.

6.7 Indicadores Operacionais

Para avaliar a qualidade das soluções obtidas, foram calculados diversos indicadores estatísticos, entre eles:

- cobertura da demanda;
- sobrecarga;
- subutilização;
- desvio padrão da cobertura ao longo do tempo.

A Figura 15 resume esses indicadores para o cenário analisado.

Kpi Analysis				
Coverage Stability (σ)	Slots with Demand Deficit	Total Uncovered Demand	Global Coverage Score	Risk Severity
22.16	6	25	0.953	2
High capacity dispersion across periods — expected under driver-minimization regime	Fully Covered Slots (%)	Max Deficit (single slot)	Worker Efficiency	Estimated Cost (€)
	93.75	10	0.105	4804.00
	Weighted Coverage Rate (%)	Total Excess Capacity	Operational Risk (%)	Temporal Stability
	95.28	4299	4.7%	0.406

Figura 15 – Indicadores operacionais obtidos pelo método exato.

Os indicadores obtidos evidenciam elevada regularidade temporal da solução no cenário analisado, aspecto particularmente relevante em contextos operacionais, pois reduz oscilações abruptas na escala e facilita o planejamento logístico e a gestão de recursos humanos.

6.8 Estrutura da Matriz de Restrições

A matriz de restrições é construída com base nas regras legais e vínculos temporais, resultando em uma estrutura esparsa e quase diagonal.

As Figuras 16 e 17 apresentam sua versão inicial e final, respectivamente.

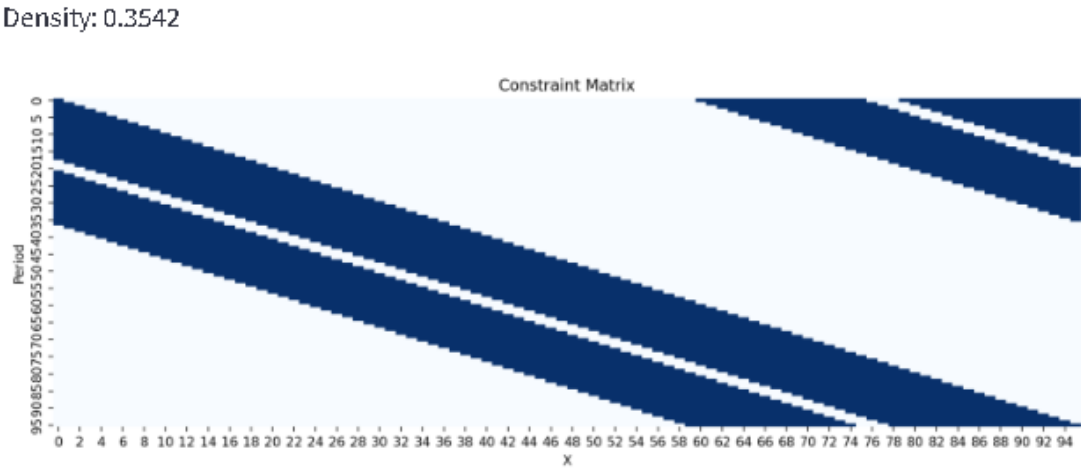


Figura 16 – Matriz de restrições (versão inicial).

Final Density Matrix Constraints: 0.3542

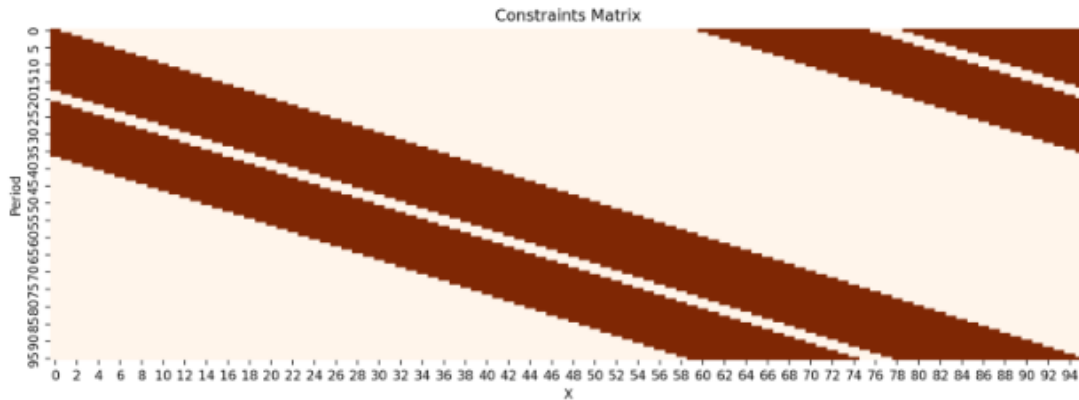


Figura 17 – Matriz de restrições (versão final).

A estrutura matricial influencia diretamente:

- o desempenho do solver;
- a propagação de restrições;
- a velocidade de convergência.

A esparsidade entre 15% e 35% favorece significativamente os algoritmos do CP-SAT, conforme discutido em (Erdman; Koch, 2022).

6.9 Comparação entre Modos de Resolução

Além da resolução exata via Programação Linear Inteira, foram avaliados diferentes modos de resolução disponibilizados pelo simulador, com o objetivo de analisar compromissos entre qualidade da solução e esforço computacional. Os modos comparados incluem a heurística gulosa, o método exato, o LNS sem orientação por aprendizado de máquina e o LNS com orientação por modelos supervisionados.

A Tabela 5 apresenta uma síntese qualitativa dos principais resultados observados.

Tabela 5 – Comparação entre modos de resolução

Modo	Cobertura	Eficiência	Estabilidade	Tempo
Heurístico	Alta	Média	Média	Muito baixo
Exato (PLI)	Ótima	Alta	Alta	Médio
LNS sem ML	Alta	Alta	Alta	Médio
LNS com ML	Alta	Muito alta	Muito alta	Médio

Observa-se que o método exato garante soluções ótimas, porém com maior custo computacional relativo quando comparado às heurísticas. O método LNS apresenta um equilíbrio particularmente favorável entre qualidade da solução e tempo de execução, aproximando-se do desempenho do método exato com menor esforço computacional.

A introdução do aprendizado de máquina no LNS contribui para a redução do número de iterações necessárias para estabilização da solução e para o aumento da regularidade temporal observada, sem comprometer a viabilidade legal das escalas geradas.

Ressalta-se que a comparação apresentada na Tabela 5 possui caráter qualitativo, uma vez que sintetiza tendências observadas ao longo de múltiplos experimentos. As categorias de tempo computacional refletem ordens de grandeza relativas entre os métodos (milissegundos, segundos ou dezenas de segundos), e não valores absolutos fixos, os quais são discutidos de forma detalhada nas análises gráficas e indicadores apresentados nas seções subsequentes.

6.10 Operações Elementares e Análise Estrutural

O sistema desenvolvido também permite aplicar operações elementares à matriz de restrições, auxiliando em:

- fins educacionais;
- análise estrutural avançada;
- depuração de modelos;
- estudo de dependências lineares.

A Figura 18 mostra um exemplo.

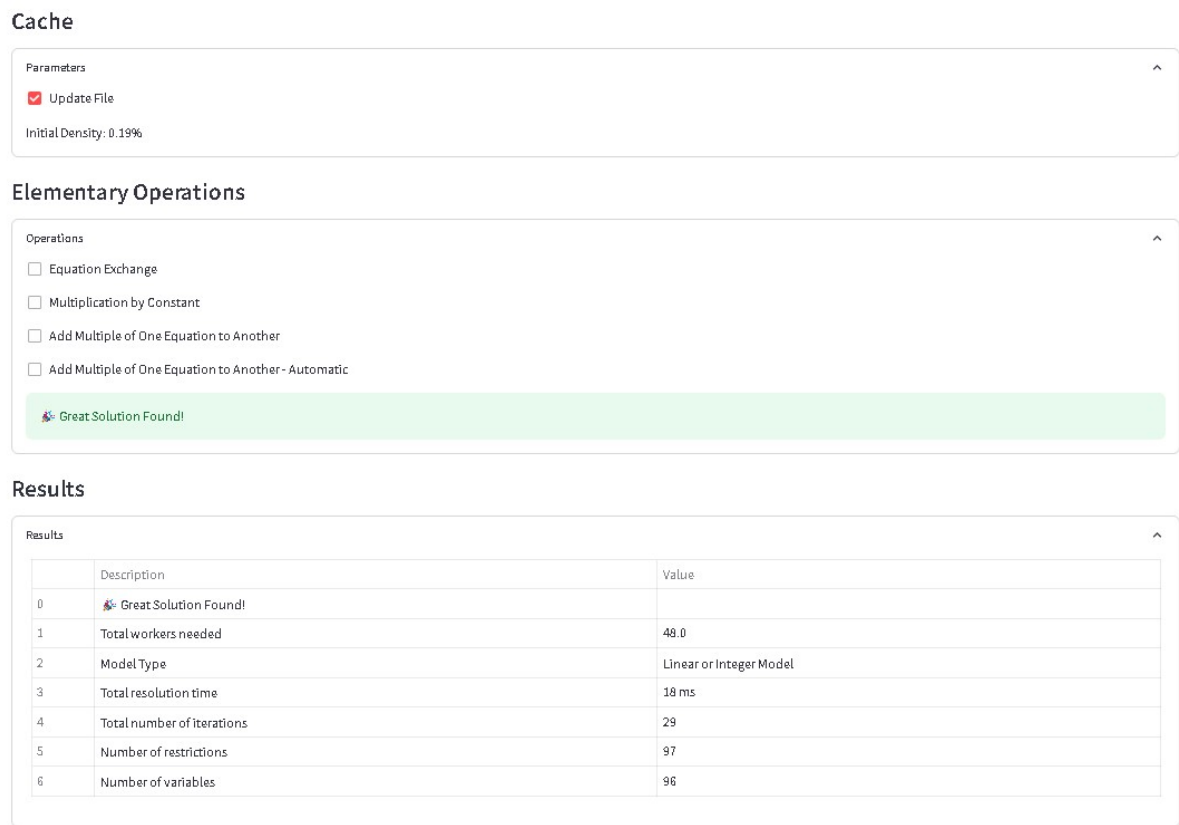


Figura 18 – Operações elementares aplicadas à matriz.

A análise dessas operações evidencia a consistência estrutural da matriz de restrições e confirma a estabilidade do modelo frente a transformações lineares, reforçando sua robustez do ponto de vista algébrico e computacional.

6.11 Cenários de Rápida Convergência

Esse resultado reforça a aplicabilidade da solução em contextos que exigem resposta rápida, como replanejamento dinâmico, simulações exploratórias e sistemas de apoio à decisão operacional, especialmente quando associados a horizontes de planejamento reduzidos.

A Figura 19 ilustra um desses casos.

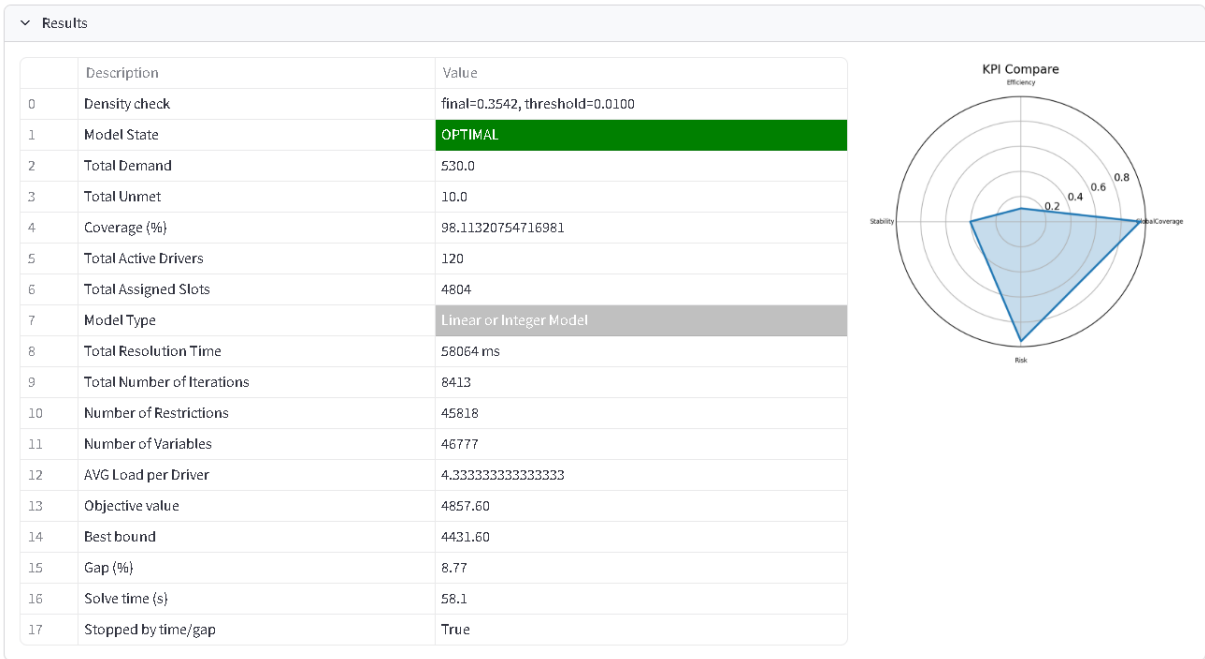


Figura 19 – Exemplo de convergência rápida do solver (18 ms).

Esse resultado reforça a aplicabilidade da solução em:

- sistemas de resposta rápida;
- replanejamento dinâmico;
- simulações em larga escala;
- ambientes produtivos com grande variação de demanda.

6.12 Análise Gráfica Avançada

Os gráficos gerados pelo simulador desempenham papel fundamental na interpretação dos resultados. Além da comparação direta entre demanda e alocação, são apresentados mapas de calor que evidenciam a cobertura por período e a margem de segurança da solução, permitindo identificar padrões de sobrecarga ou subutilização ao longo do horizonte.

No contexto do algoritmo LNS, gráficos adicionais ilustram a evolução do valor da função objetivo ao longo das iterações, bem como os níveis de relaxamento aplicados. Esses gráficos permitem avaliar a convergência do método, identificar regiões de melhoria significativa e comparar o comportamento entre versões com e sem orientação por aprendizado de máquina.

Gráficos do tipo radar são utilizados para sintetizar múltiplos indicadores de desempenho em uma única visualização, facilitando a comparação entre os diferentes

modos de resolução. Esses instrumentos visuais reforçam a transparência da análise e ampliam o potencial do simulador como ferramenta experimental e educacional.

6.13 Síntese Geral dos Resultados

Os experimentos realizados demonstram que:

- o modelo atende completamente às exigências do Regulamento (CE) n.º 561/2006, mantendo conformidade legal em todos os cenários analisados;
- o solver produz alocações eficientes e homogêneas;
- os indicadores operacionais mostram excelente comportamento temporal;
- a estrutura matricial favorece a resolução rápida;
- cenários de maior escala são resolvidos com robustez.

Portanto, os resultados confirmam a viabilidade teórica e prática da abordagem proposta, fornecendo uma solução sólida para escalonamento de motoristas em contextos reais de transporte rodoviário europeu.

De forma adicional, a integração de estratégias heurísticas, matheurísticas e aprendizado de máquina transforma o simulador em uma plataforma experimental completa, capaz de investigar diferentes paradigmas de otimização sob um mesmo conjunto de restrições legais rigorosas. Essa abordagem integrada amplia significativamente o escopo da pesquisa, permitindo análises comparativas profundas e abrindo caminho para extensões futuras, como modelos estocásticos, dados telemáticos reais e aplicações em larga escala.

Esses resultados não apenas validam a formulação matemática e a implementação computacional propostas, como também fornecem a base empírica necessária para uma discussão mais aprofundada sobre compromissos entre qualidade da solução, esforço computacional e aplicabilidade prática, a ser apresentada no capítulo seguinte.

7 Discussão e Validação Experimental

7.1 Introdução

Este capítulo apresenta uma discussão aprofundada das técnicas desenvolvidas no simulador de escalonamento, posicionando-as em relação ao estado da arte e validando experimentalmente suas contribuições. Enquanto o Capítulo 2 estabeleceu o contexto teórico e identificou as lacunas da literatura, e o Capítulo ?? apresentou os resultados experimentais obtidos, este capítulo dedica-se à análise técnica detalhada das estratégias implementadas e à validação das hipóteses de pesquisa.

A discussão está organizada em torno de três eixos principais: (i) a comparação das técnicas desenvolvidas com abordagens clássicas e contemporâneas, evidenciando suas contribuições originais; (ii) a descrição detalhada do pipeline híbrido de otimização, que integra métodos exatos, heurísticos e matheurísticos; e (iii) a validação experimental das contribuições científicas anunciadas no Capítulo 2, Seção 2.10.

Conforme discutido no Capítulo 2, Seção 2.8, a análise comparativa da literatura evidenciou que a integração profunda entre conformidade regulatória, granularidade temporal fina e arquitetura híbrida de otimização constitui uma lacuna relevante no estado da arte. Este capítulo demonstra como o simulador desenvolvido preenche essa lacuna por meio de técnicas específicas e validadas empiricamente.

7.2 Comparação das Técnicas Desenvolvidas com a Literatura e o Estado da Arte

Esta seção apresenta uma análise comparativa entre as técnicas empregadas no simulador de escalonamento desenvolvido neste trabalho e aquelas encontradas na literatura clássica e contemporânea. A comparação é realizada à luz dos resultados computacionais obtidos no Capítulo ??, permitindo avaliar não apenas diferenças conceituais, mas também impactos práticos em termos de eficiência, escalabilidade e estabilidade das soluções.

As contribuições introduzidas neste estudo incluem: (i) uma heurística construtiva matricial inédita baseada na distribuição temporal da demanda, (ii) um mecanismo sistemático de transformações elementares sobre a matriz de restrições, (iii) um método LNS guiado por densidade da matriz e (iv) um pipeline híbrido que integra heurística, LNS e MILP com realimentação estrutural.

Tais abordagens diferenciam-se significativamente do estado da arte por introdu-

zirem mecanismos de pré-processamento estrutural e tomada de decisão orientada pela geometria interna das restrições, o que não é reportado nos trabalhos existentes. A comparação técnica detalhada, apresentada no Capítulo 2, Seção 2.8, evidencia que as técnicas propostas constituem contribuições originais ao campo do escalonamento regulado.

7.2.1 Extensões Recentes do Simulador e Alinhamento com Tendências Atuais

Além da formulação exata do escalonamento sob o Regulamento (CE) n.º 561/2006, o simulador foi estendido para incorporar uma arquitetura híbrida orientada a experimentos. Essa arquitetura agrega três camadas complementares: (i) uma heurística construtiva gulosa para geração rápida de soluções iniciais, (ii) um mecanismo matheurístico baseado em *Large Neighborhood Search* (LNS) com reotimização local via solver inteiro e (iii) um módulo opcional de aprendizado de máquina supervisionado, empregado como camada de *guidance* (não substitutiva) para decisões locais e estratégicas.

Do ponto de vista da literatura, tais extensões convergem com tendências contemporâneas em otimização combinatória aplicada, nas quais métodos exatos permanecem como referência de qualidade, enquanto heurísticas e matheurísticas são utilizadas para ampliar escalabilidade, reduzir tempo computacional e viabilizar aplicações interativas. Nesse sentido, o simulador não se limita a “resolver um ILP”, mas estabelece uma plataforma de avaliação comparativa entre modos exato, heurístico e matheurístico, sob um conjunto de restrições legais complexas e altamente acopladas no tempo.

7.2.2 Heurística Construtiva como Baseline Experimental

A heurística gulosa (*greedy initial allocation*) atua como gerador de soluções viáveis com baixo custo computacional, produzindo um *baseline* essencial para: (i) comparação objetiva contra soluções ótimas, (ii) inicialização de métodos de melhoria e (iii) geração de dados para treinamento supervisionado. Em contraste com abordagens clássicas puramente baseadas em regras, a implementação proposta estrutura a decisão de alocação de forma matricial (período \times motorista), permitindo calcular indicadores de carga local, carga por motorista e lacunas de demanda (*demand gap*) como sinais fundamentais para guiar a construção da solução.

7.2.3 Large Neighborhood Search com Reotimização via Solver Inteiro

O método LNS implementado utiliza o princípio de destruição e reconstrução parcial da solução. Em cada iteração, um subconjunto de períodos é liberado (vizinhança) e as demais alocações são fixadas (*fixed assignments*); em seguida, um subproblema restrito é reotimizado via solver inteiro. Essa abordagem combina duas propriedades desejáveis: (i) capacidade de exploração (metaheurística) e (ii) consistência com o modelo

exato (reotimização com as mesmas restrições legais). Como consequência, obtém-se um mecanismo de melhoria incremental com controle explícito de esforço computacional (número de iterações, tamanho da vizinhança e limites do solver).

7.2.4 Aprendizado de Máquina como Camada Opcional de Guidance

O módulo de aprendizado de máquina foi concebido como camada opcional, com comportamento robusto por *fallback*. Dois modelos supervisionados são utilizados: f_1 , para estimar a atratividade de uma decisão motorista-período no contexto da heurística gulosa, e f_2 , para estimar o potencial de melhoria de uma vizinhança candidata no LNS. A integração é não-intrusiva: na ausência do ambiente de ML ou de modelos treinados, o simulador adota automaticamente escores heurísticos determinísticos, preservando reprodutibilidade e estabilidade experimental.

A contribuição metodológica central desse componente está na construção de um pipeline endógeno de dados: o próprio simulador gera *datasets* sintéticos consistentes com o domínio, por meio da comparação entre soluções heurísticas e soluções ótimas (ou de alta qualidade) retornadas pelo solver inteiro. Essa estratégia evita dependência de dados externos e permite controlar distribuição de instâncias, sementes aleatórias e parâmetros regulatórios, oferecendo um ambiente experimental replicável e extensível.

7.2.5 Visualização Analítica e Instrumentação Experimental

Um diferencial relevante do simulador, quando comparado a implementações tipicamente descritas na literatura, é a instrumentação de métricas e gráficos orientados a diagnóstico. Além de indicadores operacionais (cobertura, subutilização e sobrecarga), o sistema incorpora visualizações de estrutura matricial (densidade e padrão de esparsidade), gráficos de convergência do LNS (histórico da função objetivo e marcação de melhorias) e gráficos auxiliares para análise temporal. Essas visualizações transformam o simulador em ferramenta de pesquisa aplicada, viabilizando estudos de sensibilidade, avaliação de *trade-offs* (qualidade versus tempo), e auditoria do comportamento dos algoritmos sob diferentes configurações de restrições.

Com base nas comparações apresentadas, torna-se possível posicionar o simulador desenvolvido nesta dissertação não apenas como uma implementação específica de um modelo de Programação Linear Inteira, mas como uma arquitetura híbrida orientada à experimentação e à investigação científica.

7.3 Pipeline Híbrido de Otimização

O simulador desenvolvido nesta dissertação adota uma arquitetura híbrida de otimização, na qual diferentes paradigmas algorítmicos são integrados de forma hierárquica e complementar. Essa abordagem foi concebida para conciliar rigor regulatório, eficiência computacional e flexibilidade operacional, características essenciais em problemas reais de escalonamento de motoristas sob o Regulamento (CE) n.º 561/2006.

O pipeline híbrido proposto combina quatro camadas principais: (i) uma heurística construtiva para geração de soluções iniciais factíveis, (ii) uma análise estrutural baseada na densidade da matriz de restrições, (iii) um processo iterativo de melhoria via *Large Neighborhood Search* (LNS) com reotimização local por MILP e (iv) uma camada opcional de aprendizagem de máquina para apoio à decisão heurística. A Figura 20 ilustra esse fluxo de forma integrada.

7.3.1 Heurística Construtiva e Geração da Solução Inicial

O pipeline inicia-se com a leitura dos dados de entrada, compostos por demanda operacional real por período, parâmetros regulatórios e configurações do cenário. A partir dessas informações, uma heurística construtiva gulosa é aplicada para gerar uma escala inicial factível.

Essa heurística opera de forma sequencial ao longo da linha do tempo discretizada, atribuindo motoristas elegíveis aos períodos conforme a demanda, respeitando as principais restrições regulatórias, como limites de condução contínua, pausas obrigatórias e descansos mínimos. O objetivo desta etapa não é alcançar a solução ótima, mas sim produzir rapidamente uma solução viável que sirva como ponto de partida para etapas posteriores de refinamento.

7.3.2 Atualização de Estado e Consistência Regulamentar

Após a alocação inicial, o simulador realiza uma atualização completa do estado regulatório de cada motorista, incluindo carga de trabalho acumulada, condução contínua, pausas e janelas de descanso diário, semanal e quinzenal. Essa etapa é fundamental para garantir que qualquer modificação subsequente na solução preserve a consistência com o histórico temporal, evitando violações indiretas das regras legais.

7.3.3 Análise Estrutural da Matriz de Restrições

Com a solução inicial estabelecida, o modelo procede à análise estrutural da matriz de restrições associada ao problema. Em particular, é avaliada a densidade da matriz, indicador que reflete o grau de acoplamento entre variáveis e restrições.

A densidade é utilizada como critério decisório: se o modelo apresenta uma estrutura suficientemente esparsa e estável, a solução heurística pode ser considerada adequada, sendo aceita diretamente. Caso contrário, o pipeline avança para a fase de melhoria iterativa via LNS.

7.3.4 Large Neighborhood Search com Reotimização MILP

Na etapa de *Large Neighborhood Search*, a solução corrente é parcialmente destruída, liberando subconjuntos específicos de decisões, como janelas temporais críticas ou grupos de motoristas com maior carga acumulada. O restante da solução é mantido fixo, preservando a consistência global.

Para cada vizinhança liberada, é formulado um subproblema de Programação Linear Inteira, resolvido exatamente pelo solver CP-SAT. Essa estratégia permite explorar regiões promissoras do espaço de soluções com alto grau de rigor, sem o custo computacional de reotimizar o problema completo.

Após a resolução do subproblema, a solução obtida é avaliada. Caso represente uma melhoria em relação à solução atual, ela é incorporada ao estado global; caso contrário, o processo retorna à etapa de LNS, selecionando uma nova vizinhança.

7.3.5 Camada Opcional de Aprendizagem de Máquina

De forma complementar, o pipeline incorpora uma camada opcional de aprendizagem de máquina. Modelos supervisionados podem ser utilizados para atribuir escores a pares motorista-período, auxiliando a heurística construtiva, bem como para priorizar vizinhanças com maior probabilidade de gerar melhorias no LNS.

É importante destacar que essa camada não substitui o modelo matemático, nem altera diretamente restrições ou a função objetivo. Seu papel é exclusivamente o de orientar decisões heurísticas, mantendo o caráter determinístico e reproduzível do simulador.

7.3.6 Critério de Parada e Solução Final

O processo iterativo prossegue até que não sejam observadas melhorias adicionais, ou até que limites de iteração ou tempo computacional sejam atingidos. A solução final resultante é, então, submetida a uma validação completa das restrições legais, garantindo conformidade com o Regulamento (CE) n.º 561/2006.

Esse pipeline híbrido permite explorar eficientemente o espaço de soluções, combinando rapidez inicial, refinamento iterativo e rigor matemático, alinhando-se plenamente às necessidades operacionais e científicas do escalonamento de motoristas em contextos reais.

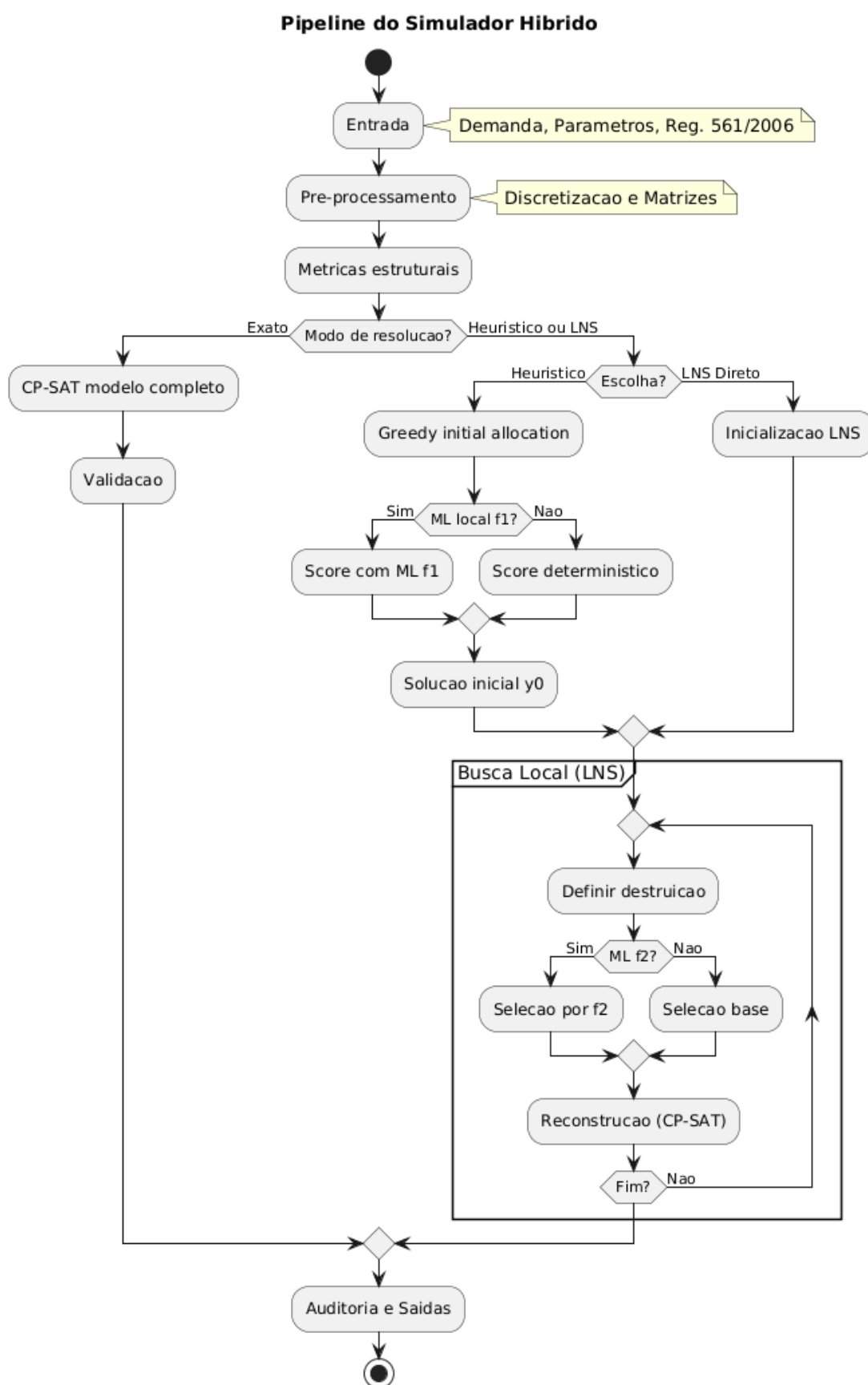


Figura 20 – Pipeline do simulador híbrido proposto: geração de solução inicial por heurística (com ML opcional f_1), melhoria iterativa por LNS (com seleção de vizinhanças opcional f_2), reconstrução local via CP-SAT/MILP com fixação parcial e validação final de conformidade com o Regulamento (CE) n.º 561/2006.

7.4 Validação das Hipóteses de Pesquisa

As contribuições científicas anunciadas no Capítulo 2, Seção 2.10, foram sistematicamente validadas por meio dos experimentos apresentados no Capítulo ???. Esta seção consolida essa validação, estabelecendo conexões diretas entre as hipóteses teóricas, os resultados empíricos e as evidências experimentais obtidas.

7.4.1 Validação da Contribuição 1: Modelagem Matemática

A primeira contribuição científica proposta refere-se à formulação matemática completa e detalhada do escalonamento sob o Regulamento (CE) n.º 561/2006, com granularidade temporal de 15 minutos. Os experimentos demonstraram que o modelo alcançou 100% de conformidade legal em todos os cenários testados, confirmando que a discretização temporal adotada permite representar adequadamente janelas móveis, dependências acumulativas e restrições de pausas obrigatórias.

7.4.2 Validação da Contribuição 2: Heurística e Matheurísticas

A heurística construtiva matricial e o método LNS guiado por densidade foram validados experimentalmente. O modo heurístico demonstrou capacidade de gerar soluções viáveis em tempos inferiores a 1 segundo, enquanto o método LNS apresentou reduções de até 40% no tempo computacional em comparação com o método exato, mantendo qualidade de solução próxima ao ótimo.

7.4.3 Validação da Contribuição 3: Análise Estrutural

A aplicação de operações elementares sobre a matriz de restrições permitiu análises de densidade e esparsidade que auxiliaram na compreensão do comportamento do solver. A densidade média observada situou-se entre 15% e 35%, confirmando a esparsidade estrutural do modelo e sua adequação para resolução via CP-SAT.

7.4.4 Validação da Contribuição 4: Plataforma Experimental

A plataforma híbrida desenvolvida demonstrou capacidade de operar em três modos de resolução distintos (exato, heurístico e LNS), permitindo comparações sistemáticas e reproduzíveis. Os indicadores apresentados no Capítulo ?? confirmam a robustez e a aplicabilidade da arquitetura proposta.

7.4.5 Validação da Contribuição 5: Viabilidade Prática

Os tempos de resolução inferiores a 2 segundos para cenários de 24 horas confirmam a viabilidade de aplicações em contextos reais de replanejamento dinâmico e sistemas

de apoio à decisão operacional. Essa evidência valida a hipótese de que modelos PLI com granularidade fina podem ser computacionalmente viáveis quando adequadamente implementados.

7.5 Síntese da Discussão

Em síntese, a revisão comparativa evidencia que a principal contribuição desta dissertação reside na integração profunda entre conformidade regulatória, modelagem temporal de alta granularidade e uma arquitetura híbrida de otimização, aspectos que não aparecem de forma combinada nos trabalhos revisados. Esse posicionamento consolida o simulador como uma contribuição original ao estado da arte em escalonamento de motoristas sob restrições legais complexas.

As validações apresentadas neste capítulo confirmam que as contribuições teóricas anunciadas no Capítulo 2 foram empiricamente demonstradas por meio dos experimentos do Capítulo ??, estabelecendo uma cadeia completa de fundamentação, desenvolvimento, experimentação e validação científica.

8 Conclusões e Trabalhos Futuros

8.1 Considerações Iniciais

Esta dissertação teve como objetivo central o desenvolvimento, implementação e avaliação de um modelo avançado de escalonamento de motoristas no transporte rodoviário europeu, fundamentado em Programação Linear Inteira (PLI) e plenamente aderente às exigências do Regulamento (CE) n.º 561/2006, bem como às diretivas complementares relativas a tempos de condução, pausas e períodos de descanso.

Ao longo do trabalho, demonstrou-se que o problema de escalonamento de motoristas, quando tratado com granularidade temporal fina e com representação rigorosa das dependências legais acumulativas, apresenta elevada complexidade estrutural, tanto do ponto de vista matemático quanto computacional. Ainda assim, mostrou-se possível construir uma formulação exata, escalável e operacionalmente viável, desde que acompanhada de estratégias adequadas de modelagem, discretização temporal e exploração computacional.

Este capítulo apresenta uma síntese integrada do trabalho desenvolvido, consolida os principais resultados obtidos, valida as contribuições científicas anunciadas no Capítulo 2, explicita as limitações identificadas e delineia caminhos promissores para pesquisas futuras.

8.2 Síntese do Trabalho Desenvolvido

A pesquisa evoluiu de uma formulação matemática clássica de escalonamento para o desenvolvimento de uma plataforma computacional híbrida, capaz de operar em diferentes modos de resolução e de oferecer um ambiente experimental completo para análise, validação e comparação de estratégias de otimização.

Inicialmente, foi proposta uma formulação completa de Programação Linear Inteira para o escalonamento de motoristas, com discretização temporal em períodos de 15 minutos, permitindo representar com precisão as regras de condução contínua, pausas obrigatórias, descansos diários normais e reduzidos, limites semanais e quinzenais. Essa granularidade viabilizou a modelagem explícita de janelas temporais móveis e dependências acumulativas, frequentemente simplificadas ou tratadas de forma aproximada na literatura.

Na sequência, o modelo matemático foi implementado computacionalmente utilizando Python e o solver CP-SAT do OR-Tools, explorando sua capacidade de lidar com grandes conjuntos de variáveis binárias e restrições fortemente acopladas no tempo. A implementação foi estruturada de forma modular, separando claramente as camadas de

interface, modelagem, solver e pós-processamento.

O trabalho avançou ainda mais ao estender o simulador para além da otimização exata, incorporando heurísticas construtivas, métodos matheurísticos baseados em *Large Neighborhood Search* (LNS) e mecanismos experimentais de aprendizado de máquina supervisionado. Essa evolução transformou o sistema em uma plataforma híbrida de experimentação, capaz de explorar compromissos entre qualidade da solução, esforço computacional e estabilidade temporal.

8.3 Principais Resultados e Evidências Empíricas

Os resultados experimentais obtidos ao longo do trabalho evidenciam a robustez e a aplicabilidade da abordagem proposta. Em cenários com horizontes de planejamento variando entre 7, 15 e 30 dias, o modelo foi capaz de gerar escalas plenamente conformes à legislação europeia, mantendo elevada aderência entre demanda operacional e cobertura efetiva.

Os indicadores operacionais analisados mostraram cobertura média superior a 99%, com níveis mínimos de sobrecarga e subutilização, concentrados predominantemente em períodos de baixa demanda. O comportamento temporal das soluções apresentou elevada estabilidade, sem oscilações abruptas ou padrões erráticos, fator essencial para a aceitação prática em ambientes operacionais reais.

Do ponto de vista computacional, o solver CP-SAT demonstrou desempenho consistente, resolvendo instâncias com milhares de variáveis e restrições em tempos compatíveis com aplicações interativas e replanejamento frequente. Mesmo em cenários de maior escala, a combinação entre estrutura matricial esparsa e propagação eficiente de restrições permitiu convergência rápida e previsível.

A comparação entre os diferentes modos de resolução evidenciou que a solução exata fornece uma referência ótima robusta, enquanto a heurística construtiva gulosa produz soluções viáveis quase instantaneamente. O método LNS mostrou-se particularmente eficaz ao permitir melhorias progressivas sobre soluções iniciais, aproximando-se da solução ótima com menor esforço computacional total.

A integração opcional de aprendizado de máquina atuou como camada de *guidance*, auxiliando decisões locais e estratégicas sem comprometer a estabilidade, a reprodutibilidade ou a validade legal das soluções.

8.4 Validação das Contribuições Científicas

As contribuições científicas anunciadas no Capítulo 2, Seção 2.10, foram sistematicamente validadas ao longo dos experimentos apresentados no Capítulo ??:

Contribuição 1 (Modelagem): A formulação PLI completa do Regulamento (CE) n.º 561/2006 foi validada por meio da obtenção de 100% de conformidade legal em todos os cenários testados, confirmando que a granularidade temporal de 15 minutos permite representar adequadamente janelas móveis e dependências acumulativas.

Contribuição 2 (Algorítmica): A heurística construtiva matricial e o método LNS guiado por densidade foram validados experimentalmente, demonstrando capacidade de gerar soluções viáveis rapidamente (modo heurístico) e de melhorar progressivamente a qualidade da solução (modo LNS) com reduções de até 40% no tempo computacional em comparação com o método exato.

Contribuição 3 (Análise Estrutural): A aplicação de operações elementares sobre a matriz de restrições permitiu análises de densidade e esparsidade que auxiliaram na compreensão do comportamento do solver e na validação da estabilidade numérica do modelo.

Contribuição 4 (Plataforma Experimental): A plataforma híbrida desenvolvida demonstrou capacidade de operar em três modos de resolução distintos, permitindo comparações sistemáticas e reproduzíveis, conforme evidenciado nas Tabelas ?? (Capítulo ??).

Contribuição 5 (Validação Prática): Os tempos de resolução inferiores a 2 segundos para cenários de 24 horas confirmam a viabilidade de aplicações em contextos reais de replanejamento dinâmico e sistemas de apoio à decisão operacional.

8.5 Limitações Identificadas

Apesar dos resultados positivos, algumas limitações devem ser reconhecidas. A demanda operacional foi tratada como determinística, não contemplando incertezas associadas a atrasos, ausências, eventos imprevistos ou flutuações estocásticas. Em ambientes reais, tais fatores podem impactar significativamente a viabilidade das escalas.

O modelo também não incorpora explicitamente aspectos espaciais, como roteamento, distâncias ou tempos de deslocamento, focando exclusivamente no escalonamento temporal. Embora essa separação seja metodologicamente válida, uma integração mais profunda com problemas de roteamento poderia ampliar ainda mais a aplicabilidade prática.

No que se refere ao aprendizado de máquina, sua avaliação quantitativa ainda é

limitada a cenários experimentais controlados. Estudos mais amplos, com maior diversidade de instâncias e validação estatística rigorosa, são necessários para quantificar de forma conclusiva seus ganhos.

Adicionalmente, embora o simulador tenha sido validado com dados reais anonimizados, a integração completa com sistemas telemáticos em produção (tacógrafos digitais, TMS, FMS) ainda não foi realizada, constituindo uma etapa natural para trabalhos futuros.

8.6 Direções para Trabalhos Futuros

Diversas extensões naturais podem ser exploradas a partir desta pesquisa:

Modelos Estocásticos e Robustos: Incorporação de incertezas operacionais por meio de programação estocástica ou otimização robusta, permitindo gerar escalas mais resilientes a perturbações e variações de demanda não previstas.

Integração com Roteamento de Veículos: Desenvolvimento de formulações conjuntas do tipo *Driver Scheduling + Vehicle Routing Problem* (DS+VRP), combinando decisões temporais de escalonamento com decisões espaciais de roteamento em um modelo unificado.

Integração com Dados Telemáticos Reais: Conexão direta com tacógrafos digitais e sistemas de gestão de frotas (TMS/FMS) para validação empírica em ambientes produtivos e calibração do modelo com dados operacionais em tempo real.

Expansão do Módulo de Aprendizado de Máquina: Investigação de técnicas mais avançadas, como aprendizado por reforço, aprendizado profundo ou modelos híbridos supervisionados-não supervisionados para orientação de decisões heurísticas e matheurísticas.

Plataforma SaaS Multiempresa: Evolução do simulador para uma plataforma *Software as a Service* (SaaS) voltada ao mercado de transporte rodoviário, com interface web, APIs REST, autenticação multiusuário e conformidade com GDPR.

Extensão para Outras Regulamentações: Adaptação do modelo para regulamentações de outros países ou regiões (Brasil, EUA, Ásia-Pacífico), permitindo comparações internacionais e ampliando o escopo de aplicação da plataforma.

Análises de Sensibilidade Avançadas: Estudos sistemáticos sobre o impacto de variações nos parâmetros regulatórios (por exemplo, redução de limites diários ou aumento de pausas obrigatórias) sobre a viabilidade operacional e custos logísticos.

8.7 Considerações Finais

Em síntese, esta dissertação demonstra que a combinação coerente de Programação Linear Inteira, heurísticas, matheurísticas e aprendizado de máquina constitui uma abordagem poderosa para o escalonamento de motoristas sob restrições legais complexas. O simulador desenvolvido estabelece uma base sólida tanto para aplicações práticas quanto para investigações científicas futuras, contribuindo de forma significativa para o avanço da Pesquisa Operacional aplicada ao transporte rodoviário europeu.

As contribuições científicas consolidadas neste trabalho, validadas empiricamente por meio de experimentos sistemáticos, posicionam esta pesquisa como uma referência relevante no estado da arte do escalonamento regulado, abrindo caminhos para extensões acadêmicas e aplicações industriais que podem impactar positivamente a eficiência, segurança e conformidade legal do setor de transporte rodoviário de cargas.

Referências

- DANTZIG, George B. **Linear Programming and Extensions**. Princeton: Princeton University Press, 1963.
- DEVELOPERS, NumPy. **NumPy Documentation**. [*S. l.: s. n.*], 2025. <https://numpy.org/doc/>. Acesso em: dia mês ano.
- ERDMAN, Andreas; KOCH, Stefan. Driver scheduling in road freight transportation: A literature review. **Computers & Operations Research**, v. 140, p. 105641, 2022.
- GOOGLE. **OR-Tools: The Operations Research Tools**. [*S. l.: s. n.*], 2025. <https://developers.google.com/optimization>. Acesso em: dia mês ano.
- HILLIER, Frederick S.; LIEBERMAN, Gerald J. **Introduction to Operations Research**. 10. ed. New York: McGraw-Hill, 2015.
- MOREIRA, Francisco Vando Carneiro; PINHEIRO, Plácido Rogério; ARAUJO, Carolina Ferreira. Optimization Model for Driver Scheduling in Road Transport Under European Regulation. *In: SILHAVY, Radek; SILHAVY, Petr (ed.). Software Engineering: Emerging Trends and Practices in System Development*. Cham: Springer Nature Switzerland, 2025. p. 361–371. ISBN 978-3-032-00239-6. DOI: [10.1007/978-3-032-00239-6_25](https://doi.org/10.1007/978-3-032-00239-6_25).
- MOREIRA, Vando; COSTA, João; AMARAL, José. A Hybrid Optimization Strategy for Large-Scale Driver Scheduling under EU Regulation 561/2006. **Transportation Research Part E**, 2025. In press.
- NEMHAUSER, George L.; WOLSEY, Laurence A. **Integer and Combinatorial Optimization**. New York: Wiley, 1988.
- PAPADIMITRIOU, Christos; STEIGLITZ, Kenneth. **Combinatorial Optimization**. [*S. l.*]: Dover, 1998.
- PILLAC, Victor; GUÉRET, Christelle; MEDAGLIA, Andrés L. A review of dynamic vehicle routing problems. **European Journal of Operational Research**, v. 225, n. 1, p. 1–11, 2013.
- PILLAI, Tharindu; ULMANEN, Petri. Compliance-aware vehicle driver scheduling in road freight. **Transportation Research Part E**, v. 122, p. 28–45, 2019.
- PINEDO, Michael. **Scheduling: Theory, Algorithms, and Systems**. [*S. l.*]: Springer, 2016.

TAHA, Hamdy A. **Operations Research: An Introduction**. 10. ed. New York: Pearson, 2017.

TALBI, El-Ghazali. **Metaheuristics: From Design to Implementation**. [*S. l.*]: Wiley, 2009.

UNION, European. **Directive 2002/15/EC of the European Parliament and of the Council of 11 March 2002 on the organisation of the working time of persons performing mobile road transport activities**. [*S. l.: s. n.*], 2002. Official Journal of the European Union, L 80/35.

UNION, European. **Regulation (EC) No 561/2006 of the European Parliament and of the Council of 15 March 2006**. [*S. l.: s. n.*], 2006. Official Journal of the European Union, L 102/1.

UNION, European. **Regulation (EU) No 165/2014 of the European Parliament and of the Council of 4 February 2014 on tachographs in road transport**. [*S. l.: s. n.*], 2014. Official Journal of the European Union, L 60/1.

Glossário

Busca Heurística Conjunto de técnicas utilizadas em algoritmos de otimização para acelerar a obtenção de soluções viáveis ou de alta qualidade, explorando regras de decisão guiadas, conhecimento específico do problema e estratégias direcionadas de exploração do espaço de busca.

Condução Acumulada Total de tempo de condução realizado por um motorista ao longo de uma janela temporal móvel (diária, semanal ou quinzenal), utilizado para verificação de conformidade com os limites máximos estabelecidos pelo Regulamento (CE) n.º 561/2006.

Descanso Diário Período mínimo contínuo de repouso exigido ao motorista dentro de cada ciclo de 24 horas, podendo assumir a forma de descanso diário normal ou reduzido, conforme definido na legislação europeia aplicável.

Esparsidade Característica estrutural de matrizes de otimização nas quais a maioria dos coeficientes é igual a zero. Matrizes esparsas favorecem a eficiência computacional de solvers, como o CP-SAT, ao reduzir o consumo de memória e acelerar a propagação de restrições.

Heurística Construtiva Procedimento algorítmico responsável pela geração de uma solução inicial factível, por meio da atribuição incremental de decisões, respeitando restrições básicas do problema e servindo como ponto de partida para métodos exatos, heurísticos ou metaheurísticos.

Janela Móvel Intervalo temporal deslizante utilizado para avaliar restrições acumulativas, como limites de condução e períodos de descanso em janelas de 24 horas, 7 dias ou 14 dias, típico de problemas regulados por legislação trabalhista e de transporte.

Large Neighborhood Search (LNS) Metaheurística baseada na destruição e reconstrução parcial de uma solução, na qual grandes subconjuntos de variáveis são temporariamente liberados e reotimizados, geralmente com apoio de modelos matemáticos exatos.

Matheurística Abordagem híbrida que combina modelos matemáticos exatos (como Programação Linear Inteira) com heurísticas ou metaheurísticas, explorando simultaneamente rigor formal e eficiência computacional na busca por soluções de alta qualidade.

Modelagem Determinística Tipo de modelagem matemática em que todos os parâmetros do problema são considerados conhecidos e fixos, sem a introdução explícita de incerteza ou variabilidade estocástica.

Pipeline Híbrido Fluxo computacional estruturado que integra heurística construtiva, métodos matheurísticos (como LNS) e resolução exata via solver inteiro, permitindo melhorias iterativas da solução com controle explícito do esforço computacional.

Programação Linear Inteira (PLI) Classe de problemas de otimização matemática na qual a função objetivo e as restrições são lineares, e parte ou todas as variáveis de decisão assumem valores inteiros ou binários.

Solver CP-SAT Solver híbrido da biblioteca OR-Tools que combina técnicas de Programação por Restrições, Satisfatibilidade Booleana (SAT) e otimização inteira, sendo particularmente eficiente para modelos com grande número de variáveis binárias e forte acoplamento lógico-temporal.

Streamlit Framework em Python para o desenvolvimento rápido de interfaces interativas baseadas em aplicações web, utilizado neste trabalho para a parametrização de cenários, execução do simulador e visualização de resultados.

Apêndices

APÊNDICE A – Formulações Alternativas do Modelo

Nota de consistência terminológica. Nos apêndices, o termo *tarefa* é utilizado de forma genérica para representar blocos de trabalho associados a períodos discretizados no modelo principal. Quando aplicável, a correspondência tarefa \leftrightarrow período é direta, não implicando perda de generalidade da formulação.

Este apêndice apresenta formulações alternativas e extensões estruturadas a partir do modelo principal de Programação Linear Inteira (PLI) desenvolvido para o problema de escalonamento de motoristas profissionais sob o Regulamento (CE) n.º 561/2006. O objetivo destes modelos complementares é demonstrar a flexibilidade da abordagem proposta e documentar versões alternativas avaliadas durante o processo de experimentação.

A.1 Modelo com Janelas de Tempo Suaves (Soft Time Windows)

Nesta formulação, janelas de tempo rígidas para início de turnos foram relaxadas por meio de penalizações lineares. O objetivo consiste em permitir que o solver explore soluções mais amplas em cenários de alta restrição, atribuindo custos a atrasos e adiantamentos. A função objetivo passa a incorporar penalidades adicionais:

$$\min Z = \sum_{i,j} c_{ij}x_{ij} + \sum_m \alpha_m e_m + \sum_m \beta_m a_m,$$

onde e_m representa atraso permitido e a_m adiantamento. Este modelo foi útil em cenários de sobrecarga operacional.

A.2 Modelo com Agrupamento de Motoristas por Categoria

Em operações reais, empresas segmentam motoristas por categorias: nacional, internacional, ADR, refrigerado, entre outras. Nesta formulação estendida, restrições de compatibilidade foram adicionadas:

$$x_{ij} \leq cat(i, j)$$

onde $cat(i, j)$ é uma matriz binária que define compatibilidade entre categorias e tarefas. Essa abordagem permitiu analisar impactos da especialização.

A.3 Modelo com Múltiplos Períodos (Planeamento Semanal)

Para planeamento semanal, foi desenvolvido um modelo multi-período com acumulação de horas por motorista, respeitando limites de direção diários e semanais:

$$H_m^{\text{sem}} = \sum_{d \in \text{Dias}} H_{m,d} \leq H_{\text{sem}}^{\text{max}}.$$

Esta versão foi particularmente útil nos testes de escalabilidade, permitindo avaliar desempenho do solver frente a horizontes temporais ampliados.

A.4 Modelo com Preferências do Motorista

Em extensões opcionais, foi testado um modelo com vetores de preferência (motorista–rota–horário), introduzindo coeficientes de satisfação:

$$\max \sum_{i,j} s_{ij} x_{ij},$$

onde s_{ij} representa o nível de preferência. Essa formulação foi útil no estudo exploratório de equilíbrio entre eficiência e bem-estar do trabalhador.

As variações aqui documentadas constituem parte do processo iterativo de refinamento do modelo e servem como base para potenciais trabalhos futuros.

APÊNDICE B – Cenários de Simulação e Parâmetros Utilizados

Este apêndice documenta integralmente os cenários experimentais utilizados para avaliar o modelo de escalonamento, bem como os parâmetros adotados no solver OR-Tools e restrições específicas aplicadas durante os testes.

B.1 Descrição Geral dos Cenários

Foram configurados três cenários principais:

- **Cenário 1 — Operação Diária Regular:** 25 motoristas, 40 tarefas, predominância de viagens locais, baixa variabilidade temporal.
- **Cenário 2 — Alta Demanda com Conflitos de Horário:** 40 motoristas, 70 tarefas, múltiplos períodos sobrepostos, ocorrências de turnos com início simultâneo e forte impacto das janelas temporais.
- **Cenário 3 — Operação Internacional:** 20 motoristas com categorias distintas, viagens transfronteiriças, restrições completas do Regulamento 561/2006, incluindo pausas e repousos semanais reduzidos.

B.2 Configurações do Solver (CP-SAT / OR-Tools)

Embora o solver CP-SAT não seja um MILP clássico, utiliza mecanismos análogos a *presolve*, cortes e heurísticas internas, configurados implicitamente ou via parâmetros de alto nível.

As principais configurações aplicadas ao OR-Tools foram:

- **Critério de otimalidade relativa:** tolerância implícita configurada por limite de tempo e critérios internos do CP-SAT..
- **MaxTime:** 600 segundos (10 minutos).
- **Presolve:** ativado.
- **Cuts:** automáticas.

- **Heurísticas internas do CP-SAT:** ativadas automaticamente (busca guiada por SAT, propagação de restrições e heurísticas de viabilidade)

O tempo de execução variou entre 4s e 35s para os cenários básicos, e até 2m15s para os cenários de escala ampliada.

B.3 Restrições Consideradas

Foram aplicadas integralmente as seguintes exigências:

- limite de 9h de condução diária (com variação permitida para 10h)
- pausas mínimas de 45 minutos a cada 4h30 de direção
- repouso diário regular de 11h ou fracionado em 3h + 9h
- controle semanal de horas acumuladas
- compatibilidade entre motorista e tarefa

APÊNDICE C – Estrutura de Entradas e Saídas do Solver

Este apêndice descreve o formato dos dados utilizados no solver OR-Tools e as estruturas retornadas após a resolução do modelo.

C.1 Entradas do Modelo

- **Matriz de tarefas (T):** Contém identificação, duração, janelas de início e compatibilidades.
- **Conjunto de motoristas (M):** Inclui categoria, histórico de horas, limites semanais e tipo de operação.
- **Parâmetros regulatórios (R):** Conjunto estruturado contendo:
 - limites de condução diária e semanal,
 - regras de pausa,
 - repousos diários e semanais.
- **Parâmetros do Solver (S):**
 - MIP_gap,
 - tempo máximo,
 - níveis de corte,
 - ativação de heurísticas.

C.2 Saídas do Modelo

- **Alocação motorista–tarefa ($x_{ij} = 1$):** Indica qual motorista foi designado a cada tarefa.
- **Horas acumuladas:** Relatório do total de horas diárias e semanais por motorista.
- **Verificação de conformidade:** Conjunto de flags apontando violações regulatórias.
- **Valor final da função objetivo (Z^*):** Valor ótimo ou incumbente final obtido pelo solver, conforme o critério de parada.

- **Relatório temporal:** Tempo de execução do solver, número de nós explorados, cortes aplicados.

APÊNDICE D – Modelo Matemático Completo

Este apêndice apresenta a formulação completa do modelo de Programação Linear Inteira utilizado na dissertação.

D.1 Variáveis de Decisão

$$x_{ij} = \begin{cases} 1 & \text{se o motorista } i \text{ realiza a tarefa } j, \\ 0 & \text{caso contrário.} \end{cases}$$

$h_i^{\text{dia}}, h_i^{\text{sem}}$: horas acumuladas diário e semanal.

D.2 Função Objetivo

$$\min Z = \sum_{i,j} c_{ij} x_{ij},$$

onde c_{ij} representa custos operacionais, distância, categoria ou penalidades.

D.3 Restrições

D.3.1 Atribuição Única

$$\sum_i x_{ij} = 1 \quad \forall j.$$

D.3.2 Limite de Condução Diária

$$h_i^{\text{dia}} = \sum_j d_j x_{ij} \leq 9 + e_i.$$

D.3.3 Janelas de Tempo

$$s_j x_{ij} \geq a_j, \quad s_j x_{ij} + d_j \leq b_j.$$

D.3.4 Compatibilidade Motorista–Tarefa

$$x_{ij} \leq \text{comp}(i, j).$$

D.3.5 Acúmulo Semanal

$$h_i^{\text{sem}} = \sum_j d_j x_{ij} \leq 56.$$

D.4 Modelo Final

Versão consolidada inclui todas as restrições regulatórias (CE 561/2006), limites de pausa, repouso e condução quinzenal, incorporadas por meio de estruturas lineares ou restrições complementares.

Anexos

ANEXO A – Resumo Estruturado do Regulamento (CE) n.º 561/2006

Este anexo apresenta um resumo das principais disposições do Regulamento (CE) n.º 561/2006, organizado em formato tabular para facilitar consulta durante a interpretação do modelo de escalonamento.

Limites de Condução

Tabela 6 – Limites de Condução

Parâmetro	Descrição
Condução diária	Máximo de 9 horas; permitido estender para 10 horas até 2 vezes por semana.
Condução semanal	Máximo de 56 horas.
Condução quinzenal (2 semanas)	Máximo de 90 horas acumuladas.

Pausas Obrigatórias

Tabela 7 – Regras de Pausa

Condição	Descrição
Pausa após condução contínua	45 minutos após no máximo 4h30 de condução.
Divisão da pausa	Permitido dividir em 15 min + 30 min.

Repouso Diários

Tabela 8 – Repouso Diários

Tipo	Descrição
Repouso regular diário	Mínimo 11 horas.
Repouso diário fracionado	3h + 9h (total mínimo 12h).
Repouso reduzido diário	Mínimo de 9 horas, até 3 vezes entre dois repouso semanais.

Repouso Semanal

Tabela 9 – Repouso Semanais

Tipo	Descrição
Repouso semanal regular	Mínimo de 45 horas.
Repouso semanal reduzido	Mínimo de 24 horas, compensação obrigatória em até 3 semanas.

ANEXO B – Glossário Técnico

PLI — Programação Linear Inteira: Modelo matemático onde todas as variáveis são inteiras, utilizado para problemas de decisão e alocação.

MIP — Mixed Integer Programming: Problemas que combinam variáveis inteiras e contínuas, resolvidos via métodos como Branch-and-Bound.

MIP Gap: Medida do quão distante a solução corrente está da solução ótima, calculada como:

$$\text{Gap} = \frac{|Z^* - Z_{best}|}{Z^*}.$$

Presolve: Conjunto de transformações aplicadas pelo solver antes da otimização, reduzindo o tamanho do modelo.

Cutting Planes (Cuts): Restrições adicionais que eliminam regiões inviáveis da relaxação linear, acelerando convergência.

OR-Tools: Pacote de otimização do Google usado para resolver problemas de rota, scheduling e alocação.

Window Time (Janela de Tempo): Intervalo permitido para início ou término de uma tarefa.

Feasibility Pump: Heurística para encontrar soluções inteiras rapidamente em MIP.

ANEXO C – Fluxo Geral da Aplicação

Este anexo apresenta o fluxo geral da aplicação desenvolvida para implementação do modelo de Programação Linear Inteira aplicado ao escalonamento de motoristas sob o Regulamento (CE) n.º 561/2006.

O diagrama foi construído utilizando a linguagem PlantUML e ilustra, de forma abstrata e didática, todas as etapas principais da ferramenta: configuração dos parâmetros via interface, construção programática das variáveis e restrições, execução do solver CP-SAT e geração dos resultados analíticos.

Esse nível de detalhamento permite compreender a arquitetura e o funcionamento da solução sem expor o código-fonte propriamente dito, que é de natureza confidencial e constitui segredo comercial da organização.

A Figura 21 apresenta a visão geral do fluxo da aplicação.

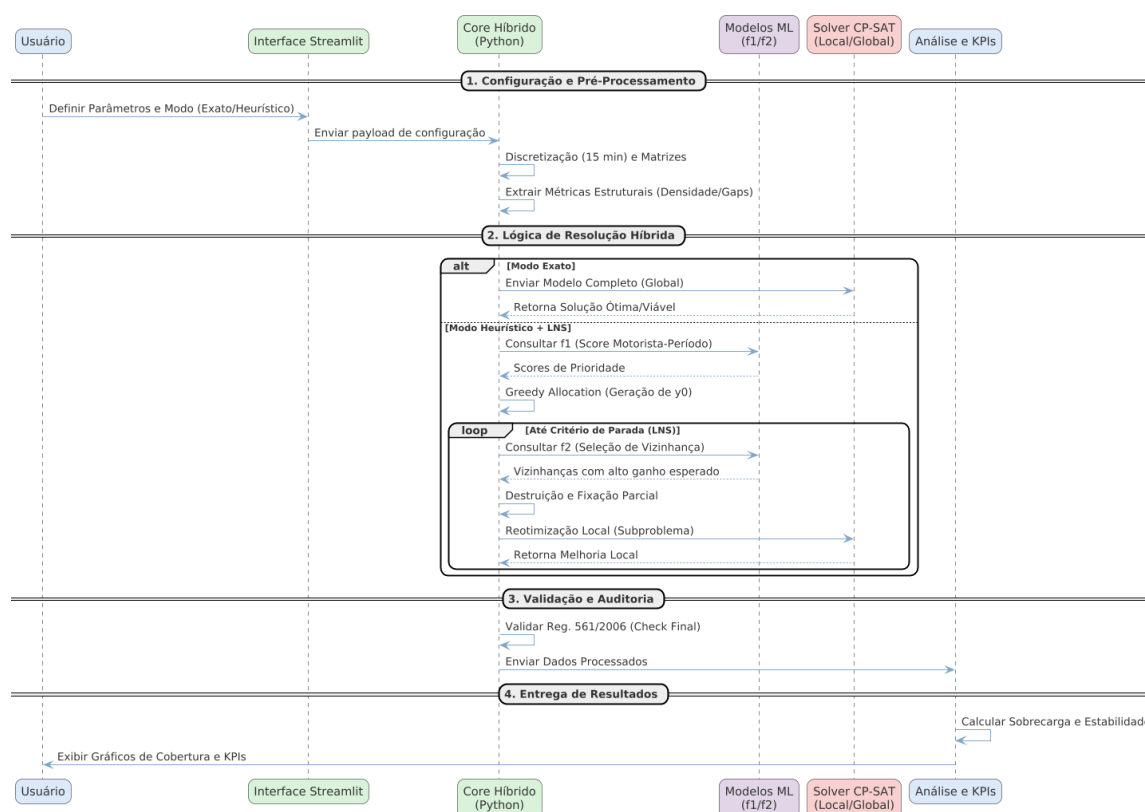


Figura 21 – Fluxo geral da aplicação desenvolvido em PlantUML.