

Universidade Federal do Maranhão

Centro de Ciências Exatas e Tecnológicas

Programa de Pós-Graduação em Ciência da Computação

HEYGON HENRRIQUE FERNANDES ARAÚJO

**DOTSIM: UMA METODOLOGIA BASEADA EM OTIMIZAÇÃO
E SIMULAÇÃO DE EVENTOS DISCRETOS PARA
DETERMINAÇÃO DA SEQUÊNCIA ÓTIMA DE DUPLICAÇÃO
EM SISTEMAS DE TRANSPORTE DE CARGAS**

São Luís
2017

HEYGON HENRRIQUE FERNANDES ARAÚJO

DOTSIM: UMA METODOLOGIA BASEADA EM OTIMIZAÇÃO
E SIMULAÇÃO DE EVENTOS DISCRETOS PARA
DETERMINAÇÃO DA SEQUÊNCIA ÓTIMA DE DUPLICAÇÃO
EM SISTEMAS DE TRANSPORTE DE CARGAS

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal do Maranhão, **como parte dos requisitos necessários** para obtenção do grau de Mestre em Ciência da Computação.

Orientador: Profº Samyr Béliche Vale

São Luís

2017

Ficha gerada por meio do SIGAA/Biblioteca com dados fornecidos pelo(a) autor(a).
Núcleo Integrado de Bibliotecas/UFMA

Araújo, Heygon.

DOTSIM : uma metodologia baseada em otimização e simulação de eventos discretos para determinação da sequência ótima de duplicação em sistemas de transporte de cargas / Heygon Araújo. - 2017.

116 f.

Coorientador(a): Alexandre Oliveira.

Orientador(a): Samyr Beliche Vale.

Dissertação (Mestrado) - Programa de Pós-graduação em Ciência da Computação/ccet, Universidade Federal do Maranhão, São Luis-MA, 2017.

1. Ferrovias. 2. Métodos Analíticos Lineares. 3. Otimização de processos. 4. Simulação de Eventos Discretos. 5. Sistemas de Transporte de Cargas. I. Beliche Vale, Samyr. II. Oliveira, Alexandre. III. Título.

Heygon Henrique Fernandes Araújo

**DOTSIM: uma metodologia baseada em otimização e
simulação de eventos discretos para determinação da
sequência ótima de duplicação em sistemas de transporte
de cargas**

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal do Maranhão, **como parte dos requisitos necessários** para obtenção do grau de Mestre em Ciência da Computação.

Aprovada em: 13/07/2017

Samyr Béliche Vale

Prof. Doutor em Ciência da Computação -
UFMA
Orientador

Alexandre César Muniz de Oliveira

Prof. Doutor em Computação Aplicada -
UFMA
Co-orientador

Vinícius Ponte Machado

Prof. Doutor em Ciência da Computação -
UFPI

Mário Antonio Meireles Teixeira

Prof. Doutor em Ciência da Computação -
UFMA

São Luís

2017

A Deus e à minha família.

Publicações

1. Araujo H. and Vale S. (2017). DOTSIM - A Simulation-based Optimization Methodology for the Optimal Duplication Sequence on Freight Transportation Systems. In Proceedings of the 19th International Conference on Enterprise Information Systems - Volume 2: ICEIS, ISBN 978-989-758-248-6, pages 459-466. DOI: 10.5220/0006337404590466

Agradecimentos

O resultado deste trabalho não teria acontecido sem os ensinamentos que minha família me passou, em especial aos meus pais Henrique e Socorro, pela dedicação, pelo amor, educação e paciência que me deram, e por lutarem junto comigo sempre para realização dos meus objetivos. Agradeço a minha esposa Arielle Araújo, por ter estado sempre ao meu lado, me apoiando e motivando, apesar das dificuldades, noites mal dormidas, e horas e horas dedicadas ao mestrado.

Aos Professores Alexandre Oliveira e Samyr Vale pela contribuição de suas vastas experiências no ramo da ciência da computação e da pesquisa, estimulando e apoiando a realização deste trabalho, participando como orientador, pesquisador, sendo uma referência em todos os aspectos.

Aos colegas de mestrado, Enrico Miranda e Eggo Pinheiro, por estarem sempre a disposição para compartilhar seus conhecimentos e me orgulho muito de ter vocês como colegas de mestrado. Em especial ao Enrico, pelo convite para participar deste programa de mestrado.

A Vale, em especial aos meus gestores Fabiano Burns e Romulo Rovetta que permitiram que essa jornada no mestrado acontecesse.

E acima de tudo a Deus, que colocou essas pessoas maravilhosas na minha vida além de me dar sempre muita força para continuar crescendo pessoalmente e profissionalmente.

“Se você acha a educação cara, experimente a ignorância.”
(Derek Bok)

Resumo

A definição da sequência ótima de duplicação em vias de sistemas de transportes de cargas consiste de um problema de complexidade intratável. Existem uma grande variedade de *Meta-heurísticas* (MHs) capazes de gerar soluções satisfatórias. Entretanto, é fastidioso conhecer a MH que produzirá a melhor solução para um dado *Problema de Sequenciamento de Duplicação* (PSD) . Não há na literatura uma metodologia para estruturar, planejar e controlar algoritmos e processos na modelagem das variedades de MHs aplicadas para encontrar a solução ótima neste tipo de problema. Este trabalho apresenta uma metodologia de desenvolvimento de processo o qual busca pela sequência ótima de duplicação em um dado sistema de transporte comparando a performance de MH com abordagens existentes, tais como *método analítico lineares* (MALs), e desta forma garantir que o sistema tenha sua capacidade maximizada no menor intervalo de tempo possível. Caso esta sequência seja priorizada de forma incorreta, a tendência será o desperdício de tempo e dinheiro em novas vias as quais não agregarão capacidade ao sistema no curto prazo. O potencial desta metodologia é demonstrado através de um estudo de caso em ferrovias.

Palavras-chaves: Simulação de Eventos Discretos, Otimização de processos, Métodos Analíticos Lineares, Ferrovias, Sistemas de Transporte de Cargas.

Abstract

The definition of the best sequence on route duplication of freight systems consists of a complex NP-hard problem. There exists a huge variety of meta-heuristics (MH) capable of generating satisfactory solutions. However, it is fastidious to know which MH will produce the best solution for a Duplication Sequence Problem (DSP). This paper proposes a process development methodology which guides to evaluate the best duplication sequence comparing the MH's performance with existing approaches such as linear analytical method (LAM), and thus to ensure that the system has its capacity maximized in the possible shortest time interval. If this sequence is prioritized incorrectly, it tends to generate wastes such as time and currency in new routes which will not add system capacity in short term. The potential of this methodology is demonstrated by a case study in railways.

Keywords: Events Discretes Simulation, Optimization, Linear Analytical Methods, Railways, Freight Transportation Systems.

Lista de ilustrações

Figura 1 – Passos para um estudo de simulação. (JERRY, 1984)	25
Figura 2 – Classificação dos sistemas para fins de modelagem. Adaptado, (FILHO, 2001)	26
Figura 3 – Interface gráfica para o DSOL. (PEIXOTO, 2002)	31
Figura 4 – Tela Inicial do Arena.	33
Figura 5 – Representação esquemática do exemplo. Fonte: (FILHO, 2008)	34
Figura 6 – Modelagem, Simulação e Otimização. Fonte: (PROTIL, 2001)	35
Figura 7 – Otimização e Simulação. Fonte: (FU, 2002)	35
Figura 8 – Algoritmo Genético	37
Figura 9 – Cromossomos da População Inicial	38
Figura 10 – Crossover	39
Figura 11 – Mutação	39
Figura 12 – Cromossomos da vigésima quinta geração	40
Figura 13 – Característica de busca do <i>simulated annealing</i> . Fonte: (NARA K.. SIMULATED ANNEALING APPLICATIONS. IN:SONG, 1999)	42
Figura 14 – Algoritmo Simulated Annealing	43
Figura 15 – Fluxograma do algoritmo <i>simulated annealing</i>	44
Figura 16 – Algoritmo de Busca Tabu	46
Figura 17 – Linha singela e linha dupla	47
Figura 18 – Pátio Ferroviário	48
Figura 19 – Pátio de Cruzamento	48
Figura 20 – Representação de segmentos de malha. Fonte: (MEIRELES, 2010)	48
Figura 21 – Exemplo de processo de licenciamento. Fonte: (BOARD; WARCHOT, 2011)	52
Figura 22 – Capacidade Teórica	55
Figura 23 – Capacidade Prática	56
Figura 24 – Visão Geral do DOTSIM	60
Figura 25 – Ciclo de Vida do DOTSIM	62
Figura 26 – A estrutura de um processo no DOTSIM.	64
Figura 27 – Estrutura geral do DOTSIM (Também disponível no ANEXO 1)	64
Figura 28 – Configurações de Vias de Tráfego	75
Figura 29 – Parâmetros de Simulação do Estudo de Caso - Ferrovia	75
Figura 30 – Exemplos de candidatos a avaliação de sequência ótima de duplicação	76
Figura 31 – Silos de Carregamento (Vale, 2013)	78

Figura 32 – Tempos de percurso nos dois sentidos, importação e exportação, entre desvios de cruzamento consecutivos de comprimento maior do que o trem em circulação ou <i>headway</i>	79
Figura 33 – Sequência de duplicação para instância de 10 vias singelas, considerando MAL.	84
Figura 34 – Sequência de duplicação para instância de 20 vias singelas, considerando MAL.	84
Figura 35 – Sequência de duplicação para instância de 30 vias singelas, considerando MAL.	85
Figura 36 – Escopo do trabalho. Adaptado, (MEIRELES, 2010)	87
Figura 37 – <i>Output</i> em arquivo de texto contendo candidatos ou sequências de duplicação e seus respectivos $F(x)$	89
Figura 38 – <i>Inputs</i> em arquivo de texto como premissas a serem consideradas pelo modelo de simulação.	90
Figura 39 – Definição das variáveis de entrada	91
Figura 40 – Definição das variáveis de entrada	91
Figura 41 – Definição das recursos do PSD	92
Figura 42 – Definição das filas dos recursos	92
Figura 43 – Licenciamento de trens em uma via singela	93
Figura 44 – Licenciamento de trens em uma via singela	94
Figura 45 – Classe JSL proposta pelo autor - Crossing Loop	94
Figura 46 – Rotina responsável pela geração de evento de interdição de via.	95
Figura 47 – Quantidade de trens descarregados no porto de acordo com o número de trens no sistema e por instância.	97
Figura 48 – η para cada instância - Sequência MAL	98
Figura 49 – Interação entre os últimos 4 processos do DOTSIM - Fase 3	99
Figura 50 – Evolução do $F(x)$ para 1000 iterações - 10 vias singelas	100
Figura 51 – Evolução do $F(x)$ para 1000 iterações - 20 vias singelas	101
Figura 52 – Evolução do $F(x)$ para 1000 iterações - 30 vias singelas	101
Figura 53 – Comparação entre solução ótima global e MAL	103
Figura 54 – Comparação entre solução ótima global e MAL	103
Figura 55 – Custos de oportunidade por vias de tráfego	105
Figura 56 – Comparação entre solução ótima global (GA) e MAL (Colson)	106
Figura 57 – Sequência ótima de duplicação para o PSD, obtido através da aplicação do GA	106
Figura 58 – Estratégia de duplicação do trechos 1 para aplicação do DOTSIM sem e com viés de custo.	107
Figura 59 – As fases e processos do DOTSIM	115

Lista de tabelas

Tabela 1 – Investimentos públicos e privados em infraestrutura de transportes - Brasil (2001-2010) em R\$ bilhões correntes	20
Tabela 2 – Analogia entre GAs e o sistema natural. Fonte: (PACHECO et al., 1999)	36
Tabela 3 – Analogia ao processo Físico	41
Tabela 4 – Capacidade Prática x Demanda da via (pares de trens/dia)	68
Tabela 5 – Sequência Ótima de Duplicação pelo MAL	68
Tabela 6 – Resultados da análise de saturação dos sistemas ferroviários.	97
Tabela 7 – Tempos computacionais para solucionar o problema de otimização (horas).	102
Tabela 8 – Comparação percentual de Performance entre MHs.	102

Lista de Siglas

AREMA	<i>American Railway Engineering Association</i>
COA	<i>Colony Optimization Algorithm</i>
DOTSIM	Duplicação, Otimização, Transporte, Simulação e Metodologia
EFC	Estrada de Ferro Carajás
FIFO	<i>First In, First Out</i>
GA	Algoritmo Genéticos
GRASP	<i>Greedy Randomized Adaptive Search Procedure</i>
GUI	Graphical User Interface
JSL	<i>Java Simulation Library</i>
LIFO	<i>Last In, First out</i>
MAL	Métodos Analítico Lineares
MHs	Meta-heurísticas
MR	Material Rodante
MTBF	<i>Mean Time Between Failure</i>
MTTR	<i>Mean Time To Repaire</i>
OAE	Obras de Artes Especiais
PCV	Problema do Caxeiro Viajante
PSD	Problema de Squenciamento de Duplicação
SA	<i>Simulation Annealing</i>
SBO	<i>Simulation Based Optimization</i>
SED	Simulação de Eventos Discretos
TS	<i>Tabu-Search</i>
UIC	<i>International Union of Railway</i>
VP	Via Permanente

Sumário

1	INTRODUÇÃO	16
1.1	Contextualização e Definições Gerais	16
1.2	Objetivo Geral	17
1.3	Objetivos Específicos	17
1.4	Justificativa e Considerações	17
1.5	Estrutura do Trabalho	18
2	REFERENCIAL TEÓRICO	19
2.1	Sistemas de Transporte: a importância de sua Infraestrutura	19
2.2	Simulação	20
2.2.1	Razões, vantagens e desvantagens	21
2.2.2	Estruturação de Modelos de Simulação	23
2.2.3	Classificação dos Sistemas para Modelagem e Simulação	25
2.2.4	Terminologia Básica utilizada em Modelagem e Simulação de Sistemas	26
2.2.5	Características e Classificação	28
2.3	Ambientes e Bibliotecas de Simulação	29
2.3.1	DSOL	30
2.3.2	JSL	31
2.3.3	Arena®	32
2.3.4	Simulação de Sistemas com Filas	34
2.4	Técnicas Heurísticas de Otimização	35
2.4.1	Algoritmos Genéticos	36
2.4.1.1	Representação dos Parâmetros	38
2.4.1.2	Seleção	38
2.4.1.3	Crossover e Mutação	38
2.4.2	Simulated Annealing	40
2.4.2.1	Analogia ao Processo Físico	40
2.4.2.2	O Algoritmo Simulated Annealing	42
2.4.3	Busca Tabu	45
2.5	Modelo matemático para cálculo de capacidade de tráfego em ferrovias	46
2.5.1	Elementos do sistema Ferroviário	46
2.5.2	Principais Trabalhos a respeito da definição da Capacidade de Tráfego	48
2.5.3	Categorização dos parâmetros utilizados no modelo matemático	50
2.5.3.1	Conceituando as Categorias	51

2.5.3.1.1	Categoria 1 - Tempo decorrente de deslocamento dos trens	51
2.5.3.1.2	Via Permanente	51
2.5.3.1.3	Material Rodante - MR	52
2.5.3.1.4	Categoria 2 - Tempo decorrente de licenciamento e controle dos trens	52
2.5.3.1.5	Categoria 3 - Tempo decorrente de interrupção da circulação de trens para intervenção programada de manutenção	53
2.5.3.2	Categoria 4 - Tempos decorrentes de interrupção da circulação de trens não programada	53
2.5.3.3	Categoria 5 - Tempo decorrente da Gestão Operacional	54
2.5.3.4	Categoria 6 - Tempo decorrente de Recurso de Via	54
2.5.4	Modelo Matemático	55
2.6	Trabalhos Relacionados	57
3	A METODOLOGIA DOTSIM	59
3.1	Visão Geral do DOTSIM	59
3.2	A metodologia DOTSIM	62
3.2.1	Fase 1: Captura e Análise de requisitos	65
3.2.2	Fase 2: Verificação, Validação e Implementação	68
3.2.3	Fase 3: Análise e Solução de Problemas	71
4	UMA APLICAÇÃO DO DOTSIM EM DIFERENTES INSTÂNCIAS DE UM SISTEMA FERROVIÁRIO DE CARGAS	74
4.1	O PSD para Ferrovias	74
4.2	Aplicação do DOTSIM	77
4.2.1	Fase 1	77
4.2.2	Fase 2	85
4.2.3	Fase 3	98
4.3	Resultados	100
4.4	Definição de sequência ótima de duplicação por um viés de custo	104
5	CONCLUSÕES E TRABALHOS FUTUROS	108
	REFERÊNCIAS	110

1 Introdução

1.1 Contextualização e Definições Gerais

Investimentos em rodovias e outros tipos de avanços em sistemas de transporte são amplamente reconhecidos como um importante meio de alcançar um crescimento econômico e desenvolvimento em níveis locais, estaduais e nacionais também. Expansões e melhorias em meios de transportes servem para reduzir os custos dos negócios e aumentar as oportunidades em uma economia. Em geral, contribuindo para aumentar os rendimentos e melhorar a qualidade de vida (WEISBROD; TREYZ, 1998).

O transporte ferroviário de cargas é um dos mais eficientes no mundo pois ele pode oferecer altas capacidades para cargas e passageiros com baixos custos em comparação com o transporte rodoviário de cargas por exemplo. O crescimento da sua infraestrutura de suporte é tão complexo quanto sua importância. Isto por que é possível identificar mais de 100 variáveis capazes de influenciar em sua capacidade de transporte, e consequentemente em qualquer estudo de investimento sobre a mesma.

Estas variáveis, além de possuírem, em sua grande maioria, um comportamento estocástico, podem estar relacionadas a interferências operacionais e de manutenção, e incluídas em diferentes ambientes: na via-permanente (infraestrutura e superestrutura de via), material rodante (locomotivas e vagões), e ainda, equipamentos ligados ao processo de carga e descarga de produtos ou pessoas.

A determinação da sequência ótima de duplicação da infraestrutura de um modal de transporte é um clássico exemplo deste tipo de complexidade.

Softwares e técnicas de simulação de processos estocásticos ainda são poucos difundidos em empresas Brasileiras. Apenas um seleto grupo de técnicos e profissionais possuem habilidades para modelagem e operação de ambientes de simulação. Este cenário tem levado as empresas a tomarem decisões, seja para sistemas produtivos já existentes ou novos projetos, baseando-se apenas em métodos analíticos lineares e estáticos, em problemas que são naturalmente complexos e não lineares.

Métodos e ferramentas baseados apenas em otimização tem provado sua eficiência com problemas operacionais em complexos campos tais como logística ou sistemas manufaturados, porém há uma discussão sobre a falta de flexibilidade na modelagem de sistemas reais. Na maioria dos casos, a inerente estocasticidade para sistemas não é inclusa nos modelos desenvolvidos, assim reduzindo sua aplicabilidade para cenários reais.

Portanto, na maioria dos estudos de simulação o nível de otimização alcançado

depende do número de cenário evoluídos, os quais em geral são apenas uma pequena fração de uma enorme disponibilidade de configurações diferentes. Assim a otimização não é assegurada com uma abordagem de simulação autônoma e também não é assegurada com uma abordagem de otimização autônoma.

Diante deste cenário, o desenvolvimento de uma metodologia contribuiria para que essas deficiências fossem superadas, combinando técnicas de otimização e simulação, permitindo avaliar a robustez das soluções encontrados em problemas que envolvem a ampliação da capacidade de sistemas de transporte de carga.

1.2 Objetivo Geral

Proposta de uma metodologia de desenvolvimento de processo a qual busca pela sequência ótima de duplicação em um dado sistema de transporte comparando a performance de MHs com abordagens existentes, tais como métodos analíticos lineares (MALs), e desta forma garantir que o sistema tenha sua capacidade maximizada no menor intervalo de tempo possível.

A eficácia da metodologia proposta é avaliada com uma aplicação em diferentes instâncias de sistemas de transporte ferroviários, tanto por um viés de aumento de capacidade produtiva quanto por minimização de custos. As instâncias estão atreladas a quantidade de vias singelas no sistema.

1.3 Objetivos Específicos

- Desenvolver metodologia a ser seguida em problemas semelhantes conforme citado no objetivo geral, composta uma arquitetura e de seus respectivos processos e atividades. Além disso, elaborar o ciclo de vida abordando desde a necessidade do cliente até o fim de seu ciclo em que as necessidades são atendidas, apresentando através de diagramas e tabelas, como as atividades e processos se relacionam umas com as outras para que se torne visível todo o processo de desenvolvimento.
- Analisar e propor o melhor fluxo e forma de integração entre os principais atores da metodologia (Simulação de Eventos Discretos - SED, otimização e MAL).
- Avaliar o desempenho da metodologia em um estudo de caso voltado para aumento de capacidade de transporte de cargas no modal ferroviário.

1.4 Justificativa e Considerações

Entre as principais justificativas para este trabalho estão:

Para muitos exemplos de PSD a busca exaustiva pela melhor solução é bastante demorada. Um problema com $N = 20$ vias singelas possui em torno de $2,4 \times 10^{18}$ soluções factíveis, e por esse motivo esse problema é considerado um problema de otimização NP-hard. Uma busca exaustiva processada por um computador com capacidade de processar um bilhão de soluções por segundo encontraria a solução ótima após aproximadamente 77 mil anos. Isto acontece por que para cada via que decida-se duplicar, haverá $N-1$ possíveis soluções até que se acabe a quantidade de vias singelas. Certamente, um decisor não irá esperar esse tempo para definir a melhor sequência de duplicação em um sistema de transporte de Carga singelo.

Técnicas de simulação de sistemas ainda são pouco difundidas nas empresas Brasileiras. O número de analistas e técnicos habilitados em modelar e operar ambientes de simulação ainda é raro. Diante deste cenário, a integração entre simulação e otimização, acaba sendo um assunto ainda mais raro e tampouco difundido.

Todo esse contexto acaba levando as empresas a utilizarem métodos analíticos, lineares e estáticos, para problemas que são naturalmente complexos e não-lineares. Estes métodos empregam estudos com tantas simplificações que muitos projetos, depois de implantados, acabam sofrendo inúmeras modificações e adaptações.

Diante da inexistência na atual literatura de uma metodologia de desenvolvimento de processo que crie um conjunto de convenções de modelagem, que forneça uma orientação em relação à ordem de execução dos processos e atividades, e em geral, que preescreva os procedimentos, regras, documentação, técnicas e ferramentas necessárias para solução do PSD em sistemas de transportes, esta pesquisa propõe a criação de uma metodologia chamada DOTSIM, a fim de proporcionar todas essas contribuições .

1.5 Estrutura do Trabalho

Esta dissertação concentra trabalhos para propor uma metodologia baseada em simulação de eventos discretos e técnicas de otimização organizado da seguinte forma: na SEÇÃO 2, uma revisão literária sobre sistemas de transporte, capacidade ferroviária, simulação de eventos discretos e técnicas de otimização de processos. Três meta-heurísticas (MHs), baseados em modelo de espaço de busca, são discutidas. Na SEÇÃO 3 explica-se a descrição da metodologia DOTSIM. Ainda na SEÇÃO 3 apresenta-se um Modelo Analítico Linear (MAL) para cálculo de capacidade ferroviária; um estudo de caso aplicado a ferrovias singelas é reportado na SEÇÃO 4.3. SEÇÃO 5 apresenta conclusões e comentários finais.

2 Referencial Teórico

2.1 Sistemas de Transporte: a importância de sua Infraestrutura

A infraestrutura de transporte é a espinha dorsal de uma moderna, competitiva e produtiva economia. Os sistemas de transporte facilitam a circulação eficiente de bens e serviços, promovem o comércio e as negociações, conectam as cadeias integradas, e reduzem os custos operacionais em todos os diversificados ramos da indústria. Em todo o mundo esses sistemas possuem características comuns, tais como, estarem em ritmo de expansão; principalmente nas economias emergentes; e por serem sistemas muito complexos. Em geral, são redes de transporte mutuamente interligadas, operadas e consolidadas por uma série de entidades públicas das cidades, estados e governos federais para empresas privadas.

Esta complexidade faz com que a manutenção da infraestrutura de transporte das nações sejam uma crítica e difícil tarefa. E ainda, a necessidade de modernização e reinvestimentos nesses sistemas não podem ser negligenciados, pois tendências de crescimento econômico e demográfico influenciarão diretamente no aumento da demanda por transporte de passageiros e cargas da mesma forma nos próximos anos.

A infraestrutura de transporte no Brasil foi, até a década de 1970, desenvolvida quase exclusivamente com investimentos públicos. Entretanto, a partir da década de 1980, com a crise fiscal do Estado e a elevação do estoque da dívida externa, o setor público passou a enfrentar dificuldades em financiar novos projetos de infraestrutura.

A partir da década de 1990, na busca por novas alternativas de financiamento de projetos de infraestrutura o setor público passou a atrair o setor privado a investir em infraestrutura por meio de instrumentos como contratos de concessão e parcerias público-privadas (PPPs). Estas novas formas de financiamento ainda estão longe de darem conta de financiar os projetos de infraestrutura que o Brasil precisa. A confederação Nacional de Transportes realiza levantamento anual sobre o estado das rodovias brasileiras. O resultado do levantamento de 2013 indica que apenas 36,2% das rodovias brasileiras se encontravam em estado bom ou ótimo ([CASTRO, 2015](#)).

O investimento em infraestrutura por períodos relativamente longos, é condição necessária tanto para o crescimento econômico como para ganhos sustentados de competitividade. Ainda de acordo com esse autor poucos países têm sido capazes de mobilizar recursos ao longo de um horizonte que vai além de 20-30 anos, sem reduções que comprometem a integridade e a qualidade dos serviços ([FRISCHTAK, 2008](#)).

O Brasil não é um desses países. A partir do início da década de 1980, com a deterioração das contas públicas, as políticas setoriais relacionadas à infraestrutura de

transportes no Brasil foram abandonadas. Os investimentos em infraestrutura de transporte têm crescido em anos recentes (TABELA 1), porém várias pesquisas têm demonstrado que estes investimentos ainda estão longe do ideal (CASTRO, 2015).

Tabela 1 – Investimentos públicos e privados em infraestrutura de transportes - Brasil (2001-2010) em R\$ bilhões correntes

Modal	2002	2003	2004	2005	2006	2007-2010	2011-2014	Pós-2014
Transporte rodoviário	5,53	3,86	5,40	6,74	8,67	33,4	50,2	2,0
Transporte ferroviário	0,72	1,11	1,90	3,21	2,53	7,9	43,9	2,1
Portos	0,44	0,20	0,44	0,50	0,58	1,6	4,8	0,3
Hidrovias	0,09	0,05	0,08	0,10	0,11	0,7	2,6	0,1

A infraestrutura de transporte em outros sistemas como o Norte Americano está passando por uma situação de baixa performance. As auto estradas, pontes, ferrovias, aeroportos, sistemas de trânsito e hidrovias, que já foram representados como os melhores da arquitetura moderna e engenharia, estão se deteriorando. Segundo o autor, entre as principais razões para esta situação é que estes recursos inevitavelmente vem sendo desgastados pela própria idade e exigidos além de sua capacidade por essa transição e incremento de demanda ocasionado pela economia moderna e crescimento populacional (ROUNDTABLE, 2015 (accessed September 7, 2016)).

Os benefícios econômicos associados a investimentos em infraestrutura podem ser poderosos e sustentáveis. Entre os benefícios intangíveis estão maior segurança e menor poluição; entre os tangíveis estão: um potencial de geração de aproximadamente 1,7 milhões de empregos nos primeiros 3 anos, um aumento estimado de \$ 3 dólares na atividade econômica para cada \$ 1 dólar investido em infraestrutura. E por fim, mitigar a perda na renda familiar de aproximadamente \$ 3.100 na média anual associada a infraestrutura Americana com baixa performance (ROUNDTABLE, 2015 (accessed September 7, 2016)).

2.2 Simulação

Frente a complexidade desses sistemas de transportes de cargas justificada principalmente pela quantidade de variáveis que estão ligadas a sua capacidade produtiva, ferramentas tecnológicas que possuem a habilidade de imitar o comportamento de sistemas e processos, tanto simples quanto complexos, têm contribuído de forma relevante nos últimos anos. Uma destas ferramentas é a simulação computacional.

A simulação tem sido cada vez mais aceita e empregada como uma técnica que permite aos analistas dos mais diversos seguimentos (administradores, engenheiros, biólogos, técnicos em informática, etc) verificarem ou encaminharem soluções com a profundidade desejada, aos problemas com os quais lidam diariamente (FILHO, 2001).

Mais do que nunca, a simulação computacional tem sido utilizada. O crescimento da utilização dessa ferramenta de uso e sofisticação dos ambientes de desenvolvimento de modelos computacionais, aliadas ao crescente poder de processamento das estações de trabalho. Contando com interfaces gráficas cada vez mais amigáveis, destinadas às mais diversas plataformas e, principalmente, fazendo intenso uso da animação dos sistemas que estão sendo simulados, a simulação deixou para trás o estigma de ser utilizada apenas “quando tudo mais já foi tentado” (FILHO, 2001).

Uma definição mais completa, abrangendo todo o processo de simulação, determina que a simulação é o processo de desenvolver, projetar um modelo computacional de um sistema real e conduzir experimentos com este modelo com o objetivo de entender seu comportamento, além de avaliar estratégias para sua operação, usualmente buscando responder perguntas do tipo *o que aconteceria se...*, também conhecido como *What-if* (PEGDEN; SADOWSKI; SHANNON, 1995). Como se observa, o autor entende a simulação como um processo mais amplo, compreendendo não somente a construção do modelo, mas, também, todo o método experimental que se segue, buscando, sobremaneira:

- Descrever o comportamento do sistema;
- Construir teorias e hipóteses considerando as observações efetuadas;
- Usar o modelo para prever o comportamento futuro, isto é, os efeitos produzidos por alterações no sistema ou nos métodos empregados em sua operação;

No grupo seletivo de softwares de simulação de modelos estocásticos pode-se citar o GPSS, JSL, Arena, ProModel, Simul8 e FlexSim dentro outros em um universo que pode chegar a mais do que 60 softwares de simulação, em sua maioria comerciais (SWAIN, 2007). No Brasil, o primeiro software de simulação não comercial foi publicado em 2012, chamado de URURAU.

2.2.1 Razões, vantagens e desvantagens

As razões mais comuns para experimentar-se com modelos simulados são as seguintes:

- O sistema real ainda não existe: nesse caso a simulação poderá ser usada para planejar o futuro sistema. Um novo hospital, uma nova fábrica ou um novo ambiente de suporte a negócios na internet, por exemplo;
- Experimentar com o sistema real é dispendioso: o modelo poderá indicar, com muito menos custo quais os benefícios de se investir em um novo equipamento, por exemplo;

- Experimentar com o sistema real não é apropriado: um caso típico é o planejamento do atendimento de situações de emergência, um desastre aéreo em um aeroporto, por exemplo. Toda a logística para o acionamento e atuação de serviços prestados pela polícia, bombeiros, ambulâncias, emergência hospitalar, etc., podem ser modelados e tratados no computador. Não se pode provocar um desastre desse tipo para testar planos de emergência.

Apesar de a simulação ser uma excelente ferramenta de análise, é preciso conhecer um pouco mais de profundidade tanto as vantagens quanto as desvantagens de seu emprego. A duas listas a seguir, baseadas nos textos de (PEGDEN; SADOWSKI; SHANNON, 1995) e (BANKS; MALAVÉ, 1984), apontam algumas delas. Inicia-se com alguns de seus benefícios e, posteriormente, com algumas de suas desvantagens.

Vantagens:

- O modelo de simulação pode ser usado quantas vezes forem necessárias;
- A simulação permite a avaliação de um sistema proposto, mesmo que os dados de entrada estejam ainda na forma de esquemas ou rascunhos;
- É mais fácil de aplicar do que métodos analíticos, pois este último método corriqueiramente não consegue abranger todas as variáveis envolvidas nos processos em geral. Por exemplo admitir coeficientes que vão representar determinadas condições, tais como, eficiência operacional, disponibilidade física de equipamentos, etc;
- Os modelos de simulação não apresentam restrições quanto ao número de dados;
- Os modelos de simulação podem ser quase tão detalhados quanto os sistemas reais, com isso as decisões podem ser tomadas sem que o sistema real seja perturbado;
- Hipóteses sobre como ou por que certos fenômenos acontecem podem ser testados para confirmação;
- O tempo pode ser controlado para que os fenômenos sejam bem estudados;
- Pode se compreender quais as variáveis são as mais importantes e como interagem entre si e com os demais elementos do sistema;
- A identificação dos gargalos pode ser obtida de forma mais fácil com a ajuda visual;
- Um sistema de simulação costuma mostrar como realmente um sistema opera, em oposição à maneira com que todos pensam que ele opera;
- Novas situações sobre as quais se tenha pouco conhecimento ou experiência podem ser tratadas.

Embora possua inúmeras vantagens, o processo de simular apresenta algumas dificuldades: Desvantagens:

- A construção de modelos requer treinamento especial. Envolve arte e, portanto, o aprendizado se dá ao longo do tempo, com a aquisição de experiência. Dois modelos de um sistema construídos por dois indivíduos competentes terão similaridades, mas dificilmente serão iguais.
- Os resultados das simulações são muitas vezes de difícil interpretação. Uma vez que os modelos tentam capturar a variabilidade do sistema, é comum que existam dificuldades em determinar quando uma observação realizada durante uma execução se deve a alguma relação significante no sistema ou a processos aleatórios construídos e embutidos no modelo;

A modelagem e a experimentação, associados a modelos de simulação consomem muitos recursos, principalmente tempo. A tentativa de simplificação na modelagem ou nos experimentos objetivando economia de recursos costuma levar a resultados insatisfatórios. Em muitos casos a aplicação de métodos analíticos (como a teoria das filas, por exemplo) pode trazer resultados menos ricos e mais econômicos.

2.2.2 Estruturação de Modelos de Simulação

Os tópicos a seguir apresentam os passos usuais para o desenvolvimento de um estudo de simulação (BANKS; NORMAN, 1995).

- **Reconhecimento do problema:** significa identificar todos os fatos e aspectos que se tem pretensão de estudar para que estes sejam considerados na formulação do modelo.
- **Formular o problema:** implica em selecionar os elementos do sistema global de interesse no estudo a ser realizado, fixar a fronteira do novo sistema o qual engloba os elementos selecionados, definir os objetivos do estudo, selecionar o conjunto de parâmetros de medida de performance do sistema, estabelecer o horizonte de tempo, e identificar os anseios do usuário final.
- **Obter e analisar dados do sistema:** isto implica que valores das variáveis de entrada, parâmetros do sistema e medidas de performance devem ser levantados. Por meio da análise estatística, os dados serão caracterizados. E no caso de variáveis aleatórias poderão ser definidos tipos de distribuições mais apropriadas para descrevê-las.

- **Verificar e validar o modelo:** verificar significa certificar se as rotinas computacionais implementadas geram os valores esperados. Validação consiste na comparação dos dados gerados pelo modelo com os obtidos do sistema real. Para tanto, devem ser utilizados procedimentos estatísticos.
- **Documentar o Modelo:** durante o desenvolvimento do modelo o programador deve elaborar relatórios que contenham detalhes como: os objetivos, as suposições consideradas, e detalhes do desenvolvimento das rotinas computacionais. Isto facilita procedimentos futuros, quando da necessidade de re-adequação do modelo.
- **Definir tipos de experimentos:** visa estabelecer para quais condições dos valores das variáveis de entrada e parâmetros do sistema os resultados gerados pelo modelo são confiáveis.
- **Estabelecer as condições de uso:** por meio dos experimentos realizados é definida para quais condições é aplicável o uso do modelo.

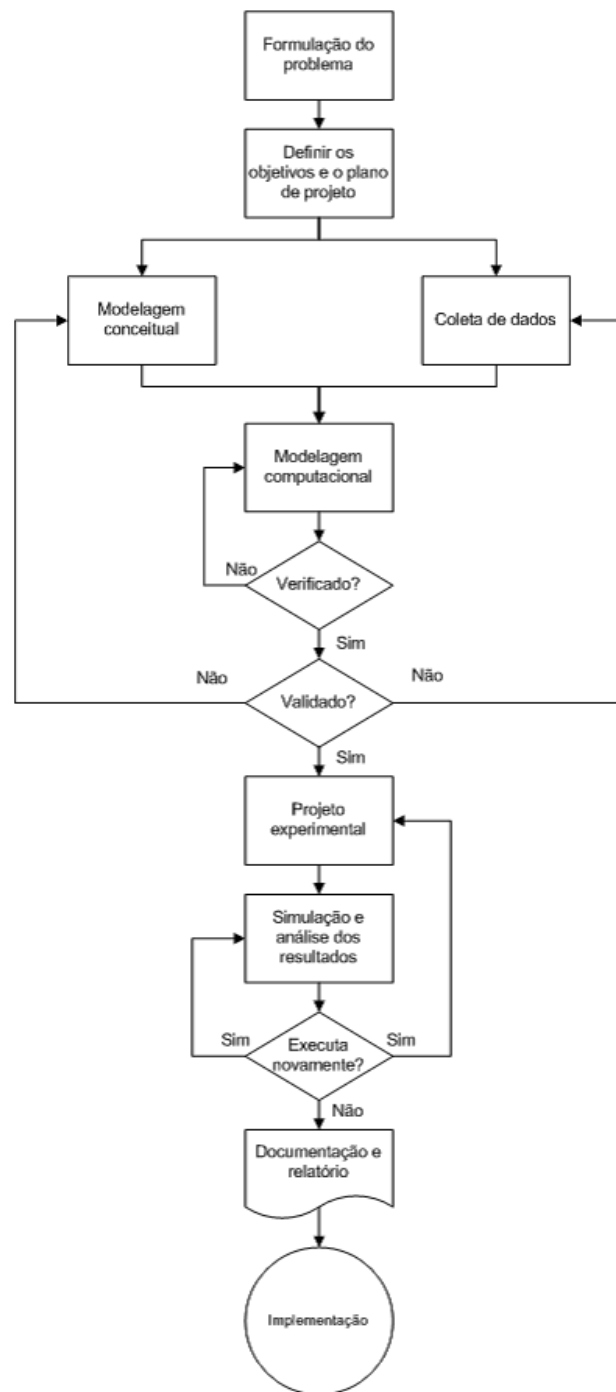


Figura 1 – Passos para um estudo de simulação. (JERRY, 1984)

2.2.3 Classificação dos Sistemas para Modelagem e Simulação

A simulação computacional utiliza-se de modelos computacionais para estudar e analisar sistemas e problemas a eles relacionados. E modelos computacionais voltados à simulação de sistemas, dependem, fundamentalmente, da natureza do sistema a ser estudado.

Sistemas podem ser definidos como um conjunto de objetos, entre os quais se

podem encontrar ou definir algum tipo de relação, que atuam e interagem (cooperam) com a intenção de alcançar um objetivo ou um propósito lógico (SCHMIDT; TAYLOR, 1970). Pode-se ainda definir sistemas como um grupo de componentes que recebem estímulos ou entradas (*inputs*) e produzem respostas ou saídas (*outputs*). São os componentes e suas relações, tanto internas quanto externas ao próprio sistema, que determinam como este converte estímulos em respostas (FILHO, 2001).

Para o propósito de modelar e simular sistemas, estes podem, de maneira geral ser classificados como: estáticos ou dinâmicos, contínuos ou discretos ou, ainda, como determinísticos ou aleatórios. A FIGURA 2 resume essa classificação. Esta pesquisa tem como escopo os sistemas que se enquadram como: dinâmicos, aleatórios e de modelagem mista.

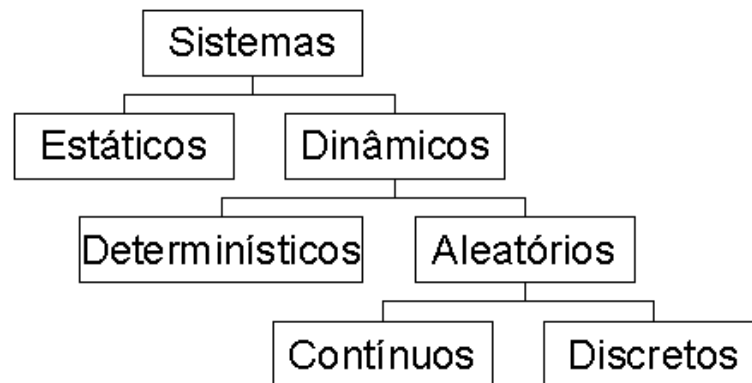


Figura 2 – Classificação dos sistemas para fins de modelagem. Adaptado, (FILHO, 2001)

Os sistemas que serão aqui referidos, estudados, modelados e simulados, são classificados como dinâmicos, uma vez que as variáveis de estado que os representam se modificam na medida em que o tempo evolui. Não são determinísticos, mas aleatórios, pois os possíveis estados das variáveis podem ser descritos, mas não predeterminados. E finalmente, são, também, classificados como de modelagem mista, pois as mudanças de estado ocorrem em pontos discretos de tempo e ainda existem variáveis dependentes do tempo que podem variar de maneira contínua.

2.2.4 Terminologia Básica utilizada em Modelagem e Simulação de Sistemas

Um conjunto de termos é comumente aplicado no campo de conceituação dos elementos básicos envolvidos na modelagem e na simulação de sistemas. Abaixo são apresentados os principais termos e alguns exemplos para diferentes áreas de aplicação tais como, manufatura, serviços ou sistemas computacionais (FILHO, 2001).

- **Variáveis de Estado:** variáveis cujos valores determinam o estado de um sistema, também conhecidas como "variáveis de estado". Em uma fábrica poderia ser um

exemplo o número de peças esperando para serem processadas na máquina (fila da máquina), ou, ainda, o estado da máquina, ocupada ou livre.

- **Eventos:** são acontecimentos, ocorrências, programados ou não, os quais, quando ocorrem, provocam uma mudança de estado em um sistemas. Ex.: uma chegada de peças, de clientes ou de tarefas.
- **Entidades e Atributos:** uma entidade representa um objeto que necessita de uma clara e explícita definição. Ela pode ser dinâmica, movendo-se através do sistema, ou, estática, servindo a outras entidades. Em uma fábrica, as peças seriam exemplos de entidades. As características próprias das entidades, isto é, aquelas que as definem totalmente, são chamadas de atributos. Entidades semelhantes possuem os mesmos atributos. Os valores dos atributos é que as diferenciam entre si. Dessa forma, pode-se, por exemplo, caracterizar ou dar o nome de peça, a toda entidade que é processada em uma determinada máquina de um sistema de manufatura.
- **Recursos e Filas de Recursos:** O recurso é considerado uma entidade estática que fornece serviços às entidades dinâmicas. Um recurso pode ter a capacidade de servir uma ou mais entidades dinâmicas ao mesmo tempo, operando como um servidor paralelo. É possível que uma entidade dinâmica não puder se apoderar de um recurso solicitado, ela deverá aguardar por ele em uma fila. O processamento de uma fila, isto é, a forma como esta será gerenciada depende, fundamentalmente, das políticas operacionais adotadas no sistema ou no modelo que o representa. A política de tratamento de filas mais comum é a FIFO (sigla em inglês para *first In, fist Out*), onde o primeiro a chegar à fila será o primeiro a ser atendido pelo recurso. Outras formas de gerenciamento podem ser adotadas. Um recurso pode ter vários estados. Os mais comuns são: ocupado e livre. Outros podem ser definidos, tais como bloqueado, falhado, indisponível, etc.
- **Atividades e Periodos de Espera:** Em simulação diz-se que uma atividade corresponde a um período de tempo predeterminado. Logo, uma vez iniciada, seu final pode ser programado. A duração de uma atividade, no entanto, não é necessariamente, uma constante. Poderá ser o resultado de uma expressão matemática, um valor aleatório com base em uma distribuição de probabilidades, ser fornecida a partir da leitura em algum arquivo ou fonte externa ou até mesmo ser dependente do estado do sistema.
- **Tempo (real) Simulado e Tempo de Simulação:** Um cuidado necessário por parte de quem está modelando um sistema diz respeito à relação entre o tempo (do sistema real) simulado e o tempo de simulação (tempo necessário à execução de um experimento no computador). Para certos sistemas, o tempo de simulação pode ser muito maior que o tempo simulado. Por exemplo, na simulação de um modelo de

uma rede de computadores, as unidades de tempo admitidas para os eventos podem ser da ordem de milissegundos. Se, no modelo, o número de entidades e o número de processos a que estas devem ser submetidas for grande (milhares de pacotes sendo roteados, por exemplo), o tempo de CPU devotado a esse processamento poderá ser razoável (dependendo da capacidade da CPU). Dessa forma, para simular, digamos 15 segundos de funcionamento desse sistema, é possível que se gastem dezenas de minutos de tempo de computador. No entanto, a recíproca também é verdadeira, e sistemas com processos estocásticos e complexos, em que, por exemplo, 1 ano de funcionamento deste sistema, é possível que se gastem alguns minutos de simulação de tempo de computador. E esta é uma das grandes vantagens da simulação.

2.2.5 Características e Classificação

Uma característica muito interessante da simulação é que se pode controlar a velocidade com que o experimento ocorre manipulando a passagem do tempo. Esta manipulação do tempo pode ser feita de duas formas: pela técnica de fatias de tempo (*time slicing*) ou pela técnica do próximo evento (*next-event technique*). A primeira divide o tempo em intervalos iguais, já a segunda utiliza incrementos variáveis no tempo. Os modelos de simulação podem ser classificados em quatro maneiras:

- **Modelos estáticos ou dinâmicos** – São estáticos aqueles que possibilitam a descrição do estado do sistema somente para dado momento. Deste modo, geralmente estes modelos não envolvem a variável tempo. Enquanto que os modelos dinâmicos são formulados para representar o estado do sistema em função do avanço da variável tempo.
- **Modelos estocásticos ou determinísticos** – Apresentam-se como modelos estocásticos aqueles que possuem pelo menos uma variável de entrada ou parâmetro do sistema tipificado como variável aleatória, enquanto que nos modelos determinísticos não se utilizam variáveis aleatórias.
- **Modelos discretos ou contínuos** – Para os modelos contínuos o avanço de tempo é procedido de forma continua em incrementos de tempo com valores iguais. Para os modelos discretos o avanço da variável tempo, é procedido quando da ocorrência de um evento. Por exemplo, no caso da modelagem de uma fila única em uma agencia bancária, podem ser considerados três eventos: chegada dos clientes, atendimento e saída dos clientes.
- **Formular e desenvolver o modelo:** nesta fase deverá ser formulado o modelo conceitual por meio de representação gráfica – exemplo fluxogramas – e em seqüência o modelo conceitual deverá ser traduzido em relações matemáticas para que este venha tomar a forma de modelo computacional (programa de computador).

Fator	Modelagem Contínua	Modelagem de Eventos Discretos
O que é modelado	Fluxos (quantidades)	Itens discretos (coisas)
Características	O número aleatório “simula” características de fluxos e deve ser repetido para cada consulta ou junção.	Ao usar atributos, características únicas são atribuídas a itens que podem ser rastreados através do modelo.
Avanços no tempo	O intervalo entre os avanços de tempo é constante. Os cálculos do modelo são sequenciais e dependentes do tempo.	O intervalo entre eventos é dependente de quando ocorrem os eventos. O modelo somente recalcula seu estado quando um evento ocorre.
Ordenação	Fluxos seguem a regra FIFO (First In, First Out)	Itens podem fluir seguindo as regras FIFO, LIFO (Last In, First Out), Prioridades ou uma ordem personalizada.
Roteamento	Fluxos precisam ser explicitamente roteados através da ativação e desativação em cada um dos ramos. Os fluxos podem ir para vários lugares ao mesmo tempo.	Itens são automaticamente roteados para o primeiro ramo disponível. Itens somente podem estar em um lugar de cada vez.
Detalhe Estatístico	Somente estatísticas gerais sobre o sistema: montante, eficiência, tempo de trânsito.	Além de estatísticas gerais, cada item pode ser individualmente rastreado: contagens, utilização, tempo de ciclo.
Usos Comuns	Científica (biologia, química, física), Engenharia (eletrônica, sistemas de controle), Processos contínuos, Systems Dynamics, Economia.	Manufatura, Setor de serviços, Processo de negócio, Pensamento estratégico, Redes (de computador, telefônica), Engenharia de sistemas.

TABELA 2.2.5 - Modelagem Contínua x Modelagem de eventos discretos (BANKS; MALAVÉ, 1984).

2.3 Ambientes e Bibliotecas de Simulação

Ambientes de simulação e bibliotecas de simulação são visões que podem ser facilmente confundidas. Os ambientes gráficos e integrados normalmente não necessitam que o usuário entre com linhas de código para construir seus modelos de simulação, além de permitir o design de animação 2D e até mesmo 3D da simulação do modelo. Já nas bibliotecas de simulação geralmente são constituídas de um grupo de rotinas codificadas

em uma linguagem de propósito geral de forma que esta esteja adequada para simular modelos.

Atualmente estão disponíveis diversos ambientes para simulação de eventos discretos: o Arena® e o ProModel® são relativamente fáceis de operar, mas tem a desvantagem do preço da licença de uso ser ainda relativamente cara e a impossibilidade de distribuir o modelo de simulação integrado ao simulador sem o pagamento de tal licença. O Arena® pode ser obtido em sua versão estudante através do site do representante oficial da empresa Rockwell Automation no Brasil, a Paragon (www.paragon.com).

Por outro lado, existem simuladores em formato de biblioteca de software com licença de uso livre como o DSOL em ([JACOBS; LANG; VERBRAECK, 2002](#)) e o JSL (Java Simulation Library) em ([ROSSETTI, 2008](#)), ambas sob a licença GNU General Public Licence (www.gnu.org).

Nas próximas seções são apresentadas algumas dessas bibliotecas de simulação, foco deste trabalho, e utilizadas mundialmente.

2.3.1 DSOL

A biblioteca em Java DSOL (Distributed Simulation Object Library) proposto por ([JACOBS; LANG; VERBRAECK, 2002](#)) permite uma modelagem usando vários formalismos como guiado a eventos, a processo e a atividade. Também permite simulação de sistemas de eventos discretos combinada com simulação de sistemas de tempo contínuo. No DSOL há a possibilidade de informações que estão distribuídas em vários sistemas de informação sejam utilizadas para alimentar o modelo de simulação. Por exemplo, uma cadeia de suprimento em que sistemas de informação estão distribuídos em toda cadeia torna-se útil um simulador que também trabalhe com dados distribuídos como DSOL faz ([VERBRAECK, 2004](#)).

Um modelo em DSOL pode ser orientado a eventos porque o modelo de simulação é descrito como uma lista de eventos ordenada por tempo. Também, o modelo pode ser orientado a atividade, por exemplo, na simulação o mecanismo de avanço de tempo é incrementado e depois as condições da atividade são testadas de acordo com a prioridade da atividade. Se uma condição da atividade é satisfeita, as ações daquela atividade são realizadas. Finalmente, um modelo pode ser feito orientado a processo, ou seja, cada processo no modelo tem sua sequência de ações. Por exemplo, em um modelo de atendimento de um posto de combustíveis os veículos seriam objetos que disparariam os processos: entrar no posto, requisitar frentista, abastecer, liberar frentista e sair.

A simulação de um modelo no DSOL é também definida e compilada em Java como no JSL, mas o DSOL interpreta o código compilado do modelo e executa o experimento que pode ser definido em um arquivo XML. A FIGURA 3 mostra a interface gráfica que

vem com a biblioteca que auxilia o analista a executar e ver a saída de seus experimentos.

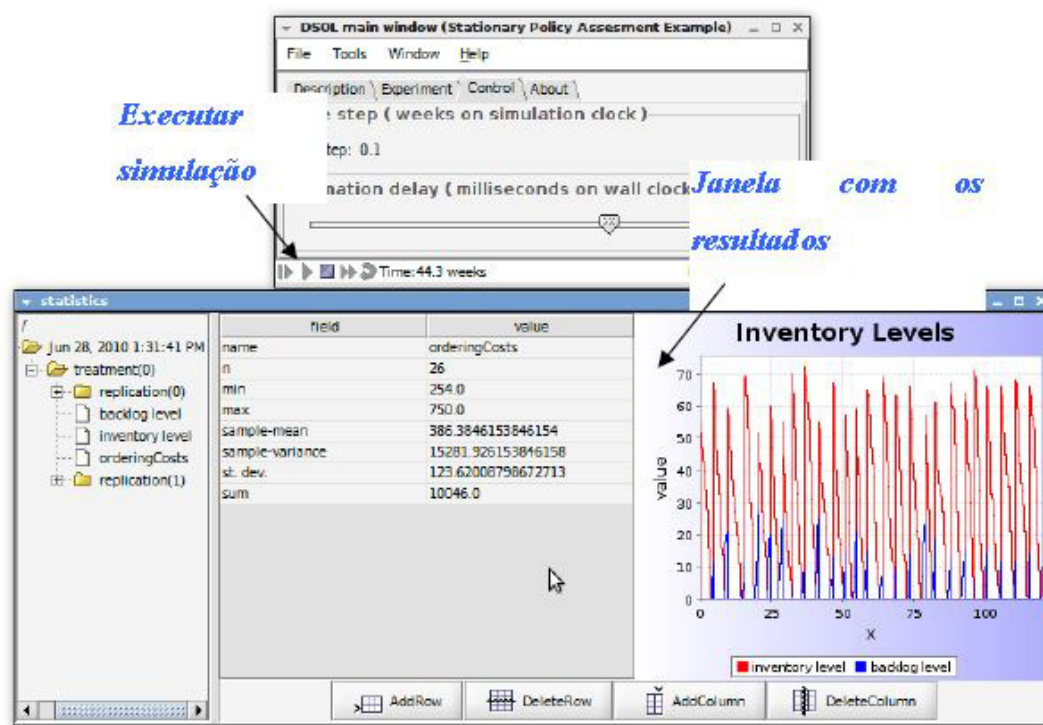


Figura 3 – Interface gráfica para o DSOL. (PEIXOTO, 2002)

2.3.2 JSL

Ambientes de simulação como o Arena® descrevem o processo (ou vida de uma entidade) através de uma serie de comandos. Isto é feito via uma representação em rede de fluxo de entidades como descrito em (JOINES; ROBERTS, 1996). O JSL implementa o processo deixando o usuário descrever o fluxo de uma entidade através de uma série de comandos. Essencialmente, os comandos são mantidos em uma lista e o comando atual que a entidade está em execução é sempre apontado para que o próximo comando possa ser executado. Isto é similar a noção de controle de estado de (JACOBS; LANG; VERBRAECK, 2002).

É importante dizer que simplesmente implementar o processo como uma thread de Java não é eficiente pois ocorre sempre troca de contexto na execução das threads. Isto foi provado em (ROSSETI, 2000) ao fazer interação de processos via thread. (WEATHERLY; PAGE, 2004) discutem esta questão e apresentam um argumento muito razoável para a adição co-rotinas à linguagem Java. (JACOBS; LANG; VERBRAECK, 2002) também discutem os fatos relacionados ao usar threads como mecanismo para fazer interação de processos em Java. Eles desenvolveram formalismos para mapear rotinas de processo no formalismo de escalonamento de eventos via um interpretador Java. No JSL também foi feito um esquema para não usar threads, então um mapeamento da interação de processos

para diagrama de transição de estados onde as funções são baseadas em escalonamento de eventos, mas sem o uso de um interpretador.

2.3.3 Arena®

No seletor grupo dos poderosos softwares de simulação comerciais está o Arena®. Com o surgimento de linguagens orientadas à simulação na década de 50, tornou-se mais fácil a modelagem de sistemas. Com o passar dos anos, essas linguagens foram desenvolvendo e outras ferramentas foram adicionadas às linguagens de simulação, de modo a torná-las uma das ferramentas mais poderosas para o projeto de sistemas.

O software Arena® (FIGURA 4) é o mais novo passo evolutivo da simulação, um ambiente englobando lógica e animação com ferramentas poderosas de análise estatística, que pode ser utilizada para testar modelos e fazer apresentações dos resultados.

Sua interface segue os padrões MS Office®, com menu de funções, comandos e botões semelhantes aos encontrados em outros softwares Windows. No Arena® a modelagem é feita visualmente com objetos orientados à simulação. Para simplificar o processo de construção de modelos, o Arena® usa uma interface gráfica para o usuário (GUI – Graphical User Interface) que automatiza o processo de construção do modelo e reduz a necessidade do teclado. O Arena® também possui ferramentas úteis como o analisador de dados de entrada (Input Analyzer) e o analisador de resultados (Output Analyzer).

O Input Analyzer permite analisar dados reais do funcionamento do processo e escolher a melhor distribuição estatística que se aplica a eles. Esta distribuição pode ser incorporada diretamente ao modelo. Já o Output Analyzer permite analisar os dados coletados durante a simulação e ainda pode efetuar comparações estatísticas.

O Arena® visualiza o sistema a ser modelado como constituído de um conjunto de estações de trabalho que prestam serviços a clientes (também chamados de entidades ou transações) que se movem através do sistema. Assim, para se montar um modelo com o Arena® deve-se, inicialmente, construir um desenho mostrando o sistema que está sendo simulado, constituído de estações de trabalho (onde a entidade receberá algum serviço) e opção de fluxos entre essas estações e as entidades. As opções de fluxo para a entidade serão tratadas pela lógica da programação do modelo.

Somando-se os dados e o modelo lógico-matemático, tem-se a representação do sistema no computador. Com esse modelo é possível realizar vários testes e coletar dados sobre o comportamento do sistema estudado.

Os tópicos, conceitos e definições apresentados abaixo para construção de um modelo no Arena®, baseiam-se nos manuais do software acessíveis nos arquivos de ajuda (Help). Quatro temas básicos do Arena® serão tratados:

- Interface básica do ambiente;
- Introdução à construção de modelos;
- Execução de simulações;
- Animação das simulações;

A Interface do ambiente Arena® consiste no conjunto dos objetos de manipulação, dos painéis e das ferramentas disponíveis na tela principal e barras de ferramentas.

A partir da tela principal é possível abrir um modelo já existente ou iniciar um novo modelo. No meio da tela principal observa-se a área de trabalho, onde são inseridos todos os módulos e suas conexões lógicas necessárias à construção e simulação de um modelo.

Sobre a área de trabalho encontram-se a barra de ferramenta com algumas das ferramentas disponíveis, tais como os painéis dedicados à implementação de elementos de animação dos modelos, desenhos, cores, etc.

À esquerda da área de trabalho tem-se uma área reservada ao anexo de painéis (templates), os quais abrigam os módulos ou comandos necessários à modelagem.

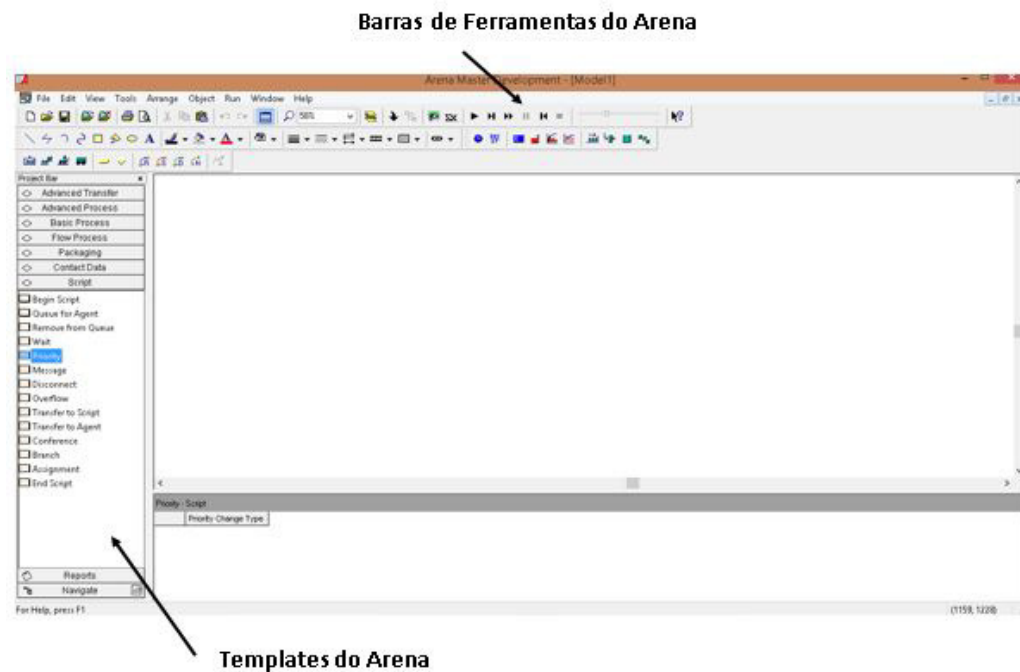


Figura 4 – Tela Inicial do Arena.

A construção de modelos no Arena® exige o conhecimento de conceitos básicos e terminologias. O exemplo adotado para discussão desse assunto trata de um sistema com um único servidor e que pode pertencer a qualquer área de atuação, tais como: manufatura, transportes, logística, serviços, redes de computadores, etc.

As entidades (clientes) chegam e tomam o controle do servidor (ou aderem à fila se este estiver ocupado), realizam um processo ou serviço (gastando algum tempo com o servidor), liberam o servidor após o término do serviço e, finalmente, deixam o sistema (FIGURA 5).

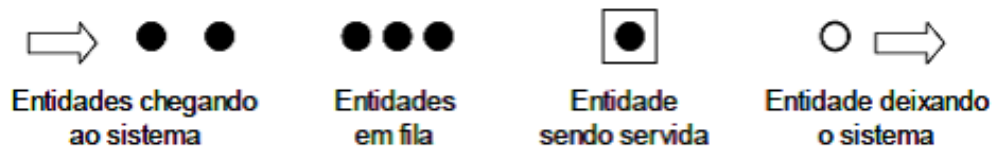


Figura 5 – Representação esquemática do exemplo. Fonte: (FILHO, 2008)

No Arena®, a modelagem é feita considerando uma visão dos processos do sistema. Pode-se simbolizar a lógica do que pode acontecer com uma entidade através de três processos: a criação ou chegada da entidade ao sistema, o serviço ou processamento que esta realiza ou recebe de um recurso e, finalmente, sua partida.

2.3.4 Simulação de Sistemas com Filas

O problema da simulação em Sistemas de Filas é descobrir o número de atendentes necessários para atender os clientes sem ultrapassar o tempo máximo do cliente (determinado por lei) na fila. Porém, às vezes é preferível pagar a multa, pois pode ser mais custoso acrescentar mais um atendente no sistema. Assim, o dimensionamento de filas se desdobra em um problema maior de otimização (MEDINA; CHWIF, 2006) que é descobrir o número de atendentes adequado para atendimento da lei ao menor custo total.

O problema pode ser aplicado à fila de banco, postos de gasolina e a call centers. Neste último, o objetivo do gerente é aumentar a produtividade do atendente (taxa de utilização) sem aumentar o número de ligações perdidas (causadas pelo excesso no tempo de espera) (FILHO, 2001).

Cabe ressaltar que simulação e otimização são distintos. A simulação manipula as entradas de um modelo e verifica suas saídas, já otimização busca obter uma saída ótima, alterando a composição das entradas (PROTIL, 2001). A FIGURA 6 ilustra tal diferença.

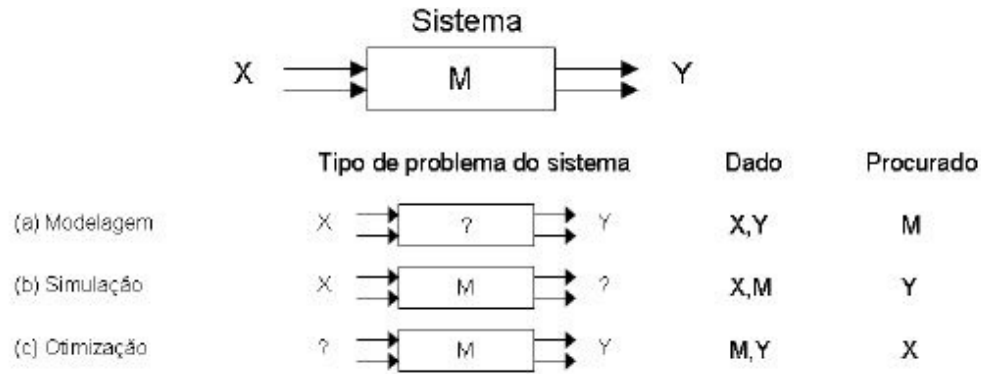


Figura 6 – Modelagem, Simulação e Otimização. Fonte: (PROTIL, 2001)

Porém, simulação e otimização podem ser complementares ((FU, 2002). A simulação fornece informações do modelo para que a otimização seja feita, além de fornecer possíveis soluções para a otimização do modelo. O modelo é então atualizado com os novos valores propostos e uma nova simulação é feita. Este ciclo, representado na FIGURA 7 é repetido até sua parada que é definida de acordo com o método de otimização (convergência nas respostas, número de replicações e tempo).



Figura 7 – Otimização e Simulação. Fonte: (FU, 2002)

2.4 Técnicas Heurísticas de Otimização

É importante enfatizar que esta pesquisa trata o problema de sequência de duplicação em ferrovias como um clássico problema de permutação o qual é considerado como NP-hard pela literatura no ramo da otimização combinatória.

(KANDA, 2014, p. 651-662) menciona que

As MHs mais utilizadas para problema similares (permutação) tais como o problema do cacheiro viajante (PCV) são: Busca-Tabu (TS), Greedy

Randomized Adaptive Search Procedure (GRASP), Simulated Annealing (SA), genetic Algorithms (GAs), e o algoritmo de otimização da colônia de formigas (ACO, do inglês ant colony optimization algorithm).

Esta SEÇÃO tem como objetivo apresentar, de forma sumária, as principais técnicas metaheurísticas referenciadas ao longo deste trabalho.

2.4.1 Algoritmos Genéticos

Algoritmos Genéticos (GAs-Genetic Algorithms) constituem uma técnica de busca e otimização, altamente paralela, inspirada no princípio Darwiniano de seleção natural e reprodução genética (GOLDBERG, 1989).

Os princípios da natureza nos quais os GAs se inspiram são simples. De acordo com a teoria de C. Darwin, o princípio de seleção privilegia os indivíduos mais aptos com maior longevidade e, portanto, com maior probabilidade de reprodução. Indivíduos com mais descendentes têm mais chance de perpetuarem seus códigos genéticos nas próximas gerações. Tais códigos genéticos constituem a identidade de cada indivíduo e estão representados nos cromossomas.

Estes princípios são imitados na construção de algoritmos computacionais que buscam uma melhor solução para um determinado problema, através da evolução de populações de soluções codificadas através de cromossomas artificiais.

Em GAs um cromossoma é uma estrutura de dados que representa uma das possíveis soluções do espaço de busca do problema. Cromossomas são então submetidos a um processo evolucionário que envolve avaliação, seleção, recombinação sexual (crossover) e mutação. Após vários ciclos de evolução a população deverá conter indivíduos mais aptos.

A analogia entre Algoritmos Genéticos e o sistema natural é representada através da TABELA 2:

Tabela 2 – Analogia entre GAs e o sistema natural. Fonte: (PACHECO et al., 1999)

Natureza	Algoritmos Genéticos
Cromossoma	Palavra Binária, vetor, etc
Gene	Características do problema
Alelo	Valor da Característica
Loco	Posição da palavra, vetor
Genótipo	Estrutura
Fenótipo	Estrutura submetida ao problema
Indivíduo	Solução
Geração	Ciclo

GAs são particularmente aplicados em problemas complexos de otimização: problemas com diversos parâmetros ou características que precisam ser combinadas em busca

da melhor solução; problemas com muitas restrições ou condições que não podem ser representadas matematicamente; e problemas com grandes espaços de busca.

Eles também tem sido aplicados a diversos problemas de otimização (PACHECO, 2004), tais como: Otimização de Funções Matemáticas, Otimização Combinatorial, Otimização de Planejamento, Problema do Caixeiro Viajante, Problema de Otimização de Rota de Veículos, Otimização de Layout de Circuitos, Otimização de Distribuição, Otimização em Negócios e Síntese de Circuitos.

De acordo com (LACERDA; CARVALHO, 1999, p. 99-148)

O primeiro passo de um algoritmo genético típico é a geração de uma população inicial de cromossomos, que é formada por um conjunto aleatório de cromossomos que representam possíveis soluções do problema a ser resolvido. Durante o processo evolutivo, esta população é avaliada e cada cromossomo recebe uma nota (denominada de aptidão no jargão da literatura de GAs), refletindo a qualidade da solução que ele representa.

```
inicializar população  $P$ ;  
repita  
  selecione uma subpopulação  $P'$ ; //pais da próxima geração  
  para  $i \leftarrow 1$  até nr_cruzamento faça  
    escolha  $S_1, S_2 \in P'$ , aleatoriamente;  
    filho  $\leftarrow$  cruzamento( $S_1, S_2$ );  
    se  $f(S_1) \geq f(S_2)$  então  $S_{aux} \leftarrow S_1$ ;  
    senão  $S_{aux} \leftarrow S_2$ ;  
    se  $f(S_{aux}) \geq f(\text{filho})$  então  
      filho substitui  $S_{aux}$  em  $P$ ;  
    fim_se;  
  fim_para;  
  para  $i \leftarrow 1$  até nr_mutações faça  
    selecione um cromossomo  $S_j$  em  $P$ ;  
     $S_j \leftarrow$  mutação( $S_j$ );  
  fim_para;  
até que critério_parada seja satisfeito;
```

Figura 8 – Algoritmo Genético

Em geral, os cromossomos mais aptos são selecionados e os menos aptos são descartados (Darwinismo). Os membros selecionados podem sofrer modificações em suas características fundamentais através dos operadores de cruzamento e mutação, gerando descendentes para a próxima geração. Este processo é repetido até que uma solução satisfatória seja encontrada. O pseudocódigo mostrado na FIGURA 8 ilustra este procedimento.

2.4.1.1 Representação dos Parâmetros

Um GA processa populações de cromossomos. Um cromossomo é uma estrutura de dados, geralmente vetor ou cadeia de bits, que representa uma possível solução do problema a ser otimizado. Em geral, um cromossomo representa um conjunto de parâmetros da função objetivo cuja resposta será maximizada ou minimizada. O conjunto de todas as configurações que o cromossomo pode assumir forma o seu espaço de busca. Se o cromossomo representa n parâmetros de uma função, então o espaço de busca é um espaço com n dimensões.

2.4.1.2 Seleção

Inspirado no processo de seleção natural de seres vivos, o algoritmo genético seleciona os melhores cromossomos da população inicial (aqueles de alta aptidão) para gerar **cromossomos filhos** (que são variantes dos pais) através dos operadores de *crossover* e mutação. Uma **população intermediária** (também chamada de *mating pool*) é utilizada para alocar os cromossomos pais selecionados. Geralmente, os pais são selecionados com probabilidade **proporcional à sua aptidão**.

A FIGURA 9 mostra a população inicial de N cromossomos para um dado problema, gerados aleatoriamente pois não existe nenhum conhecimento prévio sobre a região do espaço de busca onde se encontra a solução do problema.

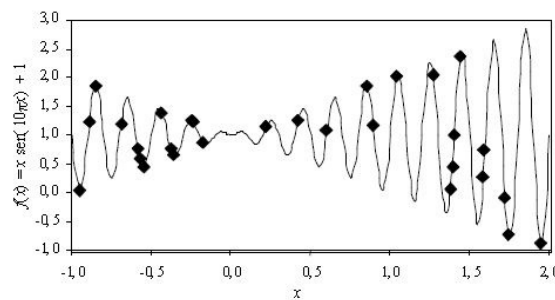


Figura 9 – Cromossomos da População Inicial

2.4.1.3 Crossover e Mutação

Os operadores de *crossover* e a mutação são os principais mecanismos de busca dos GAs para explorar regiões desconhecidas do espaço de busca. O operador *crossover* é aplicado a um par de cromossomos retirados da população intermediária, gerando dois cromossomos filhos. Cada um dos cromossomos pais tem sua cadeia de bits cortada em uma posição aleatória, produzindo duas cabeças e duas caudas. As caudas são trocadas, gerando dois novos cromossomos. A FIGURA 10 ilustra o comportamento deste operador.

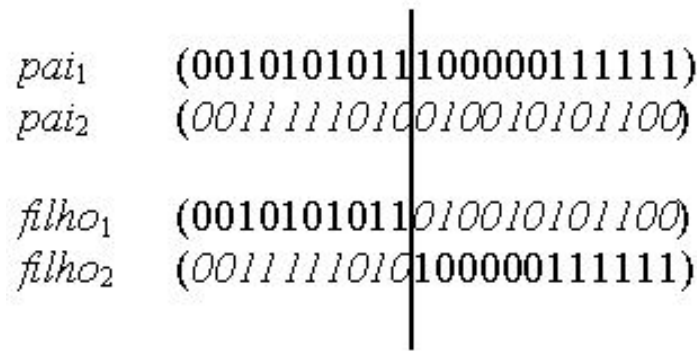


Figura 10 – Crossover

O *crossover* é aplicado com uma dada probabilidade a cada par de cromossomos selecionados. Na prática, esta probabilidade, denominada de **taxa de crossover**, varia entre 60% a 90% . Não ocorrendo o *crossover*, os filhos serão iguais aos pais (isto permite que algumas soluções sejam preservadas). Isto pode ser implementado, gerando números pseudo-aleatórios no intervalo [0,1]. Assim, o *crossover* só é aplicado se o número gerado for menor que a taxa de *crossover* (LACERDA; CARVALHO, 1999).

Após a operação de *crossover*, o operador de mutação é aplicado, com dada probabilidade, em cada bit dos dois filhos. O operador de mutação inverte os valores de bits, ou seja, muda o valor de um dado bit de 1 para 0 ou de 0 para 1. A FIGURA 11 apresenta um exemplo em que dois bits do primeiro filho e um bit do segundo sofrem mutação (bits estes que passaram no teste de probabilidade) A mutação melhora a diversidade dos cromossomos na população, no entanto por outro lado, destrói informação contida no cromossomo, logo, deve ser utilizada uma **taxa de mutação** pequena (normalmente entre 0,1% a 5% mas suficiente para assegurar a diversidade).

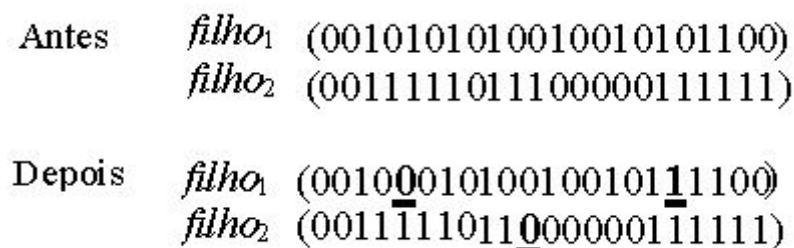


Figura 11 – Mutação

Segundo (LACERDA; CARVALHO, 1999, p. 99-148)

Após a definição da primeira população, o procedimento se repete por um dado número de gerações. Quando se conhece a resposta por um dado número de gerações. Quando se conhece a resposta máxima da função objetivo, pode-se utilizar este valor como critério de parada do algoritmo genético. Na FIGURA 12, é ilustrada a última geração, a vigésima quinta,

em que boa parte dos cromossomos representa pontos no pico do máximo global. Não há um critério exato para terminar a execução do GA. Porém, com 95% dos cromossomos representando o mesmo valor, é possível dizer que o algoritmo convergiu.

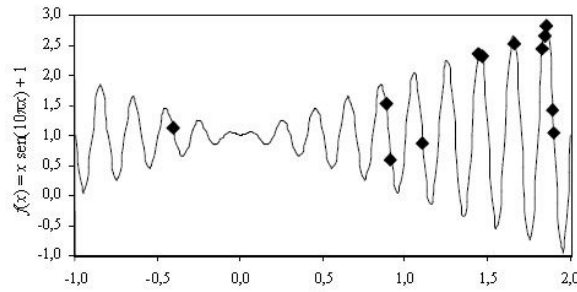


Figura 12 – Cromossomos da vigésima quinta geração

2.4.2 Simulated Annealing

Simulated annealing (SA) é uma técnica heurística utilizada em problemas de otimização combinatória. SA é como um algoritmo de descida (procura por um mínimo), que inicia com uma solução inicial, gerando outras soluções vizinhas, e calcula os custos de todas elas, se o custo for o menor, então esta nova solução é aceita, caso contrário a solução é descartada. O processo é repetido até que não haja melhorias. Às vezes, o ótimo obtido é local (não global) e uma forma de evitar isto é executar a técnica utilizando diversas soluções iniciais diferentes, e a solução final escolhida será a melhor dentre as soluções mínimas locais obtidas. A técnica *simulated annealing* permite a aceitação de uma configuração que forneça um “pior” valor para a função objetivo evitando, assim, a convergência para um mínimo local. Essa aceitação é determinada por um número aleatório e é controlada através de uma probabilidade.

2.4.2.1 Analogia ao Processo Físico

Quando um metal é aquecido até seu ponto de fusão, sua energia interna é alta e assim suas moléculas se movem rapidamente. Quando a temperatura é reduzida, as moléculas vão gradativamente diminuindo sua velocidade de movimento, à medida em que a energia interna também diminui. Assim, próximo ao ponto de congelamento, o metal se torna sólido, e o estado final das moléculas do metal são determinadas pelos seus comportamentos ou pela velocidade de resfriamento. O metal pode resultar em uma forma amorfa, sem uma forma definida como o vidro ou como um cristal com muitos defeitos em sua estrutura, quando o resfriamento for realizado de forma rápida, o que chamamos de processo quenching (esfriamento rápido). Ou, ainda, pode resultar em um cristal, onde todas as suas moléculas estão alinhadas e correspondem a uma configuração de mínima energia do sistema, quando o resfriamento é executado lentamente, chamamos de processo annealing (recozimento para uma recristalização). Essa analogia é resumida na tabela 3.

Tabela 3 – Analogia ao processo Físico

Processo Físico	Processo de Otimização
Estado	Solução
Energia	Custo ou Função Objetivo
Estado de Transição	Soluções Vizinhas
Temperatura	Parâmetro de Controle
Ponto de Congelamento	Solução Heurística

Analogamente a esse processo natural onde pode-se chegar à menor energia interna de sólidos, pode-se trabalhar com problemas de otimização combinatória. Os diferentes estados do sólido correspondem as diferentes soluções do problema, e a energia do sistema corresponde a função objetivo a ser minimizada.

De acordo com (EGLESE, 1990, p. 271-281)

Em 1953, Metropolis et al. apresentou um algoritmo que simula o processo de um grupo de átomos a uma dada temperatura, a procura do equilíbrio térmico. A cada iteração é calculada a diferença, δ , da energia do sistema, onde um átomo é aleatoriamente substituído. Se $\delta < 0$, a substituição é aceita, porém, se $\delta > 0$, a mudança é aceita com a probabilidade

$$\exp(-\delta/k * T) \quad (2.1)$$

onde T é a temperatura e k uma constante física chamada de constante de Boltzmann. Aplicando-se o algoritmo de Metropolis várias vezes, a cada temperatura são executadas várias iterações, e o sistema encontra o equilíbrio térmico para cada temperatura.

Assim, os diferentes estados do metal correspondem as diferentes soluções viáveis de um problema de otimização combinatória, sendo que a energia do sistema corresponde a função objetivo a ser minimizada. o primeiro a utilizar este algoritmo para otimização foi Kirkpatrick, em 1983, que denotou-o algoritmo *simulated annealing* (NARA K.. SIMULATED ANNEALING APPLICATIONS. IN:SONG, 1999).

No quadro é apresentado o relacionamento entre o processo físico e o processo de otimização funcional. Relações entre o processo físico e o processo de otimização funcional

Segundo (BARBOSA, 1989, p. 125-142)

Ao iniciar com uma temperatura relativamente alta, e com decréscimo gradual, espera-se que ao final do processo, o sistema estacione em um estado de energia globalmente mínima, por analogia com a física. Através dessa aplicação, pode-se ignorar a constante de Boltzmann, e reescrever a probabilidade de aceitação de uma configuração de energia mais alta como $\exp(-\delta/T)$.

Uma característica importante de *simulated annealing* é essa aceitação de configurações que apresentam maior energia, que pode parecer pior, porém permite que o método

não convirja para um mínimo local, podendo convergir para um melhor resultado (talvez o mínimo global).

Por exemplo, na FIGURA 13, encontra-se o gráfico de uma função $f(x)$. Suponha que deseja-se determinar o mínimo global B de $f(x)$ a partir do ponto x_0 . Simulated annealing pode convergir de x_0 para x_1 e depois de x_1 para B (mínimo global). Entretanto é possível que o processo de otimização convirja para x_2 e, assim, para A.

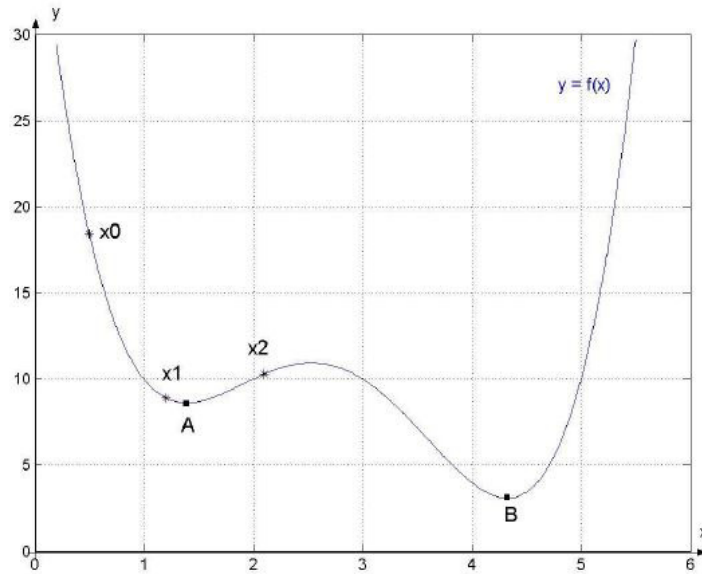


Figura 13 – Característica de busca do *simulated annealing*. Fonte: (NARA K.. SIMULATED ANNEALING APPLICATIONS. IN:SONG, 1999)

Ao começar com uma temperatura T alta, é aceita qualquer tipo de configuração e, a medida que o valor da temperatura decresce, as configurações que possuem uma maior energia tem diminuída a sua probabilidade de aceitação. Assim o processo, inicialmente, trabalha com uma enorme aceitação sem limitar as configurações. Portanto não tendendo caminhar apenas para um mínimo local.

A temperatura T diminui segundo a equação $T^{k+1} = \rho T^k$, onde ρ assume valores de 0,80 a 0,99, até alcançar o ponto de congelamento, quando o algoritmo pára. Este ponto de congelamento não necessariamente implica na temperatura zero, podendo ser considerada uma temperatura muito baixa pré-determinada.

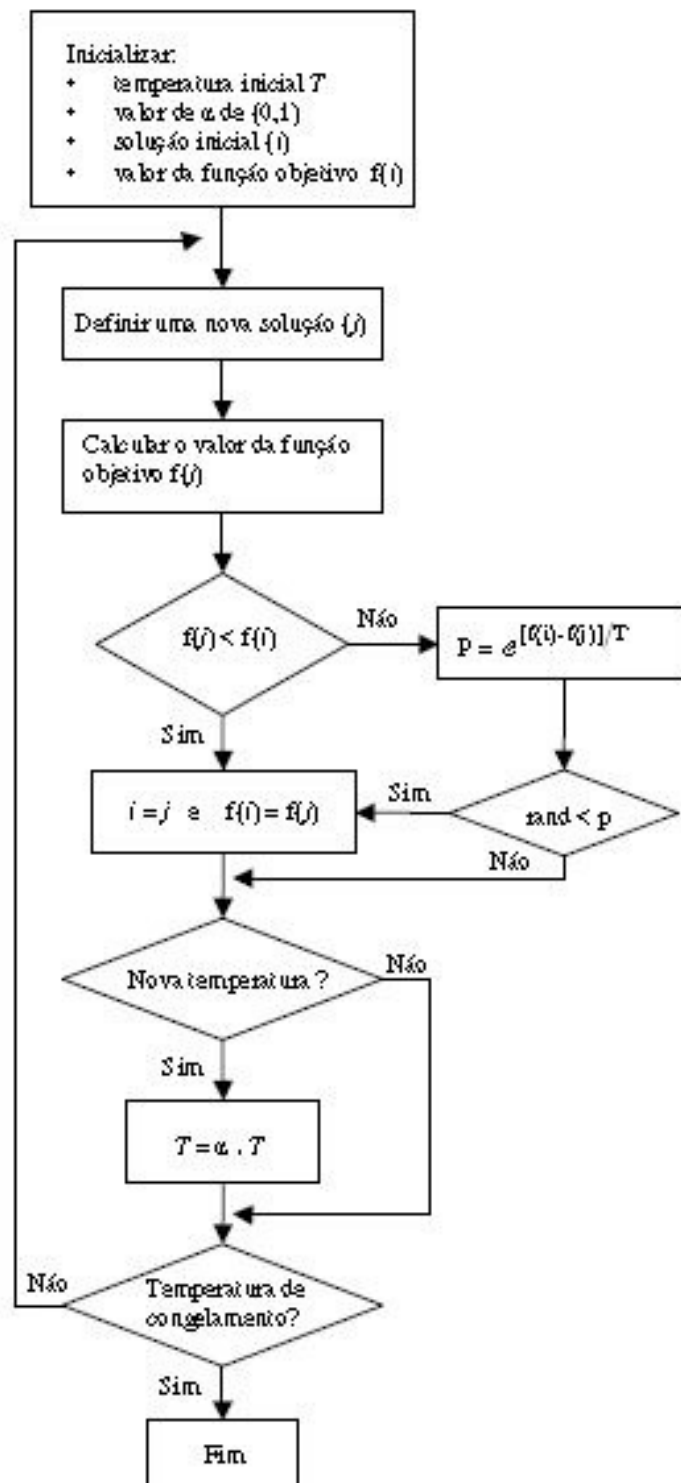
2.4.2.2 O Algoritmo Simulated Annealing

O algoritmo simplificado do *simulated annealing* é o seguinte:

```
1  inicio (i,energia(i),T, a)
2      enquanto (critério de parada)
3          para k=1:L (n° de iterações por nível de temperatura)
4              gerar vizinho(j de i)
5              calcular energia(j)
6              se (energia(j)<energia(i))
7                  então i =j
8                  energia(i)= energia(j)
9              senão
10                 calcular p = exp([energia(i)-energia(j)] / T)
11                 se (p>rand(0,1))
12                     então i =j
13                     energia(i)= energia(j)
14                 senão rejeitar j
15             fim {se}
16         fim {para}
17         T = a.T
18     fim {enquanto}
19 fim
```

Figura 14 – Algoritmo Simulated Annealing

A FIGURA 11 mostra um fluxograma do algoritmo *simulated annealing*.

Figura 15 – Fluxograma do algoritmo *simulated annealing*

Para o uso do *simulated annealing* no problema da determinação da sequência ótima de duplicação de uma ferrovia, por exemplo, são necessários alguns dados iniciais:

- quantidade de vias singelas
- tempos de percurso, em ambos os sentidos, de cada via singela

- quantidade de veículos ferroviários realizando a atividade fim de transporte de carga

Algoritmo:

Passo 1: Estima-se uma temperatura inicial T , permitindo um espaço de busca, de novas soluções, sem limitações. Através da configuração original obtém-se a energia(i) inicial.

Passo 2: Enquanto não for alcançado o critério de parada, vá para o Passo 3. Alcançado o critério de parada, vá para o Passo 8. O critério usado no trabalho está detalhado na próxima SEÇÃO.

Passo 3: Para k de 1 a L (número de iterações por nível de temperatura). Para que, de forma gradativa, haja o esfriamento do material até seu estado final de ‘congelamento’, ou seja, haja uma aumento na restrição da variação da pesquisa de uma nova solução, o valor da variável L é calculada por:

$L = \text{número máximo de iterações} / \text{número de trechos abertos}$

Após L iterações, vá para o Passo 7.

Passo 4: Gerar nova configuração de forma aleatória (j), sendo uma configuração factível, radial, não permitindo fechar o circuito ou interligar duas subestações. Calcular energia(j).

Passo 5: Se ($\text{energia}(j) < \text{energia}(i)$). Então $i=j$, vá para o Passo 3. Caso contrário calcular $p = \exp[\text{energia}(i) - \text{energia}(j)]/T$.

Passo 6: Se ($p > \text{rand}(0, 1)$) , onde $\text{rand}(0, 1)$ é um número aleatório uniformemente distribuído entre 0 e 1. Então $i=j$, vá para o Passo 3. Caso contrário rejeitar j , vá para o Passo 3.

Passo 7: Diminua o valor de T , de forma a diminuir o espaço de busca das novas soluções. Vá para o Passo 2.

Passo 8: Pare. A solução (i) é a melhor configuração da rede.

2.4.3 Busca Tabu

Busca Tabu é um procedimento de otimização local que admite soluções de piora para escapar de ótimos locais. Em sua forma original, a cada iteração procura-se um ótimo local selecionando-se o melhor vizinho s' da vizinhança $N(s)$ da solução corrente s . Independentemente de $f(s')$ ser melhor ou pior que $f(s)$, s' será sempre a nova solução corrente. Entretanto, apenas esse mecanismo não é suficiente para escapar de ótimos locais, uma vez que pode haver retorno a uma solução previamente gerada. Para evitar isso, o algoritmo usa o conceito de lista tabu. Esta lista define todos os movimentos com um certo atributo como sendo tabu por um determinado número de iterações, conhecido como tempo

tabu. Tais movimentos são proibidos a menos que a solução satisfaça a um certo critério de aspiração, em geral que essa solução seja melhor que a melhor solução encontrada até então. Os atributos são escolhidos para prevenir o retorno a soluções visitadas recentemente e são escolhidos por características que são fáceis para detectar.

O pseudocódigo do algoritmo é apresentado pela FIGURA 16 . Detalhes adicionais desse algoritmo podem ser encontrados em (GLOVER; LAGUNA, 1997).

```

procedimento  $BT(f(), N(), A(), |V|, f_{max}, |T|, BTmax, s)$ 
1   $s^* \leftarrow s;$            {Melhor solução obtida até então}
2   $Iter \leftarrow 0;$        {Contador do número de iterações}
3   $MelhorIter \leftarrow 0;$  {Iteração mais recente que forneceu  $s^*$ }
4   $T \leftarrow \emptyset;$     {Lista Tabu}
5  Inicialize a função de aspiração  $A;$ 
6  enquanto  $(f(s) > f_{max} \text{ e } Iter - MelhorIter < BTmax)$  faça
7     $Iter \leftarrow Iter + 1;$ 
8    Seja  $s' \leftarrow s \oplus m$  o melhor elemento de  $V \subset N(s)$  tal que
       o movimento  $m$  não seja tabu ( $m \notin T$ ) ou
        $s'$  atenda a condição de aspiração ( $f(s') < A(f(s))$ );
9     $T \leftarrow T - \{\text{movimento mais antigo}\} + \{\text{movimento que gerou } s'\};$ 
10   Atualize a função de aspiração  $A;$ 
11    $s \leftarrow s';$ 
12   se  $(f(s) < f(s^*))$  então
13      $s^* \leftarrow s;$ 
14      $MelhorIter \leftarrow Iter;$ 
15   fim-se;
16 fim-enquanto;
17  $s \leftarrow s^*;$ 
18 Retorne  $s;$ 
fim  $BT;$ 

```

Figura 16 – Algoritmo de Busca Tabu

2.5 Modelo matemático para cálculo de capacidade de tráfego em ferrovias

Para o estudo de caso da aplicação da metodologia DOTSIM em sistemas ferroviários nesta pesquisa, é fundamental conhecer as teorias da avaliação da capacidade de tráfego ferroviário por meio de modelos analíticos que são os amplamente utilizados. Antes de apresentar os principais trabalhos a respeito da definição da capacidade de tráfego, uma breve apresentação dos elementos de um sistema ferroviário é realizada na SEÇÃO seguinte.

2.5.1 Elementos do sistema Ferroviário

A infra-estrutura da malha é constituída dos seguintes elementos:

- Linha Singela: linha férrea que permite a circulação de trens em um único sentido por vez. A FIGURA 17 ilustra esta definição;

- **Linha Dupla:** linha férrea duplicada, que permite a circulação de trens nos dois sentidos simultaneamente ou a ultrapassagem de trens no mesmo sentido com velocidades diferentes. A FIGURA 17 ilustra esta definição;
- **Pátios Ferroviários:** São os locais da malha ferroviária onde se localizam as linhas de cruzamento, linhas de carga / descarga e linhas de atividade, onde o trem pode sair da linha de tráfego. Nestas linhas de atividades podem ser realizadas atividades de desmembramento de trens, formação de trens, troca de equipagem (maquinistas), abastecimento, manutenções preventivas em vagões e locomotivas ou aguardar em fila para carga ou descarga deste mesmo pátio ou em outro pátio. Portanto, os pátios ferroviários podem ser pátios de cruzamento, pátios de carga / descarga ou pátio de atividades. A FIGURA 18 ilustra esta definição;
- **Pátio de cruzamento:** Trechos de linha com duas ou mais linhas de tamanho fixo, permitindo que, em linhas singelas, os trens estacionem em uma das linhas, liberando a passagem para trens que estão circulando em sentido contrário. Para este caso o comprimento do trem tem ser menor que o tamanho do pátio. A FIGURA 19 apresenta o funcionamento do pátio de cruzamento;
- **Pátios de carregamento / descarga:** Pátios onde estão disponíveis linhas adicionais de carga ou descarga, onde o trem interrompe o seu movimento para realizar a operação. Estas atividades são feitas fora das linhas de trânsito, ou seja, os outros trens que circulam pela linha não sofrem a interferência desta atividade. Caso se tenha vários trens destinados para o mesmo pátio nas proximidades do ponto de carregamento, será formada uma fila aguardando que ficará aguardando a liberação para carregamento. Os pátios de carga e descarga podem também ter o formato de pêra ferroviária. Neste caso particular o trem entra, roda a pêra e sai sem alterar a ordem dos vagões e locomotivas da composição, reduzindo manobras de inversão de locomotivas necessárias para o retorno após o carregamento nos casos de pontos de carregamento sem pêra;

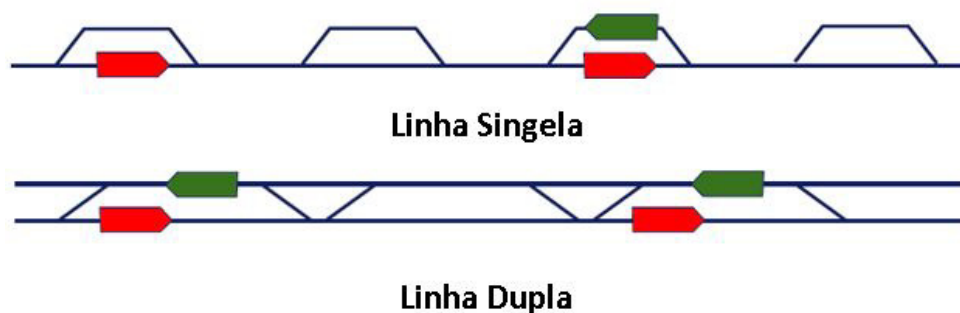


Figura 17 – Linha singela e linha dupla

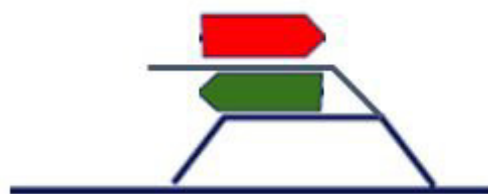


Figura 18 – Pátio Ferroviário



Figura 19 – Pátio de Cruzamento

- Segmento: São linhas ferroviárias triplas, duplas ou singelas que fazem a ligação entre pátios. Os segmentos podem ser sinalizados ou não. Se o segmento é sinalizado permite que um ou mais trens possam circular entre pátios de cruzamento, senão é sinalizado apenas um trem pode circular entre pátios de cruzamento. A FIGURA 20 representa segmentos de uma malha ferroviária;
- Entroncamentos: Convergência de três ou mais linhas da malha;
- Trechos: São os conjuntos de pátios e segmentos consecutivos, localizados entre dois entroncamentos.

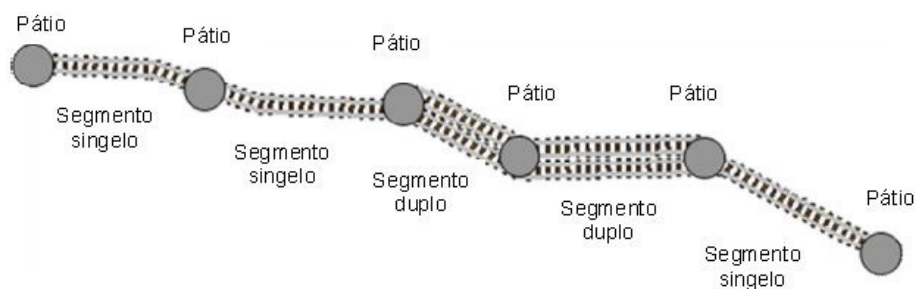


Figura 20 – Representação de segmentos de malha. Fonte: (MEIRELES, 2010)

2.5.2 Principais Trabalhos a respeito da definição da Capacidade de Tráfego

Os primeiros passos buscando uma metodologia de cálculo de capacidade de tráfego que obtivesse resultados mais refinados, com um tratamento matemático para os parâmetros ferroviários, foram iniciados com (FRANK, 1966). Ele dá um tratamento matemático para o tráfego em mão dupla numa linha singela, abordando a capacidade de

tráfego, o ciclo dos trens e o número de composições necessárias para efetuar o transporte num sistema de tráfego regular. Em seguida, (DREYFUS; LAW, 1977) apresenta alguns métodos de programação dinâmica para alocação de uma ou mais variáveis de modo a obter-se o maior aproveitamento da capacidade nas ferrovias.

Capacidade de tráfego ou vazão de um trecho ferroviário se dá pelo número de trens que poderão circular num determinado intervalo de tempo. São duas possibilidades de cálculo, sendo uma através do gráfico real de circulação dos trens, lançando nele uma sequência de trens de acordo com o horário formulado para cada um, estabelecendo um limite de capacidade. A capacidade pode também ser calculada analiticamente. Em ambos deve ser sempre respeitado o intervalo de espaço de licenciamento entre os trens.

Para o cálculo analítico de capacidade de tráfego, (BRINA, 1988) indica a necessidade de uso de um fator que diferencia as ferrovias conforme sua eficiência, variando de 60% a 80% denominado de “K”. Para ele, os fatores estruturais da via férrea são relevantes na análise de um sistema e a velocidade dos trens nas seções entre estações constituem um elemento fundamental na definição da capacidade de tráfego. De certa forma a velocidade, a estrutura da via e material rodante estão fortemente interligados. Os métodos analíticos que envolvem a circulação de trens periódicos, são utilizados para cálculo de capacidade de uma estação (nó) podendo ser calculada de maneira simplificada, levando em consideração as distâncias entre pontos de paradas, as velocidades nas seções entre as estações e os atrasos decorrentes dos cruzamentos entre trens sentidos opostos. Brina considera que a capacidade utilizada da linha é igual ao intervalo de tempo definido de análise, normalmente utilizando 24 horas, dividido pela somatória dos tempos entre desvios ou pátios de manobra consecutivos e a estes se soma os atrasos totais dos trens em função dos cruzamentos e ultrapassagens.

Para (BRINA, 1988), as fórmulas para cálculo da capacidade são aproximadas, e que uma maneira mais adequada para obtenção do número de trens em uma linha é através do diagrama espaço-tempo, acrescentando-se o máximo de trens possíveis em um período de tempo de 24 horas e cumprindo-se as regras operacionais de tráfego pré-estabelecidas.

(BRINA, 1988) apresenta o método de Colson para o cálculo da capacidade de tráfego por um período de tempo determinado e da capacidade de transporte pela programação e característica dos trens. Por fim, indica que a capacidade de tráfego de uma ferrovia pode ser aumentada, aumentando-se a velocidade dos trens e/ou diminuindo a distância entre os pátios de cruzamento/ultrapassagens, neste caso estará sendo diminuindo o comprimento dos trechos. Estas distâncias devem ser o mais iguais entre si, possibilitando o mesmo tempo de percurso. Nos casos onde o trecho tenha uma inclinação de rampa ascendente acentuada, esta distância deve ser menor, compensando a menor velocidade provocada e maior tempo de percurso. Vale ressaltar que além do mencionado pelo Brina, deve ser considerada a heterogeneidade do tipo de trens que também implica em diferença

de velocidade e tempos diferentes de percurso ([BRINA, 1988](#)).

Dependendo do contexto em que está inserido, o termo “capacidade” pode ser entendido de várias maneiras, uma vez que possui diferentes significados e aplicações. O termo capacidade é o máximo volume (trens diários) que pode ser movimentado em uma linha segundo um plano de operação, de horários e atividades (de tráfego e operacional) sem, no entanto, desrespeitar restrições preestabelecidas e sem exceder aos limites específicos. Na operação ferroviária, vários são os fatores que interferem na capacidade do sistema. Krueger desenvolveu um modelo paramétrico de capacidade para planejamento ferroviário capaz de preencher a lacuna existente entre as fórmulas empíricas e os modelos de simulação existentes. Estes parâmetros são subdivididos em parâmetros de geometria, tráfego e de operação. Ele apresenta fórmulas que quantificam os fatores que afetam a capacidade de um trecho ferroviário identificando os gargalos para aplicação de investimentos para o aumento do volume de tráfego. Por analisar a rede ferroviária como um conjunto de seções em linha singela, dotadas de trechos em linha dupla (pátios de cruzamentos e ultrapassagem de trens), o modelo não considera a relação entre os trechos interconectados entre si e a interferência dos horários dos trens que se cruzam nos pátios. A fórmula mais conhecida e citada por Krueger para cálculo de capacidade de malhas ferroviárias em linha singela deveu-se a Colson, um matemático belga que dedicou seu trabalho ao estudo de ferrovias ([KRUEGER, 1999](#)).

A próxima seção apresenta uma categorização de todos os parâmetros utilizados nos modelos matemáticos para cálculo de capacidade de tráfego, de acordo com o método de Colson.

2.5.3 Categorização dos parâmetros utilizados no modelo matemático

De acordo com ([BARROS, 2013](#), p. 62-87)

Para a análise de conteúdo aplicada aos dados, possibilitando a indicação dos tratamentos aos eventos não programados e a identificação do método analítico existente mais adequado, foram estabelecidas seis categorias. As categorias são constituídas pelos parâmetros ferroviários que foram relacionados ao tempo, caracterizando desempenho no cálculo da capacidade de tráfego, são elas:

- Categoria 1 – Tempo decorrente de deslocamento dos trens;
- Categoria 2 – Tempo decorrente de licenciamento e controle dos trens;
- Categoria 3 – Tempo decorrente de interrupção da circulação de trens para intervenção programada de manutenção;
- Categoria 4 – Tempo decorrente de interrupção da circulação de trens não programada.
- Categoria 5 – Tempo decorrente de gestão operacional;
- Categoria 6 – Tempo decorrente de recurso de via.

2.5.3.1 Conceituando as Categorias

2.5.3.1.1 Categoria 1 - Tempo decorrente de deslocamento dos trens

O tempo de deslocamento corresponde ao período de tempo gasto por um trem da partida de um determinado ponto da linha, origem, a sua chegada em outro ponto, destino. Predominantemente a via permanente e o material rodante, locomotiva e vagões, definem este tempo gasto com o deslocamento através da velocidade admissível deste conjunto (RIVES; PITA; PUENTE, 1977).

A velocidade de um trem é calculada basicamente pelo equilíbrio – igualdade – entre as forças de resistência total imposta aos veículos e a força de propulsão disponibilizada pelas locomotivas (ISLER, 2010).

Em seguida serão apresentados os parâmetros que compõem esta categoria e interferem no tempo de deslocamento.

2.5.3.1.2 Via Permanente

A via permanente é dimensionada, entre outros fatores, em função da velocidade dos trens que irá circular por ela. Velocidade maior implica em elementos geométricos e estruturais devidamente adequados e qualificados, compatíveis com os esforços gerados. As resistências ao deslocamento do trem, a via permanente contribui com a resistência das curvas e das rampas. As características planialtimétricas, geometria em planta e em perfil, da via permanente exercem forte influência no tempo de deslocamento dos trens. Estas características são definidas na fase de projeto e tem elevado custo quando passam por reformulação para adequação a um aumento de sua capacidade. O grau ou percentual de rampa ascendente provoca perda de velocidade na composição pela resistência da gravidade, exigindo às vezes até um reforço na quantidade de locomotivas ou de força tratora. A rampa descendente facilita o deslocamento do trem. Em alguns casos de longos trechos em aclave, são utilizadas as rampas de compensação, que são rampas construídas em sentido oposto a rampa existente com o propósito de “aliviar” o esforço de arrasto gerado pelas locomotivas, possibilitando pequeno ganho de velocidade ou até evitar a parada do trem.

As curvas provocam perda de velocidade pela resistência do atrito. Quanto menor este raio, maior é a retenção ou oposição ao deslocamento do trem provocado pela curva. Este feito se deve a associação do contato da roda dos vagões com os trilhos. Em contra partida quanto maior o raio de curva menor sua oposição à passagem do trem. A melhor condição é o raio infinito, que de fato é uma reta. A sequência de curvas eleva a retenção, estando o trem disposto com seus vagões sobre elas.

Os elementos de via possibilitam aos trens velocidades diferentes em função de suas características técnicas, a exemplo: trilhos, fixação, dormentes, lastro e AMV-Aparelhos de Mudança de Via.

2.5.3.1.3 Material Rodante - MR

O Material Rodante é composto por vagões e locomotivas e suas características tem substancial contribuição na velocidade e no tempo de deslocamento. Os vagões são veículos ferroviários rebocados destinados ao carregamento de carga e quando são destinados ao transporte de passageiro são denominados carros. Vários são os tipos de vagões e são dimensionados para diferentes cargas.

2.5.3.1.4 Categoria 2 - Tempo decorrente de licenciamento e controle dos trens

Tempo de licenciamento depende da tecnologia utilizada neste processo. A FIGURA 21 mostra licenciamento da estação A para estação B. Ele tem importante influência na definição da capacidade de tráfego. Este processo de emitir, receber ou cancelar a licença, de identificação do trem no trecho e controle do trem sucessor em mesmo sentido de circulação ou em cruzamento. O tempo com este processo pode gastar de milésimos de segundos, em processo automático a processos manuais que levam até 10 minutos ou mais. Multiplicando este tempo pela quantidade de trens que circulam, tem-se o nível de impacto do licenciamento e controle na capacidade de tráfego. Este tempo tende a interferir no *headway* aumentando o intervalo de tempo de um trem do seu sucessor. França (1991 apud BARROS, 2013) indica a forte dependência que tem a capacidade de tráfego com o sistema de licenciamento e controle.



Figura 21 – Exemplo de processo de licenciamento. Fonte: (BOARD; WARCHOT, 2011)

Vários são os métodos desenvolvidos de licenciamento e controle dos trens na via, porem o princípio se manteve igual. Ou seja, o deslocamento ou circulação de trens deve

ser feita de maneira rápida e segura. Deve ser garantido que tanto os trens no mesmo sentido ou em sentidos contrários não estarão licenciados ou terão permissão de ocuparem o mesmo espaço ou trecho de via ao mesmo tempo.

2.5.3.1.5 Categoria 3 - Tempo decorrente de interrupção da circulação de trens para intervenção programada de manutenção

A eficiência do transporte ferroviário está intrinsicamente ligado ao estado da via. E a política de manutenção determina o tipo e o volume de serviços a serem realizados, indicando sua localização, a sua duração e o nível de precisão que será atingido (NETO, 1985).

A interrupção na circulação de trens nos tempos destinados à manutenção programada de via permanente, e sua incidência depende principalmente do volume transportado (BRINA, 1982).

A velocidade dos trens determina o padrão de qualidade da manutenção. Maior velocidade dos trens, menor serão as tolerâncias admissíveis, maior será a incidência de intervenções e mais qualificadas.

A manutenção dos equipamentos e componentes da infraestrutura e superestrutura ferroviária tem como objetivo manter a disponibilidade da ferrovia, diminuindo as interdições e restrições de velocidade.

As manutenções da via permanente são em seus componentes ou elementos e correção geométrica.

2.5.3.2 Categoria 4 - Tempos decorrentes de interrupção da circulação de trens não programada

De acordo com a literatura, de 20% a 40% das manutenções da via permanente são não programadas. São componentes ou elementos de via que são danificados e são consertados, impactando nos desempenhos da circulação dos trens e na capacidade de tráfego (LABORATÓRIO..., 2012).

Exemplificando com uma ferrovia Brasileira ao longo de um mês de operação, ocorreram eventos que impactaram em sua capacidade de tráfego, dos não programados foram relacionados em seu controle: defeito no sistema de licenciamento de campo, defeito no equipamento de bordo de locomotiva em trem circulando, falta de energia elétrica fornecida pela concessionária, falta de maquinista, defeito de vagões em trem circulando, defeito eletromecânico de locomotivas em trem circulando, trilho quebrado em linha singela, manutenção de via permanente não programada em AMV (Aparelho de Mudança de Via), cliente não recebe trem, acidente ferroviário com descarrilamento em linha singela,

vandalismo de trem, aguardando outra ferrovia receber trem, pátio congestionado, entre outros.

2.5.3.3 Categoria 5 - Tempo decorrente da Gestão Operacional

O gerenciamento operacional adequado e o uso dos recursos com eficiência implicam em aumento da capacidade de uma ferrovia, mais que a expansão das estruturas físicas Nijkamp (1993 apud (ISLER, 2010)).

(ABRIL et al., 2008, p. 774-806) identificaram a heterogeneidade de tipos de trens e a tabela de horário como fatores relevantes na determinação de capacidade de tráfego provocando variações significativas.

2.5.3.4 Categoria 6 - Tempo decorrente de Recurso de Via

Esta categoria é formada pelas condições de recurso de linha singela, de linha dupla ou mais linhas de circulação com reflexo no tempo dos trens. Para (BURDETT; KOZAN, 2006, p. 616-632), "os atrasos provocados nos cruzamentos ou ultrapassagens de trens determinam a capacidade de processamento dos trens impactando na capacidade de tráfego, esta capacidade é determinada pela condição de linha singela ou de linha dupla".

A via com duas linhas pode ser unidirecional ou bidirecional, e ela trata com maior flexibilidade os cruzamentos ou ultrapassagens. A linha singela é bidirecional e trabalha o cruzamento ou ultrapassagem conforme definições operacionais de preferência ou prioridade de trens nos locais específicos denominados pátios ou desvios.

A conexão entre as estações ou pátio é feita pela linha de circulação. A linha singela permite apenas a circulação ou passagem de apenas um trem por vez por sentido de circulação. A linha dupla permite a passagem de dois trens em qualquer sentido de circulação ao mesmo tempo.

A distância e a quantidade de pátios de cruzamento entre as estações interferem na capacidade de tráfego. Quanto maior for o número de pátios de cruzamento entre as estações, maior será a possibilidade de cruzamento de trens em sentidos opostos e da passagem de trens uns pelos outros em mesmo sentido de deslocamento. A proximidade das estações ou dos pátios de cruzamento possibilita menor tempo de deslocamento, liberando mais rapidamente o trecho para a circulação de outros trens no mesmo sentido ou de menor tempo de espera de cruzamento de trens em sentidos opostos.

Quantidade de linhas de desvios nas estações ou nos pátios de cruzamento possibilita o recebimento de mais trens simultaneamente. Os trens de classes diferentes, de prioridades ou de preferências de circulação, interferem menos entre si, aumentando a capacidade de tráfego da ferrovia.

2.5.4 Modelo Matemático

O método analítico linear para cálculo de capacidade de tráfego para ferrovias é dado pela medida da *capacidade teórica* (FIGURA 22) e *prática*. A *capacidade teórica* é o número de trens que poderia ser executado em um determinado trecho da linha de referência de tempo definido, em caso de operação não perturbada, correspondente ao progresso para todas as classes de trens e planejamento operacional (KONTAXI; RICCI, 2010). (KRUEGER et al., 2000, p. 1191-1200) apud (ABRIL et al., 2008) conceituam a capacidade teórica como sendo

o número de trens que trafegam em uma rota, em um determinado período de tempo definido, em condições ideais, caracterizado por um cenário virtual em que os trens apresentam movimento constante com *headway* mínimo entre eles. Este parâmetro expressa o limite máximo da capacidade de tráfego, considerando que o movimento dos trens é homogêneo ao longo do dia, que estes são espaçados uniformemente e não há interrupções no sistema. Por ser um valor obtido através de fórmula empírica e não considerar os efeitos da variação do tráfego e das operações é impossível que seja praticado em condições reais. Sua equação é dada pela seguinte fórmula:

$$C = \frac{24}{T} \quad (2.2)$$

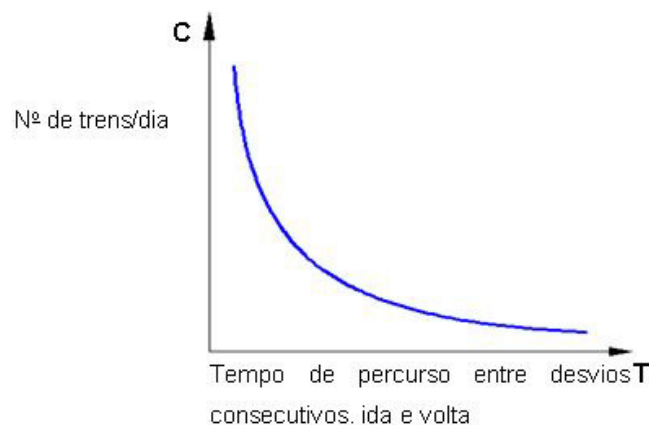


Figura 22 – Capacidade Teórica

Na EQUAÇÃO 2.2 o C representa a Capacidade Teórica em número de trens por dia ou em 24 horas, e o T representa o tempo de percurso entre desvios consecutivos, em ambos sentidos, ida e volta.

Ainda de acordo com (KRUEGER et al., 2000, p. 1194-1200) e (ABRIL et al., 2008)

A *Capacidade Prática* é o limite prático do número de unidades que podem se movimentar na linha ferroviária com certo nível de confiabilidade. Pelo fato de refletir as condições de circulações de diferentes tipos de

trens - cargueiros e de passageiros, com prioridades distintas, segundo o acúmulo de tráfego e outras condições do sistema, representa uma medida mais realista de capacidade. Como representa a combinação específica de infraestrutura, tráfego e operações para movimentar o volume máximo de produtos, com um nível de serviço e confiabilidade pré-determinados, é a medida mais significativa de capacidade do sistema ferroviário. Este parâmetro representa a capacidade que pode ser permanentemente fornecida sob condições normais de operação e representa aproximadamente 2/3 da capacidade teórica.

Eles definem a Capacidade Utilizada como a que representa o volume de tráfego e as operações realizadas, de fato, na linha ou rede ferroviária e, em geral, é menor que o valor da Capacidade Prática por diversos motivos tais como: variação do tempo de viagem entre estações nos ramais, atrasos no despacho dos trens nos terminais e pátios, atrasos decorrentes do volume de tráfego na via ou congestionamento, atrasos devido à eficiência dos equipamentos utilizados na operação, atrasos devido às restrições decorrentes das condições climáticas e vários outros eventos não programados ou indesejáveis. Por fim, definiu-se a Capacidade Disponível que representa a diferença entre a Capacidade Prática e a Capacidade Utilizada e corresponde ao volume de tráfego adicional, que pode ser inserido na rota de circulação dos trens. Havendo disponibilidade, a inserção de novos trens é considerada capacidade excedente, caso contrário, é considerada capacidade perdida (KRUEGER et al., 2000) e (ABRIL et al., 2008). Krueger representou graficamente a relação entre estes tipos de capacidade de tráfego ferroviário, conforme FIGURA 23 a seguir.

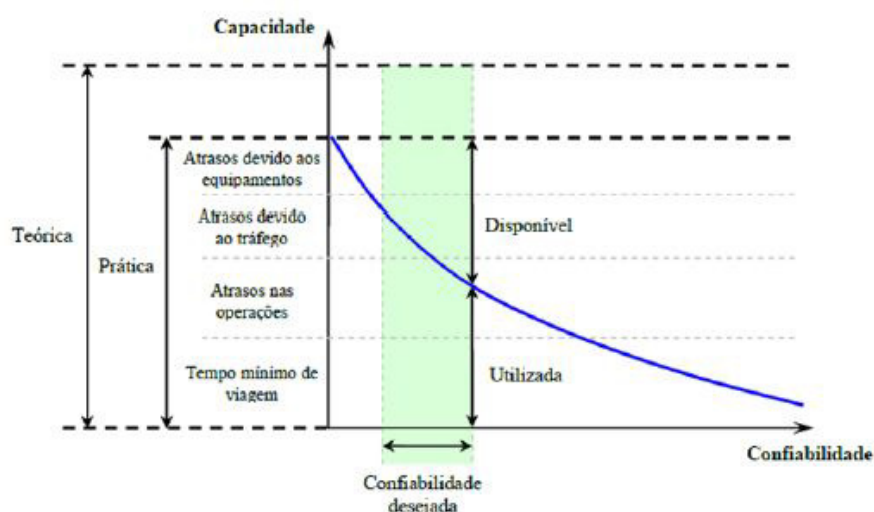


Figura 23 – Capacidade Prática

Além disso, as principais normas técnicas internacionais, indicam que a determinação da capacidade de uma ferrovia depende de diversos fatores que estão inter-relacionados, tais como, perfil da via, sistema de sinalização, manutenção programada e não programada, indisponibilidade de via por eventos diversos (falhas de via, de eletroeletrônica, de material

rodante, interdição por comunidades, etc), regularidade nos horários de chegada de trens, trem tipo, modelo operacional, eficiência operacional, tamanho dos pátios, distância entre pátios, entre outros (CODE, 2004).

A equação que representa o cálculo de capacidade prática analiticamente e de forma linear pelo método de Colson, segundo (BRINA, 1982) é dada pela seguinte fórmula:

$$C = k * \frac{24 - T_m}{T_s + T_d} \quad (2.3)$$

Na EQUAÇÃO 2.3 o T_m representa o tempo decorrente de interrupção da circulação de trens para intervenção programada de manutenção (Categoria 3 dos parâmetros dos MALs para sistemas ferroviários, seção 2.5.3).

O T_s e T_d representam o que é detalhado na categoria 1 ou tempo decorrente de deslocamento dos trens, e categoria 2 ou tempo decorrente de licenciamento e controle dos trens.

Para este método, (BRINA, 1982) indica a necessidade do uso de um fator que diferencia as ferrovias conforme sua eficiência, variando de 60% a 80% denominado de fator “K”. Além de (BRINA, 1982), (KRAFT, 1982) cita que este fator K pode variar entre 60% a 75% . (KRUEGER et al., 2000) apud (BARROS, 2013) cita que este parâmetro, o valor do fator K, representa a capacidade que pode ser permanentemente fornecida sob condições normais de operação e representa aproximadamente 2/3 da capacidade teórica.

O resultado desta equação da-se em *pares de trem por dia*, pois como mostrado na EQUAÇÃO 2.3, o *headway* (T_s e o T_d) representa o tempo decorrente de deslocamento do trem em ambos os sentidos (sentido de carregamento ou descarregamento do produto).

(ABRIL et al., 2008, p. 774-806) define headway como sendo

O maior tempo de percurso nos dois sentidos entre desvios de cruzamento consecutivos de comprimento maior do que o trem em circulação, podendo o headway variar em função do comprimento do trem. Define capacidade de tráfego teórico como sendo aquela na qual considera condições máximas ideais de tráfego, isto é, o número máximo de trens que podem trafegar em ambos os sentidos de uma linha, em um período de tempo, em condições ideais. Capacidade de tráfego real por adotar condições máximas reais, em substituição às condições ideais.

(ABRIL et al., 2008, p. 774-806) ainda define grau de utilização como sendo "o percentual que mede o nível de tráfego em relação à capacidade máxima."

2.6 Trabalhos Relacionados

Na literatura existem diversas pesquisas que utilizam da integração de técnicas de simulação e otimização (também conhecido com *simulation based optimization* - SBO

para solucionar problemas em ferrovias. Quatro dessas pesquisas são apresentadas nos próximos parágrafos:

No campo de pesquisa voltado para sistemas ferroviários, otimização baseada em simulação tem sido utilizada para buscar soluções em diferentes problemas relacionados com a concepção de operações ferroviárias e sua infra estrutura. Um método para um robusto problema chamado *timetable* ou escala horária de trens é introduzido por (KROON et al., 2008) o qual adota um modelo de otimização estocástica para alocar tempos e *buffer* de tempos em uma escala afim de torná-lo robusto contra distúrbios estocásticos operacionais durante operações reais.

(BEŠINOVIĆ; QUAGLIETTA; GOVERDE, 2013) propõe como alternativa a calibração de parâmetros em equações de movimento dinâmico de trens diante de dados reais de ocupação de vias de tráfego por meio de simulação computacional e utilizando algoritmos genéticos. No que se refere a componentes de projeto de infraestrutura é válido mencionar o trabalho de (GOMEZ et al., 2012) quem desenvolveu um simulador para projetar a configuração de seções de bloqueio as quais são consideradas soluções ótimas com respeito a confiabilidade e segurança. (QUAGLIETTA, 2014) propôs uma abordagem baseada em simulação para um *layout* ótimo para um sistema de sinalização em sistemas ferroviários o qual minimiza os custos totais para a gestão operacional e que seja também robusta com respeito a distúrbios aleatórios de operação.

No entanto, observam-se algumas lacunas nestes trabalhos, tais como, ausência de uma metodologia de desenvolvimento de processos que padronize a busca por soluções ótimas para o PSD, ausência de aplicação do SBO para o PSD e a ausência de pesquisa científica a respeito do PSD. Todas essas lacunas estão no escopo de contribuições da metodologia DOTSIM.

3 A metodologia DOTSIM

Neste capítulo é apresentada uma metodologia de desenvolvimento de processo que através de seu ciclo de vida contendo um conjunto de fases, processos e atividades e suas interações, e ainda da integração de técnicas inovadoras, tais como, algoritmos de otimização, simulação computacional e métodos analítico lineares, permite encontrar medidas de performance que atendam as expectativas do cliente, e portanto, desta forma, solucionar o PSD em complexos sistemas de transporte (ARAÚJO; VALE, 2017).

A definição da sequência ótima de duplicação em vias de sistemas de transportes de cargas consiste de um problema de complexidade intratável. Existem uma grande variedade de MHs capazes de gerar soluções satisfatórias. Entretanto, é fastidioso conhecer a MH que produzirá a melhor solução para um dado problema de sequenciamento de duplicação (PSD). Não há na literatura uma metodologia para estruturar, planejar e controlar algoritmos e processos na modelagem das variedades de MHs aplicadas para encontrar a solução ótima neste tipo de problema. Portanto, não há uma padronização, documentação à respeito das técnicas já aplicadas e tampouco um guia, roteiro, ou uma 'receita', que direcione pesquisadores, profissionais e empresas deste setor, na solução do PSD. Como consequências, empresas tem utilizado até de métodos analíticos lineares para tratar um problema de natureza complexa (ARAÚJO; VALE, 2017).

Espera-se através desta metodologia:

- Uma arquitetura com um conjunto de processos e interrelacionamentos entre os mesmos. Cada processo com sua sequência de passos, a fim de guiar os usuários na aplicação da linguagem de modelagem para realização de um conjunto de tarefas.
- Organização, padronização e a criação de uma metodologia que tenha flexibilidade de encontrar soluções do PSD para diversos sistemas de transporte de cargas (detalhado na próxima SEÇÃO);
- Gerar documentação a respeito do PSD, dado que o mesmo, atualmente, possui escassas pesquisas com foco em sua resolução.
- Indicação de outros métodos de otimização além dos analíticos lineares, para solução de um problema que naturalmente é intratável (PSD).

3.1 Visão Geral do DOTSIM

Esta metodologia também possui as seguintes características:

- o DOTSIM pode ser aplicado para diversos sistemas de transporte de carga tais como: ferrovias, VLTS's (Veículos Leves sobre Trilhos), vias expressas, rodovias, metrô, etc, os quais passam por um momento de necessidade de aumento de sua capacidade produtiva para suportar um aumento de demanda.
- A função objetivo tem origem de resultados de variáveis de um simulador computacional de eventos discretos e dinâmicos, dentro da estrutura de um algoritmo de otimização.
- A metodologia destina-se à pesquisadores, profissionais e aqueles leitores os quais tenham conhecimento básico de simulação e técnicas de otimização.

O DOTSIM integra 3 métodos para solução de problemas; algoritmos de otimização, simulação computacional e métodos analítico lineares (MALs). A FIGURA 24 ilustra como funciona essa interface, além de apresentar as entradas/saídas de cada macro processo:

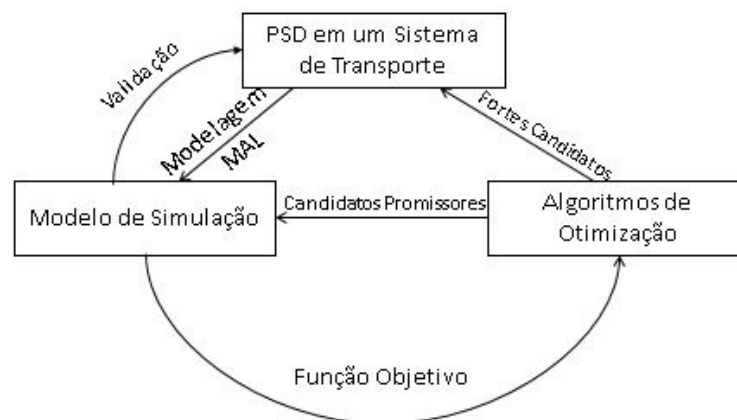


Figura 24 – Visão Geral do DOTSIM

O *PSD em um sistema de transporte* se trata do problema a ser otimizado conforme citado na SEÇÃO 1.1. O *modelo de simulação* é um modelo que literalmente imita o comportamento de um sistema de transporte. Este modelo pode ter influência de variáveis determinísticas ou estocásticas como entradas ou *inputs* tais como, quantidade de veículos no sistema, eventos de interdição de via planejados ou não, que geram indisponibilidade dos recursos. Este modelo é responsável por testar candidatos com diferentes sequências de duplicação oriundos de um *algoritmo de otimização* e do próprio sistema de transporte através do Método Analítico Linear, funcionando desta forma como uma função objetivo.

Os *algoritmos de otimização* são as MHs as quais irão escolher candidatos promissores que são enviados para o modelo de simulação e recebem um retorno do mesmo através da avaliação de uma ou mais variáveis, determinadas previamente pelo usuário de acordo com as características do problema. O tipo de variável também irá determinar

se a otimização busca uma maximização ou minimização da função objetivo. Ao final, os resultados de cada MH são comparados entre si, e posteriormente com o resultado do Método Analítico Linear (MAL) recomendado na literatura para o problema que está sendo otimizado.

A *Validação* é uma parte chave da metodologia por que ratifica que o modelo de simulação trabalha em concordância com objetivo do analista ou cliente final o qual deseja tomar uma decisão. A FIGURA 24 mostra que a validação do modelo de simulação está relacionado com o sistema de transporte em estudo, onde há uma necessidade de se evidenciar que existe aderência e coerência ao se utilizar aquele modelo de simulação para imitar o processo o qual está se estudando (detalhado na SEÇÃO 2.2.2).

A FIGURA 25 contém uma tabela onde são apresentadas as atividades e tarefas para cada processo e para cada fase do DOTSIM.

Fases	Processos	Atividade (Activity)	Nº Tarefa	Tarefa
Fase 1	Infraestrutura de via, pontos de carga e descarga	Definição de premissas de Infraestrutura de via	1	Definir quantidade de rotas singelas
		Processo de Carregamento e Descarregamento	2	Definir quantidade de pontos de carga
			3	Definir quantidade de pontos de descarga
	Tempos e movimentos, sistema de sinalização, intervenções de rotas programadas e não programadas	Definição dos tempos das rotas singelas e/ou duplicadas	4	Definir tempos ou velocidades das rotas singelas e/ou duplicadas
		Definição do sistema de sinalização	5	Obter com cliente o sistema de sinalização utilizado no sistema de transporte
		Definição do tempo de manutenção das rotas singelas (interrupções programadas)	6	Obter premissa de tempo e frequência de manutenção das rotas singelas com o cliente
		Definição das premissas de interdição de via não programado	7	Estabelecer o MTBF de todos os modos de falha do sistema com o cliente
			8	Estabelecer o MTTR de todos os modos de falha do sistema com o cliente
		Método Analítico Linear aplicado ao sistema em estudo	Definição de modelo matemático	9
Determinação da sequência ótima de duplicação com base no MAL		10	Determinar a sequência ótima de duplicação através do MAL	
Fase 2	Modelagem do sistema em ambiente ou biblioteca de simulação	Reconhecimento do problema	11	Identificar todos os fatos e aspectos que se tem pretensão de estudar
		Formular o problema	12	Selecionar os elementos do sistema global de interesse no estudo a ser realizado
			13	Fixar fronteira do novo sistema o qual engloba os elementos selecionados
			14	Definir os objetivos do estudo
			15	Selecionar o conjunto de parâmetros de medida de performance do sistema
			16	Estabelecer o horizonte de tempo
			17	Identificar os anseios do usuário final
		Obter e analisar dados do sistema	18	Levantar os valores das variáveis de entrada, parâmetros do sistema e medidas de performance
		Verificar e Validar o modelo	19	Certificar que as rotinas computacionais geram os valores esperados
			20	Comparar dados gerados pelo modelo com os obtidos no sistema real
		Documentar o modelo	21	Elaborar relatórios a respeito dos objetivos, suposições consideradas e detalhes do desenvolvimento das rotinas computacionais.
		Definir tipos de experimentos	22	Estabelecer para quais condições dos valores das variáveis de entrada e parâmetros do sistema os resultados gerados pelo modelo são confiáveis.
		Estabelecer condições de uso	23	Definir para quais condições é aplicável o modelo
	Material rodante e heterogeneidade	Definição de premissa de Material Rodante	24	Definir tamanho dos veículos e tipos de veículos de cargas (homogêneo ou heterogêneo)
			25	Definir quantidade de veículos no sistema na condição de saturação do sistema
	Meta Heurísticas	Seleção de MHs	26	Identificar e propor, através de pesquisa na literatura, as MHs capazes de gerar soluções satisfatórias para problemas desta natureza
	Integração do modelo de simulação com algoritmos de otimização	Codificação das MHs	27	Caso o ambiente de simulação seja FOSS, codificar os algoritmos de otimização (MHs) na mesma linguagem do ambiente de simulação
			28	Caso o ambiente de simulação não seja FOSS, retornar a ao processo de modelagem (também na fase 2) e implementar no modelo de simulação a geração de relatórios das medidas de performance e leitura de inputs em arquivos *.txt (de livre edição e consulta).
Fase 3	Avaliação de performance do MAL		29	Executar simulação para sequência ótima de duplicação obtida pelo MAL
	Algoritmos de otimização	Seleção de candidatos promissores	30	Selecionar candidatos promissores para avaliação de performance pelo modelo de simulação
	Modelo desenvolvido em ambiente ou biblioteca de simulação	Avaliação de performance	31	Executar simulação para candidato selecionado pela MH em N replicações e avaliar o F(x) ou medida de performance
	Atendimento de condição de Parada	Condição de parada	32	Determinar condição de parada (convergência, número de iterações, ou gatilho)
			33	Executar MHs até que condição de parada seja atendida
Seleção da sequência ótima de duplicação		34	Executar, gerar gráficos de comparação de resultados, e selecionar melhor candidato dentre as MHs e fazer comparação do resultado do MAL	

Figura 25 – Ciclo de Vida do DOTSIM

3.2 A metodologia DOTSIM

De acordo com (AVISON; FITZGERALD, 2003b, p. 78-82)

uma metodologia de desenvolvimento de software é definido com uma recomendada coleção de fases, procedimentos, regras, técnicas, ferramentas, documentação, gestão e treinamento utilizados para desenvolver um sistema.

Elas são mais fáceis de se entender quando descrito em duas fases principais (OM, 2003):

- Um conjunto de convenções de modelagem que compreende uma linguagem de modelagem (sintaxe e semântica);
- um processo, o qual:
 - Fornece uma orientação quanto à ordem das atividades;
 - Especifique qual artefatos devem ser desenvolvidos usando uma linguagem de modelagem;
 - Direciona as tarefas de desenvolvedores individuais e o time como um todo e por fim;
 - Oferece um critério para monitoramento e medição dos produtos e atividades de um projeto.

Enquanto a linguagem de modelagem fornece aos desenvolvedores um meio para modelar diferentes aspectos de um sistema, o processo determina quais atividades devem ser realizadas para desenvolver o sistema, em qual ordem e como. Em sua forma mais abstrata, um processo é uma sequência de passos - as vezes, depreciativamente, também chamada de receita - que objetiva guiar usuários em aplicar a linguagem de modelagem para realização de um conjunto de tarefas de desenvolvimento de software. O processo atua como a componente dinâmica e comportamental da metodologia, regendo o desenvolvimento técnico e gestão dos subprocessos, e portanto, abrangendo as fases, procedimentos, regras, técnicas e ferramentas prescritas pela metodologia, bem como as questões relativas à documentação e gestão de projetos. Também sugere-se que toda metodologia tem uma filosofia: o conjunto de suposições e crenças dos autores da metodologia que são refletidas na própria metodologia, e que apoiam e racionalizam sua utilização no contexto pretendido (AVISON; FITZGERALD, 2003a).

A fim de descrever como as atividades dos processos se relacionam umas com as outras para que se torne visível todo o processo de desenvolvimento (modelo de ciclo de vida), a metodologia DOTSIM segue alguns conceitos básicos descritos na norma internacional, ISO/ISEC 12207-*Software Life Cycle Processes* (SINGH, 1996). Esta norma foi escolhida pois ela tem sido uma boa prática no mercado mundial de softwares (PIGOSKI, 1996).

Baseado nessas necessidades, esta pesquisa propõe a metodologia DOTSIM. O nome tem origem das palavras-chave *Duplicação, Otimização, Sistemas de Transporte, Metodologia e Simulação*.

O ciclo de vida do DOTSIM inicia-se com uma abordagem e captura das necessidades do cliente e características a respeito do do PSD e o sistema a ser estudado. O fim de seu ciclo acontece com o processo de identificação e definição da sequência ótima de duplicação para o problema. Ao longo do ciclo, através de diagramas e tabelas, as atividades e processos são apresentadas, assim como, o modo como elas se relacionam umas com as outras para que se torne visível todo o processo de desenvolvimento.

No DOTSIM, seguindo diretrizes da ISO, todos os processos são formados por uma lista de atividades e toda atividade (*activities*) por um conjunto de tarefas (*tasks*) (FIGURA 26).

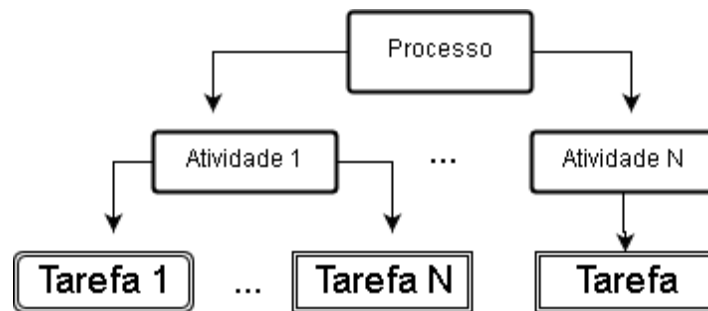


Figura 26 – A estrutura de um processo no DOTSIM.

Baseando-se dos princípios da norma ISO e do conceito de metodologia de desenvolvimento de software proposto por (AVISON; FITZGERALD, 2003b) e (OM, 2003), o DOTSIM divide-se em 3 fases (FIGURA 27 e ANEXO 1), cada um contendo seus respectivos processos: (i) Captura e Análise de requisitos; (ii) Verificação, Validação e Implementação; e (iii) Análise e Solução de Problemas.

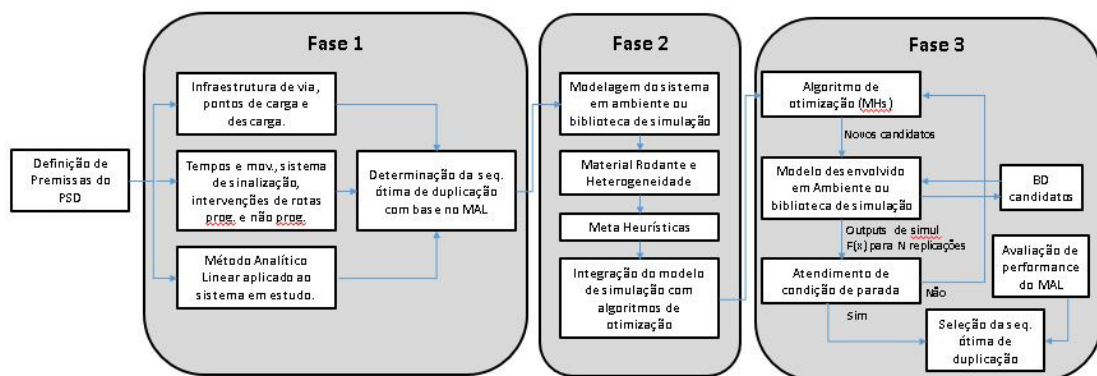


Figura 27 – Estrutura geral do DOTSIM (Também disponível no ANEXO 1)

A *fase 1* deve fornecer para *fase 2* os seguintes produtos:

- Um levantamento completo de todos os requisitos que serão utilizados na modelagem e simulação do sistema.

- A sequência ótima de duplicação definida através de um MAL.

A *fase 2* deve fornecer para *fase 3* os seguintes produtos:

- Um modelo de simulação implementado, testado e validado.
- As MHs que serão utilizadas na busca da solução ótima.

Nas próximas seções cada processo é apresentado e detalhado com suas respectivas atividades e tarefas necessárias, assim como sua função.

3.2.1 Fase 1: Captura e Análise de requisitos

Os processos desta fase são:

1. Infraestrutura de via, pontos de carga e descarga.
2. Tempos e movimentos, sistema de sinalização, intervenções de rotas programadas e não programadas.
3. Método analítico linear aplicado ao sistema em estudo.
4. Determinação da sequência ótima de duplicação através do MAL.

Na fase 1 todas as premissas e requisitos relacionados ao Problema de sequenciamento de duplicação (PSD) são capturados com o cliente. A ordem de execução deve ser de acordo com a enumeração acima.

Infraestrutura de via, pontos de carga e descarga

- *Inputs*: Planilha de dados/informações para cliente.
- *Outputs*: Quantidade de rotas singelas no sistema; quantidade de pontos de carregamento e descarregamento das cargas; caso existam rotas já duplicadas, quantidade e localização de todas as rotas ao longo do sistema.

Tempos e movimentos, sistema de sinalização, intervenções de rotas programadas e não programadas

- *Inputs*: Planilha de dados/informações para cliente.

- *Outputs*: Tempo de deslocamento ou velocidade entre rotas singelas, e/ou rotas já duplicadas, caso houver; tipo de sistema de sinalização (e.g. via rádio frequência, operação manual, sistema plenamente sinalizado, etc); tempo necessário e frequência (1 vez por semana, dias de semana, etc) para manutenção das rotas singelas e/ou duplas; discriminação de todos os modos de falhas que o sistema está suscetível (e.g. falha de via permanente, material rodante, eletroeletrônica, etc); MTBF - *Mean time between failures* ou tempo médio entre falhas e MTTR - *Mean time to repair* ou tempo médio para reparo das falhas. Estas duas últimas informações deve estar caracterizada por uma distribuição estatística de probabilidade.

Método Analítico Linear aplicado ao sistema em estudo

- *Inputs*: Para sistemas ferroviários e metroviários, as associações de referência são *UIC* (Européia) e *AREMA* (Americana). Para demais sistemas de transportes de cargas, é necessário realizar uma pesquisa na literatura em busca de associações de referência, ou até mesmo em documentação do DOTSIM.
- *Outputs*: Levantamento detalhado com métodos analíticos de capacidade de tráfego para o sistema em estudo.

Neste processo da fase 1 do DOTSIM deve-se realizar um levantamento detalhada na literatura dos métodos analíticos de capacidade de tráfego para o modal de transporte em estudo, utilizados por importantes organizações nacionais e internacionais naquela respectiva área. Este processo é fundamental, pois se trata de uma forma bastante conservadora de encontrar uma possível solução para o PSD, e serve de referência para os algoritmos de otimização que são utilizados ao longo da metodologia. Dado que se trata de um método analítico linear, e corriqueiramente tem sido aplicado para um problema naturalmente complexo, como já foi citado anteriormente, espera-se consequentemente que os resultados das MHs alcancem ganhos significativos em relação aos MALs. Apesar dessa característica, diversas empresas ainda têm utilizado o MAL para obter uma *solução ótima* para o PSD.

Para cada sistema de transporte haverá diferentes organizações nacionais e internacionais que fazem a gestão de todo o conhecimento científico a respeito de suas características, modelos matemáticos para determinação de capacidade de tráfego, normas e quaisquer outros assuntos técnicos que dizem respeito ao interesse de seus membros.

Para sistemas ferroviários, por exemplo, a organização internacional de referência para estudos de capacidade de sua infraestrutura é a *UIC* (*International Union of Railways*) que foi criada em 23 de novembro de 1921, na Itália, hoje com mais de 171 associados (companhias ferroviárias, operadores das linhas de caminho-de-ferro, gestores de

infraestruturas, fornecedores de serviços ferroviários, companhias de transportes públicos, etc.) dos mais diferentes países dos cinco continentes. A UIC procurou definir os parâmetros das características construtivas dos materiais rodante, locomotivas e vagões, e de via permanente. A partir da década de 1990 começou a trabalhar o sistema de telecomunicação de rádio e de licenciamento através de sistema telecomandado.

Seus principais objetivos são de facilitar o compartilhamento das melhores práticas entre os membros (benchmarking). Ajudar seus membros a desenvolver novos negócios e novas áreas de atuação. Propor novas formas de melhorar o desempenho técnico e ambiental. Promover a interoperabilidade, criar padrões mundiais para as ferrovias, inclusive normas comuns com outros modos de transporte. Desenvolver centros de competência para tecnologias de alta velocidade, segurança, E-Business e outros.

Determinação da sequência ótima de duplicação com base no MAL

- *Inputs*: Planilha de dados/informações para cliente.
- *Outputs*: Relatório com todas as rotas singelas do sistema e suas respectivas ordens de prioridade de duplicação de acordo com o MAL.

Por fim, uma sequência ótima de duplicação baseada no método analítico deve ser determinado, para posterior (fase 3) avaliação de performance através do modelo de simulação a ser desenvolvido na fase 2.

Neste processo do DOTSIM, as vias singelas de um dado sistema de transporte são priorizados com base na relação entre o que, (KRUEGER et al., 2000) apud (BARROS, 2013), chama de capacidade prática e a demanda daquela determinada via de tráfego.

No exemplo mostrado na TABELA 4 apresenta-se a aplicação da EQUAÇÃO 2.3 em um ferrovia hipotética com 5 vias de tráfego singelas. Há um claro desbalanceamento nas capacidades práticas (coluna 2), situação que é bem corriqueira em sistemas reais devido uma série de variáveis tais como, perfil altimétrico, influência de comunidades próximas as vias, existência de pontes e viadutos, etc (detalhado na categoria 4 dos parâmetros dos MALs para sistemas ferroviários).

A coluna 1 na TABELA 4 representa as seções singelas entre as estações, representadas pela abreviação PC (pátio de cruzamento), citado por (BRINA, 1982) como nós, desvios ou pátios de manobra (vide FIGURA 21), onde o trem poderá realizar cruzamentos (trens em sentido contrário) ou ultrapassagens (trens no mesmo sentido).

Diante dos resultados apresentados na TABELA 4, a sequência ótima de duplicação pode ser definida através da avaliação do saldo entre capacidade prática e demanda de cada via de tráfego. A priorização se dará pelas vias que possuem o menor saldo.

Tabela 4 – Capacidade Prática x Demanda da via (pares de trens/dia)

Via de Tráfego	Capacidade Prática	Demanda	Saldo
PC_0-PC_1	10.0	3.0	7
PC_1-PC_2	12.0	8.5	3.5
PC_2-PC_3	5.0	2.0	3
PC_3-PC_4	20.0	5.0	15
PC_4-PC_5	8.0	4.0	4

Tabela 5 – Sequência Ótima de Duplicação pelo MAL

Via	Prioridade
PC_0-PC_1	4
PC_1-PC_2	2
PC_2-PC_3	1
PC_3-PC_4	5
PC_4-PC_5	3

3.2.2 Fase 2: Verificação, Validação e Implementação

Durante a 2ª fase do DOTSIM, os seguintes processos são primariamente executados:

1. Modelagem do sistema em ambiente ou biblioteca de simulação.
2. Material Rodante e heterogeneidade.
3. Seleção das MHs.
4. Integração do modelo de simulação com os algoritmos de otimização.

Nesta fase, um dos principais objetivos é a garantia da fidelidade do modelo de simulação ao sistema real uma vez que ele funcionará como uma função objetivo através dos algoritmos de otimização. O critério de validação do modelo de simulação deve seguir as recomendações feitas por (BANKS; NORMAN, 1995), conforme citado na seção de estruturação de modelos de simulação (SEÇÃO 2.2.2).

Na fase 2 deve-se determinar quais *medidas de performance* o cliente deseja otimizar, através de uma maximização ou minimização. Para sistemas de transporte é muito comum a utilização da variável de tempo ou ciclo (um problema de minimização) ou produtividade, maior quantidade de carga descarregada (um problema de maximização). Essas variáveis são inversamente proporcionais, logo não faz sentido uma otimização visando estes dois objetivos paralelamente.

Além disso, as meta-heurísticas que vão auxiliar no processo de busca pela sequência ótima de duplicação devem ser selecionadas. O critério de seleção está diretamente ligado ao tipo de problema.

Por fim, deve haver a integração do modelo de simulação com as MHs selecionadas. A viabilidade da integração depende do ambiente de modelagem computacional, que pode ser black-box ou white box. Ambientes de simulação comerciais são consideradas black-box, tais como Arena®, Simul8® e possuem ferramentas de otimização que não são de livre pesquisa e codificação.

Enquanto que bibliotecas de simulação, tais como JSL e DSOL, os quais possuem código aberto, e são livremente licenciados para conceder a usuários o direito de uso, mudança e também melhoria em seu design através da disponibilidade de seu código fonte.

Modelagem do sistema em ambiente ou biblioteca de Simulação

- *Inputs:* Metodologia de modelagem e simulação de sistemas (SEÇÃO 2.2).
- *Outputs:* Modelo de simulação.

Neste processo é realizado a modelagem computacional do problema a ser otimizado através de bibliotecas ou ambientes de simulação. Conforme citado na SEÇÃO 2.2, a simulação terá como objetivo imitar o comportamento de sistemas e processos estocásticos que naturalmente são dinâmicos, aleatórios, contínuos ou discretos.

A decisão quanto a utilização de uma biblioteca ou ambiente de simulação deverá ser com base nas necessidade do cliente final da avaliação. São mais de 60 softwares disponíveis no mercado (SEÇÃO 2.2) e diversas bibliotecas, tais como, o DSOL e o JSL, ambas com licença de uso livre (não comerciais). Usualmente, o que vai direcionar esta decisão será um *trade-off* entre o custo da utilização da ferramenta de simulação e os benefícios em relação aos softwares comerciais, como os ambientes gráficos e integrados que normalmente não necessitam que o usuário entre com linhas de código para construir seus modelos de simulação. A utilização de softwares comerciais também permitirá o *design* de animação 2D e até mesmo 3D na simulação do modelo, algo que será bem mais complexo de se programar em uma biblioteca de simulação, demandando tempo e custo.

Outra informação relevante em relação aos ambientes de simulação, é que estes geralmente possuem ferramentas integradas de otimização, os quais são chamadas na literatura de "*black-box algorithms*" pois não permitem que o usuário tenha acesso a edições das técnicas e métodos utilizados. O Arena®, por exemplo, utiliza a ferramenta OptQuest® para buscar por soluções globais em problemas de otimização.

Nas bibliotecas de simulação temos a situação inversa, onde é possível utilizar quaisquer técnicas ou métodos, como as meta heurísticas, para buscar por soluções globais em problemas de otimização. Esses ambientes de desenvolvimento abertos ao programador ou usuário, são conhecidos na literatura por "*white-Box algorithms*".

Após a escolha da ferramenta de simulação a ser utilizada, todos os passos usuais citados na SEÇÃO 2.2.2 devem ser seguidos. É fundamental que nesta etapa levantem-se as medidas de performance do sistema. Por exemplo, em ferrovias, pode-se utilizar como medida de performance o ciclo dos trens (problema de minimização), a quantidade de trens que descarregaram o produto/carga em estudo na estação final (problema de maximização) ou uma função Custo C_t representada por um balanço financeiro do sistema (e.g despesas de capital de investimentos nos trechos duplicados, receitas geradas pela entrega do produto ao cliente final). Todas essas opções de medidas de performance são tratadas no DOTSIM como $F(x)$.

Após a conclusão de todas estas etapas no desenvolvimento de um estudo de simulação, é possível obter o $F(x)$ para a sequência ótima de duplicação obtida através do método analítico linear (MAL). Este resultado é designado pela letra grega η , e é fundamental pois serve como referência na avaliação da performance das MHs que serão implementadas nas etapas posteriores do DOTSIM.

Material Rodante e heterogeneidade

- *Inputs*: Planilha de dados/informações para cliente.
- *Outputs*: Quantidade de veículos de transporte no sistema a ser modelado e simulado.

Neste momento deve-se determinar o Material rodante (MR) a ser considerado no modelo de simulação, ou seja, a quantidade de veículos que estarão no sistema. A fim de evitar uma poluição dos resultados finais, avaliando-se fielmente a influência da infraestrutura de via na capacidade de transporte de cargas, é fundamental que a simulação seja executada em um regime de *stress* de MR. Isso quer dizer que o sistema deve estar saturado, e a quantidade de veículos não deve ser o gargalo operacional do sistema. (PINTO, 2012) deixa essa situação bem clara em um estudo de caso de um sistema ferroviário, quando a quantidade de trens descarregados alcança um pico e estabiliza mesmo com um incremento de trens no sistema.

Seleção de MHs

- *Inputs*: Pesquisa na literatura.

- *Outputs*: MHs capazes de identificar soluções globais para tal problema.

Conforme citado na SEÇÃO 2.4 o PSD é um clássico problema de permutação o qual é considerado como intratável pela literatura no ramo da otimização combinatória. (KANDA, 2014, p. 651-662) menciona na sua pesquisa que

As MHs mais utilizadas para problema similares (permutação) tais como o problema do cacheiro viajante (PCV) são: Busca-Tabu (TS), Greedy Randomized Adaptive Search Procedure (GRASP), Simulated Annealing (SA), genetic Algorithms (GAs), e o algoritmo de otimização da colônia de formigas (ACO, do inglês ant colony optimization algorithm).

Portanto, neste processo do DOTSIM, define-se quais destas MHs serão experimentadas e terão suas eficiências comparadas ao longo da pesquisa e implementação da metodologia.

Integração do Modelo de Simulação com algoritmos de Otimização

- *Inputs*: Informação do tipo de ambiente de simulação utilizado no processo de modelagem.
- *Outputs*: Ambiente integrado simulador - algoritmos de otimização.

Neste processo realiza-se a codificação das MHs. Ele basicamente se resume em duas atividades:

- Caso o ambiente de simulação escolhido no processo de modelagem e simulação seja *Free and Open Source* - FOSS, a implementação dos algoritmos (MHs) na mesma linguagem do ambiente de simulação ou mesmo projeto.
- Caso contrário (*softwares comerciais*), é necessário um retorno ao processo de modelagem e simulação, ainda na fase 2, e implementar a geração de relatórios das medidas de performance e leitura de *inputs*, de preferência, em arquivos *.txt (de livre leitura e consulta).

3.2.3 Fase 3: Análise e Solução de Problemas

Os processos envolvidos nesta última fase são sucintamente descritos da seguinte forma:

1. Avaliação da performance do MAL

2. Algoritmos de Otimização (MHs)
3. Modelo desenvolvido em ambiente ou biblioteca de simulação
4. Atendimento de condição de parada
5. Seleção da sequência ótima de duplicação

Esta fase inicia-se com a avaliação das medidas de performance diante da sequência ótima de duplicação identificada na fase 1 através do MAL. Esta avaliação é realizada com a utilização do modelo de simulação.

Uma busca exaustiva e robusta por um candidato ou sequência de duplicação promissor é realizada. A MH recebe a função de seleção dos melhores candidatos a serem avaliados pelo simulador, e o ambiente de simulação avalia a performance desses candidatos através da maximização ou minimização de uma variável selecionada na fase 1.

Um banco de dados armazena todos os candidatos testados, e o DOTSIM a cada iteração avalia se o mesmo já foi testado anteriormente afim de evitar a execução desnecessária e redundante, reduzindo o custo computacional.

Uma condição de parada deve ser satisfeita para encerramento de uma rodada de execução do DOTSIM. É importante que quaisquer condições estabelecidas para condição de parada, estejam atreladas a uma estabilização dos resultados, ratificando que houve uma busca exaustiva por candidatos promissores, assim como uma busca pulverizada através do espaço de busca, e evitar uma convergência para ótimos locais.

A solução global é finalmente identificada depois de avaliar o $F(x)$ de todos os candidatos fornecidos pelas MHs. O tempo computacional necessário para o algoritmo encontrar uma solução para o PSD dependerá sempre da velocidade do modelo de simulação, sendo que cada chamada à função objetivo (simulação) requer a avaliação de N replicações afim de garantir o alcance de uma precisão estatística satisfatória sobre os resultados desejados.

A sequência ótima de duplicação será o resultado da maximização ou minimização dos $F(x)$ das soluções globais identificadas pelas MHs e do candidato indicado pelo MAL.

Avaliação da performance do MAL

- *Inputs*: Sequência ótima determinar através do MAL na fase 1.
- *Outputs*: Resultados das medidas de performance avaliadas via simulação computacional.

Algoritmos de Otimização

- *Inputs:* Problema de Sequenciamento de Duplicação - PSD.
- *Outputs:* Sequências ou candidatos promissores a serem soluções globais ao problema.

Modelo desenvolvido em ambiente ou biblioteca de simulação

- *Inputs:* Candidatos a solução global para o PSD.
- *Outputs:* Resultados das medidas de performance.

Atendimento de condição de parada

- *Inputs:* Resultados das medidas de performance.
- *Outputs:* Resultado da avaliação de uma ou mais condições de parada de execução do algoritmo.

Seleção da sequência ótima de duplicação

- *Inputs:* Resultados das medidas de performance do MAL e das MHs.
- *Outputs:* Solução global para o PSD.

4 Uma aplicação do DOTSIM em diferentes instâncias de um Sistema Ferroviário de Cargas

4.1 O PSD para Ferrovias

Este capítulo apresenta um estudo de caso com a implementação da metodologia DOTSIM a fim de obter a sequência ótima de duplicação para 3 diferentes instâncias em um sistema ferroviário. As instâncias são definidas de acordo com a quantidade de vias singelas no sistema. Como citado na SEÇÃO 1.4, a complexidade do PSD e o espaço de busca cresce exponencialmente de acordo com a quantidade de vias singelas. Por este motivo, essa característica foi escolhida como critério para as instâncias.

Afim de proporcionar uma melhor descrição da problemática do PSD, a FIGURA 28 apresenta um exemplo hipotético de um sistema ferroviário com VLTs na cidade de São Luis-MA, Brasil. São 6 (seis) bairros e 1 (uma) cidade: Bacanga, Monte Castelo, Anil, COhab, Forquilha, São Cristovão e a cidade de São J. de Ribamar. Cada bairro/cidade possui sua estação de passageiros, e há no sistema um entroncamento na via singela de tráfego entre o bairro da Cohab e Forquilha em direção a cidade de São J. de Ribamar. Uma outra característica do problema é que todas as vias são singelas entre as estações.

Diante deste cenário apresentado na FIGURA 28, qual seria a sequência ótima de duplicação deste sistema, frente a um crescimento acentuado na demanda de passageiros? Seria mais apropriado iniciar a duplicação pela via 1 até a via 4, ou no sentido contrário?

A distância entre os pátios de cruzamentos (representado pelas estações de passageiros na configuração da FIGURA 28) são fundamentais para determinação da sequência ótima de duplicação. Portanto, afim de garantir uma maior aproximação com situações práticas, todas as instâncias do estudo de caso desta pesquisa consideram diferentes distâncias entre os desvios ferroviários definidos de forma arbitrária. Esta distância influenciará no tempo decorrente de deslocamento dos trens.

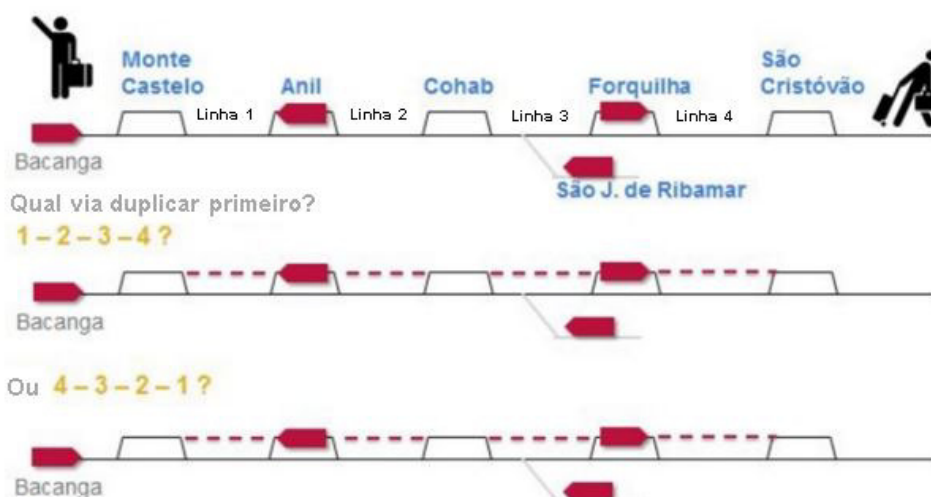
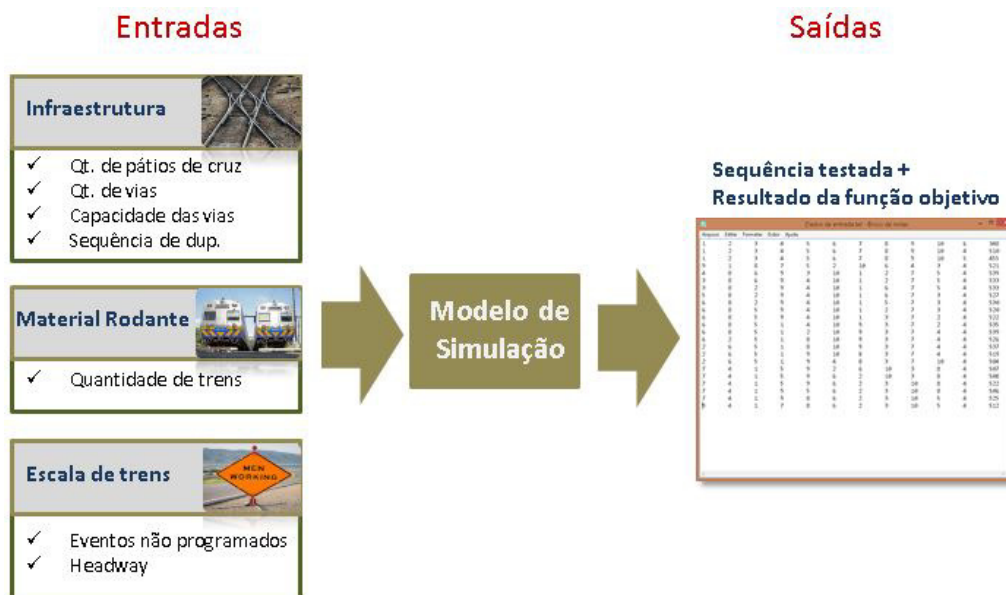


Figura 28 – Configurações de Vias de Tráfego

Além do tempo decorrente de deslocamento dos trens, outros parâmetros fundamentais que estão diretamente ligados a capacidade ferroviária (SEÇÃO 3.2.2) também são considerados no estudo de caso, tais como: (i) Tempo decorrente de interrupção da circulação de trens não programada; (ii) Tempo decorrente de licenciamento e controle dos trens e (iii) Tempo decorrente de Recurso de Via.



a via que é duplicada em um dado *step* da simulação. Ao final do período de simulação, toda a ferrovia deve estar 100% duplicada.

		ANALYTICAL SEQUENCE															
Segments	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	2	○	1	○	1	○	1	○	1	●	2	●	2	●	2	●	2
	3	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	4	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	5	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	6	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	7	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	8	○	1	●	2	●	2	●	2	●	2	●	2	●	2	●	2
	9	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	10	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2
		GA SEQUENCE															
Segments	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2
	3	○	1	○	1	●	2	●	2	●	2	●	2	●	2	●	2
	4	○	1	●	2	●	2	●	2	●	2	●	2	●	2	●	2
	5	○	1	○	1	○	1	●	2	●	2	●	2	●	2	●	2
	6	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	7	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	8	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	9	○	1	○	1	○	1	○	1	○	1	○	1	○	1	●	2
	10	○	1	○	1	○	1	○	1	●	2	●	2	●	2	●	2

Figura 30 – Exemplos de candidatos a avaliação de sequência ótima de duplicação

O *Material Rodante* - *MR* sinaliza a quantidade de vagões e locomotivas presentes no sistema. O simulador recebe essa informação em forma de entidades, e afim de simplificar como essa informação alimenta o simulador, para este estudo de caso, as entidades foram consideradas como *trens*. Conforme citado na SEÇÃO 3.2.2, o MR compõe uma das categorias que tem influência no cálculo de capacidade de tráfego. Portanto, em todas as instâncias deste estudo de caso, uma pré-avaliação de *stress* de MR naquele respectivo sistema é realizada de forma a evitar que a quantidade de vagões e locomotivas ou trens, limite a capacidade produtiva do sistema, uma vez que o objetivo da metodologia seja avaliar a capacidade de via em um processo de duplicação.

O terceiro parâmetro é o *Timetable*, composto por duas variáveis: o *headway* e as intervenções de circulação de trens não programadas. O *headway*, conforme citado na SEÇÃO 3.2.2, representa os tempos para cada via singela do sistema ferroviário, para cada sentido de deslocamento e para esta pesquisa, é aplicado para o trem de maior comprimento presente no sistema em estudo.

As intervenções de circulação de trens não programadas são representadas no modelo de simulação através de dois indicadores de confiabilidade: MTBF (*Mean Time Between Failures*) e o MTTR (*Mean Time to Repair*). No MTBF, a cada determinado ou estocástico período de tempo, um recurso de via sofrerá alguma intervenção ocasionada por falhas nos ativos de via permanente ou de Material Rodante (vagões e locomotivas).

O MTTR determinará quanto tempo deve durar estas falhas, e assim como o MTBF, pode ser tratada de forma determinística ou estocásticas.

A próxima seção apresenta a sequência de processos e atividades, de acordo com o

a metodologia DOTSIM, as quais são seguidas neste estudo de caso de forma a encontrar uma solução ótima para o PSD apresentado.

4.2 Aplicação do DOTSIM

A primeira de 3 fases da metodologia é composta por 4 processos e 8 atividades. A seguir são apresentados os resultados alcançados com a execução de todas as atividades e tarefas de cada processo da 1ª fase.

4.2.1 Fase 1

Definição de premissas de Infraestrutura de via

A fim de apresentar a eficiência da metodologia apresentada nesta pesquisa, o presente estudo avaliará 3 instâncias diferentes em relação a infraestrutura ferroviária: com 10, 20 e 30 rotas singelas.

Processo de Carregamento e Descarregamento

Neste processo, a quantidade de pontos ou terminais de carregamento e descarregamento de cargas são definidos com o cliente. É fundamental que independente da quantidade de terminais no sistema, os mesmos devem possuir uma capacidade produtiva superdimensionada ao longo do estudo, de forma a não 'poluir' os resultados para diferentes estratégias de duplicação das vias singelas, uma vez que o escopo da metodologia é solucionar um problema de duplicação de vias, e não de análise de capacidade de terminais de carregamento ou descarregamento de cargas. No entanto, sua modelagem é fundamental, uma vez que em sistemas reais sua influência na capacidade de tráfego é nítida por ser um local que serve muitas vezes de *buffer* e também como pátio de estacionamento para eventuais filas de processos. A FIGURA 31 ilustra um terminal de carregamento de trens, o qual utiliza silos para este fim.



Figura 31 – Silos de Carregamento (Vale, 2013)

Portanto, para esta pesquisa, a fim de garantir um foco dominante para a avaliação de duplicação de vias, considerou-se apenas um terminal de carregamento e um terminal de descarregamento, ambos com capacidade de estacionamento para 30 trens, e um tempo estimado de permanências do trens de 80 minutos para o terminal de carregamento e 65 minutos para o terminal de descarregamento.

Definição dos tempos das rotas singelas e/ou duplicadas

Este processo é composto pela tarefa de obter com o cliente os tempos de deslocamento entre pátios de cruzamento. A tabela abaixo apresenta os tempos de percurso nos dois sentidos, importação e exportação, entre desvios de cruzamento consecutivos de comprimento maior do que o trem em circulação ou *headway*, considerados para as 3 instâncias definidas anteriormente:

Vias	10 vias		20 vias		30 vias	
	Importação	Exportação	Importação2	Exportação3	Importação3	Exportação4
Via 1	Normal(38.9,1)	Normal(39.1,1)	Tri(28.7,29,36.8)	Tri(24.1,24,4,30.9)	Tri(28.7,29,36.8)	Tri(24.1,24,4,30.9)
Via 2	Normal(10.9,1)	Tri(30.4,40,5)	Tri(12.3,12,5,15.8)	Tri(17.4,17,6,22.4)	Tri(12.3,12,5,15.8)	Tri(17.4,17,6,22.4)
Via 3	Normal(15.7,1)	Normal(18.92,1)	Tri(7.4,7,5,9.5)	Tri(11.5,11,6,14.7)	Tri(7.4,7,5,9.5)	Tri(11.5,11,6,14.7)
Via 4	Normal(23,1)	Normal(10.66,1)	Tri(12.4,12,6,16.0)	Tri(17.0,17,2,21.8)	Tri(12.4,12,6,16.0)	Tri(17.0,17,2,21.8)
Via 5	Normal(15.7,1)	Normal(16.77,1)	Tri(11.9,12,15.2)	Tri(21.2,21,5,27.3)	Tri(11.9,12,15.2)	Tri(21.2,21,5,27.3)
Via 6	Normal(21.1,1)	Normal(7.8,1)	Tri(9.5,9,6,12.2)	Tri(13.1,13,2,16.8)	Tri(9.5,9,6,12.2)	Tri(13.1,13,2,16.8)
Via 7	Normal(12.5,1)	Normal(15.2,1)	Tri(12.9,13,1,16.6)	Tri(16.1,16,2,20.6)	Tri(12.9,13,1,16.6)	Tri(16.1,16,2,20.6)
Via 8	Normal(15,1)	Normal(9.88,1)	Tri(12.6,12,7,16.2)	Tri(16.7,16,8,21.4)	Tri(12.6,12,7,16.2)	Tri(16.7,16,8,21.4)
Via 9	Normal(11.2,1)	Normal(12.3,1)	Tri(10.1,10,2,12.9)	Tri(13.6,13,8,17.5)	Tri(10.1,10,2,12.9)	Tri(13.6,13,8,17.5)
Via 10	Normal(12.3,1)	Normal(9.9,2)	Tri(14.5,14,6,18.6)	Tri(25.2,25,5,32.3)	Tri(14.5,14,6,18.6)	Tri(25.2,25,5,32.3)
Via 11			Tri(8.7,8,8,11.2)	Tri(11.0,11,2,14.3)	Tri(8.7,8,8,11.2)	Tri(11.0,11,2,14.3)
Via 12			Tri(10.5,10,6,13.5)	Tri(17.2,17,4,22.1)	Tri(10.5,10,6,13.5)	Tri(17.2,17,4,22.1)
Via 13			Tri(5.7,6,7,6)	Tri(16.7,16,9,21.4)	Tri(5.7,6,7,6)	Tri(16.7,16,9,21.4)
Via 14			Tri(8.9,9,11,4)	Tri(11.1,11,2,14.2)	Tri(6.9,7,8,9)	Tri(6.9,7,8,9)
Via 15			Tri(11.3,11,4,14.5)	Tri(13.9,14,1,17.8)	Tri(6.9,7,8,9)	Tri(6.9,7,8,9)
Via 16			Tri(14.0,14,1,17.9)	Tri(19.7,19,9,25.3)	Tri(6.9,7,8,9)	Tri(6.9,7,8,9)
Via 17			Tri(11.4,11,5,14.6)	Tri(15.7,15,8,20.1)	Tri(11.4,11,5,14.6)	Tri(15.7,15,8,20.1)
Via 18			Tri(11.3,11,4,14.5)	Tri(17.9,18,1,23.0)	Tri(11.3,11,4,14.5)	Tri(17.9,18,1,23.0)
Via 19			Tri(12.2,12,3,15.6)	Tri(24.6,24,9,31.6)	Tri(12.2,12,3,15.6)	Tri(24.6,24,9,31.6)
Via 20			Tri(6.9,7,8,9)	Tri(11.5,11,6,14.7)	Tri(6.9,7,8,9)	Tri(11.5,11,6,14.7)
Via 21					Tri(12.43,12,56,13.69)	Tri(20.13,20,34,22.17)
Via 22					Tri(4.7,4,75,5.18)	Tri(12.03,12,15,13.24)
Via 23					Tri(8.95,9,04,9.85)	Tri(22.29,22,51,24.54)
Via 24					Tri(7.16,7,23,7.88)	Tri(14.12,14,26,15.54)
Via 25					Tri(10.49,10,59,11.55)	Tri(17.59,17,76,19.36)
Via 26					Tri(19.48,19,68,21.45)	Tri(22,22,22,24.22)
Via 27					Tri(16.73,16,9,18.42)	Tri(24.89,25,14,27.41)
Via 28					Tri(6.88,6,95,7.57)	Tri(17.34,17,51,19.09)
Via 29					Tri(8.48,8,57,9.34)	Tri(21.77,21,99,23.97)
Via 30					Tri(6.32,6,38,6.95)	Tri(15,15,15,16.51)

Figura 32 – Tempos de percurso nos dois sentidos, importação e exportação, entre desvios de cruzamento consecutivos de comprimento maior do que o trem em circulação ou *headway*

É importante reiterar que essa matriz de *headway* permite a utilização de funções do tipo *String*, as quais representam distribuições de probabilidade para variáveis estocásticas.

A fim de testar a *flexibilidade* da metodologia, os tempos são definidos de forma arbitrária para cada rota e para cada ferrovia (ou instância).

Definição do sistema de sinalização

Nesta atividade, um outro parâmetro ou condição operacional é obtido com o cliente: o sistema de sinalização. Conforme citado na SEÇÃO 3.2.2, a definição do sistema de sinalização vai definir o tempo decorrente de licenciamento e controle dos trens, uma vez que este tempo tende a interferir no *headway* aumentando o intervalo de tempo de um trem em seu sucessor. Além disso, o tempo com este processo pode gastar de milésimos de segundos, em processos automáticos, a processos manuais que levam até 10 minutos ou mais.

É possível encontrar diferentes tipos de sistemas ferroviários:

- Sistemas Ferroviários sem um sistema de sinalização e plenamente manuais
- Sistemas Ferroviários com um sistema de sinalização robusto, abrangendo toda a ferrovia.
- Sistemas Ferroviários mistos, parcialmente com um sistema manual, e a outra parte com um sistema sinalizado.

O presente estudo de caso abrange os sistemas 100% sinalizados, uma vez que sistemas inteligentes de transportes dependem fortemente das informações de localização. Além disso, sistemas sinalizados são fundamentais para garantir uma segurança operacional, evitando desta forma acidentes ferroviários.

Definição do tempo de manutenção das rotas singelas (interrupções programadas)

Esta atividade é responsável pela obtenção do tempo decorrente de interrupção da circulação de trens para intervenção programada de manutenção, com objetivo de manter a disponibilidade da ferrovia, e diminuindo as interdições e restrições de velocidade.

Esta variável consiste basicamente em duas informações que devem ser obtidas com o cliente: (i) quantidade de rotas singelas a passarem pelo processo de manutenção; e (ii) o tempo decorrente de interrupção. O (i) se trata de um informação de comportamento determinístico e o (ii) probabilístico.

A fim de reduzir a complexidade no processo de comparação dos resultados do MAL e das MHs para o PSD (fase 3), uma vez que se trata de uma variável, que caso seja considerada, deve estar presente em todas as instâncias, no modelo de simulação, inclusive na avaliação do MAL, o presente estudo de caso não tem como escopo a avaliação da influência desta variável. Além disso, este parâmetro funciona de forma comum a todas as rotas singelas, com um comportamento determinístico.

Definição das premissas de interdição de via não programado

A atividade de *Definição das premissas de interdição de via não programado* contempla resumidamente duas tarefas:

- Estabelecer o MTBF de todos os modos de falha do sistema com o cliente
- Estabelecer o MTTR de todos os modos de falha do sistema com o cliente

Distúrbios estocásticos representando eventos de interdição de via não programados são implementados no modelo de simulação com objetivo de representar as duas tarefas

acima. Uma rotina randômica é implementada para criar estes eventos que tornam os recursos de via de tráfego indisponíveis. Para tal, adotou-se uma taxa de 1,17 falhas por semana para cada via de tráfego e por semana de operação. Esta taxa de falha contempla todas as disciplinas ou modos de falhas que são corriqueiras em uma operação ferroviária tais como:

- Falhas de locomotivas;
- Falhas de vagões;
- Falhas de Equipamentos ou Máquinas de via (equipamentos de apoio operacional e de manutenção);
- Falha de Eletroeletrônica (aplicável apenas em ferrovias com sistema de sinalização);
- Falhas de Via Permanente;
- Falhas de operação (ocasionadas pela falha humana na operação de trens);
- Falhas ocasionadas por Acidentes ferroviários (descarrilamentos, tombamentos, abalroamentos, etc);
- Falhas devido incidentes (manifestação de populares, incêndio em dormentes, etc);
- E por fim, falhas relacionadas a projetos devido duplicação das vias.

Para o MTTR adotou-se uma curva Normal com uma média de 90 minutos e um desvio padrão de 10 % .

O tempo para atendimento e correção destas falhas não possuem um comportamento determinístico uma vez que uma série de fatores influenciam nesta variável, tais como: local onde aconteceu o evento de interdição de via, distância em relação ao pontos de socorro operacional ou emergência, disponibilidade de mão-de-obra, tipo de equipamento que falhou, etc.

Definição de modelo matemático

Nesta atividade é realizada uma pesquisa na literatura a procura de um modelo matemático para cálculo de capacidade de vias de tráfego do sistema de transporte em estudo.

Seguindo o roteiro estabelecido pelo DOTSIM, a referência global para estudos de capacidade em ferrovias de sua infraestrutura é a *UIC (International Union of Railway)* e *AREMA (American Railway Engineering Association)*.

Estas duas principais normas técnicas citam que a determinação da capacidade de uma ferrovia depende de diversos fatores que estão inter-relacionados, tais como, perfil da via, sistema de sinalização, manutenção programada, indisponibilidade de via por eventos (falhas de via, de eletroeletrônica, de material rodante, interdição por comunidade, etc.), regularidade nos horários de chegada de trens, trem tipo, modelo operacional, eficiência operacional, tamanho dos pátios, distâncias entre pátios, entre outros.

Conforme citado na SEÇÃO 3.2.2, (BARROS, 2013) comenta em sua pesquisa que

No Brasil é utilizada pela maioria das ferrovias a metodologia conhecida como “Fórmula de Colson” para a definição da capacidade de tráfego. Ela trabalha com o tempo disponível para o tráfego por dia de 24 horas ou 1440 minutos, o regime de licenciamento com intervalos de espaço onde um trem não pode sair de uma estação antes que seu precedente tenha atingido a estação ou sinal de bloqueio seguinte. Assim, considerando que a linha será percorrida por trens com a mesma velocidade “V”, o intervalo entre os trens será o tempo necessário para percorrer a distância máxima entre duas estações sucessivas.

Chamando de ‘d’ a distância, o tempo em horas será conforme a EQUAÇÃO 4.1 :

$$t = \frac{d}{v} \quad (4.1)$$

Sendo:

d em km,

v em km/h

Neste caso teórico onde a velocidade dos trens é a mesma, a capacidade de tráfego será conforme a EQUAÇÃO 4.2:

$$C_t = \frac{24}{t} = \frac{\frac{24}{d}}{v} = \frac{24v}{d} \quad (4.2)$$

Sendo C_t o número de trens que poderão circular em 24 horas, V a velocidade em km/h e d a distância máxima entre duas estações em km.

Adotando o tempo de percurso em minutos, o que é mais comum, tem-se a EQUAÇÃO 4.3:

$$C_t = 24 * \frac{60}{t} = \frac{1440}{t} \quad (4.3)$$

Considerando que em uma via férrea a circulação ou movimentação de trens acontece nos dois sentidos, tem-se a EQUAÇÃO 4.4:

$$C_t = 2 * \frac{1440}{(t_i + t_p)} \quad (4.4)$$

Na prática é o tempo de trens circulando em um sentido mais trens circulando no outro. Os trens que percorrem uma linha têm velocidades diferentes, os intervalos de percurso serão diferentes e a capacidade de tráfego será obtida considerando que outras condições interferem no processo. Logo, deve ser inserido na fórmula o tempo de licenciamento que varia de ordem escrita até o sistema informatizado.

Deve-se completar o cálculo considerando o tempo de licenciamento e outras perdas que neste caso denomina-se de fator redutor, ficando a EQUAÇÃO 4.5 indicando a capacidade prática:

$$C_p = k * \frac{2880 - t_m}{t_i + t_p + \theta} \quad (4.5)$$

Onde:

t_i = tempo de percurso máximo em minutos entre duas estações num sentido;

t_p = tempo de percurso máximo em minutos entre duas estações num sentido oposto;

θ = tempo de licenciamento considerando as características no sistema adotado de licenciamento e controle;

t_m = tempo destinado a manutenção com interrupção da circulação;

K = coeficiente ou fator redutor, que varia conforme a eficiência de cada ferrovia. É sugerido considerá-lo de 60% a 80% conforme “eficiência” de cada ferrovia.

(BRINA, 1988), recomenda que estes cálculos são aproximados e que caso tenha trens de passageiro circulando na mesma malha que outros trens de carga, deve-se utilizar o gráfico de horários de trens, grafar os trens de passageiros e encaixar entre eles os outros trens em maior número possível, desta forma obtém-se com maior exatidão a capacidade da malha ferroviária. A medida que se diminui as distancias entre as estações, ou instalando-se o sistema de bloqueio intermediário, subdividindo o trecho, a capacidade de tráfego aumenta, neste caso será o dobro. Em geral, se o número de seções que for dividido o trecho for “n”, a capacidade será “n” vezes maior.

Determinação da sequência ótima de duplicação com base no MAL

Esta atividade é responsável pela tarefa de determinar a sequência ótima de duplicação através do MAL identificado na atividade anterior.

Aplicando-se o modelo matemático para cálculo de capacidade prática de tráfego, através de seus respectivos tempos de *headways* e o sistema de sinalização, as seguintes sequências são definidas para cada instância:

		Método Analítico Linear															
Vias de tráfego	Via 1	○	1	●	2	●	2	●	2	●	2	●	2	●	2	●	2
	Via 2	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 3	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 4	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 5	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 6	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 7	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 8	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 9	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 10	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1

Figura 33 – Sequência de duplicação para instância de 10 vias singelas, considerando MAL.

		Método Analítico Linear																																			
Vias de tráfego	Via 1	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2		
	Via 2	○	1	○	1	○	1	○	1	○	1	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2
	Via 3	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 4	○	1	○	1	○	1	○	1	○	1	○	1	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2
	Via 5	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 6	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 7	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 8	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 9	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 10	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 11	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 12	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 13	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 14	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 15	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 16	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 17	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 18	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 19	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 20	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1

Figura 34 – Sequência de duplicação para instância de 20 vias singelas, considerando MAL.

		Método Analítico Linear																																		
Vias de tráfico	Via 1	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	
	Via 2	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 3	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 4	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 5	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 6	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 7	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 8	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 9	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 10	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 11	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 12	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 13	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 14	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 15	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 16	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 17	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 18	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 19	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 20	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 21	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 22	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 23	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 24	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 25	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 26	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 27	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 28	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 29	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○
	Via 30	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○

Figura 35 – Sequência de duplicação para instância de 30 vias singelas, considerando MAL.

A metodologia DOTSIM determina nesta tarefa que as vias a serem priorizadas são aquelas com o menor saldo entre capacidade prática e demanda em pares de trem por dia. De acordo com o MAL, o fator de eficiência operacional ou fator K, varia de acordo com a ferrovia, e portanto é comum à todas as vias de tráfego. Logo, não existe a influência deste parâmetro na determinação da sequência ótima através do MAL.

4.2.2 Fase 2

Esta fase resume-se em 4 processos: Modelagem do sistema em ambiente ou biblioteca de simulação, Material rodante e heterogeneidade, Meta Heurísticas e Integração do modelo de simulação com algoritmos de otimização.

Reconhecimento do problema

A atividade de reconhecimento do problema inclui apenas uma tarefa, o qual é responsável por identificar todos os fatos e aspectos que se tem pretensão de estudar para que estes sejam considerados na formulação do modelo.

Diante do que foi proposto como objetivos nesta pesquisa, os seguintes fatos e aspectos são relevantes e devem ser considerados na formulação do modelo:

- O modelo deve imitar o comportamento de um sistema ferroviário composto por 3 macro-processos: pontos de carregamento, ferrovia e pontos de descarregamento.

- O foco da pesquisa está na resolução de um problema (PSD) voltado para a linha tronco ou ferrovia, e suas respectivas vias de tráfego.
- Não é relevante o detalhamento dos processos inclusos nos pontos de carregamento e descarregamento.
- É importante que o modelo de simulação leia os *inputs* ou premissas em arquivo de texto.

Formulação do problema

Esta atividade contempla 6 tarefas:

- Selecionar os elementos do sistema global de interesse no estudo a ser realizado;
- Fixar fronteira do novo sistema o qual engloba os elementos selecionados
- Definir os objetivos do estudo
- Selecionar o conjunto de parâmetros de medida de performance do sistema
- Estabelecer o horizonte de tempo
- Identificar os anseios do usuário final

Baseados nos elementos de um sistema ferroviário apresentados na SEÇÃO 2.5.1, aqueles que serão de interesse neste estudo serão basicamente 3: o elemento que representa um terminal ou pátio de carregamento, um elemento contendo as vias de tráfego ou segmentos singelos, os desvios ou pátios ferroviários, e por fim 1 elemento contendo um terminal de descarregamento de cargas.

Em relação a 2ª tarefa para definir fronteiras, a FIGURA 36 apresenta claramente as fronteiras e os elementos presentes no escopo deste processo de modelagem.

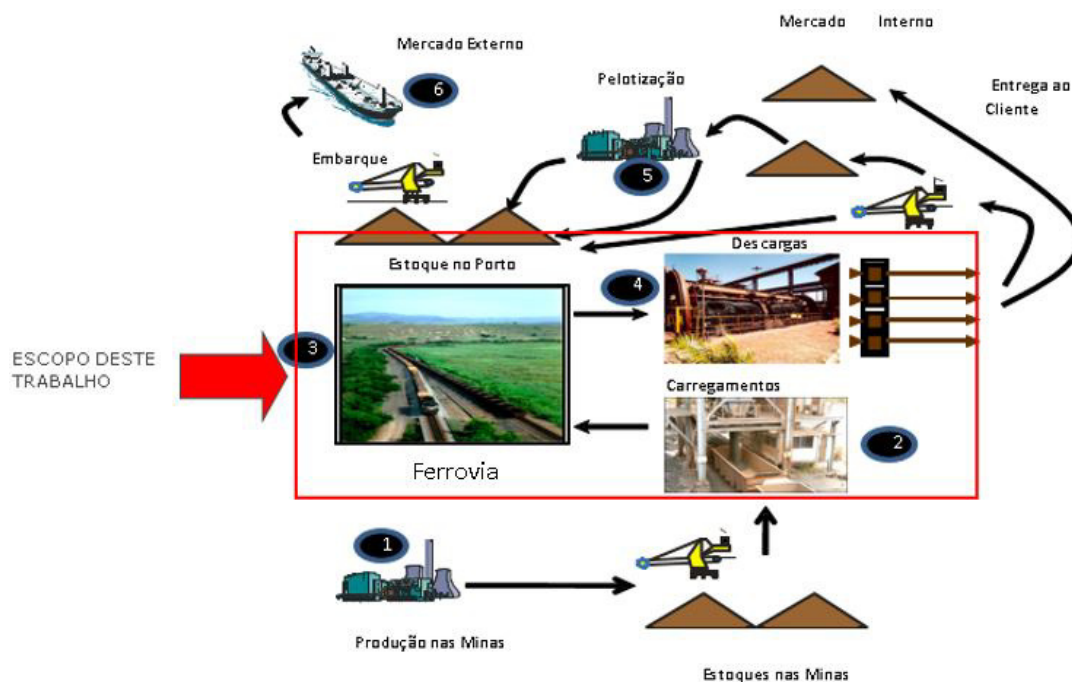


Figura 36 – Escopo do trabalho. Adaptado, (MEIRELES, 2010)

O objetivo da modelagem está associada ao cumprimento de um dos objetivos desta pesquisa que é de encontrar uma solução ótima para PSD através de métodos não-lineares, adequados a complexidade este tipo de problema (NP-Hard). Em outras palavras, o modelo computacional deve servir de função objetivo para os algoritmos de otimização.

Quanto aos parâmetros de medida de performance do sistema, espera-se após cada rodada de simulação a avaliação de apenas um parâmetro, que é o número de trens descarregados no ponto de descarga. A razão da utilização deste parâmetro nesta situação, se trata da principal finalidade de um sistema de transporte que é de garantir a entrega do produto ou da carga de um fornecedor, até o cliente final.

O horizonte de tempo que é estabelecido para esta modelagem é de 7 dias para cada segmento de via singela e para cada instância, com um tempo de aquecimento ou *warm-up* de 10 dias.

O usuário final espera o desenvolvimento de um modelo de simulação, do tipo entrada-saída, interativos, ao qual se fornece dados de entrada, obtendo-se respostas específicas para estes. Espera-se ainda que este modelo considere todas as premissas definidas na fase 1, gerando uma percepção que seu comportamento seja muito semelhante ao do sistema real.

Obter e analisar dados do sistema

Nesta atividade a seguinte tarefa deve ser executada: *levantar os valores das variáveis de entrada, parâmetros do sistema e medidas de performance*.

Os valores das *variáveis de entrada* neste estudo de caso são:

- A quantidade de vias singelas é definida de acordo com as instâncias (10, 20 e 30 vias singelas).
- Uma sequência candidata de duplicação da ferrovia em estudo, que é fornecida pelas MHs. Esta sequência é representada por uma lista com n posições, a identificação das rotas ou vias a serem duplicadas e com uma ordem de priorização com fluxo da esquerda pra direita.

Por exemplo, a lista $\{ 1,2,3,4,5 \}$ considera que a rota 1 será a primeira a ser duplicada, e a última será a rota 5.

- O *headway*, com seus respectivos tempos necessários para deslocamento do trem nas rotas singelas, em ambos sentidos. Este tempo já deve contemplar os tempos decorrentes do sistema de sinalização, de acordo com o que foi definido nas tarefas 4 e 5, fase 1.
- As premissas de MTTR e MTBF conforme definição nas atividades 7 e 8, fase 1.

Todas essas informações devem ser lidas pelo modelo em arquivo de texto, para facilitar a documentação.

A medida de performance ou $F(x)$ é dado pela quantidade de trens descarregados no terminal de descarga ao longo do período de simulação. Ou seja, a sequência ótima de duplicação será aquela que maximiza esta medida.

Verificar e Validar o Modelo

Nesta atividade deve-se confirmar que o modelo opera de acordo com intenção do cliente (sem erros de sintaxe e lógica) e que os resultados por ele fornecidos possuem crédito e sejam representativos dos resultados do modelo real. Duas atividades devem ser atendidas:

- Certificar que as rotinas computacionais geram os valores esperados;
- Comparar dados gerados pelo modelo com os obtidos no sistema real.

A validação do modelo conceitual a ser utilizado na modelagem deste estudo de caso é lastreada pela pesquisa realizada por (ARAÚJO, 2013). (ARAÚJO, 2013) utiliza um modelo

de simulação em Extend o qual representa a Estrada de Ferro Carajás - EFC, localizada entre os estados do Pará e Maranhão, para análise de capacidade e dimensionamento de Material Rodante. A ferrovia possui 56 pátios de cruzamentos, 57 rotas singelas, e interligada a um terminal de carga e um de descarga.

Documentar o modelo

A documentação do modelo é extremamente necessária pois serve como um guia para que alguém, familiarizado ou não com o modelo e os experimentos realizados, possa fazer uso do mesmo e dos resultados já produzidos. Além disso, se forem necessárias futuras modificações no modelo, toda a documentação existente vem facilitar os novos trabalhos.

A fim de garantir esta característica, o DOTSIM determina uma única tarefa para esta atividade que é de elaborar relatórios a respeito dos objetivos, suposições consideradas e detalhes do desenvolvimento das rotinas computacionais.

Primeiramente este modelo de simulação gera um banco de dados (FIGURA 37) em arquivo de texto e com todos os candidatos testados sempre que o modelo de simulação é solicitado.

Dados de entrada.txt - Bloco de notas

Arquivo	Editar	Formatar	Exibir	Ajuda							
1	2	3	4	5	6	7	8	9	10	6	348
1	2	3	4	5	6	7	8	9	10	4	510
1	2	3	4	5	6	7	8	9	10	5	455
9	1	8	7	5	2	10	6	4	3	4	521
4	8	6	9	3	10	1	2	7	5	4	539
3	8	6	9	4	10	1	2	7	5	4	533
3	8	2	9	4	10	1	6	7	5	4	533
5	8	2	9	4	10	1	6	7	3	4	527
6	8	2	9	4	10	1	5	7	3	4	530
6	8	5	9	4	10	1	2	7	3	4	520
6	8	5	9	4	10	1	3	7	2	4	522
6	8	5	1	4	10	9	3	7	2	4	535
6	8	5	1	2	10	9	3	7	4	4	539
6	2	5	1	8	10	9	3	7	4	4	526
2	6	5	1	8	10	9	3	7	4	4	537
2	6	5	1	9	10	8	3	7	4	4	519
2	6	5	1	9	4	8	3	7	10	4	504
7	4	1	5	9	2	6	10	3	8	4	507
7	4	1	5	9	6	2	10	3	8	4	508
7	4	1	5	9	6	2	3	10	8	4	522
7	4	1	9	5	6	2	3	10	8	4	506
7	4	1	9	8	6	2	3	10	5	4	525
9	4	1	7	8	6	2	3	10	5	4	512

Figura 37 – *Output* em arquivo de texto contendo candidatos ou sequências de duplicação e seus respectivos $F(x)$.

Outra documentação importante realizada nesta atividade são os dados de entrada. Estas variáveis são consolidadas também em arquivo de texto, e lidas pelo modelo de simulação. A FIGURA 38 apresenta um exemplo de dados de entrada.

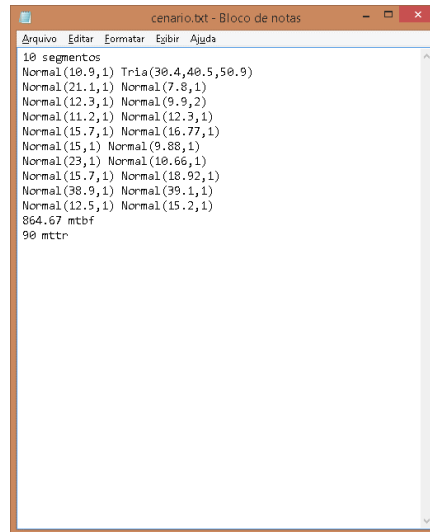


Figura 38 – *Inputs* em arquivo de texto como premissas a serem consideradas pelo modelo de simulação.

A FIGURA 38 apresenta um cenário a ser simulado contemplando 10 vias singelas, com seus respectivos *headways* no sentido importação e exportação, com um MTBF de 864.67 minutos e MTTR de 90 minutos.

A seguir são apresentados os passos utilizados para modelagem do problema através da biblioteca JSL:

Primeiro Passo: definir as variáveis de entrada e suas distribuição de probabilidade O problema exige a declaração de uma série de variáveis entre elas:

- O número de segmentos (FIGURA 39 - Linha 151) ;
- Tempo entre chegada de trens e (*input* para o comando de processo JSL - *Create*) (FIGURA 39 - Linhas 164 e 165);
- Declaração das distribuições de probabilidade para MTBF. Em termos de JSL, usa-se a classe Normal, que tem 2 parâmetros: valor médio e desvio padrão (FIGURA 39 - Linhas 170 e 171);
- Tempos de permanência nos terminais de carregamento e descarregamento, também utilizando a classe JSL Normal(FIGURA 40 - Linhas 185 e 187);
- Os tempos de *headway* são declarados através de um laço *for*, através da leitura dos dados de entrada do arquivo de texto (FIGURA 40 - entre as Linhas 189 e 197).

```

161     int upperbound = segmentos;
162     int lowerbound = 1;
163
164     DistributionIfc tecCliente = new Constant(300);
165     DistributionIfc primChegada = new Constant(0);
166     DistributionIfc tecUpdate = new Constant(1);
167     DistributionIfc primUpdate = new Constant(1);
168     //premissas para falhas na ferrovia
169     //System.out.println(matrizLeitura[segmentos+1][0]);
170     double MTBF = Double.parseDouble(matrizLeitura[segmentos+1][0]);
171     DistributionIfc tecFalhas = new Normal(MTBF, MTBF*0.1); //10% de Std Dev.
172     DistributionIfc primFalhas = new Constant(1);

```

Figura 39 – Definição das variáveis de entrada

```

185     RandomVariable processCarregamento = new RandomVariable(m, new Normal(80,0.1), "carregamento");
186     RandomVariable tempoPatioIda = new RandomVariable(m, new Constant(0.3), "tempoPatioIda");
187     RandomVariable processDescarregamento = new RandomVariable(m, new Normal(65,0.1), "descarregamento");
188
189     RandomVariable[] tempos_Ida = new RandomVariable[segmentos];
190     for(int i=0;i<=segmentos-1;i++){
191         tempos_Ida[i] = new RandomVariable(m, dist_matriz[i][0], "TTImport_Patio "+(i)+" Patio "+(i+1));
192     }
193
194     RandomVariable[] tempos_Volta = new RandomVariable[segmentos];
195     for(int i=0;i<=segmentos-1;i++){
196         tempos_Volta[i] = new RandomVariable(m, dist_matriz[i][1], "TTEExport_Patio "+(i)+" Patio "+(i+1));
197     }

```

Figura 40 – Definição das variáveis de entrada

Segundo Passo: declaração dos recursos Neste segundo passo todos os recursos envolvidos no problema são declarados. São eles:

- Terminais de Carregamento e Descarregamento. Na biblioteca JSL, é necessário declarar além do nome do recurso, a capacidade do mesmo (FIGURA 41 - Linhas 201 e 202);
- Segmentos ferroviários singelos (FIGURA 41 - Linhas 204 a 207). Os recursos serão definidos de acordo com a quantidade de segmentos lido do arquivo de *inputs*, e os mesmos são declarados através de um laço *for*. Além disso, todos os segmentos são declarados com capacidade igual a 1, de forma a representar que os mesmos são singelos;
- Pátios de cruzamento (em ambos os sentidos), conforme apresentado na FIGURA 41 - Linhas 209 a 216;

```

199 //recursos
200
201 Resource r1 = new Resource(m, 30, "recurso do carregamento");
202 Resource r2 = new Resource(m, 30, "recurso do descarregamento");
203
204 Resource[] linhas = new Resource[segmentos];
205 for(int i=0;i<=segmentos-1;i++){
206     linhas[i] = new Resource(m,2,"linha "+(i+1));
207 }
208 //linhas[7].setInitialCapacity(2);
209 Resource[] patiosIda = new Resource[segmentos];
210 for(int i=0;i<=segmentos-2;i++){
211     patiosIda[i] = new Resource(m,1,"Patio "+(i+1)+" Ida");
212 }
213 Resource[] patiosVolta = new Resource[segmentos];
214 for(int i=0;i<=segmentos-2;i++){
215     patiosVolta[i] = new Resource(m,1,"Patio "+(i+1)+" Volta");
216 }

```

Figura 41 – Definição das recursos do PSD

Declaração das filas Após a declaração dos recursos, suas respectivas filas também devem ser definidas. A FIGURA 42 apresenta a declaração das filas dos recursos segmentos ferroviários, pátios de cruzamentos e terminais de carga e descarga.

```

218 //filas
219
220 Queue q1 = new Queue(m, "Fila carregamento");
221 Queue q2 = new Queue(m, "Fila descarregamento");
222
223 Queue[] linhasQueue = new Queue[segmentos];
224 for(int i=0;i<=segmentos-1;i++){
225     linhasQueue[i] = new Queue(m,"Queue_linha "+(i+1)+" Ida");
226 }
227
228 Queue[] patiosIdaQueue = new Queue[segmentos];
229 for(int i=0;i<=segmentos-2;i++){
230     patiosIdaQueue[i] = new Queue(m,"Queue_Patio "+(i+1)+" Ida");
231 }
232 Queue[] patiosVoltaQueue = new Queue[segmentos];
233 for(int i=0;i<=segmentos-2;i++){
234     patiosVoltaQueue[i] = new Queue(m,"Queue_Patio "+(i+1)+" Volta");
235 }

```

Figura 42 – Definição das filas dos recursos

Rotina de circulação de trens Este passo busca pela modelagem do processo de licenciamento dos trens em um pátio ferroviário, de forma a garantir o deslocamento dos trens do ponto de descarga até o ponto de carga e vice versa, sem acontecer um *deadlock* (problema mostrado na FIGURA 43), ao longo de todos os segmentos.

A FIGURA 43 representa como funciona esse licenciamento:

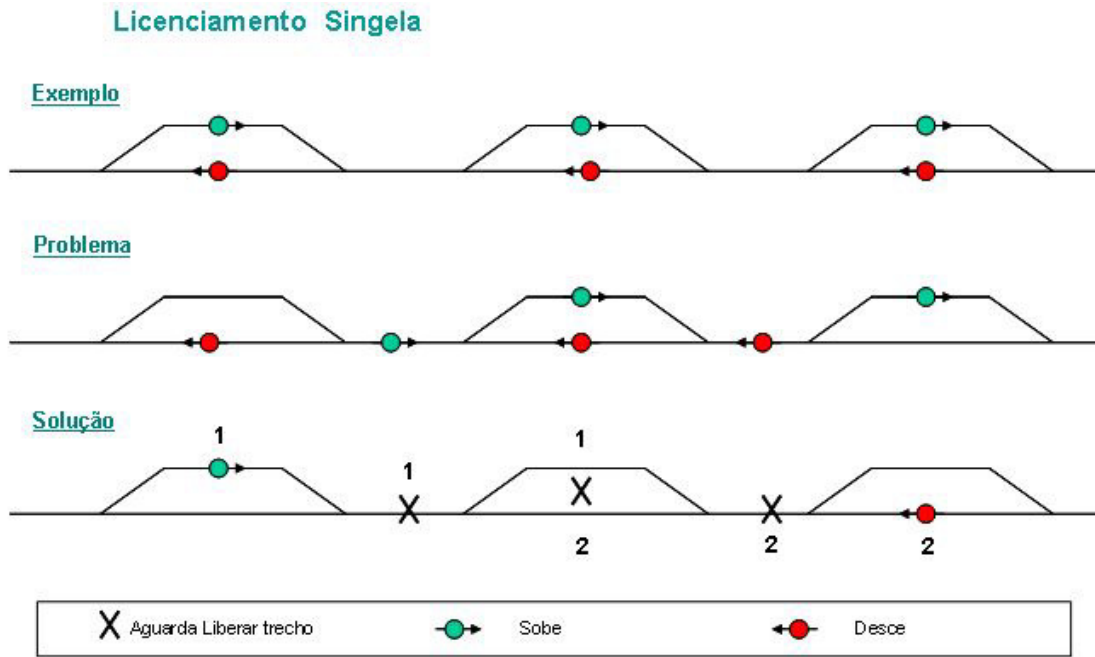


Figura 43 – Licenciamento de trens em uma via singela

A sintaxe do pátio ferroviário contempla a utilização de comandos de processos, tais como, prender (*seize*), atraso (*delay*), liberar (*release*), criação de entidade (*create*), entre outros. Estes comandos são utilizados para representar a movimentação do trem de acordo com a solução apresentada na FIGURA 43. No caso, o trem 1 (verde) solicita dois recursos para poder avançar: os dois 'x1' a sua frente, assim como o trem 2 (vermelho) no sentido inverso.

A FIGURA 44 apresenta a codificação do pátio ferroviário em JSL de acordo com os testes de licenciamentos apresentados na FIGURA 43. A biblioteca JSL tem uma excelente funcionalidade que é a possibilidade de se realizar indexação através de estruturas de repetição. Esta técnica é utilizada nas linhas 252 a 258 do código (FIGURA 44), onde um novo comando de processo chamado de *crossing loop*, o qual representa um pátio ferroviário com uma sequência de comandos de processo, pode ser repetido *n* vezes através de um laço *for*. O *Crossing Loop* é uma classe JSL criada e proposta nesta pesquisa, e funciona como um *template*, conforme FIGURA 45.

```

242 EntityProcessGenerator entityProcessGenerator = new EntityProcessGenerator(m, rail, primChegada, tecCliente, Lon
243 entityProcessGenerator.useDefaultEntityType());
244
245 //rail.addProcessCommand(new Process(m, v1, r1, q1, processCarregamento, "func carregamento"));
246 rail.addProcessCommand(new Seize(m, v1, r1, q1, 1, null));
247 rail.addProcessCommand(new DelayVariable(m, processCarregamento));
248 rail.addProcessCommand(new Release(m, r1, q1, null));
249 rail.addProcessCommand(new Seize(m, v1, linhas[0], linhasQueue[0], 1, null));
250 rail.addProcessCommand(new Seize(m, v1, patiosIda[0], patiosIdaQueue[0], 1, null));
251 rail.addProcessCommand(new DelayVariable(m, tempos_Ida[0]));
252 for(int i=0; i<=segmentos-2; i++){
253     if(i==segmentos-2){
254         rail.addProcessCommand(new crossing_loop(m, "Patio "+(i+1)+" Ida", linhas[i], linhasQueue[i], tempoPatioIda,
255             break;
256     }
257     rail.addProcessCommand(new crossing_loop(m, "Patio "+(i+1)+" Ida", linhas[i], linhasQueue[i], tempoPatioIda, lin
258 )
259 rail.addProcessCommand(new Release(m, linhas[segmentos-1], linhasQueue[segmentos-1], null));
260 rail.addProcessCommand(new DelayVariable(m, processDescarregamento));
261 rail.addProcessCommand(new Release(m, r2, q2, null));
262 rail.addProcessCommand(new RecordCounter(m, "1", "conta descarregamentos"));
263 rail.addProcessCommand(new Seize(m, v1, linhas[segmentos-1], linhasQueue[segmentos-1], 1, null));
264 rail.addProcessCommand(new Seize(m, v1, patiosVolta[segmentos-2], patiosVoltaQueue[segmentos-2], 1, null));
265 rail.addProcessCommand(new DelayVariable(m, tempos_Volta[segmentos-1]));

```

Figura 44 – Licenciamento de trens em uma via singela

```

17 * @author heygon
18 */
19 public class crossing_loop extends MacroCommand {
20
21     public crossing_loop(ModelElement parent, String name, Resource LinhaAnt, Queue LinhaAntQueue,
22         RandomVariable tempoPatioIda, Resource LinhaPost, Queue LinhaPostQueue, Resource PatioPostIda, Queue PatioPostI
23         , Resource PatioAtualIda, Queue PatioAtualIdaQueue
24         , RandomVariable HeadwayIda) {
25
26         super(parent, name+".crossingloop");
27
28         Variable req = new Variable(parent, 1, null);
29
30         this.addSubCommand(new Release(parent, LinhaAnt, LinhaAntQueue, null)); //libera linha anterior
31         this.addSubCommand(new DelayVariable(parent, tempoPatioIda)); //tempo dentro do patio
32         this.addSubCommand(new Seize(parent, req, LinhaPost, LinhaPostQueue, 1, null)); //solicita rota para ir ao prox patio
33         this.addSubCommand(new Seize(parent, req, PatioPostIda, PatioPostIdaQueue, 1, null)); //solicita rota para ir ao prox patio
34         this.addSubCommand(new Release(parent, PatioAtualIda, PatioAtualIdaQueue, null)); //libera o patio atual no sentido Ida
35         this.addSubCommand(new DelayVariable(parent, HeadwayIda)); //headway mov ate o prox patio (ida)
36
37     }
38
39 }
40

```

Figura 45 – Classe JSL proposta pelo autor - Crossing Loop

Rotina de geração de falhas de via não programados Por fim, uma rotina de geração de eventos de interção de via não programados, discretos e contínuos, é implementado.

Esta rotina utiliza dos comandos de processos: *create*, *seize* e *release* a fim de geração, interrupção de tráfego e consequentemente dos recursos que os representam.

```

289 //eventos de interdição de via
290
291 EntityProcessGenerator entity3 = new EntityProcessGenerator(m, falhas, primFalhas, tecFalhas, "falhas");
292 entity3.useDefaultEntityType();
293
294 int local = (int) (Math.floor((upperbound-lowerbound+1)* randInt(0, 1) + lowerbound));
295 falhas.addProcessCommand(new Seize(m,v1,linhas[local-1],linhasQueue[local-1],1,null));
296 RandomVariable delay_MTTR = new RandomVariable(m, new Normal(90,9), null);
297 falhas.addProcessCommand(new DelayVariable(m,delay_MTTR));
298 falhas.addProcessCommand(new Release(m, linhas[local-1], linhasQueue[local-1], null));
299
300 falhas.addProcessCommand(new Terminate(m,true,"dispose_falhas"));
301
302 rail.turnOnAutoStart();
303
304 return m;
305
306 }
307
308 }

```

Figura 46 – Rotina responsável pela geração de evento de interdição de via.

Na FIGURA 46 as linhas 291 e 292 representam o comando de processo *create*, seguido de seus respectivos parâmetros. O *tecFalhas* representa o tempo entre falhas, e é uma variável que é lida do arquivo de texto, contando os dados de entrada.

Na linha 294 um *local* é sorteado arbitrariamente, de acordo com o número de segmentos da ferrovia. Nas linhas posteriores, este local sorteado sofre um *seize*, gerando uma solicitação do recurso, um *delay* ou atraso de acordo com o MTTR definido na fase 1 do DOTSIM, e depois aquele mesmo recurso é liberado através da classe *release*.

Definir tipos de experimentos

Nesta atividade a tarefa de *estabelecer para quais condições dos valores das variáveis de entrada e parâmetros do sistema os resultados gerados pelo modelo são confiáveis* deve ser executada.

As seguintes condições estão atreladas a confiabilidade dos resultados deste modelo de simulação:

- O modelo de simulação estabelece que as variáveis de entrada de *headaway*, MTTR e MTBF estejam em forma de distribuição de probabilidades (Normal, Triangular, Uniforme, etc).
- A quantidade de vias de tráfego deve ser definido como número inteiro, assim como a quantidade de trens no sistema.

Estabelecer condições de uso

Esta atividade contempla a tarefa de *Definir para quais condições é aplicável o modelo*.

Este modelo de simulação é aplicado apenas a ferrovias que contenham os seguintes elementos de um sistema ferroviário:

- 1 terminal de carga;
- 1 terminal de descarga;
- 1 ferrovia com N pátios ferroviários e $N+1$ vias de tráfego singelas;

Além disso o presente modelo de simulação analisa o sistema com o trem de maior *headway* ou menor velocidade na ferrovia.

Quaisquer condições diferentes destes elementos, deste escopo, ou a adição de mais um elemento ferroviário, será necessário uma nova modelagem, assim como o reinício do ciclo de vida do DOTSIM.

Definição de premissa de Material Rodante

Esta atividade é formada por duas tarefas:

1. Definir tamanho dos veículos e tipos de veículos de cargas (homogêneo ou heterogêneo);
2. Definir quantidade de veículos no sistema na condição de saturação do sistema;

Reiterando o que foi mencionado na atividade de *Estabelecer as condições de uso*, este estudo de caso propõe uma modelagem de um sistema ferroviário com o trem de maior tamanho e menor velocidade (cenário mais restritivo).

A quantidade de veículos no sistema na condição de saturação é avaliado através do incremento de trens no modelo de simulação. O cenário base considera apenas 1 trem, e os demais cenários com o acréscimo de mais 1 trem por vez, até que se chegue em um ponto de saturação. Os resultados a seguir demonstram o alcance desse ponto para as diferentes instâncias avaliadas neste estudo de caso.

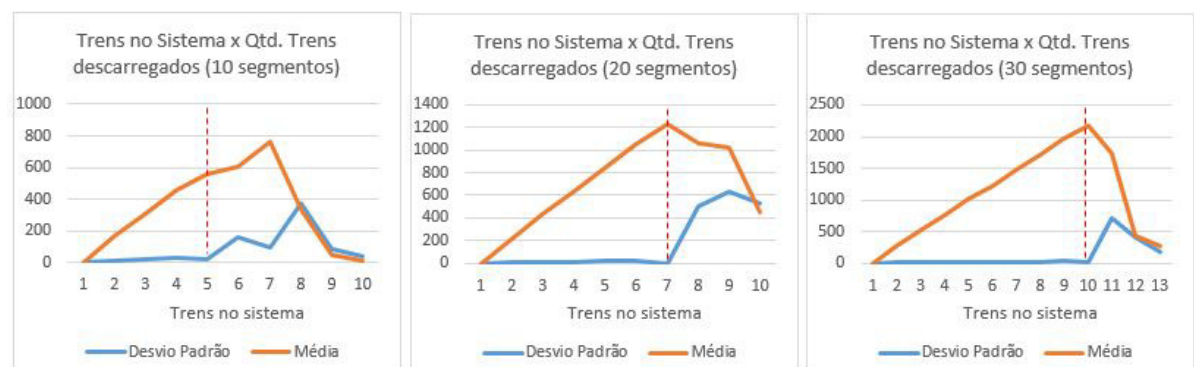


Figura 47 – Quantidade de trens descarregados no porto de acordo com o número de trens no sistema e por instância.

Completada as simulações os resultados são apresentados conforme FIGURA 47. É possível observar um comportamento similar para as curvas de *trens descarregados no porto*, nas 3 instâncias, sempre alcançando um pico e posteriormente uma estabilização ou instabilidade, com uma alta variabilidade (curva azul).

Ainda na FIGURA 47, a linha pontilhada vermelha representa os picos de saturação, os quais são utilizados respectivamente para cada instâncias. A TABELA 6 apresenta de forma consolidada estes resultados.

Tabela 6 – Resultados da análise de saturação dos sistemas ferroviários.

Instâncias	Pico de Saturação (trens)
10 vias	5
20 vias	7
30 vias	10

Seleção de MHS

Conforme citado na SEÇÃO 2.4 o PSD em ferrovias é um clássico problema de permutação o qual é considerado como NP-hard pela literatura no ramo da otimização combinatória. As MHs mais utilizadas para problema similares (permutação) tais como o problema do cacheiro viajante (PCV) são: Busca-Tabu (TS), Greedy Randomized Adaptive Search Procedure (GRASP), Simulated Annealing (SA), genetic Algorithms (GAs), e o algoritmo de otimização da colônia de formigas (ACO, do inglês ant colony optimization algorithm) (KANDA, 2014).

Neste estudo de caso, apenas os métodos de busca local foram priorizados, fazendo uma varredura do espaço através de critérios como vizinhança e busca estocástica. Estas MHs são: *GA*, *SA* e a *BT*.

Codificação das MHs

Esta atividade está inserida no processo de Integração do modelo de simulação com algoritmos de otimização, e possui duas tarefas:

- Caso o ambiente de simulação seja FOSS, codificar os algoritmos de otimização (MHs) na mesma linguagem do ambiente de simulação;
- Caso o ambiente de simulação não seja FOSS, retornar ao processo de modelagem (também na fase 2) e implementar no modelo de simulação a geração de relatórios das medidas de performance e leitura de inputs em arquivos *.txt (de livre edição e consulta).

Dado que o modelo de simulação foi desenvolvido através de uma biblioteca de simulação FOSS (Java Simulation Library - JSL), todas as MHs foram codificadas no mesmo ambiente onde houve a implementação da biblioteca de simulação.

4.2.3 Fase 3

A fase 3 é composta por 5 processos, 5 atividades e 6 tarefas. As próximas subseções descrevem cada atividade com suas respectivas tarefas e resoluções.

Avaliação de performance do MAL

Neste processo a tarefa de *Executar simulação para sequência ótima de duplicação obtida pelo MAL* deve ser concluída.

Esta tarefa consiste basicamente em executar o modelo de simulação em JSL, utilizando como *input* a sequência ótima de duplicação determinada através do MAL (método de Colson), na fase 1. Para cada instância os resultados são os seguintes:



Figura 48 – η para cada instância - Sequência MAL

A FIGURA 49 apresenta a quantidade de trens descarregados no Porto para cada instância do PSD, através da sequência de duplicação determinada através do MAL.

Seleção de candidatos promissores, Avaliação de performance e Condição de parada

As três atividades desta seção contemplam 4 tarefas, são elas:

1. Selecionar candidatos promissores para avaliação de performance pelo modelo de simulação;
2. Executar simulação para candidato selecionado pela MH em N replicações e avaliar o $F(x)$ ou medida de performance;
3. Determinar condição de parada (convergência, número de iterações, ou gatilho);
4. Executar MHs até que condição de parada seja atendida;

Estas 4 tarefas integram um fluxo contínuo de consulta à função objetivo, de candidatos promissores a serem uma solução global. Estes candidatos são selecionados pelas MHs, de acordo com a sua metodologia de busca em uma população inicial. O fluxo desses processos e atividades são apresentados na figura a seguir:

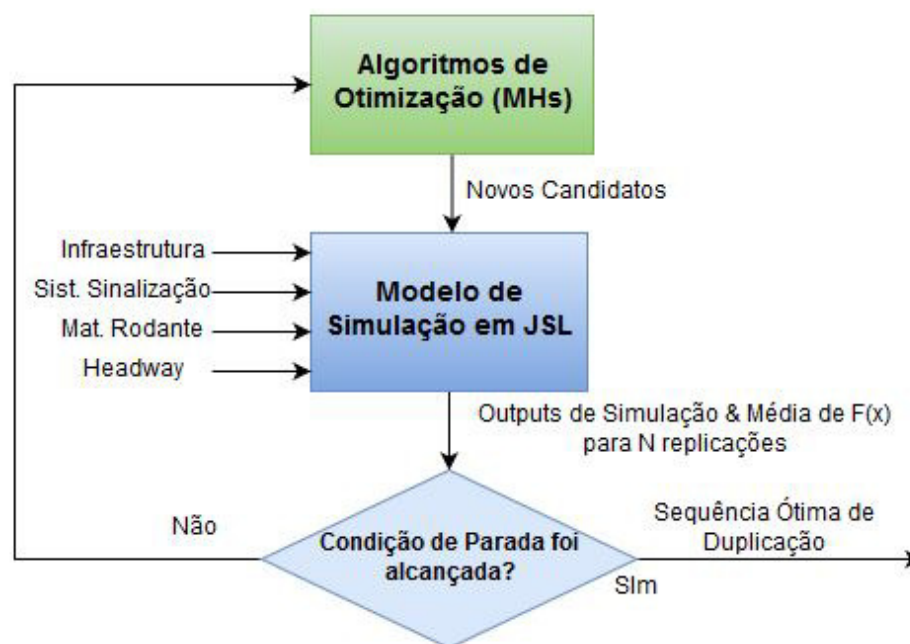


Figura 49 – Interação entre os últimos 4 processos do DOTSIM - Fase 3

Uma condição de parada deve ser definida como mostrado no item 3. Um limite máximo de 1000 iterações foi definido para a execução de todas as MHs e para cada instância.

É importante que após uma execução completa destes 4 processos, os relatórios gerados pelos algoritmos sejam analisados de forma a verificar o comportamento da exploração do espaço de busca até o atingimento da condição de parada. Uma estabilização dos resultados encontrados antes do limite máximo de iterações indicam eficiência deste parâmetro.

4.3 Resultados

Após a implementação da metodologia DOTSIM afim de encontrar uma sequência ótima de duplicação para três instâncias de um PSD em um sistema ferroviário, as medidas de performance para cada MH devem ser comparadas entre si, e com o resultado do MAL. A medida de performance deste estudo de caso é a quantidade de trens descarregados no porto e avaliadas em 3 diferentes ferrovias: com 10, 20 e 30 vias singelas,

Os parâmetros a seguir foram definidos após as MHs encontrarem soluções razoáveis em experimentos preliminares, e sem intenção de promover qualquer MH, mas garantir que soluções sejam obtidas rapidamente. GA: tamanho da população = 100, pais selecionados aleatoriamente, taxa de cruzamento = 10 % operador de cruzamento = um ponto, taxa de mutação = 100 % e seleção da nova população por elitismo; SA: temperatura inicial = 2×10^6 , taxa de aceitação de solução vizinha = aleatoriamente, taxa de resfriamento = 0,99 % e TS: tamanho da lista tabu = sem limitação.

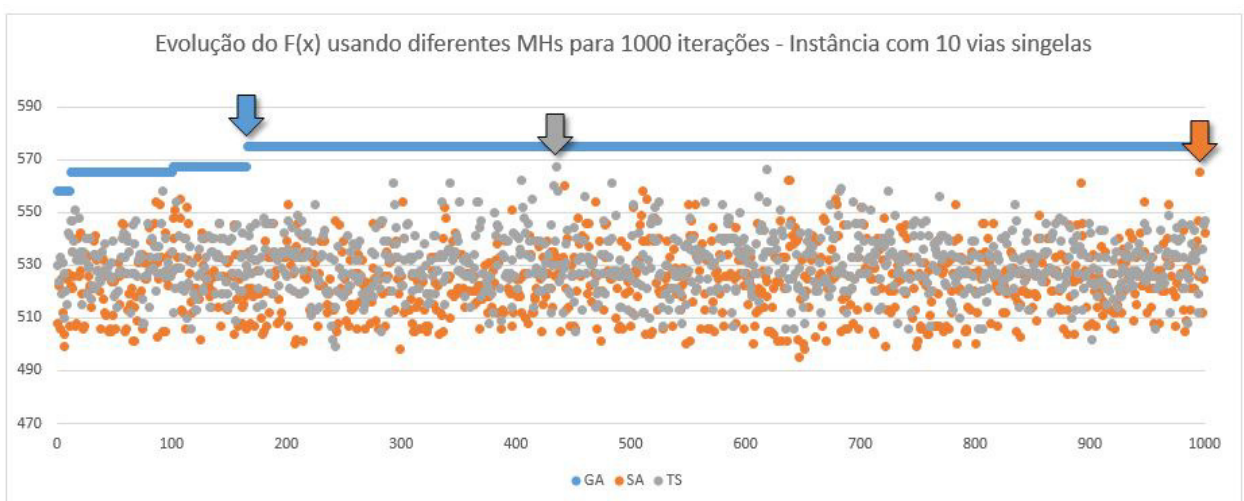


Figura 50 – Evolução do $F(x)$ para 1000 iterações - 10 vias singelas

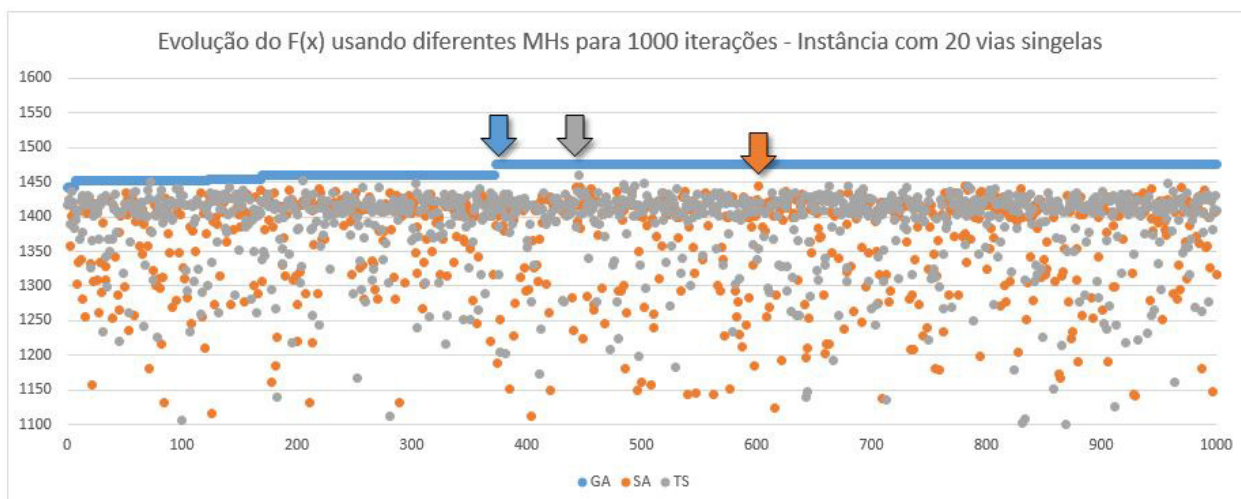


Figura 51 – Evolução do $F(x)$ para 1000 iterações - 20 vias singelas

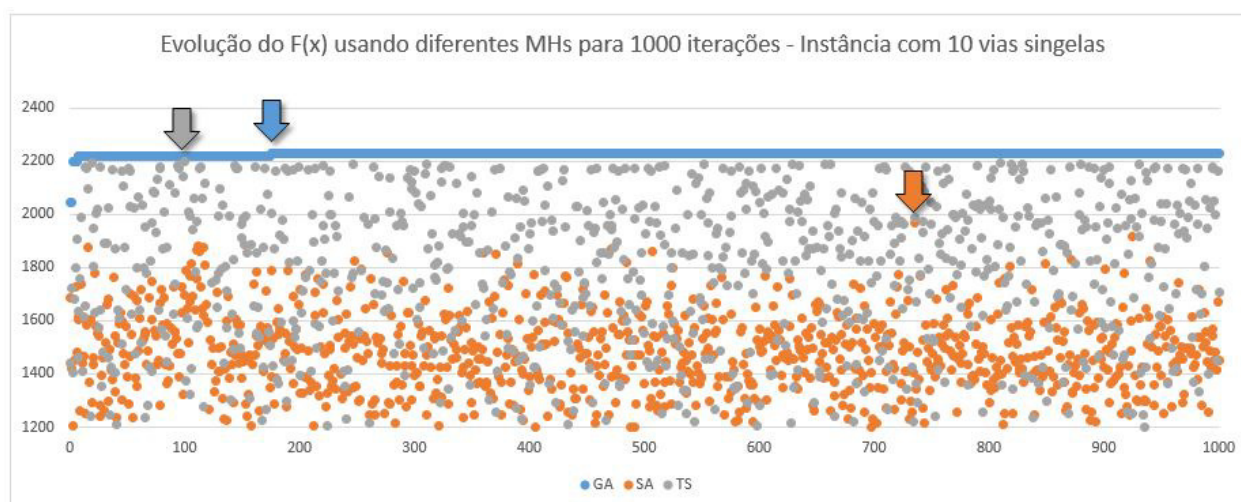


Figura 52 – Evolução do $F(x)$ para 1000 iterações - 30 vias singelas

Observa-se nas FIGURAS 50, 51 e 52 a eficiência das estratégias de exploração do espaço de busca para cada MHs ao longo de todas as 1000 iterações. Além disso, as setas com suas respectivas cores, de acordo com a MH, apontam o momento em que o maior valor da medida de performance é encontrada, uma vez que se trata de um problema de maximização.

Uma característica comum observada nesses resultados, é que o GA consegue encontrar seu pico de $F(x)$ antes de 400 iterações. Outra característica deste método de exploração é a baixa variabilidade da 1ª iteração até a última.

O TS (Tabu-Search) possui uma característica similar ao GA, e consegue levar o $F(x)$ ao seu ápice até 450 iterações. Entretanto, o TS tem ainda como comportamento uma vasta exploração do espaço de busca, uma vez que observa-se uma grande variabilidade dos resultados encontrados para a função objetivo.

O SA (Simulated Annealing) utiliza de aproximadamente todas as 1000 iterações para encontrar seu maior resultado de $F(x)$ como mostrado na FIGURA 50. No entanto, o ganho em relação ao último melhor resultado (aproximadamente 600 iterações), é de menos de 1% . Nas demais instâncias, a solução ótima é encontrada antes de 850 iterações.

Os tempos computacionais necessário para execução de cada MH, são apresentados na TABELA 7.

Tabela 7 – Tempos computacionais para solucionar o problema de otimização (horas).

MHs	10 vias	20 vias	30 vias
G.A.	3.7	16.5	40
S.A.	0.2	0.5	1.5
Tabu Search	10.3	45.4	140

Os resultados apresentados na TABELA 7 mostram que o método TS possui um alto custo computacional para encontrar uma solução para o PSD. Para 30 vias por exemplo, 1000 iterações precisam de aproximadamente 6 dias, em consequência à uma maior quantidade de 'chamadas' à função objetivo.

Por outro lado, o GA apresenta a melhor relação custo-benefício, uma vez que consegue atingir os melhores resultados dentre as MHs, e ainda consegue alcançá-los com menos de 30 % do pior caso em relação a tempo computacional, de 140 horas para o TS com 30 vias singelas.

A TABELA 8 apresenta uma comparação percentual entre a performance das MHs, em relação a melhor solução encontrada. Esta tabela também contribui na avaliação da lacuna entre estes métodos. Observa-se que o TS apesar do alto tempo computacional, consegue alcançar apenas 98,7 % da melhor solução encontrada pelo GA, enquanto o SA alcança apenas 88,5 % para uma instância com 30 vias singelas, com um tempo computacional 40 vezes mais rápido.

Tabela 8 – Comparação percentual de Performance entre MHs.

Instâncias	GA	SA	TS
10 vias	100 %	98,3 %	98,6 %
20 vias	100 %	97,9 %	99,0 %
30 vias	100 %	88,5 %	98,7 %

Os resultados consolidados para cada MH, para cada instância, e uma comparação com os resultados da simulação da sequência determinada através do MAL, são apresentados na FIGURA 53.



Figura 53 – Comparação entre solução ótima global e MAL

A gráfico apresentado na FIGURA 54 mostra uma comparação entre as soluções ótimas globais de cada instância, identificadas através da comparação entre as MHs, e o resultado da simulação da sequência identificada através do MAL.

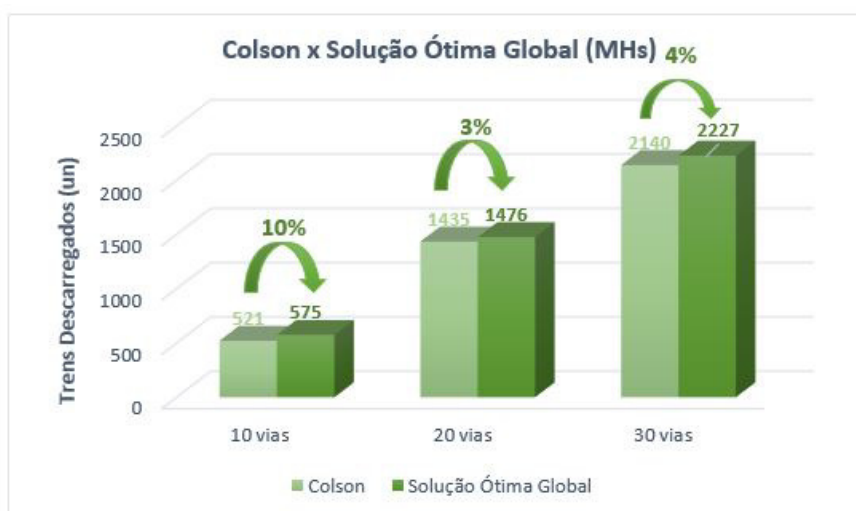


Figura 54 – Comparação entre solução ótima global e MAL

A partir destes resultados é possível observar que em todas as instâncias as MHs conseguem atingir desempenhos melhores que o MAL, a exceção do SA na terceira instância que é 8 % pior. O melhor resultado é alcançado na instância com 10 vias singelas, o qual o GA consegue atingir um resultado 10 % melhor que o MAL. Nas demais instâncias, onde o espaço de busca é ampliado exponencialmente, o GA consegue atingir um ganho de 4 % em relação ao MAL, no entanto, uma vez que sistemas de transporte usualmente trabalham com grandes volumes de cargas transportadas, este patamar de desempenho se torna bastante significativo. Além disso, estes experimentos validam a ineficiência de

métodos analíticos lineares para solução de problemas desta natureza, que apesar de oferecer uma maior agilidade em seu desenvolvimento, não fornece ao cliente final uma solução ótima para a medida de desempenho esperada, e desta forma, podendo provocar impactos na capacidade de tráfego do sistema em estudo.

4.4 Definição de sequência ótima de duplicação por um viés de custo

Além do ponto de vista da maximização do fluxo de cargas em um sistema de transporte, uma visão financeira também é primordial nestes tipos de análises, uma vez que os investimentos em infraestrutura corriqueiramente são bastante representativos, conforme citado na SEÇÃO 2.1. Os investimentos em infraestruturas ferroviárias por km linear podem variar de 1 a 3 milhões de reais, e tem uma relação direta com as obras de artes especiais (OAEs), licenças ambientais e indígenas que estão envolvidas nas respectivas vias. Entre estas OAEs estão: pontes ferroviárias, viadutos, passarelas, vias com rampas acentuadas e com necessidade de operação com locomotivas adicionais nos trens (também conhecido como *helper*), pavimentação de acessos a comunidades, etc.

A fim de avaliar a influência destas variáveis no PSD, esta pesquisa propõe a utilização de uma função objetivo com viés de custo, realizando um balanço financeiro entre um custo total de oportunidade, e as receitas geradas por cada trem descarregado no porto.

$$BalancoFinanceiro = Receita - Custo_{total} \quad (4.6)$$

Este custo total de oportunidade, representado em unidades monetárias (UMs), calcula o somatório dos custos de oportunidade de cada via de tráfego através do produto da taxa de ociosidade (%) e o custo da respectiva via, conforme apresentado na fórmula 4.7:

$$Custo_{total} = \sum_{n=1}^n Ociosidade_n * Custo_n \quad (4.7)$$

Esta equação tem como objetivo penalizar àqueles candidatos que ao longo do período simulado, tiveram uma baixa utilização daqueles recursos duplicados que são mais caros financeiramente, e os investimentos de capital foram bastantes expressivos.

Além do custo de oportunidade, as receitas geradas pelos trens descarregados no porto também são avaliadas na função objetivo, de forma a recompensar aqueles candidatos

que maximizam esta variável.

$$Receita = Total_{descarregado} * Preço \quad (4.8)$$

Para este estudo de caso com viés de custo, cada via recebe um custo de oportunidade, com valores que variam entre 1 e 100 milhões de UMs. Os recursos com um alto valor de investimento ficam no *range* entre 90 e 100 milhões de UMs, com médio valor entre 30 e 60 milhões de UMs e baixo valor entre 0 e 10 milhões de UMs. A instância escolhida para este estudo é o sistema ferroviário com 20 vias de tráfego e com a seguinte distribuição dos investimentos (FIGURA 55) :

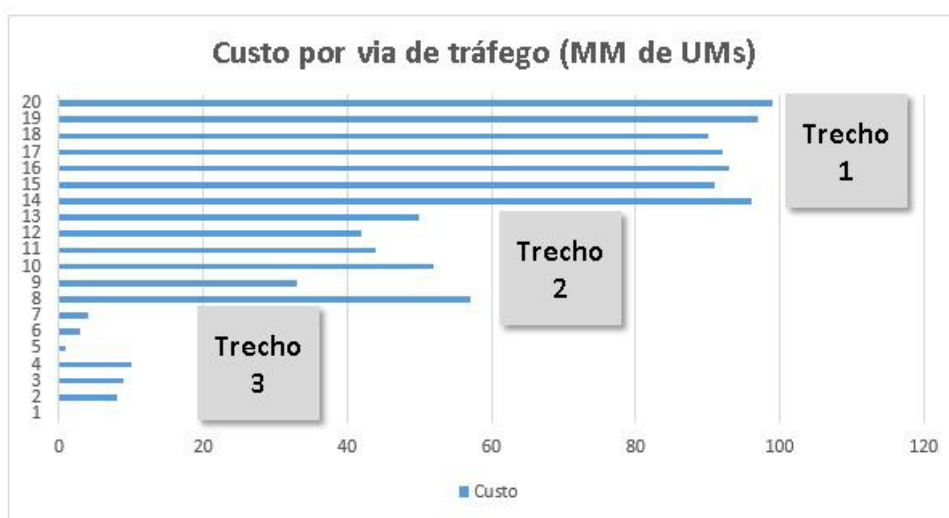


Figura 55 – Custos de oportunidade por vias de tráfego

A ferrovia é dividida em 3 trechos, cada trecho com uma característica de custo, um com nível alto de investimento, 1 com nível médio e 1 com nível baixo. Os valores selecionados, em seus respectivos níveis, são selecionados de forma arbitrária a fim de representar uma estocasticidade no processo.

Para as receitas, considera-se um preço de 2 milhões de UMs para cada trem descarregado no porto.

Diante dos resultados alcançados na seção anterior, o GA é a MH com a maior eficiência no alcance dos resultados finais, e portanto, é utilizada nesta análise. Os parâmetros do GA são mantidos conforme seção anterior, à exceção da quantidade de iterações, que diante dos resultados apresentados, mostrou que o GA consegue atingir resultados satisfatórios antes das 500 iterações.

A FIGURA 56 apresenta uma comparação entre os resultados de balanço financeiro encontrados para a aplicação da sequência ótima obtida pelo GA através de uma função objetivo que busca a maximização da capacidade de tráfego (vide seção 4.3), e o resultado

para a sequência ótima de duplicação com viés de custo na função objetivo, após 500 iterações. Os resultados são satisfatórios, mostrando que é possível obter um ganho de até 72,3 % em relação a sequência de duplicação no estudo de caso sem viés de custo.

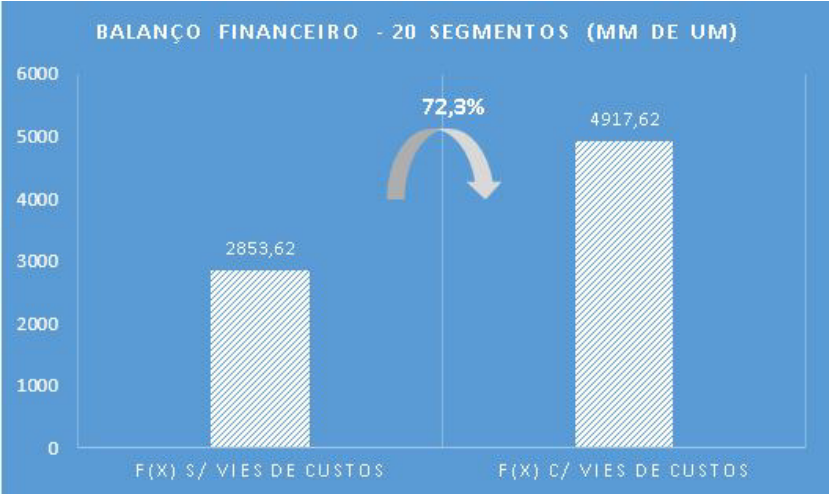


Figura 56 – Comparação entre solução ótima global (GA) e MAL (Colson)

A FIGURA 57 apresenta a sequência ótima de duplicação definida através aplicação do DOTSIM, para o presente estudo de caso.

		Sequência Ótima de Duplicação - GA -																			
Vias de tráfego	Via 1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 2	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 3	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 4	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2	●	2
	Via 5	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 6	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 7	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 8	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 9	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 10	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 11	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 12	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 13	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 14	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 15	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 16	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 17	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 18	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 19	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1
	Via 20	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1	○	1

Figura 57 – Sequência ótima de duplicação para o PSD, obtido através da aplicação do GA

Na FIGURA 58 é apresentado uma comparação entre as diferentes estratégias de duplicação adotadas pelo GA no estudo de caso com e sem viés de custos na função objetivo, no trecho 1, onde os patamares de investimentos de infraestrutura são considerados altos. Observa-se claramente que com a adição da visão de custos na função objetivo, estes trechos são despriorizados entre as semanas 9 e 17 (vide curva laranja) a fim de garantir a máxima utilização desses ativos e reduzir custos.

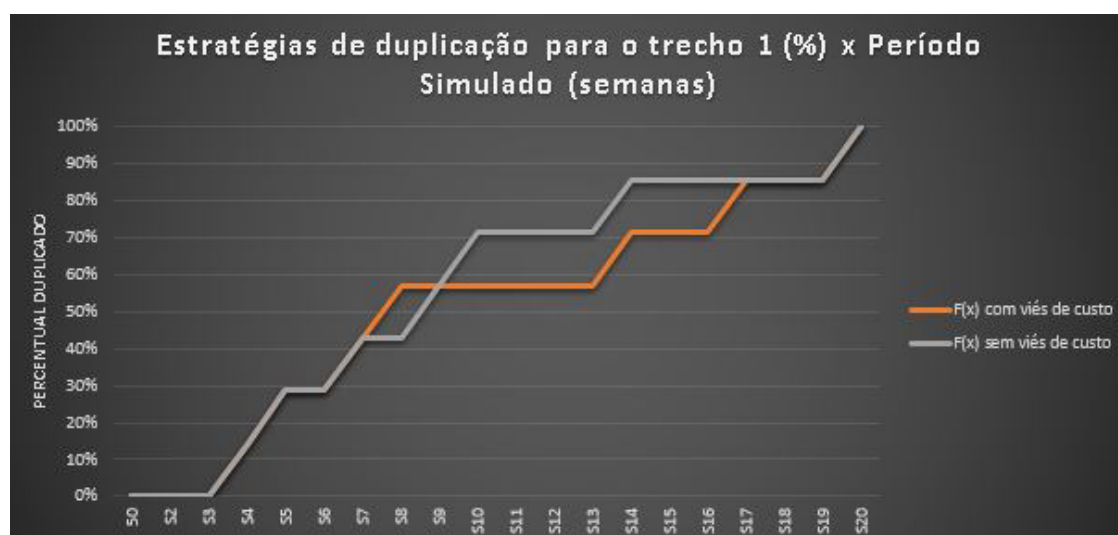


Figura 58 – Estratégia de duplicação do trechos 1 para aplicação do DOTSIM sem e com viés de custo.

5 Conclusões e Trabalhos Futuros

Profissionais e pesquisadores da ciência que estuda os sistemas de transporte de cargas, têm defrontado cenários com forte aumento nos patamares de capacidade afim de atender um crescente aumento de sua demanda, seja para cargas ou passageiros. Para atingir este objetivo, o processo de duplicação de vias de tráfego tem se tornado o 'caminho' mais comum. Tal processo, também conhecido como Problema de Sequenciamento de Duplicação ou PSD, é considerado um problema de otimização intratável, devido sua alta complexidade e variedade quando se trata de quantidade de possíveis soluções. Até agora, tanto em situações reais quanto na literatura, empresas privadas e abordagens existentes tem utilizado de métodos analítico lineares (MALs) para solucionar este tipo de problema, que naturalmente, e como reiterado ao longo desta pesquisa, se trata de um problema extremamente complexo e não linear.

Uma outra lacuna encontrada na literatura é a ausência de uma metodologia de desenvolvimento de processo a qual propõe um conjunto de processos e uma sequência de passos e fases afim de guiar os usuários na solução do PSD. Devido esta ausência, o PSD possui escassa documentação e pesquisa no meio acadêmico.

Quanto a aplicação do SBO (*Simulation based optimization*), observa-se que apesar de ser uma técnica já implementada e testada em problemas complexos na literatura (vide seção 2.6), ela ainda é pouco disseminada e utilizada em problemas complexos de transporte. Ainda é comum observar trabalhos científicos com aplicação destas técnicas mas de forma isolada, ou aplicando simulação, ou otimização, sem uma integração, aproveitando o potencial de cada técnica.

Para solucionar este problema uma metodologa baseada em otimização e simulação de processos estocásticos é apresentada nesta pesquisa a qual visa realizar uma busca por uma solução ou sequência ótima global de duplicação de vias singelas de tráfego em sistemas de transportes de cargas.

Um estudo de caso com diferentes instâncias de ferrovias é apresentado com a aplicação de 3 (três) Meta Heurísticas (MHs). O objetivo deste estudo é encontrar soluções satisfatórias quando comparadas com os resultados do respectivo MAL, e desta forma contribuir para que empresas que utilizam destes modais de transporte, possam minimizar custos e alavancar sua produtividade (e.g. aumentar sua capacidade produtiva no menor tempo possível). Além disso, a estratégia de duplicação em uma curta ou extensa ferrovia, pode ser definida considerando somente um singelo problema de otimização.

Resultados do modelo de simulação para diferentes instâncias, com N vias de tráfego singelas, variáveis estocásticas robustas as quais representam distúrbios de operação

e manutenção, afim de maximizar o número de trens descarregados no porto durante o período de construção, mostra a habilidade do DOTSIM de buscar soluções ótimas globais para suportar estratégias de duplicação em ferrovias.

Uma comparação com o MAL para ferrovias, também conhecido com método de Colson, destacou que o DOTSIM aumenta fortemente a capacidade de tráfego e consequentemente reduz investimentos de capital, uma vez que sinaliza quais vias devem de fato ser priorizadas e quais patamares de capacidade de tráfego elas conseguem atingir.

Destaca-se também que os MALs não possuem a capacidade de avaliar estratégias de duplicação através de formação de corredores uma vez que o MAL considera um fator de eficiência operacional (K) fixo e linear para todas as vias da ferrovia em estudo. Entretanto, na prática, neste tipo de estratégia, o fator de eficiência operacional para cada via singela pode ser afetado por suas vizinhança, e somente *softwares* de simulação possuem tal habilidade, o que fortalece o benefício da utilização de metodologias como o DOTSIM.

Referências

- ABRIL, M.; BARBER, F.; INGOLOTTI, L.; SALIDO, M.; TORMOS, P.; LOVA, A. An assessment of railway capacity. *Transportation Research Part E: Logistics and Transportation Review*, Elsevier, v. 44, n. 5, p. 774–806, 2008. Citado 4 vezes nas páginas 54, 55, 56 e 57.
- ARAUJO, H.; VALE, S. Dotsim - a simulation-based optimization methodology for the optimal duplication sequence on freight transportation systems. In: INSTICC. *Proceedings of the 19th International Conference on Enterprise Information Systems - Volume 2: ICEIS*, [S.l.]: ScitePress, 2017. p. 459–466. ISBN 978-989-758-248-6. Citado na página 59.
- ARAÚJO, H. F. H. Utilização de ferramentas de simulação para análise de capacidade e dimensionamento material rodante da efc. 2013. Citado na página 88.
- AVISON, D.; FITZGERALD, G. *Information systems development: methodologies, techniques and tools*. [S.l.]: McGraw Hill, 2003. Citado na página 63.
- AVISON, D. E.; FITZGERALD, G. Where now for development methodologies? *Communications of the ACM*, ACM, v. 46, n. 1, p. 78–82, 2003. Citado 2 vezes nas páginas 62 e 64.
- BANKS, J.; MALAVÉ, C. O. The simulation of inventory systems: An overview. *Simulation*, SAGE Publications, v. 42, n. 6, p. 283–290, 1984. Citado 2 vezes nas páginas 22 e 29.
- BANKS, J.; NORMAN, V. B. Justifying simulation in today's manufacturing environment. *IIE Solutions*, Institute of Industrial Engineers, Inc.(IIE), v. 27, n. 11, p. 16–19, 1995. Citado 2 vezes nas páginas 23 e 68.
- BARBOSA, V. C. Redes neuronais e simulated annealing como ferramentas para otimização combinatória. *Investigación Operativa*, v. 1, n. 2, p. 125–142, 1989. Citado na página 41.
- BARROS, J. *Avaliação dos principais métodos analíticos de cálculo de capacidade de tráfego utilizados em ferrovia nacional e internacional*. Tese (Doutorado) — Dissertação de Mestrado—UFMG, Minas Gerais, 2013. Citado 4 vezes nas páginas 50, 57, 67 e 82.
- BEŠINOVIĆ, N.; QUAGLIETTA, E.; GOVERDE, R. M. A simulation-based optimization approach for the calibration of dynamic train speed profiles. *Journal of Rail Transport Planning & Management*, Elsevier, v. 3, n. 4, p. 126–136, 2013. Citado na página 58.
- BOARD, S. T.; WARCHOT, L. P. Association of american railroads. 2011. Citado 2 vezes nas páginas 10 e 52.
- BRINA, H. L. Estradas de ferro, volume i e ii. *Rio de Janeiro, LTC: Livros Técnicos e Científicos Editora SA*, 1982. Citado 3 vezes nas páginas 53, 57 e 67.
- BRINA, H. L. Estradas de ferro volume i, capítulo 2, pag. 7. *Editora UFMG. Belo Horizonte—MG*, 1988. Citado 3 vezes nas páginas 49, 50 e 83.

- BURDETT, R. L.; KOZAN, E. Techniques for absolute capacity determination in railways. *Transportation Research Part B: Methodological*, Elsevier, v. 40, n. 8, p. 616–632, 2006. Citado na página 54.
- CASTRO, C. N. d. O agronegócio e os desafios do financiamento da infraestrutura de transportes no brasil. Instituto de Pesquisa Econômica Aplicada (Ipea), 2015. Citado 2 vezes nas páginas 19 e 20.
- CODE, U. 406 ?capacity ? *UIC, Editions Techniques Ferroviaires, Paris*, 2004. Citado na página 57.
- DREYFUS, S. E.; LAW, A. M. *Art and Theory of Dynamic Programming*. [S.l.]: Academic Press, Inc., 1977. Citado na página 49.
- EGLESE, R. Simulated annealing: a tool for operational research. *European journal of operational research*, Elsevier, v. 46, n. 3, p. 271–281, 1990. Citado na página 41.
- FILHO, P. F. Introdução a modelagem e simulação de sistemas. 2ª edição. *Editores Visual*, 2008. Citado 2 vezes nas páginas 10 e 34.
- FILHO, P. J. de F. *Introdução à modelagem e simulação de sistemas: com aplicações em Arena*. [S.l.]: Visual Books, 2001. Citado 5 vezes nas páginas 10, 20, 21, 26 e 34.
- FRANK, O. Two-way traffic on a single line of railway. *Operations Research*, INFORMS, v. 14, n. 5, p. 801–811, 1966. Citado na página 48.
- FRISCHTAK, C. R. O investimento em infra-estrutura no brasil: histórico recente e perspectivas. Instituto de Pesquisa Econômica Aplicada (Ipea), 2008. Citado na página 19.
- FU, M. C. Optimization for simulation: Theory vs. practice. *INFORMS Journal on Computing*, INFORMS, v. 14, n. 3, p. 192–215, 2002. Citado 2 vezes nas páginas 10 e 35.
- GLOVER, F.; LAGUNA, M. Tabu search, 1997. *Cited on*, p. 83, 1997. Citado na página 46.
- GOLDBERG, D. E. Genetic algorithms in search, optimization and machine learning ?addison-wesley, 1989. *Reading, MA*, 1989. Citado na página 36.
- GOMEZ, C.; SAA, R.; GARCIA, A.; GARCIA-CARBALLEIRA, F.; CARRETERO, J. A model to obtain optimal designs of railway overhead knuckle junctions using simulation. *Simulation Modelling Practice and Theory*, Elsevier, v. 26, p. 16–31, 2012. Citado na página 58.
- ISLER, C. A. *Proposta de um modelo de capacidade de processamento de trens cargueiros em redes ferroviárias de linha singela*. Tese (Doutorado) — Universidade de São Paulo, 2010. Citado 2 vezes nas páginas 51 e 54.
- JACOBS, P. H.; LANG, N. A.; VERBRAECK, A. Web-based simulation 1: D-sol; a distributed java based discrete event simulation architecture. In: WINTER SIMULATION CONFERENCE. *Proceedings of the 34th conference on Winter simulation: exploring new frontiers*. [S.l.], 2002. p. 793–800. Citado 2 vezes nas páginas 30 e 31.

JERRY, B. *Discrete-event system simulation*. [S.l.]: Pearson Education India, 1984. Citado 2 vezes nas páginas 10 e 25.

JOINES, J. A.; ROBERTS, S. D. Design of object-oriented simulations in c++. In: IEEE COMPUTER SOCIETY. *Proceedings of the 28th conference on Winter simulation*. [S.l.], 1996. p. 65–72. Citado na página 31.

KANDA, J. Y. Sistema de meta-aprendizado para a seleção de meta-heurística para o problema do caixeiro viajante. *Anais do X Simpósio Brasileiro de Sistemas de Informação (SBSI), Londrina*, p. 651–662, 2014. Citado 3 vezes nas páginas 35, 71 e 97.

KONTAXI, E.; RICCI, S. Railway capacity analysis: Methodological framework and harmonization perspectives. In: *Proceedings of the 12th World Conference on Transportation Research, Lisboa*. [S.l.: s.n.], 2010. Citado na página 55.

KRAFT, E. R. Jam capacity of single track rail lines. In: *Proceedings of the Transportation Research Forum*. [S.l.: s.n.], 1982. v. 23, n. 1. Citado na página 57.

KROON, L.; MARÓTI, G.; HELMRICH, M. R.; VROMANS, M.; DEKKER, R. Stochastic improvement of cyclic railway timetables. *Transportation Research Part B: Methodological*, Elsevier, v. 42, n. 6, p. 553–570, 2008. Citado na página 58.

KRUEGER, H. Parametric modeling in rail capacity planning. In: IEEE. *Simulation Conference Proceedings, 1999 Winter*. [S.l.], 1999. v. 2, p. 1194–1200. Citado na página 50.

KRUEGER, H.; VAILLANCOURT, E.; DRUMMIE, A. M.; VUCKO, S. J.; BEKAVAC, J. Simulation within the railroad environment. In: SOCIETY FOR COMPUTER SIMULATION INTERNATIONAL. *Proceedings of the 32nd conference on Winter simulation*. [S.l.], 2000. p. 1191–1200. Citado 4 vezes nas páginas 55, 56, 57 e 67.

LABORATÓRIO DE SISTEMA DE TRANSPORTES DA ENGENHARIA DE PRODUÇÃO E TRANSPORTES -? LASTRAN. 2012. (Acesso em outubro de 2012). Disponível em: <<http://www.producao.ufrgs.br>>. Citado na página 53.

LACERDA, E. G. de; CARVALHO, A. de. Introdução aos algoritmos genéticos. *Sistemas inteligentes: aplicações a recursos hídricos e ciências ambientais*, v. 1, p. 99–148, 1999. Citado 2 vezes nas páginas 37 e 39.

MEDINA, A. C.; CHWIF, L. Modelagem e simulação de eventos discretos. *São Paulo: Bravarte*, 2006. Citado na página 34.

MEIRELES, R. Modelagem e simulação da malha ferroviária em circuito fechado da estrada de ferro vitória a minas. 2010, 218 p. *Disseração de Mestrado, UFES–Universidade Federal do ES*, 2010. Citado 4 vezes nas páginas 10, 11, 48 e 87.

NARA K.. SIMULATED ANNEALING APPLICATIONS. IN:SONG, Y.-H. *Modern Optimisation Techniques in Power Systems*. [S.l.]: Springer Science & Business Media, 1999. v. 20. Citado 3 vezes nas páginas 10, 41 e 42.

NETO, X. R. *Padrões para manutenção da via permanente*. 1985. 177 p. Tese (Doutorado) — Dissertação de Mestrado em Transporte–Instituto Militar de Engenharia, Rio de Janeiro, 1985. Citado na página 53.

- OM, G. Unified modeling language specification. 2003. Citado 2 vezes nas páginas 63 e 64.
- PACHECO, M. Notas de aula em computação evolucionária. *Disponível em:* <<http://www.ica.ele.puc-rio.br>>. Acesso em, v. 1, 2004. Citado na página 37.
- PACHECO, M. A. C. et al. Algoritmos genéticos: princípios e aplicações. *ICA: Laboratório de Inteligência Computacional Aplicada. Departamento de Engenharia Elétrica. Pontifícia Universidade Católica do Rio de Janeiro. Fonte desconhecida*, 1999. Citado 2 vezes nas páginas 12 e 36.
- PEGDEN, C. D.; SADOWSKI, R. P.; SHANNON, R. E. *Introduction to simulation using SIMAN*. [S.l.]: McGraw-Hill, Inc., 1995. Citado 2 vezes nas páginas 21 e 22.
- PEIXOTO, T. A. *URURAU - Um ambiente de Simulação a Eventos Discretos*. Tese (Doutorado) — Tese de Mestrado. Universidade Cândido Mendes - Campos / RJ, 2002. Citado 2 vezes nas páginas 10 e 31.
- PIGOSKI, T. M. *Practical software maintenance: best practices for managing your software investment*. [S.l.]: Wiley Publishing, 1996. Citado na página 63.
- PINTO, M. A. d. M. Modeling and simulation to discover railroad trains capacity. *WSC*, 2012. Citado na página 70.
- PROTIL, R. Otimização do processo decisório utilizando simulação computacional. *XXXIII Simpósio Brasileiro de Pesquisa Operacional, Campos do Jordão, SP*, p. 1535–1546, 2001. Citado 3 vezes nas páginas 10, 34 e 35.
- QUAGLIETTA, E. A simulation-based approach for the optimal design of signalling block layout in railway networks. *Simulation Modelling Practice and Theory*, Elsevier, v. 46, p. 4–24, 2014. Citado na página 58.
- RIVES, F. O.; PITA, A. L.; PUENTE, J. M. Tratado de ferrocarriles. *Editorial Rueda*, v. 1, 1977. Citado na página 51.
- ROSSETI, M. Simfonia: An object-oriented simulation framework proceedings of the 2000 winter simulation conference ja joiner, rr barton, k. Kang, and PA Fishwick, eds, 2000. Citado na página 31.
- ROSSETTI, M. D. Java simulation library (jssl): an open-source object-oriented library for discrete-event simulation in java. *International Journal of Simulation and Process Modelling*, Inderscience Publishers, v. 4, n. 1, p. 69–87, 2008. Citado na página 30.
- ROUNDTABLE, B. *The Case for Investing in America's Transportation Infrastructure*. 2015 (accessed September 7, 2016). <<http://businessroundtable.org>>. Citado na página 20.
- SCHMIDT, J. W.; TAYLOR, R. E. *Simulation and analysis of industrial systems*. [S.l.]: Richard D. Irwin, 1970. Citado na página 26.
- SINGH, R. International standard iso/iec 12207 software life cycle processes. *Software Process Improvement and Practice*, v. 2, n. 1, p. 35–50, 1996. Citado na página 63.

SWAIN, J. Informs simulation software survey. *OR/MS Today*, 2007. Citado na página 21.

Vale. *Silo de Carregamento*. 2013. [Online; acessado em 17 de abril de 2017]. Disponível em: <<http://www.vale.com/brasil/EN/aboutvale/news/Pages/navegue-pela-galeria-de-fotos-e-saiba-mais-sobre-o-projeto-adicional-40.aspx>>. Citado 2 vezes nas páginas 10 e 78.

VERBRAECK, A. Real-time visualization and modeling of supply chains. *real-time: managing the new supply chain*, 2004. Citado na página 30.

WEATHERLY, R. M.; PAGE, E. H. Efficient process interaction simulation in java: Implementing co-routines within a single java thread. In: WINTER SIMULATION CONFERENCE. *Proceedings of the 36th conference on Winter simulation*. [S.l.], 2004. p. 1437–1443. Citado na página 31.

WEISBROD, G.; TREYZ, F. Productivity and accessibility: bridging project-specific and macroeconomic analyses of transportation investments. *Journal of Transportation and Statistics*, Bureau of Transportation Statistics, US Department of Transportation, v. 1, n. 3, p. 65–79, 1998. Citado na página 16.

ANEXO I

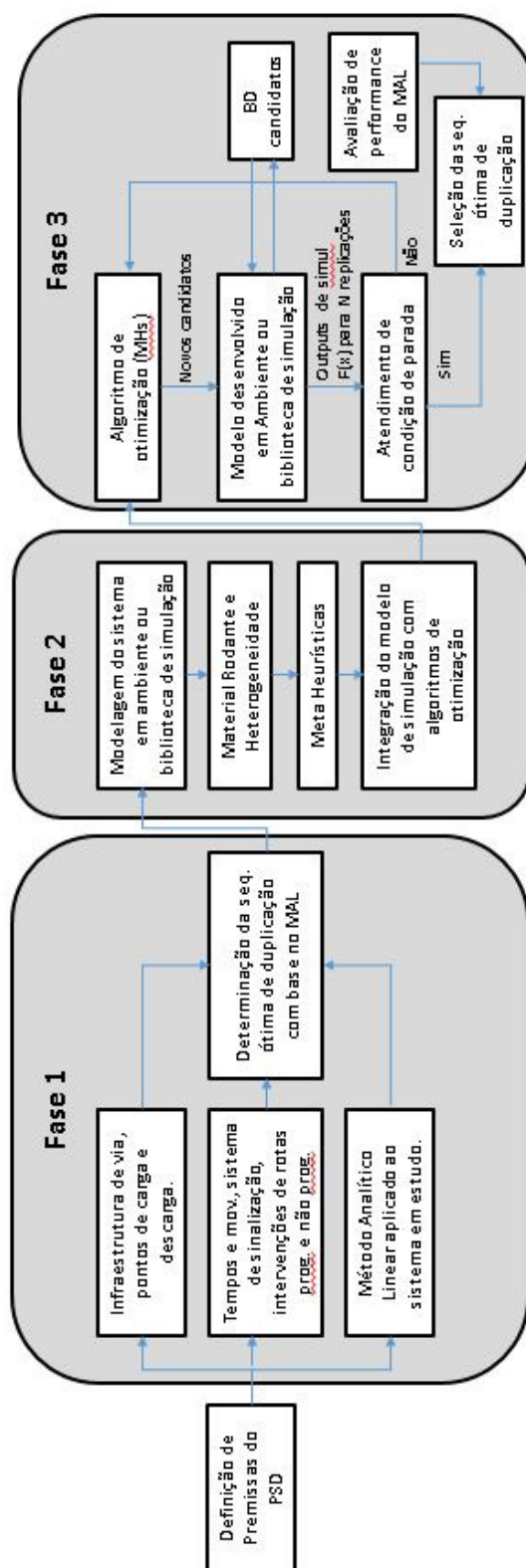


Figura 59 – As fases e processos do DOTSIM