

**MARCOS RAFAEL DA SILVA**

**UM AGENTE DELIBERATIVO APLICADO AO  
APOIO À CONDUÇÃO DE TRENS**

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática Aplicada.

**CURITIBA**

**2011**

**MARCOS RAFAEL DA SILVA**

**UM AGENTE DELIBERATIVO APLICADO AO  
APOIO À CONDUÇÃO DE TRENS**

Dissertação apresentada ao Programa de Pós-Graduação em Informática Aplicada da Pontifícia Universidade Católica do Paraná como requisito parcial para obtenção do título de Mestre em Informática Aplicada.

Área de Concentração: *Agentes de Software*

Orientador: Prof. Dr. Edson Emílio Scalabrin

Co-orientador: Prof. Dr. Fabrício Enembreck

**CURITIBA**

**2011**



ATA DE DEFESA DE DISSERTAÇÃO DE MESTRADO  
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

DEFESA DE DISSERTAÇÃO Nº 16/2011

Aos 31 dias do mês de Agosto de 2011 realizou-se a sessão pública de Defesa da Dissertação “Um Agente Deliberativo Aplicado ao Apoio à Condução de Trens” apresentada pelo aluno Marcos Rafael da Silva, como requisito parcial para a obtenção do título de Mestre em Informática, perante uma Banca Examinadora composta pelos seguintes membros:

Prof. Dr. Edson Emílio Scalabrin  
PUCPR (Orientador)

(assinatura)

Aprov.  
(aprov/reprov.)

Prof. Dr. Fabrício Enembreck  
PUCPR (co-orientador)

APROVADO

Prof. Dr. Bráulio Coelho Avila  
PUCPR

APROVADO

Prof. Dr. Gerson Linck Bichinho  
PUCPR/PPGTS

Aprovado

Conforme as normas regimentais do PPGIa e da PUCPR, o trabalho apresentado foi considerado Aprovado (aprovado/reprovado), segundo avaliação da maioria dos membros desta Banca Examinadora. Este resultado está condicionado ao cumprimento integral das solicitações da Banca Examinadora registradas no Livro de Defesas do programa.

  
Prof. Dr. Mauro Sérgio Pereira Fonseca  
Diretor do Programa de Pós-Graduação em Informática



*Dedico este trabalho à minha família, aos amigos e a todos que me apoiaram nesta importante conquista.*

## Agradecimentos

Ao nosso Deus criador, por ter me dado saúde, força e perseverança para concluir este trabalho.

À minha esposa Fernanda e ao meu filho Cauã, pela paciência e amor nesta etapa de nossas vidas.

Aos meus pais, Haroldo e Jacira pela educação e pelas oportunidades que me deram.

Aos meus tios, Airton e Zilda pelo apoio durante o desenvolvimento deste trabalho.

Especialmente ao Professor Dr. Edson Emílio Scalabrin pela amizade e auxílio incondicional para a conclusão deste trabalho.

Aos Professores, Dr. Fabrício Enembreck e Dr. Bráulio Coelho Ávila por todas as contribuições.

Aos amigos do projeto PAI-L, Allan, Richardson, Bruno, Osmar e especialmente ao André Pinz por todo o auxílio nos momentos necessários.

# Sumário

<b>Agradecimentos.....</b>	<b>ii</b>
<b>Sumário.....</b>	<b>iii</b>
<b>Lista de Figuras.....</b>	<b>iv</b>
<b>Lista de Tabelas.....</b>	<b>v</b>
<b>Lista de Quadros.....</b>	<b>vi</b>
<b>Lista de Abreviaturas.....</b>	<b>vii</b>
<b>Resumo.....</b>	<b>viii</b>
<b>Abstract.....</b>	<b>ix</b>
<b>1 INTRODUÇÃO.....</b>	<b>1</b>
1.1 Desafio.....	3
1.2 Motivação .....	3
1.3 Proposta .....	5
1.4 Contribuições.....	5
1.5 Estrutura do Trabalho .....	5
<b>2 CONDUÇÃO DE TRENS .....</b>	<b>7</b>
2.1 Grandezas, Perfil da Via, Esforços de Tração e Aderência .....	7
2.2 Cálculo de Resistências .....	10
2.3 Força Aceleradora.....	13
2.4 Tempo e Deslocamento .....	13
2.5 Exemplo de Aplicação de Resistências, Força Aceleradora, Deslocamento e Tempo.....	14
2.6 Pontos de Aceleração, Consumo e LTKB .....	21
2.7 Freios de um Trem.....	23
2.8 Velocidade de Regime e Velocidade Crítica .....	23
2.9 Considerações Finais .....	24
<b>3 AGENTES .....</b>	<b>25</b>
3.1 Introdução.....	25
3.2 Agentes de Software .....	25
3.3 Por que Agentes de Software?.....	27
3.4 Estados Mentais .....	28
3.5 Raciocínio Prático.....	28
3.5.1 IRMA.....	31
3.5.2 PRS .....	33

3.5.3 <i>AgentSpeak(L)</i> .....	34
3.5.4 <i>Jason</i> .....	40
3.6 Planejamento .....	42
3.7 Arquitetura BDI.....	44
3.8 Considerações Finais .....	45
<b>4 UM AGENTE DELIBERATIVO APLICADO AO APOIO À CONDUÇÃO DE TRENS.....</b>	<b>46</b>
4.1 Introdução.....	46
4.2 Visão Geral .....	47
4.3 Arquitetura Proposta.....	49
4.4 Raciocínio Prático - Deliberação e Planejamento.....	51
4.4.1 <i>Deliberação - Gerar Opções</i> .....	54
4.4.2 <i>Utilidade</i> .....	56
4.4.3 <i>Filtro e Intenção</i> .....	57
4.5 Execução do Plano .....	60
4.6 Replanejamento .....	61
4.7 Considerações Finais .....	62
<b>5 SIMULADOR .....</b>	<b>63</b>
5.1 Introdução.....	63
5.2 Descrição Geral .....	63
5.3 Via Férrea .....	67
5.4 Calculador.....	68
5.5 Exemplo da Simulação .....	69
5.6 Considerações Finais .....	79
<b>6 RESULTADOS OBTIDOS.....</b>	<b>80</b>
6.1 Introdução.....	80
6.2 Metodologia de Avaliação .....	80
6.2.1 <i>Experimento: Cenário 5</i> .....	82
6.2.2 <i>Experimento: Cenário 7 – Replanejamento Automático</i> .....	85
6.2.3 <i>Experimento: Cenário 8</i> .....	88
6.2.4 <i>Experimento: Cenário 8 – Replanejamento Simulado</i> .....	90
6.3 Análise e Discussão .....	94
6.4 Análise da Implementação.....	94
6.5 Considerações Finais .....	97
<b>7 CONCLUSÃO.....</b>	<b>98</b>
<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>100</b>

## Lista de Figuras

Figura 1: Um agente interagindo em seu ambiente. Fonte: (WOOLDRIDGE, 2002, p. 16)...	26
Figura 2: IRMA. Fonte: (BRATMAN; POLLACK; ISRAEL, 1988).....	32
Figura 3: Arquitetura do sistema PRS. Fonte: (GEORGEFF; INGRAND, 1989, p. 02).....	34
Figura 4: Planos em AgentSpeak(L). Fonte: (BORDINI; VIEIRA, 2003). ....	36
Figura 5: Sintaxe de AgentSpeak(L). Fonte: (HÜBNER; BORDINI; VIEIRA, 2004) .....	37
Figura 6: Ciclo da interpretação de um programa AgentSpeak(L). Fonte: (MACHADO; BORDINI, 2002) .....	39
Figura 7: Arquitetura do agente proposto.....	50
Figura 8: Estrutura de um plano .....	54
Figura 9: Deliberação .....	58
Figura 10: Execução e análise de um plano de viagem.....	61
Figura 11: Modelo de domínio .....	65
Figura 12: Representação conceitual de uma via férrea .....	68
Figura 13: Comunicação com o Calculador .....	69
Figura 14: Pontos de aceleração planejados e aplicados .....	83
Figura 15: Velocidade praticada pelo condutor humano (maquinista) e pelo agente .....	83
Figura 16: (a) Quantidade de ações planejadas, (b) Quantidade de pontos de aceleração utilizados, (c) Esforço Trator Efetivo (kgf) e Resistências (Kgf), (d) Esforços Tratores (Kgf) - (Cenário 5).....	84
Figura 17: Violação de esforços .....	86
Figura 18: Pontos de aceleração planejados/replanejados e aplicados.....	86
Figura 19: Esforços tratores após replanejamento.....	87
Figura 20: (a) Quantidade de ações planejadas e replanejadas, (b) Quantidade de ações realizadas, (c) Quantidade de pontos de aceleração utilizados, (d) Esforço Trator Efetivo (Kgf) e Resistências (Kgf), (e) Velocidades (Km/h) - (Cenário 7 - Replanejamento Automático).....	88

Figura 21: Pontos de aceleração planejados e aplicados .....	89
Figura 22: (a) Quantidade de ações planejadas, (b) Quantidade de pontos de aceleração utilizados, (c) Esforço Trator Efetivo (Kgf) e Resistências (Kgf), (d) Esforços Tratores (Kgf), (e) Velocidades (Km/h) - (Cenário 8).....	90
Figura 23: Violação de velocidade máxima permitida.....	91
Figura 24: Pontos de aceleração planejados/replanejados e aplicados.....	92
Figura 25: Velocidades (km/h).....	92
Figura 26: (a) Quantidade de ações planejadas e replanejadas, (b) Quantidade de ações realizadas, (c) Quantidade de pontos de aceleração aplicados com replanejamento, (d) Esforços Tratores (Kgf), (e) Esforço Trator Efetivo (Kgf) e Resistências (Kgf) - (Cenário 8 - Replanejamento Simulado). .....	93
Figura 27: Quantidade de ações realizadas pelo agente .....	96

## Lista de Tabelas

Tabela 1: Consumos e Potências da Locomotiva C30 (LEITE, 2009, p. 14).....	22
Tabela 2: Definição dos valores para o atributo <i>consumo</i> .....	56
Tabela 3: Definição dos valores para o atributo <i>velocidade</i> .....	56
Tabela 4: Definição dos valores para o atributo <i>força aceleradora</i> .....	57
Tabela 5: Definição dos valores para o atributo <i>esforço trator efetivo</i> .....	57
Tabela 6: Definição dos valores para o atributo <i>deslocamento</i> .....	57
Tabela 7: Estados das variáveis referentes à primeira ação planejada.....	74
Tabela 8: Estados das variáveis referentes à segunda ação planejada.....	75
Tabela 9: Estados das variáveis referentes à terceira ação planejada.....	76
Tabela 10: Estados das variáveis referentes à última ação planejada.....	78
Tabela 11: Descrição dos cenários dos experimentos.....	81
Tabela 12: Resultados.....	82
Tabela 13: Definição dos valores para o atributo <i>deslocamento e velocidade</i> .....	85
Tabela 14: Tempo de Processamento.....	96

## Lista de Quadros

Quadro 1: Função de seleção para gerar opções.....	55
Quadro 2: Função de seleção para gerar intenção .....	59

## Lista de Abreviaturas

BDI	<i>Belief, Desire, Intention</i>
IA	<i>Inteligência Artificial</i>
IAD	<i>Inteligência Artificial Distribuída</i>
IRMA	<i>Intelligent Resource-Bounded Machine Architecture</i>
Kgf	<i>Quilograma por tonelada</i>
LTKB	<i>Litro de Combustível por Tonelada Bruta</i>
PRS	<i>Procedure Reasoning System</i>
PSI	<i>Pound Force per Square Inch</i>
RCS	<i>Reaction Control System</i>
XML	<i>Extensible Markup Language</i>

## Resumo

Este trabalho apresenta um sistema computacional inteligente para a condução de trens interurbanos de cargas. Tal sistema é dotado de um módulo de raciocínio prático que o permite gerar um plano de condução que satisfaça uma meta complexa. A consecução desta meta requer duas grandes etapas: (i) planejar/replanejar as ações que permitem mover um trem de carga de um ponto P a um ponto Q, conjugando restrições naturalmente conflitantes de tempo de viagem, segurança e economia; e (ii) executar tal plano, modificando o ambiente. Para resolver ambos os problemas ligados a estas duas etapas, diferentes abordagens já foram amplamente estudadas na literatura, em particular, na especialidade de Inteligência Artificial. Dada a necessidade de dotar cada trem com um sistema inteligente de condução, a escolha da arquitetura de agente independente mostrou-se apropriada vis-à-vis a complexidade do problema. O modelo mental baseou-se na tripla: crença, desejo e intenção. Os testes foram realizados sobre nove cenários de viagens com configurações específicas em termos do peso de cada trem, número de locomotivas e trechos de uma via férrea. O resultado de cada viagem, realizada de forma simulada pelo sistema, é comparado em termos de tempo e de economia de combustível, com o resultado da mesma viagem realizada pelo maquinista em um trem real. Nos experimentos, observou-se uma redução média de 28% no consumo de combustível e 30% no tempo de viagem.

**Palavras-chave:** Inteligência Artificial; Agentes BDI; Raciocínio Prático; Planejamento.

## Abstract

This paper presents an intelligent computer system for conducting long distance freight trains. This system is equipped with a module of practical reasoning that enables you to generate a driving schedule that satisfies a target complex. This target consists of two major steps: (i) planning/replan the actions that allow you to move a freight train of a point P to point Q, combining naturally conflicting constraints of travel time, safety and economy, and (ii) to execute this plan, modifying the environment. To solve both problems related to these two steps, different approaches have been widely studied in literature, in particular, specializing in Artificial Intelligence. Given the need to provide each train with a smart driving, the choice of architecture independent agent proved to be appropriate the complexity of the problem. The mental model was based on the triple: belief, desire and intention. The tests were performed on nine trips scenarios with specific configurations in terms of weight of each train, the number of locomotives and a stretch of railway. The result of each trip, in a simulated system, is compared in terms of time and fuel economy, the driver traveling man with the same configuration in the real world. In the experiments, observed an average reduction of 28% in fuel consumption and 30% in travel time.

Keywords: Artificial Intelligence; BDI agents; Practical Reasoning; Planning.

# Capítulo I

## 1 Introdução

Os agentes de software são fundamentais para a Inteligência Artificial (IA) e Inteligência Artificial Distribuída (IAD), visto que um dos objetos de estudo é a resolução coletiva de problemas complexos por agentes. O modelo de um agente se caracteriza de acordo com a forma como ele se comporta no ambiente, sendo de forma geral, dividido em reativo ou cognitivo. Um agente reativo apresenta um comportamento mais simples, à medida que está vinculado ao estímulo/resposta provocados por mudanças em seu ambiente de atuação. Já um agente cognitivo, realiza um comportamento mais inteligente e elaborado, faz uso de conhecimentos prévios e apresentam habilidades peculiares, como aprendizagem, racionalidade e autonomia. Neste sentido, os agentes cognitivos podem auxiliar na resolução de problemas considerados complexos.

Dentre as arquiteturas existentes para a concepção de agentes, destaca-se o modelo de estados mentais BDI (*Belief, Desire, Intention*), como uma das mais promissoras, à medida que possibilita a concepção de agentes que se comportam e raciocinam de maneira semelhante ao raciocínio humano. Esta arquitetura reúne contribuições filosóficas de M. Bratman (BRATMAN, 1987) e D. Dennett (DENNETT, 1987). Rao e Georgeff (1995) transformaram estes conceitos em um modelo formal para execução de agentes que utilizam crença, desejo e intenção. A arquitetura BDI contempla a elaboração de planos, que é fundamental para determinar o comportamento proativo de um agente. O planejamento integra o raciocínio de um agente, de forma que ele delibera, estrategicamente, um conjunto de ações para atingir uma meta. Um exemplo de aplicação BDI de destaque é o PRS (*Procedure Reasoning System*) (GEORGEFF; INGRAND, 1989). Este sistema foi amplamente utilizado no auxílio

ao controlador RCS – (*Reaction Control System*) da NASA. O PRS elaborava planos de ação e corrigia falhas do RCS em uma viagem espacial. É possível afirmar que sistemas que possuem características semelhantes ao PRS, podem ser utilizados em diversos tipos de controles e auxílio à resolução de problemas complexos.

Não obstante, o uso de sistemas inteligentes para resolução de problemas tem sido amplamente investigado, como em Wolfe e Sierhuis (2008) que desenvolveram um framework para simulação do gerenciamento do fluxo de tráfego aéreo utilizando agentes BDI, onde observaram uma redução significativa no tempo de execução da simulação com o uso destes conceitos. Segundo os autores, esta arquitetura é indicada para a modelagem de sistemas que necessitam de tomadas de decisão complexas.

Karim e Heinze (2005) propuseram e avaliaram um agente de controle de veículo aéreo não tripulado. Os autores apresentam duas diferentes abordagens para o problema. A primeira consistia em adicionar uma camada de agentes sobre o controle do voo. A segunda levava a metáfora do organismo humano e programava um controlador autônomo, baseado no modelo de decisão humana e de agentes BDI. A comparação entre estas abordagens é apresentada pelos autores, ao qual demonstraram que há viabilidade na utilização de agentes BDI.

Borges (2009) propôs um agente inteligente usado no auxílio à condução de trens. O agente proposto possuía um conjunto de recursos para gerar sua política de ação, como perfil da via, características dos veículos e empregava algoritmos de aprendizagem de mineração de dados na geração de regras de condução seguras e eficientes. O principal comportamento do agente era manter, reduzir ou aumentar a velocidade do veículo, onde os esforços principais estavam concentrados em empregar regras de condução baseadas em viagens passadas na formação de planos de condução. Resultados experimentais demonstraram que algoritmos de aprendizagem de máquina podem ser utilizados com sucesso, visto que em 85% dos casos/decisões as regras descobertas eram aplicáveis.

Leite (2009) utilizou-se da técnica de programação distribuída de restrições para otimizar o consumo de combustível em viagens de trens. Para isto, foram atribuídos pesos diferentes, configurados empiricamente para os seguintes critérios: velocidade, deslocamento e consumo, possibilitando uma configuração específica para cada cenário de viagem utilizado, atingindo assim, resultados satisfatórios quanto à economia de combustível.

Neste trabalho será abordado o problema da condução de trens, que é influenciado por diferentes grandezas, como resistências, forças de tração e forças de aderência. Desta forma, o movimento do veículo está condicionado a um conjunto de restrições que influencia na tomada de decisão do condutor. Sendo assim, considera-se o sistema de condução de trens de alta complexidade. Em geral, busca-se a segurança, a rapidez e a economia na viagem.

## **1.1 Desafio**

O movimento de um trem é diretamente afetado pelas resistências que incidem sobre ele, como curvas, rampas, peso e velocidade. Além disto, as variações climáticas influenciam na aderência dos trilhos, e conseqüentemente nas tomadas de decisão do condutor.

O condutor do veículo deve aplicar um conjunto de ações que proporcione uma viagem segura, rápida e econômica. Porém, devido às restrições de viagem, estes requisitos se tornam naturalmente conflitantes. Uma tomada de decisão equivocada, a infração de uma regra de segurança, a ultrapassagem do limite de velocidade máxima permitida, ou o excesso de força de aceleração, podem oferecer danos ao meio ambiente, ao material rodante, a via férrea e causar acidentes fatais.

A maneira como um condutor realiza suas ações influencia diretamente no consumo de combustível produzido pelo veículo. Todavia, o planejamento das ações de uma viagem de trem se torna uma das tarefas mais desafiadoras para um condutor, uma vez que, em nível de complexidade, é difícil prever as modificações ocorridas no ambiente durante a viagem. Muitas vezes os planos precisam ser refeitos devido às estas mudanças ou quando os mesmos são falhos.

## **1.2 Motivação**

Os agentes são indicados para atuarem em ambientes dinâmicos e incertos, quando há necessidade de resolução de problemas complexos e elevado grau de autonomia. Eles são componentes fundamentais para enfrentar os desafios de interagir com o mundo real.

Segundo Georgeff et al. (1998), há necessidade de agentes para “informáticas populares” e demonstrar que eles podem resolver problemas não tratáveis por métodos convencionais. Atualmente muitas aplicações computacionais são algorítmicas e trabalham

com informação pré-definida. Entretanto, frente às novas exigências, esses sistemas têm necessidades de constante atualização, requerendo, portanto, arquiteturas e linguagens que possam reduzir a complexidade, o tempo para especificação e a modificação. Os agentes de software têm condições de atender essas necessidades e resolver problemas mais complexos, quando dotados de racionalidade e autonomia (GEORGEFF et al., 1998).

Uma das formas de observar a racionalidade de um agente é verificar se ele realiza planejamento. Um agente racional situado em um ambiente dinâmico, após deliberar, deverá planejar suas ações de forma a atingir seus objetivos (desejos). Os planos não devem ser aplicados por longos períodos de tempo devido às mudanças que ocorrem nestes ambientes. Devem ser utilizados para prever situações futuras na execução do sistema, e gerar possíveis ações para tratar estas mudanças. Os objetivos representam estados motivadores e são componentes essenciais para o sistema. No entanto, softwares convencionais não são orientados a objetivos. Isto implica na incapacidade de recuperação automática de falhas e no uso de oportunidades não especificadas inicialmente (Ibidem).

As vantagens de uso de sistemas BDI quanto à robustez, flexibilidade e eficiência são discutidas em (SILVA; SARDINA; PADGHAM, 2009). Os autores propõem a incorporação de planos híbridos a estes sistemas, onde o agente utiliza seus conhecimentos prévios para a elaboração de novos planos. Estes deverão estabelecer uma relação com os objetivos específicos, a fim de evitar a uma ação inconsistente. Esta forma de planejamento proporciona a realização de planos mais elaborados e não codificados inicialmente pelo programador, proporcionando maior grau de autonomia para um agente. A resolução de um problema de planejamento híbrido envolve a construção de um plano abstrato que pode ser decomposto utilizando um domínio de conhecimento em um plano primitivo. Este plano abstrato traz sobre si a relação entre o estado inicial de crença e o seu objetivo. Por fim, estes autores demonstram a importância e a capacidade dos sistemas BDI no auxílio à resolução de problemas complexos, raciocínio em tempo real e controle de requisitos.

Enfim, sistemas inteligentes aplicados ao controle autônomo necessitam de respostas imediatas, elaboração de novas estratégias e planos de ação em tempo real. O modo de condução de um trem se configura de forma que as mudanças no perfil da via (rampas), diferentes velocidades máximas permitidas, condições meteorológicas e o peso do trem, influenciam diretamente a escolha das melhores ações pelo condutor. De maneira geral, busca-se garantir uma viagem segura, rápida e econômica.

### **1.3 Proposta**

O objetivo deste trabalho é conceber um agente BDI capaz de planejar/replanejar ações capazes de conduzir um trem utilizando um conjunto de recursos, tais como: perfil da via, composição, percepção de crenças, raciocínio sobre as ações. O foco do agente condutor está voltado à escolha dos pontos de aceleração e elaboração de planos de viagem nos diferentes trechos da via férrea guiados por meio de seus objetivos e metas. A eficiência do agente se manifesta na forma de uma boa política de ação que deve ser atingida quando uma viagem é segura, rápida e econômica.

Os objetivos específicos deste trabalho são:

- Estudar as grandezas físicas da condução de trens interurbanos de carga;
- Definir uma arquitetura de agente de software que leve em conta a complexidade do problema de condução de trens interurbanos de carga norteadas pelo modelo mental: crença, desejo e intenção; e
- Avaliar o desempenho dos planos de ação do agente condutor vis-à-vis a realização de viagens com configurações específicas em termos do peso de cada trem, número de locomotivas e trechos de uma via férrea.

### **1.4 Contribuições**

As contribuições científicas deste trabalho são: (i) modelagem de um agente de software para resolução de um problema do mundo real; (ii) concepção de um sistema dotado de um módulo de raciocínio prático que consiste em planejar/replanejar ações de uma viagem de trem interurbano de cargas; e um módulo executor, que consiste em analisar e executar o plano de viagem gerado. A estratégia adotada foi suficiente para proporcionar segurança e, reduzir consideravelmente, o consumo de combustível e o tempo de viagem, em relação às viagens concretizadas por condutores humanos, após experimentos realizados.

### **1.5 Estrutura do Trabalho**

As seções subsequentes deste trabalho estão assim organizadas: O capítulo 2 apresenta a dinâmica da condução de trens, como forças de tração, forças de aderência e forças de resistência que incidem sobre este meio de transporte. O capítulo 3 apresenta os principais

conceitos relacionados à teoria de agentes BDI. O capítulo 4 apresenta a metodologia utilizada para o desenvolvimento do modelo proposto. O capítulo 5 apresenta o ambiente de simulação de viagens desenvolvido cujo objetivo é a implementação da metodologia proposta através de um software de simulação. O capítulo 6 apresenta os resultados obtidos por meio da simulação de nove cenários de viagens comparados com os resultados obtidos em viagens reais conduzidas por especialistas humanos. O capítulo 7 apresenta a conclusão final deste trabalho vis-à-vis aos objetivos apresentados.

## Capítulo II

### 2 Condução de Trens

Para conduzir um trem, o maquinista (condutor) aplica um conjunto de ações que resulta em uma força de aceleração suficiente para realizar o movimento do veículo na ferrovia. Suas ações estão condicionadas a uma série de restrições, como peso do veículo, perfil da via férrea, forças tratoras, forças aderentes, forças de resistência, variações do clima e velocidade do veículo, ao qual implicam diretamente na forma ideal de condução.

#### 2.1 Grandezas, Perfil da Via, Esforços de Tração e Aderência

Sob o ponto de vista mecânico, segundo Brina (1982, p. 170), o transporte nas ferrovias é influenciado por três grandezas: (i) o peso da carga a ser transportada juntamente com o peso do vagão; (ii) a via férrea, que representa o caminho a ser percorrido; e (iii) a força a ser aplicada para movimentar o veículo. Observa-se que neste meio de transporte é possível acoplar vagões e locomotivas de acordo com a necessidade dos volumes a serem transportados e ao material rodante.

Comparado a outros meios de transporte, o transporte ferroviário se apresenta como o mais econômico do ponto de vista energético, pois necessita de um menor esforço trator para locomover o veículo quando este se encontra em um trecho em nível. Isto se deve ao fato da natureza do material das rodas do veículo e dos trilhos, que geralmente são feitos de aço, proporcionando um menor atrito entre estes elementos. Entretanto, para se tirar maior vantagem da economia do esforço trator nas ferrovias, e devido à sensibilidade do perfil da ferrovia, deve-se adotar um sistema de rampas suaves, pois o esforço trator cresce de forma

acentuada quando o veículo se encontra em uma rampa, comparando-se ao esforço trator apresentado nas rodovias para a mesma situação (BRINA, 1982).

A *aderência* ( $F$ ) é a força resistente que se opõe ao deslizamento do veículo, entre a roda e o trilho, enquanto este se encontra parado. Esta força é limitadora e deve ser menor que o produto entre o *peso aderente* ( $p_a$ ) e o *coeficiente de atrito ou coeficiente de aderência* ( $f$ ). Caso a força de aderência ( $F$ ) desenvolva uma força de resistência maior que o produto do *peso aderente* ( $p_a$ ) pelo *coeficiente de atrito* ( $f$ ) haverá a patinagem da roda. A equação básica da tração por aderência é denominada equação de *Coulomb*, mostrada pela Equação 1 (Ibidem).

$$F \leq p_a * f \quad (1)$$

Onde:  $F$  = força de aderência em kgf

$p_a$  = peso aderente em kg

$f$  = coeficiente de aderência (ou coeficiente de atrito)

De acordo com Brina (1982, p. 6), o coeficiente de aderência ( $f$ ) varia de 0,10 a 0,33 dependendo das condições da superfície dos trilhos (trilho com óleo, úmido e sujo, úmido de orvalho, molhado pela chuva, seco, seco e limpo, completamente seco). Após a realização de alguns experimentos, ele observou que o coeficiente de aderência diminui à medida que a velocidade do veículo aumenta. Esta variação pode ser dada por meio da Equação 2.

$$f = \frac{f_0}{1 + 0,01 * V} \quad (2)$$

Onde:  $f$  = coeficiente de aderência

$f_0$  = coeficiente de aderência em repouso

$V$  = velocidade em Km/h

O peso aderente ( $p_a$ ) é o peso aportado pela roda de cada locomotiva e pode ser encontrado na mesma proporção dada pelo quociente do número de rodas com tração (eixos tratores) pelo número de rodas sem tração (eixos sem tração) em proporção ao peso total da locomotiva.

A força de tração de uma locomotiva depende da potência e do seu peso. O cálculo de lotação de um veículo férreo é feito mediante as condições do pior trecho da via, onde o veículo encontra as maiores resistências e, conseqüentemente, irá desenvolver menor velocidade possível, denominada *velocidade crítica*. A expressão geral da potência de uma locomotiva pode ser definida por meio da Equação 3 (BRINA, 1982).

$$W_{hp} = \frac{F * V}{273,24} \quad (3)$$

Onde:  $W_{hp}$  = potência geral da locomotiva em HP

$F$  = força dada em kgf

$V$  = velocidade em km/h

A potência efetiva nas rodas da locomotiva ( $W_{Hpe}$ ) é definida pela potência no motor primário da locomotiva ( $W_{hpi}$ ) multiplicada pelo coeficiente de rendimento da transferência de potência até as rodas.

A força de tração ou *esforço trator* corresponde à potência produzida necessária a realizar o movimento do veículo. O esforço trator é dado em função da velocidade, a partir da potência efetiva produzida, como mostra a Equação 4 (Ibidem).

$$F = \frac{273,24 * W_{HP}}{V} \quad (4)$$

Onde:  $F$  = esforço trator efetivo em kgf

$V$  = velocidade em km/h

$W_{HP}$  = potência da locomotiva em HP

Existem dois tipos de esforço trator: (i) *esforço trator limitado pela aderência*; e (ii) *esforço trator efetivo*. Esforço trator limitado pela aderência é aquele que permite que uma locomotiva consiga vencer a resistência total da composição sem que aconteça a patinagem. Este esforço é dado pela Equação 5, onde  $E_a$  é o esforço trator aderente. O esforço trator efetivo é a potência responsável realmente pelo movimento do trem. Este esforço é calculado pela Equação 6, onde  $E_e$  é o esforço trator efetivo. A constante 273,24 é um valor fixo e a constante 0,82 indica o rendimento do motor diesel elétrico (BRINA, 1982).

$$E_a = \frac{P * C}{1 + 0,01 * V} \quad (5)$$

$$E_e = \frac{273,24 * 0,82 * W}{V} \quad (6)$$

Onde:  $E_a$  = esforço trator aderente em kgf

$E_e$  = esforço trator efetivo em kgf

$V$  = velocidade desejada em km/h

$P$  = peso da locomotiva em kg

$C$  = coeficiente de atrito da via férrea

$W$  = potência da locomotiva em HP

O esforço trator efetivo nunca poderá exceder o esforço trator aderente, pois isto ocasiona a patinagem da locomotiva, o que pode provocar sérios danos nas rodas, trilhos e motor. O esforço trator efetivo deve ser capaz de superar a resistência dos vagões e das locomotivas. Portanto, a dinâmica do movimento leva em consideração a força de tração (esforços tratores), força de aderência e resistências.

## 2.2 Cálculo de Resistências

As resistências de um veículo ferroviário são denominadas por um conjunto de forças que se opõem ao seu movimento. Segundo Brina (1982), uma resistência é definida no ponto de contato entre as rodas e os trilhos e se opõem ao esforço trator nas rodas da locomotiva. As equações de Davis, publicadas em 1926, expressam os cálculos de resistências. Utiliza-se como unidade o Kgf/ton ou Kg/ton, denominada quilograma-força, definida por cada tonelada do veículo. As resistências de um veículo férreo são compostas por dois tipos: (i) *normal* e (ii) *acidental*.

A resistência normal é aquela que atua em todo o trecho e varia de acordo com a velocidade do veículo. Ela é dada pelos atritos das rodas nos *trilhos*, nos *eixos* e com o *ar* e é dada pela Equação 7 (para locomotiva com peso por eixo superior a 5 toneladas) e Equação 7a (para locomotiva com peso por eixo inferior a 5 toneladas); e Equação 8 (para vagão com peso por eixo superior a 5 toneladas) e Equação 8a (para vagão com peso por eixo inferior a 5 toneladas), para cada locomotiva e vagão que compõem o veículo (BRINA, 1982).

$$R'_{nl} = 1,3 + \frac{29}{w} + 0,03 * V + \frac{0,0024 * A * V^2}{n * w} \quad (7)$$

$$R'_{nl} = \frac{9,4}{\sqrt{w}} + \frac{12,5}{w} + 0,03 * V + \frac{0,0024 * A * V^2}{n * w} \quad (7a)$$

$$R'_{nv} = 1,3 + \frac{29}{w} + 0,045 * V + \frac{0,0005 * A * V^2}{n * w} \quad (8)$$

$$R'_{nv} = \frac{9,4}{\sqrt{w}} + \frac{12,5}{w} + 0,045 * V + \frac{0,0005 * A * V^2}{n * w} \quad (8a)$$

Onde:  $w$  = peso por eixo por short-ton (1 ton = 1,1 short-ton)

$V$  = velocidade em km/h

$A$  = área frontal em (sq/ft)

$n$  = número de eixos

$R'_{nl}$  = resistências em libras/ton (1tlb/short-ton = 0,5 Kgf/ft ou Kg/ton) para cada locomotiva

$R'_{nv}$  = resistências em libras/ton (1tlb/short-ton = 0,5 Kgf/ft ou Kg/ton) para cada vagão

A resistência acidental compreende as variações de *curva* e *rampa* ao longo do trecho percorrido, e pelo estado de *inércia*, quando no início do movimento para adquirir velocidade, ou na aceleração para aumento da mesma. Esta resistência é dada pela Equação 9, Equação 10, e Equação 11a (para locomotivas) e Equação 11b (para vagões), expressando respectivamente, as resistências de inércia ( $R'_i$ ), rampa ( $R'_r$ ), e curva ( $R'_c$ ) para cada locomotiva e vagão que compõe veículo (BRINA, 1982).

$$R'_i = 4 * \frac{V_2^2 - V_1^2}{l} \quad (9)$$

Onde:  $R'_i$  = resistência de inércia em kgf

$V_1$  = velocidade inicial em km/h

$V_2$  = velocidade final em km/h

$l$  = deslocamento percorrido em metros

$$R'_r = 10 * i \quad (10)$$

Onde:  $R'_r =$  resistência de rampa em kgf  
 $i\%$  = percentual de rampa

$$R'_c = 0,2 + \frac{100}{R} * (p + b + 3,8) \quad (11a)$$

$$R'_v = \frac{500 * b}{R} \quad (11b)$$

Onde:  $R'_c =$  resistência de curva para a locomotiva em kgf  
 $R'_v =$  resistência de curva para o vagão em kgf  
 $p =$  base rígida em metros  
 $b =$  bitola da linha em metros  
 $R =$  raio da curva em metros

A resistência total da composição é dada pela Equação 12 sendo expressa pelo somatório das resistências anteriores para todas as locomotivas e vagões do veículo (BRINA, 1982).

$$R_{total} = N_{locomotiva} * P_{locomotiva} * (R'_{nl} + R'_{cl} + R'_r + R'_i) + N_{vagao} * P_{vagao} * (R'_{nv} + R'_{cv} + R'_r + R'_i) \quad (12)$$

Onde:  $P_{locomotiva} =$  peso da locomotiva em toneladas  
 $N_{locomotiva} =$  número de locomotivas  
 $P_{vagao} =$  peso do vagão em toneladas  
 $N_{vagao} =$  número de vagões  
 $R'_{nl} =$  resistência normal da locomotiva em kgf  
 $R'_{cl} =$  resistência de curva da locomotiva em kgf  
 $R'_i =$  resistência de inércia em kgf  
 $R'_r =$  resistência de rampa em kgf  
 $R'_{nv} =$  resistência normal do vagão em kgf  
 $R'_{cv} =$  resistência de curva do vagão em kgf  
 $R_{total} =$  resistência total em kgf

## 2.3 Força Aceleradora

A força aceleradora do trem é dada pela diferença entre o esforço trator efetivo e a soma de todas as resistências. Esta força é a responsável pelo ganho da velocidade do veículo, e é dada pela Equação 13 (BRINA, 1982):

$$F_{ac} = E_t - R \quad (13)$$

Onde:  $F_{ac}$  = força aceleradora em kgf

$E_t$  = esforço trator em kgf

$R$  = resistência total em kgf

## 2.4 Tempo e Deslocamento

A partir dos cálculos obtidos anteriormente é possível calcular o deslocamento e o tempo gasto pelo trem. O cálculo do deslocamento de um trem é dado pela Equação 14 em metros. A partir do valor do deslocamento  $S$  é possível obter o valor do tempo gasto  $T$  em segundos para este deslocamento expresso pela Equação 15 (BRINA, 1982).

$$S = \frac{4 * PT * (V_{Final}^2 - V_{Inicial}^2)}{FA} \quad (14)$$

$$T = \frac{7,2 * S}{V_{Final} + V_{Inicial}} \quad (15)$$

Onde:  $S$  = deslocamento em metros

$V_{Inicial}$  = velocidade inicial da locomotiva em km/h

$V_{Final}$  = velocidade final da locomotiva em km/h

$PT$  = peso total dos veículos em toneladas

$FA$  = força aceleradora em kgf

$T$  = tempo gasto do deslocamento em segundos

## 2.5 Exemplo de Aplicação de Resistências, Força Aceleradora, Deslocamento e Tempo

Para demonstrar os cálculos de resistência, força aceleradora, deslocamento e tempo, será apresentado um exemplo da partida de um trem em nível e reta, até atingir a velocidade de 4 km/h . O veículo será rebocado por uma locomotiva de 80 toneladas, com a lotação de 10 vagões de 40 toneladas, totalizando 480 toneladas (BRINA, 1982).

Onde:  $P_{locomotiva} = 80 t$  (peso da locomotiva)  
 $P_{vagao} = 40 t$  (peso do vagão)  
 $n = 4$  (número de eixos)  
 $V_m =$  velocidade média  
 $F_t =$  força tratora ou esforço trator  
 $Coeficiente_{atrito} = 0,20$  (coeficiente de atrito)  
 $N_{locomotiva} = 1$  (número de locomotivas)  
 $N_{vagao} = 10$  (número de vagões)

**Cálculos para um intervalo de velocidade  $\Delta v = 2 km / h$ , ou seja, velocidade entre 0 e 2 km/h.**

$$V_m = 1 Km / h$$

**Cálculo da força tratora/esforço trator efetivo para 1 km/h:**

$$F_t = \frac{P_{locomotiva} * Coeficiente_{atrito}}{1 + 0,01 * V_m}$$

$$\frac{F_t = 80000 * 0,20}{1 + 0,01 * 1} = 15841,58 kgf$$

**Conversão de milhas para quilômetros:**

$$V_{mi/h} = 1 * 0,622 = 0,622 \text{ mi/h}$$

mi/h = milhas por hora

**Conversão toneladas para *short-ton* da locomotiva:**

1,1 = constante

$$\frac{P_{locomotiva}}{n} = \frac{80}{4} * 1,1 = 22 \text{ toneladas / eixo}$$

**Cálculo de resistência normal para a locomotiva a 1 km/h:**

$$R'_{nl} = 1,3 + \frac{29}{w} + 0,03 * V + \frac{0,0024 * A * V^2}{n * w}$$

$$R'_{nl} = 1,3 + \frac{29}{w} + 0,03 * 0,622 + \frac{0,0024 * 110 * 0,622^2}{4 * 22}$$

$$R'_{nl} = 1,3 + 1,3181 + 0,0186 + \frac{0,0024 * 110 * 0,622^2}{4 * 22}$$

$$R'_{nl} = 1,3 + 1,3181 + 0,0186 + \frac{0,0024 * 110 * 0,3868}{88}$$

$$R'_{nl} = 1,3 + 1,3181 + 0,0186 + \frac{0,1021}{88}$$

$$R'_{nl} = 1,3 + 1,3181 + 0,0186 + 0,0011$$

$$R'_{nl} = 2,6378 \text{ libras / toneladas}$$

$$R'_{nl} = 2,64 \text{ lb / t}$$

$$R'_{nl} = 2,64 * 0,5 = 1,32 \text{ kgf}$$

Onde:  $R'_{nl}$  = resistência normal da locomotiva em kgf

$w$  = peso médio por eixo em *short-ton*

$n$  = número de eixos

$A$  = área frontal em (sq-ft)

$V$  = velocidade da locomotiva em mi/h

**Conversão toneladas para *short-ton* do vagão:**

1,1 = constante

$$\frac{P_{vagao}}{n} = \frac{40}{4} * 1,1 = 10 * 1,1 = 11 \text{ toneladas / eixo}$$

### Cálculo de resistência normal para o vagão a 1 km/h:

$$R'_{nv} = 1,3 + \frac{29}{w} + 0,045 * V + \frac{0,0005 * A * V^2}{n * w}$$

$$R'_{nv} = 1,3 + \frac{29}{11} + 0,045 * 0,622 + \frac{0,0005 * 85 * 0,622^2}{4 * 11}$$

$$R'_{nv} = 1,3 + 2,6363 + 0,0279 + \frac{0,0005 * 85 * 0,3868}{44}$$

$$R'_{nv} = 1,3 + 2,6363 + 0,0279 + 0,00037$$

$$R'_{nv} = 3,9647 = 3,96 lb / t$$

$$R'_{nv} = 3,96 * 0,5$$

$$R'_{nv} = 1,98 kgf$$

Onde:  $R'_{nv}$  = resistência normal do vagão em kgf

$w$  = peso médio por eixo em short-ton

$n$  = número de eixos

$A$  = área frontal em (sq/ft)

$V$  = velocidade em mi/h

### Cálculo de resistência total: vagões e locomotiva a 1 km/h:

$$R_{total} = N_{locomotiva} * P_{locomotiva} * (R'_{nl} + R'_{cl} + R'_r + R'_i) + N_{vagao} * P_{vagao} * (R'_{nv} + R'_{cv} + R'_r + R'_i)$$

$$R_{total} = 1 * 80 * (1,32 + 0 + 0 + 0) + 10 * 40 * (1,98 + 0 + 0 + 0)$$

$$R_{total} = 80 * 1,32 + 400 * 1,98$$

$$R_{total} = 105,6 + 792$$

$$R_{total} = 897,6 kgf$$

Onde:  $R_{total}$  = resistência total em kgf

$P_{locomotiva}$  = peso da locomotiva em toneladas

$N_{locomotiva}$  = número de locomotivas

$P_{vagao}$  = peso do vagão em toneladas

$N_{vagao}$  = número de vagões

$R'_{nl}$  = resistência normal da locomotiva em kgf

$R'_{cl}$  = resistência de curva da locomotiva em kgf

$R'_i$  = resistência de inércia em kgf

$R'_r$  = resistência de rampa em kgf

$R'_{nv}$  = resistência normal do vagão em kgf

$R'_{cv}$  = resistência de curva do vagão em kgf

**Cálculo da força aceleradora a 1 km/h:**

$$F_{ac} = F_{tratora} - R_{total}$$

$$F_{ac} = 15841,58 - 897,6$$

$$F_{ac} = 14943,98 \text{ kgf}$$

Onde:  $F_{ac}$  = força aceleradora em kgf

$F_{tratora}$  = força tratora ou esforço trator em kgf

$R_{total}$  = resistência total do veículo em kgf

**Cálculo da distância percorrida ( $\Delta l$  = deslocamento) para sair de 0 a 2 km/h:**

$$\Delta l_1 = \frac{4 * P_t * (V_f^2 - V_i^2)}{F_{ac}}$$

$$\Delta l_1 = \frac{4 * 480 * (2^2 - 0)}{14943,98}$$

$$\Delta l_1 = \frac{4 * 480 * 4}{14943,98}$$

$$\Delta l_1 = \frac{7680}{14943,98}$$

$$\Delta l_1 = 0,5139 = 0,514 \text{ metros}$$

Onde:  $P_t$  = peso total em toneladas

$V_f$  = velocidade final em km/h

$V_i$  = velocidade inicial em km/h

$F_{ac}$  = força aceleradora em kgf

**Cálculo do tempo decorrido ( $\Delta t$ ) até atingir 2 km/h:**

$$\Delta t_1 = \frac{7,2 * \Delta l}{V_i + V_f}$$

$$\Delta t_1 = \frac{7,2 * 0,514}{0 + 2}$$

$$\Delta t_1 = \frac{3,7008}{2}$$

$$\Delta t_1 = 1,850$$

$$\Delta t_1 = 1,85 \text{ segundos}$$

Onde:  $\Delta l = \text{deslocamento em metros}$

$V_f = \text{velocidade final em km/h}$

$V_i = \text{velocidade inicial em km/h}$

**Cálculos para um intervalo de velocidade  $\Delta v = 2 \text{ km/h}$ , ou seja, velocidade entre 2 e 4 km/h.**

$$V_m = 3 \text{ km/h}$$

**Cálculo da força tratora/esforço trator efetivo para 3 km/h:**

$$F_t = \frac{P_{locomotiva} * \text{Coeficiente}_{\text{atrito}}}{1 + 0,01 * V_m}$$

$$\frac{F_t = 80000 * 0,20}{1 + 0,01 * 3} = 15533,98 \text{ kgf}$$

**Conversão de milhas para quilômetros:**

$$V = 3 * 0,622 = 1,866 \text{ mi/h (milhas / hora)}$$

**Conversão toneladas para short-ton da locomotiva:**

$$\frac{P_{locomotiva}}{n} = \frac{80}{4} * 1,1 = 22 \text{ toneladas / eixo}$$

**Cálculo de resistência normal para a locomotiva a 3 km/h:**

$$R'_{nl} = 1,3 + \frac{29}{w} + 0,03 * V + \frac{0,0024 * A * V^2}{n * w}$$

$$R'_{nl} = 1,3 + \frac{29}{w} + 0,03 * 1,866 + \frac{0,0024 * 110 * 1,866^2}{4 * 22}$$

$$R'_{nl} = 1,3 + 1,3181 + 0,0559 + \frac{0,0024 * 110 * 3,4819}{88}$$

$$R'_{nl} = 1,3 + 1,3181 + 0,0559 + 0,0104$$

$$R'_{nl} = 2,6844 \text{ libras / toneladas}$$

$$R'_{nl} = 2,68 \text{ lb / t}$$

$$R'_{nl} = 2,68 * 0,5 = 1,34 \text{ kgf}$$

Onde:  $R'_{nl}$  = resistência normal da locomotiva em kgf

$w$  = peso médio por eixo em short-ton

$n$  = número de eixos

$A$  = área frontal em (sq-ft)

$V$  = velocidade da locomotiva em mi/h

### Conversão toneladas para *short-ton* do vagão:

$$\frac{P_{vaga\o}}{n} = \frac{40}{4} * 1,1 = 10 * 1,1 = 11 \text{ toneladas / eixo}$$

### Cálculo de resistência normal para o vagão a 3 km/h:

$$R'_{nv} = 1,3 + \frac{29}{w} + 0,045 * V + \frac{0,0005 * A * V^2}{n * w}$$

$$R'_{nv} = 1,3 + \frac{29}{11} + 0,045 * 1,866 + \frac{0,0005 * 85 * 1,866^2}{4 * 11}$$

$$R'_{nv} = 1,3 + 2,6363 + 0,0839 + \frac{0,0005 * 85 * 3,4819}{44}$$

$$R'_{nv} = 1,3 + 2,6363 + 0,0839 + 0,00033$$

$$R'_{nv} = 4,020 \text{ lb / t}$$

$$R'_{nv} = 4,02 * 0,5$$

$$R'_{nv} = 2,01 \text{ kgf}$$

Onde:  $R'_{nv}$  = resistência normal do vagão em kgf

$w$  = peso médio por eixo em short-ton

$n$  = número de eixos

$A$  = área frontal em (sq-ft)

$V$  = velocidade da locomotiva em mi/h

### Cálculo de resistência total: vagões e locomotiva a 3 km/h:

$$R_{total} = N_{locomotiva} * P_{locomotiva} * (R'_{nl} + R'_{cl} + R'_r + R'_i) + N_{vaga\o} * P_{vaga\o} * (R'_{nv} + R'_{cv} + R'_r + R'_i)$$

$$R_{total} = 1 * 80 * (1,34 + 0 + 0 + 0) + 10 * 40 * (2,01 + 0 + 0 + 0)$$

$$R_{total} = 80 * 1,34 + 400 * 2,01$$

$$R_{total} = 107,2 + 804$$

$$R_{total} = 911,20 \text{ kgf}$$

Onde:  $R_{total}$  = resistência total em kgf  
 $P_{locomotiva}$  = peso da locomotiva em toneladas  
 $N_{locomotiva}$  = número de locomotivas  
 $P_{vagao}$  = peso do vagão em toneladas  
 $N_{vagao}$  = número de vagões  
 $R'_{nl}$  = resistência normal da locomotiva em kgf  
 $R'_{cl}$  = resistência de curva da locomotiva em kgf  
 $R'_i$  = resistência de inércia em kgf  
 $R'_r$  = resistência de rampa em kgf  
 $R'_{nv}$  = resistência normal do vagão em kgf  
 $R'_{cv}$  = resistência de curva do vagão em kgf

### **Cálculo da força aceleradora a 3 km/h:**

$$F_{ac} = F_{tratora} - R_{total}$$

$$F_{ac} = 15533,98 - 911,20$$

$$F_{ac} = 14622,78 \text{ kgf}$$

Onde:  $F_{ac}$  = força aceleradora em kgf  
 $F_{tratora}$  = força tratora ou esforço trator em kgf  
 $R_{total}$  = resistência total do veículo em kgf

### **Cálculo da distância percorrida ( $\Delta l$ = deslocamento) para sair de 2 a 4 km/h:**

$$\Delta l_2 = \frac{4 * P_t * (V_f^2 - V_i^2)}{F_{ac}}$$

$$\Delta l_2 = \frac{4 * 480 * (4^2 - 2^2)}{14622,78}$$

$$\Delta l_2 = \frac{4 * 480 * 12}{14622,78}$$

$$\Delta l_2 = \frac{23040}{14622,78}$$

$$\Delta l_2 = 1,575 \text{ metros}$$

Onde:  $P_t$  = peso total em toneladas  
 $V_f$  = velocidade final em km/h  
 $V_i$  = velocidade inicial em km/h

$F_{ac}$  = força aceleradora em Kgf

**Cálculo do tempo decorrido ( $\Delta t$ ) até atingir 4 km/h:**

$$\Delta t_2 = \frac{7,2 * \Delta l_2}{V_i + V_f}$$

$$\Delta t_2 = \frac{7,2 * 1,575}{2 + 4}$$

$$\Delta t_2 = \frac{11,34}{6}$$

$$\Delta t_2 = 1,89 \text{ segundos}$$

Onde:  $\Delta l_2$  = deslocamento em metros

$V_f$  = velocidade final em km/h

$V_i$  = velocidade inicial em km/h

Então, calcula-se o tempo total e a distância percorrida até atingir a velocidade de 4 km/h será:

$$t_{total} = \sum \Delta t = 1,85 + 1,89 = 3,74s$$

Onde:  $t_{total}$  = tempo total em segundos

$$l_{total} = \sum \Delta l = 0,514 + 1,576 = 2,09m$$

Onde:  $l_{total}$  = distância percorrida em metros

## 2.6 Pontos de Aceleração, Consumo e LTKB

O ponto de aceleração de um veículo ferroviário funciona de modo análogo às marchas de um veículo rodoviário. Entretanto, os pontos de aceleração de uma locomotiva variam no intervalo de {1 a 8} pontos, e ainda possuem as marchas {Lenta baixa; Marcha Lenta; Freio Dinâmico}, em que cada ponto tem associado a si uma potência (HP) e um consumo (l/min). Os pontos de aceleração de uma locomotiva modelo C30, bem como as potências produzidas e os seus respectivos consumos, podem ser verificados na Tabela 1.

Tabela 1: Consumos e Potências da Locomotiva C30. Fonte: (LEITE, 2009, p. 14)

<b>Locomotiva C30</b>	<b>Potência (HP)</b>	<b>Consumo (l/min)</b>
Lenta Baixa	---	0,31
Marcha Lenta	---	0,31
Dinâmico	---	1,76
1	100	0,56
2	275	1,06
3	575	1,95
4	960	3,03
5	1440	4,53
6	1930	6,18
7	2500	7,69
8	2940	9,40

Por fim, calcula-se o valor do consumo para o deslocamento do veículo, que é obtido pela Equação 16, onde  $C$  é o consumo de litros (l) por segundo (s) (BRINA, 1982).

$$C = \frac{CPA/60}{T} \quad (16)$$

Onde:  $C$  = consumo em litros

$CPA$  = consumo do ponto de aceleração

$T$  = tempo do deslocamento em segundos

O cálculo do LTKB pode ser expresso por meio da Equação 17. Ele representa o consumo de litros por tonelada bruta transportada pelo veículo (Ibidem).

$$LTKB = \frac{CA}{P * (DA/1000)} * 1000 \quad (17)$$

Onde:  $LTKB$  = litro por tonelada bruta

$CA$  = consumo acumulado em litros durante a viagem

$P$  = peso do trem em toneladas

$DA$  = deslocamento em metros

Com estas informações, pode-se buscar a forma de condução ideal no tocante ao consumo de combustível. Em termos gerais, deve-se manter o esforço trator efetivo de forma que a locomotiva não patine e garanta uma aceleração. O esforço trator diminui à medida que a velocidade do veículo aumenta. A maximização ocorre quando o veículo se mantém próximo à velocidade máxima da via.

## 2.7 Freios de um trem

Um trem possui três tipos de freios, a saber: *independente, dinâmico e automático*, onde a pressão dos mesmos é medida em *psi (pound force per square inch)*. O freio independente é aquele cuja ação é realizada apenas nas locomotivas do veículo. Este freio não deve ser aplicado em velocidades superiores a vinte Km/h. Seu uso de forma inadequada pode causar defeitos nas rodas, graves acidentes e sua resposta de aplicação é imediata. A força de frenagem deste freio se concentra na parte da frente do trem.

Segundo (Brina, 1982), o freio dinâmico é recomendado para ser utilizado em descidas em rampas e tem finalidade de reduzir o uso dos freios a ar. Sua resposta de aplicação é em torno de dez segundos e o mesmo deve ser utilizado em velocidades entre vinte e trinta Km/h. Recomenda-se também sua utilização juntamente com o freio independente para paradas em descidas, com velocidade abaixo de dez Km/h para aumentar a eficiência de uso.

O freio automático compreende a aplicação da frenagem nos vagões e nas locomotivas do veículo. Este freio compreende o sistema de ar do trem, sendo que os componentes são instalados nos vagões e nas locomotivas do veículo.

## 2.8 Velocidade de Regime e Velocidade Crítica

Um veículo ferroviário poderá encontrar inúmeras situações em relação ao perfil da via em que esteja trafegando. O veículo poderá sofrer forças aceleradoras e retardadoras que modificam a velocidade atual, gerando uma nova velocidade que poderá ser *velocidade de regime* ou *velocidade crítica*.

O veículo, ao entrar em uma rampa  $r_2$  (atual) *inferior* à rampa  $r_1$  (anterior), sofre uma *força aceleradora*. Esta força ocasionará um aumento da velocidade e redução dos esforços

tratores até que estes (esforços tratores) se igualem aos esforços resistentes, ocasionando uma nova velocidade, denominada *velocidade de regime*. Esta representa a velocidade máxima que o veículo pode atingir neste trecho da via. De acordo com Brina (1982, p. 18), as forças aceleradoras diminuem com o aumento da velocidade, isto explica o fato de que, ao se aproximar da velocidade de regime, uma locomotiva tem mais dificuldade em aumentar a sua velocidade, do que quando esta se encontra em uma velocidade baixa.

O veículo, ao entrar em uma rampa r2 (atual) *superior* à rampa r1 (anterior), sofre uma *força retardadora*. Esta força fará com que o veículo diminua a velocidade até que este atinja uma velocidade denominada *velocidade crítica* (mínima para os motores de tração).

A Equação 18 expressa o equilíbrio entre os esforços tratores e resistentes para a composição do veículo que pode ser composta por vários tipos de vagões e locomotivas. Quando o equilíbrio destas forças ocorre em um traçado de rampa máxima, o veículo atingirá uma velocidade mínima (crítica). Porém, quando o equilíbrio destas forças ocorre em uma rampa mínima do traçado, o veículo atingirá uma velocidade máxima (regime).

$$\sum_{j=1}^l n_{loco_j} * E_j = \sum_{j=1}^l n_{loco_j} * P_{loco_j} (R'_{nl} + R'_{cl} + R'_i + R'_r) + \sum_{i=1}^v n_{vaga_i} P_{vaga_i} (R'_{nv} + R'_{cv} + R'_i + R'_r) \quad (18)$$

Onde:  $l =$  tipo de locomotiva

$v =$  tipo do vagão

$n_{loco_j} =$  número de locomotivas do tipo  $j$

$n_{vaga_i} =$  número de vagões do tipo  $i$

$E_j =$  esforço trator da locomotiva do tipo  $j$  em kgf

$P_{loco_j} =$  peso da locomotiva do tipo  $j$  (toneladas)

$P_{vaga_i} =$  peso do vagão do tipo  $i$  (toneladas)

## 2.9 Considerações Finais

Realizar uma boa política de ação é o objetivo principal de um condutor de trens. Contudo, observa-se que a força de tração, os esforços resistentes e as forças de aderência são fatores que influenciam na aplicabilidade de um ponto de aceleração que priorize, simultaneamente, a segurança, a economia de combustível e um menor tempo de viagem. No próximo capítulo serão apresentados os conceitos da teoria de agentes e da arquitetura de estados mentais BDI (*Belief, Desire, Intention*).

## Capítulo III

### 3 Agentes

#### 3.1 Introdução

Este capítulo apresenta os conceitos relacionados aos agentes. Os tópicos abordados irão descrever características importantes da fundamentação teórica de agentes, raciocínio prático e lógica BDI.

#### 3.2 Agentes de Software

Os agentes representam um novo conceito de análise, projeto e implementação de softwares. Um aspecto importante a ser considerado para utilizar agentes em sistemas é a abstração. Um agente está situado em algum ambiente com capacidade de ação autônoma e flexível responsável por suportar delegação de tarefas, adaptação ao meio e aprendizagem. Não obstante, os agentes devem perceber seu ambiente por meio de sensores e responder aos estímulos de acordo com as mudanças que podem ocorrer neste ambiente, ou ainda, tomar as devidas iniciativas perante o ambiente em que atua. Exemplos de ambientes que um agente pode estar situado incluem o mundo físico ou a Internet (JENNINGS; SYCARA; WOOLDRIDGE, 1998).

A autonomia é um conceito central em que o sistema deve agir sem a intervenção direta de humanos (ou outros agentes) e deve ter controle sobre suas próprias ações e estados internos. Os agentes devem interagir, quando necessário, com outros agentes e humanos de forma ordenada a resolver problemas ou auxiliar na execução de uma atividade. Há outros

aspectos importantes em estudo a se considerar sobre o conceito agente: mobilidade e adaptabilidade (Ibidem).

Ferber (1999) define um agente com as seguintes capacidades: (i) percepção, o agente deve ser capaz de perceber o ambiente em que ele se encontra; (ii) ação: o agente deve ser capaz de agir mudando o estado do ambiente; (iii) comunicação: o agente deve ser capaz de comunicar-se com outros agentes garantindo a coordenação durante a execução das tarefas; (iv) representação: o agente deve ser capaz de representar tudo aquilo em que acredita ser verdade sobre o ambiente e sobre os outros agentes que ele interage na sociedade; (v) motivação: são considerados os estados do mundo que o agente gostaria de atingir, por isso, representam os aspectos motivacionais, com os quais, dirigem o agente à busca da realização de um objetivo; e (vi) deliberação: o agente deve ser capaz de escolher a melhor alternativa de ação a ser executada para atingir um objetivo, dada à motivação e o conhecimento atual que ele tem sobre o ambiente.

Wooldridge (2002) considera um agente inteligente quando este apresenta as seguintes características: (i) pró-atividade: capacidade de executar ações em busca de seus objetivos; (ii) reatividade: capacidade de perceber o ambiente e reagir a suas mudanças buscando a realização dos seus objetivos; (iii) habilidade social: capacidade de interação com outros agentes, ou seja, considerar aquilo que sabe os outros agentes.

A Figura 1, de acordo com Wooldridge (2002, p. 16), representa “um agente em seu ambiente. Tal agente capta informações do ambiente por meio de sensores, produz um conjunto de ações e atua no ambiente por meio de atuadores”.

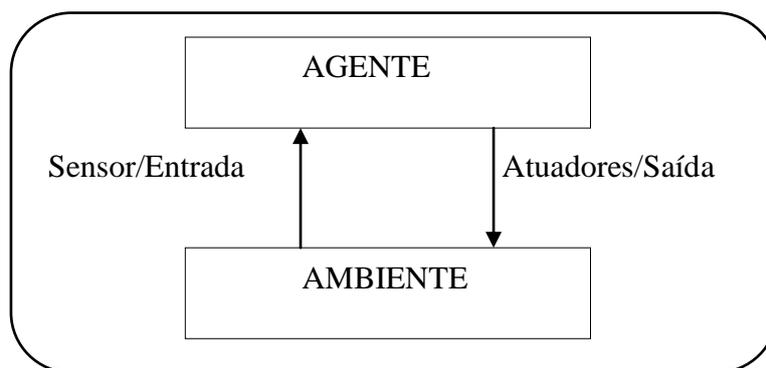


Figura 1: Um agente interagindo em seu ambiente. Fonte: (WOOLDRIDGE, 2002, p. 16).

Os agentes podem ser divididos em reativos e cognitivos. Os agentes reativos são baseados no estudo da entomologia (formigas, abelhas), sua principal característica é a reação

a estímulos gerados pelo ambiente (ação-reação), não possuem mecanismos de inferência ou aprendizado, mas o comportamento do grupo pode ser considerado inteligente. Os agentes cognitivos são baseados nas sociedades humanas, possuem representações do mundo e apresentam comportamentos inteligentes à medida que realizam a escolha das melhores ações para realizar um objetivo. Estes agentes incorporam a capacidade de aprendizagem por meio de suas experiências e autonomia na execução das tarefas. Atualmente, os agentes estão sendo aplicados em diversos sistemas, por exemplo, aplicações industriais, controle de processos, sistemas de telecomunicação, controle de tráfego aéreo, tráfego e administração de transporte, filtragem de informações, comércio eletrônico, negócios, processos administrativos, entretenimento, procedimentos médicos e em controle de satélites.

### **3.3 Por que Agentes de Software?**

O que diferencia um agente de um objeto, ou um agente de um sistema especialista? Um objeto é uma entidade computacional que encapsula um estado, realiza ações, troca mensagens por meio da passagem de parâmetros. Entretanto, existem algumas diferenças entre objetos e agentes, sendo a principal delas, que objetos não possuem controle sobre seus comportamentos, diz Wooldridge (2002, p. 25). Ainda, os objetos conceitualmente, não executam ações autônomas, não possuem capacidade de percepção e reação.

Um sistema especialista, de acordo com o mesmo autor, é capaz de resolver problemas ou auxiliar alguém em um domínio de conhecimento, mas não possui a capacidade de reatividade, comportamento proativo (agir em busca a realizar um objetivo) e não é equipado para habilidade social (interação com outros especialistas).

Entretanto, a necessidade de reação frente a uma mudança ocorrida no ambiente, ou a uma mudança nas regras de negócios ou na resolução de problemas complexos, justifica-se o uso de agentes de software (GEORGEFF et al., 1998). Baseado nestas observações, se o software a ser desenvolvido necessita realizar ações de forma autônoma, ou seja, sem a intervenção do usuário ou de outro software, também se justifica o uso do conceito agente.

### 3.4 Estados Mentais

Crença, desejo e intenção correspondem aos estados mentais que vincula um determinado agente ao mundo ao qual ele pertence. Tais estados estão diretamente relacionados com a intencionalidade, sendo considerados estados intencionais.

Dennett (1987) aborda os conceitos de intencionalidade para definir os fundamentos filosóficos para descrever agentes como entidades que são sujeitas a crenças, desejos e outras atitudes proposicionais, a saber:

- Crença é um estado mental que pode ser considerada verdadeira ou falsa, é uma representação do mundo. Uma crença pode ser aquilo que se acredita ser verdade. Ela é subjetiva à ideia de conhecimento.
- Desejo é considerado um estado mental que representa uma noção abstrata de um possível estado do mundo a ser alcançado. Um desejo pode ser inconsistente com outro desejo, e neste caso, ter um desejo não significa agir para atingi-lo. O desejo pode se considerado um estado motivador, pode ser causador de intenções e consequentemente de ações. A ideia de objetivo está associada a ele.
- A intenção assim como o desejo, pode ser considerada como um possível estado do mundo a ser atingido. Uma intenção conduz o comportamento à medida que ela pode estar associada a uma estratégia ou a um conjunto de ações a ser realizada pelo agente de forma a atingir o estado do mundo desejado.

### 3.5 Raciocínio Prático

O raciocínio prático representa um mecanismo inteligente de atuação para um agente. Este mecanismo propõe uma forma de direcionar o agente em suas tomadas de decisão, elaboração de metas e planos de ação para atingir seus objetivos. Portanto, com base em intenções iniciais o agente pode escolher a melhor opção de ações a ser executada.

Segundo Wooldridge (1999, p. 60):

“a arquitetura crença-desejo-intenção foi originada no trabalho da Agência Racional de *Stanford Research Institute* por volta de 1980. O modelo original da teoria humana do raciocínio prático foi desenvolvido por Bratman (1987), o qual focalizou particularmente nas regras das intenções no raciocínio prático”.

De acordo com Wooldridge (2002), o raciocínio prático consiste em duas atividades, *processo de deliberação* e *raciocínio meio-fim*. O processo de deliberação consiste em decidir qual o estado do mundo o agente deseja atingir. Pode ser dividido em:

- *gerar opções*: um conjunto de possíveis alternativas representando o conjunto de opções (desejos) de acordo com as crenças e intenções atuais do agente.
- *filtro*: função que se encarrega de selecionar alternativas consistentes com a meta que o agente deseja alcançar.

O raciocínio meio-fim consiste em dirigir o agente a decidir como alcançar estas metas. Esta etapa está relacionada com o processo de planejamento. Ele tem um papel muito importante no raciocínio prático, bem como no sucesso das ações realizadas pelo agente à medida que ele pode auxiliar na representação dos estados do mundo, dos objetivos e das intenções, dos planos e das ações. O problema no desenvolvimento do raciocínio prático do agente é no que se refere a abandonar intenções, à medida que é uma tarefa que apresenta perda de tempo e de recursos computacionais (WOOLDRIDGE, 2002).

Wooldridge (2009) diz que uma intenção direciona o raciocínio meio-fim. Então, se formada uma intenção, o agente deverá tentar alcançá-la, à medida que as intenções restringem deliberações futuras. O agente não deverá escolher opções que sejam incompatíveis com as intenções presentes. As intenções influenciam as crenças nas quais o raciocínio prático futuro está baseado, e se adotada uma intenção, o agente poderá planejar o futuro acreditando que alcançará esta intenção. Se o agente pretende alcançar algum estado do mundo acreditando simultaneamente que não a alcançará, então, o agente não está sendo racional.

Segundo Giraffa (1999), toda intenção é fundamental para o raciocínio prático. Ela poderá nascer de um desejo, ou na mudança de um estado mental provocado pela realização de uma ação e pela atualização da base de crenças. Para realizar uma ação, cada agente racional usa um plano, que é uma forma de influenciar na decisão de uma ação. O agente, conforme a sua necessidade ainda poderá usar um plano parcial para auxiliar na tomada de decisão e na execução de uma nova ação. De acordo com Wooldridge (2009), a lógica do raciocínio prático de um agente BDI, pode ser representada por meio de um algoritmo, que ordena seus diferentes procedimentos conforme descrito no Algoritmo 1:

```

1.   $B \leftarrow B_0$ ;    {B são crenças iniciais}
2.   $I \leftarrow I_0$ ;    {I0 são intenções iniciais}
3.  enquanto verdadeiro faça
4.       $P \leftarrow perceber()$  {Obter nova percepção}
5.       $B \leftarrow brf(B,p)$ ; {Revisar crenças}
6.       $D \leftarrow opções(B,I)$ ; {Gerar opções}
7.       $I \leftarrow filtro(B,D,I)$ ; {Filtrar alternativas}
8.       $\Pi \leftarrow plano(B,I,Ac)$ ; {Ac é um conjunto de ações}
9.      enquanto não (vazio( $\Pi$ ) ou sucesso ( $I,B$ ) ou impossível ( $I,B$ )) faça
10.          $\alpha \leftarrow$  primeiro elemento/ação de  $\Pi$ ;
11.         execute ( $\alpha$ );
12.          $\Pi \leftarrow$  remover elemento/ação de  $\Pi$ 
13.          $P \leftarrow perceber()$  {Obter nova percepção}
14.          $B \leftarrow brf(B,p)$ ; {Revisar crenças}
15.         se reconsiderar ( $I,B$ ) então
16.              $D \leftarrow opções(B,I)$ ; {Gerar opções}
17.              $I \leftarrow filtro(B,D,I)$ ; {Filtrar alternativas}
18.         fim-se
19.         se não alcançado( $\Pi,I,B$ ) então
20.              $\Pi \leftarrow plano(B,I,Ac)$  {Replanejar}
21.         fim-se
22.     fim-enquanto
23. fim-enquanto

```

Algoritmo 1: Lógica BDI. Fonte: (WOOLDRIDGE, 2009).

De acordo com o Algoritmo 1, pode-se definir o raciocínio prático de um agente BDI. De forma detalhada, a expressão  $B \leftarrow B_0$ , da linha 1, representa as *crenças* iniciais, as quais farão parte da base de crenças do agente. A expressão  $I \leftarrow I_0$  da linha 2, representa as *intenções* iniciais do agente. Logo em seguida, nas linhas 3 e 4, inicia-se um laço de repetição. Este último define o ciclo de vida do agente e a percepção do agente via um conjunto de sensores, respectivamente. A expressão  $B \leftarrow brf(B,p)$ , da linha 5, representa a função de revisão de crenças (*brf*), que atualiza as crenças à base de crenças do agente. A expressão  $D \leftarrow opções(B,I)$ , na linha 6, representa a função *geradora de opções* de novos *desejos* ( $D$ ) do agente. Esta função define o processo de deliberação do agente. A expressão  $I \leftarrow filtro(B,D,I)$ , da linha 7, representa uma outra função do processo de deliberação do agente, chamada de

função *filtro*, que seleciona o novo desejo gerado ( $D$ ) de acordo com a nova crença ( $B$ ), e intenções iniciais ( $I$ ), transformando-os em uma nova intenção ( $I$ ); e por fim, na linha 8, a expressão  $\pi \leftarrow plano(B, I, Ac)$  representa o processo de geração de planos, que associa um plano de ação ( $Ac$ ) ao desejo, ou seja, a intenção selecionada. Dado que em um plano podem existir várias ações, a satisfação de um desejo do agente requer a execução de cada ação constante no referido plano. O laço de repetição, das linhas 9 a 22, representa tal iteração sobre o plano. Então, o laço é interrompido quando: (i) não houver mais nenhuma ação no plano a ser executada; (ii) for alcançado o objetivo, ou (iii) ocorrer à impossibilidade de resolvê-lo. As expressões  $\alpha \leftarrow primeiro\ elemento/ação\ de\ \pi$  e  $execute(\alpha)$ , das linhas 10 e 11, representam respectivamente a subtração de uma ação do plano e a execução da mesma. Na linha 13, o agente realiza uma nova percepção e a expressão  $B \leftarrow brf(B, p)$  da próxima linha, adiciona essa nova crença ( $B$ ) na base por meio da função revisão de crenças (*brf*). Na linha 15, realiza-se um teste para verificar se essa nova crença deve ser considerada. Finalmente, caso o objetivo selecionado não tenha sido satisfeito com o plano presente, então um novo plano deve ser gerado.

Ao longo da evolução da teoria de agentes BDI, aplicações, ferramentas e linguagens surgiram de forma a possibilitar a aplicação destes conceitos, dentre os quais se destacam algumas das principais: (i) IRMA (BRATMAN; POLLACK; ISRAEL, 1988); (ii) PRS (GEORGEF; INGRAND, 1989); (iii) AgentSpeak(L) (RAO, 1996); e (iv) Jason (BORDINI; HÜBNER; WOOLDRIDGE, 2007).

### 3.5.1 IRMA

O IRMA (*Intelligent Resource-bounded Machine Architecture*) (BRATMAN; POLLACK; ISRAEL, 1988), contribuiu definindo um framework para o desenvolvimento de agentes utilizando o raciocínio prático e planos para a execução de ações específicas. É um modelo que descreve os conceitos fundamentais, como crença, desejo, intenção, planos, e ainda, o processo de raciocínio prático. Sua grande contribuição está baseada no desenvolvimento de um sistema abstrato que controla a execução interna de um agente à medida que serve como base para outros modelos. Seus principais módulos são:

- uma biblioteca de planos;
- uma estrutura de representação das crenças;
- uma estrutura de representação dos desejos;

- uma estrutura de representação das intenções;
- um mecanismo de raciocínio sobre argumentações do mundo;
- uma máquina de estado;
- um analisador meio-fim para analisar quais planos podem ser aplicados para realizar uma intenção;
- um analisador de oportunidades para monitorar o ambiente, e com as suas mudanças gera novas opções;
- um filtro para determinar quais ações são compatíveis com as intenções atuais; e
- um processo de deliberação para ajudar na seleção das melhores intenções a serem adotadas.

A Figura 2 ilustra de forma abstrata os componentes do IRMA.

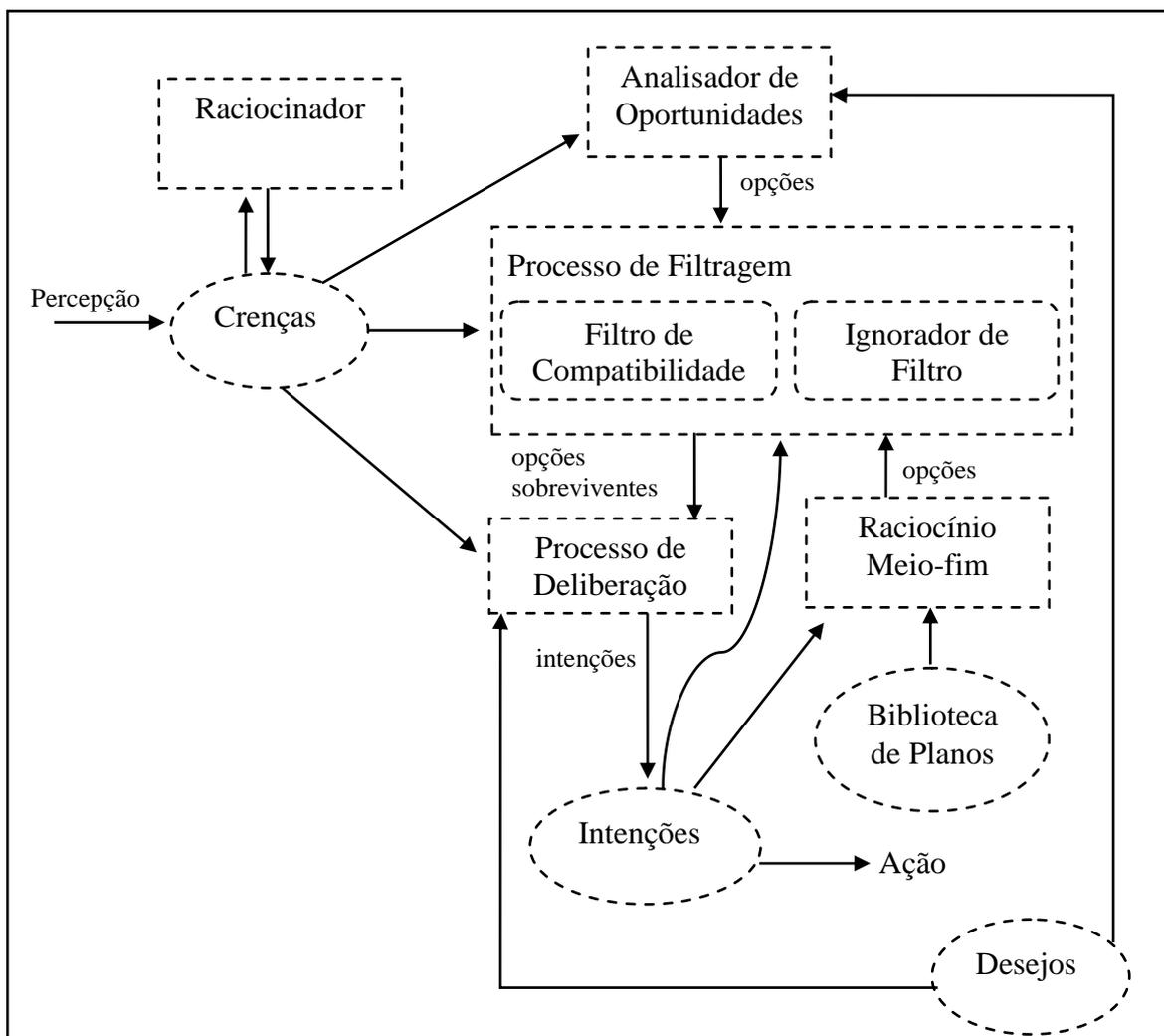


Figura 2: IRMA. Fonte: (BRATMAN; POLLACK; ISRAEL, 1988).

### 3.5.2 PRS

O PRS (*Procedural Reasoning System*) é provavelmente uma das arquiteturas híbridas mais conhecidas. Ele foi desenvolvido na Universidade de Stanford com o objetivo de resolver o problema do comportamento de sistemas em tempo real que até então não proporcionavam precisão nas respostas, principalmente em ambientes incertos (GEORGEFF; INGRAND, 1989). Como exemplo de uma aplicação, tais autores citam a utilização do PRS no auxílio ao sistema de transporte RCS (*Reaction Control System*) da NASA, proporcionando aos astronautas planos de ação para as eventuais falhas deste sistema em sua viagem espacial. O PRS comprovou seu desempenho sendo aplicado em diversos tipos de controle de processos, tais como: controle de tráfego aéreo, e redes de telecomunicações. Os autores destacam suas principais características, a capacidade de decisão e operações em tempo real por meio de um conjunto de componentes, a saber:

- uma base de dados para persistir as crenças ou fatos sobre o mundo;
- um conjunto de metas a serem realizadas;
- um conjunto de planos para descrever como certas ações podem ser executadas e testadas para atingir uma determinada meta ou reagir situações particulares;
- uma estrutura de intenção para conter planos, estes planos são escolhidos de acordo com uma determinada intenção;
- um mecanismo de conclusão ou interpretação para selecionar planos baseados nas crenças do sistema e para selecionar objetivos de acordo com a estrutura intenção, colocando-os em ação.

O sistema interage com seu ambiente incluindo outros sistemas por meio de sua base de crenças, onde adquire novas crenças com respeito a mudanças no ambiente e pelas ações que executa seguindo suas intenções. Uma descrição mais detalhada pode ser encontrada em Georgeff e Ingrand (1989). A Figura 3 ilustra de forma abstrata o sistema PRS.

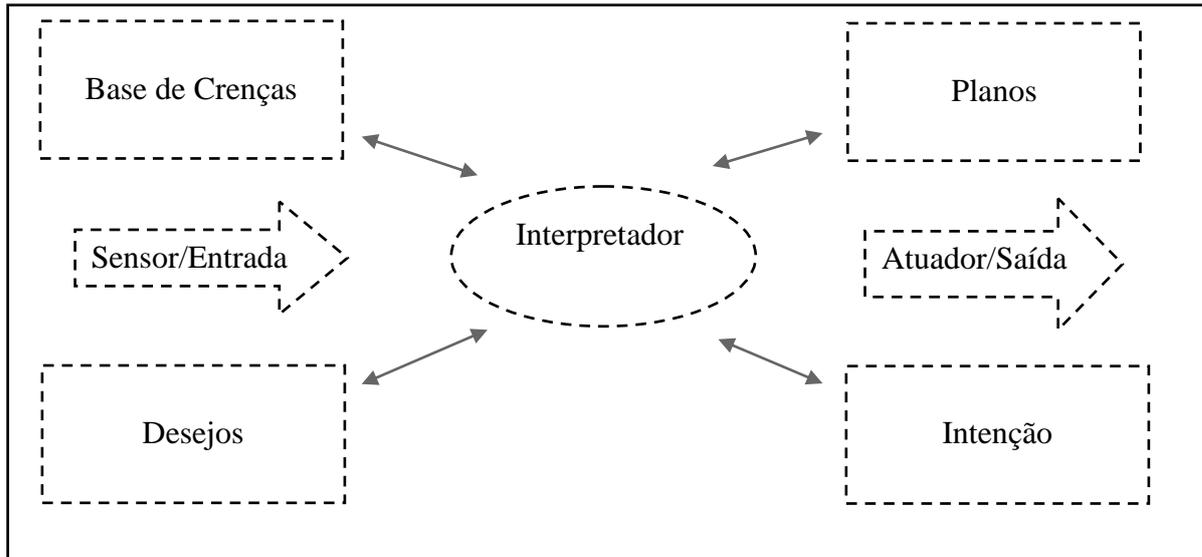


Figura 3: Arquitetura do sistema PRS. Fonte: (GEORGEFF; INGRAND, 1989, p. 02).

### 3.5.3 AgentSpeak(L)

A linguagem AgentSpeak(L) é uma extensão natural da programação lógica para a arquitetura BDI, onde propõe-se um *framework* abstrato para a programação de agentes BDI (HÜBNER; BORDINI; VIEIRA, 2004). Foi apresentada em Rao (1996), podendo ser considerada como uma abstração do PRS. A linguagem tem como principal característica o planejamento reativo, ou seja, a reação aos eventos ocorridos no ambiente é realizada por meio de planos de ações previamente instanciados (HÜBNER; BORDINI; VIEIRA, 2004).

De acordo com Bordini et al. (2005) esta linguagem é considerada como umas das principais técnicas para desenvolvimento de agentes BDI, proporcionando o desenvolvimento de aplicações em tempo real. O AgentSpeak(L) é uma linguagem de programação baseada em restrições de primeira ordem com eventos e ações. O alfabeto formal da linguagem consiste em variáveis, constantes, símbolos funcionais, predicados simbólicos, símbolos de ações, conectivos, quantificadores e símbolos de pontuação (RAO, 1996).

É importante mencionar que nesta seção, a interpretação realizada está baseada nos trabalhos desenvolvidos por ((BORDINI; VIEIRA, 2003); (BORDINI et al., 2005); (BORDINI; HÜBNER, 2006); (HÜBNER; BORDINI; VIEIRA, 2004)), em função de que a partir de tais explicações optou-se por iniciar tal discussão a partir dos conceitos relacionados ao símbolos utilizados para a especificação da linguagem AgentSpeak(L). Desta forma, são usados como conectivos de primeira ordem os seguintes símbolos: ! (para realização), ? (para

teste), ; (para sequência),  $\leftarrow$  (para implica). A especificação de um conjunto de crenças formam a base de crenças iniciais e planos. Um átomo de crença é um predicado de primeira ordem na notação da lógica usual, e literais de crenças são átomos de crenças ou suas negações. A base de crenças de um agente é uma coleção de átomos de crença (RAO, 1996).

No AgentSpeak(L) há dois tipos de objetivos: *objetivos de realização* (*achievement goals*) representados pelo operador “!” e *objetivos de teste* (*test goals*) representados pelo operador “?”. “Os objetivos de realização expressam que o agente quer alcançar um estado no ambiente onde o predicado associado ao objetivo é verdadeiro” (HÜBNER; BORDINI; VIEIRA, 2004). Entretanto, estes objetivos iniciam a execução de sub-planos, e os objetivos de teste “retornam a unificação do predicado de teste com uma crença do agente, ou falha caso não seja possível à unificação com nenhuma crença do agente” (Ibidem).

De acordo com Bordini et al. (2005, p. 8) “um agente AgentSpeak é sistema de planejamento reativo”. Os eventos podem ser internos e externos. Um *evento* interno pode ser gerado pela execução de um plano em que um sub-objetivo precisa ser alcançado. Um evento externo pode ser gerado pela atualização de crenças provocadas pela percepção do ambiente (HÜBNER; BORDINI; VIEIRA, 2004).

Um *evento ativador* (*triggering event*) pode iniciar a execução de um plano e define quais eventos podem adicionar ou remover objetivos e crenças. Existem dois tipos de *triggering event*: a adição e remoção de crenças e a adição e remoção de objetivos, são representadas pelos operadores (‘+’) e (‘-’) (HÜBNER; BORDINI; VIEIRA, 2004).

Os planos referem-se a ações básicas que um agente pode executar em seu ambiente. Tais ações são definidas como predicados de primeira-ordem, mas com símbolos de predicado especiais (chamados símbolos de ação) que as distingue de outros predicados. Um plano é formado por um evento ativador (denotando o propósito daquele plano), seguido por uma conjunção de literais de crença que representam um contexto. O contexto deve ser uma consequência lógica das crenças atuais do agente para que o plano possa ser aplicável. O resto do plano é uma sucessão de ações básicas ou sub-objetivos que o agente tem que alcançar, ou testar quando o plano aplicável é escolhido para execução.

```

+concert(A,V) : likes(A)
← !book tickets(A,V).

!book tickets(A,V) : ¬busy(phone)
← call(V);

```

Figura 4: Planos em AgentSpeak(L). Fonte: (BORDINI; VIEIRA, 2003).

A Figura 4 ilustra dois exemplos de planos em AgentSpeak(L). Inicialmente, no topo da figura a expressão **+concert(A,V) : likes(A)← !book tickets(A,V).**, representa um plano para a reserva de ingressos para o show do artista A no local V, previamente com a identificação do agente gostar do artista A, **:likes(A)**.

A expressão **+concert(A,V)** representa um evento ativador, neste caso, a adição de uma crença. A expressão: **likes(A)** representa o contexto do plano, estando ligado a crença do agente, e juntamente com a primeira expressão, fazem parte da cabeça do plano. Por último, a expressão **←!book tickets(A,V)** representa o corpo do plano, neste caso, um objetivo de realização: reservar os ingressos do artista A no local V.

O segundo plano dá continuidade a reserva dos ingressos, pois, ainda não foi realizada a reserva dos mesmos. A expressão **!book tickets(A,V) : ¬busy(phone) ← call(V);**, representa, caso o telefone não esteja ocupado, **¬busy(phone)**, após percepção do ambiente **!book tickets(A,V)**, que o agente deverá entrar em contato no local, através de **← call(V);**. Assim, seguindo a um determinado protocolo de reserva de ingressos (indicado por ‘. . .’), terminará com a execução de um subplano para a escolha de acentos em eventos desse tipo naquele local, representados pela expressão **!choose seats(A,V)** (HÜBNER; BORDINI; VIEIRA, 2004). A Figura 5 ilustra a sintaxe da linguagem AgentSpeak(L).

$ag ::= bs \ ps$	
$bs ::= b_1 \dots b_n$	$(n \geq 0)$
$at ::= P(t_1, \dots, t_n)$	$(n \geq 0)$
$ps ::= p_1 \dots p_n$	$(n \geq 1)$
$p ::= te : ct \leftarrow h$	
$te ::= +at \ / \ -at \ / \ +g \ \ / \ -g$	
$ct ::= at \ / \ \neg at \ / \ ct \wedge ct \ / \ \mathbf{T}$	
$h ::= a \ \ / \ g \ \ / \ u \ \ / \ h; h$	

Figura 5: Sintaxe de AgentSpeak(L). Fonte: (HÜBNER; BORDINI; VIEIRA, 2004).

Em AgentSpeak(L) o agente ( $ag$ ) é formado por um conjunto de crenças ( $bs$ ) e um conjunto de planos ( $ps$ ), ele é representado pela expressão:  $ag ::= bs \ ps$ . Fórmulas atômicas ( $at$ ) da linguagem são predicados onde  $P$  é um símbolo predicativo e  $(t_1 \dots t_n)$  são termos padrão da lógica de primeira ordem, são representadas pela expressão  $at ::= P(t_1 \dots t_n)$ . Uma crença é uma fórmula atômica ( $at$ ) sem variáveis, e ( $b$ ) é meta-variável para crenças. Um conjunto inicial de crenças ( $bs$ ) são formadas por uma sequência de crenças ( $at_1 \dots at_n$ ), representadas pela expressão:  $bs ::= b_1 \dots b_n$ .

Um conjunto de planos ( $ps$ ) é composto por uma sequência de planos ( $p$ ), sendo representado pela expressão  $ps ::= p_1 \dots p_n$ . Um plano ( $p$ ) é composto por um *triggering event* ( $te$ ) é o (evento ativador), pelo contexto do plano ( $ct$ ) é (uma conjunção de literais de crença) e por uma sequência de ações, objetivos ou atualizações de crenças ( $u$ ). A construção ( $te : ct$ ) é chamado cabeça do plano, e ( $h$ ) é o corpo do plano, sendo representado por  $p ::= te:ct \leftarrow h$ . Um evento ativador ( $te$ ) corresponde à adição e remoção de crenças do agente ( $+at$  e  $-at$ ), ou a adição e remoção de objetivos ( $+g$  e  $-g$ ). É representado pela expressão  $te ::= +at \ / \ -at \ / \ +g \ / \ -g$ .

De acordo com Bordini e Vieira (2003), cada plano tem em sua cabeça uma fórmula ( $ct$ ) – contexto do plano que especifica as condições para as quais o plano possa ser executado. Portanto, a fórmula ( $ct$ ) deve ser uma consequência lógica das crenças do agente

para o plano ser considerado aplicável,  $(ct)$  é representado pela expressão  $ct ::= at / \neg at / ct \wedge ct / T$ .

A sucessão  $(h)$  significa ações, objetivos e atualização de crenças e define o corpo do plano, sendo representada pela expressão  $h ::= a / g / u / h; h$ . O agente possui um conjunto de *ações básicas* que utiliza para atuar sobre o ambiente, as ações são representadas por  $a ::= A (t_1 \dots t_n)$  (HÜBNER; BORDINI; VIEIRA, 2004).

O símbolo de ação  $(A)$  é referido por predicados usuais sendo usados no lugar do símbolo predicativo. Os objetivos  $(g)$  podem ser objetivos de realização  $(!at)$  ou de teste  $(?at)$ , são representados pela expressão  $g ::= !at / ?at$ .

Os símbolos  $(+at)$  adição de crenças e  $(-at)$  remoção de crenças no corpo de um plano, representam operações de atualização da base de crenças, é representado pela expressão  $u ::= + at / - at$ .

Uma fórmula  $(!g)$  no corpo de um plano gera um evento cujo evento ativador é  $(+!g)$ . Portanto, planos escritos pelo programador que tenha um evento ativador que possa ser unificado com  $(+!g)$  representam alternativas de planos que devem ser considerados no tratamento de tal evento.

Planos com evento ativador do tipo  $(+at)$  e  $(-at)$  são utilizados no tratamento de eventos que são gerados quando crenças são adicionadas ou removidas (tanto como consequência da percepção do ambiente, como devido às alterações de crenças explicitamente requisitadas no corpo de um plano).

Eventos ativadores do tipo  $(-!g)$  são usados para o tratamento de falhas em planos, e eventos ativadores do tipo  $(+?g)$  e  $(-?g)$  não são “utilizados na implementação atual da linguagem”, (HÜBNER; BORDINI; VIEIRA, 2004).

Os mesmos autores afirmam que um interpretador para a linguagem AgentSpeak(L) deve ter acesso a sua base de crenças e a biblioteca de planos, e ainda gerenciar o conjunto de eventos e intenções. Seu funcionamento requer três funções de seleção:

- Função de seleção de eventos ( $S_E$ ) que faz a seleção de um único evento do conjunto de eventos;
- Função ( $S_{AP}$ ) que faz a seleção de um plano dentre o conjunto de planos aplicáveis para um evento selecionado;
- Função ( $S_I$ ) que faz a seleção de uma intenção do conjunto de intenções.

A Figura 6 ilustra o ciclo de atividades de um programa AgentSpeak(L).

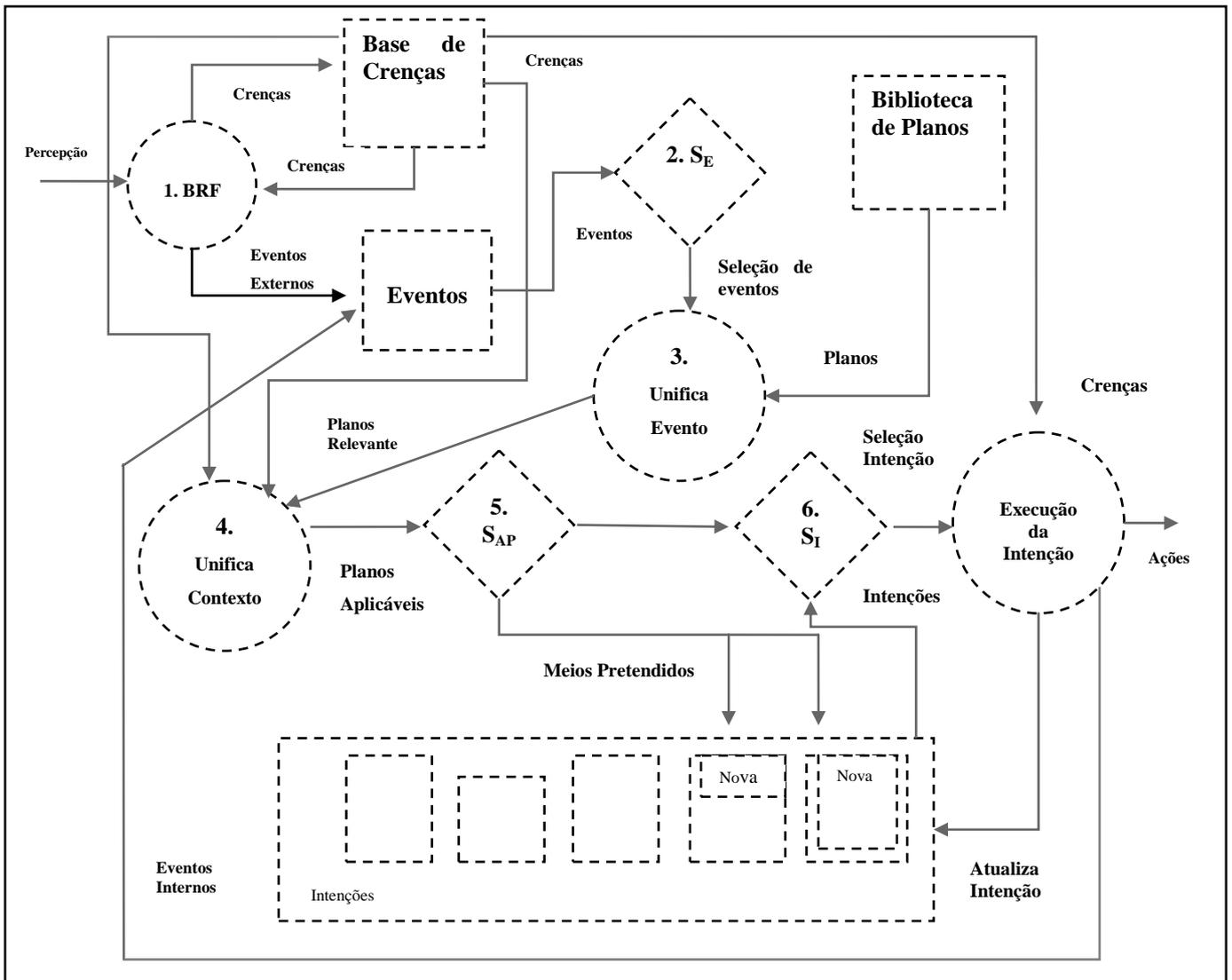


Figura 6: Ciclo da interpretação de um programa AgentSpeak(L). Fonte: (MACHADO; BORDINI, 2002).

Com base na Figura 6, após o passo (1) percepção do ambiente, há a atualização da base de crenças (BRF), e novos eventos são gerados, (2) a função ( $S_E$ ) seleciona o evento, (3) AgentSpeak (L) deverá unificar o evento com eventos ativadores nas cabeças dos planos que estão na biblioteca de planos. Será gerado então, um conjunto de planos aplicáveis para o evento escolhido. (5) A função ( $S_{AP}$ ) seleciona um único plano aplicável que se torna a razão para o tratamento do evento selecionado. Se o evento for interno, insere-se o plano no topo de uma intenção existente, em contrapartida, se o evento for externo, cria-se uma nova intenção no conjunto de intenções. (6) A função ( $S_I$ ) seleciona uma única intenção para ser executada.

Os planos parcialmente instanciados encontram-se dentro do conjunto de intenções. O evento acompanha a intenção gerada (o plano que foi escolhido para o evento será inserido no topo da intenção). Uma ação será executada pelo agente em seu ambiente, um evento interno será gerado, por um objetivo de realização, ou por um objetivo de teste (acesso à base de crenças). “Caso a fórmula seja um objetivo de realização, simplesmente um evento do tipo de adição de objetivo de realização é adicionado ao conjunto de eventos, acompanhado da intenção que gerou o evento” (HÜBNER; BORDINI; VIEIRA, 2004). A intenção deverá ser removida, ficando suspensa até que o evento interno seja escolhido. Após a escolha do plano para tratar este evento, o plano deverá ser inserido no topo da pilha de planos da intenção, e esta retornada ao conjunto de intenções (Ibidem).

Caso a escolha seja um objetivo de teste a ser realizada ou uma ação básica a fórmula do corpo da instância do plano deve ser removida do topo da intenção selecionada. Na execução de um objetivo de teste, a base de crenças será verificada com o objetivo de encontrar um átomo de crença que possa ser unificada com o predicado de teste. Se esta unificação for possível “instanciações de variáveis podem ocorrer no plano parcialmente instanciado; [...] o objetivo de teste pode ser removido do conjunto de intenções, pois já foi realizado” (HÜBNER; BORDINI; VIEIRA, 2004).

Após a seleção de uma ação básica, será necessário remover a ação do conjunto de intenções, e o interpretador deverá informar ao componente da arquitetura responsável a ação a ser executada. Quando todas as fórmulas no corpo de um plano forem executadas e removidas, o plano deverá ser removido da intenção, assim como o objetivo de realização que o gerou.

“o ciclo de execução termina com a execução de uma fórmula do corpo de um plano pretendido, e AgentSpeak(L) começa um novo ciclo, com a verificação do estado do ambiente após a ação do agente sobre ele, a geração de eventos adequados, e continuando a execução de um ciclo de raciocínio do agente [...]” (Ibidem).

### 3.5.4 Jason

Jason é um interpretador para a extensão da linguagem AgentSpeak(L) possibilitando o desenvolvimento de agentes BDI através do planejamento reativo.

De acordo com Bordini et al. (2005), o interpretador Jason possui os seguintes recursos:

- Comunicação baseada em atos de fala (incluindo informações de fontes como anotações de crenças e planos);
- Execução de sistema multiagente distribuídos em rede;
- Negação forte (*strong negation*), possibilidade tanto de sistemas que consideram mundo-fechado (*closed-world*) quanto mundo-aberto (*open-world*);
- Tratamento de falhas em planos e notações em identificadores de planos, que podem ser utilizadas na elaboração de funções personalizadas para seleção de planos;
- Suporte para o desenvolvimento de ambientes. No Jason o ambiente é programado em Java;
- Possibilidade de especializar em Java as funções de seleção de planos, funções e toda a arquitetura do agente (percepção, revisão de crenças, comunicação e atuação);
- Possui uma biblioteca básica de “ações internas”;
- Possibilita a extensão da biblioteca de ações internas programadas em Java.

Bordini et al. (2005) relatam algumas diferenças entre o Jason e o AgentSpeak(L), dentre elas é que no Jason utiliza-se literais ao invés de predicados como na linguagem original. Um literal é uma fórmula atômica  $p(t_1, \dots, t_n)$ ,  $n \geq 0$ , ou  $\sim p(t_1, \dots, t_n)$  onde ‘ $\sim$ ’ denota negação forte. A expressão “not” antecedendo a um literal representa a negação default no contexto de um plano. O conjunto de literais formam um contexto, os termos  $t_i$  podem ser variáveis, listas, um átomo, uma string, um número real ou um inteiro.

Bordini et al. (2005) explicam que no Jason as fórmulas atômicas podem ter anotações. Estas servem para “registrar as fontes das informações”, por exemplo, na base de crenças elas podem representar o ambiente, estruturas e outros agentes. Estas anotações são especificadas entre colchetes [...].

O átomo *percept* é usado para representar que uma crença surgiu da percepção do ambiente, e *self* representa que houve a inclusão de uma crença na base de crenças após a execução de um plano. Porém, segundo os mesmos autores, por causa das anotações, as unificações podem tornar-se um pouco mais complicadas.

Outra diferença do AgentSpeak(L) original citada por eles é que os agentes também podem ter metas iniciais, e a base de crenças pode ter regras e a mesma sintaxe usada no corpo do plano poderá ser utilizada no contexto do plano.

“As ações internas são ações definidas pelo usuário que executam internamente no agente; podem ser usadas no contexto e no corpo de um plano. São denotados como ações internas os sinais ‘.’ no início, e ‘.’ em qualquer lugar. Ações básicas representam à atuação do agente, estão no corpo de um plano e denotam ações que um agente deseja executar para alterar o ambiente. As ações internas no Jason são definidas pelo usuário na linguagem Java. Como exemplos de ações da biblioteca padrão destacam: (.send, .print, . desire)” (HÜBNER; BORDINI; VIEIRA, 2004).

A ação **.send** é usada por um agente para enviar uma mensagem. A ação **.print** é usada para imprimir uma mensagem e a ação **.desire** recebe um literal como argumento e tem sucesso se o literal é um dos desejos do agente (Ibidem).

De maneira geral, os modelos BDI oferecem um mecanismo de coordenação das atividades do raciocínio de um agente, como percepção do ambiente, revisão de crenças, deliberação e planejamento. Este último é uma das principais subáreas da IA e possui como objeto de investigação, as estratégias de como organizar um conjunto de ações de forma a resolver um problema complexo. Após realizar a deliberação, processo que gera e filtra um novo objetivo consistente com sua crença, o agente determina qual o plano de ação deve ser aplicado para alcançar tal objetivo.

### 3.6 Planejamento

O planejamento integra o raciocínio prático do agente “o planejamento é o raciocínio sobre como agir” (GHALLAB; NAU; TRAVERSO, 2004, p. 1). No entanto, o gênero de ação determina o tipo de planejamento a ser aplicado. Os tipos de planejamento mais comuns são: (i) planejamento do movimento, que é voltado ao planejamento do controle de ações a serem aplicadas em sistemas dinâmicos e complexos que satisfaça uma restrição; (ii) planejamento da percepção, que envolve planos de ação para a percepção das informações provenientes do ambiente através dos sensores do agente; (iii) planejamento da navegação, que combina as duas técnicas citadas anteriormente. Em geral, esta agrupa um conjunto de regras (localização e movimento) que orientam o deslocamento de um agente de um ponto inicial até um ponto final; (iv) planejamento da manipulação, que se preocupa com a manipulação de objetos. Ele

envolve planos de ação para sensores que fornecem dados como, força, visão e alcance; e finalmente, (v) planejamento da comunicação, que está relacionado aos planos de cooperação entre agentes como as consultas e as trocas de informação entre eles (GHALLAB; NAU; TRAVERSO, 2004, p. 2).

Um agente ainda deverá elaborar um plano de ação quando verificar algumas necessidades como: (i) realizar uma tarefa complexa e conflitante neste ambiente; (ii) gerar novas tarefas (sub-tarefas) à partir de outras já realizadas; e (iii) apresentar um comportamento autônomo (organização de ações). Muitas técnicas de planejamento para agentes foram amplamente estudadas, com o objetivo de tornar o raciocínio do agente mais eficiente e inteligente. Uma das primeiras teorias de planejamento, também definida como teoria clássica, se refere ao planejamento para restringir o estado-transição do sistema ligando o estado inicial ao objetivo do agente. Este tipo de planejamento resolve efetivamente muitos problemas de domínio. Para reduzir o espaço de busca e problemas desta abordagem, a linguagem denominada STRIPS foi idealizada e apresentada em Nilson e Fikes (1971).

O planejamento conhecido como uma rede hierárquica de tarefas-HTN (*Hierarchical Task Network*) foi apresentada em Sacerdoti (1977). A ideia principal desta abordagem é a decomposição de tarefas em sub-tarefas primitivas em uma rede hierárquica, onde estas se relacionam e são executadas por meio de métodos consistentes para um objetivo específico.

Para a criação de um plano, o planejador HTN usa a decomposição de tarefas, ao qual, decompõe as tarefas em sub-tarefas até que estas se tornem tarefas primitivas. Uma tarefa primitiva é uma tarefa que pode ser executada diretamente pelo planejador. Todo sistema HTN, possui uma base de métodos de planejamento. Cada método define a forma de como decompor uma tarefa em uma rede de sub-tarefas, contendo restrições de como satisfazer a condição para que o método possa ser aplicado, e restrições de relacionamento entre tarefa e sub-tarefas (NAU; SMITH; EROL, 1998).

Dada uma tarefa a ser realizada, o planejador seleciona um método aplicável e o instancia para decompor esta tarefa em sub-tarefas. Em seguida, o planejador gera e seleciona outros métodos para decompor as sub-tarefas geradas. Caso haja restrições entre as interações entre as tarefas e suas sub-tarefas que impossibilite a execução de um plano, o planejador realiza *backtracking* e tenta aplicar outro método. Como exemplos de planejadores que implementam o HTN destacam-se o SHOP2 (NAU et al., 2003) e o JSHOP2 (ILGHAMI, 2005).

### 3.7 Arquitetura BDI

Em Georgeff et al. (1998), os autores discutem a viabilidade, a eficiência e a aplicabilidade da arquitetura BDI. Os principais assuntos abordados são:

- o fato da arquitetura BDI não realizar nenhuma consideração a respeito do projeto para o comportamento de um sistema multiagente;
- a dificuldade da arquitetura BDI que não é apropriada para a construção de um sistema que necessite aprender e adaptar-se a mudanças de comportamentos;
- a questão de como incorporar na arquitetura BDI o projeto e a modelagem do comportamento de um sistema multiagente;
- como entender e incorporar a teoria com a realização prática da arquitetura BDI, tal a exigência que poderia levar a arquitetura BDI sair do papel e ser utilizada por Engenheiros de Software; e como é possível estabelecer a relação da arquitetura BDI com modelos híbridos.

Georgeff et al. (1998) afirma que devido complexidade dos sistemas atuais somente agentes de software terão condições de atender a estas necessidades. Crença, desejo, intenção e planos são os centros das atenções para estes sistemas. Em IA, crença é a representação do conhecimento dos estados do mundo. Na informática tradicional, crenças poderiam ser consideradas como o valor de uma variável, uma relação em uma base de dados ou uma expressão simbólica do cálculo de predicado. Para ele, as crenças são fundamentais, pois, elas possibilitam a representação do mundo, em contrapartida, sistemas convencionais possuem apenas uma visão local.

Georgeff et al. (1998) ainda levanta algumas considerações a respeito dos conceitos que envolvem a arquitetura BDI e cita a importância dos modelos IRMA e PRS para a construção de agentes. Salientam a possibilidade de se construir aplicações de sucesso utilizando a arquitetura BDI clássica e com o uso de modelos híbridos. Citam a importância dos planos, sub-planos e das tarefas do raciocínio prático como o raciocínio meio-fim.

Lincoln et al. (2010) apresentam a concepção de um agente BDI a ser aplicado no controle de veículos autônomos e em robôs para a exploração espacial. Abordam o problema da complexidade em se planejar uma ação futura quando um agente toma uma decisão baseada apenas no estado atual do ambiente, visto que este pode mudar tornando uma ação

inadequada. Afirmam que a utilização de uma estratégia de controle adaptativo do modelo preditivo pode auxiliar nas tomadas de decisões baseadas na previsão do estado futuro de um agente. Citam a importância da arquitetura BDI no enriquecimento do raciocínio do agente e citam o exemplo do PRS como um controlador de destaque desta arquitetura. Ainda justificam o uso de agentes quando estes apresentam intencionalidades (crença, desejo e intenção). Os autores citam o exemplo de um agente controlador que realiza o planejamento de ações, monitoramento, diagnóstico de falhas e a capacidade de reação em um ambiente geostacionário. Por fim, concluem que a estratégia adotada reduz a complexidade computacional em sistemas que necessitam de um raciocínio dinâmico incluindo as mudanças no ambiente e alterações do sistema.

Lützenberger et al. (2011) apresentam um modelo de agente BDI aplicado como um condutor autônomo em simulações de tráfego urbano. Abordam o problema dizendo que a maioria dos frameworks desenvolvidos para a simulação tráfego urbano de veículos autônomos está condicionada à dinâmica do movimento ou à dinâmica dos fluídos. Enfatizam como solução, o uso do modelo BDI para caracterizar a capacidade de raciocínio do agente principalmente quando outros condutores influenciam nas mudanças ocorridas no tráfego, como por exemplo, um engarrafamento próximo a uma estação de metrô. Para o propósito em questão, os autores definem um modelo para o ambiente que é capaz de influenciar o comportamento do agente condutor e um modelo para o comportamento do agente, capaz de compreender os eventos do ambiente e gerar as ações do agente.

### **3.8 Considerações Finais**

Este capítulo apresentou conceitualmente agentes de software e a arquitetura de estados mentais BDI. Foi possível observar aspectos relevantes sobre o uso destes conceitos em sistemas complexos. Eles são fundamentais, à medida que favorece a construção de sistemas que podem definir novas estratégias para sistemas de tempo real. Um agente contempla capacidades próprias, como: autonomia e racionalidade. Entretanto, não as encontramos em outros paradigmas de desenvolvimento, como sistemas orientados a objetos. Neste aspecto, problemas considerados complexos podem ser resolvidos com agentes. No próximo capítulo será abordada a metodologia utilizada para o desenvolvimento do agente proposto.

## Capítulo IV

# 4 Um Agente Deliberativo Aplicado ao Apoio à Condução de Trens

### 4.1 Introdução

Este trabalho conforme já mencionado, propõe um sistema inteligente para a condução de trem interurbano de cargas. Tal sistema é dotado de um módulo de raciocínio prático que o permite elaborar um plano de condução que satisfaça uma meta complexa. Esta meta consiste em duas grandes etapas: (i) planejar/replanejar as ações que permitem mover um trem de carga de um ponto P a um ponto Q, conjugando restrições naturalmente conflitantes de tempo de viagem, segurança e economia; e (ii) executar tal plano, modificando o ambiente. Entretanto, prover um conjunto de ações que garantam simultaneamente estes três requisitos é uma tarefa de alta complexidade.

As etapas principais são as seguintes:

- definir uma arquitetura de agente software que leve em conta a complexidade do problema de condução de trens interurbanos de carga norteadas pelo modelo mental: crença, desejo e intenção; e
- avaliar o desempenho dos planos de ações do agente condutor vis-à-vis a realização de viagens com configurações específicas em termos do peso de cada trem, número de locomotivas e trechos de uma via férrea.

Para efetivar o planejamento das ações de uma viagem de trem, se define os seguintes objetivos iniciais: (i) um ponto inicial da viagem; e (ii) um ponto de chegada do trem. Para

garantir o movimento do veículo deve-se aplicar uma determinada ação, a saber: acelerar, manter ou reduzir, que implica na troca de um ponto de aceleração que pode variar no intervalo  $\{0...8\}$  ou frear o trem, que implica em utilizar o ponto de aceleração  $\{-1\}$ .

Para decidir qual ponto de aceleração utilizar, obtém-se um conjunto de pontos de aceleração alternativos e se decide pela melhor opção, comparando o impacto gerado por cada um deles na situação atual que o trem se encontra na via férrea. Para escolher o melhor ponto de aceleração entre os disponíveis, utiliza-se um conjunto de critérios pré-estabelecidos definidos por meio de um conjunto de regras especialistas de condução. O ponto de aceleração escolhido nesta fase é inserido em uma estrutura denominada plano. O ciclo se repete até que o plano de viagem esteja completo. Enfim, para executar o plano, verifica-se a aplicabilidade de cada ação mediante um conjunto de restrições de condução, como excesso de velocidade e de força de aceleração. O replanejamento se torna necessário à medida que uma ação viola uma das restrições pré-estabelecidas.

## 4.2 Visão Geral

Pode-se visualizar uma viagem de trem como um conjunto de estados  $\{E_1..E_n\}$  onde cada um configura o estado atual do trem. Cada  $\{E\}$  é determinado pelo resultado do deslocamento do trem na ferrovia que assume um conjunto de variáveis  $\{v\}$  que configuram a sua situação na via férrea, dado por  $E = \{v_1, v_2, v_3, \dots, v_n\}$ . As variáveis  $\{v\}$  são do domínio da condução de trens e configuram as seguintes informações: *posição atual, posição inicial, posição final, velocidade atual, velocidade desejada, ponto de aceleração atual, coeficiente de atrito, direção, número de locomotivas, esforço trator efetivo, esforço trator aderente, resistência dos vagões, resistências das locomotivas, resistência total do trem, deslocamento, consumo, velocidade máxima permitida, velocidade máxima projetada, psi utilizado, tempo e força aceleradora.*

À medida que o trem se desloca na via, o condutor aplica uma ação que altera as características do estado atual do ambiente. O condutor pode acelerar, manter, reduzir o ponto de aceleração da locomotiva ou aplicar freio. Formalmente, uma ação pode ser definida como  $Ac = \{acelerar, manter, reduzir, frear\}$ . O condutor humano decide e aplica uma ação baseada no estado atual do trem, mas ele dificilmente (em termos de complexidade) poderá prever com precisão, as ações que serão válidas no futuro e ainda saber a respeito dos

impactos causados por elas. Uma ação configura o ponto de aceleração ou freio de uma locomotiva, sendo que o ponto de aceleração funciona de forma análoga às marchas de um automóvel, onde o ponto de aceleração pode assumir apenas um dos valores em cada estado  $\{E\}$ . Formalmente, o ponto de aceleração pode ser definido por  $Pa = \{-1,0,1,2,3,4,5,6,7,8\}$ , onde “-1” indica o uso do freio dinâmico.

Um plano de viagem  $\{Pl\}$  é uma alternativa para desencadear uma boa política de ação determinando a validade das ações. Uma alternativa para se criar um plano de viagens é delimitar o espaço de busca de cada ação no estado atual do trem e calcular o impacto gerado por elas. O grande problema é decidir qual ponto de aceleração irá, de forma eficiente, proporcionar uma força de aceleração capaz de movimentar o trem pela via férrea e, ao mesmo tempo, não violar as regras de segurança, como a ultrapassagem da velocidade máxima e a patinação das rodas sobre o trilho. A finalidade, portanto, é proporcionar viagens seguras, rápidas e econômicas. Uma viagem segura é aquela em que a velocidade atual do trem não excede a velocidade máxima da via e o esforço trator efetivo não supera o esforço trator aderente. Uma viagem econômica e rápida é aquela que está dentro dos valores pré-definidos para a viagem. Portanto, o agente deve decidir qual o ponto de aceleração utilizar mediante critérios de decisão para cada estado  $\{E\}$  na via. Este processo consiste em deliberar as ações a serem realizadas e determinam o comportamento inteligente de um agente. A deliberação resulta em uma intenção  $\{I\}$  que é um estado desejado a ser atingido pelo agente. Tais intenções estruturam o plano.

Nesta aplicação, uma intenção é composta por uma ação  $\{Ac\}$  e um ponto de aceleração desejado  $\{Pa_{desejado}\}$ . Formalmente ela pode ser dada por  $I = \{Ac, Pa_{desejado}\}$ . Ao realizar o raciocínio prático, o agente insere no plano de viagem as intenções originadas pelos ciclos de deliberação, um conjunto de pré-condições e pós-condições.

Um plano pode ser definido por ações  $\{Ac\}$ , pré-condições  $\{Pre_c\}$  e pós-condições  $\{Pos_c\}$ . Uma pré-condição é pré-requisito para realizar uma ação no momento de execução do plano. Quando uma intenção é idealizada, o resultado é utilizado para gerar um conjunto de pós-condições que representam os efeitos de uso desta intenção. Um plano é válido quando suas ações atendem as pré-condições de execução. Se um plano é inválido, ou se o ambiente é alterado, suas ações devem ser replanejadas. No entanto, um agente raciocinando, deve ser capaz de desenvolver planos de ação que sejam aplicáveis e atendam os seus objetivos. As correções de planos inválidos demandam recursos e esforços adicionais para um agente.

Entretanto, quando ocorrem mudanças no ambiente, as ações do plano podem deixar de atender as suas pré-condições de execução. Quando isto acontece, fundamentalmente, há necessidade de refazer o plano.

Um plano de viagem no contexto deste trabalho realizar-se-á dentro de um deslocamento  $\{d\}$  previsto do trem na via férrea. Em cada viagem, há um ponto de medida inicial  $\{Pm_{inicial}\}$  e um ponto de medida final  $\{Pm_{final}\}$ , que determinam o início e fim do planejamento da viagem. Finalmente, cada cenário de avaliação foi definido por situações reais de viagens de trens, onde se ajusta um local de partida e um local de chegada do trem.

### 4.3 Arquitetura Proposta

A arquitetura do sistema proposto de um modo geral está dividida sob dois aspectos: (i) virtual que simula o ambiente enquanto o sistema está planejando e (ii) real que representa o ambiente quando o sistema está executando um plano. Em outras palavras, no modo virtual o sistema/agente simula o ciclo de raciocínio-prático e tem como finalidade gerar o plano de viagem. No modo real executa-se o plano da viagem do trem, analisando e aplicando cada ação do plano de viagem gerado.

A Figura 7 ilustra de forma geral o ambiente e da arquitetura do agente proposto. O ambiente é composto por: (i) *Via Férrea*, que corresponde aos dados da via, como velocidade máxima e posição atual; (ii) *Composição*, que representa os dados do trem, como peso, e número de locomotivas e vagões; (iii) *Ambiente Virtual*, que representa as crenças do ambiente virtual do agente, como posição atual, velocidade atual, velocidade desejada, força aceleradora, dentre outras, sendo utilizado para a realização do ciclo de raciocínio prático do agente; (iv) *Ambiente Real*, que representa o ambiente de execução da viagem. A arquitetura interna do agente é dividida nos seguintes módulos: (i) *Controlador*, que realiza o controle do sistema/agente solicitando as seguintes tarefas: configuração do ambiente, deliberação, planejamento e análise do plano; (ii) *Configurador*, que determina qual ambiente deve ser considerado; (iii) *Percepção*, que está dividido em: (a) *Percepção Ambiente Virtual*, que percebe as crenças do ambiente virtual utilizadas no raciocínio prático do agente; e (b) *Percepção Ambiente Real*, que percebe as crenças do ambiente real utilizadas na execução do plano de viagem; (iv) *Raciocínio*, que realiza o raciocínio prático do agente sendo constituído por: (a) *Base de Crenças*, que possui as crenças virtuais e reais do agente; (b) *Deliberação*,

atividade que gera uma nova intenção, é dividida em duas funções, (b.1) *Gerar Opções*, que gera um conjunto de pontos de aceleração aplicáveis e atribui uma utilidade ao ponto de aceleração; (b.2) *Filtro*, que seleciona o ponto de aceleração com maior utilidade; (c) *Intenção*, que representa um estado desejado sendo composto por um ponto de aceleração desejado e uma ação a serem inseridos no plano de viagem; (d) *Atuador*, que aplica as intenções desejadas no modo de execução virtual para obtenção das pós-condições do plano de viagem; e aplica as ações do plano de viagem no ambiente real; (e) *Analizador*, analisa se cada ação do plano é aplicável, verificando as suas pré-condições e pós-condições. (f) *Plano*, representa o plano de viagem gerado após ciclo de raciocínio; e (v) *Calculador*, que realiza os cálculos necessários para a obtenção dos dados sobre resistência e esforços tratores.

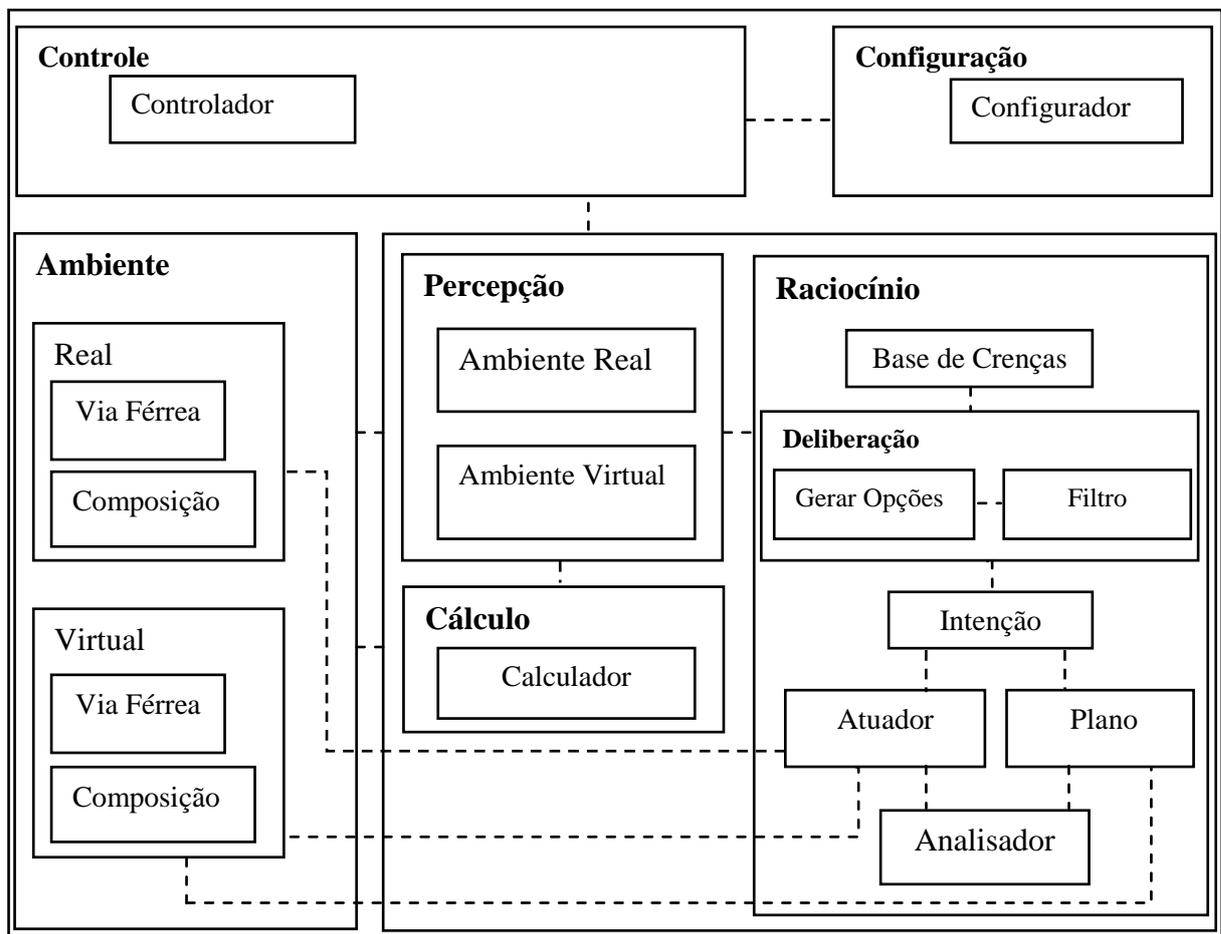


Figura 7: Arquitetura do agente proposto

Formalmente, se define o agente como um 9-tupla  $\{C, P, Bc, D, I, At, An, Pl\}$ , onde  $C$  é o controle de execução,  $P$  é a percepção do ambiente,  $Bc$  é a base de crenças,  $D$  é a

deliberação,  $I$  é a intenção,  $At$  é atuação,  $An$  é analisador e  $Pl$  é o plano do agente. O agente realiza deliberação e atribui uma utilidade para cada ponto de aceleração visando maximizar os efeitos produzidos pela escolha adotada. O ambiente é dividido em ambiente virtual  $\{Ambiente_{virtual}\}$  e ambiente real  $\{Ambiente_{real}\}$ , e podem ser formalmente visualizados como uma tripla  $\{E, f_e, E_{inicial}\}$ , onde  $E$  é o estado atual,  $f_e$  é a função transição de estado e  $E_{inicial}$  é o estado inicial do ambiente. Na execução da viagem, o agente analisa o plano e verifica se as ações são aplicáveis. Caso não sejam, realiza-se o replanejamento.

#### 4.4 Raciocínio Prático – Deliberação e Planejamento

Conforme apresentado no Capítulo 3, o raciocínio prático de um agente BDI é dividido em duas tarefas: *deliberação e raciocínio meio-fim*. Neste trabalho, o processo de deliberação consiste em: (i) determinar um conjunto de pontos de aceleração aplicáveis  $\{Pa_{aplicável}\}$ ; (ii) atribuir uma utilidade para cada um dos pontos de aceleração aplicáveis mediante os efeitos de sua utilização; e (iii) selecionar o ponto de aceleração aplicável que obtiver maior utilidade. Formalmente, o processo de deliberação pode ser definido por  $D = \{G_{opções}, F\}$ .

Para deliberar e determinar quais pontos de aceleração podem ser aplicados, o agente utiliza uma função denominada gerar opções  $\{G_{opções}\}$ , considerando o estado atual do ambiente. Este processo é realizado por um teste de cada  $\{Pa_{aplicável}\}$  para verificar o impacto provocado no ambiente. Para efetivar tal função no sistema, realiza-se comunicação com o Calculador, e em seguida, verifica-se os efeitos  $\{Ef\}$  de cada um dos pontos de aceleração aplicáveis. Os efeitos de uso cada ponto de aceleração servirá para que cada  $\{Pa_{aplicável}\}$  receba uma pontuação, denominada de utilidade  $\{U\}$ . A utilidade estabelece um critério de escolha do ponto de aceleração a ser utilizado pelo agente visando à maximização da ação no estado  $\{E\}$  do ambiente. Após atribuir a utilidade para cada ponto de aceleração aplicável, o agente utiliza uma função denominada filtro  $\{F\}$ , que seleciona o  $\{Pa_{aplicável}\}$  de maior utilidade, denominado, conseqüentemente, de ponto de aceleração desejado  $\{Pa_{desejado}\}$ . Em seguida, o agente configura a ação  $\{Ac\}$  a ser realizada, onde juntos dão origem a intenção  $\{I\}$  do agente  $I = \{Ac, Pa_{desejado}\}$ .

Após ser definida, a intenção é executada pelo agente em seu ambiente virtual  $\{Ambiente_{virtual}\}$ , assegurando a inserção das pós-condições  $\{Pos_c\}$  no plano por meio da

realização dos cálculos de resistência e de esforços tratores. Formalmente, as pós-condições de uma intenção podem ser definidas por  $Pos_c = \{V\}$ , onde  $\{V\}$  representa as variáveis no domínio de condução, dadas por  $V = \{Pa_{desejado}, posição\ atual, força\ aceleradora, velocidade\ atual, velocidade\ desejada, velocidade\ máxima, velocidade\ máxima\ projetada, coeficiente\ de\ atrito, esforço\ trator\ aderente, esforço\ trator\ efetivo, resistência\ total, resistência\ locomotivas, resistência\ vagões, consumo, consumo\ total, tempo, tempo\ total, distância\ percorrida, LTKB\}$ .

A cada ciclo de raciocínio do agente insere-se no subplano  $\{P\}$  uma ação, as pré-condições e um conjunto de pós-condições, formalmente definido por  $P = \{Ac, Pre_C, Pos_C\}$ . As ações possíveis que o agente pode realizar e inserir no plano são: *acelerar, reduzir, manter ou frear*. As pré-condições restringem e determinam se uma ação  $\{Ac\}$  do subplano  $\{P\}$  será aplicável. No domínio desta aplicação consideraram-se três pré-condições para execução de uma ação: (i) velocidade atual do trem deve ser menor ou igual à velocidade máxima da via; (ii) esforço trator efetivo deve ser menor ou igual ao esforço trator aderente; e (iii) coeficiente de atrito deve ser igual a 0,22, valor que configura o trilho seco e limpo. Uma pré-condição pode ser visualizada como um conjunto  $Pre_C = \{vp\}$ . Uma  $\{vp\}$  configura as pré-condições definidas e pode ser dada por  $vp = \{Pa_{atuab}, velocidade\ atual \leq velocidade\ máxima, esforço\ trator\ efetivo \leq esforço\ trator\ aderente, coeficiente\ de\ atrito = 0,22\}$ . Um plano de viagem completo  $\{Pl\}$  é composto por um conjunto de suplanos  $\{P_i\}$  e pode ser dado por  $Pl = \{P_1 \rightarrow P_2 \rightarrow P_3 \rightarrow \dots P_n\}$ .

Relembramos que neste trabalho, o objetivo principal do agente é, em um ambiente dinâmico, elaborar e executar planos de condução para um trem em um trecho de uma via férrea. Para isto, foi necessário dividir o ambiente de atuação do agente em duas estruturas de dados: virtual e real. O agente elabora o plano de viagem na estrutura virtual e aplica o plano na estrutura real. O ciclo de raciocínio prático inicia com a configuração do ambiente que é ajustado para virtual. Este ciclo é realizado após definição de uma distância a ser vencida, que determina o deslocamento necessário para o início e fim do planejamento da viagem virtual, definido de acordo com os seguintes passos:

1. Ajuste o ambiente para *virtual*.
2. Carregue as crenças iniciais: ponto de aceleração atual, velocidade atual, posição atual, peso do trem, distância planejada, número de locomotivas, posição inicial e posição final do trem na via férrea.

3. Realize a percepção inicial do ambiente.
4. Realize os cálculos de resistência e esforços tratores a partir dos dados percebidos.
5. Atualize a base de crenças: posição do trem, tempo de deslocamento, deslocamento previsto, velocidade máxima, velocidade atual, velocidade projetada, velocidade desejada, pressão de freios, o LTKB, o consumo, distância percorrida no planejamento.
6. Inicie o processo de deliberação:
  - a. Consulte crenças virtuais.
  - b. Gere as opções de pontos de aceleração aplicáveis.
  - c. Teste os pontos de aceleração aplicáveis, por meio da realização de cálculos e verifique os efeitos destes pontos de aceleração no estado atual do trem no ambiente. Os dados do novo estado assumem novos valores para o consumo, velocidade atual, esforço trator efetivo e deslocamento.
  - d. Atualize as informações na base de crenças.
  - e. Consulte crenças virtuais.
  - f. Atribua uma utilidade para os pontos de aceleração testados mediante os critérios estabelecidos.
  - g. Filtre o ponto de aceleração aplicável de maior utilidade.
  - h. Defina a intenção do agente, que é composta pelo ponto de aceleração de maior utilidade (ponto desejado), e pela ação que é determinada por um conjunto de regras classificadoras.
  - i. Insira uma intenção definida no plano (ponto de aceleração desejado e ação).
  - j. Aplique a intenção no ambiente virtual e obtenha as pós-condições do plano.
  - k. Finalize a configuração do plano inserindo as pré-condições de execução e as pós-condições observadas no ambiente virtual.
7. Finalize o raciocínio prático.

A Figura 8 ilustra de forma resumida a estrutura de um plano de viagem.

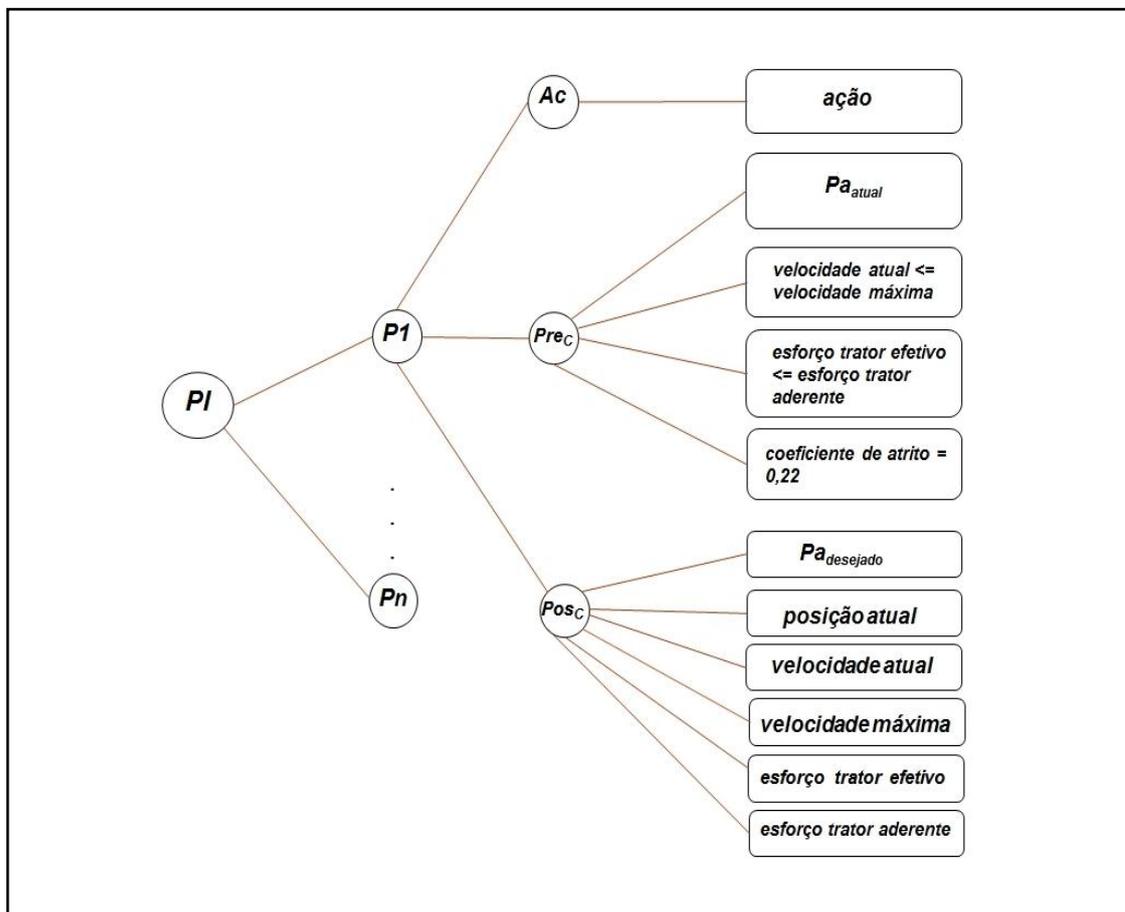


Figura 8: Estrutura de um plano

#### 4.4.1 Deliberação - Gerar Opções

Inicialmente, por meio da função de gerar opções, o agente estabelece os pontos de aceleração que podem ser aplicados. Depois, realiza-se o teste dos prováveis pontos de aceleração. Para isto, mantém comunicação com o calculador para gerar os efeitos (crenças) de cada ponto de aceleração aplicável no estado atual do trem na via férrea. Atribui-se uma utilidade para cada um deles a fim de encontrar a alternativa (ação) mais satisfatória a ser inserida no plano. As regras a seguir definem os conhecimentos para se gerar um conjunto de pontos de aceleração aplicáveis que está condicionada ao ponto de aceleração atual. Este conjunto de regras foi encapsulado em um módulo definido pela interface *go*, que recebe como parâmetro o  $\{Pa_{atual}\}$ , devolvendo um conjunto de pontos de aceleração aplicáveis. Determinam-se quais pontos de aceleração são aplicáveis de acordo com o Quadro 1:

```

função go(i: inteiro) : r;
{i: ponto de aceleração atual}
Início
  r : p;      {p=<pa>, pa é o ponto de aceleração aplicável}
  Ler(i):
    caso 0:  testar(p):
              p = 0;
              p = 1;
              p = 2;
              p = 3;
              p = 4;
              p = 5;
              p = 6;
              p = 7;
              p = 8;
            fim.
    caso 1:  testar(p):
              p = 0;
              p = 1;
              p = 2;
              p = 3;
            fim.
    caso 2:  testar(p):
              p = 0;
              p = 1;
              p = 2;
              p = 3;
            fim.
    caso 3:  testar(p):
              p = 0;
              p = 2;
              p = 3;
              p = 4;
            fim.
    caso 4:  testar(p):
              p = 0;
              p = 3;
              p = 4;
              p = 5;
            fim.
    caso 5:  testar(p):
              p = 0;
              p = 4;
              p = 5;
              p = 6;
            fim.
    caso 6:  testar(p):
              p = 0;
              p = 5;
              p = 6;
              p = 7;
            fim.
    caso 7:  testar(p):
              p = 0;
              p = 6;
              p = 7;
              p = 8;
            fim.
    caso 8:  testar(p):
              p = 0;
              p = 6;
              p = 7;
              p = 8;
            fim.
    caso -1: testar(p):
              p = 0;
              p = 1;
              p = 2;
              p = 3;
              p = 4;
              p = 5;
              p = 6;
              p = 7;
              p = 8;
            fim.
          fim.
        retorne p;
      fim

```

Quadro 1: Função de seleção para gerar opções

É importante ressaltar que “-1” configura a intenção “frear”. Esta é indicada quando se deseja reduzir a velocidade atual do veículo para que ela não exceda a velocidade limite da via. Portanto, caso necessite reduzir a velocidade do veículo, o agente deverá inserir no plano o ponto desejado “-1”, onde não será necessário atribuir uma utilidade para realizar esta escolha.

#### 4.4.2 Utilidade

Conforme já mencionado, para escolher um ponto de aceleração o agente atribui aos pontos de aceleração aplicáveis uma utilidade  $\{U\}$ . Um atributo  $\{a\}$  representa uma variável no domínio de condução de trens e é usada para definir a utilidade do ponto de aceleração aplicável. Para os experimentos assumiram-se os seguintes atributos para definir  $\{U\}$ , a saber:  $\{\text{força aceleradora, esforço trator efetivo, consumo, velocidade e deslocamento}\}$ .

Os critérios representam um conjunto de estados referentes ao impacto das pós-condições de uso de um ponto de aceleração aplicável referente ao atributo  $\{a\}$ . Cada critério está relacionado a um intervalo  $\{I_{\text{valores}}\}$  definido empiricamente por meio de estudos de viagens passadas. Por fim, encontra-se a utilidade  $\{U\}$  realizando a soma das utilidades dos atributos  $\{u(a)\}$ . As Tabelas 2, 3, 4, 5 e 6 mostram a definição dos critérios  $\{C\}$ , dos intervalos de valores  $\{I_{\text{valores}}\}$  e das utilidades de cada atributo  $\{u(a)\}$ .

Tabela 2: Definição dos valores para o atributo *consumo*

<b>Crítérios</b>	<b>Intervalo de Valores</b>	<b>Utilidade</b>
Muito alto	{consumo $\geq$ 10}	1
Alto	{consumo $\geq$ 5 e consumo $<$ 10}	2
Médio	{consumo $\geq$ 2,5 e consumo $<$ 5}	3
Baixo	{consumo $\geq$ 1 e consumo $<$ 2,5}	4
Muito baixo	{consumo $\geq$ 0 e consumo $<$ 1}	5

Tabela 3: Definição dos valores para o atributo *velocidade*

<b>Crítérios</b>	<b>Intervalo de Valores</b>	<b>Utilidade</b>
Ótima	{velocidade atual $\leq$ velocidade máxima - 6 e velocidade atual $\geq$ velocidade máxima - 10}	5
Boa	{velocidade atual $\leq$ velocidade máxima - 11 e velocidade atual $\geq$ velocidade máxima - 20}	3
Ruim	{velocidade atual $<$ velocidade máxima - 20}	1

Tabela 4: Definição dos valores para o atributo *força aceleradora*

<b>Crítérios</b>	<b>Intervalo de Valores</b>	<b>Utilidade</b>
Suficiente	força aceleradora > 0	1
Insuficiente	força aceleradora <= 0	-5

Tabela 5: Definição dos valores para o atributo *esforço trator efetivo*

<b>Crítérios</b>	<b>Intervalo de Valores</b>	<b>Utilidade</b>
Bom	esforço trator efetivo < esforço trator aderente	1
Ruim	esforço trator efetivo >= esforço trator aderente	-5

Tabela 6: Definição dos valores para o atributo *deslocamento*

<b>Crítérios</b>	<b>Intervalo de Valores</b>	<b>Utilidade</b>
Satisfatório	deslocamento < 500 m	0
Ruim	deslocamento >= 500 m e deslocamento < 1000 m	-5
Muito Ruim	deslocamento >= 1000 m	-10

A utilidade do ponto de aceleração aplicável pode ser dada por meio da Equação 19.

$$U = \sum_{a=1}^n u(a) \quad (19)$$

Onde:  $\sum$  = somatório da utilidade de cada atributo

$u(a)$  = utilidade de atributo

$U$  = utilidade do ponto de aceleração aplicável

O objetivo principal de se atribuir uma utilidade a cada critério é estabelecer um busca do melhor ponto de aceleração aplicável para o estado atual do trem na via em termos de segurança, economia e rapidez.

#### 4.4.3 Filtro e Intenção

A função filtro  $\{F\}$  seleciona o  $\{Pa_{aplicável}\}$  que obtiver maior utilidade  $\{U\}$  estabelecendo o ponto de aceleração desejado  $\{Pa_{desejado}\}$ . Posteriormente, o  $\{Pa_{desejado}\}$  será utilizado para definir uma intenção  $\{I\}$ . A Figura 9 ilustra as atividades da deliberação do agente.

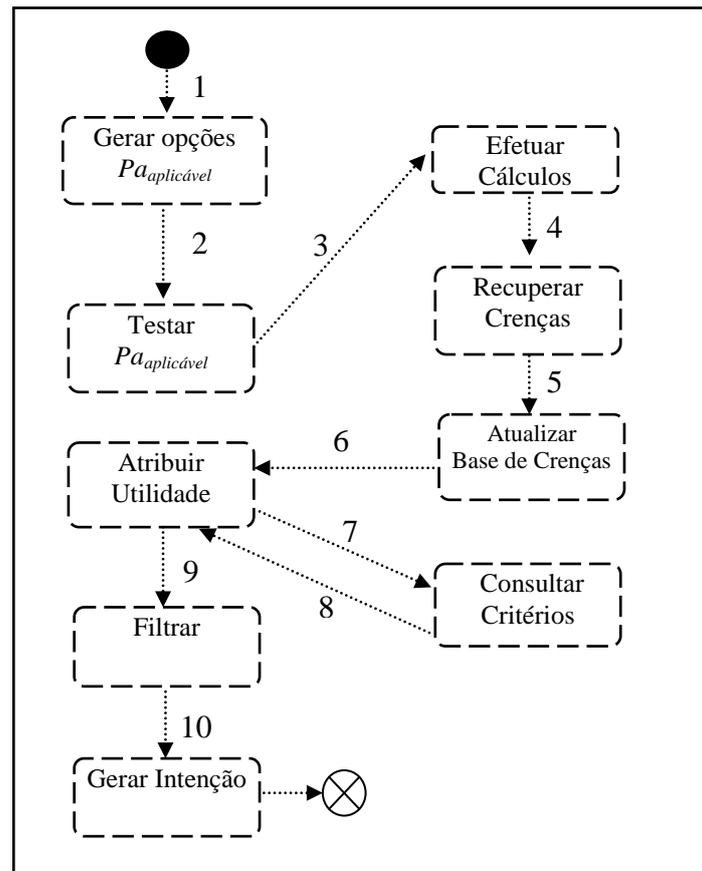


Figura 9: Deliberação

Após deliberar, o agente utiliza o ponto de aceleração atual  $\{Pa_{atual}\}$  e o ponto de aceleração desejado  $\{Pa_{desejado}\}$  para configurar uma intenção a ser inserida no plano de viagem. Uma intenção é um estado desejado composto por um ponto de aceleração e uma ação. Para determinar a intenção, o agente define o ponto de aceleração atual  $\{Pa_{atual}\}$  como um estado inicial  $\{e_{inicial}\}$  e configura o ponto de aceleração desejado  $\{Pa_{desejado}\}$  como um estado desejado  $\{e_{desejado}\}$ . Cada ação a ser inserida no plano de viagem deve passar por este ciclo, gerando assim a intenção. Tal intenção deverá ser realizada no ambiente virtual, assegurando as pós-condições do plano. As regras a seguir definem os conhecimentos para se gerar uma intenção que está condicionada aos pontos de aceleração: atual e desejado. Este conjunto de regras foi encapsulado em um módulo definido pela interface  $gi$ , que recebe como parâmetro o  $\{Pa_{atual}\}$  e  $\{Pa_{desejado}\}$ , devolve uma intenção representada por um par  $\langle ação, Pa_{atual} \rangle$ . As possíveis ações são: acelerar, manter, reduzir ou frear.

```

função gi(i, j : inteiro) : par
{i: ponto de aceleração atual, j: ponto de aceleração desejado}
Início
  par : p; {p=<ac, pa>, onde ac é a intenção e pa é o ponto de
           aceleração}
  escolha(i):
    caso 0: escolha(j):
      caso -1: p = <"frear", -1>;
      caso 0: p = <"manter", 0>;
      caso contrário: p = <"acelerar", j>;
      fim.
    caso -1: escolha(j):
      caso -1: p = <"frear", -1>;
      caso 0: p = <"acelerar", 0>;
      caso contrário: p = <"acelerar", j>;
      fim.
    caso 1: escolha(j):
      caso -1: p = <"frear", -1>;
      caso 0: p = <"reduzir", 0>;
      caso 1: p = <"manter", 1>;
      caso contrário: p = <"acelerar", j>;
      fim.
    caso 2: escolha(j):
      caso -1: p = <"frear", -1>;
      caso 0: p = <"reduzir", 0>;
      caso 1: p = <"reduzir", 1>;
      caso 2: p = <"manter", 2>;
      caso 3: p = <"acelerar", 3>;
      fim.
    caso 3: escolha(j):
      caso -1: p = <"frear", -1>;
      caso 0: p = <"reduzir", 0>;
      caso 2: p = <"reduzir", 2>;
      caso 3: p = <"manter", 3>;
      caso 4: p = <"acelerar", 4>;
      fim.
    caso 4: escolha(j):
      caso -1: p = <"frear", -1>;
      caso 0: p = <"reduzir", 0>;
      caso 3: p = <"reduzir", 3>;
      caso 4: p = <"manter", 4>;
      caso 5: p = <"acelerar", 5>;
      fim.
    caso 5: escolha(j):
      caso -1: p = <"frear", -1>;
      caso 0: p = <"reduzir", 0>;
      caso 4: p = <"reduzir", 4>;
      caso 5: p = <"manter", 5>;
      caso 6: p = <"acelerar", 6>;
      fim.
    caso 6: escolha(j):
      caso -1: p = <"frear", -1>;
      caso 0: p = <"reduzir", 0>;
      caso 5: p = <"reduzir", 5>;
      caso 6: p = <"manter", 6>;
      caso 7: p = <"acelerar", 7>;
      fim.
    caso 7: escolha(j):
      caso -1: p = <"frear", -1>;
      caso 0: p = <"reduzir", 0>;
      caso 6: p = <"reduzir", 6>;
      caso 7: p = <"manter", 7>;
      caso 8: p = <"acelerar", 8>;
      fim.
    caso 8: escolha(j):
      caso -1: p = <"frear", -1>;
      caso 0: p = <"reduzir", 0>;
      caso 6: p = <"reduzir", 6>;
      caso 7: p = <"reduzir", 7>;
      caso 8: p = <"manter", 8>;
      fim.
  fim.

```

Quadro 2: Função de seleção para gerar intenção

## 4.5 Execução do Plano

Um plano de ação deve satisfazer as suas pré-condições para ser executado. No ambiente de condução, muitas restrições podem impossibilitar que uma ação seja aplicada, dentre elas a velocidade atual do trem, não poderá exceder a velocidade máxima da via, o esforço trator efetivo não poderá ultrapassar o esforço trator aderente, visto que o mesmo provoca a patinagem das rodas sobre os trilhos. Mudanças quanto ao clima podem tornar uma ação impraticável para a situação atual do trem na via. Portanto, se faz necessário uma análise criteriosa para se aplicar um plano, visto que as ações inválidas podem comprometer e inviabilizar a continuidade da viagem.

Dá-se início a execução do plano logo em seguida a conclusão do ciclo de raciocínio prático do agente. As principais atividades desta fase contemplam as atividades de análise, execução do plano e replanejamento (se necessário). O sistema (agente) inicia a execução do plano de viagem com a configuração do ambiente ajustada para real. A sequência completa de execução de um plano pode ser descrita por meio dos seguintes passos:

1. Inicie a execução do plano de viagem
2. Ajuste o ambiente para *real*.
3. Leia o plano de viagem gerado.
4. Enquanto o plano possuir uma ação, faça:
  - a. Analise as pré-condições e pós-condições de execução de uma ação do plano.
  - b. Se ação é aplicável então faça:
    - b.1. Aplique ação no ambiente real.
    - b.2. Atualize ambiente real os seguintes dados: posição do trem, tempo de deslocamento, deslocamento previsto, velocidade máxima, velocidade atual, velocidade projetada, velocidade desejada, pressão de freios, LTKB, consumo, distância percorrida.
    - b.3. Atualize as informações na base de crenças.

Senão salve estado anterior à última ação válida realizada do plano, ajuste o ambiente para *virtual* e solicite o replanejamento de ações a partir deste estado.
5. Fim enquanto;
6. Finalize execução do plano.

A Figura 10 ilustra a execução do plano de viagem do agente.

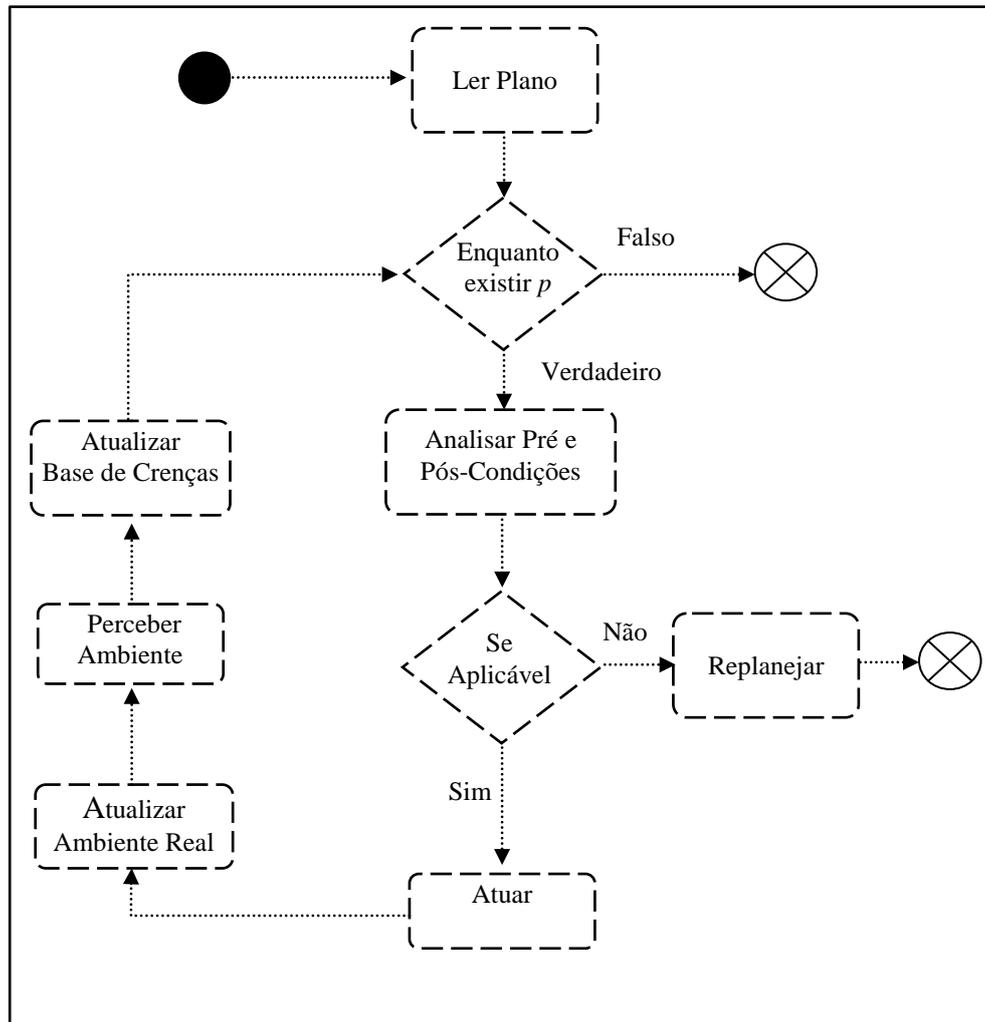


Figura 10: Execução e análise de um plano de viagem

## 4.6 Replanejamento

O replanejamento envolve o processo de gerar um novo plano para corrigir uma falha ou ainda gerar um novo plano para solucionar mudanças no ambiente de atuação do agente. O resultado da execução de um plano em um ambiente de tempo real pode causar crenças divergentes e mudanças nas metas do agente.

Não obstante, um plano é definido por um conjunto de ações, pré-condições e pós-condições. Ao realizar a leitura do plano, o agente verifica se a ação desejada é aplicável. Para descobrir se esta é aplicável, o agente verifica se uma das pós-condições desta ação viola alguma das restrições estabelecidas nas pré-condições, por exemplo, se a velocidade a ser

desenvolvida pelo trem será maior que a velocidade máxima da via. Após realizar esta verificação, se uma ação desejada é válida, então esta será aplicada. Caso contrário, um novo plano será gerado. Na proposta deste trabalho, o replanejamento poderá acontecer sob duas condições: (i) para realizar a correção do plano, se uma ação do plano gerado viola uma das pré-condições estabelecidas; e (ii) quando se realiza uma alteração no ambiente. Em ambos os casos, no modo de execução do plano, o agente ao perceber que uma ação desejada viola uma das pré-condições estabelecidas, ele grava o estado da ação anterior  $\{E_{anterior}\}$ , elimina o plano atual  $\{PI\}$  e solicita o replanejamento das ações a partir do estado anterior mencionado (última ação válida). Isto faz com que o estado da última ação realizada seja o estado inicial  $\{E_{inicial}\}$  do novo plano que se inicia após a solicitação do replanejamento e início de um novo ciclo de raciocínio prático.

#### **4.7 Considerações Finais**

Este capítulo apresentou a arquitetura proposta para o desenvolvimento do módulo inteligente de condução de trens. De fato, planejar ações de uma viagem de trens e alcançar simultaneamente uma viagem rápida, segura e econômica é um desafio para condutores de trens. Entretanto, agentes que raciocinam sobre as ações e elaboram planos podem ser uma alternativa viável para auxiliar na resolução de problemas como este. Para realizar nossa proposta, a estratégia utilizada segmentou a forma de atuação do agente condutor: (i) geração de plano de ações interagindo com um ambiente de execução virtual; e (ii) execução do plano interagindo com um ambiente de execução real. Esta estratégia possibilitou a implementação do simulador de viagens, onde o agente planeja/replaneja ações e executa o plano de ação gerado. No próximo capítulo será apresentado o simulador de viagens.

# Capítulo V

## 5 Simulador

### 5.1 Introdução

Para avaliar a metodologia proposta foi desenvolvido um ambiente de simulação de viagens. Para realizar tal simulador, o modelo, foi estrategicamente, dividido em duas estruturas: (i) execução virtual e (ii) execução real. A execução virtual realiza raciocínio prático e tem como finalidade, por meio de critérios de decisão, obter um plano de viagem. A execução real aplica a simulação da viagem real do trem, onde cada ação do plano é analisada pelo agente e executada, caso seja considerada aplicável. Quando uma ação do plano não atende uma pré-condição de execução, o plano é corrigido/refeito, caracterizando o replanejamento.

### 5.2 Descrição Geral

O sistema/agente conforme já mencionado, realiza duas atividades principais, (i) planejamento/replanejamento de ações; e (ii) análise e execução do plano. É importante destacar que os critérios para a inserção de uma ação no plano são diferentes dos critérios para a sua execução no ambiente real. Os critérios para gerar as ações do plano de viagem são considerados mais sofisticados, à medida que, o agente realiza a deliberação das intenções, utilizando-se de critérios pré-estabelecidos que determinem uma utilidade. Em contrapartida, do ponto de vista de complexidade, o agente utiliza critérios um pouco mais simples (e às vezes mais restritivos) para executar o plano, como análise das pré-condições e pós-condições, e solicitação de replanejamento. Não obstante, para realizar o replanejamento, o

agente identifica a posição atual do trem na qual a ação é inválida e elabora um novo plano a partir da posição anterior a esta. Desta forma, o plano é refeito a partir da última ação válida realizada. O agente registra o estado atual das variáveis envolvidas e solicita um novo ciclo de raciocínio prático para gerar o novo plano. O simulador é composto pelos seguintes módulos:

- CConfigurador: Tem a função de configurar o estado dos ambientes que pode ser: virtual ou real. No ambiente virtual, realiza-se o planejamento de um ponto inicial até o ponto final da viagem. No ambiente real realiza-se execução da viagem.
- PinAmbienteVirtual: Representa o ambiente virtual do sistema. Ele possui dados (crenças) como ponto de aceleração atual, esforços tratores e resistentes. Ainda há dados como a posição do trem na via, velocidade máxima, a distância percorrida, o deslocamento, a velocidade atual, o consumo, consumo total e LTKB. Ele auxilia na elaboração do plano de viagem, mais precisamente na configuração das pós-condições do plano.
- SPPercebeAmbienteVirtual: Representa a percepção do ambiente virtual. Percebe os dados (crenças) do ambiente virtual e estabelece comunicação com o calculador enviando-os à base de crenças.
- SPPercebeAmbienteReal: Representa a percepção do ambiente real. Percebe os dados do ambiente real e os envia à base de crenças.
- PinAmbienteReal: Representa o ambiente real do sistema. Possui dados resultantes (crenças) da execução do plano de viagem, como o ponto de aceleração atual, a posição atual do trem, os esforços tratores e resistentes, as velocidades, o LTKB, o consumo, a distancia percorrida e demais dados.
- BCCrenças: Representa o registro das informações (crenças) do componente de percepção. Possui crenças do ambiente virtual e do ambiente real.
- RRaciocinador: Realiza a deliberação e gera as intenções de cada plano.
- RAnalisador: Analisa o plano de viagem, avaliando as pré-condições e as pós-condições de cada ação do plano.
- Atuador: Executa uma ação planejada que pode ser acelerar, manter ou reduzir um ponto de aceleração; ou frear.
- Plano: estrutura que contém ações, pré-condições e pós-condições.



Enquanto o ambiente é virtual e a distância a ser vencida for maior que a distância percorrida, faça:

- a. Calcule os dados de resistência e de esforços tratores.
- b. Atualize as crenças virtuais na base de crenças.
- c. Inicie o processo de deliberação:
  - c.1. Consulte crenças virtuais.
  - c.2. Gere as opções de pontos de aceleração aplicáveis.
  - c.3. Teste o ponto de aceleração aplicável, por meio da realização de cálculos e verifique os efeitos deste ponto de aceleração no estado atual do trem no ambiente. Os dados do novo estado assumem novos valores para o consumo, velocidade atual, esforço tratores e deslocamento.
  - c.3. Atualize as informações na base de crenças.
  - c.4. Consulte crenças virtuais.
  - c.5. Atribua uma utilidade para cada ponto de aceleração testado mediante os critérios estabelecidos.
  - c.5. Filtre o ponto de aceleração de maior utilidade.
  - c.6. Defina uma intenção.
  - c.7. Insira a intenção (ponto de aceleração desejado e ação) no plano.
  - c.8. Aplique a intenção no ambiente virtual e obtenha as pós-condições do plano.
  - c.9. Finalize a configuração do plano inserindo as pré-condições de execução e as pós-condições observadas no ambiente virtual.
5. Finalize execução do raciocínio prático.
6. Inicie a execução do plano de viagem.
7. Ajuste o ambiente para *real*.
8. Leia o plano de viagem gerado.
9. Enquanto o plano possuir uma ação, faça:
  - a. Analise as pré-condições e pós-condições de execução de uma ação do plano.
  - b. Se ação é aplicável então:
    - b.1. Aplique ação no ambiente real.
    - b.2. Atualize ambiente real.
    - b.3. Atualize as informações na base de crenças.

Senão salve estado anterior à última ação válida realizada do plano, ajuste o ambiente para *virtual* e solicite o replanejamento de ações a partir deste estado.

11. Finalize execução do plano.
12. Finalize execução do sistema.

O cenário geral da simulação envolve um trem, um trecho de via férrea a ser percorrido e um condutor do veículo. O trem é composto por um conjunto de vagões e por uma ou mais locomotivas. Cada locomotiva possui um identificador. Cada vagão possui um peso. Os equipamentos do trem possibilitam que os seguintes dados sejam fornecidos: *força de aceleração* que indica a potência desenvolvida por um ponto de aceleração utilizado, *velocidade*, *pressão de freios*, *posição atual e distância percorrida*.

### 5.3 Via Férrea

Em termos gerais, as informações referentes à composição férrea juntamente as informações do perfil da via férrea permitem realizar os cálculos da dinâmica da condução de trens. A viagem ocorre em um trecho real de via férrea sendo representada por um conjunto de dados que definem um ponto de medida. Cada ponto de medida possui aproximadamente 20 metros de comprimento e contém uma *tupla* contendo o identificador do quilômetro bem como o percentual de rampa, o raio da curva e a velocidade máxima permitida. É por meio destas informações que o Calculador é capaz de calcular as resistências do veículo a partir de sua localização na via férrea. A Figura 12 ilustra conceitualmente uma via férrea.

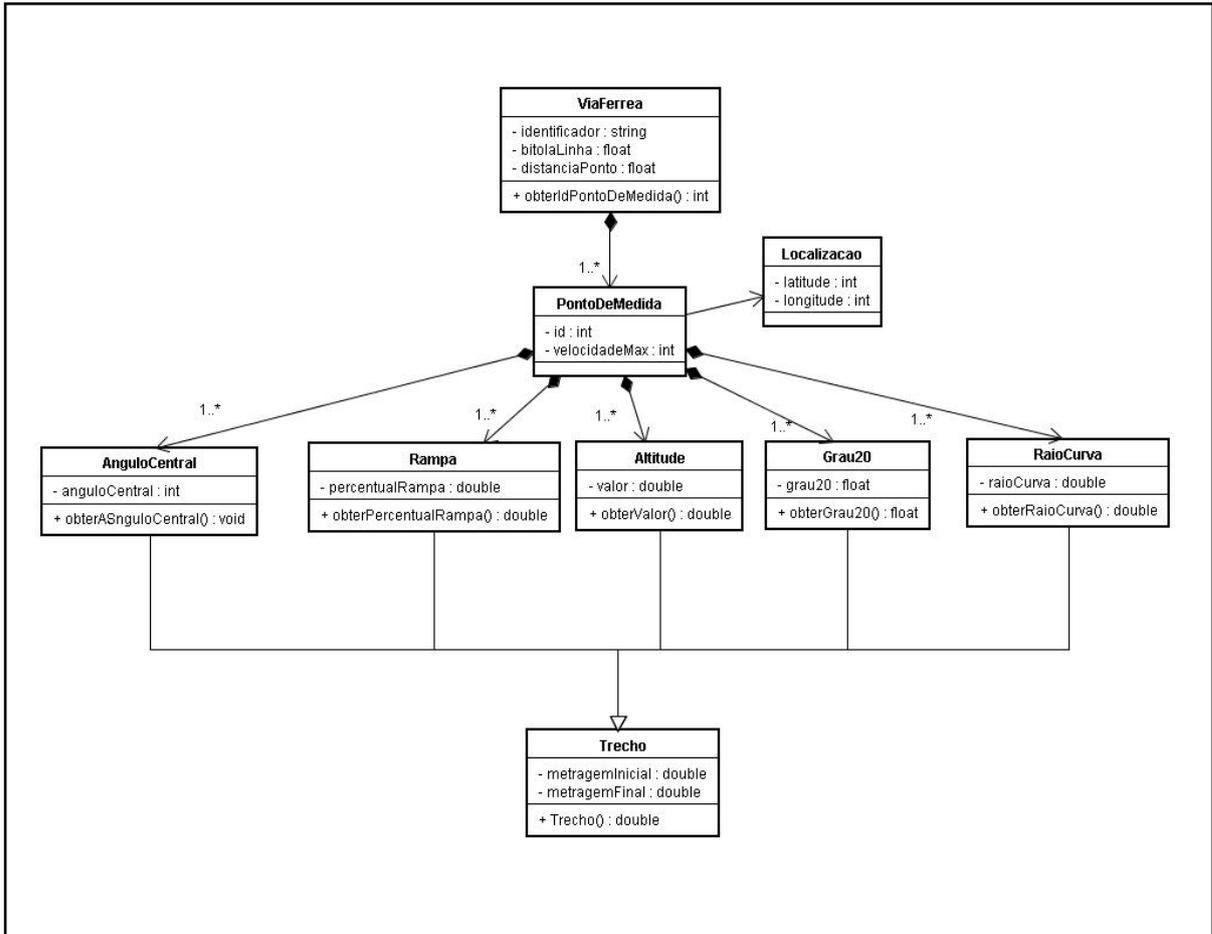


Figura 12: Representação conceitual de uma via férrea

## 5.4 Calculador

É importante mencionar que o elemento que compõe o módulo de cálculo, foi desenvolvido em trabalhos anteriores no contexto do Projeto PAI-L (*Piloto Automático Inteligente de Locomotivas*) do Grupo de Pesquisa em Agentes de Software da PUC-PR.

Este módulo fornece os cálculos necessários para que o agente possa realizar sua deliberação e ainda gerar e analisar as pós-condições do plano de viagem. O agente, sempre que necessário, interage com o módulo de cálculo que utiliza as informações do perfil da via férrea e da composição do veículo, por meio de arquivos estruturados em formato XML para a realização dos cálculos. Especificamente, as informações requeridas pelo calculador são: *posição atual, posição inicial, posição final, velocidade atual, velocidade desejada, ponto de aceleração atual, coeficiente de atrito, direção e número de locomotivas.*

Após efetuar os cálculos, ele retorna como resultado ao agente/sistema as seguintes informações: *esforço trator efetivo, esforço trator aderente, resistência dos vagões, resistências das locomotivas, deslocamento, consumo, velocidade máxima permitida, velocidade máxima projetada, psi utilizado, velocidade desejada, tempo e força aceleradora*. O Calculador atua como um servidor, fornecendo toda a infraestrutura de cálculos estabelecido por meio da comunicação com o agente via *socket*. Ele foi desenvolvido utilizando a linguagem de programação C++, enquanto que o módulo de condução, agente, foi desenvolvido utilizando a linguagem de programação Java. A Figura 13 ilustra a troca de informações entre o agente/sistema e a infraestrutura de cálculos.

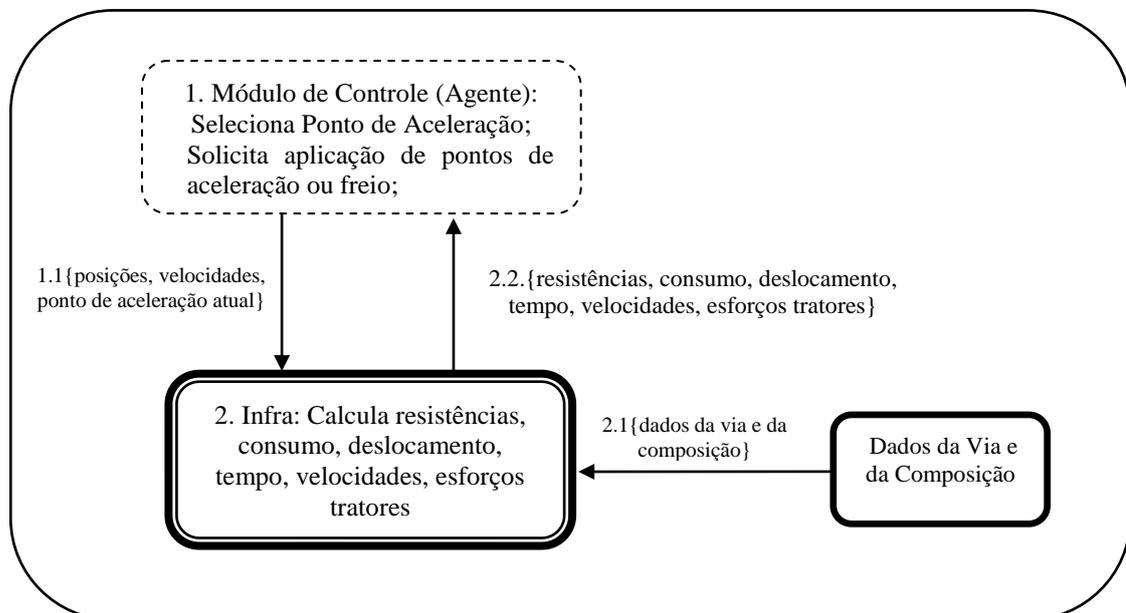


Figura 13: Comunicação com o Calculador

## 5.5 Exemplo da Simulação

Para explicar as atividades de raciocínio prático serão apresentados os estados das variáveis envolvidas durante este processo em três ações planejadas iniciais e na última ação planejada pelo agente. O exemplo foi obtido por meio da simulação de uma viagem real onde os resultados em detalhes serão apresentados no Capítulo 6 – Resultados Obtidos.

A distância do planejamento/execução do plano nesta simulação corresponde a 72,18 quilômetros, onde a localização inicial do trem na ferrovia é indicada pelo ponto de medida

“6.831” que indica a posição do trem no início do planejamento e início da execução plano, e a localização final, estabelecida pelo ponto de medida “3.222” que indica a posição final do trem na ferrovia ao término do planejamento e execução do plano. O peso total do veículo é de 3.429 toneladas sendo composto por duas locomotivas e trinta e um vagões.

**Configuração inicial:**

- ponto de aceleração atual = 0;
- velocidade atual = 0;
- velocidade desejada = 2;
- posição atual = 6.831;
- peso do trem = 3.429;
- distância planejada = 72.118 m;
- número de locomotivas = 2;
- posição inicial = 6.831;
- posição final = 3.222;
- ambiente = virtual;

**Primeira interação do raciocínio prático:**

A realização das atividades do ciclo completo do raciocínio prático é realizada enquanto o ambiente estiver no modo de execução virtual e a distância a ser vencida (planejada) for maior que a distância percorrida (realizada no planejamento e execução do plano). Calculam-se os dados de resistência e de esforços tratores para o ponto de aceleração definido inicialmente como zero. Portanto, cada ação planejada é resultante de apenas um ciclo de raciocínio prático para o planejamento de cada ação do plano. Os resultados dos cálculos iniciais estão configurados da seguinte forma:

- esforço trator efetivo = 0;
- esforço trator aderente = 73.928,8;
- força aceleradora = -3848,78;
- resistência total = 4.013,4521;
- resistência locomotivas = 329,332;
- resistência dos vagões = 3.684,12;
- pressão de freios = 90,0;

- deslocamento = 0;
- distancia percorrida planejamento = 0;
- tempo gasto = 0;
- consumo = 0;
- coeficiente de atrito = 0,22;
- tempo total = 0;
- consumo total = 0;
- LTKB = 0.

Ao dar continuidade ao ciclo de raciocínio o agente atualiza as crenças virtuais na base de crenças. Em seguida, dá-se início o processo de deliberação, com a consulta às informações na base de crenças, e pela realização da função de gerar as opções de pontos de aceleração aplicáveis (alternativos), considerando o ponto de aceleração atual.

- Ponto de aceleração atual = 0.

Conseqüentemente, determinam-se por meio de um conjunto de regras classificadoras, os possíveis pontos de aceleração alternativos:

- 0, 1, 2, 3, 4, 5, 6, 7, 8;

A seguir realiza-se o teste de cada ponto de aceleração aplicável, por meio de trocas de dados com a infraestrutura de cálculo e atualização das crenças. Logo, o agente verifica os efeitos ocasionados pelo ponto de aceleração no estado atual do trem no ambiente. Os dados do novo estado assumem novos valores para o consumo, velocidade atual, esforço tratores, deslocamento, força aceleradora. Eles determinam os critérios utilizados para se atribuir uma utilidade ou pontuação. Na sequência as informações são atualizadas e recuperadas da base de crenças. Em seguida, o agente atribui uma utilidade para cada ponto de aceleração testado mediante os critérios estabelecidos. Nesta etapa, cada ponto de aceleração alternativo recebe uma pontuação de acordo com o estado das variáveis (atributos) acima. A seguir serão demonstrados os pontos de aceleração alternativos recebendo tal pontuação após verificação dos critérios de utilidades:

- ponto de aceleração 0 = -8;

- ponto de aceleração 1 = 5;
- ponto de aceleração 2 = 12;
- ponto de aceleração 3 = 6;
- ponto de aceleração 4 = 6;
- ponto de aceleração 5 = 6;
- ponto de aceleração 6 = 6;
- ponto de aceleração 7 = 6;
- ponto de aceleração 8 = 6.

Após atribuir a utilidade para cada ponto de aceleração alternativo, o agente compara a pontuação recebida para filtrar o ponto de aceleração que obteve a maior utilidade. Nesta situação, o agente seleciona o ponto de aceleração “2” que obteve a maior pontuação “12”. Portanto:

- Ponto de aceleração desejado = 2;

As selecionar o ponto de aceleração desejado “2” e considerar o ponto de aceleração atual “0”, o agente utiliza um conjunto de regras classificadoras para definir a intenção “acelerar”. Logo, a intenção é configurada pelo ponto de aceleração desejado “2” e pela ação “acelerar”. Depois, o agente insere a intenção (ponto de aceleração desejado e ação) no plano.

A seguir, o agente aplica a intenção no ambiente virtual e obtém as pós-condições do plano. Para efetivar tal procedimento, o agente realiza a troca de informações com o calculador enviando os seguintes dados: *posição atual*, *posição inicial*, *posição final*, *velocidade atual*, *velocidade desejada*, *ponto de aceleração atual*, *coeficiente de atrito*, *direção e número de locomotivas*. Após realizar os cálculos, o calculador devolve como resultado ao agente os seguintes dados: *esforço trator efetivo*, *esforço trator aderente*, *resistência dos vagões*, *resistências das locomotivas*, *deslocamento*, *consumo*, *velocidade máxima permitida*, *velocidade máxima projetada*, *psi utilizado*, *velocidade desejada*, *tempo e força aceleradora*. Com os dados obtidos por meio da realização dos cálculos, o agente atualiza as informações no estado atual do trem em seu ambiente virtual, conforme abaixo:

- ponto de aceleração atual = 2;
- ponto de medida atual = 6.831;

- velocidade atual = 2;
- velocidade desejada = 4;
- velocidade máxima = 60;
- velocidade máxima projetada = 40;
- esforço trator efetivo = 61.615,6;
- esforço trator aderente = 73.928,8;
- força aceleradora = 36.187,6
- resistência total: 19.443,701
- resistência das locomotivas: -11.968,6;
- resistência dos vagões: 31.412,3;
- pressão de freios: 90,0;
- deslocamento: 1,51467;
- distância percorrida planejamento: 1,51467;
- tempo gasto: 5,4528;
- consumo: 0,1939018;
- ponto de medida inicial: 6.831;
- ponto de medida final: 3.222;
- coeficiente de atrito: 0,22;
- tempo total: 0,09088;
- consumo total: 0,1939018;
- LTKB: 37,36599.

Com as informações prestadas acima, o agente finaliza a configuração da primeira ação do plano inserindo as pré-condições de execução e as pós-condições observadas no ambiente virtual, referentes ao primeiro ciclo do raciocínio prático. A Tabela 7 mostra os valores referentes à primeira ação planejada.

Tabela 7: Estados das variáveis referentes à primeira ação planejada

<b>Plano</b>			
<b>Subplano: 1</b>			
<b>Ação: Acelerar</b>			
<b>Pré-Condição</b>			
Ponto de Aceleração Atual: 0	Velocidade Atual menor ou igual à Velocidade Máxima	Esforço Trator Efetivo menor ou igual ao Esforço Trator Aderente	Coefficiente de Atrito igual a 0,22
<b>Pós-Condição</b>			
Ponto de Aceleração Desejado: 2	Força Aceleradora: 36.187,6 kgf	Consumo: 0,1939018 litros	
Ponto de medida Atual: 6.831	Resistência Total: 19.443,701 kgf/tf	Ponto de Medida Inicial: 6.831	
Velocidade Atual: 2 km	Resistência das Locomotivas: -11.968,6 kgf/tf	Ponto de Medida Final: 3.222	
Velocidade Desejada: 4 km	Resistência dos Vagões: 31.412,3 kgf/tf	Coefficiente de Atrito: 0,22	
Velocidade Máxima: 60 km	Pressão de Freios: 90,0 psi	Tempo Total: 0,09088 minutos	
Velocidade Máxima Projetada: 40 km	Deslocamento: 1,51467 metros	Consumo Total: 0,1939018 litros	
Esforço Trator Efetivo: 61.615,6 kgf	Distância Percorrida no Planejamento: 1,51467 metros	LTKB: 37,36599 l/ton brt	
Esforço Trator Aderente: 73.928,8 kgf	Tempo Gasto: 5,4528 segundos		

Para gerar a próxima ação planejada (intenção), o agente considera o ponto de aceleração atual = 2, estabelecendo os possíveis pontos de aceleração alternativos: {0,1,2,3}. Logo, o agente atribui a utilidade para os pontos de aceleração alternativos conforme abaixo:

- ponto de aceleração 0 = 6;
- ponto de aceleração 1 = 6;
- ponto de aceleração 2 = 8,
- ponto de aceleração 3 = 7.

Logo, para gerar a intenção, o agente seleciona o ponto de aceleração que obteve a maior utilidade, neste caso o ponto de aceleração “2” e define a ação “manter”. A Tabela 8 mostra o estado das variáveis referentes à segunda ação planejada.

Tabela 8: Estados das variáveis referentes à segunda ação planejada

<b>Plano</b>			
<b>Subplano: 2</b>			
<b>Ação: Manter</b>			
<b>Pré-Condição</b>			
Ponto de Aceleração Atual: 2	Velocidade Atual menor ou igual à Velocidade Máxima	Esforço Trator Efetivo menor ou igual ao Esforço Trator Aderente	Coefficiente de Atrito igual a 0,22
<b>Pós-Condição</b>			
Ponto de Aceleração Desejado: 2	Força Aceleradora: 5.287,85 kgf	Consumo: 1,326974 litros	
Ponto de medida Atual: 6.829	Resistência Total: 19.539,238 kgf/tf	Ponto de Medida Inicial: 6.831	
Velocidade Atual: 4 km	Resistência das Locomotivas: -11.961,46 kgf/tf	Ponto de Medida Final: 3.222	
Velocidade Desejada: 6 km	Resistência dos Vagões: 31.500,7 kgf/tf	Coefficiente de Atrito: 0,22	
Velocidade Máxima: 60 km	Pressão de Freios: 90,0 psi	Tempo Total: 0,7128217 minutos	
Velocidade Máxima Projetada: 40 km	Deslocamento: 31,097 metros	Consumo Total: 1,5208758 litros	
Esforço Trator Efetivo: 30.807,8 kgf	Distância Percorrida no Planejamento: 32,61167 metros	LTKB: 13,612357 l/ton brt	
Esforço Trator Aderente: 72.493,2 kgf	Tempo Gasto: 37,3165 segundos		

Para gerar a próxima intenção, o agente considera o ponto de aceleração atual = 2, os mesmos pontos de aceleração alternativos observados anteriormente: {0,1,2,3}. Logo, o agente atribui uma utilidade para cada ponto alternativo conforme abaixo:

- ponto de aceleração 0 = 6;
- ponto de aceleração 1 = 6;

- ponto de aceleração 2 = 8,
- ponto de aceleração 3 = 7.

Em seguida, para gerar a intenção, o agente seleciona o ponto de aceleração que obteve a maior utilidade, ou seja, novamente ponto de aceleração “2”.

Tabela 9: Estados das variáveis referentes à terceira ação planejada

<b>Plano</b>			
<b>Subplano: 3</b>			
<b>Ação: Manter</b>			
<b>Pré-Condição</b>			
Ponto de Aceleração Atual: 2	Velocidade Atual menor ou igual à Velocidade Máxima	Esforço Trator Efetivo menor ou igual ao Esforço Trator Aderente	Coefficiente de Atrito igual a 0,22
<b>Pós-Condição</b>			
Ponto de Aceleração Desejado: 2	Força Aceleradora: 3.851,69 kgf	Consumo: 1,821754 litros	
Ponto de medida Atual: 6.825	Resistência Total: 18.278,371 kgf/tf	Ponto de Medida Inicial: 6.831	
Velocidade Atual: 6 km	Resistência das Locomotivas: - 4.385,2295 kgf/tf	Ponto de Medida Final: 3.222	
Velocidade Desejada: 8 km	Resistência dos Vagões: 22.663,6 kgf/tf	Coefficiente de Atrito: 0,22	
Velocidade Máxima: 60 km	Pressão de Freios: 90,0 psi	Tempo Total: 1,5666617 minutos	
Velocidade Máxima Projetada: 40 km	Deslocamento: 71,1534 metros	Consumo Total: 3,34263 litros	
Esforço Trator Efetivo: 20.538,6 kgf	Distância Percorrida no Planejamento: 103,76507 metros	LTKB: 9,402638 l/ton brt	
Esforço Trator Aderente: 71.112,4 kgf	Tempo Gasto: 51,2304 segundos		

Portanto, o ciclo continua até que o veículo consiga vencer a distância planejada de 72,18 km, mais precisamente quando este possa atingir a posição final planejada, o ponto de

medida “3.222”. A seguir, a Tabela 10 apresenta a última ação planejada durante a simulação da viagem. Para gerar a última ação (intenção), o agente considera o ponto de aceleração atual = -1. Neste caso, os pontos de aceleração alternativos são: {0,1,2,3,4,5,6,7,8}. Após analisar os efeitos de cada ponto de aceleração e consultar os critérios de pontuação, o agente atribui uma utilidade a cada ponto alternativo conforme abaixo:

- ponto de aceleração 0 = 6;
- ponto de aceleração 1 = 6;
- ponto de aceleração 2 = 6;
- ponto de aceleração 3 = 1;
- ponto de aceleração 4 = 6;
- ponto de aceleração 5 = 5;
- ponto de aceleração 6 = 4;
- ponto de aceleração 7 = 3;
- ponto de aceleração 8 = -8.

É possível observar um empate entre as utilidades dos pontos de aceleração: 0, 1, 2, 4. Quando esta situação acontece, o agente escolhe o menor ponto de aceleração, em outras palavras, aquele que consome uma menor quantidade de combustível, neste caso, o ponto de aceleração “0”. A Tabela 10 mostra os estados das variáveis referentes à última ação planejada.

Tabela 10: Estados das variáveis referentes à última ação planejada

<b>Plano</b>			
<b>Subplano: 589</b>			
<b>Ação: acelerar</b>			
<b>Pré-Condição</b>			
Ponto de Aceleração Atual: -1	Velocidade Atual menor ou igual à Velocidade Máxima	Esforço Trator Efetivo menor ou igual ao Esforço Trator Aderente	Coefficiente de Atrito igual a 0,22
<b>Pós-Condição</b>			
Ponto de Aceleração Desejado: 0	Força Aceleradora: -35.110,2 kgf	Consumo: 0,0593484 litros	
Ponto de medida Atual: 3.223	Resistência Total: 35.076,652 kgf/tf	Ponto de Medida Inicial: 6.831	
Velocidade Atual: 41km	Resistência das Locomotivas: -67.148 kgf/tf	Ponto de Medida Final: 3.222	
Velocidade Desejada: 43 km	Resistência dos Vagões: 35.143,8 kgf/tf	Coefficiente de Atrito: 0,22	
Velocidade Máxima: 53 km	Pressão de Freios: 90 psi	Tempo Total: 144, 40065 segundos	
Velocidade Máxima Projetada: 53 km	Deslocamento: 64,0069 metros	Consumo Total: 1.157,7712 litros	
Esforço Trator Efetivo: 0 kgf	Distância Percorrida no Planejamento: 72.235,58 metros	LTKB: 4.678259 l/ton brt	
Esforço Trator Aderente: 52.956,0 kgf	Tempo Gasto: 5,62012 segundos		

O agente ao finalizar a elaboração do plano, inicia a execução do mesmo alterando o estado do ambiente real de condução. O agente ao analisar cada ação, verifica se ela viola uma pré-condição de execução, e ainda, verifica se o ambiente real não sofreu mudanças. Caso perceba alguma quebra de restrição ou mudanças no ambiente, o agente realiza o replanejamento, caso contrário, executa as ações do plano. No modo de execução do plano, ao recuperar cada ação do plano, as crenças reais assumem os estados das variáveis (pós-condições) descritos anteriormente.

## **5.6 Considerações Finais**

O simulador proposto possui um conjunto de componentes organizados de acordo com a execução de suas principais atividades. No modo de execução virtual, realizam-se as atividades do raciocínio prático, como deliberação de intenções e planejamento de ações de uma viagem. No modo de execução real, o agente analisa as pré-condições e pós-condições associadas a cada ação do plano, executando-a ou replanejando se necessário. Para realizar o planejamento, definem-se uma posição inicial e uma posição final do trem na via configurando as metas iniciais e finais do agente. Para replanejar, o agente grava o estado atual da última ação válida, remove as ações subsequentes e recomeça a elaboração do novo plano. No próximo capítulo, serão apresentados os resultados obtidos por meio de experimentos realizados.

# Capítulo VI

## 6 Resultados Obtidos

### 6.1 Introdução

O presente capítulo tem por objetivo: (i) avaliar o desempenho dos planos de ação do agente condutor vis-à-vis à simulação de viagens de trens interurbanos de carga, comparando-os com desempenho obtido por condutores humanos em viagens reais de trem; e (ii) apresentar os resultados obtidos em termos de tempo de processamento gastos para planejamento/replanejamento de ações, e na execução do plano. Nestes termos, foi utilizado um trecho de via férrea real que compreende um percurso com início e término entre duas cidades paranaenses. Trata-se de um trecho de via férrea com aclives, declives, retas e diferentes velocidades máximas. Cada viagem executada foi configurada com diferentes quantidades de locomotivas e de vagões totalizando, em particular, pesos diferenciados.

### 6.2 Metodologia de Avaliação

Para avaliar a modelagem do sistema proposto, foram realizados experimentos em nove cenários de viagens. Os cenários reproduzem viagens reais concretizadas por condutores humanos considerando os mesmos dados da composição do veículo e do perfil da via férrea. Cada cenário descreve uma configuração de avaliação de desempenho de uma viagem de trem interurbano de carga. Assim, cada configuração requer: (i) um trem que é composto por  $m$  locomotivas e  $n$  vagões. Cada trem tem um peso, determinado pelo somatório dos pesos individuais de cada locomotiva e cada vagão. Cada vagão tem um peso específico, dado pela soma de sua tara e sua carga; (ii) uma descrição do trecho de via férrea por onde deve passar o

trem. Cada ponto desta descrição reúne um conjunto de dados, como velocidade máxima, aative, declive, curva, reta, etc.; e (iii) uma indicação do ponto início e do ponto término da viagem.

Conforme já citado, as informações de configuração de cada cenário dos experimentos são de viagens reais de trens interurbanos de cargas já realizadas por condutores humanos. Desta forma, é possível comparar o resultado de uma mesma viagem executada por um condutor humano e pelo agente proposto. O comportamento do agente condutor foi avaliado nas seguintes situações: (i) segurança da viagem, que verifica a ocorrência do excesso de velocidade e patinagem das rodas sobre o trilho durante o percurso; (ii) tempo de viagem, que representa o tempo decorrido desde o início do deslocamento até a chegada do trem; e (iii) consumo de combustível em LTKB (*Litros por Tonelada Bruta Transportada*).

Nos experimentos realizados foi possível observar: (i) planejamento de pontos de aceleração; (ii) percentual de ações planejadas; (iii) pontos de aceleração aplicados; (iv) percentual de pontos de aceleração utilizados; (v) esforço trator e resistências; (vi) esforços tratores; e (vii) velocidades. A Tabela 11 mostra de forma resumida os dados dos experimentos, onde cada linha representa um cenário.

Tabela 11: Descrição dos cenários dos experimentos

Cenários	Distância (km)	Origem*	Destino*	Número de Locomotivas	Número de vagões	Peso (ton)
1	68,00	6.612	3.184	3	58	6.278
2	66,70	6.550	3.250	4	100	6.342
3	66,90	6.642	3.295	4	70	6.541
4	65,00	6.431	3.284	2	30	3.393
5	72,18	6.831	3.222	2	31	3.426
6	70,61	6.626	3.234	3	47	5.199
7	67,50	6.429	3.184	2	31	3.441
8	73,30	6.707	3.184	4	59	6.579
9	71,60	6.636	3.184	2	28	3.118

\*Ponto de medida

Apenas nos Cenários 7 e 8 foram simulados o replanejamento de ações. No Cenário 7, após mudar a configuração padrão de execução, acrescentou-se critérios de utilidade para novos atributos, onde houve a necessidade do replanejamento, devido a uma violação de restrição de esforços tratores.

No entanto, no Cenário 8, alterou-se os valores da velocidade máxima permitida para que a velocidade atual do trem a excedesse em um determinado trecho da viagem.

Basicamente, este procedimento foi usado para que uma pré-condição do plano invalidasse uma ação planejada para situação atual do trem na via. A Tabela 12 apresenta uma comparação entre os resultados obtidos e percentuais de ganhos entre as viagens realizadas por um condutor humano e viagens simuladas utilizando o agente condutor.

Tabela 12: Resultados

Cenário de viagem	Peso do trem (t)	Tempo (min)			LTKB (l/ton brt)		
		Condutor Humano	Agente	Ganhos %	Condutor Humano	Agente	Ganhos %
1	6.278	287	184	35%	6,19	4,68	24,3%
2	6.432	223	154	30,9%	5,68	4,92	13,3%
3	6.541	162	133	17,9%	6,23	4,1	34,1%
4	3.393	146	139	4,7%	---	4,2	---
5	3.426	219	145	33,7%	6,49	4,68	27,8%
6	5.199	259	154	33,7%	6,29	4,57	27,3%
7 Replanejamento	3.441	221	117	47%	6,17	4,2	31,9%
8	6.579	235	120	48,9%	6,3	4,32	31,4%
8 Replanejamento	6.579	235	125	46,8%	6,3	4,58	27,3%
9	3.118	339	183	46%	6,26	4,16	33,5%

A Tabela 12 apresentou os resultados obtidos na simulação de nove cenários de viagens. A coluna peso do trem indica o peso total do veículo em toneladas (t), composto por locomotivas e por vagões. A coluna tempo, LTKB e ganhos, mostram, respectivamente, o tempo total em minutos (m), o consumo das viagens e os ganhos obtidos pelo agente em comparação ao condutor humano. Na sequência, de forma detalhada, serão demonstrados os resultados obtidos dos cenários 5, 7 e 8.

### 6.2.1 Experimento: Cenário 5

Conforme os dados apresentados na Tabela 11, o experimento no contexto deste cenário corresponde a uma distância a ser percorrida de 72,18 km, um conjunto de duas locomotivas, trinta e um vagões, um peso bruto a transportar de 3.426 toneladas. A Figura 14 ilustra os valores referentes aos pontos de aceleração utilizados neste cenário, visto que o planejado e o executado foram os mesmos.

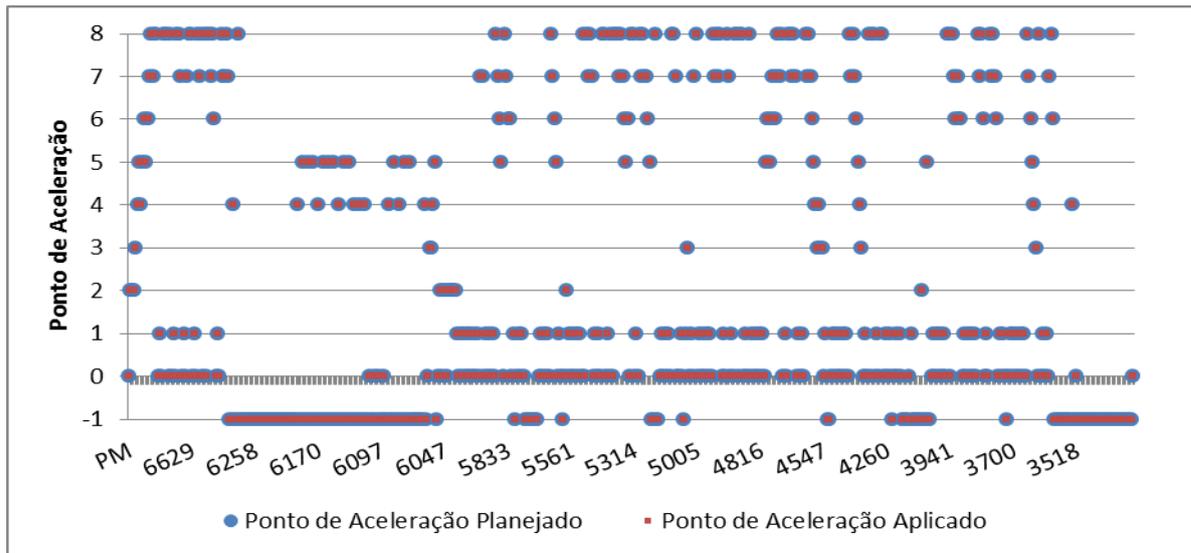


Figura 14: Pontos de aceleração planejados e aplicados

Ao aplicar um determinado ponto de aceleração, o veículo se movimenta a uma velocidade que não deve exceder a velocidade máxima permitida. Esta é determinada por restrições de segurança para uma determinada viagem. Durante o deslocamento, para haver eficácia, a velocidade deve ser conservada próxima à velocidade máxima permitida, e constante, otimizando assim, o tempo de viagem e o consumo. A Figura 15 ilustra a velocidade praticada pelo agente em comparação a velocidade praticada pelo condutor humano (maquinista).

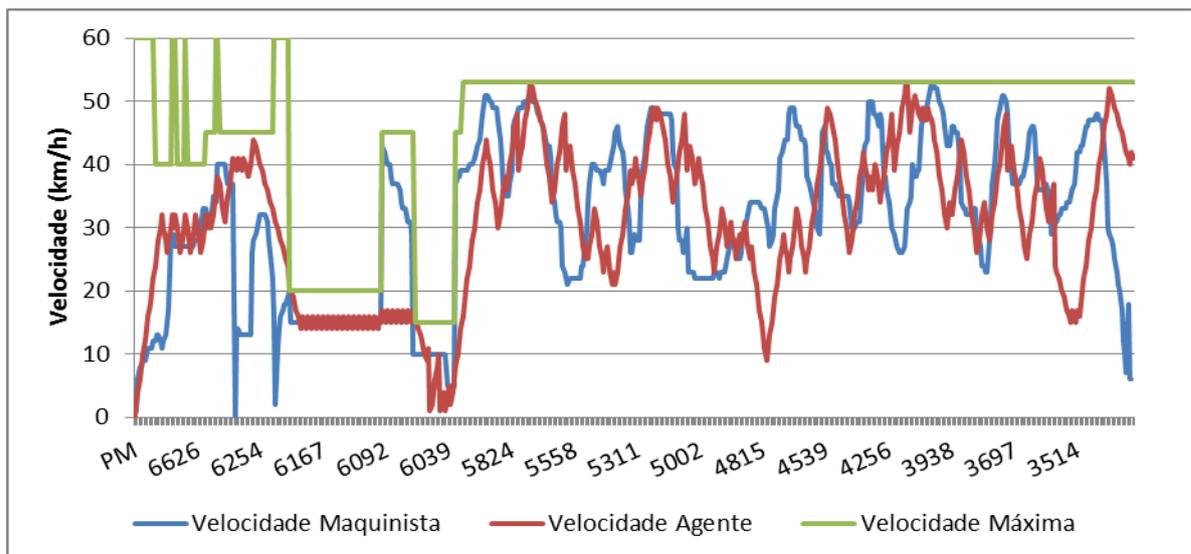


Figura 15: Velocidade praticada pelo condutor humano (maquinista) e pelo agente

Após analisar os resultados obtidos com os experimentos realizados neste Cenário, observa na Figura 16 (a) que o planejamento da ação “acelerar” foi realizada em 35% e a ação “frear” foi realizada em 28% do percurso de um total de 589 ações planejadas. Percebeu-se neste cenário a necessidade em se produzir uma força de aceleração necessária para vencer as resistências do percurso e razoável utilização de freio. Isto se deve ao fato que o perfil da via neste trajeto se mostrou bastante acidentado, com variações de restrições de velocidade, o que faz com que o agente utilize o recurso (freio) para reduzir a velocidade do trem, mesmo que ele sempre buscasse atingir a velocidade máxima do trecho.

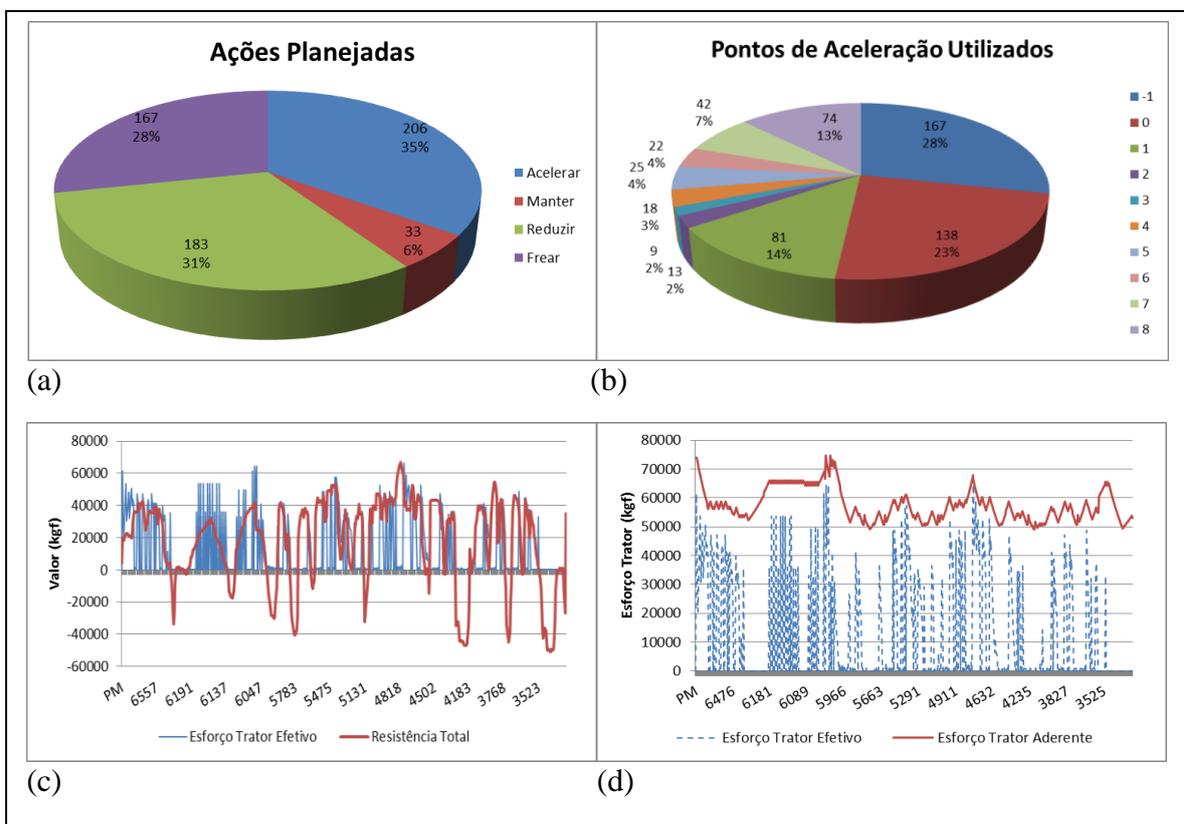


Figura 16: (a) Quantidade de ações planejadas, (b) Quantidade de pontos de aceleração utilizados, (c) Esforço Trator Efetivo (kgf) e Resistências (Kgf), (d) Esforços Tratores (Kgf) - (Cenário 5)

De acordo com a Figura 16 (b), verificou-se que o ponto de aceleração “8” foi utilizado em 13% do percurso, isto mostra a utilização de potência máxima para vencer as resistências do percurso e aumentar a velocidade. O ponto de aceleração “1” foi utilizado em 14% das ações realizadas pelo agente à medida que proporcionou um esforço trator suficiente para a manutenção da velocidade do veículo e a economia de combustível. Finalmente, o agente utilizou o ponto de aceleração “0” em 23 % do percurso, aproveitando as forças

aceleradoras provocadas por resistências negativas, economizando combustível e forças de tração.

### 6.2.2 Experimento: Cenário 7 – Replanejamento Automático

Neste cenário, ao se utilizar a configuração padrão de regras do sistema, não foi possível obter um conjunto de ações suficientes para que o trem chegasse até o final do percurso, devido à falta de potência ocasionada pelos pontos de aceleração selecionados pelo agente. Os resultados apontaram força de aceleração insuficiente para continuidade da viagem a partir de uma determinada posição da via férrea. Houve a necessidade de realizar mudanças nos critérios de pontuação para alterar a utilidade dos atributos, com o objetivo de aumentar a força de aceleração e determinar a continuidade da simulação da viagem. Além dos critérios já mencionados, foram utilizados os seguintes critérios de pontuação para efetivar os experimentos neste cenário.

Tabela 13: Definição dos valores para o atributo *deslocamento e velocidade*

<b>Crítérios</b>	<b>Intervalo de Valores</b>	<b>Utilidade</b>
Adequado	deslocamento > 300 m e deslocamento <= 500 m	+5
Almejada	velocidade atual = velocidade desejada	+5

Entretanto, ao se realizar os novos experimentos, verificou-se que no ponto de medida “4.800”, a ação “aumentar” selecionou o ponto de aceleração “7”, o que ocasionou naquela ocasião, um aumento significativo no esforço trator efetivo que foi de 65.889 kgf. Comparado ao esforço efetivo na mesma localização do trem na via, que foi de 64.369 kgf, conforme mostra a Figura 17, houve excesso de esforço trator efetivo. Não obstante, esta situação causa a patinação das rodas das locomotivas sobre o trilho. Ela foi definida como uma das pré-condições de execução de uma ação do plano.

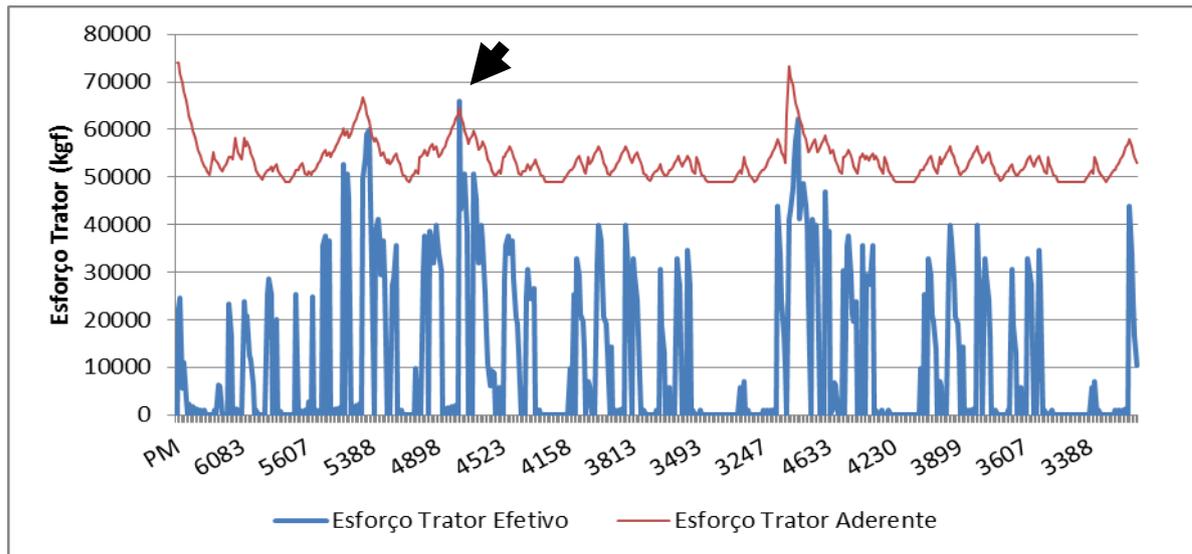


Figura 17: Violação de esforços

No momento da execução do plano, o agente ao perceber a quebra de restrição citada, solicita que um novo plano seja criado, salva o estado anterior à posição do trem referida, e inicia um novo plano de ação. Observou-se que o novo plano foi criado a partir do ponto de medida “4.816”. Para iniciar o novo plano de ação, o agente define um novo ponto de aceleração que servirá como entrada inicial ao ciclo de raciocínio, para que ele possa definir os novos pontos de aceleração aplicáveis e as novas intenções. Contudo, esta nova ação do plano deve produzir um esforço trator efetivo que não supere o esforço trator aderente dando sequência a elaboração e execução de um novo plano de viagem. A Figura 18 ilustra os pontos de aceleração planejados/replanejados e aplicados neste cenário.

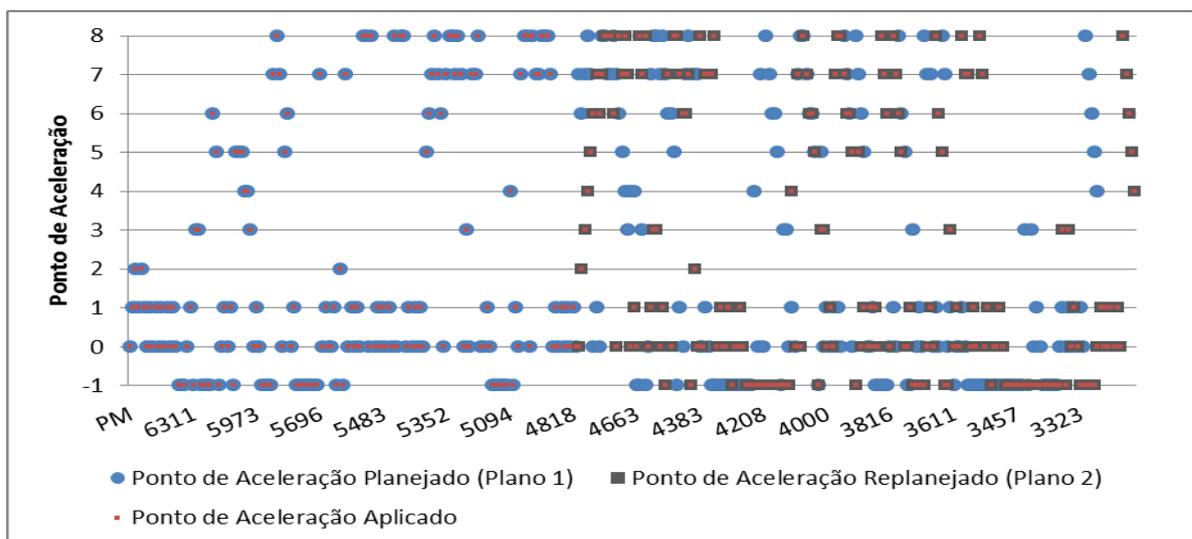


Figura 18: Pontos de aceleração planejados/replanejados e aplicados

De acordo com a com a Figura 19, após replanejamento e execução do novo plano, observou-se que a patinagem verificada anteriormente na Figura 17 não ocorre, o que indica que o novo plano foi suficiente para que o esforço trator efetivo não superasse o esforço trator aderente, nesta nova situação.

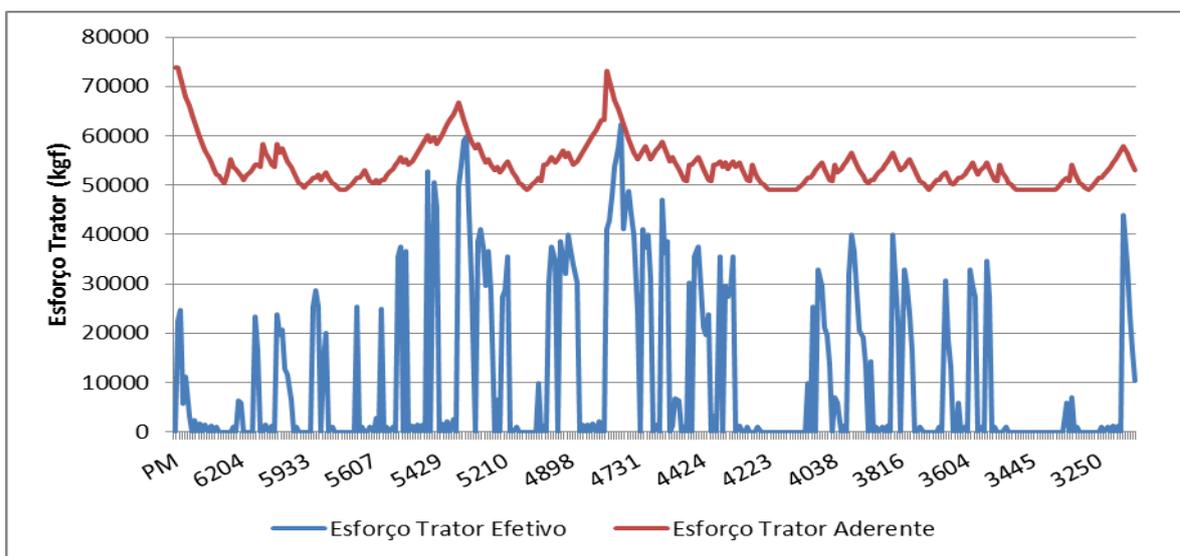


Figura 19: Esforços tratores após replanejamento

Analisando a Figura 20 (b), observa-se que a ação “acelerar” foi realizada em 34% do percurso de um total de 350 ações realizadas. Este fato nos leva a considerar que o agente em boa parte da viagem, necessitou aumentar a força de aceleração para vencer as resistências e aumentar a velocidade. Em contrapartida, verificou-se que ação “reduzir” foi realizada em 32% do trajeto. Tal ação nos leva a supor que o agente procurou utilizar pontos de aceleração que aproveitassem as forças aceleradoras proporcionadas pelo perfil da via que apresentou poucas variações de restrições de velocidade máxima (Figura 20 (e)). Verificou-se ainda que, o agente realizou frenagem em 25% do percurso, subtendo a redução da velocidade do trem para que evitasse a quebra de restrições de velocidade máxima permitida.

De acordo com a Figura 20 (c), verificou-se que o ponto de aceleração “0” foi utilizado em 25% do percurso, seguido pelo ponto de aceleração “1” que foi utilizado em 14% do trajeto. O ponto de aceleração “0” indica uma marcha lenta e consome uma baixa quantidade de combustível. Geralmente ele é utilizado quando o trem se encontra em resistências negativas.

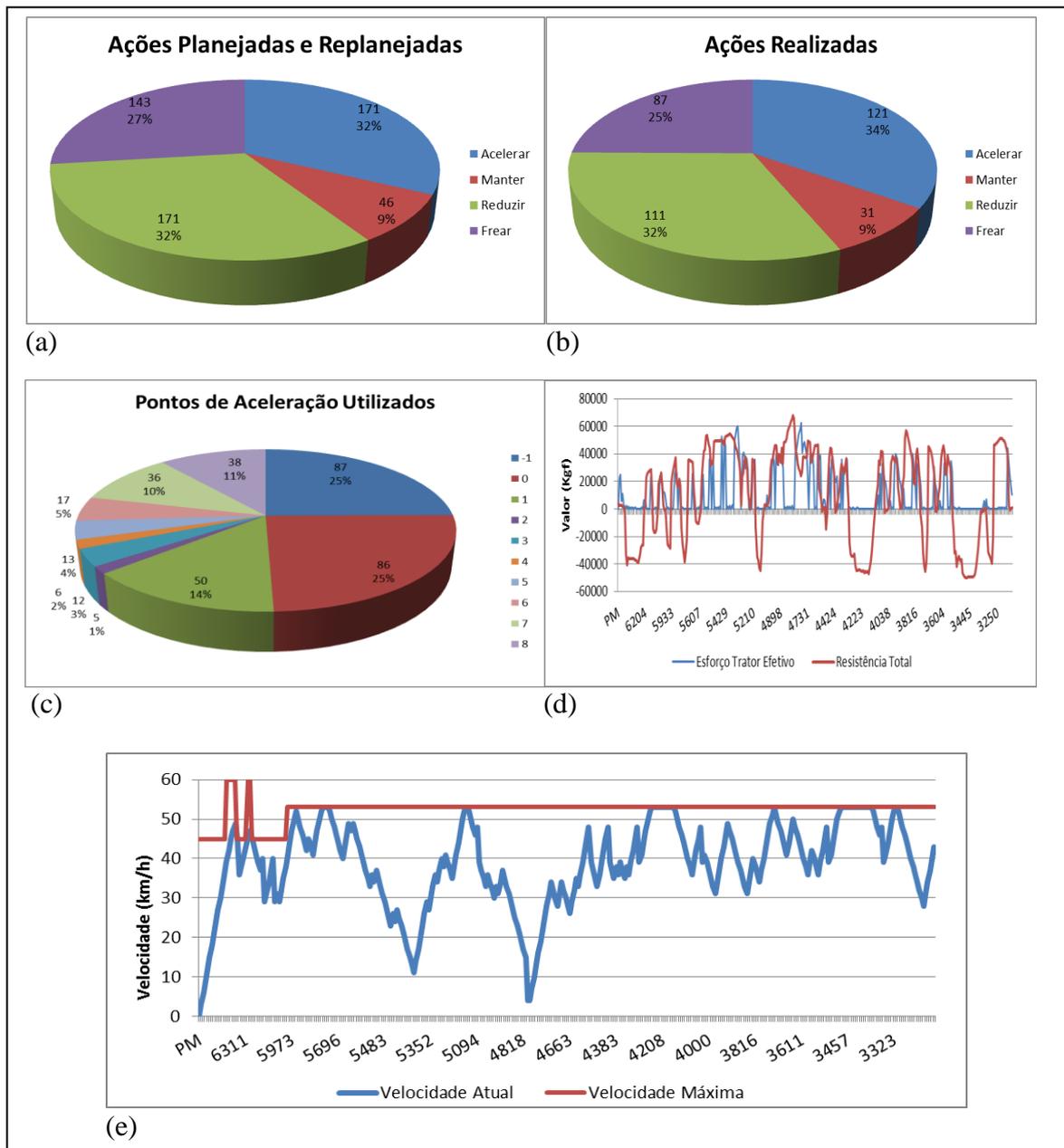


Figura 20: (a) Quantidade de ações planejadas e replanejadas, (b) Quantidade de ações realizadas, (c) Quantidade de pontos de aceleração utilizados, (d) Esforço Trator Efetivo (Kgf) e Resistências (Kgf), (e) Velocidades (Km/h) - (Cenário 7 - Replanejamento Automático)

### 6.2.3 Experimento: Cenário 8

O experimento no contexto deste cenário corresponde a um trajeto de 73,30 km, um conjunto de quatro locomotivas cinquenta e nove vagões, um peso bruto a transportar de 6.579 toneladas. A Figura 21 ilustra o gráfico relativo aos pontos de aceleração planejados e aplicados, no entanto, sem replanejamento de ações, visto que o planejado e o executado foram iguais.

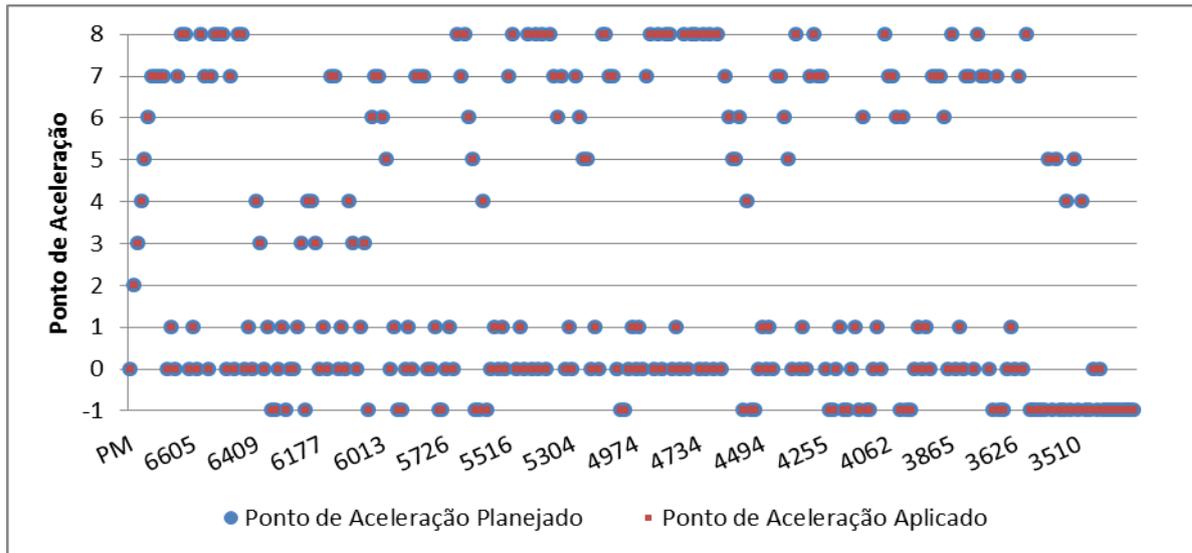


Figura 21: Pontos de aceleração planejados e aplicados

Analisando a Figura 22 (a), observa-se que o planejamento da ação “acelerar” foi realizada em 41% do percurso de um total de 271 ações planejadas. Nesta situação, o agente aumentou a força de aceleração para ganhar velocidade e vencer as resistências deste trecho. Observou-se que ação “reduzir” foi realizada em 30% do trajeto, supostamente para aproveitar os pontos de aceleração que produzissem uma força suficiente para desenvolver uma velocidade necessária para o movimento e ainda otimizar o consumo de combustível. Observou-se também, a utilização do freio em 19% da viagem para a redução de velocidade, evitando a ultrapassagem da velocidade máxima permitida da via.

De acordo com a Figura 22 (b), verificou-se que o ponto de aceleração “0” foi utilizado em 28% do percurso e o ponto de aceleração “7” foi utilizado em 15% do trajeto. Este ponto de aceleração gera uma potência suficiente para percorrer trechos da ferrovia que apresentam baixa e alta resistência.

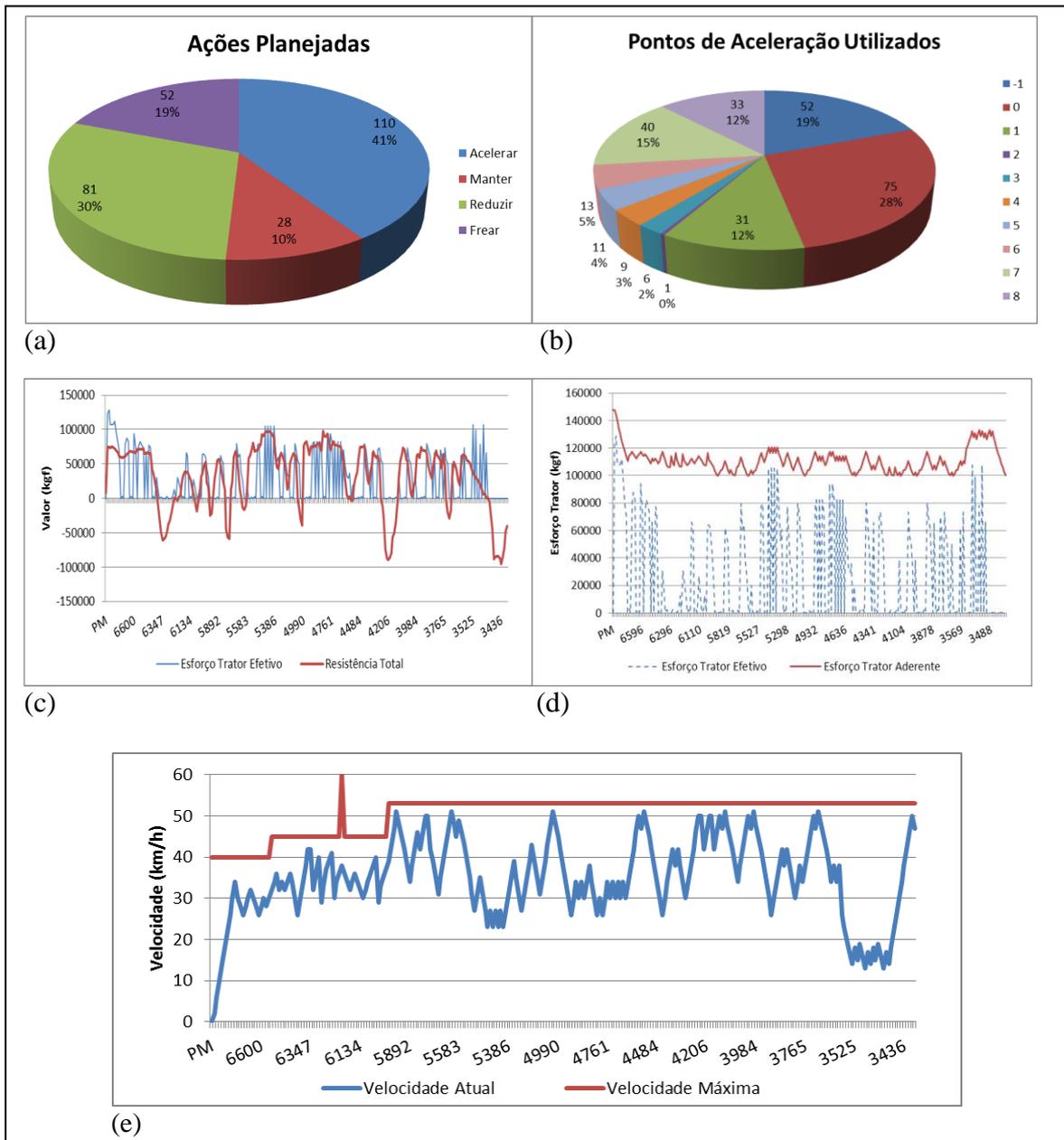


Figura 22: (a) Quantidade de ações planejadas, (b) Quantidade de pontos de aceleração utilizados, (c) Esforço Trator Efetivo (Kgf) e Resistências (Kgf), (d) Esforços Tratores (Kgf), (e) Velocidades (Km/h) - (Cenário 8)

#### 6.2.4 Experimento: Cenário 8 – Replanejamento Simulado

Para simular o replanejamento neste cenário, alterou-se o valor da velocidade máxima permitida em uma determinada posição na via férrea, com o objetivo de tornar a velocidade atual do trem maior que a velocidade máxima da via, e consequentemente, invalidar o plano de ação. Esta é uma das pré-condições previstas pelo sistema para que uma ação possa ser

realizada. No entanto, quando uma pré-condição não é satisfeita, o agente realiza o replanejamento das ações.

O ponto de medida escolhido para realizar tal procedimento foi o “3.711”. Neste ponto de medida a velocidade atual do veículo era de 50 Km/h e a velocidade máxima da via era de 53 km/h. A velocidade máxima foi alterada para 48 Km/h onde proporcionou uma violação de restrição de velocidade. De acordo com a Figura 23, é possível perceber a redução de velocidade máxima (linha vermelha, próximo do final da viagem) provocando tal quebra de restrição.

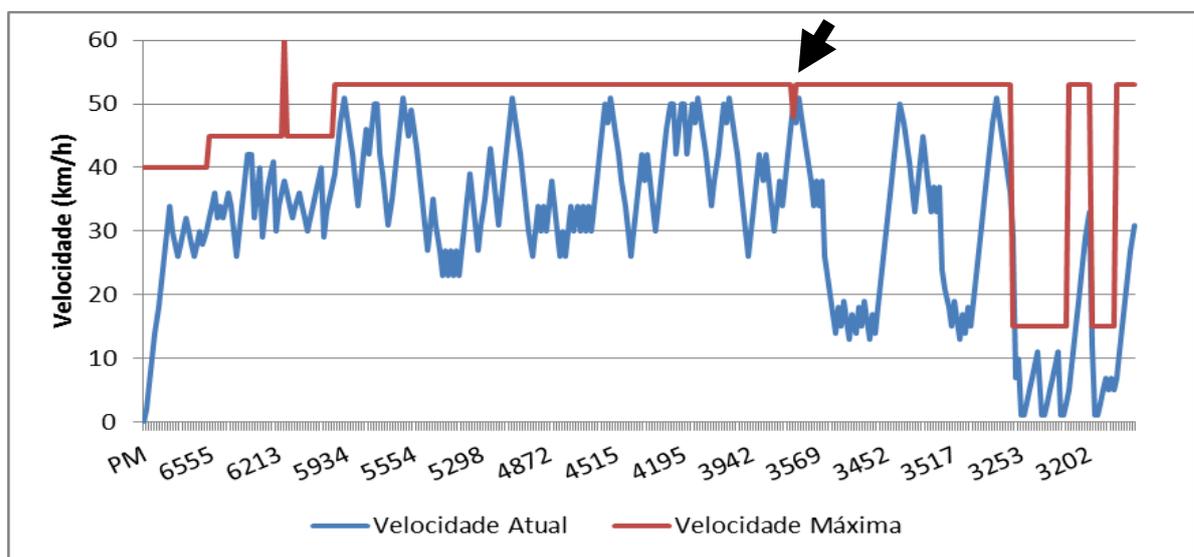


Figura 23: Violação de velocidade máxima permitida.

Não obstante, ao executar o plano de viagem, o agente analisa a aplicabilidade da ação neste ponto de medida, percebe a violação da restrição de velocidade mencionada e solicita um novo plano a partir da posição anterior (3.711). Neste momento, o estado atual das variáveis referentes ao ponto de medida anterior “3.730” é registrado, e o agente configura o ambiente como virtual solicitando um novo plano a partir desta posição. Como houve um excesso de velocidade, a ação inicial do novo plano é configurada como “frear”. Esta ação é recomendada para reduzir a velocidade do veículo. A Figura 24 ilustra o gráfico relativo à geração dos pontos de aceleração planejados/replanejados e aplicados.

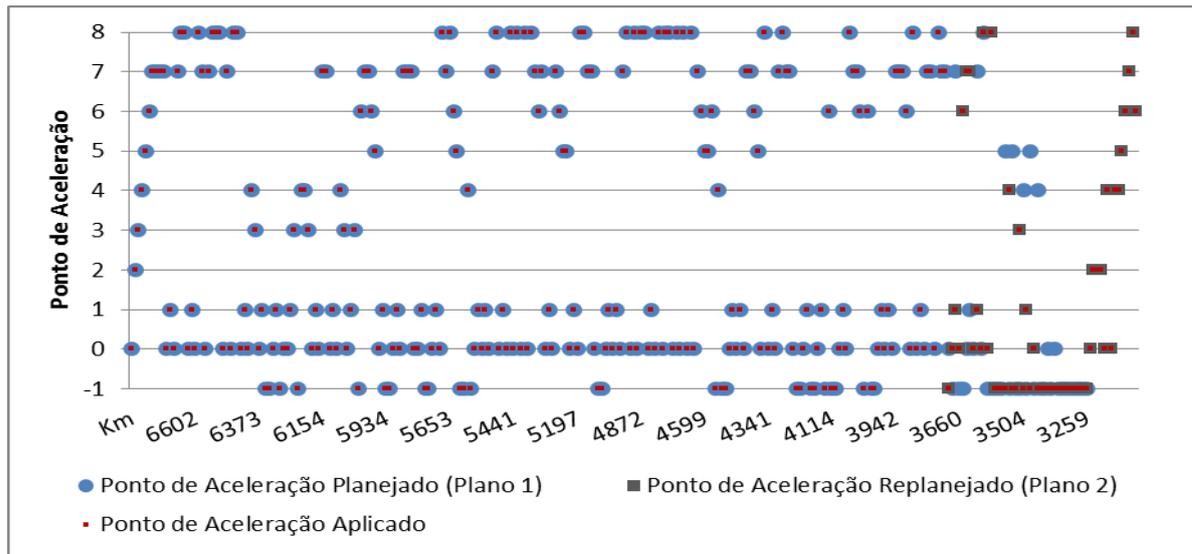


Figura 24: Pontos de aceleração planejados/replanejados e aplicados

Contudo, ao observar a Figura 25, é possível verificar que a violação de restrição de velocidade não se repete o que nos leva a inferir que o replanejamento foi útil para resolver tal problema.

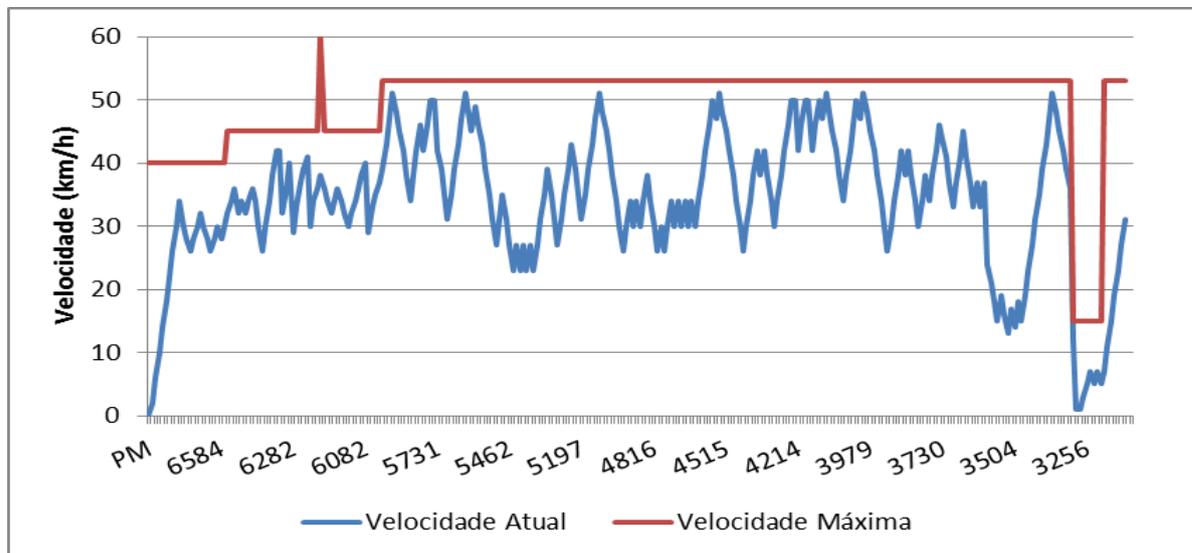


Figura 25: Velocidades (km/h)

A Figura 26 ilustra os gráficos relativos à execução do plano gerado após planejamento e replanejamento de ações. A Figura 26 (e) ilustra as resistências e os esforços tratores desenvolvidos durante a simulação da viagem que determinam o ganho de velocidade

do trem. Portanto, conclui-se que o veículo ganhou velocidade quando o esforço trator efetivo desenvolvido foi maior que a resistência, e perdeu velocidade quando o esforço trator efetivo foi menor que a resistência.

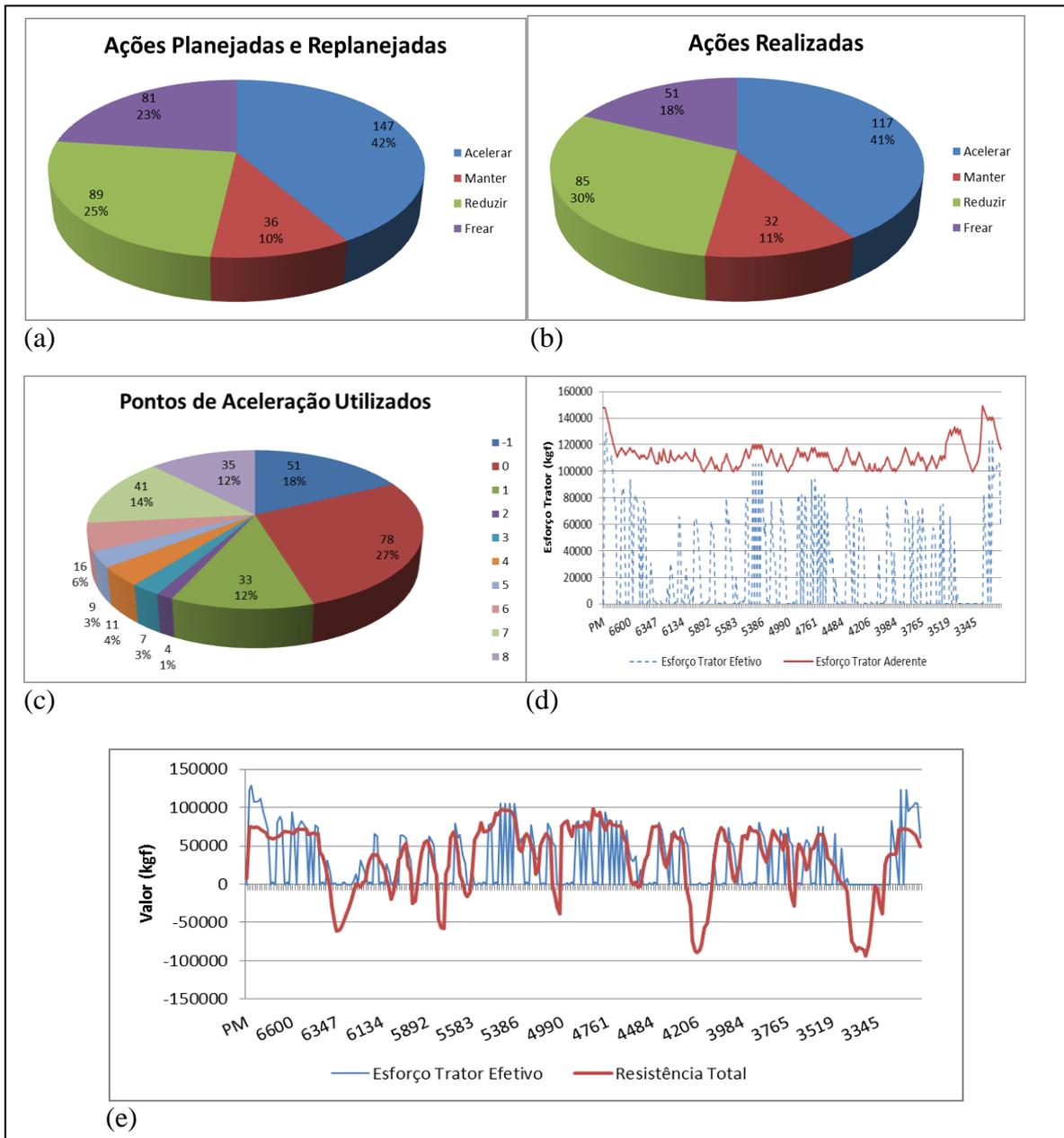


Figura 26: (a) Quantidade de ações planejadas e replanejadas, (b) Quantidade de ações realizadas, (c) Quantidade de pontos de aceleração aplicados com replanejamento, (d) Esforços Tratores (Kgf), (e) Esforço Trator Efetivo (Kgf) e Resistências (Kgf) - (Cenário 8 - Replanejamento Simulado)

### **6.3 Análise e Discussão**

A eficiência na otimização do combustível durante as simulações de viagens realizadas, foi observada quando a velocidade se manteve próxima à velocidade limite da via, ou sem bruscas variações. Entretanto, é difícil estabelecer de maneira precisa, a relação entre o ponto de aceleração adequado e a velocidade ideal para uma ação eficiente e segura em diferentes perfis de vias, principalmente para especialistas humanos. Assim, é possível concluir que a combinação das forças de resistências em conjunto com as restrições de velocidade máxima permitida e os pontos de aceleração utilizados, são fatores que determinam os níveis de consumo do LTKB e o tempo de viagem.

Percebeu-se após os experimentos, que a técnica empregada na escolha das ações do plano, proporcionou, em determinados momentos, que a velocidade do veículo permanecesse próxima à velocidade limite da via, na qual a redução do LTKB e do tempo de viagem foi alcançada em todas as simulações realizadas. Finalmente, observou-se nos experimentos, redução média de 28% no consumo de combustível e 30% no tempo de viagem.

### **6.4 Análise da Implementação**

Conforme apresentado anteriormente, o ambiente de atuação do agente foi dividido em duas estruturas: virtual e real. A utilização desta estratégia de divisão, junto à escolha da arquitetura de estados mentais BDI, proporcionou uma visão global dos requisitos do sistema, ao qual facilitou a definição dos componentes do software e a elaboração do fluxo de raciocínio do agente, como a elaboração e a análise dos planos de viagens, de forma efetiva no ambiente de condução de trens. Sob este aspecto, observou-se um comportamento proativo, quando o agente desenvolveu atividades do raciocínio prático, como deliberação, tomadas de decisão, elaboração de intenções e estruturação de planos de viagens. Ainda sob o ponto de vista comportamental, o agente também agiu de forma adequada ao atuar no ambiente de execução do plano (real), à medida que analisou as pré-condições e pós-condições de cada ação do plano, solicitando o replanejamento (quando necessário).

A eficiência do agente em termos de segurança, tempo e economia de combustível, foi determinada pela estratégia adotada nas tomadas de decisão e escolha das ações no processo de deliberação do agente, conforme apresentada no Capítulo 4, sendo que, para realizar tais

decisões, utilizou-se um conjunto de critérios baseados em regras de condução de trens, que o permitiu atribuir uma utilidade para cada ponto de aceleração aplicável tendo como objetivo alcançar em média, a ação mais satisfatória em termos de segurança, força de aceleração, deslocamento e consumo. Verificou-se que o uso desta estratégia viabilizou menor consumo de combustível e tempo de viagem nos cenários apresentados.

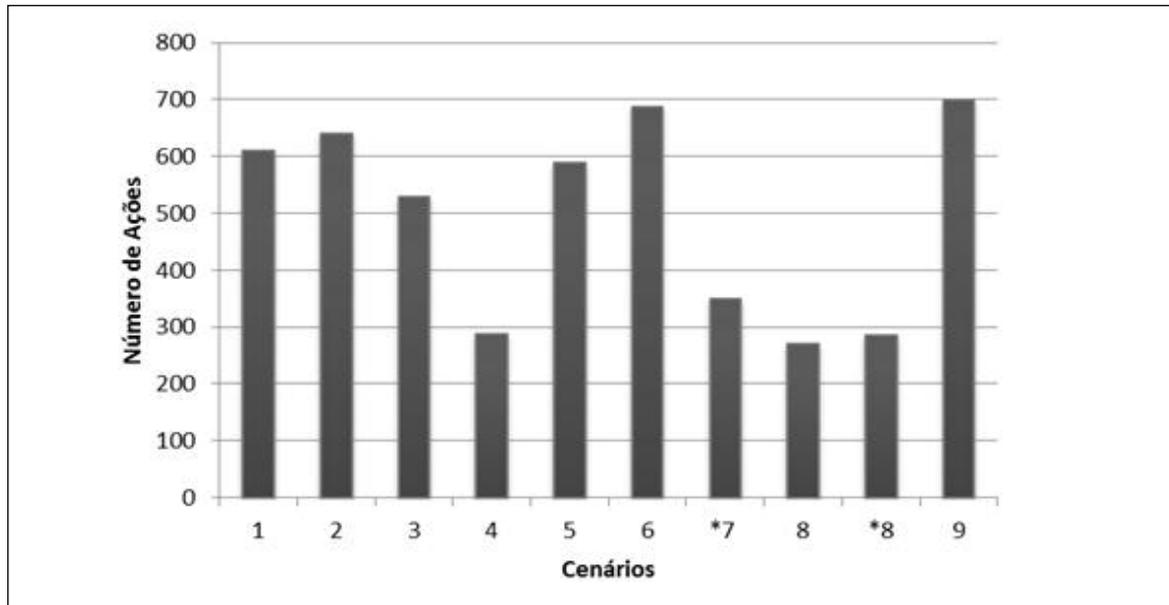
Foi possível observar também, que a técnica utilizada está sempre à procura do melhor ponto de aceleração e poderá encontrar um menor consumo, para uma velocidade suficiente a movimentar o veículo de forma a não extrapolar a força aceleradora.

A inclusão de novas pré-condições podem tornar o sistema ainda mais realista visto que, neste trabalho, consideraram-se apenas algumas restrições do ambiente de condução de trens. Contudo, o sistema se mostrou bastante flexível no que diz respeito à alteração e inclusão de novos critérios que determinam a utilidade de uma ação.

Como limitação deste trabalho, é possível citar a ausência da “reconsideração de intenções”. Ela é uma das atividades previstas pela lógica BDI, em que um agente deve analisar se uma intenção é válida antes que a mesma seja inserida no plano, abandonando-a caso necessário. Em nosso modelo, o agente buscou o melhor ponto de aceleração para inseri-lo no plano. Em situações análogas ao experimento realizado no Cenário 7, o trem pode não atingir o estado final desejado (fim da viagem) por falta força de aceleração, a partir de uma determinada posição da via. Com a utilização deste conceito (reconsideração de intenções), certamente o agente resolveria o problema mencionado, dispensando a necessidade de alteração nos critérios de pontuação (utilidades) conforme foi apresentado neste experimento. No momento em que estiver planejando, o agente ao reconsiderar uma intenção, pode inserir no plano, pontos de aceleração suficientes para garantir o movimento do veículo até o seu destino final. No entanto, o simulador se mostrou eficiente quando foi necessário o replanejamento das ações. Na primeira situação (Cenário 7), o agente reelaborou o plano para corrigir um problema de patinagem ocorrido no momento do planejamento e, na segunda situação (Cenário 8), o agente replanejou as ações para resolver um problema que foi configurado de forma dinâmica para provocar uma mudança no ambiente (quebra de restrição de velocidade).

Quanto ao tempo de processamento dos algoritmos de planejamento e execução do plano, observou-se que estes foram relativamente menores em simulações que obtiveram um menor número de ações, ao qual cada ação representou a escolha de um ponto de aceleração,

ainda que os tempos de execução do plano foram relativamente maiores em simulações que apresentaram replanejamento. A Figura 27 ilustra a quantidade de ações praticadas pelo agente condutor durante os experimentos realizados.



\* Simulação com replanejamento

Figura 27: Quantidade de ações realizadas pelo agente

A Tabela 14 apresenta os resultados obtidos em termos de tempo de processamento para o planejamento/replanejamento de ações e execução do plano, nos diferentes cenários apresentados.

Tabela 14: Tempo de Processamento

Cenários	Planejamento (milissegundos) (A)	Execução-Plano (milissegundos) (B)	Tempo-Total (milissegundos) (C)	Tempo-Total (minutos)	B÷C
1	90.885	6.504	97.389	1 m e 37 s	7%
2	67.343	7.038	74.381	1 m e 14 s	9%
3	52.427	6.209	58.636	58 s	11%
4	26.689	3.095	29.784	30 s	10%
5	40.789	6.975	47.764	48 s	15%
6	59.674	8.348	68.022	1 m e 8s	12%
7*	26.720	17.192	43.912	44 s	39%
8	40.870	6.195	47.065	46 s	13%
8*	46.655	20.486	67.141	1 m e 7 s	31%
9	61.509	8.187	69.696	1 m e 9 s	12%

\*Simulação com replanejamento

A implementação do simulador foi feita em Java™ e os testes foram realizados em um computador com processador Intel® Core™ Duo CPU T7250 2.00 GHz e Memória RAM de 4 GB.

## 6.5 Considerações Finais

Este capítulo apresentou os resultados obtidos em termos de (i) avaliação do desempenho dos planos de viagem comparados ao desempenho de condutores humanos; e (ii) no tempo de execução gasto para o planejamento/replanejamento de ações na simulação de nove cenários de viagens. O desempenho do agente foi avaliado da seguinte forma: (i) segurança da viagem, que verificou o excesso de velocidade e patinagem das rodas sobre o trilho durante o percurso, (ii) tempo gasto que representou o tempo decorrido desde o início do deslocamento até a chegada do trem e, (iii) LTKB (*Litros por Tonelada Bruta Transportada*), que representou o consumo de combustível das locomotivas. Cada cenário de viagem foi configurado de forma específica, como quantidade de locomotivas e vagões, pesos e diferentes percursos a serem percorridos na via, compreendendo um percurso entre duas cidades paranaenses. Cada cenário de viagem foi apresentado como o planejamento da viagem e execução do plano. Em particular nos Cenários 7 e 8, realizou-se a simulação do replanejamento de ações, onde foi comprovado a aplicabilidade desta funcionalidade, que pode ser necessária em situações em que o plano inicialmente gerado não seja aplicável ou ainda devido a mudanças no ambiente. Por fim, verificou-se a eficiência dos planos de ação desenvolvidos pelo agente, onde se demonstrou a redução do consumo de combustível e do tempo em todos os cenários apresentados.

## Capítulo VII

### 7 Conclusão

Este trabalho apresentou um agente BDI capaz de elaborar planos de ação no auxílio à condução de trens. O agente proposto se mostrou adequado para realizar as principais atividades do raciocínio prático, como deliberação, planejamento/replanejamento e execução de planos de viagens.

Para realizar as atividades propostas, utilizaram-se diferentes estratégias para elaboração e execução do plano. Para gerar um plano, o agente realizou deliberação para determinar os pontos de aceleração que resultassem em um estado de maior utilidade em termos de força de aceleração, segurança, consumo e deslocamento. Em contrapartida, para executar o plano, o agente utilizou estratégias mais simples do ponto de vista de complexidade, como análise das pré-condições e pós-condições de execução de ações.

Para estruturar o agente, utilizou-se a arquitetura de estados mentais BDI, ao qual permitiu: (i) organização de ações; (ii) comportamento proativo na elaboração de planos; e (iii) comportamento adequado na execução do plano e solicitação do replanejamento.

Para verificar a eficiência do modelo proposto, avaliou-se o comportamento do agente em termos de segurança, economia e rapidez, em nove simulações de cenários de viagens, comparando os resultados alcançados com o desempenho obtido por especialistas humanos (maquinistas) em viagens reais de trem. Desta forma, observou-se satisfatoriamente um mecanismo inteligente para elaboração e execução de planos, assegurando viagens significativamente mais rápidas e econômicas que as realizadas por condutores humanos. Em dois dos cenários apresentados, o agente foi submetido a realizar o replanejamento. Os resultados mostraram que o agente agiu de forma dinâmica ao ser capaz de corrigir os

problemas apresentados. No primeiro caso (Cenário 7), corrigiu um plano considerado falho, e no segundo caso (Cenário 8), replanejou as ações para garantir a segurança na viagem devido a uma alteração não prevista a uma regra do sistema.

É possível concluir que, ao elaborar um plano de ação aplicável, o agente economiza recursos, à medida que evita esforços adicionais e a necessidade de corrigir planos insatisfatórios. Entretanto, se o ambiente é alterado, o replanejamento se torna necessário para atender as mudanças de requisitos do sistema.

Contudo, para o desenvolvimento de uma aplicação utilizando agentes, é indispensável o conhecimento da sua estrutura essencial de funcionalidade, como percepção do ambiente, raciocínio sobre as ações (planejamento) e o seu mecanismo de atuação. Por fim, é possível afirmar que, o conceito agente é uma alternativa eficaz para resolução de problemas complexos, à medida que, a abordagem facilita a representação dos requisitos do software e proporciona um desenvolvimento incremental do sistema.

Como trabalho futuro, a implementação da reconsideração de intenções pode corrigir uma intenção antes que ela seja inserida no plano. Ainda, o aumento do número de restrições e pré-condições para executar uma ação, podem aumentar as possibilidades de alterações no ambiente real do agente, tornando o sistema ainda mais dinâmico. Por fim, este trabalho poderia se estender também, em simulações levando em consideração a existência de outros agentes condutores para realizar o planejamento/replanejamento de ações.

## Referências Bibliográficas

**ALL.** *Procedimentos na condução de trens.* Curitiba: América Latina Logística do Brasil S.A., 2008.

**BORDINI, R.H., HÜBNER, J.F., WOOLDRIDGE, M.** *Programming Multi-Agent Systems in AgentSpeak Using Jason.* Published by John Wiley & Sons Ltd., October, 2007.

**BORDINI, R.H., HÜBNER, J.F.** BDI Agent programming in AgentSpeak using Jason. *In Francesca Toni and Paolo Torroni, editors, Computational Logic in Multi-Agent Systems: 6th International Workshop, CLIMA VI, London, UK, June 27-29, 2005*, volume 3900 of LNCS, Springer, p. 143-164, 2006.

**BORDINI, H.R., DASTANI, M., DIX J., SEGHROUCHNI, A.E.F.** *Multi-agent Programming- Languages, Platforms, and Applications.* Springer, New York, 2005.

**BORDINI, R.H., VIEIRA, R.** Linguagens de programação orientadas a agentes: uma introdução baseada em AgentSpeak(L). *Revista de Informática Teórica e Aplicada*, Instituto de Informática da UFRGS, Brasil, X(1): p. 7-38, 2003.

**BORGES, A.P.** *Uma Arquitetura de Aprendizagem para a Condução Automática de Veículos.* 2009. 123 f. Dissertação (Mestrado) - Programa de Pós-Graduação em Informática Aplicada - Pontifícia Universidade Católica do Paraná. Curitiba, 2009.

**BRATMAN, M.E., ISRAEL D.J., POLLACK. M.E.** Plans and Resource-bounded Practical Reasoning. *Computer Intelligence*, 4: p.349 - 355, 1988.

**BRATMAN, M.E.** *Intention, Plans, and Practical Reason.* Harvard University Press, Cambridge, MA, 1987.

**BRINA, H. L.** *Estradas de Ferro 2.* Rio de Janeiro: LCT - Livros Técnicos e científicos Editora S.A., v. 2, 1982.

**BRINA, H. L.** *Estradas de Ferro 1.* Rio de Janeiro: LCT - Livros Técnicos e científicos Editora S.A., v. 1, 1979.

**CARVALHO, M. P.** *Curso de Estradas – Estudo, Projetos e Locação de Ferrovias e Rodovias.* Editora Científica. Rio de Janeiro, 1973.

**DENNETT, D. C.** *The Intentional Stance.* The MIT Press, Cambridge, MA, 1987.

**FERBER, J.** *Multi-Agents System: An Introduction to Distributed Artificial Intelligence.* Adison Wesley. New York, 1999.

**FOX, M., GEREVINI, A., LONG, D., SERINA, I.** Plan stability: Replanning versus Plan Repair. *Proceedings of the Sixteenth International Conference on Automated Planning and Scheduling (ICAPS)*, 2006.

**GEORGEFF, M., PELL, B., POLLACK, M., TAMBE, M., WOOLDRIDGE, M.** The Belief-Desire-Intention Model of Agency. In *Proceedings of the 5th International Workshop on Intelligent Agents, Theories, Architectures and Languages*, 1998.

**GEORGEFF, M. P., INGRAND, F. F.** Decision-making in an embedded reasoning system. In *Proceedings of the Eleventh International Joint Conference on Artificial Intelligence (IJCAI-89)*, Detroit, MI, p. 972-978, 1989.

**GEORGEFF, M. P., LANSKY, A. L.** Reactive reasoning and planning. In *Proceedings of the Sixth National Conference on Artificial Intelligence (AAAI-87)*, Seattle, WA, p. 677-682, 1986.

**GHALLAB, M., NAU, D., TRAVERSO, P.** *Automated Planning - Theory and Practice*. Morgan Kaufmann Publishers is an imprint of Elsevier, San Francisco, 2004.

**GIRAFFA, L.M.M.** *Uma Arquitetura de Tutor Utilizando Estados Mentais*. 177 f. Tese (Doutorado) - Programa de Pós-Graduação em Computação. Universidade Federal do Rio Grande do Sul, Porto Alegre, 1999.

**HÜBNER, J. F., BORDINI, R. H.** An overview of Jason. *Association for Logic Programming Newsletter*, 19(3), August 2006.

**HÜBNER, J. F., BORDINI, R. H., WOOLDRIDGE, M.** Plan patterns for declarative goals in agentspeak. In Stone, P., and Weiss, G., eds., *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multi-Agent Systems, Hakodate, Japan*, ACM Press. Short paper, p. 1291-1293, 2006.

**HÜBNER, J. F., BORDINI, R. H., VIEIRA R.** Introdução ao Desenvolvimento de Sistemas Multiagentes com Jason. In *Fernando Takashi Itakura et. al., editor, XII Escola de Informática da SBC - Paraná*, chapter 2, Ed. da UNICENTRO, Guarapuava-PR, p. 51-89, 2004.

**ILGHAMI, O.** *Documentation for JSHOP2*. University of Maryland, Technical Report, 2005.

**JENNINGS, N.R., SYCARA, K., WOOLDRIDGE, M.** "A Roadmap of Agent Research and Development". *Autonomous Agents and Multi-Agent Systems*, v.1, Kluwer Academic Publishers, p. 275-306, 1998.

**KARIM, S., HEINZE C.** Experiences with the Design and Implementation of an Agent based Autonomous UAV Controller. *Proceedings of the fourth international joint conference on Autonomous Agents and Multi-agent Systems (AAMAS'05)*, Utrecht, Netherlands, 2005.

**KINNY, D., GEORGEFF, M., RAO, A.** A methodology and modelling technique for systems of BDI agents. *Proceedings of the 7th European workshop on Modelling Autonomous Agents in a Multi-agent World: agents breaking away*, Eindhoven, Netherlands, p.56 -71, 1996.

**LEITE, A.R.** *Um Esquema para Redução do Consumo de Combustível em Sistemas de Condução Férrea Baseado em Otimização Distribuída de Restrição*. 2009. 79 f. Dissertação (Mestrado) - Programa de Pós-Graduação em Informática Aplicada - Pontifícia Universidade Católica do Paraná. Curitiba, 2009.

**LINCOLN N. K., VERES S. M., LOUISE, D., FISHER, M., LISITSA A.** An Agent Based Framework for Adaptive Control and Decision Making of Autonomous Vehicles. *In Proceedings Workshop Adaptation and Learning in Control and Signal Processing*. IFAC Workshop (ALCOSP' 2010), Turkey, 2010.

**LÜTZENBERGER, M., MASUCH, N., HIRSCH, B., AHRNDT, S., HEBLER, A., ALBAYRAK, S.** The BDI Driver in a Service City (Extended Abstract). *Proceedings of 10th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2011)*, Tumer, Yolum, Sonenberg and Stone (eds.), Taipei, Taiwan, p. 1257-1258, 2011.

**NAU, D. S., AU, T.-C., ILGHAMI, O., KUTER, U., MURDOCK, W., WU, D., YAMAN, F.** SHOP2: An HTN planning system. *Journal of Artificial Intelligence Research*, v. 20, p. 379 - 404, 2003.

**NAU, D. S., SMITH, S. J. J., EROL, K.** Control Strategies in HTN Planning: Theory Versus Practice. *Proceedings In AAAI-98/IAAI-98*, pages 1127-1133, 1998.

**NILSON, N.J., FIKES, R.E.** STRIPS: A New Approach to the Application of Theorem Proving To Problem Solving. *Artificial Intelligence*, V. 2, Elsevier Science Publishers Ltd., Essex, UK, p.189-208, 1971.

**MACHADO, R., BORDINI, R. H.** Running AgentSpeak(L) agents on SIM\_AGENT. *In John-Jules Meyer and Milind Tambe, editors, Intelligent Agents VIII – Proceedings of the Eighth International Workshop on Agent Theories, Architectures, and Languages (ATAL-2001)*, August 1-3, 2001, Seattle, WA, number 2333 in Lecture Notes in Artificial Intelligence, Springer-Verlag, Berlin p. 158-174, 2002.

**RAO, A.S.** AgentSpeak(L): BDI Agents Speak Out in a Logical Computable Language. In Van de Velde, W. e Perram, J., editors. *Proceedings of the Seventh Workshop on Modelling Autonomous Agents in a Multi-Agent World (MAAMAW'96)*, 22-25 January, Eindhoven, The Netherlands, number 1038 in Lecture Notes in Artificial Intelligence, London. Springer-Verlag, p. 42-55, 1996.

**RAO, A.S., GEORGEFF, M.P.** *Bdi Agents: From Theory to Practice*. Technical Note 56, Australian Artificial Intelligence Institute, Carlton, Victoria, 1995.

**RAO, A.S.** *Decision Procedures for Propositional Belief-Desire-Intention Logics*. Technical Note 44, Australian Artificial Intelligence Institute, Carlton, Victoria, 1993.

**RAO, A.S., GEORGEFF, M.P.** *Modelling Rational Agents within a BDI-Architecture*. Technical Note 14, Australian Artificial Intelligence Institute, Carlton, Victoria, 1991.

**SARDINA, S., SILVA, L., PADGHAM, L.** Hierarchical Planning in BDI Agent Programming Languages: A Formal Approach. *Proceedings in 6th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS'06)*, Hakodate, Hokkaido, Japan, 2006.

**SILVA, L., SARDINAM S., PADGHAM, L.** First Principles Planning in BDI Systems. *Proceedings in 8th International Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2009)*, Budapest, Hungary, 2009.

**WEISS, G.** Editor. *Multi-agent Systems - A Modern Approach to Distributed Artificial Intelligence*. MIT Press, Cambridge, MA, 1999.

**WOLFE, S. R., SIERHUIS, M.** To BDI, or Not to BDI: Design Choices in an Agent-Based Traffic Flow Management Simulation. *In Proceedings of the Spring Simulation Multiconference (SpringSim' 08)*, 2008.

**WOOLDRIDGE, M.** *An Introduction to Multi-Agent Systems*. John Wiley and Sons, 2009.

**WOOLDRIDGE, M., SCHUT, M., PARSONS, S.** The Theory and Practice of Intention Reconsideration. In *Journal of Experimental and Theoretical Artificial Intelligence*, 16(4): October-December, p. 261-293, 2004.

**WOOLDRIDGE, M.** *An Introduction to Multi-Agent Systems*. John Wiley and Sons, 2002.

**WOOLDRIDGE M.** *Reasoning about Rational Agents*. The MIT Press, Cambridge, MA, 2000.

**WOOLDRIDGE, M., JENNINGS, N. R., KINNY, D.** A Methodology for Agent-oriented Analysis and Design. *Proceedings of the third annual conference on Autonomous Agents, Seattle, Washington, United States*, April, p.69-76, 1999.

**WOOLDRIDGE, M.** *Intelligent agents*. In Weiss, G., editor, *Multi-agent Systems - A Modern Approach to Distributed Artificial Intelligence*, chapter 1, MIT Press, Cambridge, MA, p. 27 - 77, 1999.

**WOOLDRIDGE, M., JENNINGS, N. R.** Intelligent Agents: Theory and Practice. *The Knowledge Engineering Review*, 10:2, p. 115-152, 1995.