

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

MATHEUS STREISKY

**APLICAÇÃO DE ALGORITMO PARA RECUPERAÇÃO DE CASOS NO
PROBLEMA DE CONDUÇÃO DE TRENS DE CARGA**

TRABALHO DE CONCLUSÃO DE CURSO

PONTA GROSSA

2019

MATHEUS STREISKY

**APLICAÇÃO DE ALGORITMO PARA RECUPERAÇÃO DE CASOS
NO PROBLEMA DE CONDUÇÃO DE TRENS DE CARGA**

Trabalho de Conclusão de Curso apresentada como requisito parcial à obtenção do título de Bacharel em Ciência da Computação, do Departamento Acadêmico de Informática, da Universidade Tecnológica Federal do Paraná.

Orientador: Prof. Dr. André Pinz Borges

PONTA GROSSA

2019



Ministério da Educação
Universidade Tecnológica Federal do Paraná
Câmpus Ponta Grossa

Diretoria de Graduação e Educação Profissional
Departamento Acadêmico de Informática
Bacharelado em Ciência da Computação



TERMO DE APROVAÇÃO

APLICAÇÃO DE ALGORITMOS PARA RECUPERAÇÃO DE CASOS NO PROBLEMA DE CONDUÇÃO DE TRENS DE CARGA

por

MATHEUS STREISKY

Este Trabalho de Conclusão de Curso (TCC) foi apresentado em 12 de junho de 2019 como requisito parcial para a obtenção do título de Bacharel em Ciência da Computação. O candidato foi arguido pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

Prof. Dr. André Pinz Borges
Orientador

Prof. Dr. André Koscianski
Membro titular

Prof. Dr. Gleifer Vaz Alves
Membro titular

Prof. MSc. Geraldo Ranthum
Responsável pelo Trabalho de Conclusão
de Curso

Prof. MSc. Saulo Jorge Beltrão de Queiroz
Coordenador do curso

Dedico este trabalho à minha família, em especial a minha mãe Rosana Silva, quem nunca poupou esforços e sacrifícios para que eu continuasse tendo condições de estudar longe de casa durante todos estes anos e pudesse construir meu próprio futuro.

AGRADECIMENTOS

Durante todos estes anos que estou estudando, certamente passei por diversos altos e baixos durante essa fase de minha vida. Provavelmente não irei lembrar de citar todos que participaram e colaboraram, de forma positiva ou negativa, com o meu desenvolvimento pessoal durante o percorrer deste curso.

Primeiramente agradeço a minha família, por me incentivar e me dar todo o apoio emocional e financeiro necessário para que eu fosse capaz de sair de minha casa para morar sozinho e estudar em uma faculdade federal. Graças a isso fui capaz de aprender a caminhar com minhas próprias pernas e me preparar para a vida adulta e o mercado de trabalho.

Agradeço também aos meus amigos, os quais me proporcionaram diversos momentos de descontração fora das salas de aula, que me ajudam a liberar o estresse do dia a dia. Na faculdade tive a oportunidade de conhecer pessoas de diferentes regiões do Brasil com diferentes pensamentos e experiências de vida, os quais me tiraram “fora da caixa” e fizeram perceber o quão diversificado e diferente é nosso país.

Por fim, quero agradecer a todos os professores do Departamento de Informática que me passaram muito conhecimento da área, vários nos quais não via sentido no momento que estudava, mas hoje estando empregado posso perceber o quão importante foi para meu desenvolvimento profissional. Em especial ao meu orientador Prof. Dr. André P. Borges que me aceitou como orientando, e com muita paciência e atenção me ensinou muito sobre minha área de interesse, Inteligência Artificial.

O sucesso em criar inteligência artificial seria o maior evento na história da humanidade. Infelizmente, também pode ser o último, a menos que aprendamos a evitar seus riscos (HAWKING, Stephen, 2014).

RESUMO

STREISKY, Matheus. **Aplicação de algoritmo para recuperação de casos no problema de condução de trens de carga**. 2019. 100f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Universidade Tecnológica Federal do Paraná. Ponta Grossa. 2019.

Este trabalho tem como objetivo principal utilizar a técnica de Raciocínio Baseado em Casos (RBC) no intuito de auxiliar na condução de trens de carga. RBC é composto de etapas, entre elas a recuperação de casos similares à um problema, reuso de casos recuperados para gerar uma solução candidata à aplicada, revisão e retenção da solução aplicável. Neste trabalho o foco dar-se-á sobre a etapa de recuperação de casos, a qual utiliza uma base de casos, composta por casos e suas respectivas soluções, e algoritmos para selecionar casos que mais se assemelhem com um determinado problema. Os casos recuperados são utilizados e tratados pelas outras etapas do método RBC. No domínio e aplicação trabalhados, cada caso corresponde à um conjunto de características referentes ao deslocamento de um trem em uma via férrea. Com a implementação de algoritmos de recuperação, espera-se obter melhorias no desempenho da condução de trens, assim como diminuição de combustível gasto e conseqüentemente diminuição da poluição causada pela queima desse combustível. As melhorias poderão ser alcançadas uma vez que, ao se deparar com o problema de conduzir um trem em um trecho de via, a técnica de RBC irá recuperar um conjunto de ações possíveis a partir da base de casos disponível e o condutor poderá aplicar estas ações.

Palavras-chaves: Raciocínio Baseado em Casos. Recuperação de Casos. Condução de Trens de Carga.

ABSTRACT

STREISKY, Matheus. **Application of algorithms for cases retrieval in the problem of freight trains conduction**. 2019. 100p. Work of Conclusion Course (Graduation in Computer Science) - Federal Technology University - Paraná. Ponta Grossa, 2019.

This paper has as main objective to use the Case Based Reasoning technique in order to help in freight trains conduction. CBR is composed of steps, like similar cases retrieval, reuse of retrieved cases in order to generate a candidate solution to the applied one, review and retention of the applicable solution. In this work, the focus will be on the data retrieval step, which uses a case base composed by cases and their respective solutions, and algorithms to select the most similar cases to a determined problem. The retrieved cases are used and treated by the other steps of the CBR method. On the focused domain, each case corresponds to a set of traits that refer to a displacement of a train in a railway. With the implementation of the retrieval algorithm improvements are hoped in the train's conduction performance, just like the decrease of fuel expense and consequently a decrease in the pollution caused by it. The improvements can be reached because, once facing the train conduction problem in a part of a railway, the CBR technique will retrieve a set of possible actions from the available case base and the train conductor can apply these actions.

Keywords: Case-Based Reasoning, Case Recovery, Freight Train's Conduction.

LISTA DE FIGURAS

Figura 1 - Representação de um Caso.	15
Figura 2 - O ciclo RBC.	17
Figura 3 - Pseudocódigo da Recuperação de Dois Níveis.	20
Figura 4 - Exemplo de Recuperação de Dois Níveis.	21
Figura 5 - Pseudocódigo da Recuperação com Árvore K-D.	23
Figura 6 - Exemplo de Recuperação com Árvores K-D.	24
Figura 7 - Estrutura da via férrea.	34
Figura 8 - Aplicação dos pontos de aceleração de um plano solução S em um trecho de 403 m.	43
Figura 9 - Exemplo de um Caso representado computacionalmente.	44
Figura 10 - Diagrama do funcionamento do Método RBC.	45
Figura 11 - Diagrama do Ciclo RBC.	46
Figura 12 - Diagrama da Recuperação de Casos.	47
Figura 13 - Diagrama do Ciclo RBC da RDN.	48
Figura 14 - Diagrama da Recuperação de Casos em Dois Níveis.	49
Figura 15 - Diagrama da RDND.	53
Figura 16 - Aplicação do Limiar.	56
Figura 17 - Diagrama da Recuperação em Dois Níveis Dinâmica.	57
Figura 18 - Diagrama da Filtragem da Base de Casos.	57
Figura 19 - Diagrama da Comparação de Casos.	58
Figura 20 - Faixa de valores de e	65
Figura 21 - Faixa de valores de σ	66
Figura 22 - Gráfico da taxa de acertos do Cenário A.	68
Figura 23 - Gráfico da média de similaridade do Cenário A.	69
Figura 24 - Gráfico da média de LTKB do Cenário A.	69
Figura 25 - Gráfico da taxa de acertos do Cenário B.	71
Figura 26 - Gráfico da troca de critérios de filtragem do Cenário B.	71
Figura 27 - Gráfico da média de similaridade do Cenário B.	72
Figura 28 - Gráfico da média de LTKB do Cenário B.	72
Figura 29 - Gráfico da taxa de acertos do Cenário C.	74
Figura 30 - Gráfico da troca de critérios de filtragem do Cenário C.	74
Figura 31 - Gráfico da média de similaridade do Cenário C.	75
Figura 32 - Gráfico da média de LTKB do Cenário C.	75
Figura 33 - Comparação da taxa de acertos entre os cenários.	77
Figura 34 - Comparação da troca de critérios de filtragem entre os cenários.	78
Figura 35 - Comparação da média de similaridade entre os cenários.	80
Figura 36 - Comparação do LTKB entre os cenários.	81

LISTA DE TABELAS

Tabela 1 - Configuração dos trens utilizados nos experimentos.....	63
Tabela 2 - Cenários simulados nos experimentos.....	66
Tabela 3 - Redução de consumo obtido no Cenário A.....	70
Tabela 4 - Redução de consumo obtido no Cenário B.....	73
Tabela 5 - Redução de consumo obtido no Cenário C.....	76
Tabela 6 - Comparação do consumo obtido.....	81
Tabela 7 - Comparação da porcentagem de redução de consumo obtida.....	82

LISTA DE QUADROS

Quadro 1 - Exemplo de uma BC.....	50
Quadro 2 - Exemplo de uma BC Normalizada.....	50
Quadro 3 - Resultado da filtragem da RDN.....	51
Quadro 4 - Resultado dos cálculos de similaridade na BC filtrada.....	51
Quadro 5 - Resultado da filtragem utilizando o par de atributos {EAC, Perfil}.....	59
Quadro 6 - Resultado da Filtragem utilizando o par de atributos {IS, FS}.....	60
Quadro 7 - Resultado da Filtragem utilizando o par de atributos {MS, %R}.....	60

LISTA DE ABREVIATURAS

Árvore K-D Árvore K-Dimensional

LISTA DE SIGLAS

AMV	Aparelhos de Mudanças de Via
BC	Base de Casos
C	Caso
<i>cf</i>	CrITÉrios de Filtragem
IA	Inteligência Artificial
KNN	<i>K-Nearest Neighbor</i>
LTKB	Litros por Tonelada Bruta Transportada
PN	Passagem de Nível
RBC	Raciocínio Baseado em Casos
RDN	Recuperação em Dois Níveis
RDND	Recuperação em Dois Níveis Dinâmica
RI	Recuperação de Informações
SDR	Sistema de Diagnóstico Remoto

LISTA DE SÍMBOLOS

e	Parâmetro de Erro
σ	Limiar

SUMÁRIO

1.1 OBJETIVOS.....	7
1.2 JUSTIFICATIVA.....	8
1.3 ESCOPO.....	9
1.4 ORGANIZAÇÃO	10
2 RACIOCINIO BASEADO EM CASOS (RBC)	11
2.1 HISTÓRIA	11
2.2 O MÉTODO	13
2.3 CASOS	14
2.4 BASE DE CASOS	16
2.5 CICLO DE FUNCIONAMENTO DO RBC.....	17
2.5.1 Recuperação de Casos	18
2.5.2 Reutilização de Casos	27
2.5.3 Revisão de Casos.....	28
2.5.4 Retenção de Casos	29
2.6 VANTAGENS E LIMITAÇÕES DO RBC.....	30
2.7 CONSIDERAÇÕES FINAIS	31
3 CONDUÇÃO DE TRENS.....	32
3.1 CARACTERÍSTICAS DO TREM.....	32
3.2 CARACTERÍSTICAS DA FERROVIA	34
3.3 CICLO DE CONDUÇÃO	35
3.3.1 Aceleração	36
3.3.2 Frenagem	37
3.4 CONSIDERAÇÕES FINAIS	38
4 DESENVOLVIMENTO.....	39
4.1 APLICAÇÃO DO MÉTODO RBC	39
4.2 O CASO	41
4.3 A RECUPERAÇÃO DE CASOS	44
4.3.1 Recuperação em Dois Níveis (RDN)	47
4.3.2 Recuperação em Dois Níveis Dinâmica (RDND).....	52
4.4 CONSIDERAÇÕES FINAIS	61
5 EXPERIMENTOS E RESULTADOS	62
5.1 CENÁRIOS DE TESTE.....	62
5.1.1 Cenário A.....	67
5.1.2 Cenário B.....	70
5.1.3 Cenário C.....	73
5.2 COMPARAÇÃO DE RESULTADOS	76
5.2.1 Taxa de acertos	77
5.2.2 Troca do conjunto de critérios de filtragem	78

5.2.3 Similaridade	79
5.2.4 Consumo de Combustível.....	80
5.2.5 Redução do Consumo	81
5.3 CONSIDERAÇÕES FINAIS	82
6 CONCLUSÃO.....	84
6.1 TRABALHOS FUTUROS	86
7 REFERÊNCIAS	87

1 INTRODUÇÃO

Nesta pesquisa, o problema estudado é o transporte ferroviário. Este método de transporte teve início entre o século XVIII e XIX com o desenvolvimento do motor a vapor, sendo usado para o transporte do minério extraído das minas de carvão, um componente importante durante a revolução industrial. Com o avanço da tecnologia o modelo ferroviário foi aprimorado, deixando de ser um transporte lento e limitado para transportar passageiros e outros tipos de mercadorias. Comboios elétricos foram lançados e o motor a vapor foi substituído pelo diesel, uma melhoria que impulsionou o setor (SILVA, 2017).

O transporte ferroviário foi importante para a criação de novas indústrias e categorias profissionais, tendo impacto no desenvolvimento socioeconômico do mundo. Na colonização da América do Norte, por exemplo, este tipo de transporte ajudou a desbravar o território do oeste americano e durante as duas grandes guerras serviu para a movimentação de homens, mantimentos e armas (REBELO, 2010).

Dada sua importância e impacto no mundo, o modelo ferroviário permanece sendo utilizado, segundo Neto (2011) em países desenvolvidos o transporte ferroviário de passageiros aumenta a cada dia chegando a concorrer com os aviões, pois o tempo de espera mais o tempo do voo, fazem com que a opção ferroviária seja vantajosa. A capacidade de transportar grandes quantidades de carga por longos trechos de terra também passou a ser utilizada, sendo uma opção rentável em virtude da quantidade de produtos transportados.

Porém, este modal apresenta um elevado consumo de combustível durante as viagens, o que ocasiona problemas, como impactos ambientais causados pela queima de combustível fóssil. Diante disso, estuda-se a possibilidade de empregar algoritmos de Inteligência Artificial, como o método de Raciocínio Baseado em Casos (RBC), para auxiliar na condução de trens. O auxílio dá-se na busca e recuperação das melhores ações a serem empregadas para conduzir um trem em um trecho de via previamente mapeado e conhecido.

Raciocínio Baseado em Casos (RBC) é um método de Inteligência Artificial que visa reutilizar soluções já registradas em uma memória, para solucionar novos problemas, utilizando conhecimento prévio adquirido em problemas passados para

reaproveitar soluções bem-sucedidas e evitar que falhas se repitam (Aamodt e Plaza, 1994).

No modal férreo, a escolha das ações que o maquinista deverá executar ao longo da viagem é uma tarefa que demanda experiência e conhecimento do trem conduzido e da via percorrida. Uma escolha errada pode, por exemplo, ocasionar a parada de um trem e, conseqüentemente, aumento no consumo de combustível em função da retomada (etapa onde ocorre uma elevada queima de combustível). Neste cenário, a recuperação e reaproveitamento de ações bem-sucedidas pode evitar que falhas ocorram e também resultar em economia de combustível.

Para tanto, este trabalho objetiva aplicar e analisar diferentes algoritmos de recuperação de casos e avaliar o impacto de cada algoritmo na escolha de quais ações deverão ser executadas durante uma viagem de trem. As ações serão escolhidas de acordo com o trecho da estrada férrea, como reta, subida ou descida, além de considerar demais características dos trens (e.g. velocidade, quantidade de locomotivas e vagões, etc.), para selecionar a melhor decisão possível e aplicá-la durante a viagem. Uma ação é considerada eficiente quando: (i) consome o mínimo de combustível possível ao longo da viagem; (ii) não causa danos à via ou ao trem e (iii) são necessárias poucas alterações na ação sugerida. Os itens (i) e (ii) serão usados como métricas de avaliação da eficiência dos algoritmos em relação ao domínio de aplicação, enquanto (iii) será adotado como critério de comparação das técnicas em um foco computacional.

1.1 OBJETIVOS

O objetivo geral deste trabalho é auxiliar a condução de trens através de uma condução assistida, utilizando método RBC para fornecer sugestões de ações a serem tomadas pelos maquinistas durante uma viagem de trem. Através do método RBC, focando-se na etapa de Recuperação, procura-se implementar e aplicar diferentes técnicas de recuperação, para posteriormente analisar a sua eficiência na condução de trens, examinando principalmente o quesito da redução de consumo de combustível.

Para alcançar o objetivo geral, são necessários atingir os seguintes objetivos específicos:

- Pesquisar na literatura, fundamentos sobre o método RBC, assim como sobre a etapa da recuperação de casos e os elementos que a compõem;
- Pesquisar algoritmos de recuperação de casos, e escolher um ou algoritmos aplicáveis no problema;
- Criar um novo algoritmo de recuperação, se baseando em um algoritmo já existente.
- Analisar os algoritmos selecionados e implementá-los no problema;
- Analisar os resultados obtidos após a aplicação dos algoritmos e confrontá-los com Borges (2015a) em termos de consumo de combustível e eficiência na etapa de recuperação (i.e., quantidade de modificações feitas pela etapa de revisão no caso recuperado e similaridades alcançadas).
- Analisar os resultados obtidos pelo algoritmo de recuperação criado, e compará-los com o seu precursor.

1.2 JUSTIFICATIVA

O método de Raciocínio Baseado em Casos em Inteligência Artificial é um tópico extenso e detalhado e tem sido utilizado em diversas pesquisas ao longo dos últimos anos. Pode-se citar o trabalho de Plotegher (2005) que descreve um Sistema de Diagnóstico Remoto (SDR) para ambiente industrial aplicado em máquinas-ferramenta (MF). Combinando o SDR com o uso do método RBC, objetiva reduzir os tempos e custos de manutenção das máquinas. Na área da medicina, SILVA, R. P. (2005) realizou pesquisas que auxiliam diagnóstico médico utilizando RBC. Devido à complexidade de tal diagnóstico, que exige intuição, raciocínio, experiência e a análise de informações de fontes distintas, o método RBC é uma ferramenta promissora nesta área, pois retém casos de pacientes tratados e recupera esses casos para ajudar outros pacientes.

Neste trabalho o foco será na Recuperação de Casos como objeto de estudo e implementação de um algoritmo a ser escolhido, procurando mostrar sua importância no método RBC em uma aplicação com características reais. A aplicação dá-se no problema de condução de trens de carga, onde RBC é usado como uma

técnica capaz de recuperar ações que maquinistas devem executar para percorrer determinados trechos de via, conforme descrito em Borges (2015a).

A técnica mostrou-se eficaz pois o compartilhamento das experiências dos maquinistas para conduzir um trem necessita de experiência e conhecimento, não é algo fácil de ser dividido e explicado. Logo, a partir de um conjunto de dados obtidos por sensores no interior de locomotivas, é gerado uma base de casos com dados e informações derivadas de cálculos matemáticos e cruzamentos com informações reais. Ao longo da viagem, ao deparar-se com uma situação nova, um conjunto de informações é capturado, processado e submetido à base de casos gerada, a qual irá retornar uma ou mais ações aplicáveis que julga eficiente.

Espera-se que a melhora na etapa de recuperação de casos reduza os esforços empregados na etapa de reuso, uma vez que ao recuperar casos muito próximos à solução, as modificações a serem realizadas serão em menor quantidade.

Assim, com a melhoria no processo de condução, espera-se que traga benefícios nas áreas econômica (diminuição do gasto com combustível), ambiental (diminuição da emissão de gases poluentes na atmosfera) e científicas (colaboração com a comunidade pela divulgação dos resultados obtidos através do algoritmo a ser escolhido e implementado).

1.3 ESCOPO

Este trabalho é continuação das pesquisas elaboradas por Borges (2015a), tais como Borges (2009a, b), Borges (2011a, b), Borges (2012), Borges (2014), Borges (2015b, c, d), Dordal (2011a, b) e Dordal (2013), em que a técnica de RBC foi empregada na elaboração de um simulador de condução de trens de carga. A pesquisa englobou as 4 etapas de RBC no domínio do modal férreo, onde foram empregadas as técnicas como *KNN* para recuperação de casos, Algoritmo Genético para reuso, conceitos inerentes ao problema na etapa de revisão dos casos e retenção em arquivos *csv*.

Devido ao tamanho do problema e complexidade para compreensão do modal, neste trabalho foi escolhido o somente a etapa de Recuperação de Casos para realizar o estudo e implementação, procurando mostrar sua importância no método RBC. As 3 demais etapas que compõem o RBC serão empregadas conforme Borges

(2015a), pois cada uma das etapas é complexa e possui a abordagem de diferentes técnicas de computação, havendo o potencial de uma pesquisa focada para cada etapa, tratando seu funcionamento e técnicas que possam ser utilizadas.

O problema alvo é o modelo ferroviário, o qual busca-se auxiliar na condução de trens com o método RBC para buscar a melhor forma de condução. Demais modelos, como o rodoviário por exemplo, não serão abordados nesta pesquisa. O modelo ferroviário por si é um modelo complexo e envolve infinitas possibilidades de ações de acordo com a situação enfrentada, assim para melhor desempenho do método um único domínio foi escolhido. Mas é possível alterar o domínio da aplicação se for alterado a base de casos. Ademais, o auxílio irá sugerir ações e não as aplicá-las diretamente no computador de bordo das locomotivas, os quais não temos acesso.

1.4 ORGANIZAÇÃO

O presente trabalho está organizado na seguinte forma. O Capítulo 2 descreve o método raciocínio baseado em casos, citando brevemente sua história de origem e os componentes que compõe o método. Também é descrito o funcionamento do método, mostrando com mais detalhes os objetivos de cada uma de suas etapas. Em seguida o Capítulo 3 irá descrever as características do ambiente alvo deste trabalho, a condução de trens. Este capítulo apresentará característica dos trens e das vias férreas, explicando o funcionamento do processo de condução. Posteriormente, no Capítulo 4 será apresentado o desenvolvimento do trabalho, descrevendo as técnicas de recuperação que serão utilizadas, além de explicar com detalhes o funcionamento de tais técnicas escolhidas e como elas impactam o sistema RBC. O Capítulo 5 irá demonstrar os resultados obtidos pelas técnicas implementadas, analisando os dados de acordo com a similaridade dos casos, o consumo de combustível obtido e a redução de consumo adquirido. Por fim no Capítulo 6 serão apresentadas as conclusões finais e os trabalhos futuros.

2 RACIOCÍNIO BASEADO EM CASOS (RBC)

O Raciocínio Baseado em Casos (RBC) é um método para solução de problemas e aprendizado baseado em experiências passadas. O RBC resolve problemas recuperando e adaptando experiências (casos) que estejam armazenadas em uma base de casos, e salva a solução destes problemas para usar posteriormente.

Este capítulo irá apresentar um breve resumo sobre o surgimento do método, visando abordar o que é RBC, o seu funcionamento e de cada uma de suas etapas. Também será explicado o que são os casos, como representá-los, tratá-los e armazená-los. Por fim serão discutidas as vantagens e limitações do método, comparando-o a outras técnicas usualmente utilizadas em IA.

2.1 HISTÓRIA

O surgimento do método de RBC se teve com o trabalho de Shank e Abelson (1977) sobre Memória Dinâmica. O objetivo foi a criação de modelos cognitivos para solução de problemas e aprendizado com base em memórias episódicas (experiências que ocorreram com algo ou alguém em determinado momento e local). Por exemplo, um indivíduo se lembra de sua festa de aniversário de 10 anos, isto é uma memória episódica. Segundo Shank e Abelson (1977), o conhecimento humano é gravado em memória na forma de roteiros (casos), usados para obtenção de resultados relacionando a causa do problema e a ação a ser tomada para solucionar o problema (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003).

Os roteiros foram sugeridos na década de 1980 como uma forma de representar e organizar dados em uma memória, descrevendo as informações a respeito de um evento específico. Por exemplo, um médico ao atender um paciente, pode-se listar um conjunto de ações que são esperados deste encontro, como: aguardar paciente, detectar sintomas físicos e/ou emocionais, recolher, analisar e interpretar os sintomas e por fim diagnosticar o paciente e lhe receitar um medicamento. A partir do roteiro de ações montado, a atividade de consulta médica se torna previsível, com as ações acontecendo quase sempre da mesma maneira. A hipótese cognitiva dos roteiros é que humanos armazenam conjunto de ações em

blocos, e lembramos deles individualmente de acordo com que seja necessário (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003).

O RBC também foi influenciado pelas pesquisas de Raciocínio por Analogia de Gick e Holyoak (1980), podendo considerar o RBC como uma especialização deste método. O raciocínio analógico tem como objetivo transformar o conhecimento de um domínio conhecido, para outro domínio, parcial ou totalmente incompreendido. Por exemplo, analisar como um peixe utiliza as barbatanas para nadar e a partir disto criar um remo para mover um barco sobre a água. No RBC não ocorre essa mudança de domínio, os problemas são resolvidos exclusivamente para o mesmo domínio da aplicação. Por exemplo: para solucionar problemas com notebooks que não ligam são usados conhecimentos sobre computadores que não ligam (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003).

As áreas da filosofia e psicologia também contribuíram para o desenvolvimento do RBC com as pesquisas da Teorias da Formação de Conceitos, da Resolução de Problemas e da Aprendizagem Experimental (TULVING, 1977; SMITH e ADAMS, 1978; SMITH e MEDIN, 1981). Roger Schank pesquisou a importância que a memória de situações passadas e os padrões que essas memórias estão organizadas, tem na resolução e aprendizado de problemas. Esta teoria supõe que a compreensão, o aprendizado e a abstração de situações são processos mentais intimamente ligados (apud VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003).

Ao passar dos anos, ocorreram diversas tentativas de desenvolvimento de sistemas RBC. O primeiro sistema raciocinador baseado em casos foi o chamado CYRUS, desenvolvido por Janet Kolodner (KOLODNER, 1983), que tinha como objetivo encontrar uma situação diplomática semelhante à solução proposta por Cyrus Vance (ex-secretário de estado dos Estados Unidos). Alguns exemplos de sistemas, entre diversos existentes, que foram desenvolvidos e incrementaram com seus resultados para o RBC foram: o JULIA (HINRICHS, 1992) que planeja menus de refeições e introduziu o tratamento de restrições na solução dos problemas e o PROTOS (PORTER e BAREISS, 1986) que abordou a utilização de Aprendizado de Máquina para tarefas de classificação.

2.2 O MÉTODO

Raciocínio Baseado em Casos (RBC) é um método de Inteligência Artificial (IA) que consiste em utilizar situações já registradas em memória para solucionar novos problemas, partindo da hipótese que problemas similares possuem soluções similares. As situações são registradas no formato de casos que, em teoria, são todos os atos realizados por um algo ou alguém e guardados em memória. O uso destes casos faz com que o conhecimento prévio adquirido em problemas passados seja reutilizado, reaproveitando soluções bem-sucedidas e evitando que falhas se repitam (RIESBECK, 1989).

Em IA, o RBC é uma tecnologia inspirada no modelo de cognição e comportamento humano. Esta tecnologia procura modelar o raciocínio e os pensamentos humanos na tentativa de construir e melhorar sistemas computacionais inteligentes.

O RBC também propõe uma forma de resolução de problemas que difere da forma fundamental que é utilizada na IA. Em vez de se basear somente em um agregado generalizado de informações ou realizar associações entre os problemas e possíveis soluções, o RBC utiliza o conhecimento específico, os casos presentes na base de casos possuem experiências adquiridas no passado sobre um determinado domínio.

Um sistema RBC é formado através da junção dos seguintes elementos:

- **Representação do Conhecimento:** o conhecimento é representado na forma de casos que descrevem experiências passadas que tenham sido solucionadas ou não.
- **Medida de Similaridade:** para encontrar um caso na base de casos que possa solucionar o problema enfrentado, o sistema deve ser capaz de determinar qual caso é similar para cada problema encontrado.
- **Adaptação:** é comum que um caso recuperado não seja completamente igual ao problema atual devido, por exemplo, às mudanças nas características do ambiente onde é executado. Assim, o RBC adapta o caso encontrado e verifica se pode ser utilizado no problema antes de ser efetivamente aplicado.

- **Aprendizado:** As informações contidas na base de casos, conforme o tempo passe podem se tornar obsoletas, assim para se manter atualizado e evoluir constantemente, o RBC salva todas as soluções que ele resolver e as utiliza para solucionar casos futuros.

Apresentado a filosofia por trás do RBC e os elementos que a compõe, as próximas seções irão abordar como o método representa e organiza o conhecimento que possui e como este conhecimento é armazenado para ser utilizado futuramente. Em seguida será explicado o funcionamento do ciclo RBC, mostrando como cada etapa do método manuseia os casos de forma que seja possível resolver problemas e aprender com eles.

2.3 CASOS

A principal forma de descrever o conhecimento de um sistema RBC são os casos. Caso é todo o conhecimento armazenado em memória pelo método e representa um episódio em que um problema foi total ou parcialmente resolvido (KOLODNER, 1993). Os casos podem ter diferentes formas e tamanhos, contendo diferentes conteúdos e representações dependendo do domínio e do objetivo da aplicação, mas são geralmente representados por uma combinação da descrição do problema com sua respectiva solução. Casos podem representar, por exemplo:

- Os sintomas de um paciente e a aplicação de um tratamento médico.
- A descrição de um defeito em um computador e os procedimentos para consertá-lo.
- Os motivos pelo qual um carro não dá partida e as ações necessárias para ligá-lo.

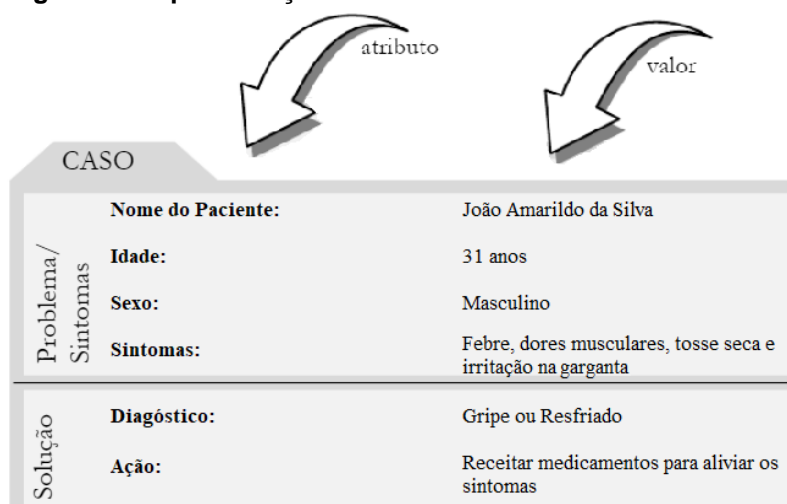
Um caso *a priori* contém informações concretas e experiências de uma situação específica. Porém, um caso também pode conter outros itens, como os efeitos da aplicação da solução, o motivo pelo qual aquela solução foi adotada, ou conter dados administrativos como o número do caso ou data de criação, por exemplo.

A forma mais simples e comumente usada de representação do conhecimento (casos) do RBC é através de vetores atributo-valor. Esta forma resolve grande parcela dos problemas de aplicação de RBC. A utilização de grafos também pode ser

empregada, assim como em situações em que a complexidade dos casos é maior, representação orientado a objetos pode ser adotada.

Na representação de atributo-valor, um item é representado por um par atributo-valor. Por exemplo: nome do paciente, idade, sexo e sintomas (atributos) e “João Amarildo da Silva”, “31 anos”, “Masculino” e “Febre, dores musculares, tosse seca e irritação na garganta” (valores). Então, podemos descrever um caso como junção da descrição do problema em conjunto com sua solução, ambos representados por um conjunto de pares atributo-valor, como mostrado na Figura 1. Conjuntos de atributos podem ser fixos para todos os casos, ou pode variar entre os casos individualmente.

Figura 1 - Representação de um Caso.



Fonte: Adaptado de VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003.

Cada atributo do caso é associado a um tipo de dado, onde os tipos básicos geralmente são utilizados: números (real ou inteiro), booleano, data e *string*. Tais dados irão conter informações que representem características relevantes dos objetos e irão moldar o comportamento deles.

A representação de casos por meio de pares atributo-valor possui várias vantagens:

- A representação e implementação é fácil e simples;
- Facilita a execução de medidas de similaridade eficientes;
- Possui fácil armazenamento (por exemplo, em banco de dados);

- É possível realizar uma recuperação eficiente.

No entanto, sua principal desvantagem está em não ser possível representar nenhum tipo de informação estrutural ou relacional, sendo recomendada para representar tarefas de diagnóstico que lidam com grandes bases de casos.

2.4 BASE DE CASOS

Para serem utilizados, os casos são armazenados em uma Base de Casos (BC). Geralmente a BC é composta de experiências positivas, formadas por soluções que possam contribuir com a resolução de problemas futuros. Experiências negativas também compõem a base de casos, representando tentativas frustradas de solucionar um problema e tem como objetivo indicar possíveis erros e prevenir a repetição de erros passados (KOLODNER, 1993).

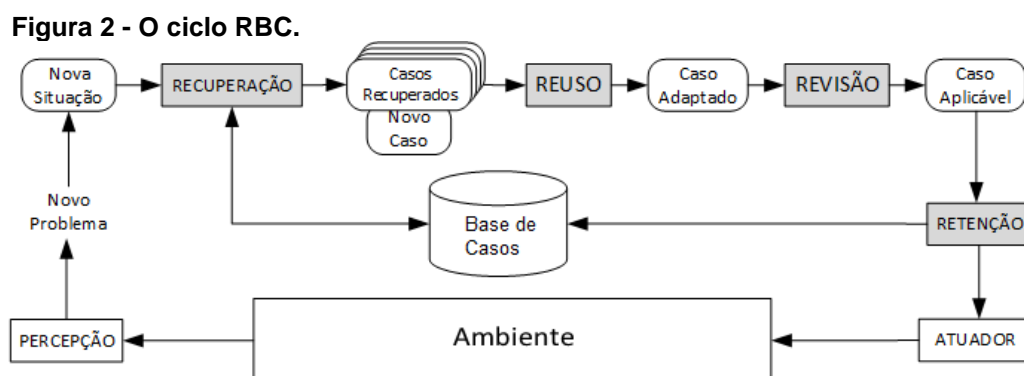
Para o sistema RBC armazenar conhecimento, existem quatro tipos diferentes de repositórios de armazenamento, denominados Repositórios de Conhecimento (RICHTER, 1995), que são:

- **Vocabulário:** é utilizado para descrever o conhecimento geral do domínio e é utilizado em diversos estágios durante o processo do RBC, sendo composto de um vocabulário repleto de termos comuns e/ou técnicos relacionados ao domínio da aplicação;
- **Casos:** são as experiências passadas armazenadas em uma base de casos (como descrito anteriormente);
- **Identificação:** é o conhecimento sobre como os casos podem ser úteis para o problema atual. Esta utilidade é determinada pelo cálculo de similaridade entre as características do problema atual e os casos armazenados na base, utilizando uma medida de similaridade (por exemplo a Distância Euclidiana);
- **Adaptação:** é o conhecimento de como modificar os casos recuperados com o objetivo de satisfazer completamente ao problema atual.

Dependendo do objetivo final da aplicação RBC, a representação do conhecimento pode variar de um repositório de conhecimento para outro.

2.5 CICLO DE FUNCIONAMENTO DO RBC

A representação mais comum do método RBC é o proposto por Aamodt e Plaza (1994), que descreve um ciclo de raciocínio para solução de problemas. O ciclo é composto por 4 etapas principais: Recuperação, Reuso, Revisão e Retenção de casos (cf. Figura 2).



Fonte: Borges (2015a).

Dado um novo problema a partir da percepção de um ambiente, para encontrar uma nova solução é realizada uma consulta na base de casos (composta de vários casos). A etapa de Recuperação busca um ou mais casos semelhantes ao novo problema (geralmente os casos mais semelhantes são os melhores para o mesmo problema por apresentarem soluções consideradas satisfatórias), formando um conjunto de Casos Recuperados.

Quando for necessário, é realizada uma adaptação (reuso) no caso escolhido para tentar solucionar o problema, visto que cada caso recuperado pode não ser completamente igual ao novo problema. Após a adaptação, o caso modificado deve ser revisado para verificar se este pode ser aplicado e, se resolver o problema de forma adequada, o caso é aplicado no ambiente através de atuadores e retido na base de casos. A retenção dá-se para ajudar na resolução de problemas futuros, armazenando as soluções obtidas. Se a solução adaptada não for aplicável, o caso deve ser novamente adaptado, modificando-o na etapa de reuso (KOLODNER, 1993).

Apresentado o ciclo básico do funcionamento do RBC, as próximas seções apresentarão com um pouco mais de detalhes sobre as etapas deste ciclo, abordando a importância de cada etapa para o método RBC.

2.5.1 Recuperação de Casos

O objetivo da recuperação é encontrar um caso ou um conjunto de casos na BC que contenha uma solução útil para o problema atual. Para realizar essa recuperação é necessário casar a descrição do problema alvo com as descrições dos problemas já solucionados presentes na BC, aplicando uma medida de similaridade (por exemplo, Distância Euclidiana) (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003).

O processo de recuperação de casos pode ser descrito por um conjunto de tarefas, estas são:

- **Avaliação da situação:** é a tarefa mais complexa, pois consiste em representar os problemas do mundo real de forma que o sistema RBC entenda e procure a melhor solução. Os problemas do mundo real são complexos e apresentam diversas características, podendo ocorrer problemas que não são igualmente representados na BC. Devido a isto, a recuperação procura encontrar casos potencialmente úteis para solucionar o problema atual, utilizando como entrada do processo de recuperação descritores relevantes do problema, chamados *descritores de entrada*, estes determinados pelo domínio da aplicação. A identificação de quais descritores serão relevantes é um dos maiores fatores de complexidade de um sistema RBC, quanto menos descritores forem necessários para descrever o problema, mais eficiente o sistema será. Porém para compreender um problema, saber apenas seus descritores de entrada não é o suficiente, é necessário tomar cuidados como: filtrar ruídos não relacionados ao problema, checar se os valores são coerentes, etc. Descritores desconhecidos podem ser ignorados ou pode-se solicitar que o usuário que os explique.
- **Casamento:** tem por objetivo obter um caso ou um conjunto de casos da BC úteis para o problema atual. Os descritores de entrada são usados como índices de uma consulta, que irá procurar não somente casos idênticos, mas também de casos potencialmente úteis, que correspondam parcialmente à consulta. O casamento associa os descritores do problema

atual aos descritores dos casos da BC, comparando-os através de uma medida de similaridade. Os algoritmos de casamento podem ser computacionalmente caros, pois requerem a combinação da busca e comparação de casos. Diversas técnicas podem ser aplicadas, variando de acordo com a estrutura da BC e a medida de similaridade adotada.

- **Seleção:** dado um conjunto de casos similares, é realizada a escolha do melhor caso, onde geralmente esta escolha é determinada pelo grau de casamento (grau de similaridade). Quando um casamento é considerado forte ele sofrerá uma tentativa de aplicação, se falhar o caso não é aplicado e pode ocorrer uma nova tentativa com outro caso do conjunto. Os casos geralmente são classificados por alguma métrica ou critério de ordenação. Métodos de seleção geralmente realizam ordenação com base no grau de similaridade, mas também é possível utilizar pesos nos atributos dos casos, que podem mudar completamente a ordem do conjunto de casos recuperados.

Para realizar a recuperação podemos utilizar diversos algoritmos, desde técnicas simples como a Busca Sequencial, até técnicas mais complexas como Redes de Recuperação de Casos. A seguir será apresentado o funcionamento de alguns algoritmos de recuperação existentes.

2.5.1.1 Recuperação de Dois Níveis

O princípio básico deste algoritmo é reduzir o máximo possível o número de comparações de casos, aplicando filtros e limitações com objetivo de determinar se os casos são úteis ou não para problema atual. A aplicação dessas heurísticas é efetuada antes do processo de medida de similaridade, utilizando comparações simples para determinar quais casos com certeza não são úteis, e descartá-los. Os casos marcados como úteis são sujeitos a comparações mais complexas, onde então é aplicado a medida de similaridade (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003). A execução destas tarefas se dá por um processo de dois níveis:

- **Pré-seleção dos casos candidatos:** o primeiro passo é determinar entre todos os casos da BC, quais são os possíveis candidatos por meio de comparações simples e rápidas.
- **Ordenação dos candidatos de acordo com grau de similaridade:** o segundo passo é escolher entre os pré-selecionados os melhores candidatos. Nesta etapa, a execução das comparações é efetuada de forma sequencial, realizando tarefas complexas para determinar a similaridade dos candidatos. Efetuadas as comparações, é realizada uma reordenação entre os casos candidatos de acordo com sua similaridade.

A principal vantagem deste método é fornecer uma melhora considerável no desempenho da recuperação, se o número de pré-selecionados seja muito menor que o total de candidatos ao final da primeira etapa, isto reduz o custo computacional para a aplicação das medidas de similaridade na etapa seguinte. Em contrapartida, o método possui a desvantagem de ser sujeito a erros, havendo dois tipos possíveis de erros: (i) um caso potencialmente útil não ser incluído entre os pré-selecionados; (ii) um caso que não seja potencialmente útil incluído entre os pré-selecionados. Outra desvantagem está em não haver garantia da redução de candidatos, podendo acarretar nenhuma melhora ou até mesmo piora de desempenho (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003). O pseudocódigo do algoritmo é mostrado na Figura 3.

Figura 3 - Pseudocódigo da Recuperação de Dois Níveis.

```
Algoritmo de recuperação
```

```

FUNCAO RestringeBase (CaseBase,Consulta,m): ConjCasos
INICIO
    ...
    RETORNE (ConjCasos)
FIM

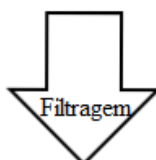
FUNCAO Selec2N (CaseBase,Consulta,m): ListaCasoSim
INICIO
    RETORNE ( SelecSeq(RestringeBase(CaseBase,
                          Consulta,m),Consulta,m) )
FIM
  
```

Fonte: VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003.

Por exemplo, o problema de um médico ao atender um paciente, utilizando uma BC com 10 elementos. A filtragem é realizada comparando os sintomas dos pacientes, se houver dois ou mais sintomas similares o caso é considerado útil para o problema. Dado o caso {João Amarildo da Silva; 31; Masculino; Febre, Dores musculares, Tosse seca, Irritação na garganta}, para descobrir a doença do paciente, é efetuada a filtragem (cf. Figura 4).

Figura 4 - Exemplo de Recuperação de Dois Níveis

BC					
Caso	Nome	Idade	Sexo	Sintomas	Doença
1	Eduarda Ribeiro	26	F	Dores musculares, Cansaço, Enjoo, Vômito	Hepatite
2	Maria Luiza de Souza	15	F	Cansaço, Pele pálida, Boca seca	Anemia
3	José Carlos Monteiro	36	M	Febre, Tosse seca, Dificuldade em dormir	Gripe
4	Renata Pereira	37	F	Calafrios, Febre, Falta de apetite	Gripe
5	Daniela Maria	68	F	Perda de peso, Cansaço, Irritabilidade	Diabetes
6	Paulo César	52	M	Dores musculares, Falta de apetite, Tosse seca	Gripe
7	Osmar Amaral	78	M	Irritabilidade, Boca seca, Cansaço	Diabetes
8	Alexandra Oliveira da Silva	44	F	Enjoo, Dor de cabeça, Sonolência,	Pressão alta
9	Paulo César Batista	7	M	Febre, Falta de apetite, Manchas vermelhas	Catapora
10	Beatriz Pinheiro Guimarães	22	F	Falta de apetite, Inchaço no rosto, Febre	Caxumba



BC					
Caso	Nome	Idade	Sexo	Sintomas	Doença
3	José Carlos Monteiro	36	M	Febre, Tosse seca, Dificuldade em dormir	Gripe
6	Paulo César	52	M	Dores musculares, Falta de apetite, Tosse seca	Gripe

Fonte: Autoria própria.

Após a filtragem, são feitas novas comparações (como idade e sexo neste exemplo) para determinar o caso mais similar entre os casos pré-selecionados. O caso mais similar para o exemplo é o caso 3 {José Carlos Monteiro; 36; M; Febre, Tosse seca, Dificuldade em dormir; Gripe}, determinando a doença para o paciente João Amarildo da Silva como “Gripe”.

2.5.1.2 Recuperação com Árvores K-D

O principal objetivo em utilizar árvores como estrutura de recuperação está em organizar a busca, utilizando a estrutura pré-montada da árvore para otimizar o processo de medição da similaridade. Uma árvore *k-d* é uma árvore binária *k*-dimensional que organiza dados de acordo com *k* chaves, diferente da árvore binária comum que organiza as informações em apenas uma chave. As chaves são distribuídas para cada um dos níveis da árvore, e o número de chaves é igual ao número de atributos dos casos (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003).

Devido a essa organização, cada chave (ou nodo) da árvore contém um subconjunto de casos da BC, com a raiz representando toda a base de casos. A divisão da BC é realizada de acordo com o valor do atributo (*a*) (este atributo deve ser o que melhor divide os casos em dois subconjuntos diferentes e de tamanho igual) multiplicado pelo nível que a árvore se encontra (*n*) mais um, representado pela fórmula $a(n + 1)$. A divisão dos casos deve ser realizada até que sobre apenas um conjunto com um tamanho máximo de casos, definido anteriormente. Para a árvore *k-d* ter uma recuperação eficiente e rápida, a escolha do atributo de partição para cada nível da árvore é de extrema importância, pois irá definir se os conjuntos de casos serão corretamente distribuídos (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003). É importante frisar que os atributos escolhidos para a divisão dos casos não podem se repetir em diferentes níveis da árvore.

A pesquisa de casos similares neste tipo de árvore é realizada de forma recursiva, como uma pesquisa em árvore binária, buscando possíveis candidatos entre as chaves da árvore. A busca é realizada inicialmente através de um cálculo de similaridade global das chaves, procurando informar se há números razoáveis de candidatos úteis dentro dela. Se sim, o processo avança para o próximo estágio, que calcula a similaridade para todos os casos presentes no repositório, selecionando os mais similares. Do contrário a busca passa para o próximo nodo, dando continuidade ao processo de busca de candidatos úteis. O pseudocódigo do algoritmo é mostrado na Figura 5.

Figura 5 - Pseudocódigo da Recuperação com Árvore K-D.

```

ALGORITMO recupere(K: árvore-KD)
INÍCIO
  SE K é folha ENTÃO
    PARA cada caso F de K FAÇA
      SE sim(Q,F) > scq[m] ENTÃO
        insira F em scq
  SENÃO (* nodo interno*)
    SE  $A_i$  é o atributo e  $v_i$  o valor que rotulam K
      SE  $Q[A_i] \leq v_i$  ENTÃO
        recupere( $K_{\leq}$ )
        SE Teste-EIL é verdade ENTÃO
          recupere( $K_{>}$ )
      SENÃO
        recupere( $K_{>}$ )
        SE Teste-EIL é verdade ENTÃO
          recupere( $K_{\leq}$ )
    SE Teste-EDL é verdade ENTÃO
      termine recuperação com scq
FIM

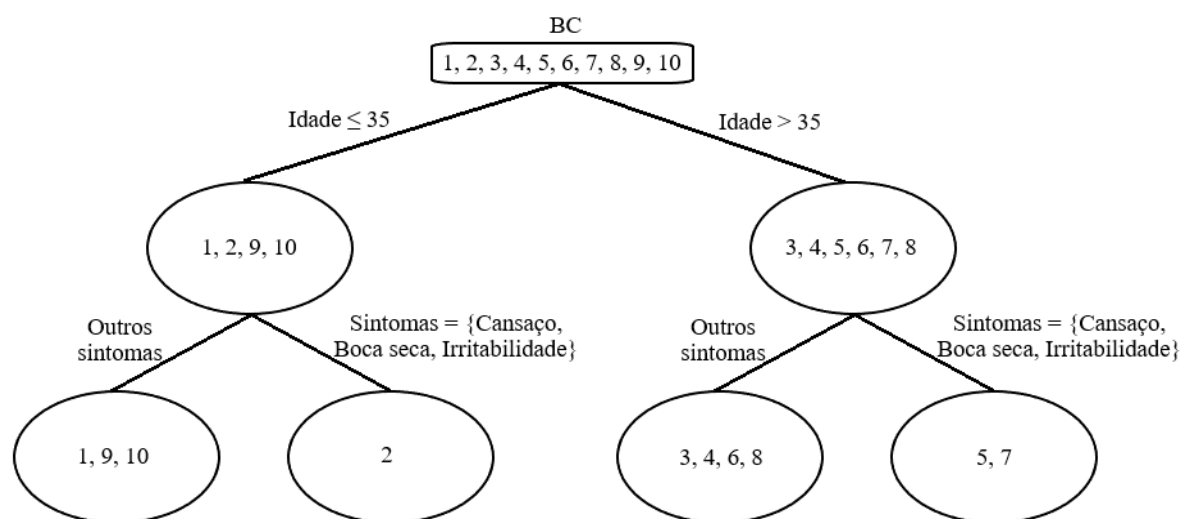
```

Fonte: VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003.

A utilização de árvores k-d possibilita uma recuperação de casos extremamente eficiente, uma vez que a otimização obtida pelo método é logarítmica, algo de grande utilidade para grandes bases de casos. Tal característica unida a facilidade de armazenamento da estrutura e de adição de novos casos à estrutura são as principais características do algoritmo. Sua desvantagem está no esforço necessário para criação da estrutura da árvore, e a restrição de armazenar casos que não possuam medidas de similaridade simples para realizar a divisão dos casos da BC pela árvore k-d.

Para exemplificar, será utilizado o mesmo problema e a mesma BC do exemplo anterior, mas desta vez utilizando árvore k-d para o caso {Rosa Oliveira; 56; F; Cansaço, Boca seca, Irritabilidade}. Os casos serão representados na árvore pelos seus números na BC, e as chaves para divisão serão a idade e os sintomas (dois sintomas iguais ou mais) respectivamente (cf. Figura 6).

Figura 6 - Exemplo de Recuperação com Árvores K-D



Fonte: Autoria própria.

Após a montagem da árvore, a recuperação é realizada percorrendo a árvore até uma de suas folhas, quando então será calculada a similaridade para o conjunto de casos presente nesta folha. Para este exemplo, o caso mais similar encontrado foi o caso 5 {Daniela Maria; 68; F; Perda de peso, Cansaço, Irritabilidade; Diabetes}, informando que o paciente possui “Diabetes”.

2.5.1.3 Métricas de Similaridade

A similaridade é um aspecto importante do método RBC e principalmente da etapa de Recuperação de Casos, pois será ela que diferenciará os casos mais semelhantes e úteis dos casos menos úteis de uma BC. Devido a essa importância, é necessária escolher uma métrica que determine o quão dois casos são parecidos de forma precisa, sempre analisando todos os atributos dos casos, afim de encontrar o caso solucionado ideal presente na BC.

A escolha da métrica utilizada irá impactar todo o processo do sistema RBC. Se houver um cálculo de similaridade preciso, os casos recuperados se adequarão melhor aos problemas enfrentados, e posteriormente nas etapas de Reuso não será necessário aplicar modificações no caso recuperado, o que conseqüentemente gera uma menor demanda de tempo e processamento.

Na literatura não há um consenso sobre uma medida de similaridade que seja aplicável para todos os tipos de domínios ou para todos os tipos de variáveis que

podem existir em uma base de casos. Os pesquisadores empregam o uso de medidas variadas, como: Distância Euclidiana, Distância Euclidiana Ponderada e Distância de *Manhattan*. A seguir tais distâncias serão analisadas neste trabalho.

2.5.1.3.1 Distância Euclidiana

É uma medida de distância frequentemente empregada quando as variáveis são quantitativas. Sua aplicação é recomendada para dados não padronizados, porém há uma desvantagem se houverem dados com diferença de escala (Cm e Km, por exemplo). No momento da transformação (a conversão de Cm para Km) os resultados euclidianos sofrem uma influência das dimensões que possuem valores maiores, em outras palavras, os valores maiores acabam aplicando um “peso” maior no cálculo.

A Distância Euclidiana pode ser calculada através da Equação 1 onde x e y representam os atributos dos casos a serem comparados.

$$d = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1)$$

Um exemplo de utilização desta métrica seria para calcular a rota de GPS de um helicóptero. Por se tratar de um veículo aéreo apenas um segmento de reta entre 2 pontos indica a distância a ser percorrida. Supondo que os pontos (4, 1) e (3, 7) representem x e y (origem e destino) respectivamente, realizando os cálculos temos a distância a ser percorrida:

- $d = \sqrt{(4 - 3)^2 + (1 - 7)^2}$
- $d = \sqrt{1 + 36}$
- $d = 6,082$

2.5.1.3.2 Distância Euclidiana Ponderada

O funcionamento da Distância Euclidiana Ponderada é semelhante ao da Distância Euclidiana comum, porém há a adição de pesos nas características dos

dados relativos à sua importância. O cálculo é mostrado na Equação 2, onde x e y representam os atributos dos casos a serem comparados e w é o peso atribuído a essas características.

$$d = \sqrt{\sum_{i=1}^n w_i (x_i - y_i)^2} \quad (2)$$

Exemplo: Um GPS de carro disponibiliza 2 opções de estrada, ambas de mesma distância, a percorrer do ponto de origem x (5, 5) para ponto destino y (2, 7). A primeira opção de estrada ($d1$) possui um pedágio de custo $w=5$, a outra estrada ($d2$) não possui pedágio, sendo assim assume valor de custo $w=1$. Realizando os cálculos temos os seguintes custos de cada estrada:

- $d1 = \sqrt{5 \cdot (5 - 2)^2 + 5 \cdot (5 - 7)^2} = \sqrt{5 \cdot 9 + 5 \cdot 4} = \sqrt{65} = 8,062$
- $d2 = \sqrt{1 \cdot (5 - 2)^2 + 1 \cdot (5 - 7)^2} = \sqrt{1 \cdot 9 + 1 \cdot 4} = \sqrt{13} = 3,605$

2.5.1.3.3 Distância de Manhattan

A Distância de Manhattan, possui funcionamento similar à Distância Euclidiana, mas não é influenciado pela diferença de escala sobre o resultado já que não há elevação ao quadrado das características a serem medidas. Mesmo sendo mais fácil de calcular que as distâncias citadas anteriormente, esta métrica pode não ser adequada se os atributos estão correlacionados, porque não existe a garantia da qualidade dos resultados obtidos (COLE, 1998). Para calcular a Distância de Manhattan utiliza-se da Equação 3, onde x e y representam os atributos dos casos a serem comparados.

$$d = \sum_{i=1}^n |x_i - y_i| \quad (3)$$

Para exemplificar, uma pessoa pega um taxi no centro de uma cidade no ponto $x(1, 1)$ e deseja ir até o ponto $y(3, 4)$. Calculando, temos a seguinte distância percorrida:

- $d = [|1 - 3| + |1 - 4|]$
- $d = [| -2| + | -3|]$
- $d = 5$

2.5.2 Reutilização de Casos

Nesta etapa é executada uma reutilização do conhecimento, em que será transferido o conhecimento do caso conhecido e solucionado, para o caso atual ainda não solucionado. A reutilização consiste principalmente em adaptar a solução do caso anterior para o caso atual. As técnicas de reutilização de casos procuram resolver problemas de adaptação, como quais atributos devem ser adaptados ou quais modificações devem ser realizadas (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003).

Pode ocorrer de casos recuperados da BC não satisfazerem completamente os requisitos do novo problema, assim é necessário adaptar a solução do caso recuperado antes de aplicá-la ao caso atual. Existem estratégias que contornam inteiramente o problema da adaptação, segundo Leake (1996), ao invés de tentar adaptar os casos recuperados, o modo mais simples é inflar a BC ao máximo na tentativa de garantir que todo problema possível possua uma solução ideal na BC. Essa estratégia pode ser adequada para tarefas analíticas (classificação, diagnóstico ou suporte a decisão, por exemplo), porém não é apropriada para tarefas sintéticas (configuração, projeto ou planejamento de soluções, por exemplo).

Problemas com maior complexidade requerem uma adaptação inteligente, sendo necessário que o sistema RBC seja capaz de adaptar os casos recuperados para que atenda às necessidades da situação corrente. A adaptação tem papel fundamental na flexibilidade dos sistemas RBC, a capacidade de resolver novos problemas depende da sua habilidade em adaptar soluções recuperadas e consertar soluções que falham em serem aplicadas (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003).

Diversas técnicas e estratégias podem ser usadas para realizar a adaptação, visando consertar as diferenças e resolver o problema enfrentado. Pode-se usar uma série de algoritmos, de simples a complexos, que aplicam integral ou parcialmente a solução encontrada. Também é possível utilizar não somente a solução mais similar, como também uma combinação de diversas soluções similares, com o objetivo de gerar a melhor proposta para o problema.

Entre as técnicas mais usualmente empregadas temos a Adaptação Nula, que considera que o caso mais similar encontrado é o ideal para solucionar o problema (total ou parte dele), desconsiderando qualquer tipo de modificação. Esta adaptação é útil e recomendada para tarefas de classificação, que normalmente são compostas de questões simples que possuem soluções simples, mas podem envolver raciocínios complexos (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003). A utilização desta adaptação conseqüentemente resulta numa aplicação mais rápida, pois a desconsideração de mudanças faz com que o sistema tenha uma menor demanda computacional.

2.5.3 Revisão de Casos

A revisão de casos consiste basicamente em duas tarefas (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003): avaliar a solução gerada pelo reuso, se estiver correta, a solução é retida na BC; caso contrário, reparar a solução utilizando conhecimento acerca do domínio.

O primeiro passo é detectar falhas avaliando as soluções fornecidas pelo sistema RBC. A avaliação se dá pela aplicação da solução no ambiente real em que o sistema se encontra ou em um ambiente simulado (mais recomendado), onde os resultados da aplicação serão analisados para descobrir se a solução pode ser aplicada no ambiente e retida na BC.

O segundo passo é reparar falhas do sistema RBC que forem detectadas. O reparo envolve a detecção de falhas na solução proposta e a recuperação ou geração de explicações para estas falhas. As explicações das falhas são utilizadas para modificar novamente a solução, de forma que o caso atual seja resolvido e a falha não ocorra novamente. Se a modificação satisfizer o problema enfrentado e não seja

detectado demais falhas, a solução pode ser aplicada no ambiente e o novo caso é retido na BC.

2.5.4 Retenção de Casos

Retenção de casos é o processo de armazenamento de um conhecimento que tenha sido útil para solucionar um problema, atualizando constantemente a BC. Assim, o sistema RBC é capaz de incrementar continuamente novos conhecimentos à BC, se tornando mais inteligente e poderoso com o passar do tempo (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003).

A retenção automática de conhecimento em RBC, além de simplesmente armazenar um caso onde a solução foi um sucesso, também pode utilizar técnicas de Aprendizado de Máquina. Pode-se distinguir basicamente três tipos de retenção em sistemas RBC:

- **Sem retenção de casos:** os sistemas mais simples não utilizam retenção automática de novos conhecimentos na BC. Este tipo de sistema utiliza uma BC fixa, que é modelada e desenvolvida de forma que satisfaça os problemas de um domínio específico, sendo desnecessário a inclusão de novos casos.
- **Retenção de solução de problemas:** é a forma mais comum de aprendizado de RBC, consistindo em toda vez que um novo problema for resolvido a experiência é retida para auxiliar a solução de problemas similares no futuro. Esta forma de aprendizado é uma das maiores vantagens do RBC, pois o aprendizado e a aplicação do conhecimento aprendido estão integrados. Alguns sistemas RBC também retêm experiências que falharam, utilizando esse conhecimento para evitar que esta falha se repita. A retenção pode ser executada através da força bruta (armazena todo caso solucionado na BC) ou utilizar técnicas inteligentes (filtragem ou avaliação de conhecimento, por exemplo).
- **Retenção de documentos:** este tipo de aprendizado RBC ocorre de forma assíncrona ao processo de solução de problemas. O conhecimento é incrementado à BC de forma independente, separada do processo de

solução de problemas, sendo dependente da disponibilidade de novos conhecimentos do domínio da aplicação.

A escolha da forma de retenção deve ser decidida de acordo com o objetivo final da aplicação e a forma como será feita a manutenção dos casos presentes na base de casos.

2.6 VANTAGENS E LIMITAÇÕES DO RBC

O RBC é um paradigma de solução de problemas fundamentalmente diferente de outras técnicas da IA. Ao invés de se basear unicamente em um conhecimento geral acerca do domínio da aplicação ou realizar associações entre os descritores do problema e os descritores das possíveis soluções, o RBC utiliza conhecimento de ações vivenciadas anteriormente na forma de casos.

A principal característica é que o RBC imita a forma como os humanos solucionam problemas, relembrando situações passadas. Conforme a experiência de um humano cresce, sua capacidade de resolver problemas mal definidos ou com conhecimento incompleto aumenta, habilidade que é altamente desejável em sistemas inteligentes. Abordagens tradicionais de IA que utilizam de banco de dados ou recuperação de informações têm falhado em atingir tal objetivo (LEAKE, 1996).

Os bancos de dados tradicionais recuperam registros com campos idênticos, já o objetivo dos sistemas RBC é justamente recuperar itens com correspondências inexatas (LEAKE, 1996). Recuperar um caso em uma BC é uma tarefa complexa devido às heurísticas e avaliações necessárias, que geram uma demanda de processamento que pode se tornar inaplicável em BC com grandes volumes de casos. Uma das limitações do desenvolvimento de sistemas RBC está no desafio de criar algoritmos de recuperação que sejam eficientes quando aplicados a centenas ou milhares de casos (ABEL, 1996).

Sistemas de Recuperação de Informações (RI) armazenam conhecimento sob a forma de palavras chaves, chamadas termos, onde os termos expressam o assunto principal desse item. Apesar de ser possível recuperar e indexar informações genéricas, sistemas RI apresentam sérias limitações a respeito da acurácia da recuperação (KOLODNER, 1993). O RBC e o RI possuem objetivos e abordagens básicas semelhantes, porém sistemas RI possuem mecanismos de recuperação mais

simples e menos flexíveis que o RBC, não possuem capacidade de adaptação e recuperam informações sem nenhum senso crítico.

Sistemas RBC também possuem uma fácil manutenção se comparados a sistemas baseados em conhecimento. Tais sistemas podem apresentar muitas regras e interdependências que quando alteradas podem ter efeito em todo o sistema, o que em RBC não acontece. Novos casos podem ser adicionados facilmente à base de casos e não inviabilizam ou alteram o funcionamento do sistema. Devido ao aprendizado incremental utilizado no RBC, a adição de um novo conhecimento pode ser realizada em pacotes (um conjunto de “casos-semente”), melhorando seu desempenho com o uso (VON WANGENHEIM e GRESSE VON WANGENHEIM, 2003).

Em contrapartida a essa flexibilidade de manutenção oferecida pelo RBC, está a dificuldade em construir e definir as características que irão formar o método. Há a necessidade de selecionar as características mais importantes e determinantes do domínio do problema, assim como as atividades que alteram essas características, procurando representar tais informações da forma mais sucinta possível. O que em alguns casos é difícil definir, devido à grande massa de informação necessária, ou a quantidade de regras atreladas ao domínio.

2.7 CONSIDERAÇÕES FINAIS

O RBC é um método que procura reproduzir o comportamento cognitivo humano. Ao se deparar com uma situação problemática, o método simula o processo de lembrança, procurando um problema parecido já solucionado na memória e tentando utilizar a solução deste problema na situação enfrentada. É um método que conforme soluciona problemas vai se aperfeiçoando cada vez mais, pois toda vez que resolve um problema, salva esta solução para utilizar posteriormente, fazendo com que a cada nova situação enfrentada, se torne melhor e mais eficiente. Devido a este autoaperfeiçoamento que o RBC possui, este foi escolhido para ser aplicado no domínio de condução de trens. Assim, a cada viagem as sugestões de ações a serem tomadas se tornam melhores.

3 CONDUÇÃO DE TRENS

Em 1803 o engenheiro inglês Richard Trevithick construiu o primeiro veículo similar a uma locomotiva, pesando 5 toneladas e atingindo velocidades de 5 km/h. Mas foi o também engenheiro inglês George Stephenson o verdadeiro criador da tração a vapor, construindo em 1813 a locomotiva Blucher, possuindo 8 vagões com 30 toneladas. A partir de 1840 houve uma expansão das estradas de ferro na Inglaterra, a facilidade de transporte de cargas pesadas por longas distâncias e por frete reduzido, foi fundamental para consolidar a Inglaterra como potência mundial a partir da Revolução Industrial (NETO, 2011) (PORTO, 2004).

Porém os trilhos apresentavam desgastes, o que encarecia a manutenção de vias permanentes. Foi Henry Bessemer em 1856 que fabricou trilhas de aço que praticamente não desgastavam, fazendo com que as estradas de ferro passassem a trafegar sobre trilhos de aço, mais seguros e resistentes. Durante as duas grandes guerras ocorre o desenvolvimento da tração a diesel, que aumentou a velocidade de transporte de passageiros e carga, com a primeira experiência sendo executada na Alemanha em 1933, com um trem de passageiros que percorreu a linha Berlim-Hamburgo a 160 km/h (PORTO, 2004).

No século passado, com a pressão da população para diminuir a fumaça dos trens que atravessavam as cidades, houveram diversas tentativas de aplicar energia elétrica na tração dos trens, com Von Siemens obtendo sucesso em 1895. Porém, no começo somente as ferrovias de intenso tráfego utilizaram essa inovação, havendo diversos outros trens em trechos afastados da cidade ainda utilizando tração a diesel (PORTO, 2004).

Dado um breve resumo do surgimento dos trens e das vias férreas, as próximas seções apresentarão características que compõem os trens, as vias e como funciona a condução desses trens.

3.1 CARACTERÍSTICAS DO TREM

Um trem é um sistema complexo que envolve veículos, carga, molas e massas rotativas que interagem com a via férrea de várias formas. Essa interação varia de acordo com distribuição dos veículos na via, o comprimento total do trem (quantidade

de vagões), vagões cheios ou vazios, velocidade do trem e características particulares da locomotiva. Devido a tais razões pode-se afirmar que não existem dois trens iguais, sendo assim não existe uma única maneira de se conduzir um trem, as ações a serem tomadas durante o trajeto variam de acordo com a situação enfrentada no momento.

Os trens são formados por uma ou mais locomotivas e um ou mais vagões, onde estes possuem características particulares que influenciam na condução. Atualmente, para transportar cargas e/ou pessoas a maioria das locomotivas utilizam motor diesel-elétrico ou elétrico, podendo chegar a pesar cerca de 100 a 210 toneladas e medir entre 16.7 e 22.5 metros. Para andar sobre os trilhos, tanto as locomotivas quanto os vagões são posicionados em cima de uma estrutura chamada *truque*, e é nessa estrutura que são paralelamente posicionados os eixos, que contém as rodas do trem. Locomotivas modernas possuem eixos de classe B-B (dois truques independentes com dois eixos por truque) ou C-C (dois truques independentes com três eixos por truque), que normalmente medem de 2.7 a 3.2 metros de largura (BORGES, 2015a).

É possível combinar e controlar diferentes locomotivas ao mesmo tempo, as ligando eletricamente ou através de sinais de rádio, fazendo com que as ações de tração e frenagem executadas na “locomotiva principal” tenham efeito nas demais. Em casos de trens muito longos, é praticável distribuir uma ou mais locomotivas ao longo do trem no intuito melhorar o desempenho de frenagem. Mas também pode-se posicionar várias locomotivas na cabeça do trem, que desempenharão o papel de uma única locomotiva mais poderosa (BORGES, 2015a).

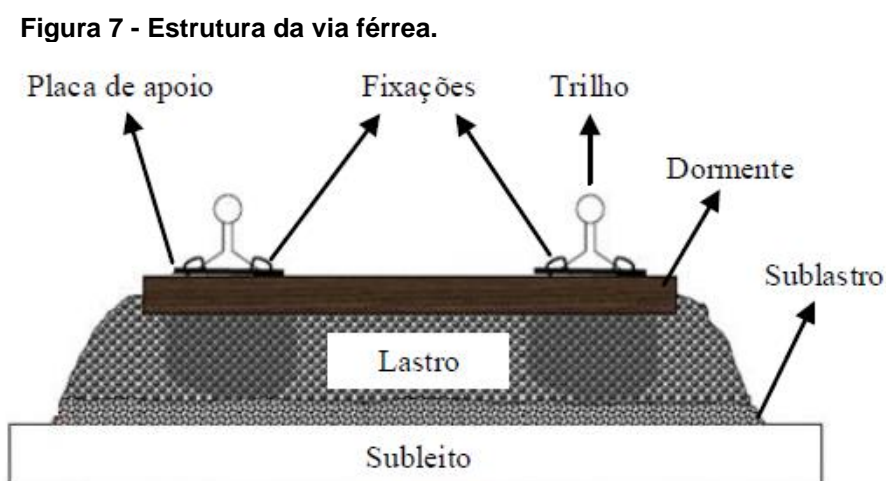
Normalmente, as locomotivas possuem computadores de bordo responsáveis pela leitura diversos sensores espalhados ao longo do trem. Estes sensores indicam informações como: velocidade, ponto de aceleração, pressão dos freios, se o freio dinâmico está em uso ou não, etc. Para se comunicar com a Central de Controle de Trafego as locomotivas são providas de rádios, e também são equipadas com GPS. De acordo com o percurso do trem, o computador de bordo identifica qual trecho da via o trem se encontra e pode informar ao maquinista qual a velocidade permitida naquele trecho (BORGES, 2015a).

Para o transporte de cargas são utilizados os vagões, que pesam cerca de 30 a 125 toneladas cada. Os vagões possuem demarcações que especificam

informações como: peso do vagão vazio, a capacidade de carga suportada pelo vagão (recomendada e máxima), além de recomendações para ajuste dos freios.

3.2 CARACTERÍSTICAS DA FERROVIA

Para que seja possível um trem se movimentar, é necessária uma estrada apropriada, chamada via férrea ou via permanente. Esta via é o meio físico que permite o trem se locomover, sendo composto de trilhos paralelamente fixados a um dormente. Essa estrutura transmite a carga dos trens aos lastros, que por sua vez transmite aos sublastros que absorvem todo peso da via e do trem, conforme a Figura 7 (CHANDRA e AGARWAL, 2007) (KUTZ, 2011) (apud BORGES, 2015a).



Fonte: Borges (2015a).

A distância entre os trilhos é uma característica intrínseca da via e é chamada de *bitola*. As *bitolas* são medidas em metros e possuem tamanhos variados, podendo haver mais de um tipo de *bitola* em uma mesma via. As *bitolas* são padronizadas em função do país, organização ferroviária e velocidade da via (vias de maior velocidade, possuem *bitolas* maiores), mas geralmente seguem medidas como: 1.0m, 1.435m ou 1.6m. No Brasil, a maioria das vias utiliza a *bitola* de 1.0 metros (CHANDRA e AGARWAL, 2007) (apud BORGES, 2015a).

A organização da via férrea é composta por uma linha principal, que é onde o trem se desloca e interliga as estações, e linhas secundárias, que são usadas quando a linha principal está em manutenção. A via também possui desvios à linha principal,

que servem para manobrar, realizar ultrapassagens e cruzamentos entre os trens. Para os trens se moverem entre as linhas, existem dispositivos instalados nos trilhos que auxiliam a execução dessa mudança, chamados de Aparelhos de Mudanças de Via (AMV). Quando uma via férrea cruza com uma rodovia no mesmo nível, esse encontro é denominado Passagem de Nível (PN).

As plantas de uma ferrovia são compostas de estudos topográficos do terreno, onde o projeto reúne informações dos perfis horizontal e vertical da via. O perfil horizontal demarca o trajeto a cada 20 metros, e o perfil vertical demarca o trajeto a cada metro. O objetivo dessas marcações, é representar toda a geologia do terreno em que a via está localizada, retratando as curvas em forma de coordenadas planas (x, y) fornecidas pelo perfil horizontal, enquanto o perfil vertical informa elevações, desníveis e acidentes geográficos no terreno na forma de outra coordenada (z). Tais diferenças de altura no terreno também são chamadas de *rampas*, e indicam o percentual de inclinação do terreno (BORGES, 2015a).

3.3 CICLO DE CONDUÇÃO

Uma boa condução de trem será aquela que evite danos a via permanente, aos equipamentos, a carga transportada e junte rapidez com eficácia de consumo de combustível (BORGES, 2015a). Para realizar uma boa condução o maquinista deve ser capaz de usar uma combinação adequada do acelerador e dos freios, com intuito de economizar combustível, diminuir os esforços entre o veículo e a estrutura da via e evitar danos as rodas, aos trilhos, aos equipamentos e a carga.

O momento em que se deve ser acelerar ou frear um trem, geralmente é definido pela via. É recomendado que os maquinistas mantenham uma velocidade abaixo da velocidade máxima permitida e acima da velocidade cruzeiro (é a faixa entre a velocidade mínima ou ideal e máxima da via), para que não haja riscos de o trem descarrilhar ou parar no meio do trajeto. Por exemplo, um trem trafega a 45 km/h em uma via que permite velocidades de até 50 km/h. Caso o trem esteja a uma velocidade decrescente próxima ou abaixo de 40 km/h (supondo que esta seja a velocidade mínima ou ideal da via) o maquinista deve acelerar o trem, assim como se a velocidade estiver aumentando e estiver próxima de 50 km/h, deve-se diminuir a aceleração ou frear o trem. Além disso, a velocidade deve ser mantida.

Este trabalho tem o objetivo de realizar uma condução assistida, que se caracteriza por um programa de computador que auxilie a condução dos trens. Essa condução utiliza a leitura em intervalos de tempo, dos sensores instalados na locomotiva, para informar o computador de bordo com os dados do ambiente. Nem todas as informações são obtidas durante o percurso, sendo algumas delas arranjadas por ordem de despacho (número de veículos, origem, informações técnicas das locomotivas e perfil de toda a via). Outros dados podem ser obtidos pelo próprio computador de bordo, como por exemplo a posição do trem na via, determinada por um GPS (BORGES, 2015a).

Mas apenas a obtenção desses dados não é o suficiente para decidir e aplicar uma ação. Maquinistas realizam conduções intuitivas com base em suas experiências, mas para uma condução assistida, são necessários cálculos que projetem a situação atual e a situação futura (após aplicar uma ação) do trem, verificando possíveis erros. Para estes cálculos, Borges (2015a) lista um conjunto de equações que tem como entrada informações relativas de um trem, como: resistência total, força de tração necessária, força de tração máxima aplicável, aceleração, consumo, dentre outras. Tais equações tem por objetivo determinar qual ação (acelerar, frear ou manter velocidade) deve ser executada, sempre respeitando restrições como, sinalização e velocidade máxima permitida na via. Um exemplo dos cálculos realizados durante a condução pode ser encontrado na seção 2.4.4 de Borges (2015a).

A maior vantagem em utilizar uma condução assistida, está na possibilidade de analisar as ações durante a condução através de cálculos. As ações a serem efetuadas podem ser investigadas em tempo de execução ou até mesmo antes de serem aplicadas, podendo negar ações potencialmente equivocadas para que não ocorra erros, ou na pior das hipóteses, acidentes com o trem e com a via.

3.3.1 Aceleração

Para realizar uma boa condução os maquinistas devem executar procedimentos de aceleração, administrando a velocidade do trem e aproveitando o perfil do terreno da via (BORGES, 2015a). Outros procedimentos recomendados são:

- Ao dar partida ou acelerar um trem em movimento, deve-se aguardar e certificar que todos os freios da composição tenham sido soltos;

- Ao dar partida em um trem, a aceleração deve ser feita ponto a ponto (como marchas de um carro), com um intervalo de dois a três segundos. Deve-se evitar avanços bruscos no acelerador, para que o trem absorva a potência do ponto selecionado para então selecionar o próximo ponto;
- Aproveitar o perfil do terreno, não acelerando o trem desnecessariamente para manter velocidade se na sequência for preciso desacelerar ou frear o trem;
- Acelerar o trem de acordo com restrição de velocidade permitida na via, não acelerar o trem para logo depois freia-lo;
- Se houver patinação de rodas, informada por uma lâmpada específica no painel, os pontos de aceleração devem ser reduzidos até que a lâmpada apague. Se a luz não apagar, o trem deve ser parado para verificar se as rodas giram livremente;
- Não utilizar a aceleração para manter o trem parado em rampas, pois esta atividade danifica os motores da tração.

Por fim, recomenda-se não deixar motor diesel ligado nos pátios, postos de abastecimentos, pontos de cruzamento, etc. Se em 30 minutos o trem não for ser movimentado, o motor deve ser desligado senão desperdiçará combustível.

3.3.2 Frenagem

A frenagem é uma operação mais complexa devido a existência dos diferentes tipos de freios, que são: automático, dinâmico e independente. O freio automático atua nas rodas dos vagões e se propaga da locomotiva para o resto do trem, tendo um intervalo de tempo para a ativação, pois é necessário que o ar comprimido que ativará os freios percorra toda a extensão do trem a partir da locomotiva. O freio dinâmico é utilizado invertendo o ponto de aceleração, exercendo uma força contrária ao movimento do trem. Este freio geralmente utiliza o mesmo controlador do acelerador e possui um ponto máximo e mínimo permitido para ser usado. Por fim o freio independente, que assim como o automático, atua nas rodas da locomotiva por meio da força aplicada em sapatas de freio (Loumiet, Jungbauer, & Abrams, 2005) (apud BORGES, 2015a).

A aplicação do freio deve ser feita em duas etapas, aplicando inicialmente uma redução de 5 a 8 psi (libras) com um intervalo de 20 segundos, complementando com a segunda aplicação com a redução desejada. Recomenda-se que o esforço de frenagem em trechos sinuosos ou em curvas acentuadas deve ser mantido no mínimo e que não se aplique pressões excessivas nos freios enquanto o trem estiver em movimento. Também deve-se ficar atento se houver algum ruído de ar no equipamento de freio da locomotiva ou algum sinal evidente que esteja ocasionando a soltura involuntária dos freios, nesses casos deve-se imediatamente aumentar a aplicação do freio no mínimo em 5 psi (BORGES, 2015a).

Para reduzir a velocidade ou parar o trem, não é recomendado utilizar somente o freio independente, assim como aplicar o freio independente para controlar ou reduzir a velocidade do trem acima de 20 km/h. Em velocidades na faixa de 20 a 30 km/h é mais eficaz utilizar o freio dinâmico, aguardando cerca de 10 segundos após a utilização para usar novamente. O freio independente não deve ser utilizado simultaneamente com o freio dinâmico para velocidades acima de 10 km/h, e velocidades abaixo de 10 km/h o freio independente é apropriado para compensar a perda de eficácia do freio dinâmico. Se avisos de “DESLIZE DE RODAS” ou de “EXCESSO DE FRENAGEM” (informados por lâmpadas no painel) acenderem, deve-se reduzir a frenagem dinâmica até que a luz se apague, do contrário parar o trem e revisar a locomotiva (BORGES, 2015a).

3.4 CONSIDERAÇÕES FINAIS

A condução de trens é uma tarefa complexa e requisita do maquinista atenção e cuidado em cada trecho do percurso, para que não ocorra acidentes ou haja danos ao trem, a via e a carga, sempre visando uma condução segura e econômica. As ações executadas pelo maquinista não seguem um padrão e a sua experiência é de vital importância no momento de tomada de decisões. Assim, na tentativa de imitar as ações dos maquinistas foram estudadas técnicas de Inteligência Artificial para serem desenvolvidas, como raciocínio baseado em casos e técnicas de recuperação de casos, com objetivo de criar uma condução assistida para o problema de condução de trens de carga.

4 DESENVOLVIMENTO

Este capítulo descreverá os passos da construção deste trabalho, como o método RBC é usado no problema de condução de trens de carga, descrevendo o problema a ser solucionado, a construção dos casos e o modo como serão realizadas as operações do método. O foco do desenvolvimento será na etapa de recuperação, onde foi desenvolvido um método de Recuperação em Dois Níveis Dinâmico, em que os atributos utilizados para a filtragem podem ser alterados durante a execução de uma viagem.

4.1 APLICAÇÃO DO MÉTODO RBC

A condução de um trem de carga é realizada por maquinistas, e os trechos de atuação destes maquinistas é definido de acordo com sua experiência em enfrentar obstáculos e minimizar riscos durante o percurso. Sendo assim, maquinistas iniciantes comumente são alocados em trechos mais simples, onde não há muitas inclinações verticais na ferrovia, poucas curvas e a velocidade máxima se mantém quase sempre constante. Já maquinistas experientes são designados a trechos mais complexos, onde a ferrovia possui diversas inclinações verticais e muitas curvas. Devido a isto, são necessárias diferentes restrições de velocidade para diferentes trechos da viagem e, conseqüentemente, é necessário que haja várias trocas dos pontos de aceleração e aplicação dos freios.

Devido aos diferentes obstáculos que podem ser encontrados durante uma viagem, a aplicação do método RBC na condução de trens de carga tem como objetivo realizar uma condução assistida, analisando situações no decorrer de um trajeto e sugerir ações para auxiliar maquinistas durante a condução. Com essas sugestões, a aplicação do método RBC também tem por objetivo promover um melhor aproveitamento dos recursos disponíveis, como: combustível, locomotivas e vagões.

Este trabalho é baseado na tese de Borges (2015a), na qual ele realizou uma pesquisa aprofundada sobre o domínio (condução de trens), sobre o funcionamento completo do método RBC e sobre todas as características que envolvem a condução de um trem e uma via férrea. Ao final de sua pesquisa, foi implementado um simulador de condução de trem. O simulador permite inserir informações sobre uma via férrea,

sobre um trem e exemplos de ações de maquinistas, afim de simular a condução deste trem nesta via férrea utilizando o RBC. Com o uso de técnicas como o algoritmo *KNN* para realizar a busca de casos, fórmulas como a da Distância Euclidiana para determinar a similaridade entre casos e a utilização de Algoritmo Genético para efetuar a adaptação destes casos, é possível simular a condução automática de um trem em uma via férrea.

Para isso é necessário ter conhecimento de diversas informações a respeito do domínio do problema, como: (a) informações sobre a ferrovia que será efetuada a condução (tamanho da via, pontos onde haverá curvas e níveis, velocidades nos diferentes pontos da via, etc), (b) informações sobre o trem (modelo da locomotiva, número de vagões, modelo do motor, consumo do motor, etc) e (c) uma Base de Casos (BC) pré-montada composta de casos reais (ações de maquinistas). Todas estas informações, foram organizadas e armazenadas em arquivos .csv, conforme Borges (2009), para que possam ser acessadas de forma rápida e fácil.

Sendo assim, o objetivo deste trabalho é realizar uma mudança e melhora na forma como é feita a recuperação de casos úteis. Uma vez que um caso é recuperado, quanto mais semelhante ao problema atual ele for, menor serão as modificações necessárias nas etapas de Reuso e Revisão, procurando obter uma melhora no desempenho do método como um todo.

A proposta é implementar uma Recuperação em Dois Níveis Dinâmica, que permita diferentes critérios de filtragem de casos, sendo que os critérios podem mudar durante o percurso. O problema em selecionar critérios fixos para a filtragem é que não há garantia que a filtragem e seleção de um caso será sempre eficiente, podendo ocasionar na recuperação de um caso não útil. Isso ocasionaria, na etapa de revisão, mais alterações para tornar o caso útil, e isso consequentemente tornaria o funcionamento do método mais lento.

Desta forma, serão escolhidos antecipadamente entre os atributos do caso, três diferentes critérios para realizar a filtragem. Cada vez que um caso for recuperado, após a etapa de revisão será comparado o ponto de aceleração aplicado antes e o depois da etapa, afim de saber as diferenças entre o ponto de aceleração recuperado e o ponto efetivamente aplicado. Essa diferença indica o quanto o caso teve que ser modificado para solucionar o problema atual. Quanto mais modificações forem necessárias, menos semelhantes estão sendo os casos recuperados. Se for

constatado que as recuperações estejam selecionando casos distintos do caso alvo, isso indicará que o critério não está sendo eficiente, portanto, o critério selecionado será trocado por outro dos critérios escolhidos.

As seções seguintes irão descrever como o trabalho foi realizado, dissertando como é feita a representação, recuperação e seleção dos casos e a troca dos critérios de filtragem.

4.2 O CASO

Para a representação de um caso nesta aplicação do método RBC, será utilizado a representação clássica de atributo-valor. Este é um modo simples e prático de organizar e representar informações do mundo real. Neste formato, cada atributo de um caso pode assumir um tipo de valor diferente, podendo haver valores descritivos (*string*) e/ou numéricos (*int*, *float*, *double*) dentro de um mesmo caso.

Para a condução de trens poucas informações descritivas serão necessárias nas simulações aqui realizadas, pois a maioria das informações serão de valores numéricos, como a velocidade atual e velocidade máxima do trem, por exemplo. Devido a isso, é desnecessário que a representação do caso seja complexa, pelo contrário, há necessidade de que seja uma representação simplificada para que o acesso as informações do caso e cálculos necessários sejam simples e rápidos de serem realizados.

O tempo de resposta do sistema durante a condução de um trem não pode demorar, pois as condições do trem e da via podem mudar, o que inutilizaria a resposta do problema solucionado. Contudo, este é um aspecto técnico que não será preocupação deste trabalho, uma vez que as simulações são realizadas apenas de modo virtual e não em uma ferrovia de fato.

Além disso, o RBC é um método incremental que conforme soluciona casos, os salva em sua BC. Assume-se que a BC pode chegar a ter grandes tamanhos devido a uma quantidade enorme de casos solucionados pelo método. Como a representação atributo-valor é simples, facilita cálculos (como as medidas de similaridade) e funciona melhor com grandes bases de casos, está foi a escolhida.

A representação atributo-valor assume que um caso C é definido por um conjunto A de n pares de atributo-valor, representados por

$$A = \{[a_0; v_0], [a_1; v_1], \dots, [a_n; v_n]\} = \{[a_i; v_i]\}_0^n = \{A_i\}_0^n$$

onde $[a_i; v_i]$ é um par atributo-valor (A_i), sendo a_i que indica a descrição dos atributos e v_i seus respectivos valores (BORGES, 2015a).

No RBC, as maiores dificuldades encontradas para construir o sistema estão em: (a) qual estrutura usar para representar um caso (como descrito anteriormente, neste trabalho foi escolhida a representação atributo-valor); (b) o que reter em um caso; e (c) como organizar os casos de modo que facilite a recuperação e reuso. Devido a condução de um trem ser uma tarefa complexa e envolver diversos fatores é difícil representar essa ação de forma sucinta computacionalmente, para isso é necessário selecionar as características determinantes do problema.

Para representar uma situação enfrentada e uma ação executada por um maquinista durante uma viagem, é necessário selecionar as informações determinantes e atuantes na movimentação de um trem. Para este domínio de problema, um caso irá representar um momento ou intervalo específico de uma situação enfrentada em uma viagem. Desta forma, as principais características que este caso precisa representar se referem à descrição do trem (número de locomotivas, número de vagões), a descrição do comportamento/situação do trem (velocidade inicial, velocidade máxima, ação aplicada no trem) e a descrição da via férrea onde está sendo realizada a viagem (posição atual na via, percentual de rampa).

Para formar a estrutura de um caso computacionalmente, foram selecionadas as seguintes características consideradas determinantes neste domínio: ação executada (EAC), tipo do perfil a percorrer (Perfil), quilômetro inicial (KM), número de locomotivas (NL), número de vagões (NW), velocidade inicial (IS) em km/h, velocidade final (FS) em km/h, velocidade máxima (MS) em km/h, percentual de rampa (%R) e deslocamento total (CL) em metros. Estas características juntas irão formar o conjunto inicial de um caso, o problema, e seguem o mesmo padrão definido por Borges (2015a), para ser possível comparar os resultados.

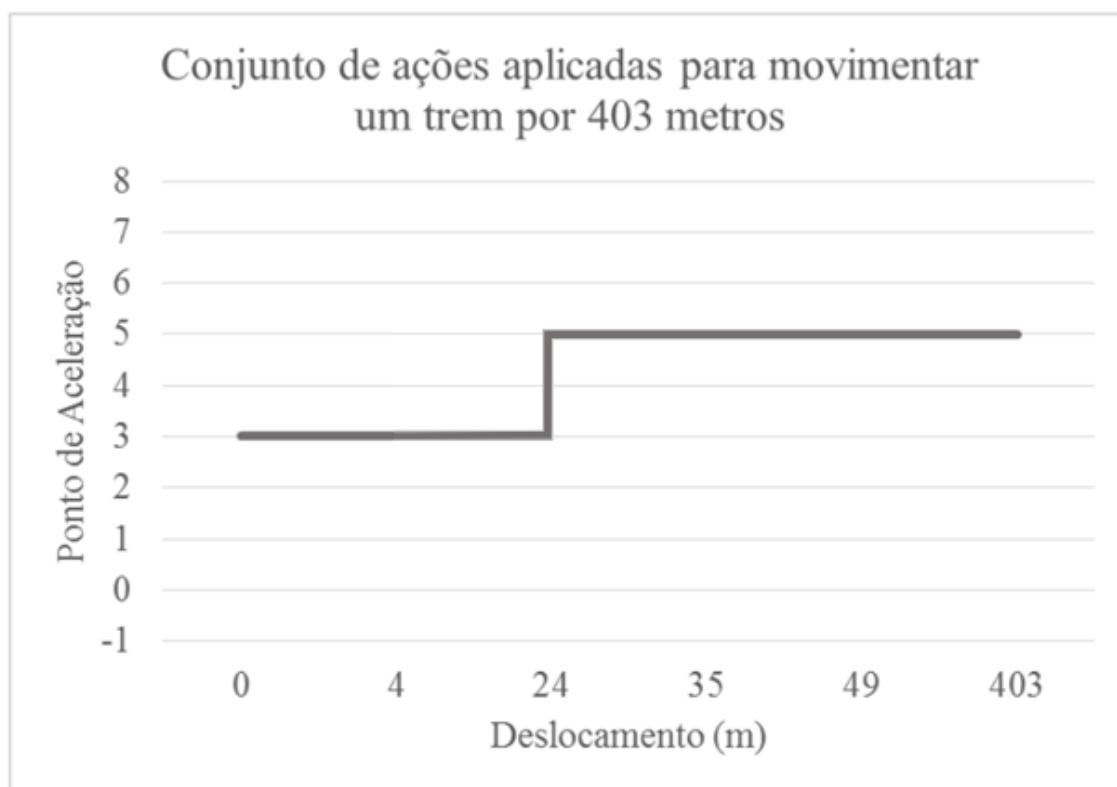
Porém, um caso é determinado por um problema e sua respectiva solução. Para este domínio é necessário identificar qual é a principal atividade que altera todas as demais informações listadas no problema. Esta atividade são os pontos de aceleração, que funcionam de forma semelhante a marchas de um carro, e são

usados para aumentar, manter ou diminuir a velocidade do trem. Desta forma a solução S de um caso é determinada por um conjunto de pares atributo-valor, representador por

$$S = \{[pa_0; m_0], [pa_1; m_1], \dots, [pa_n; m_n]\} = \{[pa_i; m_i]\}_0^n = \{S_i\}_0^n$$

onde $[pa_i; m_i]$ é um par atributo-valor (S_i), sendo pa_i um ponto de aceleração e m_i a posição (ou momento) de aplicação de pa_i (BORGES, 2015a). Cada m_i alcançada representa um deslocamento do trem. A Figura 8 mostra uma aplicação de um conjunto S , demonstrando onde cada pa_i será aplicado ao deslocar um trem por um trecho de 403 m.

Figura 8 - Aplicação dos pontos de aceleração de um plano solução S em um trecho de 403 m.



Fonte: Adaptado de Borges (2015a).

Desta forma, define-se um caso C como sendo a combinação dos pares atributo-valor do problema (A_i) com os pares atributo-valor da solução (S_i). Definimos C por

$$C = \langle A; S \rangle$$

A Figura 9 mostra um exemplo de um caso contendo um problema e sua respectiva solução.

Figura 9 - Exemplo de um Caso representado computacionalmente.

$$C = \left\{ \langle EAC, Acelerar \rangle, \langle Perfil, Em Nível \rangle, \langle KM, 339.404 \rangle, \langle NL, 3 \rangle, \langle NW, 58 \rangle, \langle IS, 10 \rangle, \langle FS, 11 \rangle, \langle MS, 45 \rangle, \langle \%R, -0.41 \rangle, \langle CL, 0.403 \rangle, \right. \\ \left. \langle S, \{ \langle 3,0 \rangle, \langle 3,4 \rangle, \langle 5,24 \rangle, \langle 5,35 \rangle, \langle 5,49 \rangle \} \rangle \right\}$$

Fonte: Adaptado de Borges (2015a).

Um problema não solucionado seria um caso C contendo apenas as informações do problema (A_i) sem conter uma solução (S_i). O principal objetivo do RBC é, utilizar as informações do problema de um caso C_1 sem solução para encontrar um caso C_i que possua solução e tenha as informações do problema semelhantes a C_1 , para então utilizar a solução de C_i . A próxima seção irá descrever como será realizada a recuperação de um caso, a partir das informações do problema de um caso.

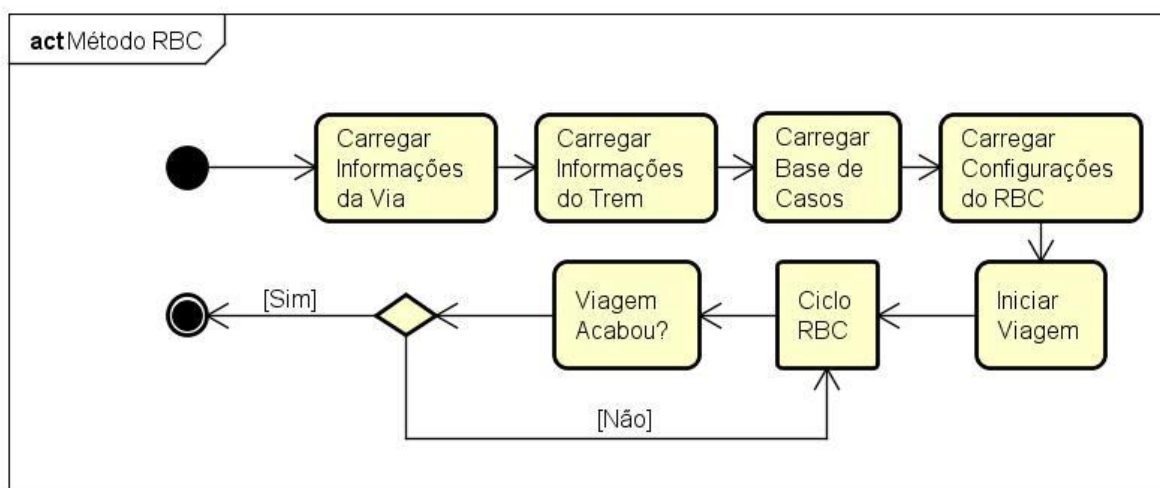
4.3 A RECUPERAÇÃO DE CASOS

A etapa de recuperação é o foco principal deste trabalho. Nesta etapa, dado um determinado problema alvo, será realizada a filtragem e a seleção de um caso potencialmente útil presente na BC para ser reutilizado. O principal objetivo desta etapa é selecionar o caso mais semelhante possível ao problema alvo de forma rápida, para que na etapa de Revisão seja preciso o mínimo de alterações antes de aplicar a solução do caso recuperado.

Para o funcionamento do método RBC, as primeiras operações a serem realizadas, serão o carregamento das informações contidas nos arquivos .csv. Esses arquivos contém as informações relativas a Via Férrea, ao Trem e a Base de Casos e serão carregadas nas etapas de “Carregar Informações da Via”, “Carregar Informações do Trem” e “Carregar Base de Casos”. A etapa “Carregar Configurações do RBC” será responsável por configurar o método, informando a quantidade de casos

a serem recuperados, como será calculada a similaridade e o tipo de adaptação a ser utilizada, além de outras configurações a respeito da via, do trem e o local onde os arquivos .csv serão encontrados. O diagrama do método RBC é representado pela Figura 10.

Figura 10 - Diagrama do funcionamento do Método RBC



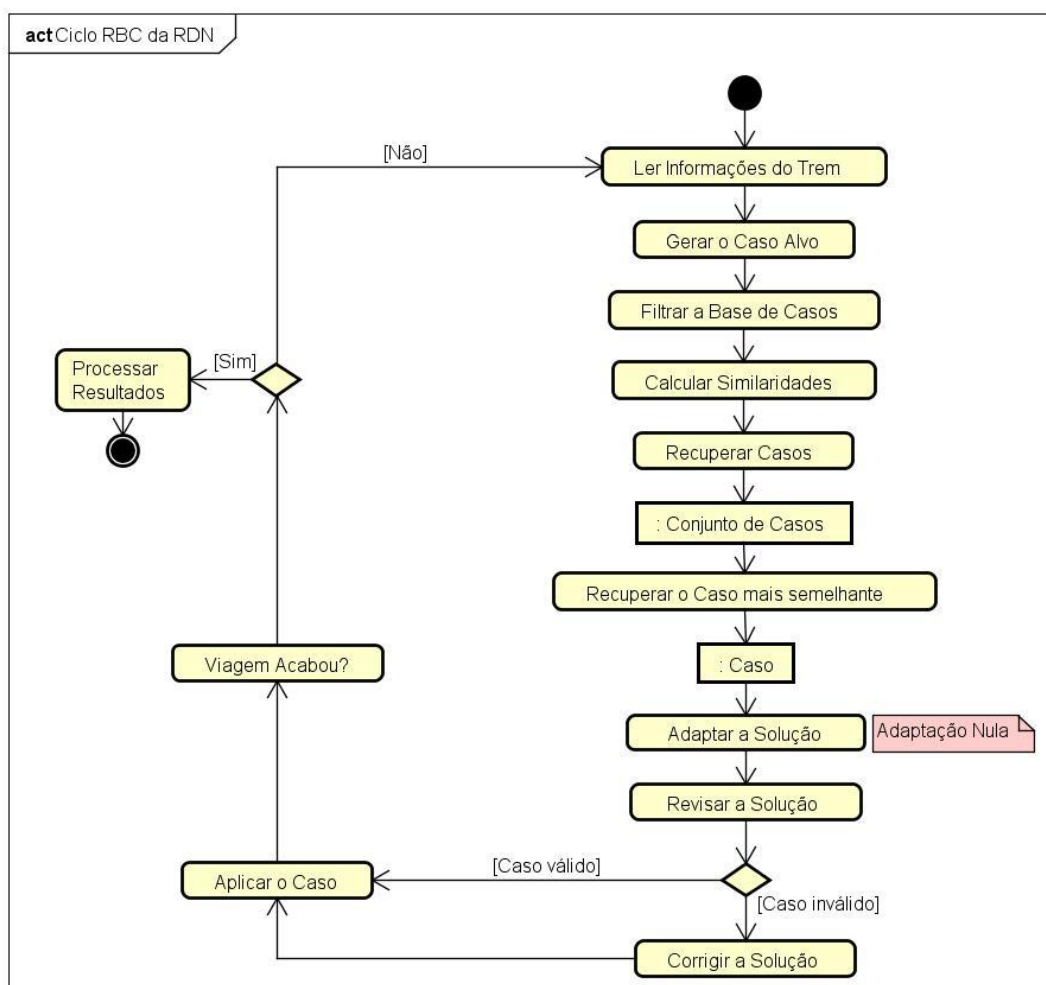
Fonte: Autoria própria.

Com todas as informações necessárias carregadas e configuradas, a etapa “*Iniciar Viagem*” irá começar o processo do “*Ciclo RBC*”. A Figura 11 mostra resumidamente o funcionamento desta etapa. Dado uma situação, as informações atuais do trem serão captadas através de sensores, e então será gerado um caso alvo a ser solucionado. A partir deste novo problema, será calculada as similaridades entre a BC e o caso alvo, e o caso mais semelhante presente na BC será recuperado.

Com o caso recuperado, o mesmo será levado a etapa de Adaptação (Reuso). Neste trabalho os casos recuperados não serão adaptados, sendo utilizada a Adaptação Nula para que não haja alterações no caso. A próxima etapa será a de Revisão. Nesta etapa o caso será validado afim de descobrir se é válido para ser efetivamente aplicado ao trem. Se não for válido, o caso pode sofrer modificações no conjunto da solução *S* até que se torne válido.

Para este trabalho, a principal etapa do diagrama da Figura 11 é a “*Recuperar Casos*”. É dentro desta etapa que está localizado toda a aplicação dos métodos a serem abordados adiante. A Figura 12 demonstra em detalhes o funcionamento desta etapa.

Figura 11 - Diagrama do Ciclo RBC.



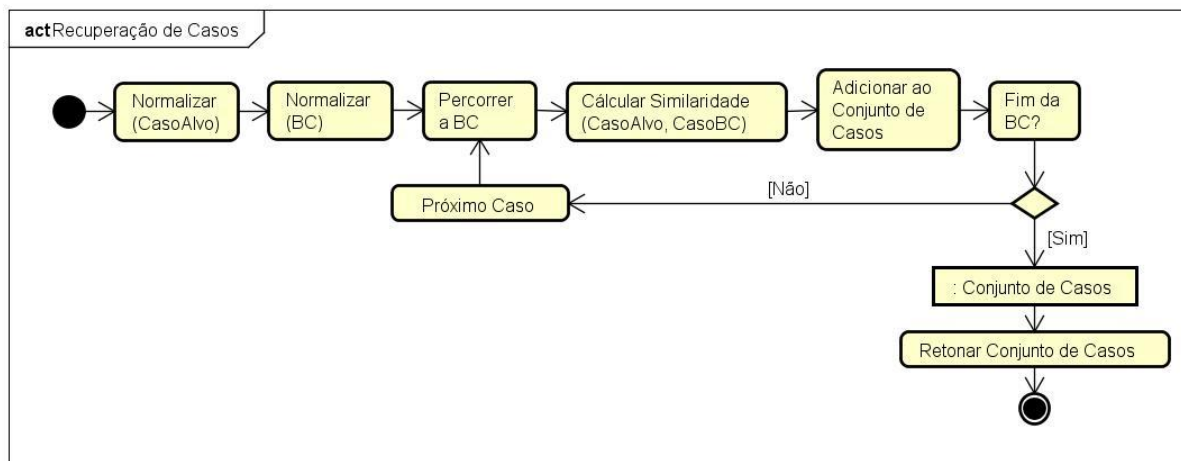
Fonte: Autoria própria.

Na etapa de Recuperação de Casos, antes de efetuar os cálculos é necessário aplicar uma normalização tanto no caso alvo como em toda a BC, no intuito de evitar que alguns atributos que possuam valores maiores, tenham maior impacto na similaridade. Assim, com a normalização todos os atributos passam a possuir a mesma faixa de valores (de 0 a 1), tendo o mesmo impacto na similaridade independente do seu valor real.

Após isto, a BC será percorrida para calcular as similaridades entre o caso alvo e os casos da BC. Esse processo irá gerar um novo conjunto de casos (BC Filtrada), que será composto dos casos com suas respectivas similaridades. Estes casos serão ordenados pela sua similaridade. O conjunto de casos gerado será

retornado para o *Ciclo RBC* (Figura 11), para que então a etapa “*Recuperar o Caso mais semelhante*” selecione o caso mais similar ao caso alvo enfrentado.

Figura 12 - Diagrama da Recuperação de Casos.



Fonte: Autoria própria.

Para a implementação da recuperação de casos, foram selecionados dois métodos. O primeiro método é a Recuperação por Dois Níveis e o segundo método é a Recuperação em Dois Níveis Dinâmica. O primeiro método irá consistir em realizar uma filtragem na BC antes de realizar os cálculos de similaridade. O segundo método é uma variação do primeiro, onde os atributos utilizados para a filtragem são alternados durante a execução da viagem. As seções a seguir irão descrever o funcionamento destes dois métodos.

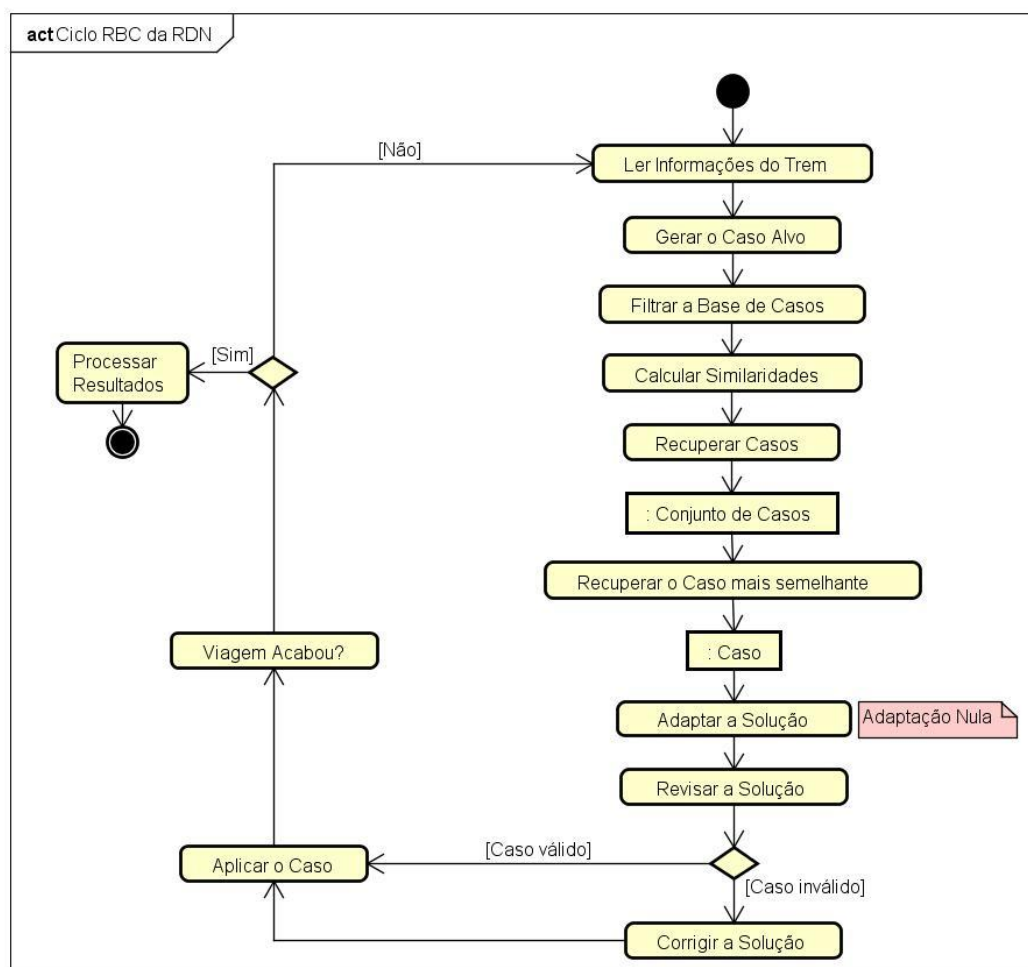
4.3.1 Recuperação em Dois Níveis (RDN)

Como descrito na Seção 2.5.1.1, este é um método em dois passos, na qual a primeiro passo objetiva utilizar filtros e limitações para reduzir ao máximo o número de casos candidatos, para então em seu segundo passo calcular as medidas de similaridade entre os casos.

4.3.1.1 Funcionamento

Nesta recuperação, existe mais uma etapa no diagrama do método RBC (Figura 11), que é a etapa “*Filtragem da Base de Casos*”. Com exceção dessa nova etapa, todo o processo de recuperação será o mesmo. A Figura 13 ilustra o funcionamento do Ciclo RBC da RDN.

Figura 13 - Diagrama do Ciclo RBC da RDN.

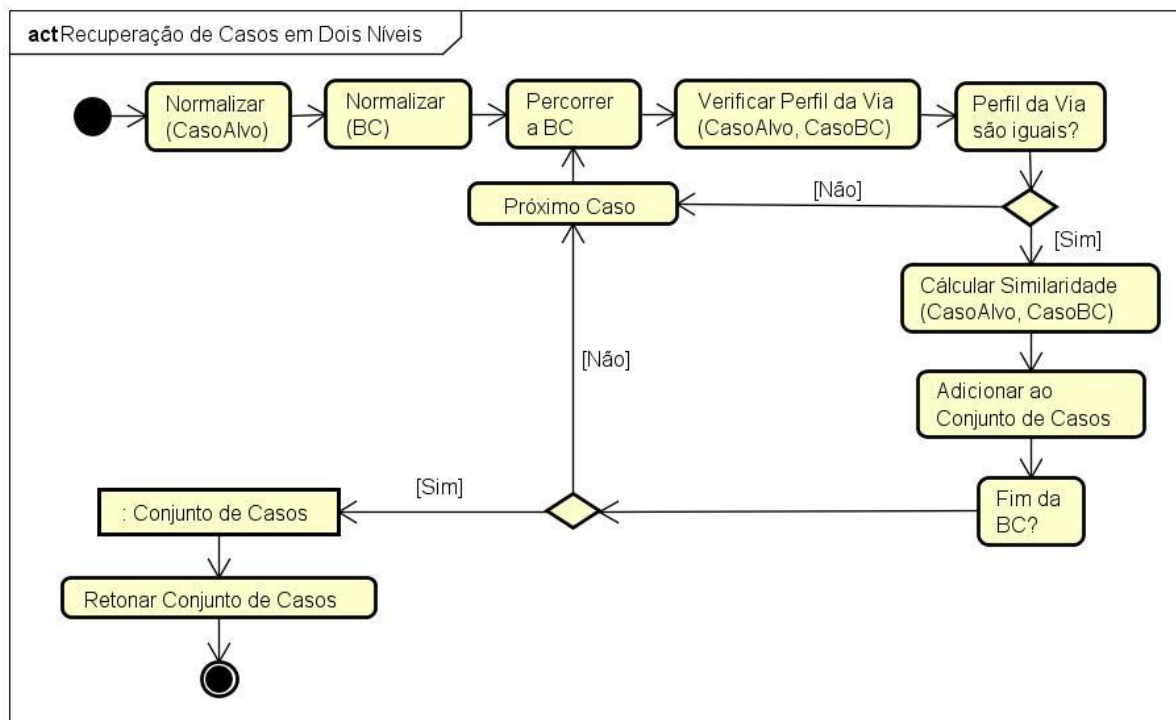


Fonte: Autoria própria.

Após ser gerado um caso alvo, para a realização da filtragem é selecionado um atributo do caso, que geralmente é o atributo de maior relevância, para determinar quais são os casos potencialmente úteis em uma BC. Neste trabalho foi selecionado o atributo *Perfil*. O resultado da filtragem será um novo *Conjunto de Casos* (BC Filtrada). Este conjunto possuirá somente os casos com o valor do atributo

selecionado igual ao do caso alvo. A Figura 14 ilustra o funcionamento desta recuperação.

Figura 14 - Diagrama da Recuperação de Casos em Dois Níveis.



Fonte: Autoria própria.

Em comparação com o diagrama da Recuperação de Casos da Figura 12, a *Recuperação de Casos em Dois Níveis* haverá a adição da etapa “*Verificar Perfil da Via()*”. Essa etapa fará a filtragem dos casos, verificando se o atributo *Perfil* de ambos os casos são iguais. Se forem, é calculada a similaridade, e o caso da BC é adicionado ao *Conjunto de Casos*, do contrário o próximo caso da BC é verificado.

Por utilizar atributo fixo, não foi detectada a necessidade de gerar uma nova BC somente com os casos filtrados antes de calcular as similaridades. No intuito de economizar processamento, o filtro é aplicado em formato de condição, agilizando o processo. Desta forma, se os casos possuírem o atributo selecionado em comum (que neste exemplo é o *Perfil*), a similaridade é calculada e o caso da BC é adicionado ao *Conjunto de Casos*. A partir deste conjunto, o método segue o fluxo definido na Figura 13.

4.3.1.2 Exemplo

Para exemplificar uma RDN, será suposto uma BC com 10 casos solucionados, estes casos são ilustrados no Quadro 1.

Quadro 1 - Exemplo de uma BC.

Base de Casos											
CASO	EAC	Perfil	KM	NL	NW	IS	FS	MS	%R	CL	SOLUÇÃO (S)
1	Acelerar	Em Nível	320,465	3	58	11	22	45	-0,41	0,605	{<3,0>,<3,4>,<5,24>,<5,35>,<5,49>}
2	Manter	Declive	314,763	3	40	13	13	25	0,37	0,365	{<7,49>,<7,64>,<7,79>,<7,96>,<7,109>}
3	Freiar	Declive	400,742	3	60	16	14	25	-0,31	0,385	{<5,0>,<5,4>,<3,24>,<3,35>,<3,49>}
4	Freiar	Declive	473,643	2	32	10	7	15	-0,33	0,393	{<7,3>,<7,11>,<5,19>,<5,28>,<3,37>}
5	Acelerar	Aclive	284,567	3	54	17	19	30	-0,27	0,275	{<1,0>,<1,7>,<2,18>,<3,31>,<3,49>}
6	Acelerar	Aclive	365,634	1	22	14	15	30	-0,34	0,475	{<1,22>,<1,34>,<1,55>,<3,69>,<3,83>}
7	Manter	Aclive	214,563	3	49	25	26	30	-0,14	0,426	{<2,47>,<2,56>,<2,68>,<2,80>,<3,96>}
8	Acelerar	Aclive	363,732	5	79	27	30	40	-0,44	0,365	{<4,29>,<5,39>,<5,50>,<5,62>,<6,83>}
9	Manter	Em Nível	500,434	4	53	35	37	40	-0,51	0,264	{<3,9>,<3,15>,<4,24>,<4,32>,<4,44>}
10	Manter	Em Nível	101,899	2	34	7	14	20	-0,19	0,195	{<1,2>,<1,9>,<1,19>,<1,28>,<1,37>}

Fonte: Autoria própria.

Selecionando o atributo Perfil da Via (*Perfil*) como critério de filtragem, será efetuada uma recuperação para o caso alvo C dado por

$$C = \left\{ (EAC, Manter), (Perfil, Declive), (KM, 378,543), (NL, 4), \right. \\ \left. ((NW, 39), (IS, 20), (FS, 15), (MS, 30), (%R, -0,22), (CL, 0,500)) \right\}$$

Porém antes de efetuar os cálculos, o caso alvo e a BC devem ser normalizados. A BC normalizada é ilustrada no Quadro 2.

Quadro 2 - Exemplo de uma BC Normalizada.

Base de Casos											
CASO	EAC	Perfil	KM	NL	NW	IS	FS	MS	%R	CL	SOLUÇÃO (S)
1	Acelerar	Em Nível	0,320	0,3	0,58	0,11	0,22	0,45	-0,41	0,605	{<3,0>,<3,4>,<5,24>,<5,35>,<5,49>}
2	Manter	Declive	0,315	0,3	0,4	0,13	0,13	0,25	0,37	0,365	{<7,49>,<7,64>,<7,79>,<7,96>,<7,109>}
3	Freiar	Declive	0,401	0,3	0,6	0,16	0,14	0,25	-0,31	0,385	{<5,0>,<5,4>,<3,24>,<3,35>,<3,49>}
4	Freiar	Declive	0,474	0,2	0,32	0,1	0,07	0,15	-0,33	0,393	{<7,3>,<7,11>,<5,19>,<5,28>,<3,37>}
5	Acelerar	Aclive	0,285	0,3	0,54	0,17	0,19	0,3	-0,27	0,275	{<1,0>,<1,7>,<2,18>,<3,31>,<3,49>}
6	Acelerar	Aclive	0,366	0,1	0,22	0,14	0,15	0,3	-0,34	0,475	{<1,22>,<1,34>,<1,55>,<3,69>,<3,83>}
7	Manter	Aclive	0,215	0,3	0,49	0,25	0,26	0,3	-0,14	0,426	{<2,47>,<2,56>,<2,68>,<2,80>,<3,96>}
8	Acelerar	Aclive	0,364	0,5	0,79	0,27	0,3	0,4	-0,44	0,365	{<4,29>,<5,39>,<5,50>,<5,62>,<6,83>}
9	Manter	Em Nível	0,500	0,4	0,53	0,35	0,37	0,4	-0,51	0,264	{<3,9>,<3,15>,<4,24>,<4,32>,<4,44>}
10	Manter	Em Nível	0,102	0,2	0,34	0,07	0,14	0,2	-0,19	0,195	{<1,2>,<1,9>,<1,19>,<1,28>,<1,37>}

Fonte: Autoria própria.

E o caso alvo normalizado C é dado por

$$C = \left\{ (EAC, Manter), (Perfil, Declive), (KM, 0,379), (NL, 0,4), (NW, 0,39), (IS, 0,2), (FS, 0,15), (MS, 0,3), (\%R, -0,22), (CL, 0,5) \right\}$$

A normalização é efetuada dividindo o valor do atributo com o maior valor de sua ordem numeral. Por exemplo, o Caso 1 tem seu atributo KM com valor de 320.465, então o maior valor de sua ordem numeral é 1000. Dividindo os valores, se obtém o valor normalizado de 0,320.

Com a BC e o caso alvo normalizados, é então realizada a filtragem dos casos. O Quadro 3 demonstra o resultado desta filtragem nesta BC.

Quadro 3 - Resultado da filtragem da RDN.

Base de Casos Filtrada											
CASO	EAC	Perfil	KM	NL	NW	IS	FS	MS	%R	CL	SOLUÇÃO (S)
2	Manter	Declive	0,320	0,3	0,58	0,11	0,22	0,45	0,37	0,365	{<7,49>,<7,64>,<7,79>,<7,96>,<7,109>}
3	Frear	Declive	0,500	0,4	0,53	0,35	0,37	0,4	-0,31	0,385	{<5,0>,<5,4>,<3,24>,<3,35>,<3,49>}
4	Frear	Declive	0,102	0,2	0,34	0,07	0,14	0,2	-0,33	0,393	{<7,3>,<7,11>,<5,19>,<5,28>,<3,37>}

Fonte: Autoria própria.

Após a filtragem, a similaridade é calculada para todos os atributos dos casos. Utilizando a formula da Distância Euclidiana para efetuar os cálculos, obtém-se o seguinte resultado demonstrado pelo Quadro 4.

Quadro 4 - Resultado dos cálculos de similaridade na BC filtrada.

Base de Casos Filtrada												
CASO	EAC	Perfil	KM	NL	NW	IS	FS	MS	%R	CL	SOLUÇÃO (S)	SIMILARIDADE
2	Manter	Declive	0,320	0,3	0,58	0,11	0,22	0,45	0,37	0,365	{<7,49>,<7,64>,<7,79>,<7,96>,<7,109>}	0,61
3	Frear	Declive	0,500	0,4	0,53	0,35	0,37	0,4	-0,31	0,385	{<5,0>,<5,4>,<3,24>,<3,35>,<3,49>}	0,15
4	Frear	Declive	0,102	0,2	0,34	0,07	0,14	0,2	-0,33	0,393	{<7,3>,<7,11>,<5,19>,<5,28>,<3,37>}	0,15

Fonte: Autoria própria.

Para determinar qual dos casos recuperados é o mais semelhante ao caso alvo, é preciso aplicar a formula da Equação 4:

$$d = 1 - \text{similaridade} \quad (4)$$

O valor da similaridade (através da Distância Euclidiana) indica quanto dois casos são distintos entre si. Desta forma, aplicando a Equação 4, quanto maior for o valor de d (mais próximo de 1) maior será a semelhança entre dois casos. Os resultados de d para a BC filtrada são:

- Caso 2, $d = 0,39$;
- Caso 3, $d = 0,85$;
- Caso 4, $d = 0,85$.

Neste exemplo, os Casos 3 e 4 empataram, não importando qual caso seja escolhido já que ambos são igualmente semelhantes. Desta forma, é suposto que o Caso 3 foi selecionado como o mais semelhante ao caso alvo C. Então o sistema seguirá seu fluxo normal, prosseguindo para a etapa de Adaptação e posteriormente a etapa de Revisão, para eventualmente aplicar o caso recuperado.

4.3.2 Recuperação em Dois Níveis Dinâmica (RDND)

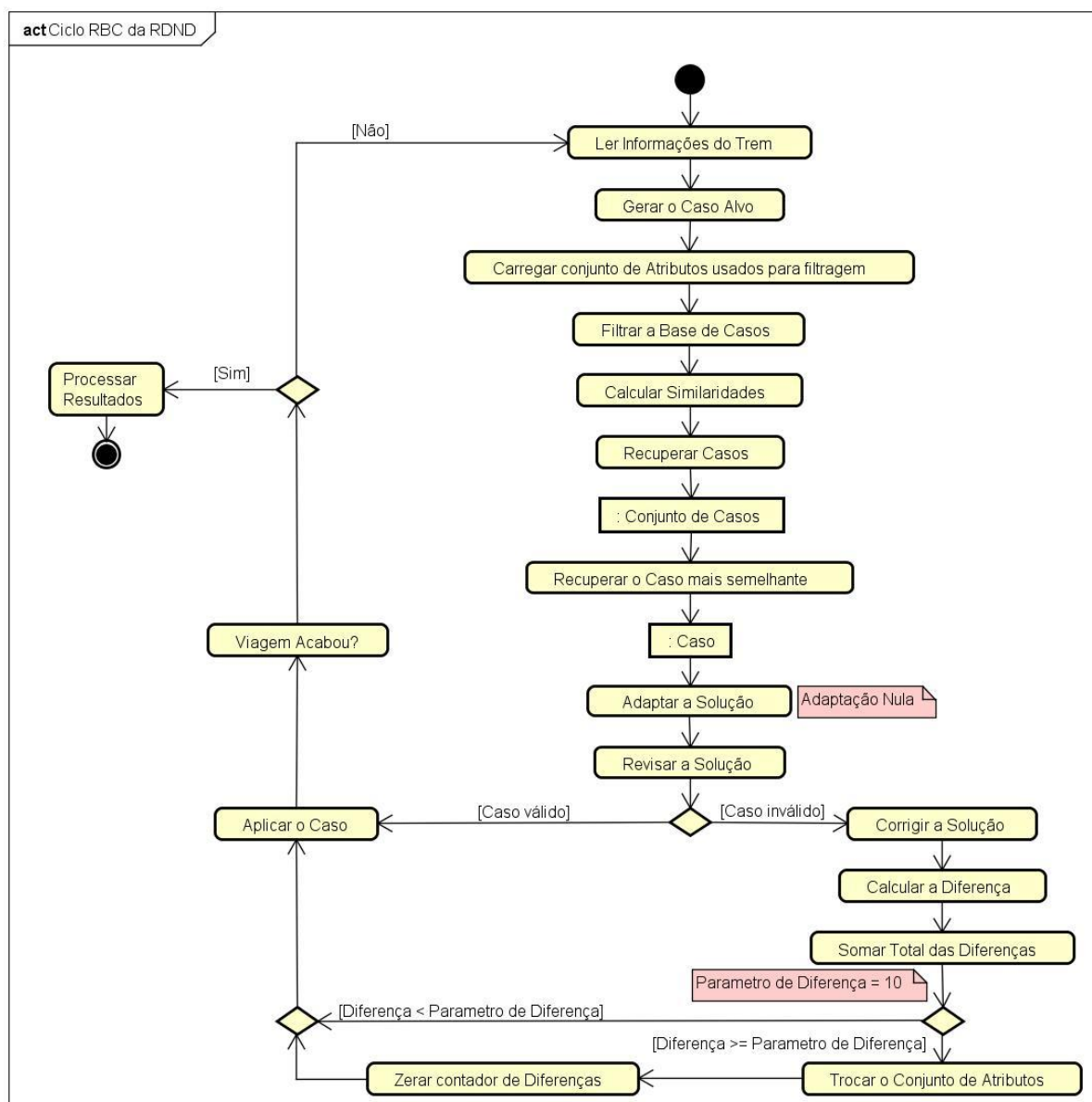
A RDN clássica é um método simples e de fácil implementação, consistindo basicamente em realizar uma filtragem antes de calcular as similaridades e então selecionar o caso mais parecido com o problema alvo. Devido a isso, foi decidido realizar a implementação de uma RDND, que consiste em uma RDN onde os critérios selecionados para a filtragem dos casos sejam variantes e fiquem alternando durante o decorrer da viagem.

4.3.2.1 Funcionamento

Nesta recuperação além da adição da etapa de “*Filtragem da Base de Casos*”, existem diversas etapas a mais. Uma das etapas foi a de “*Carregar conjunto de Atributos usados para a Filtragem*”, que será executada para carregar o conjunto de critérios e selecionar o primeiro critério do conjunto.

Além disso, diversas etapas foram adicionadas após a etapa de “*Corrigir a Solução*”, que servirão para verificar se as recuperações realizadas estão sendo boas e precisas. Se for constatado que não, então é realizada a troca dos critérios. O diagrama do Ciclo RBC da RDND é ilustrado na Figura 15.

Figura 15 - Diagrama da RDND.



Fonte: Autoria própria.

Diferente da RDN Clássica, para a implementação da RDND novas regras e parâmetros serão adicionados ao seu funcionamento, estes são:

- Um conjunto de critérios de filtragem cf ;
- Um parâmetro de erro e ;
- Um limiar de valores σ .

O conjunto de critérios de filtragem (cf) consiste em seleccionar previamente um número n de critérios que devem ser atributos do caso. Os critérios serão utilizados

para realizar as filtragens da BC durante o decorrer da viagem, alternando entre os critérios presentes no conjunto toda vez que seja constatado que as recuperações não estejam obtendo um bom desempenho.

A escolha dos critérios que consistem cf , deve ser igual a escolha do critério de filtragem da RDN Clássica. Os critérios selecionados devem conter informações relevantes do domínio do problema, e por meio deles deve ser possível diminuir o número de casos candidatos em uma BC.

O principal objetivo e também a maior vantagem da RDND, é possibilitar a seleção de diversos critérios de filtragem distintos, aplicando-os em diferentes momentos da execução do algoritmo. Seu comportamento difere da RDN Clássica, onde é selecionado um único critério fixo para todas as verificações do algoritmo.

O cf pode ser formado de atributos únicos, pares ou até mesmo trios, quartetos, etc. Não existe uma forma concreta de selecionar qual a estrutura cf deve assumir ou quantos critérios devem ser selecionados, a escolha dependerá do domínio da aplicação e da quantidade de atributos que formam o caso. Para formar cf , o conjunto é definido por

$$cf = \{[c_0], [c_1], \dots, [c_n]\} = \{[c_i]\}_0^n = \{cf_i\}_0^n$$

Para exemplificar o funcionamento de cf na RDND iremos considerar um conjunto de critérios formado por pares de atributos, representado por

$$cf = \{[EAC, Perfil], [IS, FS], [MS, \%R]\}$$

Ao iniciar uma viagem, a filtragem de casos da BC começará sendo feita com o par de atributos $[EAC, Perfil]$. No momento em que for constatado que os atributos selecionados não estejam realizando boas recuperações, os atributos serão trocados pelo próximo par presente no conjunto cf , neste caso o par $[IS, FS]$ e posteriormente o par $[MS, \%R]$. Quando o par selecionado for o último existente no conjunto e houver a necessidade de uma nova troca, será selecionado o primeiro par do conjunto novamente e este ciclo será repetido até o fim da viagem.

Porém é preciso saber quando os atributos de *cf* devem ser trocados, e para isso é utilizado o *parâmetro de erro e*. Através desse parâmetro é definido um valor fixo que servirá de base para realizar as trocas dos atributos presentes em *cf*. Selecionado um par de atributos e realizada a recuperação e adaptação (que neste trabalho será a Adaptação Nula) de um caso, este passará por uma validação na etapa de Revisão afim de verificar se é possível aplicá-lo.

Se a Revisão constatar que o caso recuperado não é adequado a ser aplicado diretamente, algumas modificações serão realizadas no conjunto *S* para tornar esse caso aplicável ao problema atual. Após realizar essas modificações, é comparado o caso recuperado antes e depois da etapa de Revisão, com o propósito de descobrir a diferença gerada. Essa diferença é somada e armazenada em um controlador *X* e então o fluxo RBC seguirá normalmente, repetindo o processo de verificação da diferença toda vez que um novo caso for recuperado e validado. No momento em que $X \geq e$, será selecionado um novo par de atributos presente em *cf* e então *X* assumirá valor igual à 0.

Outro parâmetro importante utilizado na implementação da recuperação é o *limiar* σ , parâmetro este que será utilizado durante a filtragem. Esse parâmetro terá o objetivo de impor uma tolerância na diferença dos valores dos atributos, pois não há garantia que existirão casos na BC com atributos que possuam valores idênticos ao do caso alvo, podendo ocorrer de casos muito similares não serem selecionados por não passarem na filtragem devido a pequenas diferenças de valores.

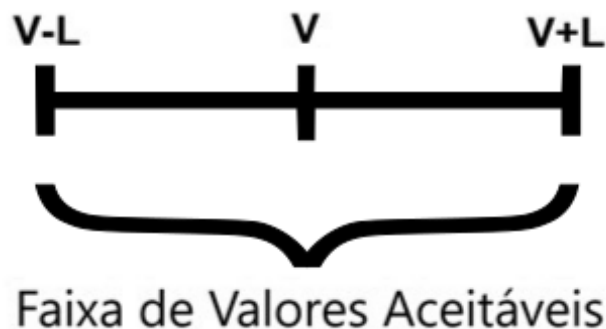
Devido a isso, essa tolerância busca abranger os potenciais casos antes de realizar os cálculos de similaridade. Dado um valor *V* de um atributo do caso, o σ irá determinar uma faixa de valores aceitáveis para mais e para menos de *V*, conforme ilustra a Figura 16.

A aplicação de σ será feita nos atributos selecionados no conjunto *cf*. Para o caso definido na seção 4.2, os atributos Ação Executada (EAC) e Perfil da Via (Perfil) assumem valores descritivos (*strings*) e para eles será necessário realizar um tratamento diferente. Cada um destes atributos assume um valor de um conjunto específico de valores pré-determinados. Em virtude disso será atribuído um valor numérico para representar cada um dos possíveis valores que cada um destes atributos pode assumir. A representação será determinada por:

- EAC: Frear (0.0), Reduzir (0.33), Manter (0.66), Acelerar (1.0).

- Perfil: Declive (0.0), Em Nível (0.5), Active (1.0).

Figura 16 - Aplicação do Limiar.



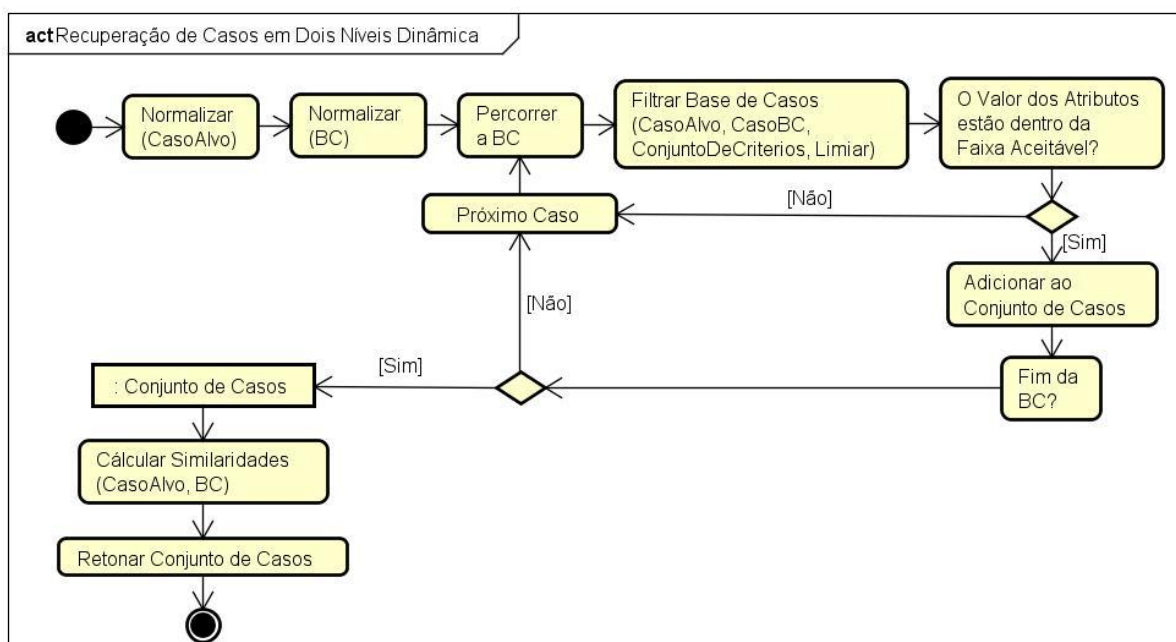
Fonte: Autoria própria.

O σ será aplicado sempre que os atributos já estiverem normalizados, em razão disso o valor de σ não pode ser maior que 1. O objetivo utilizando esse parâmetro é abranger a quantidade dos possíveis casos úteis, porém não é recomendado que σ assumam valores altos, pois isso pode gerar brechas para casos não úteis também serem selecionados na filtragem. É necessário escolher um valor que ajude a seleção dos casos sem prejudicar sua eficácia.

Com as novas regras e parâmetros do RDND definidos os seus valores deverão ser configurados na etapa “*Carregar Configurações do RBC*” ilustrada no diagrama da Figura 10. O funcionamento deste método será semelhante ao do RDN. O procedimento se baseia em percorrer a BC e fazer uma verificação para saber se os valores dos atributos informados em *cf* estão dentro da faixa aceitável permitida pelo σ . Essa verificação é realizada para cada caso da BC. Se os valores forem aceitáveis para todos os atributos de *cf*, os casos são adicionados em um novo *Conjunto de Casos* (BC Filtrada) (cf. Figura 17).

Se ao final do processo, o número de Casos contidos no *Conjunto de Casos* for maior que zero, este conjunto será retornada ao *Ciclo RBC do RDND* para ser utilizado, do contrário à BC original é retornada.

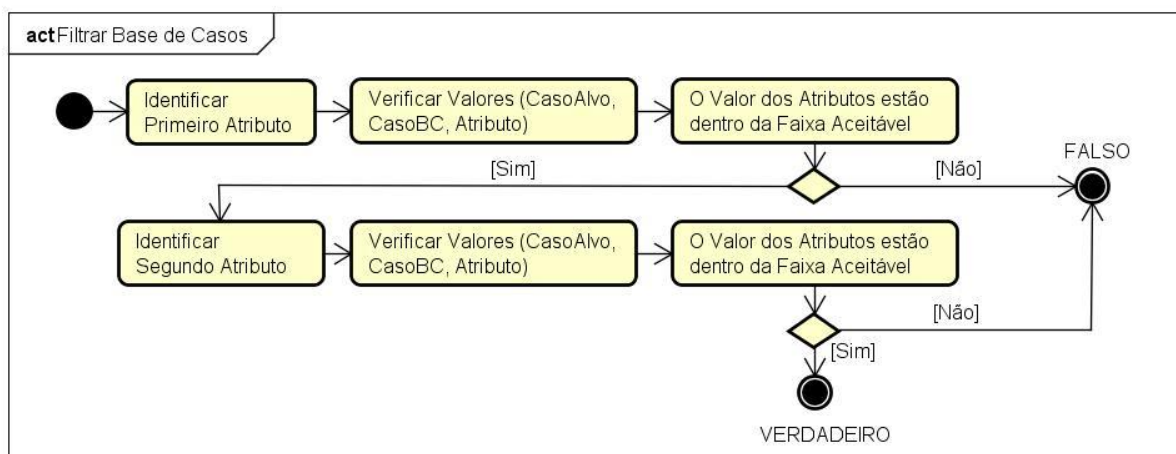
Figura 17 - Diagrama da Recuperação em Dois Níveis Dinâmica.



Fonte: Autoria própria.

Nesta recuperação, a etapa de “*Filtrar Base de Casos()*” é a responsável por selecionar os casos que serão úteis ou não para o problema alvo enfrentado. O diagrama da Figura 18 demonstra o funcionamento desta etapa. Como neste trabalho, foi decidido trabalhar com uma *cf* representada por pares de atributos, o diagrama ilustra apenas duas verificações, uma para cada atributo de *cf*.

Figura 18 - Diagrama da Filtragem da Base de Casos.

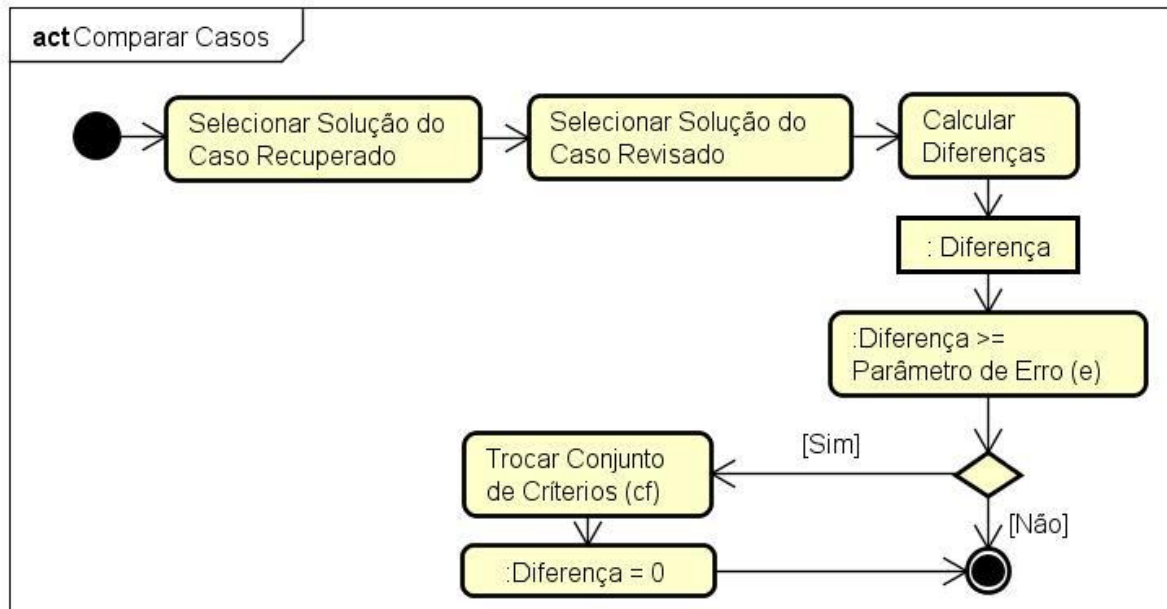


Fonte: Autoria própria.

Essa etapa irá identificar qual o primeiro e segundo atributo a serem verificados entre os casos, e calcula se seus valores estão dentro da faixa aceitável. A etapa “*Verificar Valores()*” terá a tarefa de aplicar a verificação do limiar. Se os valores do caso da BC estiverem dentro da faixa de valores para mais ou para menos do caso alvo, este é considerado um caso válido e então a função retorna VERDADEIRO informando que o caso é útil, do contrário é retornado FALSO informando que não é um caso útil. Se os atributos informados para a etapa “*Filtrar Base de Casos()*” não pertencerem ao caso, é retornado FALSO automaticamente.

Após a filtragem da BC, o método seguirá o fluxo definido pelo diagrama da Figura 15. O caso mais semelhante será adaptado (Adaptação Nula) e revisado. Após a revisão, se for constatado que o caso não é válido, este caso sofrerá alterações no conjunto solução *S* para torná-lo um caso válido. Após estas modificações, o método fará a verificação para saber as diferenças entre o antes e depois da validação (cf. Figura 19).

Figura 19 - Diagrama da Comparação de Casos.



Fonte: Autoria própria.

O valor destas diferenças geradas serão armazenadas no controlador *diferença*, e assim que $diferença \geq parametro\ de\ erro$, os pares de atributos selecionados são trocados, o contador *diferença* será zerado e o fluxo do método seguirá normalmente.

4.3.2.2 Exemplo

Para exemplificar o funcionamento da RDND, será utilizada a mesma BC e o mesmo caso alvo (já normalizados) utilizados no exemplo da RDN, ambos definidos na seção 4.3.1.2. Os parâmetros e variáveis necessárias serão configuradas da seguinte maneira:

- $cf = \{[EAC, Perfil], [IS, FS], [MS, \%R]\}$.
- $e = 10$.
- $\sigma = 0,5$.
- Distância Euclidiana para cálculo de similaridade.

Em um problema real, a cada interação do método RBC o caso alvo seria diferente, pois representaria uma diferente situação do trem em uma via férrea. Porém para esse exemplo será considerado um caso alvo fixo com o objetivo de mostrar as diferentes recuperações utilizando diferentes critérios de filtragem. Para a primeira filtragem, será utilizado o par de atributos {EAC, Perfil} do conjunto cf . O resultado é mostrado no Quadro 5.

Quadro 5 - Resultado da filtragem utilizando o par de atributos {EAC, Perfil}.

Base de Casos Filtrada												
CASO	EAC	Perfil	KM	NL	NW	IS	FS	MS	%R	CL	SOLUÇÃO (S)	SIMILARIDADE
1	Acelerar	Em Nível	0,320	0,3	0,58	0,11	0,22	0,45	-0,41	0,605	{<3,0>, <3,4>, <5,24>, <5,35>, <5,49>}	0,22
2	Manter	Declive	0,315	0,3	0,4	0,13	0,13	0,25	0,37	0,365	{<7,49>, <7,64>, <7,79>, <7,96>, <7,109>}	0,61
9	Manter	Em Nível	0,500	0,4	0,53	0,35	0,37	0,4	-0,51	0,264	{<3,9>, <3,15>, <4,24>, <4,32>, <4,44>}	0,37
10	Manter	Em Nível	0,102	0,2	0,34	0,07	0,14	0,2	-0,19	0,195	{<1,2>, <1,9>, <1,19>, <1,28>, <1,37>}	0,31

Fonte: Autoria própria.

O resultado desta filtragem trouxe quatro casos distintos. Aplicando a Equação 4, temos os seguintes resultados:

- Caso 1, $d = 0,78$.
- Caso 2, $d = 0,39$.
- Caso 9, $d = 0,63$.
- Caso 10, $d = 0,69$.

Para essa recuperação o Caso 1 é o mais similar, com $d = 0,78$. Normalmente o caso recuperado seria levado a etapa de Adaptação (Adaptação Nula) e depois Revisão onde seria validado e talvez modificado (se necessário). Posteriormente

seriam calculadas as diferenças do caso, esta diferença seria somada e armazenada ao controlador X .

Porém para continuarmos o exemplo, será considerado que após esta recuperação o controlador X atingiu o valor de dez. Desta forma a condição $X \geq e$ é atendida, o par de atributos {IS, FS} é selecionado e X assume valor igual a zero. Ao executar uma nova filtragem o resultado é mostrado no Quadro 6.

Quadro 6 - Resultado da Filtragem utilizando o par de atributos {IS, FS}.

Base de Casos Filtrada												
CASO	EAC	Perfil	KM	NL	NW	IS	FS	MS	%R	CL	SOLUÇÃO (S)	SIMILARIDADE
3	Frear	Declive	0,401	0,3	0,6	0,16	0,14	0,25	-0,31	0,385	{<5,0>,<5,4>,<3,24>,<3,35>,<3,49>}	0,15
5	Acelerar	Aclive	0,285	0,3	0,54	0,17	0,19	0,3	-0,27	0,275	{<1,0>,<1,7>,<2,18>,<3,31>,<3,49>}	0,23

Fonte: Autoria própria.

Com o novo par de atributos para realizar a filtragem, desta vez somente dois casos foram selecionados. Aplicando novamente a Equação 4 os resultados são:

- Caso 3, $d = 0,85$;
- Caso 5, $d = 0,77$.

Para esta recuperação o Caso 3 é o mais similar, com $d = 0,85$. Pode-se notar que com essa mudança a quantidade de casos recuperados diminuiu, sendo necessário calcular a similaridade menos vezes que com o par de atributos {EAC, Perfil}.

Considerando novamente controlador X atingiu o valor de dez, e a condição $X \geq e$ é atendida, será selecionado o par de atributos {MS, %R} e então X assume valor igual a zero novamente. Por fim, a nova filtragem tem o resultado mostrado no Quadro 7.

Quadro 7 - Resultado da Filtragem utilizando o par de atributos {MS, %R}.

Base de Casos Filtrada												
CASO	EAC	Perfil	KM	NL	NW	IS	FS	MS	%R	CL	SOLUÇÃO (S)	SIMILARIDADE
5	Acelerar	Aclive	0,285	0,3	0,54	0,17	0,19	0,3	-0,27	0,275	{<1,0>,<1,7>,<2,18>,<3,31>,<3,49>}	0,23

Fonte: Autoria própria.

Desta vez apenas um caso foi selecionado pela filtragem, aplicando a Equação 4 temos $d = 0,77$. Dentre todas as filtrações realizadas, a que utilizou o par {MS, %R} foi a mais rápida, pois com apenas um caso selecionado pela filtragem este é imediatamente considerado o mais similar. Porém se comparado com as

recuperações anteriores, está não foi a mais precisa, pois o par {IS, FS} conseguiu recuperar um caso com $d = 0,85$.

4.4 CONSIDERAÇÕES FINAIS

A atividade de condução de trens é uma tarefa complexa, pois envolve diversos fatores, como o número de locomotivas e vagões que compõem o trem, as características da via, e a situação que o trem se encontra em um determinado instante da via férrea.

Sendo assim, para formar um Caso foi necessário selecionar as informações determinantes na movimentação de um trem, escolhendo somente as características (tanto do trem quanto da via férrea) que melhor representassem a movimentação de um trem, assim como qual atividade faz com que essas características sejam alteradas.

Na recuperação de casos, além da utilização da clássica Recuperação em Dois Níveis (RDN), que consiste em filtrar candidatos e calcular similaridades, foi criado um novo método de recuperação se baseando no RDN. Assim com esse novo método, foi possível adicionar uma flexibilidade no RDN, permitindo selecionar mais critérios de filtragem, onde estes são trocados durante a execução de uma viagem.

5 EXPERIMENTOS E RESULTADOS

O método RBC foi avaliado por diferentes experimentos no domínio da condução de trens em um ambiente simulado. É importante lembrar que o principal foco deste trabalho foi somente na etapa de Recuperação, assumindo que o funcionamento das demais etapas estarão de acordo com o que foi descrito por Borges (2015a).

Para realização da implementação do novo método de recuperação, foi utilizada a linguagem *java* através da IDE NetBeans 8.2. A execução dos testes foi efetuada em um ambiente simulado criado por Borges (2015a). O simulador de condução de trens também foi desenvolvido em linguagem *java* e executado no NetBeans 8.2.

Os testes realizados tentaram determinar o quão bem o método RBC conseguia recuperar casos e aplicá-los sem que fosse preciso adaptá-los. Para isso foi utilizado Adaptação Nula, pois assim o caso recuperado não é modificado. Porém, se a etapa de Revisão determinar que a solução do caso não seja boa, causando velocidade acima do permitido ou patinação na via férrea, a etapa realiza as medidas necessárias para evitar tais problemas.

Por meio destes testes, será analisado o número de vezes que um caso recuperado precisou ou não ser modificado pela Revisão, o número de trocas realizadas no conjunto de critérios e o grau de similaridade dos casos que estejam sendo recuperados. Além disso, será verificado qual foi a porcentagem de redução do consumo de combustível que o método adquiriu em comparação com o gasto de uma viagem real.

5.1 CENÁRIOS DE TESTE

Os experimentos foram conduzidos no simulador de condução de trens criado por Borges (2015a), que recria computacionalmente características de trens, via férreas e ações de maquinistas. Todas estas informações foram retiradas de cenários reais para simular a condução de trens. Em sua pesquisa, Borges (2015a) define duas vias férreas distintas (ST_1 e ST_2), cada uma com características próprias, onde são realizadas as conduções. Porém para a realização dos testes neste trabalho, foi

utilizado somente a via férrea ST_1 , pois o intuito é aplicar diferentes métodos de recuperação e analisar os diferentes resultados produzidos em uma mesma via.

Vale ressaltar que as informações a respeito da via férrea (curvas, desníveis e velocidades permitidas) não serão abordadas ou discutidas neste trabalho, pois são dados de um ambiente real pré-definido. O objetivo dos experimentos é, dado um trem nesta via férrea, analisar uma determinada situação problemática e recuperar uma solução para tal situação.

Como a via férrea será sempre a mesma, será realizada a condução de trens utilizando diferentes combinações de locomotivas e vagões, estas combinações podem conter vagões vazios e/ou carregados. Assim como vias férreas diferentes, tais variações na composição dos trens também torna a condução uma tarefa única em cada viagem. Essas diferenças irão ocorrer devido à diferença no peso total da composição do trem, o que influencia nas ações (Acelerar, Manter, Reduzir ou Frear) que serão tomadas em cada trecho da via.

Nos experimentos realizados, foram utilizadas as composições de trens definidas pela Tabela 1. Vale ressaltar que as características dos trens utilizados nos experimentos representam dados reais, onde viagens já foram executadas com tais configurações de trens. Assim como a via, as informações a respeito das composições não serão abordadas ou discutidas.

Tabela 1 - Configuração dos trens utilizados nos experimentos.

Configuração do Trem	Quantidade		Peso Total (T)
	Locomotivas	Vagões	
1	3	58	6278
2	4	100	6342
3	4	58	6541
4	2	31	3426
5	3	47	5199
6	2	31	3441
7	5	59	6579
8	2	28	3118

Fonte: Borges (2015a).

Com a via férrea e as composições dos trens definidas, o próximo passo é definir as configurações dos métodos de recuperação RDN e RDND. Para RDN, a única configuração necessária é escolher qual atributo será utilizado como critério de

filtragem para as viagens. O critério selecionado foi o *Perfil*, pois com essa informação é possível determinar um tipo de ação padrão relacionada. Por exemplo, em uma subida o comum é acelerar o trem para que o mesmo tenha força para atravessar este trecho.

Para RDND, é necessário definir como será formado (atributos únicos, pares, trios, etc) e quais serão os atributos que irão formar *cf*, além de definir o valor dos parâmetros e e σ . Para *cf*, o conjunto será formado por pares de atributos, e os atributos escolhidos para formarem os pares, foram escolhidos de forma que apresentassem uma “ligação” entre eles. Desta forma, *cf* será organizado da seguinte maneira:

- **Primeiro Par:** ação executada (EAC) e tipo do perfil a percorrer (Perfil).
- **Segundo Par:** velocidade inicial (IS) e velocidade final (FS).
- **Terceiro Par:** percentual de rampa (%R) e deslocamento total (CL).

O primeiro par será formado por EAC e Perfil, pois como citado anteriormente, os atributos possuem uma relação, uma vez que geralmente a ação a ser tomada pelo maquinista é de acordo com o perfil da via que o trem está percorrendo no momento. O segundo par será formado por IS e FS, porque assim como o primeiro par os atributos possuem uma relação, que indicam a velocidade que o trem começou e terminou em um determinado trecho. Assim em trechos semelhantes, as velocidades no início e no fim do trecho tendem a ser semelhantes.

Já o terceiro par, formado por MS e %R, não existe uma relação próxima entre os atributos, porém ainda há uma ligação entre eles. Em uma descida com uma porcentagem de rampa fixa, a velocidade máxima da via férrea tende a não mudar, pois o trem terá o mesmo comportamento por todo o trecho. Porém caso a porcentagem da rampa mude, deixando a via mais íngreme por exemplo, a tendência é que a velocidade máxima permitida diminua para evitar o risco de acidentes. Sendo assim, *cf* é definido por

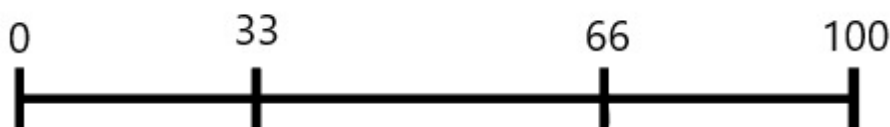
$$cf = \{[EAC, Perfil], [IS, FS], [MS, \%R]\}$$

Não foram formados mais pares para *cf* pois não haviam mais atributos onde seus dados fossem altamente variantes. Para que haja uma filtragem de casos

eficiente, é necessário que os dados dos atributos usados como critério assumam valores distintos, assim somente os casos com valores semelhantes a tal atributo serão selecionados. Se atributos como número de locomotivas (NL) e número de vagões (NW) fossem escolhidos por exemplo, não iriam realizar boas filtragens, pois são atributos com dados fixos e não são alterados em momento nenhum da viagem, desta forma apenas deixaria o processo de recuperação mais lento.

Já o *parâmetro de erro* e , diferente de cf que possui uma configuração única, terá duas configurações diferentes, sendo $e = 33$ e $e = 66$. Assumindo que $e = 100$ é o valor máximo que essa variável pode assumir, 33 e 66 são valores intermediários na faixa de 0 a 100, como demonstra Figura 20.

Figura 20 - Faixa de valores de e .



Fonte: Autoria própria.

Durante o processo RBC, após a etapa de Revisão ser finalizada, o conjunto solução S do caso validado será comparado com a solução do caso inicialmente recuperado. Como o conjunto solução é formado apenas de pontos de aceleração e momentos de aplicação, se ocorrer alterações no caso validado, apenas os pontos de aceleração serão alterados, mantendo os momentos de aplicação intactos.

Então para definir qual a maior diferença que pode ocorrer entre o caso recuperado e o caso validado, é preciso analisar os pontos de aceleração que um trem pode assumir. Grande parte das locomotivas possuem o ponto -1 como o menor ponto de aceleração (pontos negativos representam frenagem), enquanto o maior ponto de aceleração é o ponto 8.

Com isto em consideração, o pior cenário seria onde um caso recuperado possua uma solução cuja o ponto de aceleração é igual a -1, e após a validação o ponto de aceleração passa a ser 8. Desta forma a maior diferença que pode ser gerada entre os pontos de aceleração de um caso recuperado e um caso validado, é igual a 9. O cálculo da diferença é dada pela Equação 5.

$$diferença = |pontoRecuperado - pontoValidado| \quad (5)$$

Por fim o *limiar* σ terá uma configuração semelhante a e . Esse parâmetro será aplicado na filtragem, funcionando como uma “tolerância” na diferença de valores de um atributo entre dois casos. Atributos como *IS* e *FS* por exemplo, podem assumir valores bastante variados, o que gera uma dificuldade para encontrar casos na BC onde esses valores sejam exatamente iguais, podendo resultar em uma filtragem onde nenhum caso da BC fosse selecionado.

Como σ será aplicado sempre com a BC e o caso alvo normalizados, o maior valor que este parâmetro pode assumir é $\sigma = 1$. Os valores definidos para realizar os testes foram $\sigma = 0,33$ e $\sigma = 0,66$, pois assim como e estes são valores intermediários, porém no intervalo entre 0 e 1 como demonstra Figura 21.

Figura 21 - Faixa de valores de σ .



Fonte: Autoria própria.

Com as variáveis e parâmetros configurados para ambos os métodos RDN e RDND, os cenários de teste definidos para avaliar a recuperação de casos são descritos na Tabela 2.

Tabela 2 - Cenários simulados nos experimentos.

Cenário	Método de Recuperação	Configuração do Trem (Cf. Tabela 1)	Parâmetro de Erro (e)	Limiar (σ)
A	RDN	[1;8]	-	-
B	RDND	[1;8]	33	0,33
C	RDND	[1;8]	66	0,66

Fonte: Autoria própria.

Vale ressaltar que todos os cenários serão aplicados na mesma via férrea (ST_1 definida por Borges, 2015a) e utilizarão reuso de casos, ou seja, os casos solucionados durante a execução de uma viagem serão reutilizados nas viagens posteriores.

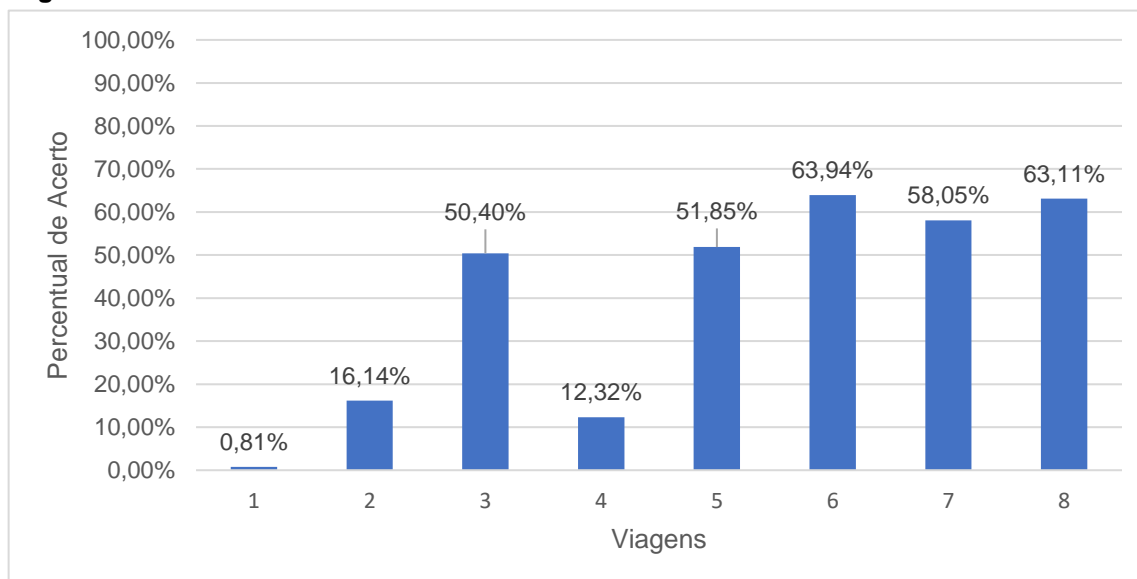
Cada cenário realizará oito viagens, uma viagem para cada composição de trem definida na Tabela 1. É importante deixar claro que para cada cenário, o procedimento começará com a BC contendo 3329 casos (casos sementes), e irá aumentando conforme soluciona problemas no decorrer das viagens. Esta BC inicial foi retirada de Borges (2015a). A adaptação para todos os cenários e todas as viagens será Adaptação Nula. Não serão realizadas paradas com o trem durante a simulação, uma vez iniciada a viagem, o trem não irá parar até o fim da viagem.

Para o cenário A, não foram indicados valores para e e σ pois os parâmetros não são utilizados no método RDN. Os demais cenários que utilizam o método RDND foram construídos para testar duas combinações de valores entre os parâmetros do método, com intuito de descobrir qual configuração gera os melhores resultados. A seguir são discutidos os resultados obtidos por cada cenário.

5.1.1 Cenário A

O primeiro aspecto a ser analisado, será a taxa de acertos que o método obteve ao recuperar um caso e aplicá-lo sem que a etapa de Revisão precisasse realizar modificações na solução do caso. Como será utilizada Adaptação Nula, o objetivo é verificar quão precisas estão sendo as recuperações realizadas. Os acertos indicam casos que não precisaram ter sua solução modificada (cf. Figura 22).

É possível observar que a primeira viagem obteve uma taxa de acertos quase nula. Isso ocorreu devido à BC conter apenas casos sementes que não possuíam soluções boas para a combinação da via férrea com a composição do trem. Conforme o método foi solucionando problemas e adicionando essas soluções à BC, a taxa de acertos foi aumentando nas viagens subsequentes. Tal comportamento é confirmado na viagem 3, onde a composição do trem é muito semelhante à da viagem 1, e a taxa de acertos aumentou em 49,59%.

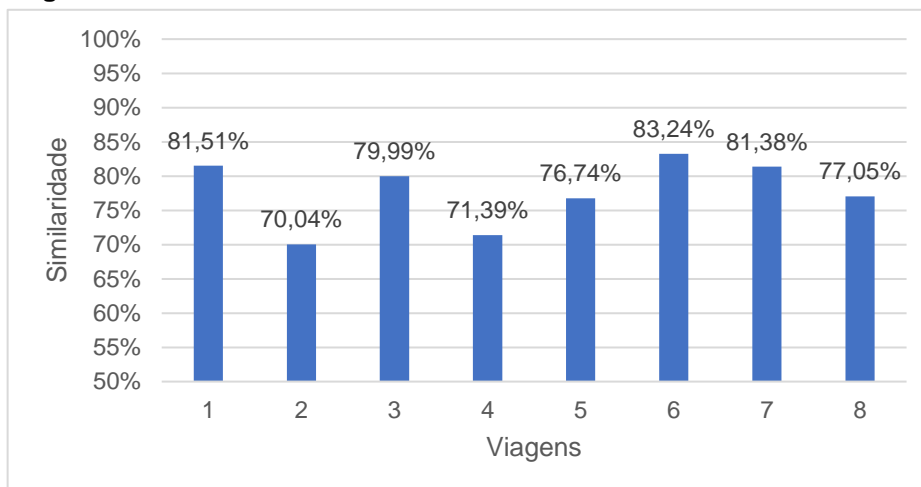
Figura 22 - Gráfico da taxa de acertos do Cenário A.

Fonte: Autoria própria.

Nota-se que na viagem 4 houve uma queda na taxa de acertos. Isso ocorreu devido a composição do trem utilizada, que comparado as composições anteriores possui uma configuração diferente. Esta composição possui um número menor de vagões, e o peso total é quase metade das composições 1, 2 e 3. Após a queda na taxa de acertos na viagem 4, todas as demais viagens obtiveram uma taxa de acertos acima de 50%. Para as oito viagens realizadas, a média do percentual de acerto foi de 39,58%.

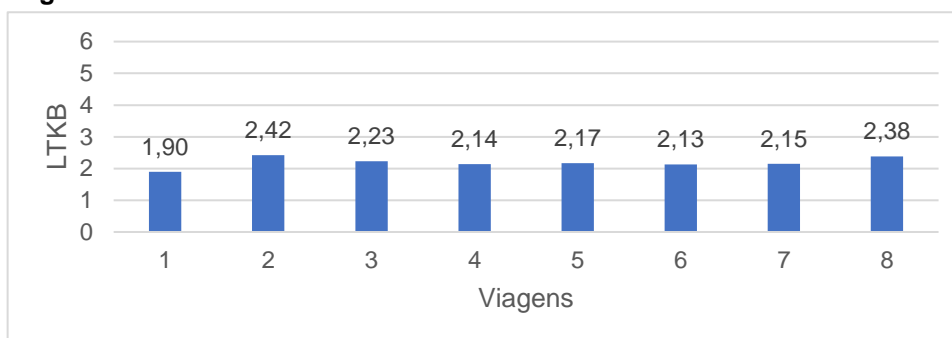
A segunda análise realizada, é quão os casos recuperados foram similares aos problemas enfrentados durante a execução das viagens. A média da similaridade das oito viagens foi de 77,67%. A Figura 23 demonstra a média de semelhança dos casos recuperados pelo método para cada viagem.

Pela Figura 23 pode-se observar que as viagens 2 e 4 foram as que tiveram as menores médias de similaridades dentre todas as viagens. Isso ocorreu devido às características das composições dos trens. A média de vagões para as composições listadas neste trabalho é de 51 vagões, e as composições 2 e 4 são as que possuem as maiores diferenças em relação a essa média, com 100 e 31 vagões respectivamente. Essa discrepância nas características dos trens gerou uma menor taxa de similaridade, porém a média tende a aumentar em viagens futuras com essas mesmas composições, pois a BC possuirá casos solucionados com tais características.

Figura 23 - Gráfico da média de similaridade do Cenário A.

Fonte: Autoria própria.

A terceira análise será sobre os consumos de combustível obtidos nos experimentos. Observou-se que conforme as viagens executadas, o consumo médio foi de 2,19 litros por tonelada bruta transportada (LTKB). Segundo Borges (2015a), o LTKB de uma viagem real possui a média de 6,19. O gráfico da Figura 24 mostra o consumo por viagem realizada.

Figura 24 - Gráfico da média de LTKB do Cenário A.

Fonte: Autoria própria.

Por fim será analisado o consumo de combustível que cada trem teve ao final de sua viagem, comparando com o consumo real de um trem com a mesma composição utilizada na simulação. Os dados de consumo real das composições foram retirados de Borges (2015a). A comparação é apresentada na Tabela 3.

Tabela 3 - Redução de consumo obtido no Cenário A.

Configuração do Trem	LTKB		Redução (CRB - Real) %
	Real	CBR	
1	6,19	1,90	69,38%
2	5,68	2,42	57,33%
3	6,23	2,23	64,21%
4	6,49	2,14	67,03%
5	6,29	2,17	65,48%
6	6,17	2,13	65,50%
7	6,3	2,15	65,85%
8	6,26	2,38	61,97%

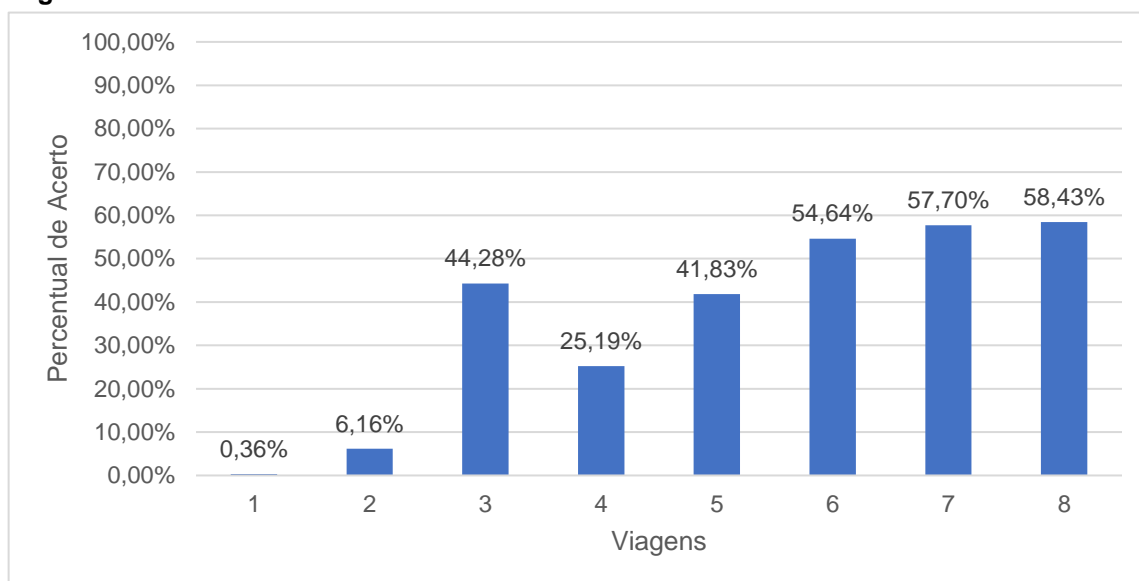
Fonte: Autoria própria.

Pode-se notar que para todos os cenários houve uma redução considerável no consumo de combustível. A utilização do método RBC com a abordagem RDN, apresentou uma média de 64,59% de redução de consumo, um resultado bastante satisfatório. Os próximos cenários a serem analisados utilizarão o método de RDND, variando as suas configurações para que possa ser comparado os diferentes resultados que elas geram.

5.1.2 Cenário B

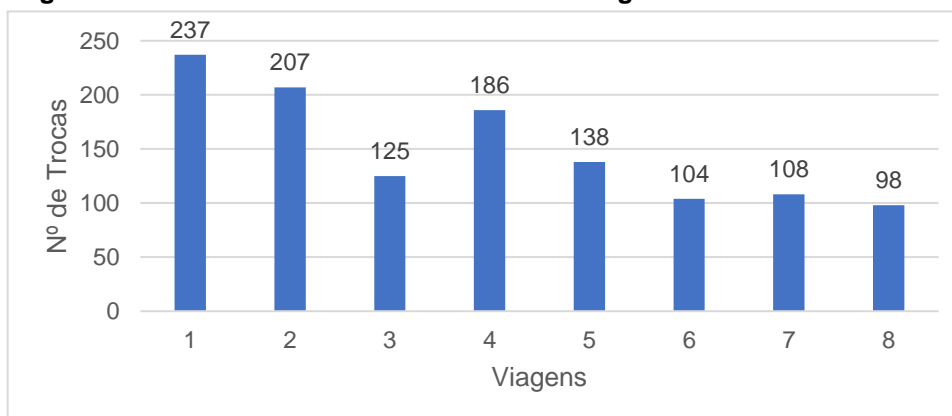
Neste cenário, a análise da taxa de acertos obteve um comportamento semelhante ao Cenário A. A primeira viagem teve uma taxa de acertos quase nula, até mesmo menor que o Cenário A, e foi melhorando conforme solucionava novos problemas (cf. Figura 25).

Porém, a progressão do aprimoramento das recuperações foi menor neste cenário. Na viagem 2, este cenário apresentou um percentual de acerto 10% menor que o cenário anterior, enquanto a viagem 3 mostrou-se 5% menor. Além disso, após a queda na taxa de acertos da viagem 4, as viagens do cenário A demonstraram ser sempre maiores que 50%. O que não ocorre neste cenário, onde as taxas de acertos acima de 50% ocorrem somente da viagem 6 em diante. As médias também se mostraram menores que no Cenário A, com 36,08%.

Figura 25 - Gráfico da taxa de acertos do Cenário B.

Fonte: Autoria própria.

Para a RDND também será analisada a quantidade de troca dos critérios de filtragens utilizados. A Figura 26 mostra esse número para cada viagem.

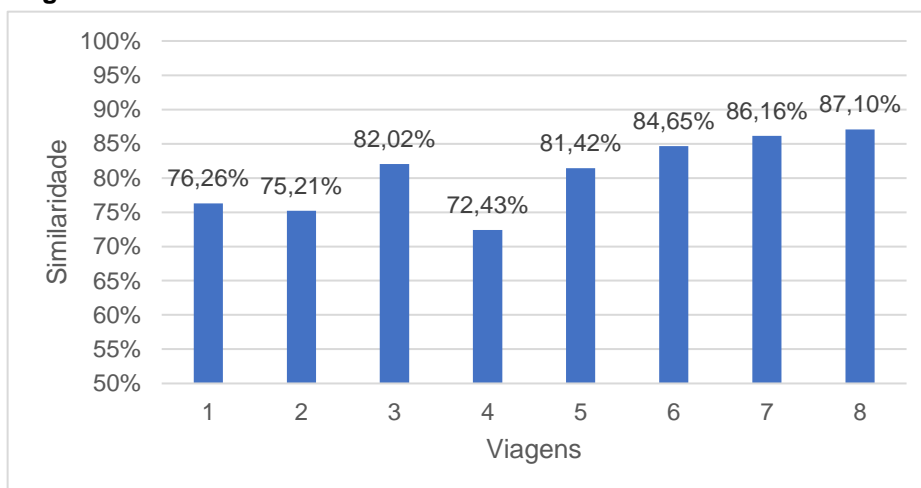
Figura 26 - Gráfico da troca de critérios de filtragem do Cenário B.

Fonte: Autoria própria.

É possível perceber que a proporção das quantidades de trocas que ocorrem nos critérios de filtragens utilizados, é inversamente proporcional a taxa de acertos da Figura 25. Um comportamento esperado, já que quanto melhor a recuperação menos modificações serão necessárias na Revisão, e a taxa de acerto dos pontos de aceleração serão maiores, com isso a frequência que os critérios são trocados é menor.

Assim como as outras análises, a média de casos similares recuperados para este cenário também apresentou um comportamento semelhante ao Cenário A, porém houve um aumento médio de 2,99% nas similaridades dos casos recuperados. A Figura 27 mostra a média para cada viagem.

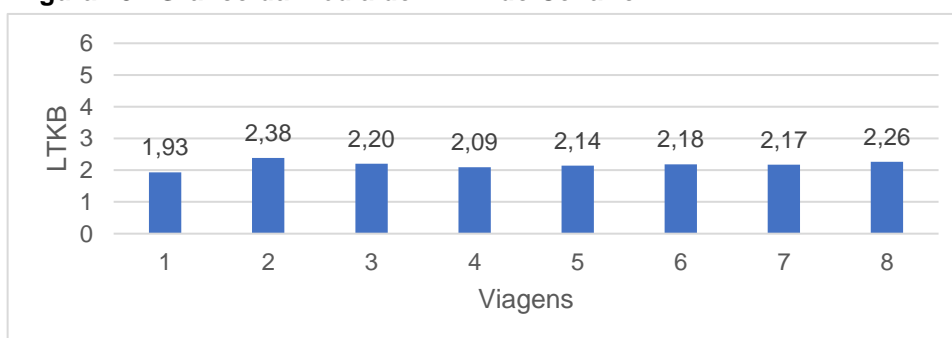
Figura 27 - Gráfico da média de similaridade do Cenário B.



Fonte: Autoria própria.

A análise dos consumos de combustível também apresentou valores finais semelhantes ao Cenário A. Adquiriu-se uma melhora de apenas 0,02 LTKB em relação ao cenário anterior, com uma média de 2,17 LTKB. O LTKB gasto em cada viagem é ilustrado na Figura 28.

Figura 28 - Gráfico da média de LTKB do Cenário B.



Fonte: Autoria própria.

Como a melhora no LTKB em relação ao cenário anterior foi baixa, a Tabela 4 que apresenta o consumo de combustível mostrou dados muito parecidos com a

Tabela 3, sendo ligeiramente melhores. A média de redução de consumo foi de 64,90%.

Tabela 4 - Redução de consumo obtido no Cenário B.

Configuração do Trem	LTKB		Redução (CRB - Real) %
	Real	CBR	
1	6,19	1,93	68,83%
2	5,68	2,38	58,05%
3	6,23	2,20	64,62%
4	6,49	2,09	67,79%
5	6,29	2,14	65,90%
6	6,17	2,18	64,62%
7	6,3	2,17	65,53%
8	6,26	2,26	63,87%

Fonte: Autoria própria.

A utilização do método RBC com a abordagem RDND (com parâmetros do Cenário B configurados) não apresentou resultados muito distintos da abordagem RDN. Houve uma pequena melhora, que em uma única viagem não traria grande economia de combustível, porém a longo prazo refletiria em uma economia considerável.

O próximo cenário também utilizará o método RDND, porém os valores de e e σ serão dobrados para analisar as diferenças que a configuração desses parâmetros causa na recuperação.

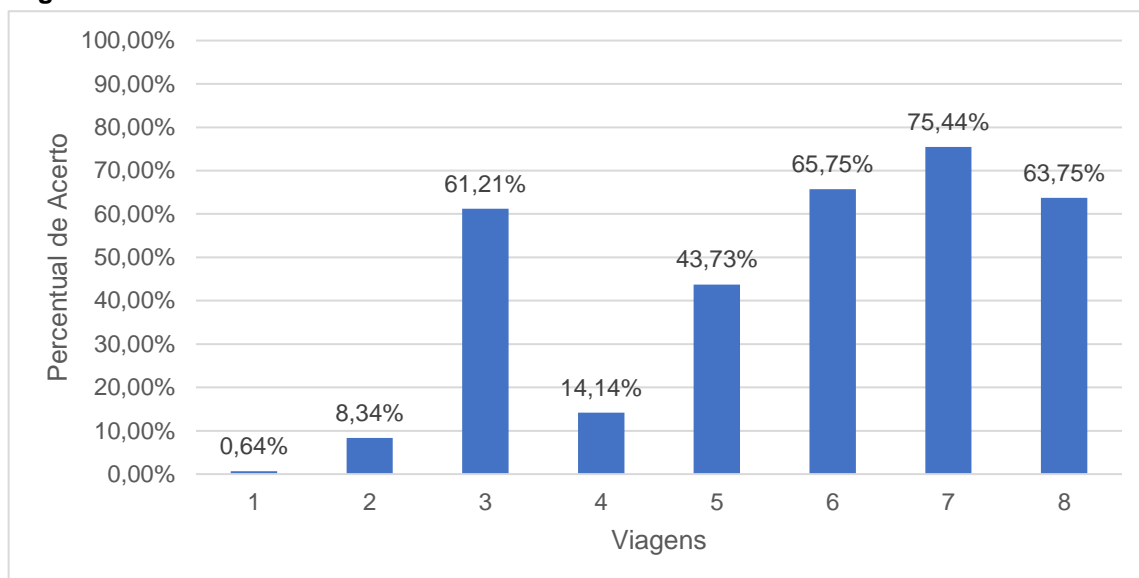
5.1.3 Cenário C

Neste cenário a análise da taxa de acertos manteve o comportamento dos Cenários A e B. A primeira viagem teve uma taxa quase nula que melhorou no decorrer das viagens seguintes, com uma queda na a viagem 4 (cf. Figura 29). Porém esse cenário obteve a maior média de taxa de acertos, com 41,63%.

A viagem 1 deste cenário foi a segunda melhor, ficando atrás do Cenário A. Com a utilização da abordagem RDND, pode-se perceber um comportamento em comum entre os cenários B e C (que utilizaram esta abordagem). Após a queda no percentual de acertos na viagem 4, ambos os cenários demonstraram uma dificuldade maior em reestabelecer uma taxa de acertos melhores, necessitando de uma viagem

a mais que a abordagem RDN para que o número de acertos passasse a ser maior que 50%.

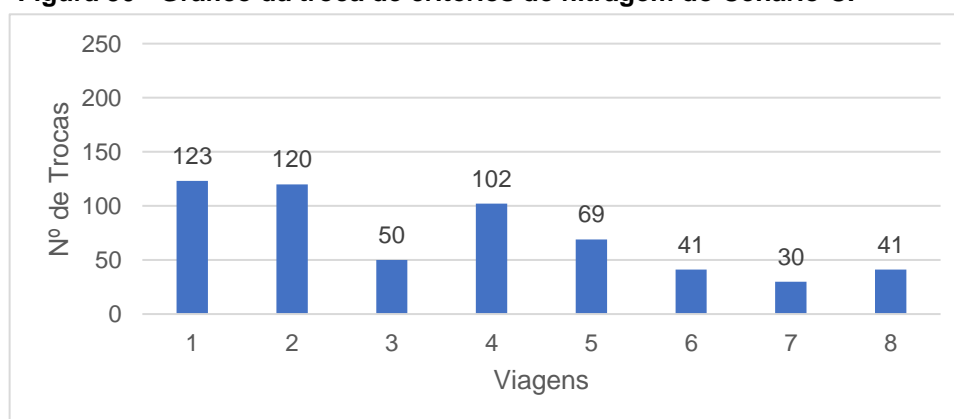
Figura 29 - Gráfico da taxa de acertos do Cenário C.



Fonte: Autoria própria.

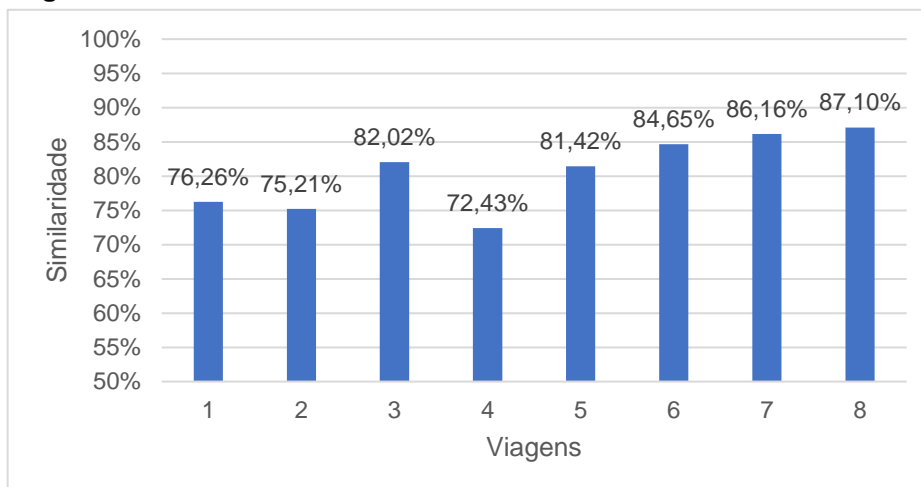
Assim como apontou a análise da taxa de acertos, o número de vezes que ocorreu trocas de critérios de filtragem, também apresentaram uma melhora (cf. Figura 30).

Figura 30 - Gráfico da troca de critérios de filtragem do Cenário C.



Fonte: Autoria própria.

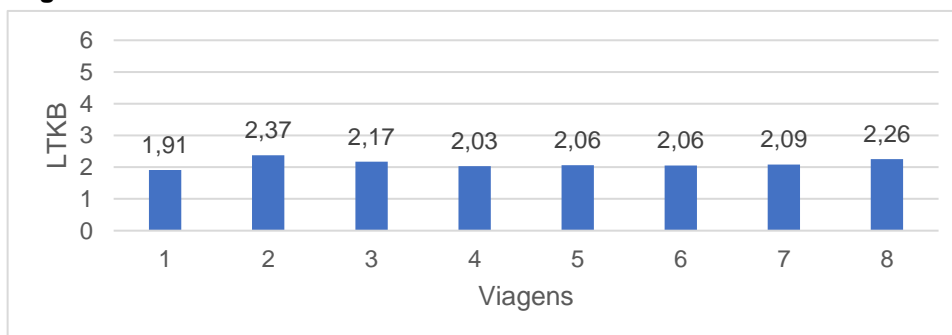
A média dos casos similares recuperados também obteve uma melhora, sendo a maior média entre os cenários testados, com 83,43% (cf. Figura 31).

Figura 31 - Gráfico da média de similaridade do Cenário C.

Fonte: Autoria própria.

As similaridades deste cenário tiveram uma melhora de 5,76% em relação ao Cenário A, e 2,77% em relação ao Cenário B. Mesmo os Cenários B e C utilizarem a mesma abordagem de recuperação (RDND), a diferença entre as configurações dos parâmetros mostraram que pode haver resultados distintos dentro da mesma abordagem.

A análise da média de consumo de combustível também obteve melhora, com uma média de 2,12 LTKB, a menor dentre os cenários (cf. Figura 32). A média de consumo foi 0,07 menor que o Cenário A e 0,05 menor que o cenário B.

Figura 32 - Gráfico da média de LTKB do Cenário C.

Fonte: Autoria própria.

Por mais que o LTKB tenha sido o menor dentro os três cenários, ainda assim a diferença não foi gritante. Graças a isto, a redução de consumo de combustível

apresentada na Tabela 5 não difere muito da Tabela 3 ou da Tabela 4. A média de redução foi de 65,75%.

Tabela 5 - Redução de consumo obtido no Cenário C.

Configuração do Trem	LTKB		Redução (CRB - Real) %
	Real	CBR	
1	6,19	1,91	69,12%
2	5,68	2,37	58,25%
3	6,23	2,17	65,14%
4	6,49	2,03	68,74%
5	6,29	2,06	67,28%
6	6,17	2,06	66,68%
7	6,3	2,09	66,86%
8	6,26	2,26	63,96%

Fonte: Autoria própria.

As próximas seções irão realizar um comparativo entre os cenários estudados neste trabalho, verificando as diferenças entre eles.

5.2 COMPARAÇÃO DE RESULTADOS

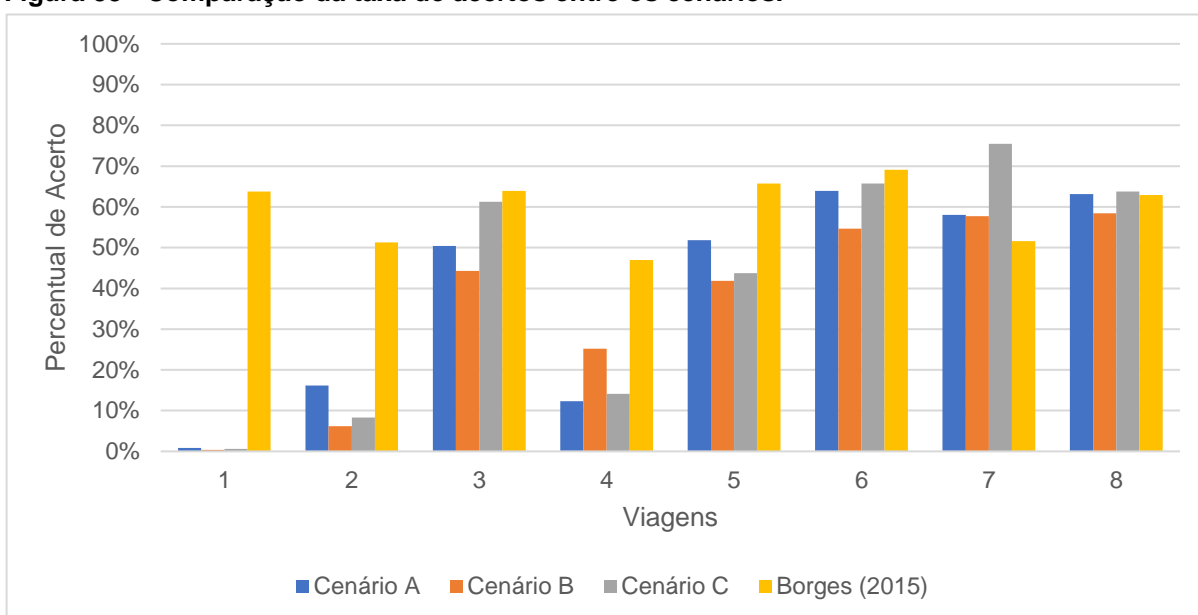
Nesta seção serão analisados os resultados dos três cenários testados anteriormente, comparando os resultados obtidos entre si e com os resultados de Borges (2015a). Entre os Cenários A, B e C serão comparadas a eficiente de acertos dos casos recuperados, a média de similaridade, o consumo e a redução de consumo de combustível. O número de trocas dos critérios de filtragem será comparado somente entre os cenários B e C, pois estes que utilizam o método RDND.

As comparações com Borges (2015a) não serão realizadas somente para o número de trocas dos critérios de filtragem, pois se trata de um aspecto particular deste trabalho. Os demais aspectos dissertados nos experimentos serão comparados. O principal objetivo da comparação de resultados com Borges (2015a), é analisar a diferença que a etapa de Adaptação faz no método RBC. Em sua pesquisa foi utilizado Algoritmo Genético para adaptação de casos recuperados, enquanto neste trabalho é utilizado Adaptação Nula.

5.2.1 Taxa de acertos

O gráfico da Figura 33 apresenta o comparativo da taxa de acerto entre os três cenários testados e Borges (2015a). É possível verificar que a primeira viagem teve uma taxa de acertos extremamente baixa para todos os cenários. Isso se deve à BC conter somente casos sementes e não estar populada com casos solucionados.

Figura 33 - Comparação da taxa de acertos entre os cenários.



Fonte: Autoria própria.

Porém Borges (2015a) obteve uma taxa de acertos muito alta desde a primeira viagem. Isso se deve graças a etapa de Adaptação, que ao modificar os casos, os deixa mais semelhantes ao problema enfrentado e conseqüentemente faz a taxa de acertos aumentar.

Outro aspecto interessante é na viagem 4, onde para todos os cenários ocorreu uma queda no percentual de acerto, com o Cenário A possuindo a maior queda. Em Borges (2015a), a Adaptação fez com que a queda na taxa de acertos fosse mais branda, cerca de 17% em comparação a 35% dos cenários com Adaptação Nula. Como a composição do trem utilizado na viagem 4 diferia bastante das demais, foi preciso que o método gravasse novos casos solucionados para esse tipo de composição, afetando a precisão da recuperação de todos os cenários.

O mesmo efeito ocorre com a composição da viagem 2, porém como o método ainda estava “aprendendo” e não apresentou uma queda na taxa de acertos. Porém é importante analisar que a composição do trem da viagem 6 é muito semelhante à viagem 4, porém como o método já havia “aprendido”, resultou em um percentual de acerto muito mais alta na viagem 6 para todos os cenários.

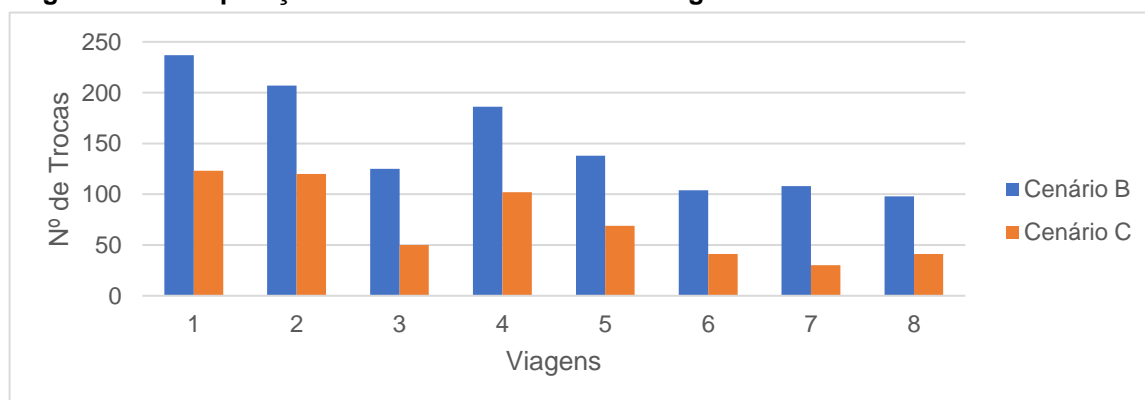
Desta forma, pode-se concluir toda vez que o método realiza uma viagem com uma composição “inédita”, as recuperações de casos realizadas não serão boas e necessitarão de alterações no conjunto solução S . Porém as próximas viagens que tiverem composições semelhantes a viagens já realizadas, irão realizar recuperações muito mais precisas e úteis, necessitando de menos modificações na solução do caso.

As viagens 7 e 8 do Cenário C apresentaram os melhores resultados, com a viagem 7 alcançando a maior porcentagem de acertos de todas as viagens e de todos os cenários, com 75,44%. O cenário C também apresentou a melhor média entre as oito viagens com 41,63%, enquanto o Cenário B apresentou a pior média com 36,08%.

5.2.2 Troca do conjunto de critérios de filtragem

Para esta análise somente os cenários B e C serão considerados, pois essa característica é exclusiva do método RDND. A Figura 34 mostra o número de trocas realizadas durante as viagens.

Figura 34 - Comparação da troca de critérios de filtragem entre os cenários.



Fonte: Autoria própria.

Devido ao Cenário B ter a configuração $e = 33$ e o Cenário C $e = 66$, isso reflete na quantidade de trocas realizadas. Como e é mais baixo no Cenário B, ele é

atingido mais facilmente e então os critérios são trocados com maior frequência. Como no Cenário C o *e* é o dobro que o Cenário B, as trocas realizadas caem proporcionalmente, ou seja, a taxa de trocas dos critérios cai pela metade em todas as viagens.

Ao comparar as médias de trocas de critérios entre os cenários, temos o Cenário B com média igual a 150, enquanto o Cenário C tem uma média de 72. Para ser possível uma comparação mais direta, é preciso considerar que o Cenário C tem *e* como sendo o dobro do Cenário B. Sendo assim, ao dividirmos a média do Cenário B pela metade, obtemos um resultado de 75, algo mais próximo do Cenário C. Ainda assim, ao comparar os resultados de ambos os cenários, o Cenário C apresentou um número de trocas nos critérios de filtragem menor que o Cenário B.

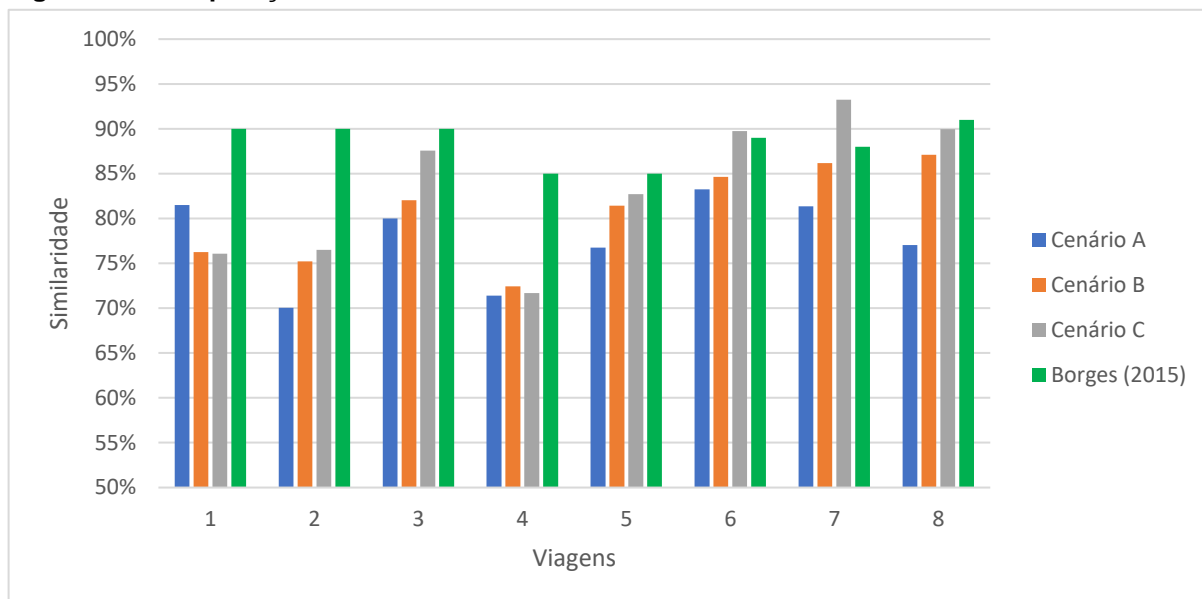
5.2.3 Similaridade

Nesta análise é possível verificar que com exceção da viagem 1, onde a melhor média foi do Cenário A, e a viagem 4 onde a melhor média foi do Cenário B, em todas as demais viagens o Cenário C teve as melhores médias de similaridade dos testes realizadas, atingindo o pico de 93,23% na viagem 7, sendo maior até que o resultado de Borges (2015a). O Cenário A só obteve bons resultados na viagem 1, depois apresentou a pior média de similaridade para o resto das viagens efetuadas, atingindo o vale de 70,04% na viagem 2 (cf. Figura 35).

Porém ao analisarmos os resultados de Borges (2015a) em comparação com os cenários testados, é possível ver que para todas as viagens, a similaridade alcançada é alta, com a viagem 1 sendo de 90% enquanto a maior similaridade obtida nos testes é do Cenário A com 81,51%. Este comportamento se mantém para quase todas as viagens, sendo menor apenas que as viagens 6 e 7 do Cenário C.

Isso mostra o impacto que o processo de adaptação de casos tem no método RBC. Na viagem 1 todos os cenários apresentaram similaridades mais baixas que a média, porém o resultado de Borges (2015a) apresentou similaridade altas desde a primeira viagem, mesmo com a BC contendo somente casos sementes. O mesmo ocorre na viagem 4, onde todos os cenários tiveram uma queda brusca na similaridade. Para os cenários testados, a queda da viagem 3 para a viagem 4 foi de aproximadamente 11%, enquanto para Borges (2015a) foi de 5%.

Figura 35 - Comparação da média de similaridade entre os cenários.



Fonte: Autoria própria.

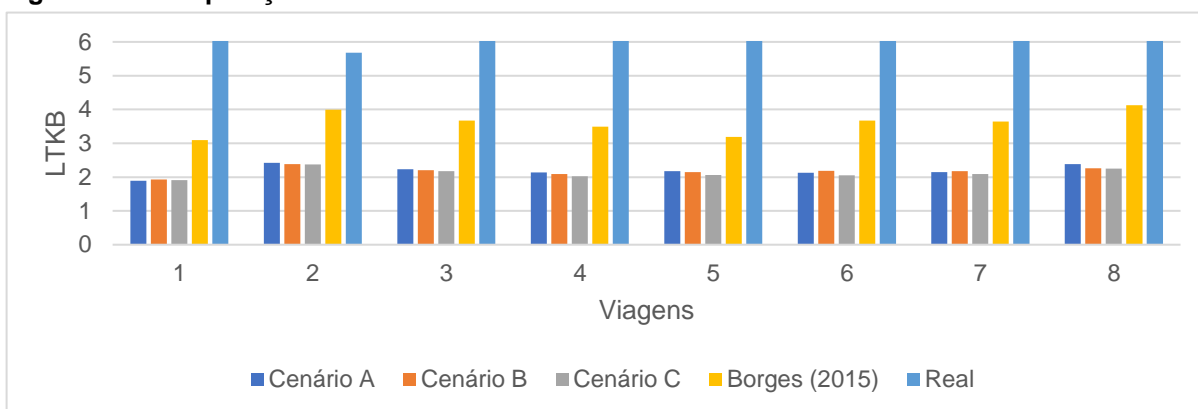
O Cenário C que atingiu os resultados mais próximos a Borges (2015a) na maior parte das viagens, e até mesmo apresentando resultados melhores para algumas viagens, mesmo sem utilizar processo de adaptação alguma.

5.2.4 Consumo de Combustível

O consumo de combustível entre os cenários também demonstrou ser muito semelhante. Na viagem 1 o LTKB foi quase o mesmo para os três cenários. A partir da viagem 2 até a 8, o Cenário C obteve os menores LTKB. Em contrapartida o Cenário A obteve a maior LTKB para a maior parte das viagens, sendo menor que o Cenário B somente nas viagens 6 e 7 (cf. Figura 36).

Ainda assim é notável que todos os cenários obtiveram LTKB menor que Borges (2015a). Tal informação não indica que os resultados adquiridos neste trabalho sejam melhores. Essa diferença ocorreu devido a uma condição na viagem, pois nos testes realizados não consideraram o tempo de viagem do trem. Sendo assim, é possível que Borges (2015a) esteja realizando uma viagem mais rápida, e consequentemente esteja consumindo mais combustível.

Figura 36 - Comparação do LTKB entre os cenários.



Fonte: Autoria própria.

Como pode haver uma variação no tempo da viagem, decorrida da velocidade que o trem percorreu a via férrea, não é possível realizar uma comparação direta entre os dados obtidos neste trabalho e Borges (2015a).

5.2.5 Redução do Consumo

Devido a possível diferença no tempo da execução da viagem, citada anteriormente, a redução no consumo de combustível também resultou em uma maior economia que Borges (2015a). As causas são as mesmas citadas na seção anterior, um maior gasto de combustível com o intuito de realizar uma viagem mais rápida. A Tabela 6 compara o LTKB dos cenários testados e de Borges (2015a).

Tabela 6 - Comparação do consumo obtido.

Configuração do Trem	LTKB				
	Real	Cenário A	Cenário B	Cenário C	Borges (2015a)
1	6,19	1,90	1,93	1,91	3,09
2	5,68	2,42	2,38	2,37	3,99
3	6,23	2,23	2,20	2,17	3,67
4	6,49	2,14	2,09	2,03	3,49
5	6,29	2,17	2,14	2,06	3,19
6	6,17	2,13	2,18	2,06	3,67
7	6,3	2,15	2,17	2,09	3,64
8	6,26	2,38	2,26	2,26	4,13

Fonte: Autoria própria.

E a Tabela 7 compara a porcentagem de redução do consumo de combustível.

Tabela 7 - Comparação da porcentagem de redução de consumo obtida.

Configuração do Trem	Redução			
	(Cenário A - Real) %	(Cenário B - Real) %	(Cenário C - Real) %	(Borges (2015a) - Real) %
1	69,38%	68,83%	69,12%	50,00%
2	57,33%	58,05%	58,25%	30,00%
3	64,21%	64,62%	65,14%	41,00%
4	67,03%	67,79%	68,74%	46,00%
5	65,48%	65,90%	67,28%	49,00%
6	65,50%	64,62%	66,68%	40,00%
7	65,85%	65,53%	66,86%	42,00%
8	61,97%	63,87%	63,96%	34,00%

Fonte: Autoria própria.

Devido aos testes não terem levado em consideração o tempo de viagem, houve uma discrepância de aproximadamente 23,58% entre os resultados dos cenários e de Borges (2015a). Com isto, a comparação da redução do consumo de combustível se torna insustentável, pois a contagem de tempo na execução da viagem, gerou um novo aspecto a ser analisado e testado neste trabalho.

Entre os próprios cenários, o Cenário C foi o que apresentou a maior redução de consumo com média de 65,75%, seguido do Cenário B com média de 64,90% e por fim o Cenário A com média de 64,59%. Com esses resultados, é possível afirmar que o método RDND teve a uma maior redução de consumo de combustível, pois tanto os Cenários B e C que utilizavam essa abordagem, foram melhores que o Cenário A que utilizava a abordagem RDN.

5.3 CONSIDERAÇÕES FINAIS

Para a execução dos experimentos, foram selecionadas oito configurações diferentes de trens e uma única via férrea. Configurando os métodos RDN e RDND, e construindo três cenários distintos, foi analisado quais destas três configurações de recuperação obtinham os melhores resultados.

Pode-se observar o aprendizado do método, obtendo resultados não satisfatórios nas primeiras viagens e melhorando consideravelmente nas viagens posteriores. Também foi possível verificar como a etapa de adaptação fez diferença

na aplicação do RBC, pois com a utilização da Adaptação Nula foram necessárias 6 viagens para que resultados fossem semelhantes à um RBC com adaptação.

Ainda assim, mesmo com Adaptação Nula o método apresentou grande redução no consumo de combustível. Porém ao comparar o consumo de combustível entre os métodos implementados, o gasto real e Borges (2015a), faltou analisar o tempo de execução da viagem, no intuito de saber qual viagem estava indo mais rápido e consumindo menos combustível, pois quanto mais rápido o trem for, mais combustível será gasto.

6 CONCLUSÃO

Este trabalho apresentou a implementação de um novo algoritmo de recuperação de casos para o método RBC, baseado em um algoritmo existente na literatura. O ambiente de aplicação foi o modal férreo, onde a tomada de decisão de é uma tarefa complexa e exige experiência dos maquinistas para que haja uma viagem sem riscos ou danos para o trem ou para via. O foco de desenvolvimento foi exclusivamente na etapa de Recuperação do método RBC, ignorando as nuances das demais etapas, pois cada etapa do método possui suas complexidades que teriam o potencial de um estudo específico.

O objetivo principal da pesquisa foi utilizar o método de Recuperação em Dois Níveis (RDN), e criar um novo método, a Recuperação em Dois Níveis Dinâmica (RDND), que possui um comportamento levemente diferente. Ambos os métodos executam filtragem para reduzir o número de casos em uma BC, utilizando atributos do Caso para efetuar essa tarefa. A grande diferença entre os métodos, é que a RDN possui um filtro fixo, e RDND possui um filtro variante que altera ao decorrer da viagem.

Para a execução do RDND, a maior dificuldade encontrada foi em determinar os valores de configuração ideal para os parâmetros do método. O *limiar* σ define o quão bem seria a filtragem realizada, porém foi necessário determinar valores que reduzissem a maior quantidade de casos e ao mesmo tempo fosse possível selecionar somente os casos úteis. Algo difícil de equilibrar, pois existe a possibilidade de casos com o valor do atributo (que esteja sendo utilizado para filtragem do método) terem valores “fora da faixa” configurada, e ainda sim serem úteis.

Outro parâmetro difícil de equilibrar foi o *parâmetro de erro* e , este que ditava a troca dos critérios de filtragem. Este parâmetro também foi necessário determinar valores equilibrados, pois se um atributo utilizado para filtragem estivesse trazendo muitos casos não úteis, era necessário que houvesse a troca o quanto antes. Porém ainda sim, essa troca não deveria ser algo frequente, pois por mais que um atributo esteja realizando boas recuperações, é comum que ocorram erros, entretanto em número reduzido. Então se uma houvesse uma troca muito rápida, poderia ocorrer de trocar atributos que estavam gerando boas filtrações, por atributos que gerariam filtrações ruins.

Foram utilizados três cenários diferentes para executar os testes da recuperação de casos no ambiente simulado criado por Borges (2015a). Através dos resultados pode-se concluir que o RDND obteve melhores resultados que o RDN. Porém também foi possível determinar que o RDND apresentou uma taxa de “aprendizagem” menor que RDN, sendo necessárias à execução de mais viagens para que o método começasse a realizar boas recuperações.

Ao comparar os resultados obtidos com o de Borges (2015a), pode-se notar a diferença que a etapa de Adaptação fez ao método RBC. Em Borges (2015a), desde as primeiras viagens, foram adquiridas altas taxas de acertos na Recuperação, e casos com ótimas similaridades, sempre mantendo uma média alta. Diferente de RDN e RDND, onde era preciso ao menos 4 viagens para que o método começasse a recuperar casos com boas similaridades.

Um aspecto que impossível realizar comparação com cenário real, foi a redução no consumo de combustível. No cenário real, os trens realizavam paradas nas viagens, atividade esta que aumentava consideravelmente o consumo médio do trem, pois era necessário realizar uma nova “arrancada” depois. Este aspecto não foi considerado neste trabalho, e devido a isto os resultados apresentados aqui foram consideravelmente menores.

Outro aspecto que tornou a comparação insustentável foi de também a redução do consumo de combustível, porém com Borges (2015a). Como não houve uma análise no tempo de execução da viagem do trem, não se pode concluir se as viagens estavam sendo realizadas em uma mesma velocidade, ou se Borges (2015a) estava executando viagens mais rápidas (e gastando mais combustível para isto) que nos cenários testados.

Ao final da análise dos testes, pode-se concluir que dentre os cenários estudados, o Cenário C é o que possui a melhor configuração para realização da recuperação de casos. Este cenário apresentou as melhores médias na taxa de acertos (casos que não precisaram ser modificados na etapa Revisão), a menor quantidade de trocas no conjunto de critérios, a maior porcentagem de casos similares e a maior redução no consumo de combustível. Ou seja, para todos os aspectos analisados, o Cenário C se sobressaiu.

6.1 TRABALHOS FUTUROS

Para a trabalhos futuros, sugere-se a adição da atividade de parada do trem durante as viagens, para que seja possível realizar uma comparação justa no gasto de combustível que os métodos de recuperação abordados neste trabalho obtiveram em relação ao cenário real. Outra adição seria a análise do tempo de viagem de um trem, para realizar a comparação adequada entre os cenários e Borges (2015a).

Outra atividade sugerida, é a execução do método RDND com outros valores de configuração para o método, para analisar e comparar os resultados com os já obtidos neste trabalho. Combinações como $e = 33$ e $\sigma = 0,66$ ou $e = 66$ e $\sigma = ,0,33$, não foram testadas nesta pesquisa e poderiam demonstrar comportamentos diferentes dos obtidos com as configurações definidas nos cenários de teste. Também pode-se analisar a porcentagem de similaridade que cada par de atributos do conjunto de critérios obteve, e compara-los entre si para determinar o melhor.

Além disso, pode ser feita a adição de um método de recuperação, tal como a *Árvore K-D* que também foi citada nesta pesquisa ou o método *KNN*, para ver a diferença na qualidade das recuperações em relação ao método criado.

7 REFERÊNCIAS

AAMODT, A.; PLAZA, E. **Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches**. In: AICOM – Artificial Intelligence Communications. Vol. 7. Ed. IOS Press, EUA, 1994.

ABEL, M. **Um estudo sobre Raciocínio Baseado em Casos**. 1996. 41 f. Dissertação (Pós-Graduação em Ciência da Computação) – Universidade Federal do Rio Grande do Sul. Porto Alegre, 1996.

BORGES, ANDRÉ. P. **Uma contribuição para geração de políticas de ações para condução de trens de carga usando Raciocínio Baseado em Casos**. 2015. 220 f. Dissertação (Doutorado) – Programa de Pós-Graduação em Informática da Pontifícia Universidade Católica do Paraná. Curitiba, 2015.

BORGES, ANDRE P.; DORDAL, OSMAR B.; RIBEIRO, RICHARDSON; ÁVILA, BRÁULIO C.; SCALABRIN, EDSON E. **Generation of Economical Driving Plans Using Continuous Case-Based Planning**. Lecture Notes in Business Information Processing. 1ed.: Springer International Publishing, 2015, v., p. 192-213.

BORGES, ANDRE P.; SILVA, V. B.; DORDAL, OSMAR B.; AVILA, BRAULIO C.; SCALABRIN, EDSON E. **Safety in Multi-Agent Systems: Reputation based on Dossier**. In: 28th International Florida Artificial Intelligence Research Society Conference, 2015, Hollywood. Proceedings of 28th International Florida Artificial Intelligence Research Society Conference, 2015. v. 1. p. 348-353.

BORGES, ANDRE P.; DORDAL, OSMAR B.; DOS SANTOS, EMERSON L.; RIBEIRO, RICHARDSON; AVILA, BRAULIO C.; SCALABRIN, EDSON E. **An approach for train driving using case-based reasoning**. In: 2015 European Control Conference (ECC), 2015, Linz. 2015 European Control Conference (ECC). v. 1. p. 1037-7.

BORGES, ANDRE P.; DORDAL, OSMAR B.; SATO, DENISE M. V.; AVILA, BRAULIO C.; ENEMBRECK, FABRICIO; SCALABRIN, EDSON E.; RIBEIRO, RICHARDSON. **A multi-layer architecture proposal for conducting trains employing CBR**. In: 2014 IEEE 18th International Conference on Computer Supported Cooperative Work in Design (CSCWD), 2014, Hsinchu. Proceedings of the 2014 IEEE 18th International Conference on Computer Supported Cooperative Work in Design (CSCWD). p. 23-8.

BORGES, ANDRE P.; DORDAL, OSMAR B.; SATO, DENISE M. V.; AVILA, BRAULIO C.; ENEMBRECK, FABRICIO; SCALABRIN, EDSON EMILIO. **An intelligent system for driving trains using Case-Based Reasoning**. In: 2012 IEEE International Conference on Systems, Man and Cybernetics SMC, 2012, Seoul. 2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC). p. 1694.

BORGES, ANDRE P.; GRANATYR, JONES; DORDAL, OSMAR B.; RIBEIRO, RICHARDSON; AVILA, BRAULIO C.; ENEMBRECK, FABRICIO; SCALABRIN, EDSON E. **Knowledge discovery applied in modal rail**. In: 2011 15th International

Conference on Computer Supported Cooperative Work in Design (CSCWD), 2011, Laussane. Proceedings of the 2011 15th International Conference on Computer Supported Cooperative Work in Design (CSCWD). v. 1. p. 253.

BORGES, A. P.; DORDAL, O.; ENEMBRECK, F.; FAVARIM, F.; BARBOSA, M. A. C.; ROCHA, F. R. L.; LINARES, K. S. C.; RIBEIRO, R. **Um Modelo de Aprendizagem Baseado em Agentes Inteligentes: Um estudo de caso em processos avícolas.** In: XXXVII Conferencia Latinoamericana de Informática, 2011, Quito. Proceedings of XXXVII Conferencia Latinoamericana de Informática, 2011.

BORGES, ANDRE P.; RIBEIRO, RICHARDSON; AVILA, BRAULIO C.; ENEMBRECK, FABRICIO; SCALABRIN, EDSON E. **A learning agent to help drive vehicles.** In: , 2009, Santiago. . p. 282-287.

BORGES, A. P.; RIBEIRO, R.; LEITE, A. R.; DORDAL, O.; GIACOMET, B.; ÁVILA, Bráulio Coelho; ENEMBRECK, F.; SCALABRIN, Edson Emílio. **Uma Arquitetura de Aprendizagem para a Condução Automática de Veículos.** In: XXXV Latin American Informatics Conference, 2009, Pelotas. XXXV Latin American Informatics Conference, 2009.

CHANDRA, S.; AGARWAL, M. M. **Railway Engineering.** Oxford University Press, 2007.

DORDAL, OSMAR B.; BORGES, ANDRE P.; SATO, DENISE M. V.; ENEMBRECK, FABRICIO; SCALABRIN, EDSON E.; AVILA, BRAULIO C. **An Intelligent System for train overtaking using distributed coordination.** In: IECON 2013 39th Annual Conference of the IEEE Industrial Electronics Society, 2013, Vienna. IECON 2013 - 39th Annual Conference of the IEEE Industrial Electronics Society. v. 0. p. 3329-3334.

DORDAL, OSMAR B.; BORGES, ANDRE P.; RIBEIRO, RICHARDSON; ENEMBRECK, FABRICIO; SCALABRIN, EDSON E.; AVILA, BRAULIO C. **Towards an optimal driving trains in single line using crossing loops.** In: 2011 15th International Conference on Computer Supported Cooperative Work in Design (CSCWD), 2011, Laussane. Proceedings of the 2011 15th International Conference on Computer Supported Cooperative Work in Design (CSCWD). p. 772.

DORDAL, OSMAR B.; BORGES, ANDRE P.; RIBEIRO, RICHARDSON; ENEMBRECK, FABRICIO; SCALABRIN, EDSON E.; AVILA, BRAULIO C. **Strong reduction in fuel consumption driving trains in bi-directional single line using crossing loops.** In: 2011 IEEE International Conference on Systems, Man and Cybernetics SMC, 2011, Anchorage. 2011 IEEE International Conference on Systems, Man, and Cybernetics. p. 1597.

GICK, M. L.; HOLYOAK, K. J. **Analogical Problem Solving.** Cognitive Psychology, 12, 1980.

HINRICHS, T. R. **Problem Solving in Open Words: A Case Study in Design.** Nothvale, NJ: Erlbaum, 1992.

KOLODNER, J. **Reconstructive memory: a computer model**. Cognitive Science 7(4), 1983 Apud KOLODNER 1993.

KUTZ, M. **Handbook of Transportation Engineering**. McGraw Hill Professional, 2011.

LEAKE, D. B. **CBR in Context: The Present and Future**. In: D. Leake (ed.), Case-Based Reasoning Experiences: Lessons Learned & Future Directions, 1996.

LOUMIET, J.; JUNGBAUER, W. & ABRAMS, B. **Train Accident Reconstruction and FELA and Railroad Litigation**. Lawyers & Judges Publishing Company. 2005.

NETO, C. B. **Manual Didático de Ferrovias**. Universidade Federal do Paraná, 2011.

PLOTEGHER, S. L.; FERNANDES, M. M. **Raciocínio Baseado em Casos Aplicado a um Sistema de Diagnóstico Remoto**. 2005. In: XXV Congresso da Sociedade Brasileira de Computação. 2005. São Leopoldo, RS. 2005.

PORTER, B.; BAREISS, R. **PROTOS: An Experiment in Knowledge Acquisition for Heuristic Classification Tasks**. Proc. Of the 1st Int. Meeting on Advances in Learning, France 1986.

PORTO, T. G. **Ferrovias**. Escola Politécnica da Universidade de São Paulo, 2004.

REBELO, A. F.; FONSECA, A. C. V.; FERREIRA, C. D. S.; BORGES, J. P. S.; ARAÚJO, J. P.; TEIXEIRA, S. P. **Os Comboios em Portugal**. 2010. Dissertação (Mestrado) – Universidade do Porto. Porto, 2010.

RICHTER, M. M. **On the Notion of Similarity in Case-Based Reasoning**. G. della Riccia et.al (eds.), Mathematical and Statistical Methods in Artificial Intelligence, Springer Verlag, 1995.

RIESBECK, R. S. **Inside Case-Based Reasoning**. Lawrence Erlbaum, 1989.

SHANK, R.; ABELSON, R. **Scripts, Plans, Goals and Understanding**. Hillsdale NJ, Erlbaum, 1977.

SILVA, J. C. L. **Breve História das Ferrovias**. Brasil Escola. Disponível em <<http://brasilecola.uol.com.br/geografia/ferrovias.htm>>. Acesso em 19 de setembro de 2017.

SILVA, R. P. **Modelo de apoio ao diagnóstico no domínio médico: aplicando raciocínio baseado em casos**. 2005. Dissertação (Mestrado) – Universidade Católica de Brasília. Brasília, 2005.

SMITH, E. E.; ADAMS, N. **Fact Retrieval and the Paradox of Interference**. Cognitive Psychology, 1978.

SMITH, E. E.; MEDIN, D. L. **Categories and Concepts**. Cambridge, MA, Harvard University Press, 1981.

TULVING, E. **Episodic and Semantic Memory**. In E. Tulving & W. Donaldson (eds), *Organisation of Memory*, Academic Press, 1977.

VON WANGENHEIM, Aldo.; GRESSE VON WANGENHEIM, Christiane. **Raciocínio Baseado em Casos - 1ª ed.** Editora Manole Ltda. 2003.