



REVISTA MILITAR de CIÊNCIA e TECNOLOGIA



Versão online: ISSN 2316-4522
Volume XXXVIII - 4º Trimestre de 2021

ENGENHARIA MILITAR

**Mobilidade,
Contramobilidade e
Proteção**

Monitoramento da influência da velocidade de circulação dos trens na força de impacto por meio de sensores de fibra óptica – Pág.03

A viabilidade da implantação de medidas de redução de consumo de água em prédios de Brasília/DF – Pág.16

Determinação do tirante mínimo da hidrovia do Tocantins e Araguaia – Pág.26

Uma arquitetura baseada em micro serviços para apoio a transparência em processos de compras E-Gov – Pág.37

Motivação para a cartografia colaborativa: um experimento no Exército Brasileiro – Pág.49

Um algoritmo de aprendizado iterativo baseado no ciclo de marcha humana para a locomoção de um robô humanoide – Pág.60

Data Modeling for Cassandra – Pág.70

Uma proposta de ferramenta para a experimentação do aprendizado profundo no reconhecimento da fala para o idioma Português – Pág.83





REVISTA MILITAR DE CIÊNCIA E TECNOLOGIA

COMANDANTE DO EXÉRCITO:

Gen Ex Paulo Sérgio Nogueira de Oliveira

CHEFE DO DEPARTAMENTO DE CIÊNCIA E TECNOLOGIA:

Gen Ex Guido Amin Naves

CHEFE DO DEPARTAMENTO DE EDUCAÇÃO E CULTURA DO EXÉRCITO:

Gen Ex André Luis Novaes Miranda

DIRETOR DA BIBLIOTECA DO EXÉRCITO:

Cel Art Eduardo Biserra Rocha

COMANDANTE DO INSTITUTO MILITAR DE ENGENHARIA:

Gen Bda Juraci Ferreira Galdino

DIREÇÃO, IMPRESSÃO E DISTRIBUIÇÃO:

• Diretor: Cel Art Eduardo Biserra Rocha – BIBLIEx • Editor executivo: Cel R1 Leocir Dal Pai – BIBLIEx

CORPO REDATORIAL:

• Editor geral: Marcelo de Miranda Reis – IME (SE/2) • Editor adjunto: Paulo Henrique Coelho Maranhão – IME (SE/4) • Engenharia Civil, Transportes e Ciências Ambientais: Filipe Almeida Corrêa do Nascimento – IME (SE/2) • Engenharia Nuclear: Sérgio de Oliveira Vellozo – IME (SE/7) • Engenharia Elétrica: Antônio Eduardo Carrilho da Cunha – IME (SE/3) • Engenharia Mecânica: André Luiz Tenório Rezende – IME (SE/4) • Ciência dos Materiais: André Ben-Hur da Silva Figueiredo – IME (SE/8) • Ciência da Computação: Paulo Cesar Salgado Vidal – IME (SE/9) • Engenharia de Defesa: Giuseppe Miceli Júnior – IME (SE/10) • Engenharia Química e Química: Kátia Regina de Souza – IME (SE/5) • Engenharia Cartográfica: Matheus Pinheiro Ferreira – IME (SE/6) • Outras áreas: Aderson Campos Passos – IME (SE/9).

EDITORES ASSOCIADOS EXTERNOS:

• Dr. André Fenili – Universidade Federal do ABC, Santo André, SP • Dr. Artur Ziviani – Lab. Nacional de Computação Científica (LNCC), Petrópolis, RJ • Dr. Fernando Fachini Filho – Instituto Nacional de Pesquisas Espaciais, SP • Dr. José Carlos Costa da Silva Pinto – Universidade Federal do Rio de Janeiro, RJ • Dr. José Carlos Maldonado – Universidade de São Paulo, São Carlos, SP • Dr.ª Júlia Célia Mercedes Strauch – Escola Nacional de Ciências Estatísticas, RJ • Dr. Luiz Pereira Calóba – Universidade Federal do Rio de Janeiro, RJ • Dr. Otto Corrêa Rotunno Filho – COPPE/Universidade Federal do Rio de Janeiro, RJ • Dr. Richard Magdalena Stephan – COPPE/Universidade Federal do Rio de Janeiro, RJ • Dr. Webe João Mansur – COPPE/Universidade Federal do Rio de Janeiro, RJ • Dr. Carlos Alberto Nunes Consenza – COPPE/Universidade Federal do Rio de Janeiro, RJ

ADMINISTRAÇÃO, REVISÃO, PROGRAMAÇÃO E DIAGRAMAÇÃO:

• Coordenador: Rubenildo Python de Barros • Administração e OJS: Vanessa Ferreira Belchior, Lucas Sebastião de Paula, José Welton Moreira • Revisores: Giana Elvira de Oliveira Marquês; Juniellen de Oliveira Scipel; Diego Ribeiro Ferreira Serpa • Diagramador: Luiz Tadeu Carqueija Mota

PROJETO GRÁFICO, IMPRESSÃO E DISTRIBUIÇÃO:

BIBLIEx



Desde 1949

"A Gráfica do Exército" - Compromisso com a Qualidade

Impresso na Gráfica do Exército

Al. Mal. Rondon - Setor de Garagens - QGEx - SMU - CEP: 70630-901 - Brasília - DF

Tel: 3415-4367 - Site: <http://www.graficadoexercito.eb.mil.br>

E-mail: divcmcl@graficadoexercito.eb.mil.br



REVISTA MILITAR de CIÊNCIA e TECNOLOGIA

REVISTA MILITAR DE CIÊNCIA E
TECNOLOGIA - Volume XXXVII
4º Trimestre de 2021

BIBLIOTECA DO EXÉRCITO

EDITORA (BIBLIEx)

Palácio Duque de Caxias - Praça D. de Caxias, 25

3º andar - Ala Marcílio Dias

Centro - Rio de Janeiro-RJ

CEP 20.221-260

Tel.: (21) 2519-5707

INSTITUTO MILITAR DE ENGENHARIA - IME

Praça General Tibúrcio, 80 - Praia Vermelha -

Rio de Janeiro-RJ - CEP 22.290-270 -

Tel.: (21) 2546-7115

Website: <http://rmct.ime.eb.br> - Email: rmct@ime.eb.br

ACESSE NOSSAS REVISTAS DIGITAIS



Nossa capa:

Homenagem ao Corpo de Engenheiros





Caros leitores interessados em assuntos relacionados à Ciência e Tecnologia, é com grande satisfação que damos publicidade à quarta edição de 2021. Nesta edição, trazemos aos nossos leitores oito artigos com temas em três áreas da engenharia.

O artigo da Engenharia Cartográfica aborda o emprego do mapeamento colaborativo aplicado ao Exército Brasileiro aproveitando-se da técnica da gamificação aumentando o engajamento dos militares em um sistema de mapeamento colaborativo.

Três trabalhos na área de Engenharia de Fortificação e Construção abordam uma avaliação das condições de navegabilidade da hidrovia do Tocantins e Araguaia tanto em trechos em que a navegação já ocorre livremente durante a maior parte do ano, como em locais onde necessitam de obra de dragagens; uma análise de viabilidade de métodos de redução de consumo de água em edificações residenciais no Plano Piloto, em Brasília e uma demonstração do funcionamento de um novo equipamento no Brasil, que faz uso de sensores de fibra ótica para a medição de força de impacto que as rodas de ferroviários causam nos trilhos da via permanente.

A Engenharia de Computação nos apresenta com quatro artigos. O primeiro descreve uma arquitetura baseada em micro serviços que contribui para a integração das informações que hoje se encontram espalhadas nas diversas funcionalidades do sistema “Compras Governamentais”. O segundo apresenta um algoritmo de aprendizado iterativo que prioriza a estabilidade estática em vez da estabilidade dinâmica, com a finalidade de diminuir a complexidade da marcha de um robô humanoide. O terceiro trabalho mostra um ambiente que possibilite a experimentação de modelos na tarefa de reconhecimento de fala, bem como implementação de modelos de classificação utilizando redes neurais de aprendizado profundo. O último artigo apresenta diretrizes para ajudar o usuário com a modelagem de dados no sistema de gerenciamento de banco de dados Cassandra NOSQL visando a redução do número de visões materializadas a serem criadas.

Esperamos que a leitura seja proveitosa, servindo de motivação para novas pesquisas. Fomentar a ciência e a tecnologia é a pedra fundamental do desenvolvimento tecnológico de uma nação forte. Aproveitem esta edição!

Paulo Cesar Salgado Vidal

SUMÁRIO

- 03** • Monitoramento da influência da velocidade de circulação dos trens na força de impacto por meio de sensores de fibra óptica
Danilo Martelli e Luiz Antônio Silveira Lopes
- 16** • A viabilidade da implantação de medidas de redução de consumo de água em prédios de Brasília/DF
Jorge Thiago Duarte da Silva Vianna
- 26** • Determinação do tirante mínimo da hidrovia do Tocantins e Araguaia
Marcelo de Miranda Reis, Cassia Vanessa de Castro, José Carlos Cesar Amorim
- 37** • Uma arquitetura baseada em micro serviços para apoio a transparência em processos de compras E-Gov
Marco A Luís Vinicius Pinho Bueno de Carvalho, Ricardo Choren Noya
- 49** • Motivação para a cartografia colaborativa: um experimento no Exército Brasileiro
Eduardo Guedes, Luciano Augusto Terra Brito
- 60** • Um algoritmo de aprendizado iterativo baseado no ciclo de marcha humana para a locomoção de um robô humanoide
Fábio Suim Chagas, Luis David Peregrino de Farias, Matheus Bozza, Paulo Fernando Ferreira Rosa
- 70** • Data Modeling for Cassandra
Thiago B Leão, Maria Claudia R Cavalcanti, Raquel C G Pinto
- 83** • Uma proposta de ferramenta para a experimentação do aprendizado profundo no reconhecimento da fala para o idioma Português
Julio Cesar Duarte, Matheus Rodrigues Affonso, Rebeca Pinheiro dos Reis

Monitoramento da influência da velocidade de circulação dos trens na força de impacto por meio de sensores de fibra óptica

Danilo Martelli* e Luiz Antônio Silveira Lopes
Instituto Militar de Engenharia (IME)
Praça General Tibúrcio, 80, CEP:22290-270,
Praia Vermelha, Rio de Janeiro/RJ, Brasil,
*danmec04@yahoo.com.br

RESUMO: Este artigo consiste em uma demonstração do funcionamento de um equipamento novo no Brasil, que faz uso de sensores de fibra óptica para a medição de força de impacto que as rodas de veículos ferroviários causam nos trilhos da via permanente. Ademais, abrange a análise da relação existente entre a velocidade do trem e a força de impacto. De posse dos dados, conclui-se que não foi possível estabelecer uma equação que correlacione satisfatoriamente força de impacto e velocidade, segundo a análise de Regressão Linear Simples, dada a variabilidade dos veículos ferroviários. Entretanto, é possível a construção de envoltórias de segurança com vistas a diminuir o impacto na operação ferroviária e os riscos de acidentes e ocorrências com trilhos quebrados e rolamentos travados.

PALAVRAS-CHAVE: Envoltória de segurança. Força de Impacto. Roda. Trem. Trilho.

ABSTRACT: This article is a demonstration of the operation of a new equipment in Brazil, which uses fiber optic sensors to measure the impact force that rail vehicles wheels cause on permanent track rails. In addition, it covers the analysis of the relationship between train speed and impact force. With the data, it was concluded that it was not possible to establish a single equation that satisfactorily correlates impact force and speed, according to the analysis of Simple Linear Regression, given the variability of railway vehicles. However, it is possible to construct safety curves to reduce the impact on the railway operation and the risks of accidents and occurrences with broken rails.

KEYWORDS: Impact Force. Safety curve. Trail. Train. Wheel.

1. Introdução

Nos Estados Unidos, entre 2001 e 2010, os descarrilamentos que ocorreram com maior frequência foram ocasionados por quebras de trilhos ou soldas [1], como demonstra a **figura 1**. No Brasil, os dados oficiais da ANTT (de 2006 a 2013) mostram apenas o percentual de acidentes por circunstâncias macro, sendo a via permanente a maior causa (aproximadamente 40% dos acidentes) [2].

Uma empresa do setor logístico brasileiro ligada ao transporte ferroviário, tendo em vista todo o impacto financeiro gerado¹, decidiu monitorar a força de impacto pela qual a via permanente (mais

especificamente o trilho) é solicitada na passagem do trem. Com isso, fez a aquisição de um equipamento de detecção de impacto de rodas.

Durante a realização da revisão bibliográfica para implantação do equipamento, verificou-se a possibilidade de haver relação entre a força de impacto e a velocidade. A **figura 2** mostra todas as possibilidades de relações existentes:

- inversamente proporcional: defeitos em fase inicial (pequenos), que causam forças de impacto altas quanto menor for a velocidade;
- constante: defeitos em fase intermediária (médios), que causam sempre a mesma de força de impacto, independentemente da velocidade;

¹ Interdição de tráfego, danos a veículos ferroviários e via permanente, por causa de falhas e acidentes causa trilho, tala ou solda quebrados, ocorridos na malha sobre sua concessão.

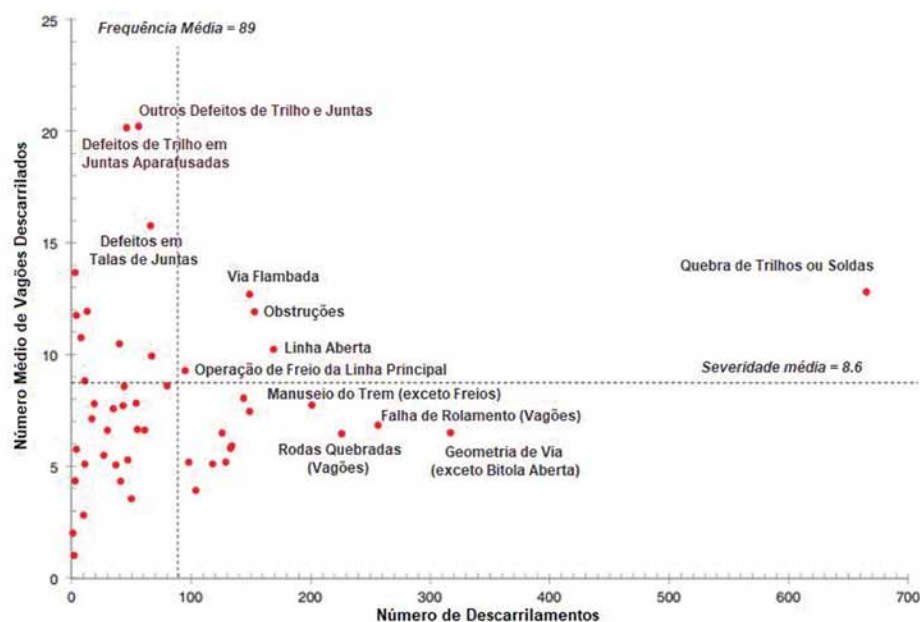


Fig. 1 - Gráfico de frequência e severidade de descarrilamentos na linha principal das ferrovias de carga Class I dos EUA. Fonte: [1], traduzido pelo autor.

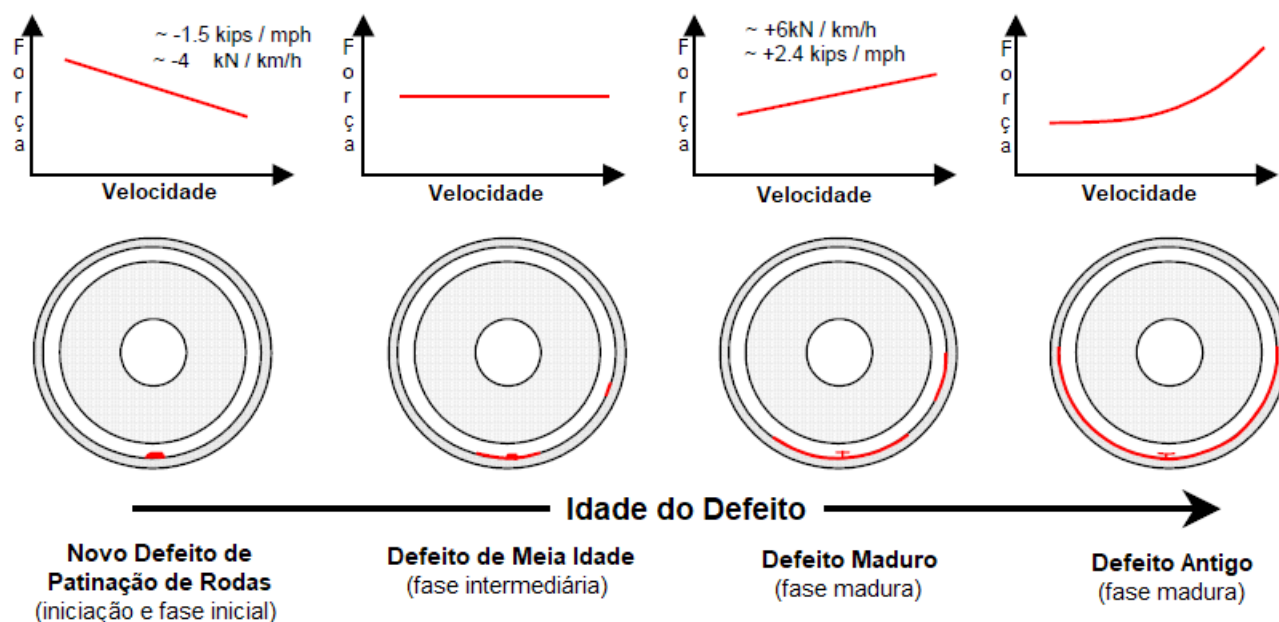


Fig. 2 - Prováveis curvas de velocidade *versus* força de impacto, dependendo da maturidade do defeito. Fonte: [3], traduzido pelo autor.

- diretamente proporcional: defeitos em fase avançada (grandes), que causam forças de impacto altas quanto maior for a velocidade;
- exponencial: defeitos em fase muito avançada (ainda maiores), que causam forças de impacto ainda mais altas quanto maior for a velocidade;

Dessa forma, o objetivo desse estudo foi relacionar a força exercida nos trilhos (que pode ser acrescida de forças dinâmicas decorrentes de defeitos de rodas) com a velocidade de circulação dos comboios. O detector de impacto adquirido faz uso de sensores de fibra óptica instalados sob o trilho para realizar a medição da força exercida pelos veículos na via.

O equipamento foi instalado na região de Itú/SP, como mostra a **figura 3**, onde a via permanente é de bitola mista, ou seja, possui três trilhos para a passagem de trens de bitola larga (1,6 metro) e bitola métrica (1,0 metro). Faz a medição de trens de três empresas ferroviárias diferentes, as quais passam pelo local (a detentora da concessão e duas que pagam direito de passagem, uma com trens de bitola larga e outra com trens de bitola métrica).

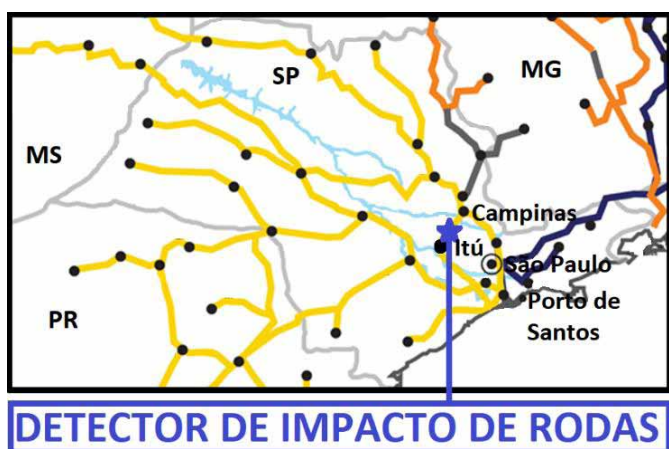


Fig. 3 - Diagrama esquemático da malha ferroviária do estado de São Paulo, focando local de instalação do detector de impacto. Fonte: [4], modificado pelo autor.

2. Teoria contato roda-trilho

Em 1881, o físico alemão Heinrich Hertz apresentou sua teoria sobre contato, publicando em

1882 um artigo intitulado “On the Contact of elastic solids” [5]. No estudo, Hertz afirma que a região de contato teórica entre dois corpos sobrepostos é um ponto, e que, ao aplicar uma carga conhecida, normal a região de contato, forma-se uma área finita, a qual varia de acordo com a forma geométrica dos corpos que estão em contato. Como exemplo, dois corpos sobrepostos que possuam curvaturas distintas e eixos principais perpendiculares entre si, possuem área de contato elíptica, como é o caso do contato entre a roda e trilho ferroviários [5].

A teoria de Hertz ganhou destaque apenas em meados do século XX, sendo a ferrovia a principal beneficiária de seus estudos, notadamente no que tange ao entendimento da interação entre roda e trilho [5]. A **figura 4** exemplifica a área de contato entre roda e trilho, a chamada elipse de Hertz.

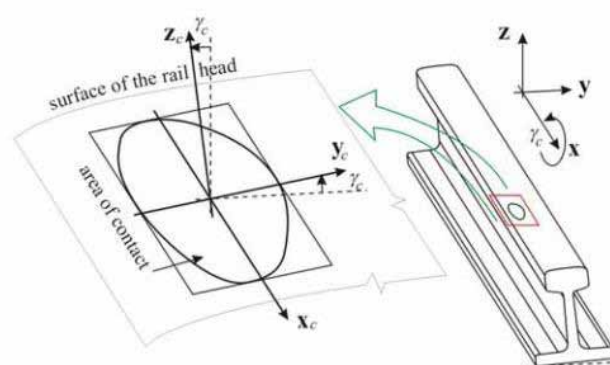


Fig. 4 - Exemplo de área de contato elíptica entre roda e trilho. Fonte: [6].

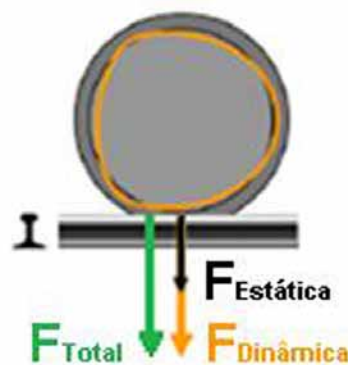


Fig. 5 - Esquema das forças existentes entre roda e trilho. Fonte: [6], modificado pelo autor.

A chamada força de impacto ocorre quando existe um defeito na roda, ou no trilho, fazendo com que a área de contato, vista na **figura 4**, não aconteça.

A Força Total de Impacto é composta por duas forças: a Força Estática, que representa a força peso do veículo, dividida pela quantidade de rodas que nele existem, e a Força Dinâmica, que representa a magnitude da força ocasionada por um defeito existente na roda, conforme exemplificado na **figura 5**.

O manual americano de normas e práticas recomendadas da Association of American Railroads (AAR) [8], que para a ferrovia brasileira é somente uma referência, considera impacto de rodas prejudicial à via permanente quando a Força Total for maior ou igual a 90 kips (400 kN). Além disso, o manual de campo da AAR [9] também faz referência à 65 kips (289 kN), a partir da qual é preciso realizar tratamento por oportunidade. A Federal Railroad Administration (FRA) [10] também faz uso dos limites referenciados no Manual de Campo da AAR.

3. Detector de impacto de rodas

Segundo Weed e Lonsdale [11], existem, potencialmente, diversas variáveis que podem influenciar na diferença de leitura dos detectores de impacto durante a passagem do trem. Dentre essas variáveis, pode-se citar: tipo de sensor utilizado, velocidade do trem, tipo de dormente usado e rigidez e estabilidade do subleito. Os autores ainda afirmam que conforme aumenta a velocidade, a carga registrada também aumenta.

O sistema de detecção de impacto utilizado para realizar as medições apresentadas nesse estudo consiste de: um computador servidor (localizado na sede da empresa), no qual está instalado o programa gerenciador do equipamento (supervisório); um interrogador ótico, que faz o envio e a interpretação dos sinais, localizado ao lado da via permanente; e 12 sensores de Bragg em fibra ótica (6 em cada trilho da bitola larga), como pode ser visto na **figura 6**.



Fig. 6 - Foto dos sensores instalados na via permanente. Fonte: [12], modificado pelo autor.

As redes de Bragg em fibra ótica (**figura 7**) são microestruturas inscritas de forma permanente no núcleo de uma fibra ótica monomodo (transmissão de apenas um sinal de luz). Elas refletem um espectro estreito de luz, centrado no comprimento de onda de Bragg, no momento em que iluminadas por uma fonte de espectro largo [13].

A cada passagem de uma roda sobre os sensores, o equipamento mede toda a circunferência dela pelo menos uma vez (podendo chegar a medir até 2 ou 3 vezes, dependendo do diâmetro da roda) [7].

Os sensores medem cargas de impacto que não são exercidas diretamente sobre cada um deles. A faixa de medição está limitada no software de análise a 3,5m em cada direção, ou seja, a faixa de medição de cada sensor cobre no máximo 7m [14].

Cada sensor recebe do interrogador ótico um feixe de luz, que é conduzido por um fio de fibra ótica. Este feixe, ao chegar no sensor, encontra uma combinação de lentes otimizadas que o dividem em vários feixes, os quais incidem em um refletor ótico, que, quando em repouso, fica posicionado paralelamente ao último espelho, fazendo com que os raios de luz incidentes sejam refletidos na mesma direção em que incidiram, como demonstrado na **figura 8**.

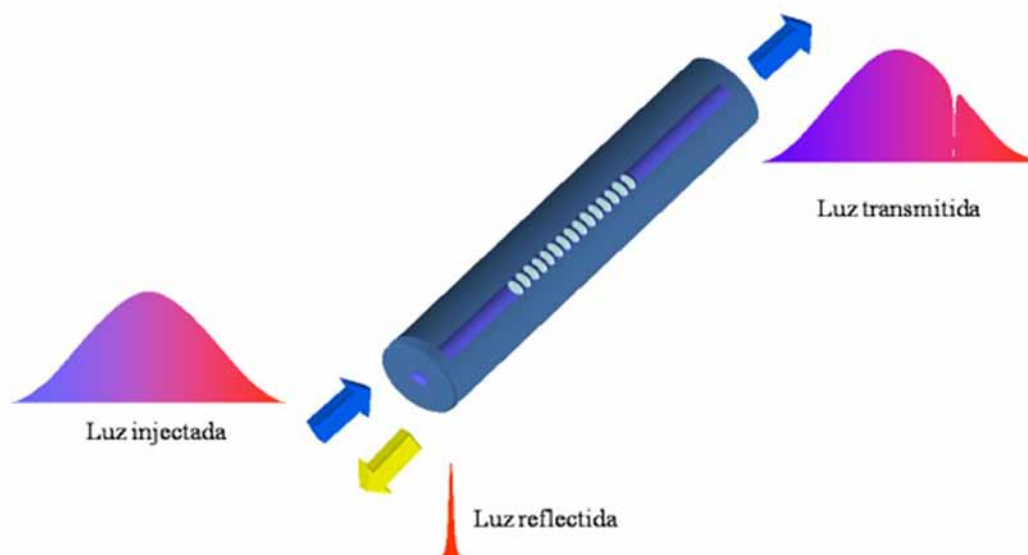


Fig. 7 – Esquema de operação das redes de Bragg. Fonte: [13].

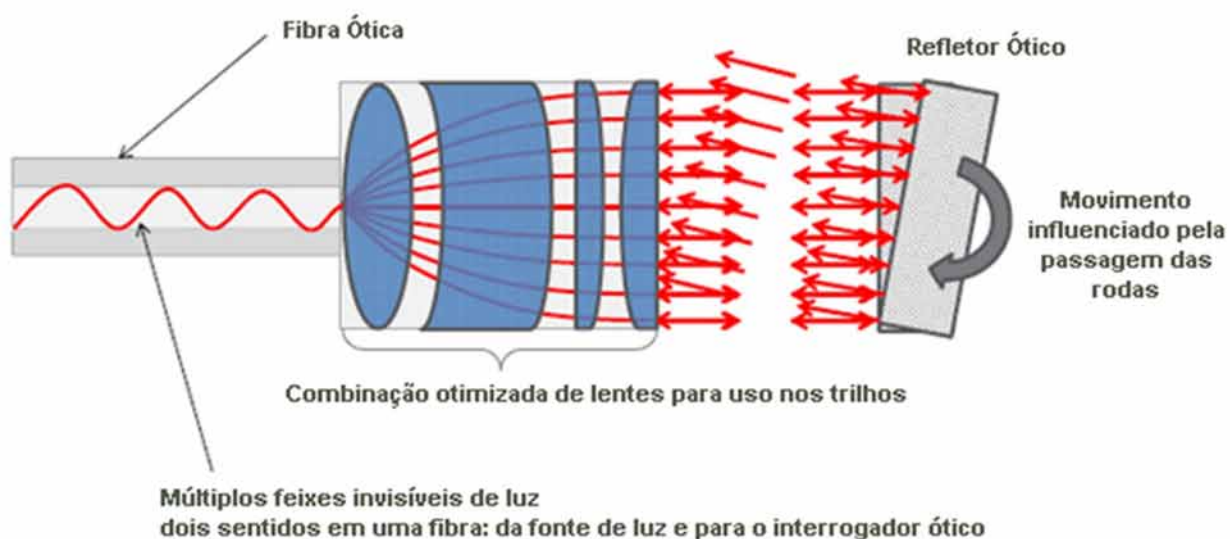


Fig. 8 -Esquema de funcionamento de cada sensor de fibra ótica. Fonte: [12].

Quando o trem passa por um sensor, o refletor ótico oscila e os feixes de luz são reenviados para os espelhos em uma direção diversa da anterior. Assim, os feixes são convertidos novamente em um único, chegando ao interrogador ótico no formato de um sinal, correspondente à deflexão do trilho [14]. Caso haja um impacto, a oscilação do refletor fará com que

o sinal tenha uma forma distinta, que neste caso será como um “pulso”, conforme a região “D” da **figura 9**.

A **figura 9** mostra o sinal padrão de medição (deflexão do trilho) na passagem de uma roda com defeito em apenas um dos sensores do detector do impacto de rodas. A posição A representa REPOUSO, ou seja, não existe carga sobre o sensor. A posição B

representa a APROXIMAÇÃO DA RODA, ou seja, o sensor está detectando a roda passando pelo dormente logo antes do mesmo. A posição C representa a FLEXÃO MÁXIMA, ou seja, a roda está em cima do sensor. A posição D representa o DEFEITO. Nesse caso, a roda estava se afastando do sensor, passando sobre o outro dormente, mas havia um defeito que gerou o impacto detectado como um “pulso”.

O equipamento, através de um algoritmo, faz o tratamento do sinal e registra a Força Estática. Na sequência, verifica a amplitude do impacto, caso exista, e soma o valor à Força Estática, chegando ao valor da Força de Total ou Força de Pico.

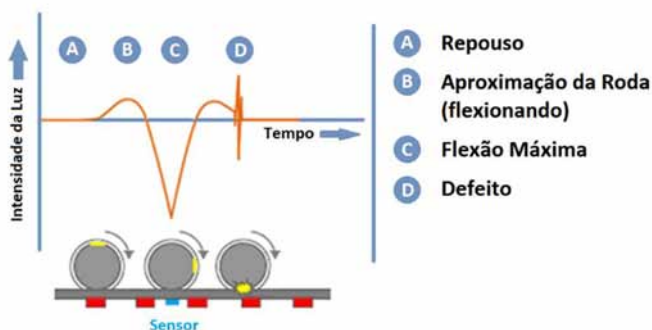


Fig. 9 - Leitura do sinal característico da passagem da roda com defeito sobre um sensor. Fonte: [12], modificado pelo autor.

O registro da velocidade é feito em dois momentos distintos: na entrada e na saída da composição no equipamento. Para isso, existem dois transdutores de distância (contadores de eixo), posicionados de forma a calcular o tempo consumido para percorrer a distância, pelo primeiro e pelo último eixo.

Uma das vantagens da utilização do sensor de fibra ótica é a não interferência eletromagnética, que pode existir nos outros tipos de sensores. A sua precisão é de mais ou menos 3% de erro na medição de massa, conforme informações do fabricante [12].

Os sensores são calibrados durante sua confecção. A verificação da calibração, após a instalação do equipamento, pode ser feita de duas formas distintas:

- Manual - utilizando vagões aferidores (composição formada por vagões carregados

aferidos em balança – massas conhecidas), os quais são movimentados sobre o equipamento esporadicamente para comparação manual e posterior correção da medição da massa com o valor conhecido;

- Automática – utilizando um veículo padrão, pré-cadastrado, com massa conhecida, para que a cada passada desse veículo, o equipamento se autocalibre.

4. Metodologia

A metodologia do trabalho consistiu, em seu primeiro passo, na aquisição do aparelho de detecção de impacto de roda, na análise do melhor local para instalação (linha tangente que não necessita de aplicação de freio do trem) e posterior instalação.

O equipamento foi instalado em dezembro de 2014 no local citado na **figura 3**. Inicialmente a instalação consistiu na confecção de um alicerce de alvenaria e uma gaiola de trilhos (antivandalismo) para alojamento do armário que abrigaria o interrogador óptico. Na sequência, foram instalados os sensores na via de acordo com o esquema apresentado na **figura 10**.

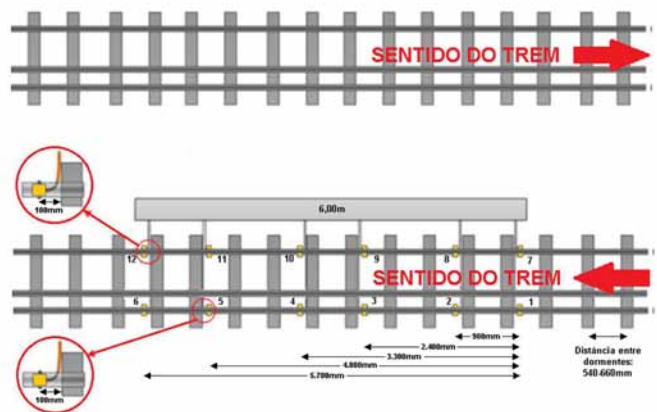


Fig. 10 - Esquema de instalação dos sensores na via permanente. Fonte: [15], modificado pelo autor.

Para a instalação dos sensores, fez-se necessária a preparação da via permanente. Retirou-se o lastro da posição onde os 12 sensores seriam instalados, para eliminar a oxidação da parte inferior do patim. A seguir, buscou-se garantir um bom contato entre

sensor e trilho, e evitar o contato entre os sensores e o lastro na passagem do trem sobre o equipamento.

A interrupção do tráfego na linha, durante a instalação, demandou curto intervalo de tempo, sendo necessários, aproximadamente, 4 minutos para a instalação de cada sensor (entre limpeza de lastro e oxidação, fixação e torqueamento dos parafusos). A **figura 11** apresenta um dos sensores instalado no patim de um dos trilhos.

O próximo passo foi o envio dos dados coletados, através de modem 3G, para uma equipe que passou a tabulá-los em planilhas de Excel. A partir daí, foram feitas verificações acerca da necessidade de calibração do equipamento.

Posteriormente, foi realizado o cadastro da locomotiva da General Eletric (GE) AC 44i como

massa padrão, a qual realiza a autoverificação da calibração a cada passagem. Devido à grande variação de “peso por eixo” dos veículos, já que a frota que passa pelo equipamento é bem diversificada (vagões de 4 eixos, com capacidades de carga diferentes – 80, 100 e 130 toneladas de carga bruta total - e locomotivas de 4, 6 ou 8 eixos, com massas diferentes), os dados considerados nas análises foram os de Força Dinâmica, desconsiderando assim a Força Estática.

Finalmente, foram realizadas regressões lineares, usando o software Minitab, na tentativa de correlacionar as grandezas em análise (Força e Velocidade). A técnica de regressão linear adotada no estudo foi o Método dos Mínimos Quadrados. A **figura 12** mostra um resumo da metodologia utilizada:



Fig. 11 – Sensor de fibra ótica instalado sob o trilho. Fonte: [15].

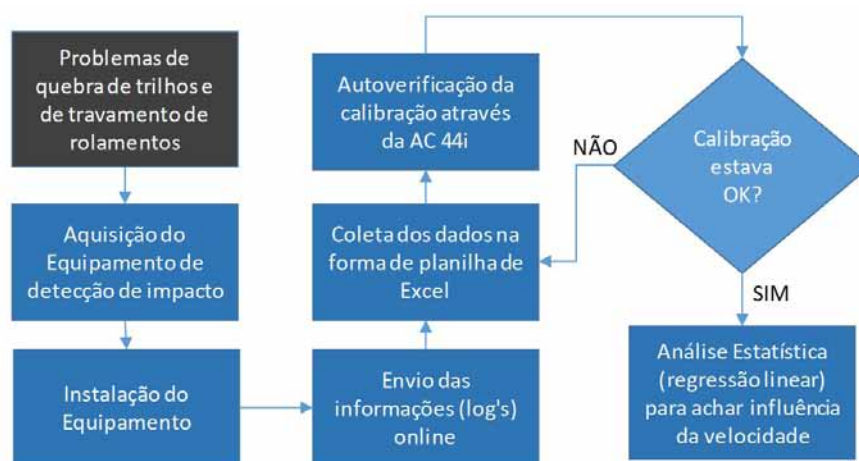


Fig. 12 – Fluxograma das etapas desenvolvidas no estudo. Fonte: O autor.

5. Análise dos resultados

A primeira coleta de dados foi realizada de dezembro de 2014 até junho de 2015, com o intuito de verificar a calibração dos sensores. A análise desses dados mostrou a necessidade de realização de uma calibração manual, a qual foi feita em junho de 2015. Além disso, foi realizada também a configuração da locomotiva AC 44i para fazer a autoverificação da calibração. Em vista disso, a primeira coleta de dados foi descartada para análise, e os dados que foram utilizados nesse estudo foram coletados entre os dias 25/06/2015 e 05/05/2016, durante uma segunda coleta.

De posse dos dados de força de impacto e velocidade, foi possível plotar no Minitab um gráfico de dispersão, que pode ser visto na **figura 13**. E, utilizando o recurso de regressão linear, obter a **tabela 1** de análise de variância.

Tab 1 – Análise de variância de todos os dados.

Fonte	GL	SQ	QM	F	P
Regressão	1	5445549	5445549	571,50	0,000
Erro	4325	41210687	9528		
Total	4326	46656236			

Fonte: O autor.

GL – Graus de Liberdade

SQ – Soma de Quadrados

QM – Quadrado Médio

F – Teste F

P – P-valor

Analizando o gráfico e a **tabela 1**, verifica-se uma grande dispersão entre os dados de força de impacto e velocidade, o que significa que não existe uma boa correlação linear ou não linear entre os dois parâmetros. O desvio padrão e o coeficiente de

determinação (R^2) encontrados na regressão linear não são representativos: o valor do desvio padrão é muito alto e o valor do R^2 , muito baixo (11,7%).

Como os dados acima referem-se a vários veículos diferentes (1638), decidiu-se analisar como foi a evolução do impacto em relação à velocidade para cada veículo separadamente. Dessa forma, foram eliminados os dados de força de impactos de veículos que registraram menos de 4 impactos na passagem pelo equipamento.

Por conseguinte, a análise restringiu-se a 231 veículos que apresentaram mais de 5 impactos cada, totalizando 41% dos dados da população (1644 impactos). O resultado das análises de cada veículo foi a existência de correlação diretamente proporcional, inversamente proporcional e inexistência de correlação entre força de impacto e velocidade, como pode ser visto com mais detalhes nas próximas seções (5.1 e 5.2):

5.1 Regressão linear (relação diretamente proporcional)

Dos 231 veículos, 12 (5,2% dos veículos da amostra), que representam 232 impactos (7,2% dos impactos da amostra), apresentaram correlação forte entre força de impacto e velocidade, tendo coeficiente angular variando de 4,503 a 13,33 e R^2 de 71,2% a 90,2%.

Para melhor análise, plotou-se dois gráficos. O primeiro com 8 veículos que apresentaram coeficientes angulares mais baixos, conforme **figura 14**. O outro com os 4 veículos, que apresentaram coeficientes angulares mais altos, conforme **figura 15**.

A **tabela 2** apresenta os dados obtidos através da regressão linear de cada veículo. Para esses casos, é possível afirmar que existe correlação forte entre força de impacto e velocidade, sendo possível escrever uma equação para cada veículo.

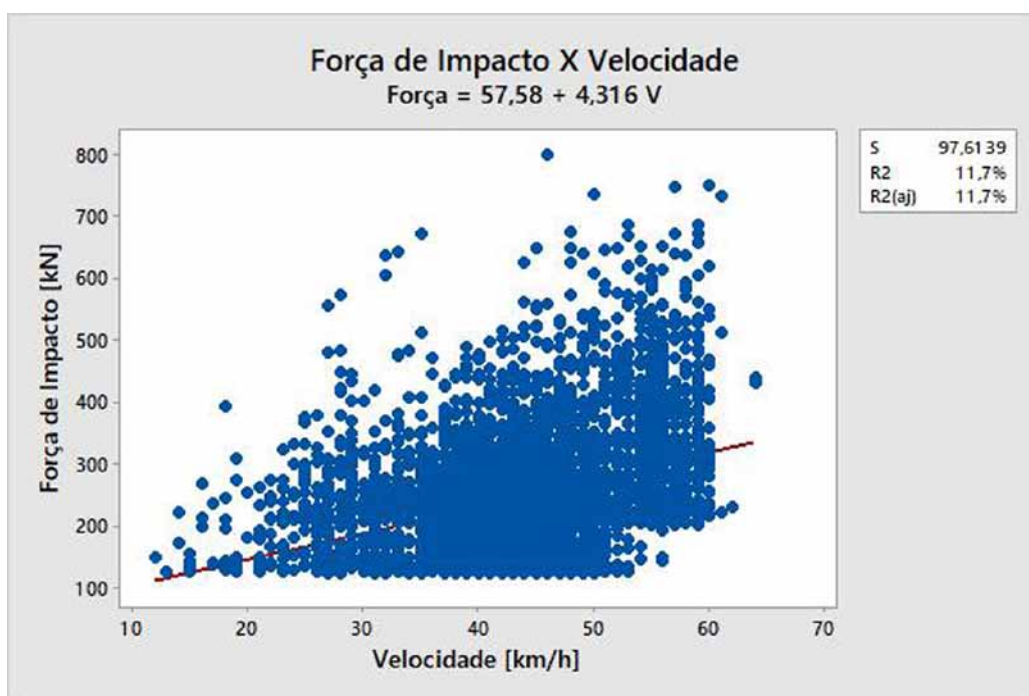


Fig. 13 – Gráfico de dispersão de força de impacto por velocidade. Fonte: O autor.

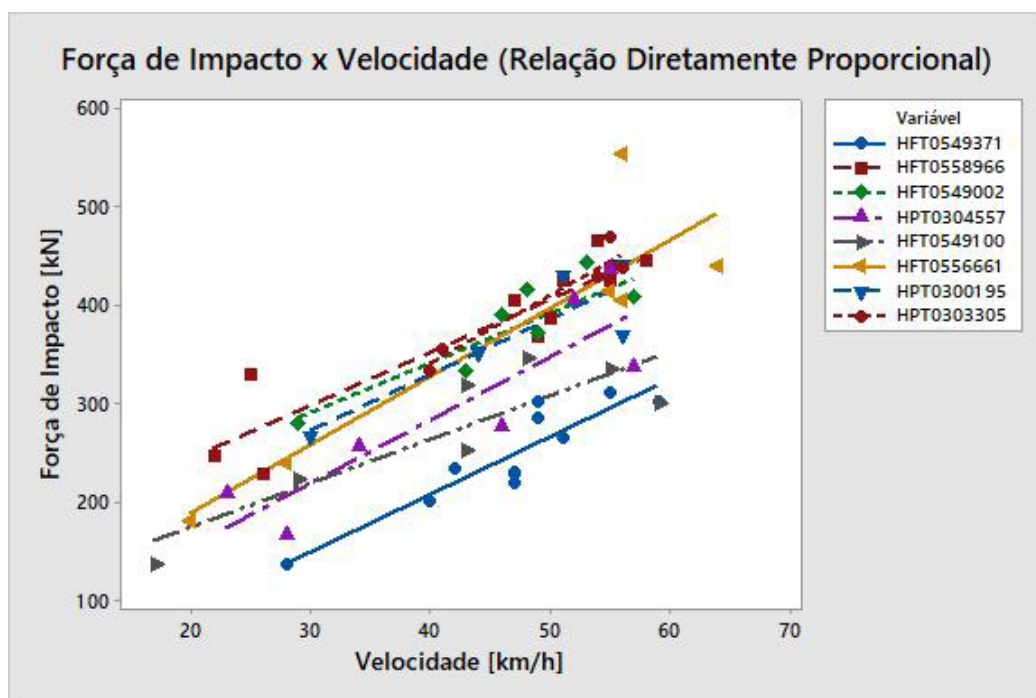


Fig. 14 – Gráfico de dispersão sobreposto dos 8 veículos. Fonte: O autor.

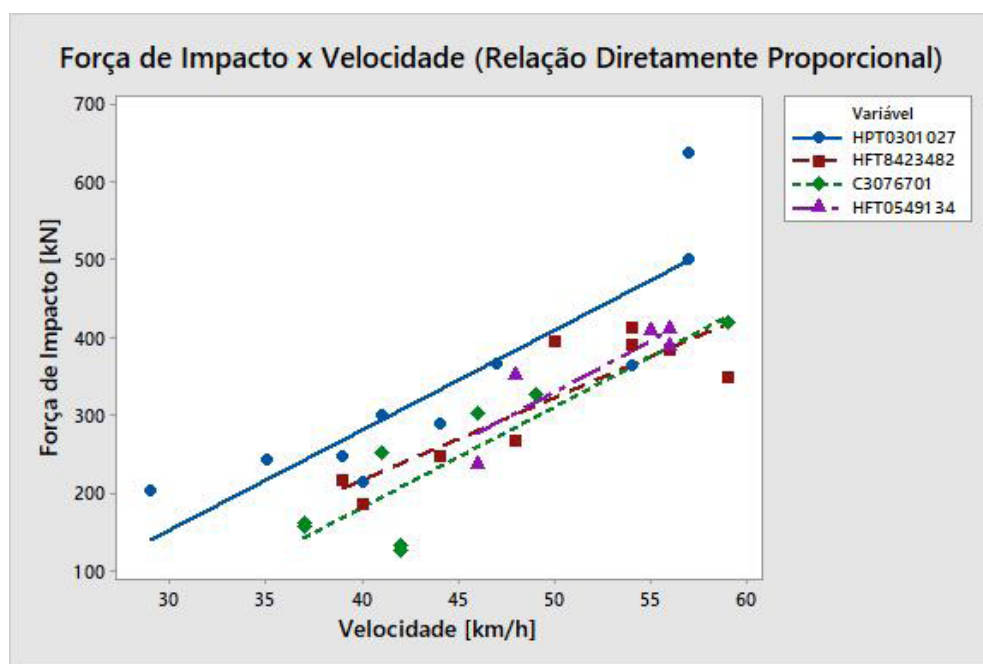


Fig. 15 – Gráfico de dispersão sobreposto dos 4 veículos. Fonte: O autor.

Tab. 2 - Resumo das análises de variância dos 8 veículos.

Veículo	Coef. Angular	Coef. Linear	S	R ²	R ² (aj.)
HFT 0549371	5,898	-29,30	22,9248	83,0%	81,2%
HFT 0558966	5,425	135,3	31,9989	85,3%	83,7%
HFT 0549002	5,074	138,7	26,5906	77,8%	74,1%
HPT 0304557	6,419	26,34	51,2744	78,0%	73,6%
HFT 0549100	4,503	83,30	39,0108	77,5%	73,0%
HFT 0556661	6,946	48,52	67,0434	81,1%	76,4%
HPT 0300195	5,672	102,4	33,7634	77,4%	71,7%
HPT 0303305	6,966	61,26	18,0823	92,7%	90,2%
HPT0301027	12,87	-234,1	70,8414	76,8%	73,9%
HFT8423482	10,60	-207,0	46,8023	74,8%	71,2%
C3076701	13,03	-341,3	56,1452	76,8%	72,9%
HFT0549134	13,33	-337,4	38,8829	78,4%	71,3%

Fonte: O autor.

S – Erro Padrão

R² – Coeficiente de Determinação

R²(aj.) – Coeficiente de Determinação Ajustado

5.2 Regressão linear (relação inversamente proporcional)

Nesse caso, dos 231 veículos, 4 (1,7% dos veículos da amostra), que representam 52 impactos (1,6% dos impactos da amostra), apresentaram correlação forte entre força de impacto e velocidade, tendo coeficiente

angular variando de -4,210 a -1,307 e R² de 84,0% a 95,1%. A figura 16 apresenta um gráfico com os dados dos 4 veículos.

A tabela 3 é o resultado da regressão linear de cada veículo, a qual apresenta coeficientes angulares negativos, ou seja, relação inversamente proporcional.

Tab. 3 – Resumo das análises de variância dos 4 veículos.

Veículo	Coef. Angular	Coef. Linear	S	R ²	R ² (aj.)
HFS0527629	-2,814	261,3	8,93389	88,0%	85,0%
HPT0304417	-1,307	274,0	2,29105	96,4%	95,1%
HFS0525219	-3,934	449,5	12,9864	89,7%	86,3%
FTT0323209	-4,210	412,2	8,98870	88,0%	84,0%

Fonte: O autor.

S – Erro Padrão

R² – Coeficiente de Determinação

R²(aj.) – Coeficiente de Determinação Ajustado

Nesses casos, a força de impacto diminuiu com o aumento da velocidade, ou vice-versa, representando um possível defeito na fase inicial.

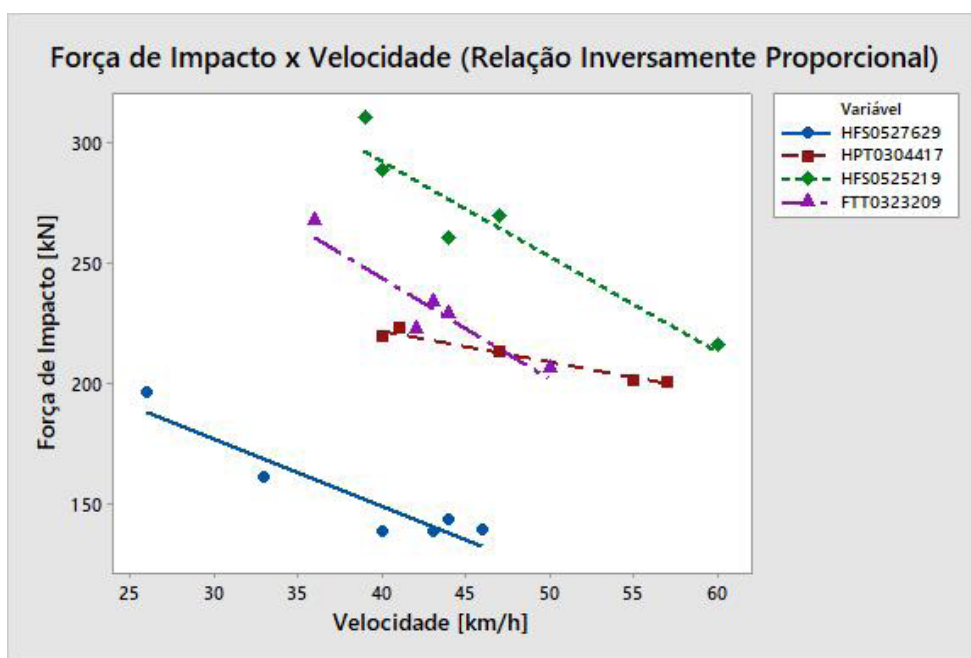


Fig. 16 – Gráfico de dispersão sobreposto dos 4 veículos. Fonte: O autor.

5.3 Correlação entre força de impacto e velocidade

Perante as análises feitas anteriormente, foi possível escrever 16 equações diferentes, uma para cada veículo. Foram correlacionadas força de impacto e velocidade, mas essas representam apenas 6,9% dos veículos e 8,8% dos impactos da amostra.

Sendo assim, 93,1% dos veículos (215) e 91,2% dos impactos (1361) da amostra não apresentaram correlação significativa entre força de impacto e velocidade, ou seja, a correlação entre os dois parâmetros não é significativa para toda a população. Assim, foi desnecessário criar uma equação que normalize os valores de força medida com a velocidade.

Voltando à **figura 13**, é possível traçar curvas envoltórias (não lineares), as quais funcionam como “Linhas Limite de Assertividade” dos dados. A **figura 17** apresenta o gráfico de dispersão de todos os dados, com três curvas envoltórias: uma que compreende 99,9% (curva preta) dos dados; outra que compreende 95,2% (curva vermelho claro); e a última que compreende 90,1% (curva vermelho escuro).

Como exemplo, a **tabela 4** mostra que, ao se adotar como referência a velocidade de 40 km/h, 99,9% dos

dados de força de impacto estão abaixo de 744 kN; 95,2% dos dados de força de impacto estão abaixo de 421 kN e 90,1% dos dados de força de impacto estão abaixo de 369 kN.

Tab. 4 - Análise de variância de todos os dados.

Velocidade [km/h]	99,9%	95,2%	90,1%
12	146	146	146
20	400	263	241
30	601	355	316
40	744	421	369
50	855	472	410
60	946	514	444

Fonte: O autor.

Sendo assim, essas curvas podem ser usadas como envoltórias de segurança. Para o caso de um veículo, que ao passar pelo detector de impacto com uma velocidade de 40 km/h, registra uma força de impacto de 400 kN, pode-se tomar a decisão de reduzir a velocidade do trem para, por exemplo, 20 km/h, tendo a certeza de que a força de impacto será reduzida para aproximadamente 260 kN (com 95,2% de certeza), tornando-se menos prejudicial aos trilhos e rolamentos. Dessa forma, é possível efetuar a redução de velocidade do trem, ao invés de pará-lo para manobra do vagão com defeito.

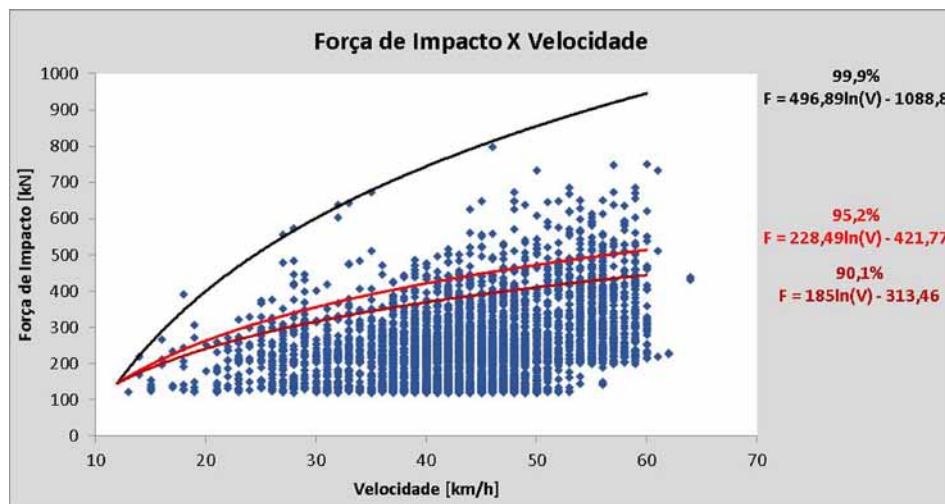


Fig. 17 – Gráfico de dispersão com curvas envoltórias. Fonte: O autor.

6. Considerações finais

Perante todos os dados apresentados, pode-se concluir que a obtenção de uma equação empírica que correlacione, por meio dos dados experimentais, as grandezas força e velocidade, não se verifica satisfatória para todos os veículos ferroviários utilizados nas medições de campo desse estudo. Apenas 16 veículos em 231 apresentaram correlação forte entre os parâmetros, variando o coeficiente angular de -4,21 a 13,33, ou seja, de correlação inversamente proporcional a diretamente proporcional.

Neste estudo, não foi aplicada a análise de regressão não linear entre força de impacto e velocidade para cada veículo, pois a amostra não sugeriu esta possibilidade, como pode ser visto na **figura 13**.

Todavia, foi possível traçar envoltórias de segurança utilizadas para decisões operacionais quanto ao tráfego dos trens.

Também foi possível concluir a efetividade da nova tecnologia inserida no Brasil através do equipamento de detecção de impacto com sensores de fibra ótica. Além de necessitar de pouca manutenção (autoverificação da calibração feita automaticamente), não apresentar interferência eletromagnética.

Como sugestão para trabalhos futuros, indica-se inserir o tipo de defeito da roda como um parâmetro de análise. Assim poderão realizar um estudo de força de impacto versus velocidade para cada tipo de defeito de roda (shelling, spalling, calo), de acordo com o tamanho de cada defeito.

Referências Bibliográficas

- [1] Liu, X., Saat, M. R., & Barkan, C. P. L., Analysis of Major Derailment Causes on Heavy Haul Railways in the United States, International Heavy Haul Association Conference, New Delhi, India. 2013. Disponível em: <http://railtec.illinois.edu/wp/wp-content/uploads/2019/01/Liu%20et%20al%202013.pdf>
- [2] Agência Nacional de Transportes Terrestres – ANTT. Relatório 01/2014/GEROF/SUFER/ANTT, 2014. Disponível em: <https://www.gov.br/antt/pt-br>. Acessado em: 26 set. 2020.
- [3] Trackside Intelligence PTY LTDA. WMS Data System WMS Web User Guide. Austrália: TRACK IQ, 2012. 73 p.
- [4] Associação Nacional dos Transportes Ferroviários. Histórico. ANTF, 2016. Disponível em: <http://www.antf.org.br/historico/>. Acesso em: 10 dez. 2016.
- [5] Santos, Francisco de Carvalho. Análise do Contato Roda-Trilho e sua Influência na Vida em Serviço de Rodas Ferroviárias. 2000. 151 f. Dissertação (Mestrado) - Curso de Engenharia Mecânica, Departamento de Projeto Mecânico, Universidade Estadual de Campinas, Campinas, 2000.
- [6] Rovira, A. et al. Experimental and numerical modelling of wheel-rail contact and wear. Wear, [s.l.], v. 271, n. 5-6, p.911-924, jun. 2011. Elsevier BV. Disponível em: <http://dx.doi.org/10.1016/j.wear.2011.03.024>. Acesso em: 10 dez. 2016.
- [7] Voestalpine Company. ATLAS FO – Wheel Defect Detection. Alemanha: VAE, 2014. 10 p. Folheto elaborado para divulgação do produto aos consumidores.
- [8] Association of American Railroads. Manual of standards and recommended practices: section G-II: Wheel and Axle Manual. Washington, D.C: AAR, 2012.
- [9] Association of American Railroads. 2015 Field Manual of the AAR Interchange Rules. Rule 41. Washington, D.C: AAR, 2015
- [10] U.S. Department of Transportation Federal Railroad Administration. An Implementation Guide for Wayside Detector Systems. Washington, D.C: FRA, 2019
- [11] Weed, Dennis; Lonsdale, Cameron. Review and Analysis of Wheel Impact Load Detector (WILD) and Wheel Removal Data. Mechanical Association Railcar Technical Services (MARTS). 2004. Disponível em: [http://marts-rail.org/2009 Presentations/2004 Presentations/MARTS Weed-Lonsdale 2004.pdf](http://marts-rail.org/2009%20Presentations/2004%20Presentations/MARTS%20Weed-Lonsdale%202004.pdf). Acesso em: 03 nov. 2015.
- [12] Voestalpine Company. Detecção de impacto de Roda e Pesagem Dinâmica, Brasil: VAE, 2016. 12 slides, color.
- [13] Barbosa, Maria Cristina Beirão. Caracterização e Avaliação dos Efeitos do Estado de Conservação de Rodas de Composições Ferroviárias na Infra-Estrutura. 2009. 62 f. Dissertação (Mestrado) - Curso de Especialização em Estruturas, Engenharia Civil, Universidade do Porto, Porto, 2009. Disponível em: <https://repositorio-aberto.up.pt/bitstream/10216/58853/1/000136926.pdf>. Acesso em: 27 set. 2020.
- [14] Voestalpine Company. ATLAS FO – System Description: Alemanha: VAE, 2013. 21 p. Folheto elaborado para divulgação do produto aos consumidores.
- [15] Voestalpine Company. Training ATLAS FO – Sensor Installation: Alemanha: VAE, 2014. 21 slides, color.

A viabilidade da implantação de medidas de redução de consumo de água em prédios de Brasília/DF

Jorge Thiago Duarte da Silva Vianna*

Universidade de Brasília

Brasília, DF, 70910-900

* jorge@thiago.us

RESUMO: O crescimento populacional, o uso indiscriminado da água e as mudanças climáticas vêm produzindo crises de abastecimento urbano de água. Nesse contexto, se encontra Brasília, uma área altamente urbanizada com predominância de nascentes, longos períodos de seca e risco de desabastecimento. Diante disso, este trabalho analisou a viabilidade dos métodos de redução de consumo de água em edificações residenciais no Plano Piloto, em Brasília. Para a análise, determinou-se o padrão de consumo das unidades habitacionais, os consumos de água dos aparelhos e a necessidade de obra para implantação. O texto considerou o padrão tipo de bloco referente ao Plano Piloto com seis andares construído sobre pilotis. Não foi utilizado nenhuma obrigatoriedade legislativa para instalação. Os economizadores ganharam destaque quanto a viabilidade e facilidade de instalação.

PALAVRAS-CHAVE: Conservação de água. Medidas redutoras. Viabilidade.

ABSTRACT: Population growth, the indiscriminate use of water and climate change have led to water supply crises. In this context, is Brasília, a highly urbanized area with a predominance of springs, long periods of drought and risk of shortages. Therefore, this work analyzed the feasibility of methods of reducing water consumption in residential buildings in the Plano Piloto, in Brasília. For the analysis, we determined the consumption pattern of housing units, the water consumption of the appliances and the need for construction work. The text considered the standard block type for the six-story Pilot Plan built on pilotis. No legislative requirement for installation was used. Savers have gained prominence in terms of feasibility and ease of installation.

KEYWORDS: Water conservation; Reduction measures; Viability

1. Introdução

A água é um recurso natural fundamental para a existência humana. Apenas 3% das águas do Planeta são doces e, somente um centésimo desse montante está acessível para o uso. Com o passar dos anos, ocorreu a poluição de alguns mananciais e o aumento expressivo da demanda mundial, influenciados pelo crescimento populacional e o uso indiscriminado de água. Tal fato, juntamente com questões climáticas, vem produzindo crises de abastecimento de água e limitando as atividades econômicas.

No Brasil, a Lei 9.433/97, que instituiu a Política Nacional de Recursos Hídricos, seguindo o conceito de uso sustentável que ganhou notoriedade no Relatório Brundtland (1987), estabelece a necessidade de assegurar às gerações atual e futuras, a disponibilidade de água necessária e com padrões de qualidade adequados aos seus usos. Entretanto, o crescimento

populacional e os aglomerados urbanos exercem pressão sobre os recursos hídricos. Esse aumento da pressão caracteriza-se pelo uso inadequado à água e contribui para os conflitos de uso.

Esse crescente consumo de água tende a aumentar a demanda mundial de água, de 55% até 2050, mesmo com as existências da desigualdade de acesso à água [1]. Já no Brasil, estima-se esse aumento em 28% até 2025 [2], enquanto a projeção do crescimento populacional é de cerca de 10% [3].

O consumo doméstico brasileiro de água ocupa o segundo lugar com 11% da demanda [4]. Nesse contexto, o presente estudo toma como exemplo o caso do Distrito Federal, altamente urbanizado, com altos consumos de água e inserido numa região de nascentes de água. Tais motivos levaram a capital do Brasil a ser auditada pelo TCDF (Tribunal de Contas do Distrito Federal), buscando averiguar se a quantidade de água produzida é suficiente para atender à demanda atual e futura. A conclusão desta auditoria foi que o DF

corre risco de desabastecimento. A disponibilidade hídrica máxima é de 8.820 L/s, enquanto o consumo registrado foi de 7.120 L/s [5]. Como providências de aumento da oferta, a CAESB iniciou uma série de obras de expansão na produção de água e suprimento de água até 2040, como, por exemplo, os sistemas de Corumbá 4, Bananal e Paranoá [6].

No ano 2000, o Distrito Federal apresentava disponibilidade hídrica inferior ao recomendado para um abastecimento normal, alertando para uma situação difícil. Atualmente, o Distrito Federal ocupa o primeiro lugar do Brasil em estresse hídrico, com média de disponibilidade de água de 0,96 km³/hab/ano [7], caracterizando uma escassez crônica, ou seja, quando se atinge uma disponibilidade entre 0,5 km³/hab/ano e 1 km³/hab/ano [8]. Tal fato ocorre devido ao alto crescimento populacional e à baixa disponibilidade de água, visto que a capital não tem grandes rios, encontra-se em área de nascentes e tem um longo período de estiagem, obrigando a construção de reservatórios que dependem da época chuvosa para recarga do sistema.

Em 2004, o Distrito Federal utilizava 67,7% da capacidade de abastecimento dos sistemas produtores de água [9]. Com as estiagens prolongadas, o reservatório do Descoberto, que atende mais de 60% da população de Brasília, atingiu menos de 20% de sua capacidade máxima em novembro de 2016. Tal situação levou a busca de alternativas de uso racional e conservação de água.

Com o aumento na demanda por água e escassez na oferta, situação típica dos grandes centros urbanos, agravada pelo consumo de energia das estações elevatórias, surge a necessidade de conservar a água. Para tentar suprir essa necessidade, a atenção volta-se para as regiões de maior consumo per capita do Distrito Federal: Plano Piloto e Lago Sul [10]. Como os imóveis no Lago Sul caracterizam-se por residências unifamiliares, o foco será aos grandes consumidores multifamiliares, que representam, isoladamente, as parcelas de maior consumo de Brasília. Esses locais destacam-se pelo alto consumo, caracterizado numa única fatura de água, visto que não apresentam medidores individualizados, concentrando o consumo

de água de várias unidades habitacionais. Uma das principais causas para o sobreconsumo registrado em edifícios residenciais recai sobre a inexistência de medidores individualizados, isto é, a concentração do cômputo do consumo em uma única conta, dificultando o controle. Há uma cultura equivocada e reprovável de que, à medida em que se paga por taxa igualmente dividida do consumo do grupo, a economia individual não compensa [11].

Diante do exposto, a abordagem deste trabalho consistiu na escolha de um estudo de caso composto por um imóvel multifamiliar vertical típico de Brasília. Neste imóvel foram analisadas algumas alternativas para redução do consumo de água, tendo dois casos: uma edificação existente e outra a construir.

2. Objetivos

Esta pesquisa tem por objetivo analisar as alternativas de redução de consumo de água em edificações residenciais no Plano Piloto, em Brasília, tendo duas abordagens: edificações existentes e a construir.

2.1 Objetivos específicos

Os objetivos específicos desta pesquisa são identificar a viabilidade das alternativas tecnológicas com melhor desempenho na redução do consumo de água através da:

- Análise de custos de implantação e manutenção;
- Economia de água; e
- Tempo de retorno do investimento.

3. Fundamentação teórica

No intuito de avaliar e conhecer as medidas de redução do de água, seu emprego, operação e restrições, foi necessário aprofundar o conhecimento na literatura especializada. Portanto, o presente capítulo trata dos conceitos, aplicações, indicadores e discussões a respeito do tema.

3.1 Estratégias para conservação de água

O uso eficiente é dividido em quantitativo (definido

a partir de critérios que busquem desempenho) e qualitativo (a partir de percepções socioculturais e da necessidade do usuário). O primeiro corresponde ao volume diminuído do consumo de água, foco desse trabalho, até atingir o volume estritamente necessário, respeitando os parâmetros ambientais, de saúde, higiene e vigilância sanitária. Um exemplo dessa aplicação refere-se ao uso de economizadores de água, como o arejador. O segundo tem relação com a ideia de incorporar a subjetividade e a perspectiva de quem usa o serviço [12]. Um exemplo desse item é a utilização de água não-potável nas bacias sanitárias.

O desenvolvimento e adequação tecnológica é responsável por inovar, voltando-se basicamente para a pesquisa [13]. Com isso, são concebidos produtos com melhor desempenho no uso de água. Exemplo desse uso é na forma de instalação de mictórios, com o uso de sensores e nas bacias com o uso de válvulas de acionamento duplo. Esse sistema, na verdade, são dois botões. Um faz com que o volume a ser liberado durante o acionamento seja a metade do máximo (usado para urina, em geral); enquanto o outro botão mantém a vazão plena (usado para fezes e sólidos, em geral). Desta forma, com o acionamento adequado à necessidade, consegue-se reduzir o volume de água utilizado.

3.2 Redução da pressão

Com o advento das edificações verticais, com cada vez mais andares, as pressões de água começaram a se elevar nos andares mais afastados do reservatório superior. Tal mudança começou a gerar a necessidade de implantação de dispositivos redutores de pressão por segurança na operação da tubulação, a fim de não ultrapassar o limite mecânico de resistência das paredes dos tubos, o que resultaria em seu rompimento.

A norma ABNT NBR 5626 recomenda que, em condições dinâmicas, ou seja, com escoamento, a pressão não deve ser inferior a 1 mca em qualquer peça, sendo a única exceção, a caixa de descarga acoplada, cuja pressão dinâmica poderá ser de até

0,5 mca, e a válvula de descarga, que deverá ser no mínimo de 1,5 mca [14].

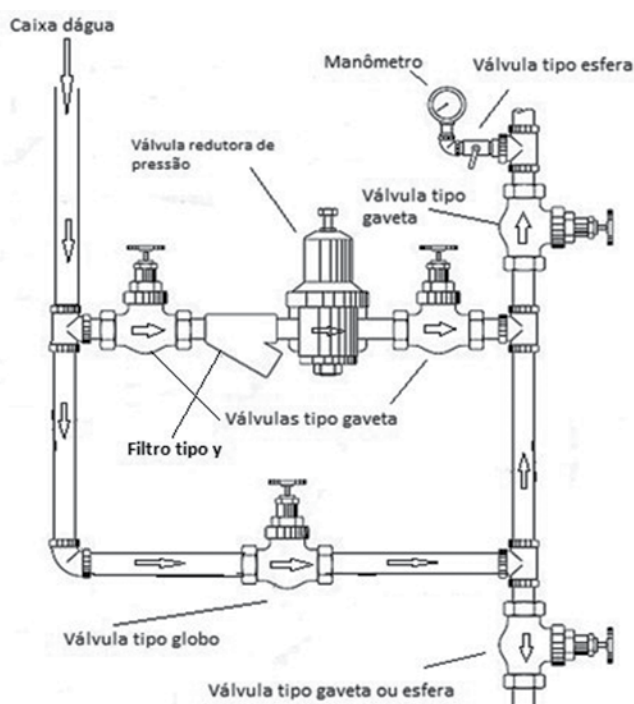


Fig. 1 – Sistema de válvula para redução da pressão. Fonte: [15]

Pode-se notar que, em um edifício, os andares mais próximos da cobertura terão pressões no limite do preconizado por norma, sendo normalmente esse o foco do dimensionamento. Já nos inferiores, a pressão será alta, aumentando o consumo.

O controle da pressão representa diminuição no consumo de água, da frequência de ruptura de tubulações e minimização dos eventuais vazamentos. Estima-se que uma redução de pressão de 30mca para 17mca pode resultar em uma economia de até 30% do consumo de água [16].

O método mais comum de reduzir a pressão é por meio do emprego de uma estação redutora de pressão do tipo mostrado na **figura 1**, devido a sua facilidade de manutenção e eficiência do processo. Já existem válvulas eletrônicas que podem permitir uma gestão mais eficiente de acordo com o nível do reservatório superior.

3.3 Medição individualizada

A prática de individualizar as medições resulta

numa redução de até 22% no volume de água consumido [17]. Porém, muitas vezes é necessário setorizar os edifícios para realizar as leituras, sejam de forma visual ou eletrônica. Algumas dessas setorizações podem ser simples ou complexas. Na maioria das vezes, essa setorização dependerá do número de colunas de distribuição que o edifício possuir [18].

Um fator relevante é relativo à classe alta, que, em geral, é a que menos reduz o consumo da água, comprovando a tese de que o usuário com padrão de vida elevado pode, inconscientemente, ignorar as ações e metodologias de economia de água, ao contrário daquele que possui menor poder aquisitivo, que, na sua grande maioria, implementa ao máximo ações e hábitos que reduzam gastos.

Há diversas formas de medir o volume de água utilizado em um setor. A **figura 2** exemplifica alguns casos, tais como: a medição coletiva, a medição no hall (andar), a medição no barrilete e a no andar térreo (uma das mais difíceis de implantar devido as altas perdas de carga) [19].

No caso de sistemas de medição por telemetria, são instalados hidrômetros digitais nos locais desejados. Porém, sua medição será realizada por um computador ou painel eletrônico, ficando fixo num local desejado na edificação. Por questões de controle e segurança, este local costuma ser a portaria.

Buscando a maior conservação de água, é recomendado a aplicação de medições individuais de forma a controlar melhor o uso da água e a detecção de vazamentos [18]. Nestes casos, é possível identificar o apartamento em que ocorre o vazamento, facilitando a manutenção.

Esta pesquisa considera que, com o advento das tecnologias de telemetria, não há razão para que um sistema de medição individualizada não seja implantado em edificações novas. Em virtude disso, essa opção de conservação de água não será vista como opção e sim obrigação construtiva para novos empreendimentos, em termos técnicos-econômicos. No Distrito Federal, a Lei 3.557, de 2005, dispõe sobre a individualização de instalação de hidrômetro nas edificações verticais residenciais e nas de uso misto e

nos condomínios residenciais do Distrito Federal.

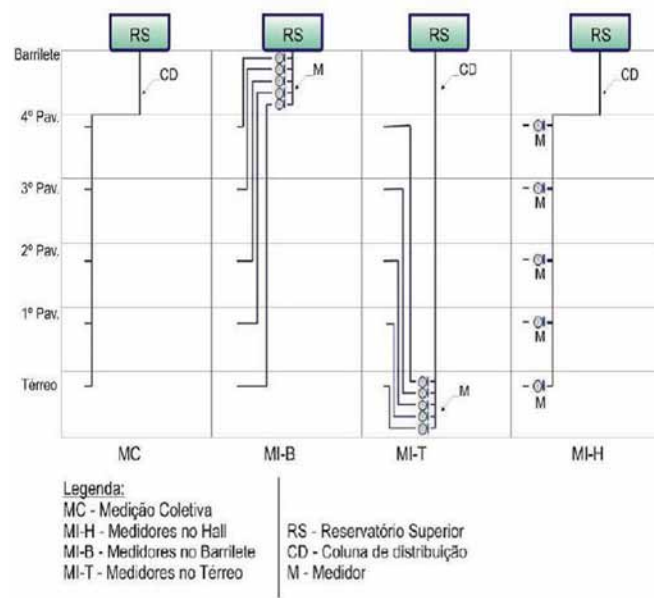


Fig. 2 – Formas de medição individualizada. Fonte: [19]

Nesta Lei, é normatizado que, para a aprovação dos projetos de edifício novos, ou seja, a serem construídos, será obrigatória a instalação de hidrômetros individuais para cada uma das unidades.

Para as edificações existentes, foi conferido um prazo até 2015 para adequação à Lei. Contudo, em 2008, a Lei foi alterada e permitiu a não individualização para os casos de inviabilidade técnica e econômica. Essa exceção foi assim definida pela mesma Lei: “considera-se inviável a instalação de hidrômetro individual, do ponto de vista técnico, quando as condições estruturais do prédio não a permitam e, do ponto de vista econômico, quando resulte, por qualquer dos modelos acreditados pela concessionária, em custo econômico-financeiro desproporcional aos benefícios que dela se esperam”.

Utilizando da exceção da Lei, da idade avançada das edificações de Brasília, da quantidade de prumadas de água que os prédios possuem e do custo de uma obra em prédios que datam de 1950, a maioria dos edifícios não aderiram a medição individualizada.

A Lei 13.312, de julho de 2016, alterou as diretrizes de saneamento básico nacional, estabelecendo que as

novas edificações condominiais deverão ter medição individualizada. Contudo, as edificações existentes não foram tratadas, ficando sem o benefício da tecnologia.

3.4 Economizadores de água

Os economizadores de água são, em geral, aparelhos que reduzem a vazão ou limitam o tempo de acionamento do dispositivo com fechamento automático dele. Eles podem ser facilmente instalados em edificações novas, contudo, em edificações existentes e ocupadas, o procedimento pode ser complexo e oneroso. Alguns equipamentos para instalações hidráulico-sanitárias mais modernos já apresentam essa redução. Exemplo típico são as bacias sanitárias que utilizavam 12 litros por descarga e hoje têm duplo acionamento, consumindo no máximo 6 litros. Mais moderna ainda é a bacia sanitária a vácuo com gasto de apenas 1 litro de água.

A economia que cada tipo de dispositivo pode oferecer está presente na **tabela 1**.

Tab. 1 – Economia gerada pelos dispositivos

Local	Aparelhos Indicados	Economia estimada (%)	Tempo de retorno (meses)
Chuveiro	Registro regulador de vazão	20	6
	Válvula de fechamento automática	20	12
Lavatório	Registro regulador de vazão	20	5
	Arejador para bica ou torneira	20	5
	Torneira automática	25	8
	Torneira eletrônica	35	12
Bacia sanitária	Bacia para 6 litros	50	6
	Caixa de descarga duplo acionamento	50	8
	Válvula de descarga duplo acionamento	20	8
Pia de cozinha	Arejador para bica ou torneira	20	5
	Registro regulador de vazão	20	5

Fonte: Grisolia [20]

3.5 Usos finais de aparelhos

Existem poucos estudos nacionais apontando o perfil de consumo dos diversos aparelhos em residências. A maioria dos estudos referem-se a edificações com interesse social, que possui suas peculiaridades de consumo reduzido em função da renda [21-23]. Há ainda, estudos referentes a

escritórios, universidades e edificações comerciais que fogem ao escopo desse trabalho. Uma pesquisa mais fidedigna foi realizada no Plano Piloto e monitorou dezenove apartamentos por sete dias por meio do uso de cronômetros, questionários e dados dos equipamentos. A tipologia considerada foi de blocos com predominância de 4 ou 6 pavimentos, com área média de 91m² e 3 moradores. Foi considerada área de garagem (no subsolo) de 1.090m² e jardins no entorno com 1.000m². Nesse levantamento foi obtida a média de consumo 172 L/hab/dia, consumo de uso comum de 3,5%, e para os equipamentos as médias mostradas na **figura 3** [24].

Alguns aspectos são importantes durante o estudo do perfil de consumo e devem ser considerados pois promovem resultados distintos. Em geral, hábitos, cultura regional, localização geográfica, tradições, número de moradores, tempo de permanência, clima, renda e valor da tarifa de água costumam influenciar tais resultados [25].

A **tabela 2** aponta os dados de uma pesquisa para os consumos referentes aos usos nas áreas comuns de uma edificação. Durante a estação chuvosa, a irrigação restringe-se apenas a vasos com plantas ornamentais com 0,002 L/m²/dia [24].

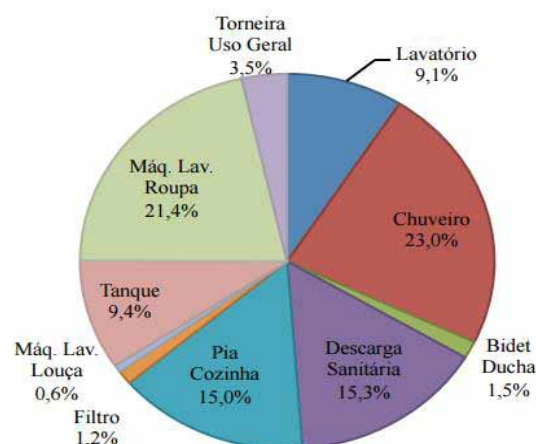


Fig. 3 – Uso em edificações multifamiliares. Fonte: [24]

A análise desse gráfico mostra que os chuveiros e as máquinas de lavar roupa são os equipamentos com maior parcela de uso. O primeiro tem seu consumo diretamente relacionado ao modo como as pessoas

utilizam o equipamento (uma questão cultural), tal como duração do banho, fechamento durante o processo de ensaboar etc. Por outro lado, equipamentos de redução de vazão instalados no chuveiro colaboram significativamente com essa redução.

Tab. 2 – Consumo da área comum

Utilização	Consumo (%)	Consumo* (L/m²/dia)
Irrigação de Jardins	1,05	3
Limpeza Condominial	3,15	4

Fonte: Chain *et al.* [26] e Sella [27]*

Em segundo lugar, tem-se as máquinas de lavar roupa, devido existir uma questão cultural relacionada à utilização da máquina com carga plena de roupas a fim de otimizar o uso da água e a outra relacionada a eficiência do equipamento, intrínseco do fabricante, principalmente na relação volume de água gasto versus quantidade de roupa lavada.

4. Metodologia

A tipologia padrão do Plano Piloto já foi estudada em outros artigos, sendo utilizada nesse trabalho: média de quatro pessoas por unidade, consumo de 151 L/hab/dia e blocos com 48 apartamentos [24, 28].

A metodologia de cálculo consiste em calcular a fatura de água com a economia propiciada pelo equipamento em questão. O lucro obtido da diferença das faturas antigas e atuais será computado mensal, assim como o custo de manutenção e o valor de implantação. Ao final será calculado o tempo de retorno (VPL=0), considerando taxa de juros da SELIC, referente a 1,1% ao mês.

4.1. Implantação e economizadores

Cada aparelho de cada apartamento foi analisado e teve considerada a implantação de um dispositivo economizador acoplado. O custo dos dispositivos na edificação foi considerado como a soma dos custos em um apartamento multiplicado pelo número de apartamentos, conforme **tabela 3**.

Tab. 3 – Custo dos economizadores

Nome	Valor (R\$)	Qtd por apartamento	Preço por apartamento (R\$)
Válvula de descarga DUO	299,90	3,00	899,70
Restritor de vazão para chuveiros	14,00	3,00	42,00
Arejador fixo para torneira de lavatório	12,40	3,00	37,20
Arejador articulável para torneira de cozinha	32,90	1,00	32,90
Subtotal			1.011,80
Qtd aptos			48,00
Total			48.566,40

Não foi encontrada nenhuma fonte sobre custos de manutenção. Com isso, foi adotado que os dispositivos economizadores têm sua manutenção condicionada à substituição de peças complementares, tais como reparos, carrapetas etc. Os arejadores e restritores não necessitam de manutenção [29]. Com isso, foi realizada pesquisa de preços do material e estimou-se a sua frequência, tendo como referência a experiência de diversos engenheiros da área de manutenção. Tais dados foram compilados na **tabela 4**. Os custos referentes à mão de obra foram suprimidos devido à simplicidade da troca das peças em questão.

Tab. 4 – Custo da manutenção

Nome	Valor (R\$)	Frequência
Reparo válvula de descarga	35,90	1 a cada 5 anos
Subtotal		R\$ 7,18
Quantidade de apartamentos		48
Total		R\$ 344,64
Custo total da manutenção		R\$ 28,72

4.2. Implantação de medição individual

A economia média de consumo esperada é de 22%. O custo dependerá do número de prumadas existentes no apartamento, tendo sido adotado o sistema de rádio frequência para diminuir a extensão das obras civis.

Estimou-se para este trabalho um custo de R\$ 500,00, incluindo as obras civis, para cada prumada

de alimentação existente em um apartamento – os apartamentos atuais possuem uma única prumada, mas os antigos chegam a ter cinco. A manutenção mensal foi cotada em R\$ 4,00 por apartamento. Este custo teve origem em pesquisa de preço com a empresa CONSERGEL (março de 2015) e na Internet com sites de orientação aos síndicos [30] e reportagens [31].

A vida útil de um hidrômetro por telemetria é de 8 anos, enquanto o equipamento de RF (leitor de rádio frequência) é de 15 anos [32], valores que foram adotados neste trabalho.

Para as construções novas, que não necessitam de telemetria, poderiam ser utilizados os hidrômetros convencionais. Nesse caso, diversas concessionárias cobram uma taxa mensal referente à substituição ou manutenção do hidrômetro. Portanto, o custo de manutenção dos hidrômetros seria estimado pelo Decreto 4.442/2009 que estabelece a taxa de manutenção para os hidrômetros em Olímpia/SP. Esse valor é de R\$/m³ 0,056 e foi aplicado ao volume de água estimado a ser gasto, incluindo as economias dos métodos.

Contudo, a fim de modernizar os edifícios, facilitar o controle de vazamentos, obter dados históricos de modo rápido e diminuir o acesso de funcionários das companhias nos andares, adotou-se nessa pesquisa, em ambos os estudos de casos, os hidrômetros por telemetria.

4.3. Implantação de sistema de válvulas redutoras de pressão

Conforme descrição do item 3.1, o sistema de válvula é composto por 2 manômetros, 6 registros, 2 filtros e 2 válvulas redutoras. Em pesquisa de mercado, realizada em março de 2016, constatou-se que esse sistema tem um custo em média de R\$5.160,00 por sistema, valor que foi utilizado neste trabalho.

Não foi verificado material disponível sobre custos de manutenção com esse dispositivo. Como a válvula trata-se de dispositivo equivalente a um registro, ela enquadra-se na previsão de vida útil mínima de 20 anos [33]. Com isso, estimou-se que haverá troca dos filtros presentes no sistema a cada 3 anos e realizou-se a pesquisa de preço do mesmo, totalizando R\$ 62,80 na loja EMBRAR, cotado em dezembro de 2016. Assim, tem-se que o custo

mensal é de R\$ 3,48 para cada prumada.

5. Resultados

Este capítulo apresenta os resultados e discussões obtidos com a metodologia adotada no trabalho. Os resultados são apresentados de acordo com o sequenciamento das etapas propostas na metodologia.

5.1. Edificação existente

As análises das medidas de redução, através da viabilidade econômica, geraram sete opções com seus respectivos períodos de retorno, conforme é mostrado na **tabela 5**. Algumas medidas não tiveram um tempo de retorno passível de ser determinado matematicamente, pois não demonstraram sua viabilidade econômica (receitas inferiores às despesas) e com isso foram eliminadas. Tal fato ocorreu devido ao baixo fluxo de caixa mensal.

Nos casos em que isto ocorreu, chamou-se o tempo de retorno de indeterminado.

Tab. 5 – Medidas de redução de consumo de água

Opção	Tempo de Retorno (meses)	Custo (R\$)
Economizador	50	49.578,20
Medição individual + economizador	Indeterminado	169.578,20
Redutor de pressão	Indeterminado	206.400,00
Economizador + Redutor de pressão	Indeterminado	255.978,20
Economizador + Redutor de pressão + Medição individual	Indeterminado	375.978,20
Medição individual	Indeterminado	120.000,00
Medição individual + redutor de pressão	Indeterminado	326.400,00

5.2. Edificação a ser construída

Na edificação a ser construída, os resultados estão na **tabela 6**. Algumas medidas não tiveram um tempo de retorno passível de ser determinado matematicamente

pois não demonstraram sua viabilidade econômica (receitas inferiores às despesas) e com isso foram eliminadas. Tal fato ocorreu devido ao baixo fluxo de caixa mensal.

Tab. 6 – Medidas de redução de consumo de água

Opção	Tempo de retorno (meses)	Custo (R\$)
Medição individual	19	24.000,00
Redutor de pressão	24	41.280,00
Medição individual + redutor de pressão	26	65.280,00
Economizador + Redutor de pressão + Medição individual	34	114.858,20
Medição individual + economizador	38	73.578,20
Economizador + Redutor de pressão	40	90.858,00
Economizador	50	49.578,20

Numa construção nova, os economizadores deveriam ser analisados sob um aspecto singular. Afinal, em um edifício a ser reformado, já existe um aparelho que precisou ser substituído, onerando a implantação dos sistemas. Na construção de um edifício, todos os aparelhos serão adquiridos. Esse fato leva à possibilidade da aquisição de aparelhos que atendam ao padrão de economicidade hídrica. Considerando a competição entre fabricantes, existe uma tendência de que os novos aparelhos já sejam econômicos, não havendo sobrecurso na obra. Portanto, este trabalho considerou que os custos devidos a estes aparelhos foram nulos para o caso da construção.

Outro aspecto cabe a exigência legal da implantação de medição individual em construções novas. Da mesma forma, esse custo também não foi computado. Com isso, foi possível reescrever a **tabela 6** produzindo a **tabela 7** com as alterações comentadas.

Tab. 7 – Medidas de redução de consumo de água

Opção	Tempo de retorno (meses)	Custo (R\$)
Economizador	0	0
Medição individual + economizador	0	0
Medição individual	0	0
Economizador + Redutor de pressão + Medição individual	11	41.280
Economizador + Redutor de pressão	16	41.280
Medição individual + redutor de pressão	16	41.280
Redutor de pressão	24	41.280

6. Conclusão

Propôs-se nesse trabalho analisar as alternativas para redução do consumo de água em edificações residenciais no Plano Piloto, em Brasília/DF.

Utilizou-se uma tipologia padrão já estudada e apresentada. Isso apenas foi possível em virtude da padronização da urbanização do Plano Piloto proposto por Lúcio Costa. Entretanto, torna-se necessário ressaltar que tal modelagem não pode ser expandida a outras localidades, devendo haver um estudo específico para qualquer outra tipologia de edificação, antes de uma generalização de resultados.

A estimativa de consumo existente na bibliografia é muito variável, dependendo da localidade, da classe econômica, dos tipos de equipamentos (novos ou antigos), do hábito de uso da água, e da tipologia dos prédios. Portanto, foi utilizado o valor experimental. O refinamento desse valor através da microanálise em regiões administrativas ratifica a melhor escolha.

Diante do resultado econômico mostrado na **tabela 5**, afirma-se que a melhor opção é a utilização de economizadores. Tal fato acontece devido à idade dos prédios, com tubulações antigas, em sua maioria de ferro galvanizado para água fria e tubos de ferro fundido para esgoto e água pluvial. Algumas têm seu trajeto através de pilares, dificultando sua substituição e diversas prumadas para alimentação de um mesmo apartamento.

Para as edificações a construir, a **tabela 7** demonstra que a opção de utilizar a medição individualizada em conjunto com economizadores como forma de

redução de consumo possui o melhor desempenho. O acréscimo dos redutores de pressão, além de aumentar o tempo de retorno em poucos meses, proporciona o aumento da vida útil da instalação devido ao trabalho com menores pressões. Assim, além de utilizar as três medidas por questões legais, conforme item 3, demonstrou-se a viabilidade delas numa construção.

O uso de economizadores foi destaque positivo em ambos os casos, mostrando sua robustez e

performance. Tal fato ratifica os princípios de conservação de água propostos há quase 20 anos [11] e a ratificação da “regra dos três R”, onde o primeiro passo é a redução.

Do exposto nesse trabalho considera-se viável, rentável e factível que toda unidade habitacional inicie a substituição de seus aparelhos por outros de melhor desempenho (economizadores). Essa ação irá produzir vantagens ambientais e financeiras aos usuários.

Referências bibliográficas

- [1] WUCB, A water conservation guide for commercial, institutional and industrial users. New Mexico: Ed. Water Use and Conservation Bureau, 1999.
- [2] Atlas Brasil, Abastecimento urbano de água: panorama nacional. Brasília. 2010. Agência Nacional de Águas; Engecorps/Cobrape.
- [3] Instituto Brasileiro de Geografia e Estatística, Censo Demográfico 2000. Disponível em: < <http://www.ibge.gov.br>>. Acesso em: 10 jan. 2016.
- [4] R. F. Gonçalves, (Org.), Uso racional de água e energia. Conservação de água e energia em sistemas prediais e públicos de abastecimento de água. Prosab-Finep Edital 5. Rio de Janeiro: Abes, 2009. 354 p
- [5] Distrito Federal, Auditoria Operacional na Gestão dos Recursos Hídricos do Distrito Federal: relatório (versão simplificada). Tribunal de Contas. Conselheiro-Relator Manoel de Andrade, 2010. Brasília.
- [6] S. Araújo, Novos sistemas de captação de água vão aliviar reservatórios do DF em épocas de seca. Agência Brasília/DF. Brasília, 2016 Disponível em: <<http://www.agenciabrasilia.df.gov.br/2016/09/25/novos-sistemas-de-captacao-de-agua-vao-aliviar-reservatorios-do-df-em-epocas-de-seca/>>. Acesso em :25/11/2016
- [7] A. C. Rebouças; B. Braga; J. G. Tundisi, (Org.), Águas doces no Brasil. 2ª.ed. São Paulo: Escrituras Editora, 2002. 703p
- [8] G. B. Beekman, G. B., Gerenciamento Integrado dos Recursos Hídricos, IICA. Brasília, 1999. 64 p.
- [9] L. O. A. B Silva; M.A.A. Souza; N. J. Allam, Uma proposta de reúso de água em condomínios verticais em Brasília – DF. São Paulo, 2004. I Conferência Latino-Americana de construção sustentável x encontro nacional de tecnologia do ambiente construído.
- [10] Companhia De Saneamento Ambiental Do Distrito Federal (CAESB), A água, o cidadão e a CAESB – Em busca do uso sustentável da água no DF. Brasília, 2010
- [11] A. C. Coelho, Manual de Economia de Água – Conservação de Água. Olinda: Comunigraf Editora, 2001
- [12] O. M. Gonçalves; M. S. O. Ilha; S. Amorim, Indicadores de uso racional de água em escolas públicas de ensino fundamental e médio. São Paulo, 2006. Relatório de pesquisa desenvolvida dentro do edital CT-HIDRO do CNPq.
- [13] P. B. Cheung, Consumo de água. In: R. F. Gonçalves, (Coord.). Uso racional da água e energia: conservação de água e energia em sistemas prediais e públicos de abastecimento de água. Rio de Janeiro: ABES, 2009. p. 36-98.
- [14] Associação Brasileira De Normas Técnicas (ABNT). NBR 5.626: Instalação predial de água fria. Rio de Janeiro, 1998.
- [15] Sindicato Da Construção De São Paulo (SIDUSCONSP). Conservação e reúso de água em edificações. São Paulo: Prol Editora Gráfica, 2005.
- [16] S. R. S. Silva, Avaliação do sistema de medição individualizada de água em prédios populares situados na cidade de Salvador-Bahia. Salvador: UFBA, 2010. Dissertação (Mestrado Profissional em Gerenciamento e Tecnologia Ambiental no Processo Produtivo). Disponível em: <<https://repositorio.ufba.br/ri/handle/ri/9585>>. Acesso em: out. 2015.
- [17] H. O. Tamaki, A medição setorizada como instrumento de gestão da demanda de água em sistemas prediais –

- estudo de caso: Programa de Uso Racional da Água da Universidade de São Paulo São Paulo: USP. Dissertação de Mestrado, Escola Politécnica.
- [18] M. Lima, Gestão de água em edifício público administrativos: recomendações baseadas no estudo de caso no TCU. Brasília, 2015. Dissertação de Mestrado em Arquitetura e Urbanismo. Programa de Pós-Graduação da Faculdade de Brasília -UnB. 192p
- [19] P. Z. Grisolia, Gestão da demanda para uso racional da água (URA). São Paulo, 2015. Manual de sustentabilidade Condominial -SECOVISP.
- [20] A. L. Rocha; D. Barreto; E. Ioshimoto, Caracterização e monitoramento do consumo predial de água. Brasília: MPOG, 1998. Documento Técnico de Apoio nº E1, PNCD A
- [21] L. H. Oliveira, Metodologia para a implantação de programa de uso racional da água em edifícios. São Paulo: USP, 1999. Tese de Doutorado, Escola Politécnica da Universidade de São Paulo
- [22] D. B. Barreto, Perfil do consumo residencial e usos finais da água Ambiente Construído. Porto Alegre, 2008. v. 8, n. 2, abr./jun. Associação Nacional de Tecnologia do Ambiente Construído. p. 23-40.
- [23] D. Sant´ana; L. Boeger; L. Vilela, Aproveitamento de águas pluviais e o reúso de águas cinzas em edifícios residenciais de Brasília – Parte 2: viabilidade técnica e econômica. Brasília: Paranoá, Brasília, nº 10, 2013. p. 85-93.
- [24] A. V. Hafner, Conservação e reúso de água em edificações – experiências nacionais e internacionais. Rio de Janeiro: UFRJ, 2007. Dissertação de Mestrado em Engenharia Civil.
- [25] R, R, Chain; C. A. M. Figueira Netto; E. Messuti, Sistema de Reaproveitamento de Água para Edificações. Rio de Janeiro, 1999. Trabalho V-053. Anais do 20º Congresso Brasileiro de Engenharia Sanitária e Ambiental.
- [26] M. B. Sella, Reúso de águas cinzas: avaliação da viabilidade da implantação do sistema em residências. Porto Alegre: UFRS, 2011. Trabalho de Graduação em Engenharia Civil, Departamento de Engenharia Civil. 72p.
- [27] J. T. D. S. Vianna, Análise de alternativas para conservação de água em edificações residenciais funcionais em Brasília-DF. Brasília, 2017. Dissertação. UnB. 128 p
- [28] DOCOL, Instalação de Produtos Economizadores de Água. Web curso, 2015. Disponível em [http://www.clubedocol.com.br/media/03_apostila_2015_Mar%C3%A7o .pdf](http://www.clubedocol.com.br/media/03_apostila_2015_Mar%C3%A7o.pdf). Acesso em Jan-2017
- [29] SINDICONET, Custos e gestão da medição individualizada. Disponível em: <<https://www.sindiconet.com.br/informese/7229/individualizacao-de-hidrometros/custos-e-gestao-da-medicao-individualizada>>. Acesso em: 15 dez. 2015.
- [30] I. Leite, Como economizar água com instalação de hidrômetros individuais nos condomínios. Disponível em: <<http://g1.globo.com/sao-paulo/blog/como-economizar-agua/post/como-economizar-agua-com-instalacao-de-hidrometros-individuais-no-condominios.html>>. Acesso: em 10 Ago. 2015.
- [31] TECHMETRIA, Sistema de medição individual. Perguntas e Respostas frequentes. Disponível em: <<http://www.techmetria.com.br/site/faq.php>>. Acesso em: 22 ago. 2016.
- [32] Associação Brasileira De Normas Técnicas (ABNT), NBR 15.575: Desempenho de edificações habitacionais– Requisitos. Rio de Janeiro, 2013.

Determinação do tirante mínimo da hidrovia do Tocantins e Araguaia

Marcelo de Miranda Reis, Cassia Vanessa de Castro*, José Carlos Cesar Amorim
 Instituto Militar de Engenharia (IME)
 Praça General Tibúrcio, 80, CEP:22290-270,
 Praia Vermelha, Rio de Janeiro/RJ, Brasil,
 *ccassiav@gmail.com

RESUMO: O presente trabalho teve como objetivo calcular o tirante de escoamento com permanência de 90% de nível d'água mínimo anual, com período de recorrência de 10 anos para as estações fluviométricas existentes na Hidrovia do Tocantins e Araguaia. Para a análise dos dados utilizou-se os métodos estatísticos de Gumbel para mínimos e Weibull. Esses estudos são importantes para avaliar as condições de navegabilidade da hidrovia tanto em trechos em que a navegação já ocorre livremente durante a maior parte do ano, como em locais onde necessitam de obra de dragagens. Para o cálculo dos volumes de dragagem é necessário conhecer o tirante mínimo com permanência de 90% para TR de 10 anos. Os resultados mostram que a distribuição de Weibull há uma tendência de ocorrerem níveis d'água menores comparados com os resultados da distribuição de Gumbel para mínimos para as séries dos postos fluviométricos da hidrovia do Tocantins e Araguaia.

PALAVRAS-CHAVE: Curva de permanência. Cotas mínimas. Gumbel. Weibull. Hidrovia.

ABSTRACT: A calculation of the depth of fluid flow in the fluviometrics stations located in the Tocantins and Araguaia Waterway is presented with 90% of the minimum annual water level in a recurrence time of 10 years. For the data analysis the Gumbel and the Weibull statistics methods were used. This study is important to determine the navigability conditions of the waterway in locations that are navigable through most of the year and in parts that need dredging works. To proper calculate the dredging volumes it is necessary to know the minimum depth of flow with 90% of the minimum annual water level in a recurrence time of 10 years. The results obtained with the Weibull distribution show a tendency for smaller water levels when compared with the results obtained with the Gumbel distribution.

KEYWORDS: Permanence curve. Minimum depth. Gumbel. Weibull. Waterway

1. Introdução

A Hidrovia Tocantins e Araguaia apresenta longos trechos navegáveis e posição geográfica privilegiada com grande potencial para o escoamento da produção de grãos e minérios. Por esse motivo foi objeto de estudo do Plano Estratégico de Recursos Hídricos da Bacia Hidrográfica dos rios Tocantins e Araguaia [1]. Também fez parte do projeto de integração nacional do corredor de transporte multimodal eixo Norte-Sul desenvolvido pela Confederação Nacional de Transporte [2]. O plano é ligar a Região Norte à Região Sul alternando os modos de transporte entre aquaviário e rodoviário por meio das hidrovias dos rios Tocantins – Araguaia e Paraná e das rodovias BR-153, BR-472, BR-258, BR-158 e BR-163.

Para que seja possível realizar o transporte

pela hidrovia e atender a um calado que comporte embarcações tipo, deve-se conhecer a cota limite para a navegação, acima dela é possível navegar. A curva de permanência define a parcela do tempo amostrado que poderá existir navegação [3]. Para esse estudo foi definido a permanência do tirante mínimo em 90% do tempo para um tempo de retorno de 10 anos e a partir desse tirante pode-se estimar os volumes a serem dragados ou derrocados.

O objetivo desse trabalho é estabelecer uma metodologia de cálculo de tirante mínimo de escoamento com permanência de 90% de nível d'água anual, com período de recorrência de 10 anos (TR=10 - estudo de cotas mínimas) para estações fluviométricas existentes na calha de uma hidrovia aplicado ao Tocantins e Araguaia. O cálculo de tirante mínimo foi realizado utilizando a distribuição de Weibull e Gumbel para mínimas. Dentro dessa linha de pesquisa Gottschalk *et al.* [4] e Finkler *et al.*

[5] estudaram distribuições estatísticas aplicadas para vazões mínimas, Leme *et al.* [6] determinaram a vazão mínima anual de sete dias consecutivos com período de retorno de 10 anos para o médio rio Jaguarí Mirim e Kankam-Yeboah *et al.* [7] caracterizaram o regime de baixo fluxo da Bacia do Volta Preta usando vazões médias diárias por meio da curva de permanência. No entanto, não há muitos estudos no Brasil referente ao tirante mínimo para navegação. Observa-se que a maioria dos estudos são relacionados à disponibilidade hídrica [8], [9], [10].

Segundo DNIT [11], no período das cheias, é possível navegar até 2.000 km pelas águas do rio Tocantins. Neste percurso é utilizado uma eclusa existente na barragem da Usina Hidrelétrica de Tucuruí. No período de seca, a navegação fica comprometida a partir do lago de Tucuruí devido ao Pedral do Lourenço, um trecho de afloramentos rochosos que impedem a navegação comercial de grande porte entre o final do reservatório da hidrelétrica e a cidade de Marabá (PA).

O resultado oriundo da metodologia permitirá calcular os volumes de dragagem para viabilizar a utilização do rio como via navegável, tanto em trechos onde a navegação já ocorre livremente durante a maior parte do ano, como em locais onde necessitam de obras, ou seja, terá disponibilidade de calado em regime permanente para atender e transportar principalmente as cargas agrícolas provenientes das safras de grãos das regiões Norte e Centro-Oeste do país. Não faz parte dessa pesquisa o cálculo de volumes de dragagem.

2. Área de estudo

A Hidrovia do Tocantins e Araguaia tem aproximadamente 4.375 km de extensão e abrange os estados do Goiás, Tocantins, Pará, Maranhão, Mato Grosso e Distrito Federal [12].

A Bacia Hidrográfica do Tocantins/Araguaia tem 967.059 km² de área, o equivalente a 11% do território brasileiro, sendo Goiás (26,8%), Tocantins (34,2%), Pará (20,8%), Maranhão (3,8%), Mato Grosso (14,3%) e Distrito Federal (0,1%) [13].

As regiões abrangentes da Bacia do Tocantins estão compostas de: rio Araguaia, rio Paranã, rio das Almas, rio Maranhão, rio Tocantinzinho, rio Cana Brava, rio Dos Patos, rio Santa Clara e rio Uru.

- Os trechos estudados têm extensões de:
- Da foz do rio Tocantins até Peixe com 1.515km;
- Da foz do rio Araguaia até Barra do Garças com 1.716km;
- Ilha do Bananal o trecho do berço do Javaés com 427km;
- Do rio das Mortes de sua foz até Nova Xavantina com 567km;

Da foz do rio Itacaiúnas até a foz do rio Parauapeba com aproximadamente 150km [12].

A **figura 1** apresenta a visão geral da área de estudo.

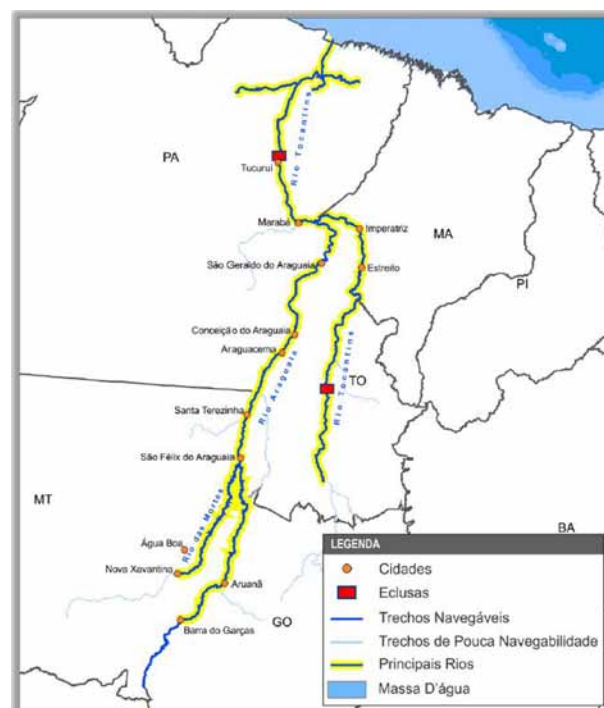


Fig. 1: Visão geral da área de estudo Fonte: ANTAQ [14]

3. Materiais e métodos

Buscou-se pesquisar os dados hidrológicos de fontes confiáveis e utilizar diretrizes empíricas fundamentadas. Desta forma, foi realizada uma consulta no mapa da Rede Hidrometeorológica e no Inventário de Estações Fluviométricas existente no sítio eletrônico da (ANA) [15].

Primeiro, detectou-se todas as estações fluviométricas existentes nos segmentos de rios da hidrovia do Tocantins e Araguaia. As **tabelas 1 a 5** apresentam as estações existentes em cada trecho com o respectivo código da estação utilizado pela ANA e usado no estudo para a identificação de cada estação.

Tab. 1 – Rio Tocantins até Araguaia

Estação	Código Estação
Nazaré dos Patos	29750000
Tucuruí	29700000
Itupiranga	29200000
Marabá	29050000
Descarreto	23700000
Tocantinópolis	23600000
Tupiratis	23100000
Miracema do Tocantins	22500000
Peixe	22050001

Fonte: ANA [15] adaptado

Tab. 2 – Rio Araguaia – Barra das Garças

Estação	Código Estação
Araguatins	28850000
Xambioá	28300000
Conceição do Araguaia	27500000
Araguacema (Chácara Araguaia)	26320000
São Felix do Araguaia	26350000
Luiz Alves	25950000
Bandeirantes	25700000
Aruanã	25200000
Araguaiana	24850000
Barra das Garças	24700000

Fonte: ANA [15] adaptado

Tab. 3 – Rio Araguaia – Ilha do Bananal

Estação	Código Estação
Barreira da Cruz	26800000
Fazenda Dois Rios	26713000
Fazenda Coprape	26711000

Fonte: ANA [15] adaptado

Tab. 4 – Rio das Mortes – Nova Xavatina

Estação	Código Estação
Antônio do Leverguer	26300000
Trecho Médio	26200000
Xavatina	26100000

Fonte: ANA [15] adaptado

Tab. 5 – Rio Itacafunas – Parauapeba

Estação	Código Estação
Fazenda Alegria	9100000

Fonte: ANA [15] adaptado

Com as estações definidas conforme apresentadas nas **tabelas de 1 a 5**, foi possível coletar os dados de nível médio diário para todos os anos disponíveis nas séries históricas das estações. Teve-se o cuidado de obter somente os níveis d'água consistidos, os quais representam dados tratados pela ANA, dando mais confiabilidade e segurança quanto aos resultados obtidos nesse estudo.

As estações fluviométricas Fazenda Dois Rios (Código 26713000) e Fazenda Coprape (Código 26711000) do Trecho Ilha do Bananal não apresentaram níveis d'água médios diários no Sistema de Informações Hidrometeorológicas da Agência Nacional de Águas. Portanto, não foi possível realizar os estudos para essas estações.

3.1 Série anual de níveis d'água com permanência de 90%

Para cada ano com dados, utilizando os valores médios diários, foi traçada uma curva de permanência e se obteve o valor de nível d'água anual com permanência de 90% (H90%). Esse valor representa o nível existente no curso d'água em 90% do tempo, isto é, um nível pequeno. O risco assumido é de que há possibilidade de 10% do mesmo ser inferior a um valor estimado, e neste caso, haveria problemas com disponibilidade de calado para a hidrovia. Uma observação adicional pode ser feita. Quanto menor o intervalo de análise dos dados (dados diários, mensais ou anuais) mais segura será a interpretação da curva de permanência.

O percentual de 90% foi usado em virtude de ser uma referência utilizada em recursos hídricos e na legislação ambiental.

O procedimento para obtenção da curva de permanência foi baseado na obtenção de classes de frequência.

Para definição dos valores limites dos intervalos de observação, foram calculados o número de intervalos

ou classes no qual os dados serão separados. As equações 1 e 2 determinam os valores extremos para o número de classes. Estas são baseadas na quantidade de dados N da série analisada.

$$N^{\circ} \text{ mínimo de classes} = [10 + 1,33 \ln N] \quad (1)$$

$$N^{\circ} \text{ máximo de classes} = [10 \cdot \log_{10} N] \quad (2)$$

Onde N é o número de valores da série diária.

As curvas de permanência são construídas a partir dos dados com frequência simples, que são somadas cumulativamente no sentido crescente (curva de permanência de níveis d'água crescente) ou decrescente das grandezas (curva de permanência de níveis d'água decrescente). A curva crescente contínua é traçada pelos limites superiores dos intervalos e a curva decrescente contínua é traçada pelos limites inferiores.

Como exemplo a **figura 2** apresenta a curva de permanência de cotas médias diárias do ano de 1970 da série histórica da estação Fluviométrica de Xambioá (Código – 28300000). A partir da curva ou de forma mais precisa dos dados acumulados que geraram a curva observa-se que a cota ou nível d'água com permanência de 90% no ano de 1970 da estação Fluviométrica de Xambioá (Código – 28300000) foi de 179 cm.

Repetindo o procedimento de determinação nível d'água anual com permanência de 90%, pode-se obter para a estação fluviométrica uma série histórica anual de níveis d'água anual com permanência de 90%. A **tabela 6** apresenta como exemplo a série da estação Fluviométrica de Xambioá (Código – 28300000).

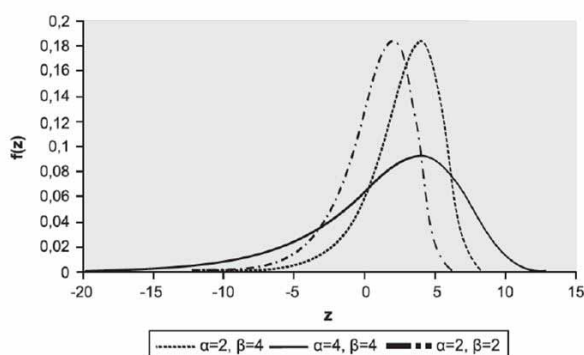


Fig. 2 – Curva de permanência de 1970 da série de cotas da estação de Xambioá (Código – 28300000).

Tab. 6 – Série de níveis anuais com permanência de 90% - Estação de Xambioá (Código – 28300000)

Ano	Nível com permanência de 90% (cm)	Ano	Nível com permanência de 90% (cm)	Ano	Nível com permanência de 90% (cm)
1970	179	1985	234	2000	229
1971	176	1986	220	2001	187
1972	175	1987	207	2002	201
1973	192	1988	218	2003	204
1974	240	1989	229	2004	215
1975	180	1990	224	2005	192
1976	196	1991	239	2006	210
1977	229	1992	232	2007	169
1978	236	1993	223	2008	185
1979	256	1994	214	2009	223
1980	240	1995	223	2010	165
1981	223	1996	208	2011	184
1982	267	1997	252	2012	179
1983	236	1998	186	2013	195
1984	226	1999	192	-	-

3.2 Determinação do tirante com permanência de 90% para um TR de 10 anos

Para determinar o tirante com permanência de 90% na estação para um tempo de recorrência de 10 anos aplicou-se aos dados da série anual de tirantes ou níveis d'água com permanência de 90% os métodos estatísticos de Gumbel amostra para mínimos e de Weibull. Pois pertencem à mesma família de distribuição e são úteis para representar as distribuições mínimas anuais [16].

Os métodos estatísticos são os mais apropriados para determinação de valores extremos de nível d'água, entretanto sua eficácia depende da quantidade de dados disponíveis em uma série histórica que servirá como amostra do universo e da estabilidade das características principais do regime do curso d'água. Isto é, entre o momento da coleta e o momento da utilização dos dados históricos o rio não deve ter sofrido nenhuma modificação hidrológica importante como desvio, retificação, construção de barragem, urbanização das margens, etc. Além disso, deve-se trabalhar com séries históricas com dados confiáveis sem falhas. Desta forma, os anos com falhas foram excluídos das séries e optou-se por utilizar os dados consistidos disponibilizados pela ANA.

Em estudos de variáveis hidrológicas máximas

define-se Tempo de Recorrência ou Período de Retorno (TR) como o tempo médio em anos que evento é igualado ou superado pelo menos uma vez. Assim, o período de retorno representa o inverso da frequência (F) com que um evento (X, Xi) pode ser igualado ou superado, ou seja, reflete a probabilidade (P) com que uma dada variável hidrológica possa ser igualada ou superada, pelo menos uma vez, em um ano qualquer. Ao se ajustar uma distribuição de probabilidades aos dados de frequência de uma série histórica, utiliza-se a probabilidade de excedência para estimar um tempo de recorrência, que é obtido em anos. Este é expresso pela **equação 3**:

$$\frac{1}{F(X \geq Xi)} \cong \frac{1}{P(X \geq Xi)} \quad (3)$$

No entanto, quando o objeto de estudo consiste de uma série histórica de dados hidrológicos mínimos, o tempo de recorrência a ser estimado também está associado à probabilidade com que o valor mínimo considerado pode ser inferior ao esperado, ou seja:

$$\frac{1}{F(X \leq Xi)} \cong \frac{1}{P(X \leq Xi)} \quad (4)$$

Assim, ao se ajustar uma distribuição de probabilidades aos dados de frequência de uma série histórica, utiliza-se a probabilidade de não excedência para estimar um tempo de recorrência. No problema proposto está sendo buscado o nível d'água com permanência de 90% que não é igualado ou superado pelo menos uma vez a cada dez anos em média.

Quanto maior o número de dados da amostra utilizada para os cálculos, mais representativa esta amostra é do universo e melhor a inferência estatística realizada.

Em problemas de variáveis discretas, é possível responder à pergunta: qual a probabilidade de um número qualquer ser sorteado (evento X) dentro de um espaço amostral finito qualquer, constituído por N números, sendo este um evento aleatório. No entanto, na hidrologia, em grande parte das vezes, o que é buscado é avaliar qual a possibilidade de um determinado evento ser maior ou igual (ou menor ou igual) a um dado valor Xi e isto remete a necessidade de uso de uma variável contínua. Assim é preciso ajustar aos dados discretos

uma curva de distribuição de probabilidade contínua. Além disso, os dados da amostra nem sempre possuem uma frequência de ocorrência compatível com o tempo de recorrência de projeto.

Desta forma, a aplicação dos métodos estatísticos parte de uma amostra definida pela série de dados de nível d'água e supõe que esses dados se ajustem com uma precisão a uma curva de distribuições de probabilidades contínua. Como essa curva é contínua, se pode determinar valores extremos para diversos tempos de recorrência através das equações ou tabelas derivadas dessas equações que definem as curvas contínuas de distribuição de probabilidade. As curvas de distribuição de probabilidades mais utilizadas no cálculo de eventos hidrológicos extremos máximos são a Log Normal, a Log Pearson Tipo III e a assintótica dos extremos Tipo I ou de Gumbel (de máxima). O cálculo de tirantes para hidrovias e de vazões ecológicas são problemas hidrológicos de busca de valor mínimo e assim são usadas as curvas de distribuição de probabilidades de eventos hidrológicos extremos mínimos de Gumbel e de Weibull. Estas serão as curvas utilizadas nesse estudo.

Alguns conceitos estatísticos são utilizados nas metodologias de cálculo. Média (Xm) é a média aritmética do conjunto de dados expressa por:

$$Xm \cong \frac{\sum Xi}{n} \quad (5)$$

Onde: Xi = valor do evento i e n = número total de eventos.

A variância de uma variável aleatória ou processo estocástico é uma medida da sua dispersão estatística, indicando “o quão longe” em geral os seus valores se encontram do valor esperado. Dentro de uma amostra a variância med = -1,1396 é a dispersão em relação à média. Já o desvio padrão é definido como sendo a raiz quadrada da variância. A **equação 6** mostra a expressão de cálculo do desvio padrão (S).

$$S = -\frac{\sqrt{\sum (Xi - Xm)}}{n-1} \text{ ou } S = -\frac{\sqrt{\sum (Xi)^2 - n(Xm)^2}}{n-1} \quad (6)$$

O coeficiente de variação (CV) denota a relação entre o desvio padrão e a média.

$$CV = \frac{S}{\bar{X}_m} \quad (7)$$

A assimetria é um parâmetro importante na medida em que avalia a forma como os dados estão distribuídos em relação à média. Para que os dados apresentem distribuição normal, a assimetria deve ser próxima ou igual a zero. Nessa situação, a média, a moda e a mediana são iguais. Contudo, quando este valor for distante de zero, apresentará um padrão de distribuição com a maior quantidade de dados à esquerda (assimetria positiva) ou à direita (assimetria negativa).

O ajuste de uma distribuição de probabilidades é conduzido com base em 2 ou 3 parâmetros. A estimativa destes parâmetros é feita com base na Inferência Estatística a partir da amostra existente de dados. A partir deste conceito Ven Te Chow propôs uma equação geral que pode ser aplicada a todas as curvas de distribuição de probabilidades.

$$X_T = X_m + k_T S_x \quad (8)$$

Onde: X_T é a variável de interesse (no caso nível d'água) que se deseja calcular para um período de retorno TR; X_m é a média amostral; S_x é o desvio padrão amostral; K_T é um fator de frequência que depende do tempo de recorrência e eventualmente da assimetria e é tabelado por curva de distribuição de probabilidades.

3.3 A distribuição e o método de Gumbel (para mínimos)

Esse método supõe que a amostra definida pela série de dados de nível d'água se ajustam com uma precisão a curva de distribuições assintótica de extremos do Tipo I ou de Gumbel. A distribuição de valores extremos do Tipo I recebeu as seguintes outras denominações: distribuição de Gumbel, Fisher-Tippett tipo I e dupla exponencial.

Em 1928, Fisher e Tippett, tomando de vários conjuntos de muitas amostras o maior valor de cada conjunto, mostraram que a distribuição dos valores extremos é independente da distribuição original e se comporta como função limite. Gumbel, em 1945,

sugeriu que essa distribuição de valores extremos seria apropriada para a análise de frequência valores hidrológicos extremos, desde que a série fosse anual, isto é, cada nível da série de valores extremos fosse o nível extremo de uma amostra de 365 possibilidades (valor extremo do ano).

No caso de valores máximos, a função densidade da distribuição de Gumbel é dada por:

$$f(y) = \frac{1}{\alpha} \exp\left[-\frac{y-\beta}{\alpha} - \exp\left(-\frac{y-\beta}{\alpha}\right)\right] \quad (9)$$

Onde: α representa o parâmetro de escala e β o parâmetro de posição; de fato, β também é a moda de Y.

A distribuição de Gumbel (máximos) é a distribuição de valores extremos mais usada na análise de frequência de variáveis hidrológicas, com inúmeras aplicações na determinação de relações intensidade duração-frequência de precipitações intensas e estudos de vazões de enchentes.

Já no caso de valores mínimos, que é o objetivo deste estudo, a função densidade da distribuição de Gumbel é dada por:

$$f(z) = \frac{1}{\alpha} \exp\left[\frac{z-\beta}{\alpha} - \exp\left(\frac{z-\beta}{\alpha}\right)\right] \quad (10)$$

Apoiando-se no argumento de Fisher e Tippett, Gumbel sugeriu que a probabilidade de ocorrência de uma variável de magnitude menor ou igual a um dado valor x (probabilidade de não excedência) pode ser expressa por:

$$F(Z \geq z) = 1 - \exp\left[-\exp\left(\frac{z-\beta}{\alpha}\right)\right] \quad (11)$$

para $-\infty < Z < \infty, -\infty < \beta < \infty, e \alpha > 0$

Onde: α representa o parâmetro de escala e β o parâmetro de posição; de fato, β também é a moda de Z.

A média ou valor esperado, a variância e o coeficiente de assimetria de Z são, respectivamente:

$$Z_m = \beta - 0,05772\alpha \quad (12)$$

$$Var(Z) = \sigma^2 = \frac{\pi^2 \alpha^2}{6} \quad (13)$$

$$\gamma = -1,1396 \quad (14)$$

Observe, portanto, que a distribuição Gumbel (mínimos) possui um coeficiente de assimetria negativo e constante. A **figura 3** ilustra a função densidade Gumbel, para alguns valores específicos dos parâmetros α e β .

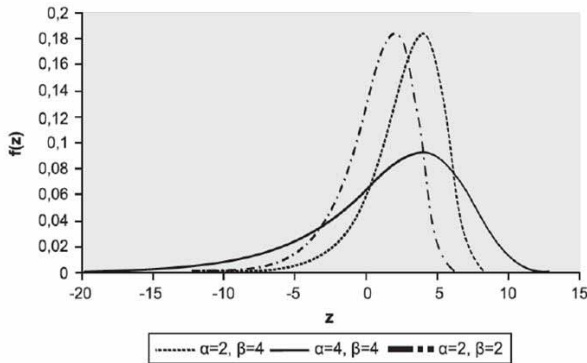


Fig. 3 – Função densidade Gumbel, para alguns valores específicos dos parâmetros α e β .

A determinação dos valores mínimos em função do tempo de recorrência é realizada através da expressão da inversa da FAP de Gumbel (mínimos), ou função de quantis, expressa por:

$$\begin{aligned} Z(F) &= \beta + \alpha \ln[-\ln(1-F)] \text{ ou} \\ Z(TR) &= \beta + \alpha \ln[-\ln(1 - \frac{1}{TR})] \end{aligned} \quad (15)$$

Onde F é a frequência e TR o tempo de recorrência.

O ajuste dos parâmetros α e β para os valores da amostra (**equações 12 e 13**), conduz a **equação 16**:

$$Z(TR) = Z_m + \sigma_Z [0,451 + 0,7797 \ln[-\ln(1 - \frac{1}{TR})]] \quad (16)$$

Desta forma, poder-se-ia aplicar a equação de Ven Te Chow (**equação 8**) com o valor do fator de frequência K_t dado por:

$$\begin{aligned} K_T &= \frac{\sqrt{6}}{\pi} [0,577 + \ln \ln(1 - \frac{1}{TR})] \text{ ou} \\ K_T &= 0,451 + 0,7797 + \ln \ln(1 - \frac{1}{TR}) \end{aligned} \quad (17)$$

Após calcular a média e o desvio padrão da série histórica de nível d'água com permanência de 90% para cada estação fluviométrica, e aplicar a **equação**

16 é possível determinar o tirante mínimo relacionado ao tempo de recorrência.

3.4 A distribuição e o método de Weibull (para mínimos)

Esse método supõe que a amostra definida pela série de dados de nível d'água se ajustam com uma precisão a curva de distribuições assintótica de extremos do Tipo III ou de Weibull para valores mínimos. A distribuição de Weibull refere-se à forma assintótica limite para um conjunto de N variáveis aleatórias originais $\{X_1, X_2, \dots, X_N\}$, independentes e igualmente distribuídas conforme um modelo $F_X(x)$ de cauda inferior limitada. A distribuição de extremos mínimos do Tipo III recebeu a denominação de distribuição de Weibull por ter sido usada pela primeira vez pelo engenheiro sueco Waloddi Weibull.

No caso de valores máximos, a função densidade da distribuição de Weibull é dada por:

$$f_Z(z) = \frac{\alpha}{\beta} \left(\frac{z}{\beta}\right)^{\alpha-1} \exp\left[-\left(\frac{z}{\beta}\right)^\alpha\right] \quad (18)$$

Onde: α representa o parâmetro de forma e β o parâmetro de escala. Para $\alpha = 1$, a distribuição de Weibull é a exponencial com parâmetro de escala β .

A função de probabilidades acumuladas da distribuição de Weibull é:

$$F(Z \geq z) = 1 - \exp\left[-\left(\frac{z}{\beta}\right)^\alpha\right] \quad (19)$$

para $Z \geq 0, \beta \geq 0$ e $\alpha > 0$

Onde: α representa o parâmetro de forma e β o parâmetro de escala.

A média ou valor esperado, a variância, o coeficiente de variação e o coeficiente de assimetria de Z são, respectivamente:

$$Z_m = \beta \text{ Gama}\left(1 + \frac{1}{\alpha}\right) \quad (20)$$

$$\text{Var}(Z) = \sigma_Z^2 = \beta^2 \left[\text{Gama}\left(1 + \frac{2}{\alpha}\right) - \text{Gama}^2\left(1 + \frac{1}{\alpha}\right) \right] \quad (21)$$

$$\gamma = \frac{Gama\left(1+\frac{3}{\alpha}\right) - 3Gama\left(1+\frac{2}{\alpha}\right)Gama\left(1+\frac{2}{\alpha}\right) + 2Gama^2\left(1+\frac{2}{\alpha}\right)}{\sqrt{[Gama\left(1+\frac{2}{\alpha}\right) - Gama^2\left(1+\frac{1}{\alpha}\right)]^3}} \quad (22)$$

$$CV = \frac{\sqrt{Gama\left(1+\frac{2}{\alpha}\right) - Gama^2\left(1+\frac{1}{\alpha}\right)}}{Gama\left(1+\frac{1}{\alpha}\right)} \quad (23)$$

Onde: $A(\alpha) = Gama(1+1/\alpha)$ e $B(\alpha) = Gama(1+2/\alpha)$. A **figura 4** ilustra a função densidade Weibull, para alguns valores específicos dos parâmetros α e β .

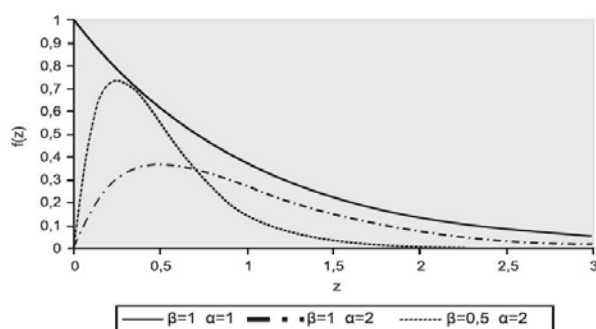


Fig. 4 – Função densidade Weibull, para alguns valores específicos dos parâmetros α e β .

Conhecidos os valores da média, da variância e do coeficiente de variação, os valores de α , β , $A(\alpha)$ e $B(\alpha)$ podem ser ajustados a amostra através da solução do sistema composto pelas **equações 20, 21 e 23**.

A determinação dos valores mínimos em função do tempo de recorrência é realizada através da expressão da inversa da FAP de Weibull (mínimos), ou função de quantis, expressa por:

$$z(F) = \beta[-\ln(1-F)]^{1/\alpha} \quad (24)$$

Ou

$$z(TR) = \beta[-\ln(1-1/TR)]^{1/\alpha}$$

onde F é a frequência e TR o tempo de recorrência.

4. Resultados e discussões

Conforme metodologia descrita, foram determinados os tirantes mínimos com permanência

de 90% para todas as estações fluviométricas com dados disponíveis para a hidrovia Tocantins e Araguaia.

A **figura 5** mostra o gráfico e a equação resultante do ajuste dos dados do tirante mínimo para a estação fluviométrica de Xambioá aos tempos de retorno, obtidos da série de 44 anos, pela distribuição de Gumbel para mínimos. Os resultados da **tabela 7** e **figura 6**, mostram que o tirante mínimo anual com período de retorno de 10 anos (Z_{10}) foi de 178,21 cm.

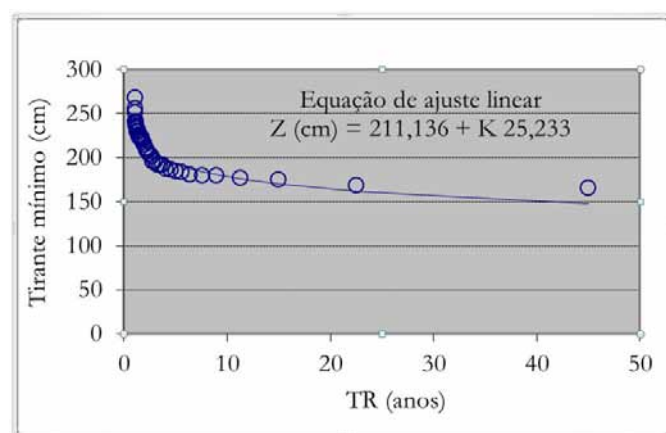


Fig. 5 – Tirante em função de TR - ajuste por Gumbel para mínimos para a estação fluviométrica Xambioá.

Tab. 7 – Previsão de tirante mínimo para estação fluviométrica de Xambioá para a distribuição de Gumbel

Parâmetros	Resultados			
TR (anos)	2	5	10	25
Z (cm)	215,28	192,98	178,21	159,56
P (%)	50	20	10	4

P = probabilidade de não-excedência do tirante mínimo (%).

A **figura 6** mostra o gráfico e a equação resultante do ajuste dos dados do tirante mínimo para a estação fluviométrica de Xambioá aos tempos de retorno, obtidos da série de 44 anos, pela distribuição de Weibull. Os resultados da **tabela 8** e **figura 6**, mostram que o tirante mínimo anual com período de retorno de 10 anos (Z_{10}) foi de 177,40 cm.

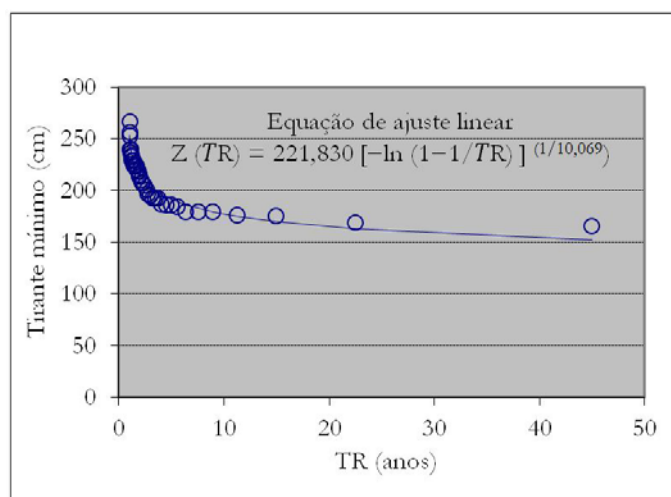


Fig. 6 – Tirante em função de TR - ajuste por Weibull para a estação fluviométrica Xambioá.

Tab. 8 – Previsão de tirante mínimo para estação fluviométrica de Xambioá para a distribuição de Wibull

Parâmetros	Resultados			
TR (anos)	2	5	10	25
Z (cm)	213,90	191,13	177,40	161,46
P (%)	2	5	10	25

P = probabilidade de não-excedência do tirante mínimo (%)

A seguir serão apresentados em tabelas os resultados de níveis mínimos de todas estações da Hidrovia do Tocantins e Araguaia comparando os métodos de cálculo entre Gumbel para mínimos e Weibull por segmento da hidrovia. **tabela 9 a 13.**

Tab. 9 – Tirante mínimos calculados para o segmento Rio Tocantins até Peixe

Estação	Resultados		Nível d'água adotado (cm)
	Método de Gumbel	Método de Weibull	
Nazaré dos Patos	187,56	186,60	186,60
Tucuruí	134,71	133,40	133,40
Itupiranga	165,37	167,46	165,37
Marabá	162,40	160,91	160,91
Descarreto	77,57	77,26	77,26
Tocantinópolis	26,57	26,91	26,57
Tupiratis	57,99	58,32	57,99
Miracema do Tocantins	44,01	43,70	43,70
Peixe	101,67	102,53	101,67

Tab. 10 – Tirante mínimos calculados para o segmento Rio Araguaia até Barra das Garças

Estação	Resultados		Nível d'água adotado (cm)
	Método de Gumbel p/ mínimos	Método de Weibull	
Araguatins	252,96	252,79	252,79
Xambioá	178,21	177,40	177,40
Conceição do Araguaia	201,03	200,46	200,46
Araguacema	229,83	211,24	211,24
São Félix do Araguaia	270,06	269,27	269,27
Luiz Alves	263,02	262,04	262,04
Bandeirantes	356,23	355,74	355,74
Aruanã	101,33	100,53	100,53
Araguaiana	94,87	94,23	94,23
Barra das Garças	34,94	36,99	34,94

Tab. 11 – Tirante mínimos calculados para o segmento Rio Das Mortes até Xavantina

Estação	Resultados		Nível d'água adotado (cm)
	Método de Gumbel p/ mínimos	Método de Weibull	
Santo Antônio do Levergers	178,30	177,46	177,46
Trecho Médio	238,15	237,87	237,87
Xavantina	38,83	38,49	38,49

Tab. 12 – Tirante mínimos calculados para o segmento Rio Itacaiunas - Foz Parauapeba

Estação	Resultados		Nível d'água adotado (cm)
	Método de Gumbel p/ mínimos	Método de Weibull	
Fazenda Alegria	252,72	250,77	250,77

Tab. 13 – Tirante mínimos calculados para o segmento Rio Araguaia - Ilha do Bananal

Estação	Resultados		Nível d'água adotado (cm)
	Método de Gumbel p/ mínimos	Método de Weibull	
Barreira da Cruz	43,80	44,27	43,80

As estações fluviométricas Fazenda Dois Rios (Código 26713000) e Fazenda Coprape (Código 26711000) do Trecho Ilha do bananal não apresentaram séries históricas de dados de níveis d'água médios diários no Sistema de Informações Hidrometeorológicas da ANA, para essas estações não foram possíveis fazer a análise do tirante mínimo.

De todas as estações estudadas, somente a estação fluviométrica Barreira da Cruz que apresentou registro de série histórica de 23 anos, enquanto que as outras estações possuíam registro variando de 32 a 56 anos, sendo a estação de Tocantinópolis a que apresentou maior número de eventos registrados.

De forma conservadora para os cálculos de níveis a serem utilizados na determinação dos volumes de dragagem, recomenda-se para estação o tirante com permanência de 90% para um TR de 10 anos adotar o menor dos valores calculados pelos dois métodos. Sendo que apenas 25% dos resultados o método de Gumbel apresentou cotas mínimas menores para a hidrovia do Tocantins e Araguaia.

5. Conclusão

A metodologia é perfeitamente aplicável com bons resultados. Como o número de eventos das estações fluviométricas variaram de 23 a 56 anos e quanto

maior o número de dados da amostra utilizada para os cálculos, mais representativa esta amostra é do universo e melhor a inferência estatística realizada, pode-se considerar que os resultados obtidos são representativos, e além disso os métodos de cálculo também se mostraram adequados para esse tipo de cálculo de tirantes mínimos.

Das vinte e quatro estações fluviométricas analisadas, somente seis foram considerados como nível d'água mínimo os resultados calculados pelo método de Gumbel amostra infinita.

Para as outras 18 estações o Método de Weibull apresentou resultados menores, e mesmo assim a diferença foi muito pequena variando em torno de 0,57% a 1,27%, praticamente desprezível, ou seja, os resultados mostram que a distribuição de Weibull e Gumbel apresentam probabilidade de ocorrerem níveis d'água com valores muito próximos, praticamente semelhantes.

Referências bibliográficas

- [1] ANA - Agência Nacional de Águas, Plano estratégico de recursos hídricos da bacia hidrográfica dos rios Tocantins e Araguaia. Brasília, 2009. Disponível em: <http://semarh.to.gov.br/conteudo/relatorio-sintese-do-plano-estrategico-da-bacia-hidrografica-dos-rios-tocantins-e-araguaia/414>. Acessado em 18 jun. 2015.
- [2] CNT - Confederação Nacional de Transporte, Plano CNT de Transporte e Logística. 2011. Disponível em: <http://www.cnt.org.br/Imagens%20CNT/.../PlanoCNTdeLog201>. Acessado em 18 jun. 2015.
- [3] Tucci, C E M. Regionalização de vazões; Editora da Universidade Federal do Rio Grande do Sul, 2002, cap. 8.
- [4] Gottschalk, L.; YU, K.; Leblois, E.; Xiong L. Statistics of low flow: Theoretical derivation of the distribution of minimum streamflow series. *Journal of Hydrology*, Vol.481, pp.204-219, 2013, doi: 10.1016/j.jhydrol.2012.12.047.
- [5] Finkler, N. R.; Mendes, L. A.; Schneider, E. H. M.; Bortolin, T. A.; Schneider, V. E.; Análise de funções de distribuição de probabilidades na determinação de vazão mínima sazonal; XXI Simpósio de Recursos Hídricos, Brasília, Brasil, 2015.
- [6] Leme, E. J. K.; Chaudhry, F. H. Vazão Mínima do Vale Médio do Rio Jaguari Mirim. *RBRH: revista brasileira de recursos hídricos*, v. 10, n. 4, p. 127-136, out./dez. 2005, doi: 10.21168/rbrh.v10n4.p127-136.
- [7] Kankam-Yeboah, K.; Logah, F.; Amisigo, B.; Bekoe, E.; Ofori, D.; Akudago. Low Flow Analysis Of The Black Volta River At Lawra In Ghana. *Journal of Environmental Hydrology*, Vol.21(2), pp.1-16, 2013. Disponível em: <http://www.hydroweb.com/protect/pubs/jeh/jeh2013/aku.pdf>. Acessado em: 25 fev. 2016.
- [8] Detzel, D. H. M.; Fernandes, C. V. S.; Mine, M. R. M.; Estacionariedade na Construção de Curvas de Permanência com Vistas à Outorga de Recursos Hídricos (2016), doi: <http://dx.doi.org/10.21168/rbrh.v21n1.p80-87>
- [9] Vestena, L. R.; Oliveira, É. D.; Cunha, M. C. Da; Thomaz, E. L.; Vazão ecológica e disponibilidade hídrica na bacia das Pedras, Guarapuava-PR, (2012) doi: <http://dx.doi.org/10.4136/ambi-agua.840>
- [10] Cruz, J. C.; Tucci, C. E. M.; Estimativa da Disponibilidade Hídrica Através da Curva de Permanência (2008), doi: 10.21168/rbrh.v13n1.p111-124.
- [11] Departamento Nacional de Infraestrutura de Transportes; Hidrovia do Tocantins Araguaia; 27/12/2018; <https://www.gov.br/dnit/pt-br/assuntos/aquaviario/hidrovia-do-tocantins-araguaia>, acessada em setembro de 2020.
- [12] CODOMAR – Companhia de Docas do Maranhão. Edital nº 3/2013. Disponível em: <http://www.codomar.gov>.

- br /sistem/biblioteca/ 20131126172525.pdf. Acessado em: 17 dez. 2014.
- [13] Projeto Brasil Das Águas, Região hidrográfica do Tocantins-Araguaia. Disponível em: <http://brasildasaguas.com.br/educacional/regioes-hidrograficas/regiao-hidrografica-do-tocantins/>. Acessado em: 07 ago. 2015.
- [14] ANTAQ - Agência Nacional de Transportes Aquaviários, Plano Nacional de Integração Hidroviária – Bacia do Tocantins Araguaia. 2013. Disponível em: <http://www.antaq.gov.br/portal/PNIH/BaciaTocantinsAraguaia>. Acessado em 07 ago. 2015.
- [15] ANA - Agência Nacional de Águas, Sistema Nacional de Informações sobre Recursos Hídricos. Disponível em: <http://www2.snirh.gov.br/home/webmap/viewer.html?webmap=b517f199e5624a729aa85bca15310229>. Acessado em 17 dez. 2014
- [16] Clarke, R. T.; Hidrologia Ciência e Aplicação; Editora UFGRS, 2009, cap. 17.

Uma arquitetura baseada em micro serviços para apoio a transparência em processos de compras E-Gov

Luís Vinicius Pinho Bueno de Carvalho*, Ricardo Choren Noya
Instituto Militar de Engenharia (IME),
Praça General Tibúrcio, 80, 22290-270,
Praia Vermelha, Rio de Janeiro, RJ, Brasil.
*lvini2@yahoo.com.br

RESUMO: Ferramentas de governo eletrônico surgiram com o objetivo de divulgar um conjunto de informações geradas e manuseadas pelos governos, que são de interesse do cidadão e que estão associadas à prestação de serviços públicos. Dentre estas informações se destacam as relacionadas ao processo de compra governamental, pois essas estão diretamente relacionadas ao emprego do dinheiro oriundo dos diversos tributos pagos pelo cidadão. Sendo assim, as ferramentas que armazenam essas informações são chamadas de sistemas de compras governamentais e devido à alta complexidade do processo de contratação de bens e serviços feito pela administração pública, esses sistemas acabam por armazenar uma grande quantidade de informações. Com isso, esses sistemas devem prover uma grande quantidade de funcionalidades o que por consequência causa o espalhamento dessas informações e dissipa o conceito de transparência, principalmente em relação ao entendimento e usabilidade da informação requerida pelo cidadão. O presente trabalho apresenta uma arquitetura baseada em micro serviços que contribui para a junção das informações que hoje se encontram espalhadas nas diversas funcionalidades do sistema “Compras Governamentais” com o objetivo de melhorar a percepção de transparência das informações.

PALAVRAS-CHAVE: Micro Serviços. Transparência. Governo Eletrônico.

ABSTRACT: E-government tools have emerged with the aim of disseminating a set of information generated and handled by governments, which are of interest to the citizen and which are associated with the provision of public services. Among this information, those related to the government procurement process stand out, as they are directly related to the use of money from the various taxes paid by the citizen. Therefore, the tools that store this information are called government procurement systems and due to the high complexity of the procurement of goods and services made by the public administration, these systems end up storing a large amount of information. As a result, these systems end up having to provide a large amount of functionality, which consequently causes the spread of this information and dispels the concept of transparency, especially in relation to the understanding and usability of the information required by the citizen. The present work presents an architecture based on micro services that contributes to the junction of the information that today is scattered in the several in the “Governmental Procurement” system with the objective of improving the perception of transparency of this information.

KEYWORDS: Micro Services. Transparency. E-government.

1. Introdução

As ferramentas de governo eletrônico surgiram no Brasil no fim da década de 90 com o objetivo de ofertar serviços públicos ao cidadão através da internet [1]. Com o passar dos anos, o conceito evoluiu e passou a incorporar melhorias nos processos decisórios da administração pública, na medida em que os serviços ofertados ao cidadão tinham como características a integração entre governos e a transparência das informações disponibilizadas [2].

Dentre muitos tipos de ferramentas de governo eletrônico desenvolvidas, se destacam as relacionadas a promover os processos de compras feitos pelo governo e as que apresentam as informações relacionadas a

estes processos. Estas ferramentas são chamadas de sistemas de compras governamentais e são destinadas a apresentar ao cidadão as informações relacionadas a contratos celebrados pela administração pública, tais como, os preços praticados nos contratos, os prazos de entrega dos bens e serviços adquiridos e a qualidade do que está sendo contratado pelo governo [3].

Os processos de contratação adotados pelos governos normalmente são muito burocráticos, envolvem diferentes fases e em cada fase são gerados um grande conjunto de informações. Por sua vez, essas são armazenadas pelos sistemas de compras governamentais através de muitas funcionalidades, o que implica no espalhamento das informações. Consequentemente, existe muita dificuldade para o cidadão entender e analisar tais informações.

Particularmente no Brasil, o sistema “Compras Governamentais” é a aplicação desenvolvida pelo governo brasileiro que apoia o processo de contratação e fornece amplo e irrestrito acesso às informações. O processo de contratação possui várias etapas e cada etapa é apoiada por uma funcionalidade ou subsistema específico, que tem a função de armazenar as informações geradas ao longo do processo. Por ofertar muitas funcionalidades, as informações acabam por se espalhar e todo o processo de tratamento e recuperação das informações acaba por ser transferido para o usuário do sistema. Desta forma, a transparência das informações acaba por ser comprometida, na medida que a recuperação destas vai depender exclusivamente do conhecimento do cidadão sobre o processo de contratação e sobre o uso do sistema.

Por exemplo, imagine o caso no qual um cidadão deseje saber qual fornecedor assinou um determinado contrato administrativo junto ao governo do Brasil, suas datas de licitação, seus valores monetários e suas datas do contrato. Inicialmente, ele teria que acessar o sistema com o objetivo de encontrar a funcionalidade relacionada ao contrato e posteriormente a relacionada à licitação que originou o contrato junto a administração pública. Após conseguir identificar essas funcionalidades, o cidadão necessitará entender os dados de entrada requeridos por cada funcionalidade.

Durante esse processo de acesso às duas funcionalidades, cabe ao próprio cidadão, todo o trabalho de armazenar os dados intermediários e posteriormente juntá-los com o objetivo obter as informações que deseja. Pensando ainda em termos de transparência, não se pode esquecer que essas informações poderão ser usadas como subsídio para alguma tomada de decisão.

De fato, sob a ótica de transparência governamental, esse é um problema geral: as informações disponibilizadas pelas esferas governamentais são diversas, geradas de várias formas e que muitas vezes tratam uma mesma entidade de formas distintas, comprometendo o entendimento, análise e discussão das informações, que são o objetivo final da

transparência [4].

Assim, este trabalho propõe um ambiente voltado a auxiliar os cidadãos a conseguir, de fato, uma transparência em relação aos dados disponibilizados em sistemas governamentais. Esse ambiente é composto de duas partes: uma biblioteca de micro serviços voltados para a coleta e o tratamento de informações disponibilizadas em sistemas governamentais, e uma ferramenta de orquestração desses micro serviços. Esse ambiente tem como objetivo final focar as componentes de transparência relacionadas à análise e discussão da informação [4].

O artigo está estruturado da seguinte forma: a Seção 2 apresenta os conceitos básicos e necessários ao entendimento do trabalho, a Seção 3 apresenta a solução proposta, a Seção 4 apresenta os resultados da solução proposta e a Seção 5 apresenta a conclusão do trabalho.

2. Conceitos básicos

2.1 Transparência em Sistemas Governamentais

Ao longo das três últimas décadas, o governo brasileiro tem implementado um arcabouço normativo para dar maior transparência das ações governamentais ao cidadão. A primeira iniciativa foi a lei geral de licitações (Lei 8666/1993) [5], que define um arcabouço de regras para a contratação de bens e serviços pela administração pública. Por sua vez, a lei complementar 131, conhecida como lei da transparência [6], trouxe uma inovação no sentido de tornar pública todas as informações relacionadas à execução orçamentária dos diferentes entes governamentais. Por fim, a lei de acesso à informação [7] trouxe uma contribuição mais abrangente na medida em que proporcionou ao cidadão o acesso a qualquer informação governamental.

Como forma de atender a legislação vigente e ao mesmo tempo fornecer de forma proativa e transparente o acesso às informações governamentais, diversos sistemas foram desenvolvidos pelo governo brasileiro. Entretanto, a transparência é um conceito muito mais amplo do que simplesmente um conjunto

de sistemas que permitem um amplo acesso às informações. Segundo Sommerville [8] e Pressman [9] a transparência, pode ser compreendida como sendo um requisito de qualidade, sendo muito mais amplo do que simplesmente ofertar um amplo acesso à informação, sendo que o conceito também inclui a capacidade de uso, entendimento e auditoria da informação requerida [10].

As características da transparência podem ser compreendidas como etapas de um grande ciclo [4], sendo que a representação por etapas mostra que o conjunto de características estão interligadas. O ciclo de transparência se inicia na etapa de divulgação da informação, sendo procedido respectivamente pelas etapas de simplificação e disseminação da informação.

Após concluídas estas três primeiras etapas, a informação passa a ser de domínio público e o conjunto de características passa a ser complementado pela etapa de entendimento e análise da informação que está associada à possibilidade de compreensão e uso da informação pelo cidadão, pela etapa de discussão da informação que está associada a possibilidade dos cidadãos discutirem o conteúdo da informação e pela etapa de feedback da informação que está relacionada à possibilidade do cidadão questionar o conteúdo da informação.

A **figura 1** mostra o Ciclo de transparência bem como as suas respectivas etapas.

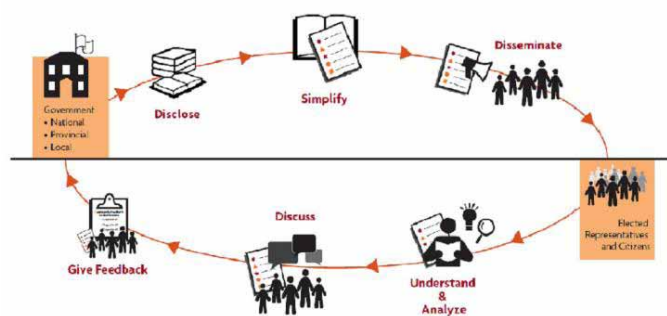


Fig. 1 – Ciclo de transparência [4]

No que se refere ao Brasil, mais especificamente ao Governo Federal, o sistema “Compras Governamentais” busca ser a ferramenta de transparência sobre a gestão de recursos públicos. No entanto, esse sistema apresenta várias

limitações relacionadas à questão da transparência, ficando restrita às três primeiras fases do ciclo da transparência, mas oferecendo pouco ou nenhum suporte à análise, ao entendimento e à discussão da informação. Isto fica evidenciado pelas seguintes deficiências: a possibilidade do cidadão não conseguir utilizar as funcionalidades providas pelo sistema, a falta de orientação ao usuário acerca dos parâmetros de entrada requeridos pelas funcionalidades, a falta de orientação sobre os mecanismos de funcionamento das funcionalidades ofertadas, a falta de mecanismos que armazenem as informações que o usuário consegue recuperar e principalmente a falta de mecanismos que automatizem o processo de junção das informações recuperadas pelo usuário.

2.2 Arquitetura orientada a micro serviços

Uma arquitetura baseada em micro serviços é apresentada como uma evolução da arquitetura monolítica na medida em que deixa de ser vista como um conjunto de recursos de software integrados em uma única aplicação [11] e passa a ser vista como um conjunto de componentes, sendo que cada componente trabalha de maneira independente e executa uma tarefa distinta [12]. Um micro serviço é um componente coeso, executa seus próprios processos [13], representa um único conceito (abstração) [14], pode se comunicar com outros micro serviços através de interfaces padronizadas [15], além de executar uma tarefa distinta e independente [16], o que tem contribuído para a evolução das soluções monolíticas para soluções distribuídas.

Além disso, o uso de micro serviços tende a tornar mais fácil o processo de desenvolvimento de aplicações, reduzindo sua complexidade por meio do reuso dos próprios micro serviços, trazendo também outras vantagens, tais como, novos atributos de qualidade, a facilidade de manutenção, de escalabilidade e de disponibilidade da aplicação [17]. Outro aspecto importante a ser destacado é que um conjunto de micro serviços pode trabalhar de forma orquestrada, similar a um processo de negócio na qual existe a figura de uma tarefa que controla e coordena as

demais tarefas dentro do processo de negócio [18,19].

Uma orquestração é normalmente utilizada para executar fluxos de trabalhos orientados a serviços, tendo como principais vantagens o controle sobre o processo e o suporte a automação dos processos [20]. Como exemplo, é possível citar uma orquestração de três micro serviços que tenha como objetivo recuperar, apresentar e armazenar as informações específicas de um contrato administrativo. O primeiro micro serviço da orquestração recebe como insumo um conjunto de atributos do contrato e com isso busca as informações relacionadas ao contrato, sendo que ao término do processamento o micro serviço devolve como resultado as informações recuperadas.

O resultado obtido com o processamento do primeiro micro serviço servirá de insumo para o processamento do segundo micro serviço que recebe as informações recuperadas, apresenta estas informações para o cidadão e posteriormente as devolve como resultado. Por último, o terceiro micro serviço recebe as informações devolvidas pelo segundo micro serviço e com isso irá armazenar estas informações em um histórico, sendo esta a última etapa da orquestração.

Assim, em uma orquestração, micro serviços trabalham em conjunto com o objetivo de implementar uma necessidade de transparência requerida pelo cidadão.

3. Solução proposta

Com base na necessidade de transparência definida pelo ciclo da transparência, este trabalho inicialmente buscou desenvolver uma biblioteca de micro serviços genéricos e customizáveis. O objetivo desta biblioteca é permitir que um cidadão, dada sua necessidade de transparência, possa configurar, orquestrar e usar tantos quantos desses micro serviços sejam necessários. Cabe salientar que, por ser uma biblioteca, o conjunto de micro serviços disponibilizados pode ser estendido, caso novas demandas sejam levantadas. Em outras palavras, a solução proposta não busca limitar a biblioteca de micro serviços ao conjunto inicial já disponibilizado pelo trabalho – sendo essa uma vantagem do emprego da arquitetura de micro

serviços

Para buscar um conjunto inicial de micro serviços o mais genérico possível, esse trabalho disponibiliza os seguintes componentes:

Micro Serviço de Leitura: Micro serviço que implementa a tarefa “Ler Informações de Entrada” na medida em que analisa os atributos da informação requerida pelo usuário do sistema “Compras Governamentais”. Este micro serviço recebe como parâmetro de entrada um arquivo XML (que contém os atributos de pesquisa previamente definidos pelo usuário), faz a localização e a abertura do arquivo XML, posteriormente faz a leitura dos atributos e os armazena em uma lista auxiliar. Ao término da leitura do arquivo XML a lista auxiliar que contém todos os atributos lidos é retornada pelo micro serviço de Leitura

Micro Serviço de Busca: Micro serviço que implementa a tarefa “Buscar Informações” na medida em que vai buscar o código HTML da funcionalidade requerida pelo usuário e que pode conter os atributos capturados pelo micro serviço de Leitura. Este micro serviço recebe como parâmetros de entrada o endereço web da funcionalidade da aplicação “Compras Governamentais” e o conjunto de atributos capturados pelo micro serviço de Leitura. Após receber os parâmetros de entrada, o micro serviço de Busca verifica de forma preliminar se o endereço web é válido e em caso afirmativo o micro serviço de Busca faz a captura do código HTML da funcionalidade, armazenando-o em uma variável auxiliar, sendo esta retornada ao término do processamento do micro serviço de Leitura.

Micro Serviço de Captura: Micro serviço que implementa a tarefa “Capturar Informações” na medida em que vai fazer a verificação de existência dos atributos retornados pelo micro serviço de Leitura no código HTML retornado pelo micro serviço de Busca. Este micro serviço recebe como parâmetros de entrada os atributos capturados pelo micro serviço de Leitura e o código HTML retornado pelo micro serviço de Busca. Após receber os parâmetros de entrada, o micro serviço de Captura cria uma lista auxiliar e faz a verificação dos atributos definidos pelo

usuário com o conteúdo das “tags” do código HTML. Caso o atributo esteja contido no código HTML será armazenado na lista auxiliar o nome do atributo e o seu respectivo valor e caso contrário será armazenado na lista auxiliar o atributo e a frase “Não Encontrado”. Ao término do processamento a lista auxiliar contendo as informações obtidas é devolvida pelo micro serviço de Captura.

Micro Serviço de Listagem: Micro serviço que implementa a tarefa “Listar Informações”, na medida em que apresenta as informações capturadas pelo micro serviço de Captura. Este micro serviço recebe como parâmetro de entrada a lista devolvida pelo micro serviço de Captura e de posse desta lista o micro serviço de Listagem recupera as informações e as apresenta ao usuário. Ao término do processamento o micro serviço de Listagem devolve a lista contendo as informações capturadas pelo micro serviço de Captura.

Micro Serviço de Gravação: Micro serviço que implementa a tarefa “Armazenar Resultados”, na medida em que armazena os resultados obtidos pelo micro serviço de Captura, permitindo a construção de um histórico de informações, sendo este o responsável por apresentar uma visão unificada acerca de todas as informações recuperadas. Este micro serviço recebe como parâmetro de entrada a lista devolvida pelo micro serviço de Captura e de posse desta lista o micro serviço de Gravação armazena estas informações em um histórico. Ao término do processamento o micro serviço de Gravação devolve o histórico com as informações recuperadas.

Com o objetivo de fazer com que os micro serviços listados trabalhem de forma orquestrada é necessário que exista um micro serviço que possa instanciar cada um dos micro serviços listados anteriormente, fornecendo os insumos necessários para o correto funcionamento de cada micro serviço da orquestração, recuperando os resultados intermediários produzidos e apoiando a apresentação do resultado final da orquestração ao cidadão.

Este micro serviço que irá instanciar os micro serviços que compõem a orquestração é conceituado como micro serviço orquestrador, sendo a sua função

instanciar respectivamente os micro serviços de Leitura, Busca, Captura, Listagem e Gravação. Ao longo deste processo, o micro serviço orquestrador vai recuperar e armazenar os parâmetros de entrada para cada micro serviço, bem como os resultados intermediários obtidos.

Sendo assim, ao instanciar o micro serviço de Leitura, o micro serviço orquestrador recebe um resultado intermediário e isto será o insumo que o micro serviço orquestrador utilizará para instanciar o micro serviço de Busca.

Por sua vez, com término do processamento do micro serviço de Busca o micro serviço orquestrador recupera os resultados obtidos e com isso instancia o micro serviço de Captura. Ao término do processamento, o micro serviço de Captura devolve informações para o micro serviço orquestrador que posteriormente serão utilizadas pelo micro serviço orquestrador para instanciar o micro serviço de Listagem.

Com término do processamento do micro serviço de Listagem, as informações são enviadas ao micro serviço orquestrador. Com isso, o micro serviço orquestrador instancia o micro serviço de Gravação, que recebe as informações requeridas e as armazena em um histórico. Aqui se encerra o processo de orquestração e neste ponto o cidadão tem atendida a sua necessidade de transparência.

4. Cenário de execução

Neste cenário de execução, o cidadão necessita ter uma visão das informações do contrato administrativo que possui como informação identificadora o código “07001950001102006” e das informações do respectivo processo de licitação que deu origem a esse contrato. Para ter atendida a sua necessidade de transparência, o cidadão necessita que todas as informações sejam mostradas de forma consolidada. Além disso, para melhor uso, compreensão e discussão sobre os parâmetros do contrato assinado, o cidadão precisa de informações relacionadas a todas as datas dos processos de licitação e da assinatura do contrato.

Sendo assim, cada etapa do processo de negócio

vai corresponder a um micro serviço da arquitetura e a execução do processo de negócio está associada à execução de forma sequencial de um conjunto de micro serviços, cada um atendendo uma tarefa específica. Inicialmente, o usuário irá criar um arquivo XML com o objetivo de definir os atributos relacionados à funcionalidade de contrato e posteriormente irá criar outro arquivo XML para definir os atributos relacionados à funcionalidade de licitação. Os arquivos XML serão salvos na pasta “schemas” do diretório “C” do sistema operacional “Windows”. Esses atributos correspondem ao tipo de informação que usuário acha que pode estar contido na funcionalidade de contrato e na funcionalidade de licitação.

Para a funcionalidade de contrato foram definidos pelo usuário os atributos “Data”, “Valor”, “Modalidade” e “Hora”. Por sua vez, para a funcionalidade de licitação foram definidos os atributos “Data”, “Responsável”, “Nomenclatura” e “Hora”.

A **tabela 3** apresenta os atributos relacionados à funcionalidade de contratos em formato de um arquivo XML.

Tag Inicial	Tag Final
<list>	
<schema><atributo> Data	</schema></atributo>
<schema><atributo> Valor	</schema></atributo>
<schema><atributo> Modalidade	</schema></atributo>
<schema><atributo> Hora	</schema></atributo>
	</list>

Tab. 3 – XML relacionado a funcionalidade de contrato

A **tabela 4** apresenta os atributos relacionados à funcionalidade de licitação em formato de um arquivo XML.

Tag Inicial	Tag Final
<list>	
<schema><atributo> Data	</schema></atributo>
<schema><atributo> Responsável	</schema></atributo>
<schema><atributo> Nomenclatura	</schema></atributo>
<schema><atributo> Hora	</schema></atributo>
	</list>

Tab. 4 – XML relacionado a funcionalidade de licitação

Após a definição dos atributos que serão utilizados pela arquitetura, o usuário selecionará de forma

sequencial os micro serviços de Leitura, Busca, Captura, Listagem e Gravação, que trabalharão de forma orquestrada com o objetivo de recuperar as informações referentes ao contrato. Esses micro serviços serão instanciados pelo micro serviço orquestrador.

Conforme apresentado na **figura 2**, o usuário fornece parâmetros de entrada utilizando uma interface padronizada para o processo de pesquisa a todas as funcionalidades do sistema “Compras Governamentais”, sendo elas: a “URL” corresponde ao endereço da funcionalidade de contrato (“http://compras.dados.gov.br/contratos/doc/contrato/”), o campo “Complemento” corresponde ao código do contrato (“07001950001102006.html”), o campo “Localização do Schema (Nome da Pasta)” corresponde ao endereço do arquivo XML que contém os atributos relacionados a contrato (“schemas”) e o campo “Schema” contém o arquivo XML com os respectivos atributos (“contrato”).

Fig. 2 – Interface Padronizada Para Pesquisa

Com base nas informações fornecidas pelo usuário, micro serviço orquestrador é acionado, recebe e armazena os parâmetros de entrada e logo em seguida instancia o micro serviço de Leitura passando como parâmetros o endereço do arquivo que contém os atributos configurados pelo usuário e o nome do próprio arquivo XML que contém os atributos. De posse destas informações o micro serviço de Leitura faz a leitura dos atributos de pesquisa, os armazena em uma estrutura auxiliar e ao término do seu processamento retorna a estrutura auxiliar para o micro serviço orquestrador.

O micro serviço orquestrador armazena esta estrutura e depois instancia o micro serviço de Busca passando o endereço da funcionalidade de contrato e o código do contrato requerido pelo usuário (“07001950001102006”)

e com isso o micro serviço de Busca faz a leitura do código HTML desta funcionalidade, o armazena e ao término do seu processamento ele devolve ao micro serviço orquestrador o código HTML da funcionalidade.

Depois o micro serviço orquestrador instancia o micro serviço de Captura passando como parâmetros o código HTML da funcionalidade de contrato e a estrutura auxiliar com os atributos definidos pelo usuário e lidos previamente pelo micro serviço de Leitura. De posse dos parâmetros, o micro serviço de Captura faz a leitura de cada atributo relacionado a contrato e verifica se este atributo está contido em alguma “tag” do código HTML, armazenando os resultados em uma nova estrutura auxiliar, que é retornada para o micro serviço orquestrador ao término do processamento.

Com isso o micro serviço orquestrador instancia o micro serviço de Listagem passando como parâmetro a nova estrutura gerada pelo micro serviço de Captura. Esta estrutura é lida pelo micro serviço de Listagem e os resultados são apresentados ao usuário.

Com o término do processamento do micro serviço de Listagem, a estrutura auxiliar é retornada para o micro serviço orquestrador, que instancia o micro serviço de Gravação passando como parâmetros as informações recuperadas pelo micro serviço de Captura. De posse destas, o micro serviço de Gravação efetua a gravação dos atributos e o seu respectivo valor (caso o atributo tenha sido encontrado) ou “não encontrado” (caso o atributo não tenha sido encontrado), construindo assim um histórico com as informações recuperadas.

A **figura 3** apresenta as informações relacionadas ao contrato que foram recuperados pela nova arquitetura.

Atributo:	Valor
Schema:	contrato
URL:	http://compras.dados.gov.br/contratos/doc/contrato/
Data de Termina da Vigência	24/04/2009
Data de Assinatura	25/10/2005
Data de Inicio da Vigência	25/10/2006
Valor inicial	R\$ 124,000.00
Modalidade da Licitação	S: PREGÃO
Hora	Não Encontrado

Botões: Repetir Operação, Gravar no Histórico

Fig. 3 – Informações do Contrato Recuperadas Pela Arquitetura

Após gravar os atributos referentes à funcionalidade de contrato no histórico, há o retorno para o micro serviço orquestrador, que por sua vez inicia novamente o processo de orquestração, tendo o preenchimento do formulário de entrada com o campo “URL” (“http://compras.dados.gov.br/licitacoes/doc/licitacao/”), o campo “Complemento” (“07001905000822006.html”), o campo “Localização do Schema (Nome da Pasta)” (“schemas”) e o campo “Schema” (“licitacao”). A **figura 4** apresenta os dados de entrada fornecidos pelo usuário para a pesquisa de informações na funcionalidade de licitação.

URL*: http://compras.dados.gov.br/licitacoes/doc/licitacao/

Complemento*: 07001905000822006.html

Localização do Schema (Nome da Pasta)*: schemas

Schema*: licitacao

Botões: Pesquisar, Histórico

Fig. 4 – Interface Padronizada Para Pesquisa

Com base nas informações o micro serviço orquestrador armazenará os dados de entrada e instancia o micro serviço de Leitura passando como parâmetros o endereço do arquivo que contém os atributos configurados pelo usuário e o nome do próprio arquivo XML que contém os atributos. De posse dessas informações o micro serviço de Leitura faz a leitura dos atributos de pesquisa, os armazena em uma estrutura auxiliar e ao término do processamento esta estrutura auxiliar é devolvida ao micro serviço orquestrador. O micro serviço orquestrador armazena essa estrutura e depois instancia o micro serviço de Busca passando o endereço da funcionalidade de licitação e o código da licitação requerida (“07001905000822006”) pelo usuário e com isso o micro serviço de Busca faz a leitura do código HTML desta funcionalidade, o armazena e ao término do seu processamento este código é devolvido para o micro serviço orquestrador.

Logo em seguida, o micro serviço orquestrador instancia o micro serviço de Captura passando como parâmetros o código HTML da funcionalidade de licitação e a estrutura auxiliar com os atributos

definidos pelo usuário e recuperados pelo micro serviço de Leitura. De posse dos parâmetros o micro serviço de Captura faz a leitura de cada atributo relacionado a licitação e verifica se este atributo está contido nas “tags” do código HTML, armazenando os resultados em uma nova estrutura auxiliar, que é retornada para o micro serviço orquestrador ao término do processamento.

Com isso, o micro serviço orquestrador instancia o micro serviço de Listagem passando como parâmetro a estrutura gerada pelo micro serviço de Captura, que será manipulada pelo micro serviço de Listagem com o objetivo de recuperar e apresentar as informações ao usuário.

A **figura 5** mostra as informações relacionadas a licitação recuperadas pela arquitetura de micro serviços.



Fig. 5 – Informações da Licitação Recuperadas Pela Arquitetura

Com o término do processamento do micro serviço de Listagem a estrutura auxiliar é retornada para o micro serviço orquestrador, que instancia o micro serviço de Gravação passando como parâmetros as informações recuperadas pelo micro serviço de Captura. Dessa forma, o micro serviço de Gravação armazena estas informações no histórico de pesquisas. Com a adição das informações relacionadas a licitação, o histórico passa a contar com informações de contrato e informações de licitação vistas de forma unificada pelo usuário. Neste ponto, o usuário passa a ter todas as informações requeridas de forma unificada e isto vai facilitar o entendimento, o uso, a

discussão e a possibilidade de questionamento do ente governamental.

A **figura 6** apresenta as informações que foram consolidadas pela arquitetura de micro serviços.



Fig. 6 – Histórico de Informações Recuperadas

Posteriormente, o próprio usuário poderá incluir novas informações no histórico através de novas configurações nos arquivos XML ou até mesmo utilizando os mesmos arquivos XML para realização de consultas em outras funcionalidades do sistema “Compras Governamentais”.

Sendo assim, há perceptível melhoria na transparência das informações na medida em que: 1- o usuário passa a poder compreender melhor as informações disponibilizadas pela arquitetura através de uma visão unificada; 2- passam a existir subsídios para discussão das informações recuperadas; 3- passam a existir maiores possibilidades de questionar o ente governamental ou os órgãos de controle sobre as informações disponibilizadas no sistema “Compras Governamentais”.

5. Análise de resultados

A arquitetura proposta no presente trabalho implementa a necessidade de transparência definida pelo usuário do sistema “Compras Governamentais” através da orquestração de um conjunto de micro serviços. Esta abordagem apresenta como principais melhorias a transferência para o software de todo o tratamento e recuperação da informação requerida

pelo usuário, a possibilidade de armazenamento dos resultados obtidos no processo bem como a junção das informações contidas nas diversas funcionalidades do sistema “Compras Governamentais” e a posterior visão unificada destas mesmas informações pelo cidadão.

Para mostrar a efetividade da solução proposta (o quanto ela pode ajudar o cidadão na tarefa de recuperar informação governamental para uma demanda de transparência), esse trabalho utiliza três métricas: número de redirecionamentos entre páginas Web, número de consultas auxiliares em diferentes sistemas e número de informações de entrada requeridas. Nas subseções a seguir, apresentamos os resultados levantados para tais métricas aplicadas ao cenário discutido na seção 4.

5.1 Redirecionamentos entre páginas web

Esta métrica tem como objetivo avaliar a quantidade de redirecionamentos entre páginas web feitas pelo sistema “Compras Governamentais” e pela solução proposta. Esta métrica vai avaliar o esforço requerido do cidadão desde o momento em que ele fornece o primeiro parâmetro de entrada para cada uma das soluções até o momento em que ele consegue obter as informações requeridas.

Sem utilizar a solução proposta, o cidadão terá que usar apenas os sistemas governamentais disponíveis. Na funcionalidade de consulta a informações de contrato ocorrem no total 6 redirecionamentos entre páginas web: o primeiro ocorre após o usuário selecionar a opção de consulta do sistema; o segundo ocorre após o usuário selecionar a opção “contrato”; o terceiro ocorre após o usuário informar o cadastro do contrato no SIGS; o quarto ocorre após o usuário inserir as informações relacionadas ao órgão responsável pela assinatura do contrato; o quinto ocorre após o usuário informar o período de assinatura do contrato e o sexto ocorre quando o usuário seleciona o contrato de uma lista de contratos ordenados por código.

Já na funcionalidade de consulta a informações de licitação ocorrem no total 6 redirecionamentos entre páginas web: o primeiro ocorre após o

usuário selecionar a opção de consulta do sistema; o segundo ocorre após o usuário selecionar a opção “licitação”; o terceiro ocorre após o usuário informar a funcionalidade que mostra os resultados das licitações; o quarto ocorre após o usuário inserir as informações relacionadas ao órgão responsável pela condução da licitação; o quinto ocorre após o usuário informar o período do fechamento da licitação e a sua respectiva modalidade e o sexto ocorre quando o usuário seleciona a licitação de uma lista apresentada.

Utilizando a solução proposta, não existem redirecionamentos entre páginas web. Após o fornecimento dos parâmetros de entrada a solução executa a orquestração dos micro serviços e apresenta os resultados ao usuário.

5.2 Consultas em outros sistemas

Esta métrica tem como objetivo avaliar a necessidade do cidadão em realizar consultas específicas em sistemas distintos. Esta métrica vai analisar o esforço requerido do cidadão no sentido de fornecer as informações de entrada, tanto pelo sistema “Compras Governamentais” quanto pela solução proposta.

Sem utilizar a solução proposta, na funcionalidade de consulta a informações de contrato são necessárias 4 consultas auxiliares sob o risco do processo de pesquisa não retornar o contrato desejado, sendo elas: a primeira para verificar se o contrato está ou não cadastrado no SISG (Sistema de Serviços Gerais); a segunda para verificar se o órgão público fez a contratação no período requerido pelo usuário; a terceira para verificar o código da unidade administrativa; a quarta para verificar se a modalidade de contrato está compreendida no período requerido pelo usuário.

Na funcionalidade de consulta a informações de licitação são necessárias 5 consultas auxiliares sob o risco do processo de pesquisa não retornar o contrato desejado, sendo elas: a primeira para verificar se a licitação foi homologada; a segunda para verificar se o órgão público fez a contratação no período requerido pelo usuário; a terceira para verificar o código da unidade administrativa; a quarta para verificar a

modalidade da licitação, a quinta para verificar a data de homologação da licitação ou o seu código identificador.

Utilizando a solução proposta, o cidadão somente precisa fazer quatro consultas auxiliares, sendo duas consultas para buscar o endereço do código HTML da funcionalidade de contrato e seu código identificador e mais duas consultas auxiliares para buscar o endereço do código HTML da funcionalidade de licitação bem como seu código identificador.

5.3 Entradas requeridas

Esta métrica tem como objetivo avaliar a volumetria de informações requeridas para realização de pesquisa nas funcionalidades do sistema. Esta métrica vai analisar a eficiência do sistema “Compras Governamentais” e da solução proposta em relação a quantidade de informações requeridas como parâmetro de entrada para que o processo de pesquisa seja concluído e os resultados requeridos sejam apresentados ao usuário.

Sem utilizar a solução proposta, na funcionalidade de consulta a informações de contrato são necessárias 8 informações de entrada, sendo elas: o Estado da federação em que se localiza o órgão público que assinou o contrato, o órgão público que celebrou o contrato, a unidade administrativa que assinou o contrato, o período de assinatura do contrato que se desdobra em mais quatro informações (mês e ano do início do período de assinatura e mês e ano do período de término da assinatura), a modalidade do contrato bem como o seu código.

Já na funcionalidade de consulta a informações de licitação, são necessárias 5 informações de entrada, sendo elas: o Estado da federação em que se localiza o órgão público que elaborou a licitação, o órgão público que conduziu a licitação, a unidade administrativa que homologou a licitação, a modalidade de licitação e a data de homologação da licitação.

Utilizando a solução proposta, o cidadão somente precisa passar como informação de entrada para a solução 4 dados para utilização da funcionalidade de consulta a informações de contratos, sendo eles:

o endereço HTML da funcionalidade de contrato, o código identificador do contrato, o nome da pasta que contém o arquivo XML com os atributos previamente configurados e o nome do arquivo XML. Para a utilização da funcionalidade de consulta a informações de licitação também são requeridos 4 dados de entrada, sendo eles: o código HTML da funcionalidade de licitação, o código identificador da licitação, o nome da pasta que contém o arquivo XML com os atributos previamente configurados e o nome do arquivo XML.

De uma forma consolidada, tem-se os seguintes valores para as métricas, para as duas funcionalidades (contrato e licitação). A “Redução (%)” apresentada mostra o percentual de redução de esforço do usuário para ter a sua necessidade de transparência atendida pela solução proposta.

Métrica	“Compras Governamentais”	Micro Serviços	Redução (%)
(1)	6	0	100%
(2)	4	2	50%
(3)	8	4	50%
Total	18	6	66,7%

Tab. 5 – Resultados – Informações de Contrato

Métrica	“Compras Governamentais”	Micro Serviços	Redução (%)
(1)	6	0	100%
(2)	5	2	60%
(3)	5	4	20%
Total	16	6	62,5%

Tab. 6 – Resultados – Informações de Licitação

Métrica	“Compras Governamentais”	Micro Serviços	Redução (%)
(1)	12	0	100%
(2)	9	4	55,5%
(3)	13	8	38,4%
Total	34	12	64,7%

Tab. 7 – Resultados – Consolidados

6. Conclusão

Este trabalho teve como objetivo a definição de uma arquitetura baseada em micro serviços para a busca e reunião de informações que atualmente se encontram espalhadas pelas diversas funcionalidades do sistema “Compras Governamentais”. Desta maneira, busca-se atender uma ou mais necessidades de transparência

de um usuário do sistema, sendo que esta necessidade de transparência está associada a recuperação da informação requerida, o seu entendimento e uso pelos requisitantes, a possibilidade de discussão do seu conteúdo pelo usuário e a formalização de um feedback ao ente governamental.

Analisando a literatura existente se verificam trabalhos que buscam tratar a questão da busca e consolidação de informações. O trabalho apresentado em Paiva e Revoredo [21] foca na apresentação de técnicas de tratamentos de dados que permitem a estruturação de forma mais clara e elucidativa. Por sua vez, no trabalho apresentado em Carvalho *et al* [22], existe um foco no processamento sequencial aliado a técnicas de mineração de dados e clusterização com o objetivo de criar um banco de dados baseado nas informações disponibilizadas. Por fim, os trabalhos de White [23] e Lin [24] utilizam técnicas de computação paralela para tratamento das informações.

Entretanto, essas soluções apresentam como grande limitação a questão de trabalhar com uma grande quantidade de dados. Além disso, estas soluções não permitem que o usuário tenha poder de decisão sobre quais informações poderão ser recuperadas e posteriormente consolidadas. Por fim, estas soluções também não corrigem a questão do espalhamento das informações pelas diversas funcionalidades do sistema “Compras Governamentais”.

Uma arquitetura baseada em micro serviços contorna as limitações apresentadas, pelo fato de

um micro serviço ser um componente genérico, coeso e que implementa uma tarefa específica dentro de um sistema de software. Além disto, os micro serviços podem trabalhar de forma orquestrada, com o objetivo de entregar uma necessidade de transparência definida pelo usuário do sistema “Compras Governamentais”. Além disso, uma arquitetura baseada em micro serviços é extensível, na medida em que a simples adição de novos micro serviços contribui para a implementação de novas necessidades de transparência.

As principais melhorias apresentadas pela solução são: a delegação da configuração dos atributos de pesquisa ao usuário do sistema “Compras Governamentais”, a implementação de uma interface única para pesquisa de informações em todas as funcionalidades disponibilizadas pelo sistema “Compras Governamentais” e a implementação de um mecanismo que automatiza o processo de armazenamento das informações recuperadas e auxilia o cidadão a ter uma visão unificada acerca das informações.

Portanto, este trabalho amplia o conceito de transparência fornecido pelo sistema “Compras Governamentais”, na medida em que a reunião das informações implementada pela arquitetura contribui para um melhor entendimento e discussão da informação requerida, bem como possibilita que o próprio usuário questione o ente governamental acerca de possíveis inconsistências nas informações

disponibilizadas pela administração pública.

Referências bibliográficas

- [1] Soares, L.L; Transparência em compras públicas: proposta de um índice da transparência na gestão de compras públicas aplicado aos websites de municípios brasileiros com mais de 100 mil habitantes; Dissertação de Mestrado, Universidade Federal de Santa Catarina, Brasil, 2013.
- [2] Cunha, M.A; Meios eletrônicos e transparência: a interação do vereador brasileiro com o cidadão e o poder executivo; 10º Congresso Internacional Sobre a Reforma da Administração Pública, Santiago, Chile, 2005.
- [3] Alves, V.F; O impacto da adoção de licitações eletrônicas na Gestão Pública: Uma abordagem baseada na visão dos Pregoeiros do Tribunal de Justiça de Pernambuco; Dissertação de Mestrado – Universidade Federal de Pernambuco, Recife, 2008.
- [4] Bank, T; Budget Transparency Initiative; <http://www.worldbank.org/en/topic/socialdevelopment/publication/budget-transparency-iniciative>, acessado em novembro de 2018.
- [5] Franco, I; Lei 8666, de 21 de junho de 1993; http://www.planalto.gov.br/ccivil_03/Leis/L8666cons.htm, acessa-

do em março de 2021.

- [6] Silva, L; Lei Complementar 131, de 27 de maio de 2009; http://www.planalto.gov.br/ccivil_03/LEIS/LCP/Lcp131.htm, acessado em março de 2021.
- [7] Roussef, D; Lei 12527, de 18 de novembro de 2018; http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2011/lei/l12527.htm, acessado em março de 2021.
- [8] Sommerville, I; Engenharia de Software – 6ª Edição; Pearson, Brasil, 2003.
- [9] Pressman, R.S; Engenharia de Software – Uma Abordagem Profissional – 8ª Edição; Bookman, USA, 2016.
- [10] Cappelli, C; Uma Abordagem para Transparência em Processos Organizacionais Utilizando Aspectos, Tese de Doutorado, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, Brasil, 2009.
- [11] Nagothu, D.; Xu, R; Nikouei S. Y.; Chen, Y; A Microservice-enabled Architecture for Smart Surveillance using Blockchain Technology; IEEE International Smart Cities Conference, Kansas, USA, 2018.
- [12] Perera, K. J. P. G.; Perera, I.; A Rule-based System for Automated Generation of Serverless-Microservices Architecture; IEEE International Systems Engineering Symposium, Rome, Italy, 2018.
- [13] R, Chen; S, Li; Z., Li; From Monolith to Microservices: A Dataflow-Driven Approach; 24th Asia-Pacific Software Engineering Conference; Nanjing, China, 2017.
- [14] STAA, A; Programação Modular; Campus, Brasil, 2000.
- [15] Hassan, S.; Bahsoon, R; Microservices and Their Design Trade-Offs: A Self-Adaptive Roadmap; IEEE International Conference on Services Computing; San Francisco, USA; 2016.
- [16] Bhamare, D; Samaka, M; Erbad A; Jain, R; Gupta, L; Chan, H. A; Multi-objective scheduling of micro-services for optimal service function chains; IEEE International Conference on Communications; Paris, France; 2017.
- [17] Li, S; Understanding quality attributes in micro service architecture; Asia-Pacific Software Engineering Conference Workshops; Nanjing, China, 2017.
- [18] Brocke, J.V; Rosemann, M; Manual de BPMN: Gestão de Processos e Negócios; Bookman; Brasil; 2015.
- [19] Cavalcanti, R; Modelagem de Processos de Negócio: Roteiro para a Realização de Projetos de Modelagem de Processos de Negócios; Brasport, Brasil, 2015.
- [20] Jaradat, W.; Dearle, A; Barker, A; A Dataflow Language for Decentralized Orchestration of Web Service Workflows; IEEE Ninth World Congress on Services, USA, 2013.
- [21] Paiva, E; Revoredo K; Big data and transparency: Using mapreduce functions to increase public expenditure transparency; XII Simpósio Brasileiro de Sistemas de Informação – SBSI; Florianópolis; Brasil; 2016.
- [22] Carvalho, R; Paiva, E; Rocha, H; Mendes, G; Using clustering and text mining to create a reference price database learning and nonlinear models; Learning & Nonlinear Models (LNLM) journal; Brasil; 2014.
- [23] White, T; Hadoop: The definitive guide; O'Reilly Media, Inc; USA; 2012.
- [24] Lin, J; Dyer, C; Data-intensive text processing with MapReduce. Synthesis Lectures on Human Language Technologies; Morgan and ClayPoll Publisher; USA; 2010.

Motivação para a cartografia colaborativa: um experimento no Exército Brasileiro

Carlos Eduardo Guedes*, Luciano Augusto Terra Brito

^aInstituto Militar de Engenharia (IME),
Praça General Tibúrcio, 80, 22290-270,
Praia Vermelha, Rio de Janeiro, RJ, Brasil.

*caduguedess@gmail.com

RESUMO: O presente trabalho aborda o tema mapeamento colaborativo aplicado ao contexto do Exército Brasileiro aproveitando-se da técnica da gamificação com o objetivo de aumentar o engajamento dos militares em um sistema de mapeamento colaborativo. Foi verificado experimentalmente que os elementos de design de jogos (pontos, medalhas e tabela de classificação), do modo que foram aplicados, não se mostraram eficazes na potencialização da motivação dos militares. Contudo, foi corroborada a tese de que a Inteligência Coletiva da Força Terrestre é capaz de produzir mapeamento vantajoso para a cartografia de interesse do Exército Brasileiro.

PALAVRAS-CHAVE: Mapeamento Colaborativo; Gamificação; Motivação.

ABSTRACT: The present work addresses the theme of collaborative mapping applied to the context of the Brazilian Army, taking advantage of the gamification technique in order to increase the military engagement in a collaborative mapping system. It was experimentally verified that the game design elements (points, badges and leaderboards), in the way they were applied, did not enhance the motivation of the military. However, the work corroborated the thesis that the Collective Intelligence of the Brazilian Army is capable of producing advantageous mapping for the cartography of itself interest.

KEYWORDS: Collaborative Mapping; Gamification; Motivation.

1. Introdução

As informações geográficas são essenciais para o adequado planejamento e suporte às tomadas de decisão das autoridades. No meio militar essa necessidade é ainda mais acentuada, tendo em vista que, frequentemente, essas decisões envolvem deslocamento de tropas armadas. Isso torna o conhecimento do terreno um fator importante, que pode proporcionar uma vantagem significativa em combate [1].

A desatualização das cartas topográficas e de outros dados geoespaciais de interesse da Força Terrestre é uma queixa frequente dos militares do Exército Brasileiro [2]. Diante disso é fundamental a atualização regular desses dados para atender as demandas de mapeamento de interesse da Força Terrestre com uma acurácia temporal adequada.

Além disso, outras medidas e indicadores de qualidade de dados geográficos [3] devem ser levadas em consideração para que as decisões tomadas

com base nesse mapeamento tenham o melhor embasamento possível.

Uma possível solução para mitigar o efeito da desatualização dos dados geoespaciais é o uso de informações geográficas voluntárias.

No contexto das geociências, o conjunto de informações geográficas obtidas por meio de mapeamento colaborativo é denominada Informação Geográfica Voluntária, conhecida pelo termo VGI – *Volunteered Geographic Information* [4], a qual apresenta diversas diferenças quando comparada com a informação geográfica obtida pelos órgãos oficiais de mapeamento. Dentre elas, destacam-se:

- Menor custo e tempo de coleta. Em contraponto ao método de produção cartográfica tradicional, cujo tempo médio de produção de uma carta topográfica é de 315 horas [5], a VGI coletada pode atualizar o banco de dados geoespaciais em poucos minutos. Contudo, é necessário um esforço prévio no desenvolvimento de um sistema de mapeamento

colaborativo, reunião de uma massa crítica de usuários para que o sistema seja sustentável [6] e, após a aquisição, é necessário um esforço em processamento e filtragem da grande quantidade de dados resultante do mapeamento colaborativo.

- O conhecimento local dos colaboradores a respeito das feições mapeadas é aproveitado no mapeamento [7], reduzindo a dificuldade de obtenção de informações em locais distantes das agências de mapeamento.

A aquisição das informações é realizada de maneira mais simples que a convencional, não necessitando de voluntários com formação específica na área da geoinformação, tampouco *softwares* como sistemas de informação geográfica, conforme utilizado pelas agências de mapeamento[8]. Por essa razão, a qualidade da VGI apresenta variações significativas [3], pois as contribuições podem não seguir uma metodologia predefinida. Para que um sistema de mapeamento colaborativo tenha sucesso é necessário que os usuários estejam motivados em contribuir, caso contrário, uma potencial força de trabalho poderá não ser aproveitada.

A motivação pode ser subdividida, conforme a Teoria da Autodeterminação [9], em motivação intrínseca e motivação extrínseca. A motivação intrínseca é aquela em que a pessoa não necessita de uma recompensa externa, pois ela tem um interesse natural nas atividades que fornecem novidades e desafios. Assim, essa conduz a um aprendizado e performance superiores, potencializando os resultados das organizações, por isso deve ser buscada sempre que possível [10].

Já a motivação extrínseca é aquela que surge de pressões ou recompensas externas, tal como uma recompensa financeira ou evitar uma punição.

A **tabela 1** apresenta as formas encontradas na literatura de motivar um usuário a utilizar um sistema de mapeamento colaborativo.

Tab.1 – Formas de motivação para contribuir com um sistema de mapeamento colaborativo

Forma de Motivação	Exemplo
Benefício ao colaborador	Signalez-nous [11]
Propaganda do sistema	Urbangene [11]
Gamificação	Urbanopoly [12]

Fonte: Autor.

No projeto Signalez-nous os usuários da cidade suíça Yverdon-les-Bains realizam o mapeamento colaborativo de sinais de trânsito com defeito. Desse modo as autoridades públicas recebem a localização do sinal a ser reparado e o usuário do sistema usufrui do benefício de ter os sinais de trânsito reparados mais rapidamente.

No projeto Urbangene, os usuários mapeiam as localizações de lagos próximos às zonas urbanas, pois o ecossistema aquático nessas áreas é ameaçado pelas atividades humanas, as quais fragmentam as massas d'água e reduzem a conectividade desses habitats. O experimento realizado contou com campanha em rádio e mídias digitais. Foi verificado que a propaganda do sistema aumentou significativamente as contribuições com o mapeamento.

O projeto Urbanopoly é um jogo baseado em localização para Android, cujo objetivo é conquistar o maior número de propriedades, que são pontos de interesse na vida real. O usuário precisa se locomover a determinadas áreas, coletando dados geoespaciais.

Um sistema de mapeamento colaborativo voltado às necessidades da Força Terrestre iria, em tese, atualizar os dados geoespaciais utilizados pelos próprios militares, proporcionando benefícios aos mesmos. Além disso, sendo o sistema desenvolvido pelo próprio Exército Brasileiro, adaptado à Força e difundido dentro dela, a propaganda seria realizada facilmente. Resta analisar a possibilidade de utilizar a gamificação como mais uma forma de motivação.

A gamificação consiste no uso de elementos de *design* de jogos em contextos não relacionados aos jogos, visando alterar o comportamento da pessoa para alcançar os objetivos de uma organização [13]. Alguns elementos de *design* de jogos são elencados na **tabela 2**.

Tab. 2 – Exemplos de elementos de *design* de jogos. [14]

Elemento	Descrição
Avatares	Representações visuais do personagem
Bens Virtuais	Ativos com valor percebido dentro do espaço do jogo
Caçadas	Desafios pré-definidos com objetivos e recompensas particulares
Chefões	Desafios de dificuldade elevada ao final de um nível
Coleções	Conjunto de itens ou medalhas possíveis de acumular e exibir
Combate	Batalha determinada, geralmente curta
Conquistas	Objetivos de jogo definidos
Desbloqueio de Conteúdo	Conteúdo disponível apenas após alcançar determinados objetivos
Gráficos Sociais	Representações numéricas do progresso do jogador
Medalhas	Representações visuais das conquistas
Níveis	Etapas definidas no progresso do jogador
Pontos	Representações numéricas do progresso do jogador
Presentes	Oportunidades de compartilhar recursos com outros jogadores
Tabelas de Classificação	Apresentação visual do progresso do jogador e de suas conquistas
Times	Grupos definidos de jogadores com mesmo objetivo

A escolha dos elementos de *design* de jogos é uma tarefa complexa, pois cada combinação pode ter um impacto diferente na motivação dos usuários do sistema. Muitas empresas gastam milhões de dólares no desenvolvimento de jogos que simplesmente fracassam por não aplicar corretamente alguns aspectos do *design* de jogos [15].

A combinação dos elementos de *design* de jogos mais recorrente na literatura é a tríade PBL (*Points, Badges and Leaderboards*, do inglês, Pontos, Medalhas e Tabelas de Classificação). Por esse motivo e pela sua relativa simplicidade de implementação, essa combinação foi escolhida para verificar a sua influência na motivação dos militares do Exército Brasileiro em contribuir com um sistema de mapeamento colaborativo.

Ainda há uma série de questões em aberto a respeito da motivação dos usuários em contribuir com um sistema de mapeamento colaborativo [16], demandando pesquisas no sentido de permitir uma melhor compreensão desse tema multidisciplinar.

O objetivo principal do presente artigo é verificar se a tríade PBL da gamificação é um fator que motiva os militares do Exército Brasileiro a contribuir com um sistema de mapeamento colaborativo. Para tal intento, será mensurado o impacto da gamificação nesse caso e analisado o experimento sob diferentes aspectos, tais como resposta a estímulos, faixa etária e distribuição geográfica.

2. Trabalhos relacionados

Diversos estudos foram conduzidos descrevendo sistemas de mapeamento colaborativo com elementos de *design* de jogos.

Marti *et al* [17] estudaram uma proposta de aplicativo para celular no qual os jogadores coletam observações de poluição sonora em diversos locais designados pelo aplicativo. O aplicativo solicita que o jogador realize as medições em locais específicos, sendo útil para obter atualizações constantes em diversas áreas. O estudo sugeriu que tarefas com níveis crescentes de dificuldade (mais medições e em locais mais distantes) tendem a ser mais eficazes para motivar os usuários.

Martella *et al* [12] realizaram um experimento na *University of Münster*, no qual estudantes e professores realizaram o mapeamento colaborativo *indoor* da universidade por meio de dois aplicativos para Android. Um dos aplicativos continha pontos e tabelas de classificação enquanto o outro não. O experimento foi realizado com 28 participantes e foi verificado, por meio de questionários, que a maioria dos usuários preferiram a versão do aplicativo com gamificação.

Prandi *et al* [6] analisaram o comportamento de 4 usuários em 2 jogos para smartphones com o objetivo de coletar informações geográficas. No jogo “Geo-Zombie”, os usuários devem coletar dados, como o preço de alguns produtos, em locais solicitados pelo aplicativo antes que um zumbi virtual alcance os jogadores. Por outro lado, no jogo “HINT!” os jogadores não precisam coletar tão rapidamente os dados, pois não há um zumbi virtual indo na direção dos jogadores, mas eles

são recompensados, após cada aquisição, com uma peça para montar um quebra-cabeça. O estudo ressaltou a necessidade de atingir uma massa crítica de usuários para que um sistema possa se tornar eficiente.

No âmbito da Força Terrestre, o estudo de Brito [18] sugere que a Inteligência Coletiva do Exército Brasileiro é capaz de realizar mapeamento colaborativo vantajoso para a cartografia, de interesse do Exército. Partindo dessa premissa, o presente estudo pretende verificar se a gamificação é capaz de potencializar os resultados advindos da Inteligência Coletiva do Exército Brasileiro.

3. Materiais e métodos

Para atingir o objetivo traçado foi realizado um experimento para medir a influência da gamificação em um sistema de mapeamento colaborativo no Exército Brasileiro. O universo dos militares convidados a participar do experimento foi todo o efetivo de carreira da ativa do Exército Brasileiro formado na AMAN (Academia Militar das Agulhas Negras) ou na EsSA (Escola de Sargentos das Armas) totalizando 34.476 militares.

Foi desenvolvido um sistema de mapeamento colaborativo com base no projeto *OpenStreetMap* (figura 1).

A estrutura de dados vetoriais utilizada pelo projeto *OpenStreetMap* possui algumas diferenças em relação à definida pelas Especificações Técnicas para Aquisição de Dados Geoespaciais Vetoriais (ET-ADGV) [19], que é a utilizada na linha de produção cartográfica do Serviço Geográfico do Exército. Para o escopo da pesquisa, as diferenças relevantes entre as duas estruturas de dados são estas:

- Na ET-ADGV os dados geográficos são armazenados em banco de dados como geometrias do tipo ponto, linha, polígono ou complexo. Por outro lado, no *OpenStreetMap* os dados geográficos são armazenados na forma de *node* (equivalente ao ponto), *way* (equivalente à linha quando aberto ou ao polígono quando fechado) ou *relation* (equivalente ao complexo). O *way* é uma lista ordenada de nodes e o *relation* é uma lista ordenada de nodes, ways e relations;
- Na ET-ADGV são definidos os atributos obrigatórios a serem preenchidos ou selecionados para as feições de cada classe. No *OpenStreetMap* os dados são armazenados com uma lista de *tags* associada ao objeto adquirido. As *tags* não são obrigatórias, podendo haver mais de uma *tag* para um objeto.

Pelo fato de o projeto ser de código aberto, foi possível desenvolver as funcionalidades necessárias para inserir

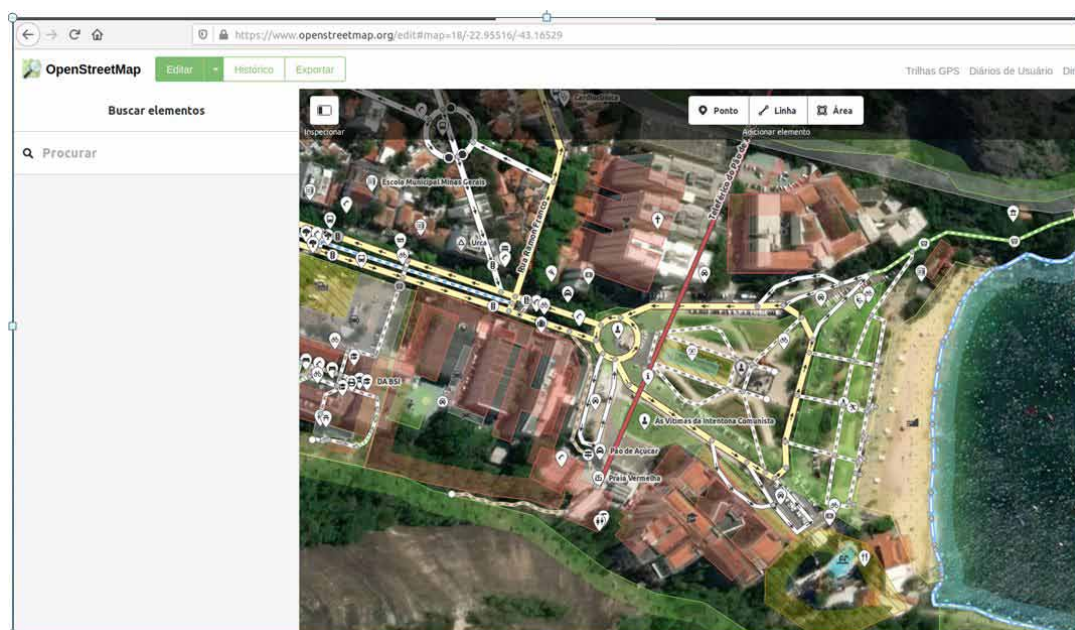


Fig. 1 – Tela de Edição do OpenStreetMap.

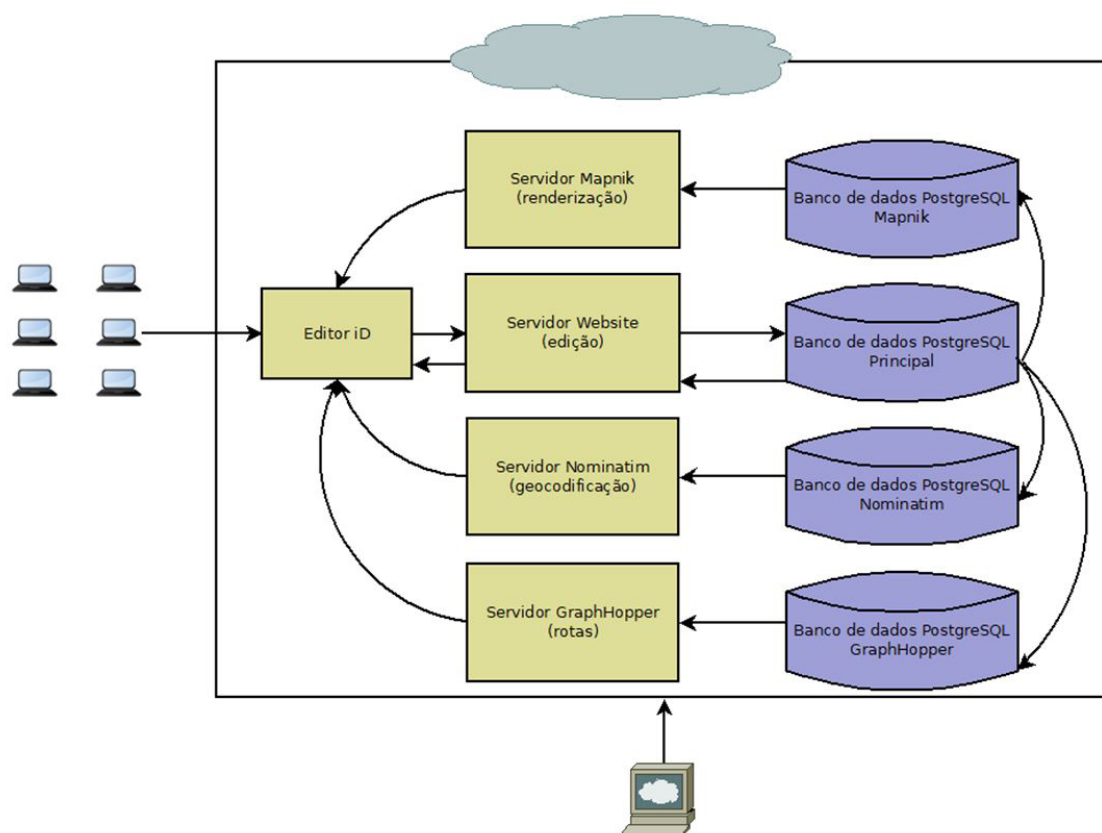


Fig. 2 – Infraestrutura dos servidores OpenStreetMap.

os elementos de gamificação da tríade PBL.

Os participantes do experimento acumulam os pontos ao criarem, editarem ou deletarem objetos no mapa, sendo concedido 1 ponto para cada node. Foram criadas medalhas virtuais para serem exibidas pelos usuários conforme suas contribuições no mapa ou execução de outras ações como postar comentários no sistema ou adicionar outros participantes como amigos no sistema. Por fim, foi implementada uma lista dos participantes em ordem decrescente de pontos exibindo as medalhas obtidas pelos participantes, servindo como a tabela de classificação.

Para medir a influência da gamificação na motivação dos militares, os participantes foram divididos em dois grupos: o grupo experimental e o grupo de controle. Os participantes do grupo experimental acessaram o sistema com os elementos de gamificação implementados, enquanto os participantes do grupo de controle acessaram o sistema sem esses elementos. Para isso foi implementada uma funcionalidade que, aleatoriamente,

inclui os participantes em um dos grupos no momento em que realizam o cadastro no sistema.

Diferentemente dos estudos supracitados, o sistema de mapeamento colaborativo desenvolvido não se baseia em um aplicativo de celular, mas em uma infraestrutura na qual o usuário se conecta por meio de um computador. É possível contribuir com o sistema acessando um navegador em um celular. Porém, como as funcionalidades de gamificação foram desenvolvidas para um computador, o participante pode não ter a experiência otimizada a partir de um celular.

A infraestrutura ilustrada na **figura 2** contém o editor iD, que é a interface com a qual os usuários interagem com o servidor principal, denominado Website, e editam os vetores do mapa. Os servidores secundários fornecem suporte às funcionalidades adicionais do mapa, sendo o *Mapnik* responsável pela renderização do mapa a ser visualizado na tela inicial. O *Nominatim* é responsável pela geocodificação e o *GraphHopper*, responsável por traçar as rotas.

Por dificuldades técnicas, não foi possível sincronizar os servidores secundários do experimento com o principal. Por essa razão, o servidor principal foi sincronizado com os servidores secundários oficiais do *OpenStreetMap*, de modo que a visualização da tela inicial do mapa, a geocodificação e as rotas são oriundas de dados do servidor oficial do *OpenStreetMap* e não do experimento. Porém, essa dificuldade não impediu a execução do experimento, pois apenas o servidor *Website* é utilizado para editar os dados.

Após o desenvolvimento do sistema, foi enviado um e-mail aos militares convidando-os a participar do experimento e mapear livremente os objetos no sistema. Alguns dias após o início do experimento, foram enviados e-mails aos participantes cadastrados, solicitando que realizem a aquisição de classes específicas, de modo a conduzir o mapeamento colaborativo além de relembrar os participantes do experimento.

A **tabela 3** exibe a data de envio dos e-mails e o grupo de destinatários.

Tab. 3 – Data dos e-mails enviados e destinatários.

Data do e-mail	Destinatários
19/03/2020	Todo o universo selecionado
31/03/2020	Militares cadastrados no sistema
30/04/2020	Militares cadastrados no sistema
15/06/2020	Todo o universo selecionado

Ao final do segundo mês, foi enviado um novo e-mail ao universo dos militares da ativa formados na AMAN e na EsSA. Solicitou-se tanto a participação no experimento, quanto a aquisição de classes específicas. O experimento teve duração de 3 meses.

Como o banco de dados utilizado pelo projeto *OpenStreetMap* é o *PostgreSQL*, é possível realizar consultas para obter diversas informações a respeito do experimento. A **figura 3** ilustra uma consulta na qual são exibidos os nomes dos usuários e as respectivas pontuações, além de filtrar apenas os usuários do grupo experimental e os pontos criados a partir de uma data específica.

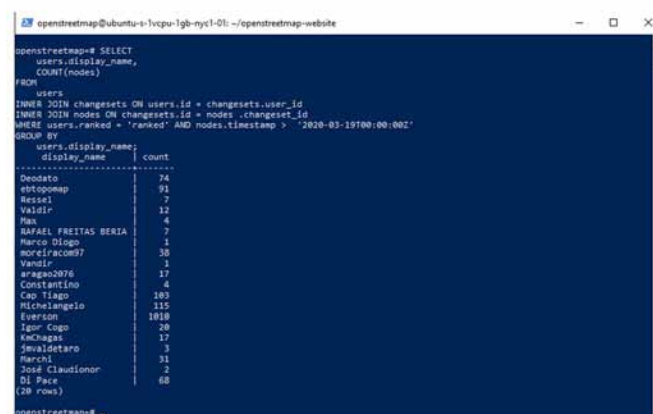


Fig. 3 – Consulta ao banco de dados do sistema durante o experimento.

Desse modo, foi possível obter os dados para a próxima seção.

4. Resultados obtidos

Foi observado, como esperado, o crescimento da curva de militares cadastrados no sistema. Isso se deu logo após as datas em que foram enviados os e-mails aos militares do universo selecionado, conforme ilustra a **figura 4**.

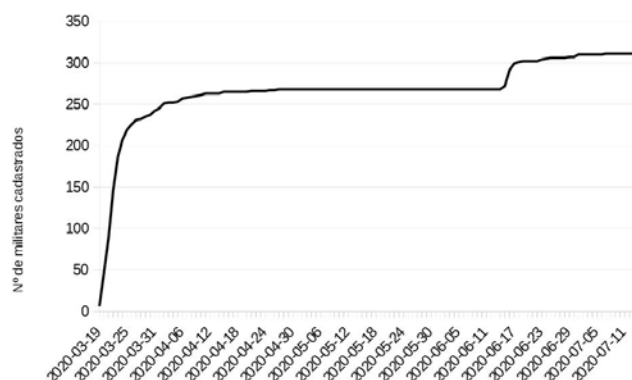


Fig. 4 – Quantidade de militares cadastrados no sistema.

Nem todos os participantes que se cadastraram inseriram informações geográficas no sistema de mapeamento colaborativo. Dos 311 participantes cadastrados, apenas 45 realizaram alguma colaboração no sistema, correspondendo a 0,13% do universo convidado.

A taxa de resposta de pesquisas por e-mail depende de diversos fatores, tais como quantidade de perguntas a serem respondidas, forma de elaboração das perguntas e direcionamento do público-alvo, apresentado uma taxa de resposta, em média, de 31% [20]. Sendo assim, já era esperada uma taxa de resposta pequena, pois foi solicitado aos participantes que realizassem quatro tarefas:

- Assistir ao vídeo explicativo do sistema enviado por e-mail com duração de 3 minutos;
- Realizar o cadastro no sistema, inserindo dados cadastrais como e-mail e nome de usuário;
- Confirmar o cadastro, clicando em um link enviado ao e-mail cadastrado;
- Acessar o sistema e inserir as informações geográficas.

Apesar de o sistema poder ser utilizado por pessoas que não tenham formação na área das geociências, é necessário alguma familiaridade com informática. Utilizar um sistema de mapeamento colaborativo é significativamente mais complexo do que responder a um questionário *online*. Além disso, muitos militares tentaram acessar o sistema através da rede corporativa do Exército (EBNET), a qual, por questões de segurança cibernética, bloqueava o acesso ao sistema e aos vídeos explicativos, a depender do nível de acesso do usuário na EBNET.

O início do experimento ocorreu em 19/03/2020, quando foi enviado o e-mail solicitando aos participantes que mapeassem livremente quaisquer feições visíveis na imagem. O experimento foi conduzido com outros estímulos via *e-mail*, solicitando que os militares realizassem contribuições em uma classe específica.

4.1 Trecho rodoviário

A classe *highway*, equivalente à classe 'trecho_rodoviário' da ET-ADGV, compreende as vias terrestres de quaisquer tipo, seja ela uma via primária, secundária, terciária, pavimentada ou não, incluindo ciclovias e vias de pedestres. Dentre todas as classes, ela obteve a maior quantidade de pontos mapeados pelos participantes.

Na segunda quinzena de abril, observou-se a

queda no interesse pelo mapeamento desta classe. Então, em 30/04/2020, foi enviado um estímulo aos participantes, solicitando que mapeassem elementos dessa classe, não surtindo efeito nos participantes. Em 15/06/2020, foi realizado um novo estímulo, solicitando mapeamento de objetos desta classe. Desta vez, o e-mail foi enviado para todo o universo inicialmente selecionado, o que surtiu o efeito desejado, conforme ilustrado na **figura 5**

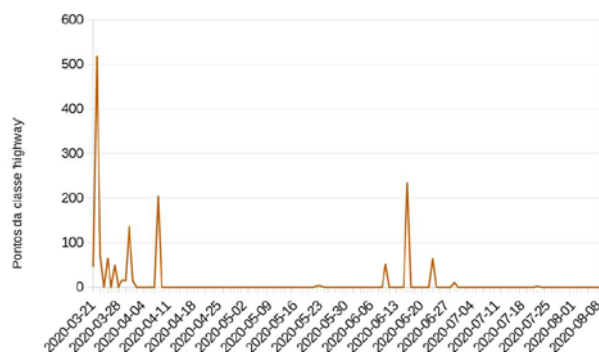


Fig. 5 – Quantidade de pontos da classe highway em função do dia.

4.2 Hidrografia

Observou-se uma peculiaridade do projeto *OpenStreetMap* ao se analisar os resultados da Hidrografia. A categoria Hidrografia da ET-EDGV contém diversas classes, tais como 'massa_dagua', 'trecho_drenagem', dentre outras. No *OpenStreetMap* não há uma divisão da categoria em subcategorias (ou classes). Portanto, foi necessário pesquisar na consulta SQL diversas classes as quais, juntas, são de certo modo equivalentes à categoria Hidrografia.

Além disso, o *OpenStreetMap* permite que o usuário crie uma classe no momento da aquisição caso ela não esteja disponível. Isso pode gerar dificuldades no processo de validação dos dados. Por outro lado, permite que o usuário realize a contribuição, mesmo não tendo certeza sobre qual classe o objeto pertença. No presente caso, foi verificada a criação da classe Igarapé ao adquirir a Hidrografia.

O estímulo solicitando a aquisição da Hidrografia foi enviado aos participantes em 31/03/2020, tendo sido

verificado um aumento significativo nas contribuições poucos dias após o estímulo, conforme ilustra a **figura 6**.

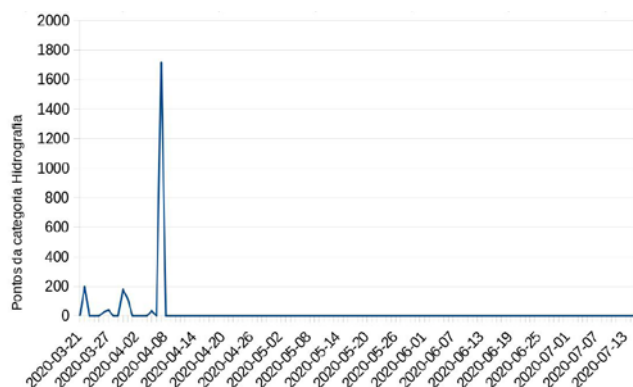


Fig. 6 – Quantidade de pontos da categoria Hidrografia em função do dia.

Verificou-se, portanto, que é possível realizar o mapeamento seletivo para atender campanhas da Força Terrestre com necessidades específicas. A intensidade com a qual uma determinada categoria foi mapeada aumentou significativamente após o estímulo enviado aos participantes.

4.3 Aquisição em geral

Como forma de medir a motivação dos militares, consideraram-se todas as classes disponíveis para mapeamento, inclusive as classes criadas pelos próprios participantes do experimento.

A **figura 7** ilustra a aquisição de objetos no sistema. Verifica-se que os estímulos foram essenciais para motivar a aquisição de feições no experimento.

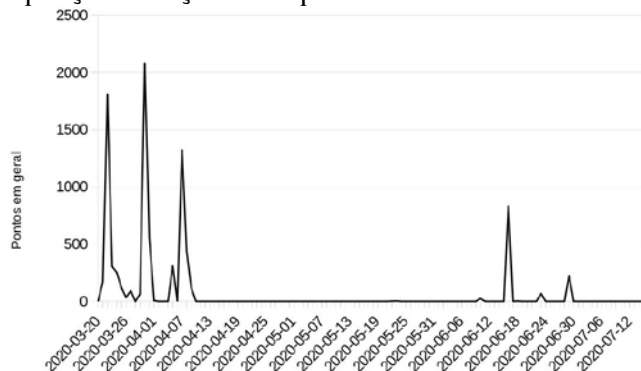


Fig. 7 – Quantidade de pontos adquiridos em função do dia.

Ao segregar o grupo experimental (com gamificação) do grupo de controle (sem gamificação), verificou-se que, no início do experimento, o grupo experimental realizou uma quantidade maior de contribuições. Porém, após o primeiro estímulo, a situação se inverteu.

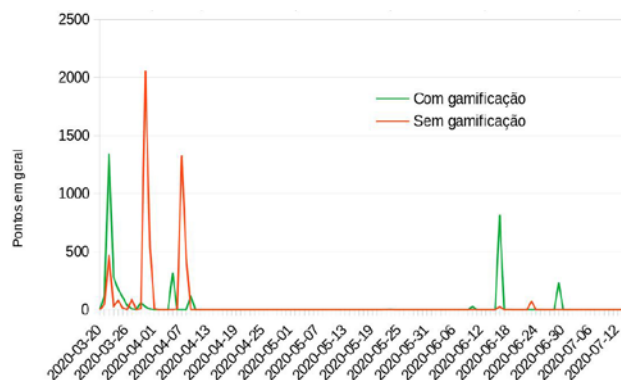


Fig. 8 – Quantidade de pontos adquiridos em função do dia nos grupos experimental e de controle.

O grupo experimental mapeou um total de 7.250 pontos, enquanto o grupo de controle mapeou 10.436, apresentando uma performance 44% superior. Os elementos de *design* de jogos escolhidos, na forma que foram implementados, não aparentou ser um fator motivador para os militares contribuírem com um sistema de mapeamento colaborativo. Cabe ressaltar que outras combinações de elementos de *design* de jogos podem apresentar resultados diferentes.

Há algumas hipóteses que podem ter contribuído para esse resultado:

- O sistema de mapeamento colaborativo restrito aos integrantes do Exército Brasileiro, por si só, já foi uma novidade. Talvez até mais interessante que a gamificação;
- Os militares já possuem naturalmente a motivação necessária para contribuir com um mapeamento colaborativo corporativo, conforme citado anteriormente;
- Os elementos de gamificação escolhidos podem não ter agradado ou não terem sido suficientes para motivar os militares;
- A hierarquia e disciplina presentes na caserna

fazem parte de um contexto diametralmente oposto ao contexto lúdico dos jogos, podendo não ser viável a gamificação em um ambiente militar.

4.4 Distribuição geográfica

A distribuição geográfica dos 45 militares participantes é ilustrada pela **figura 9**, a qual reflete a distribuição dos militares ao longo do país. O maior efetivo de militares está nas guarnições do Rio de Janeiro, Porto Alegre e Brasília [21].

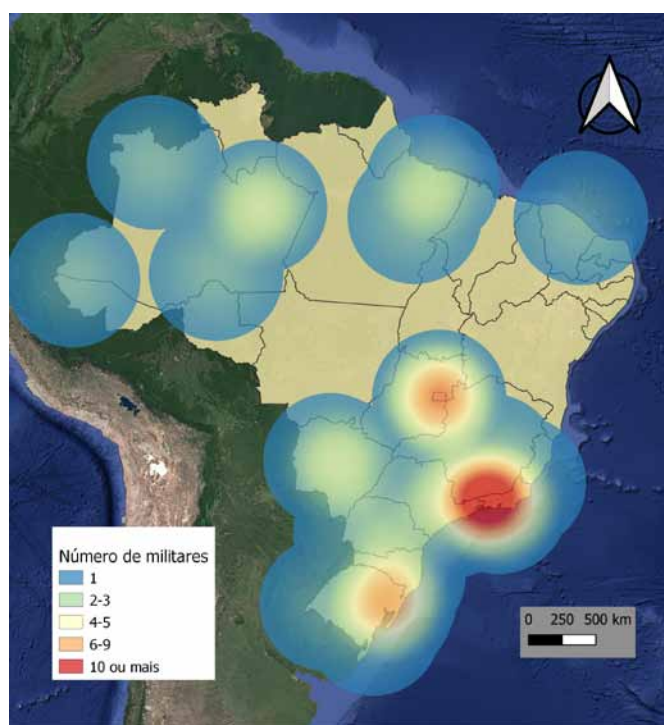


Fig. 9 – Distribuição dos militares que realizaram contribuições ao sistema.

Por outro lado, ao considerar a quantidade de pontos mapeados pelos militares, observa-se que a região norte se destacou. Desse modo, constata-se que na região norte participaram poucos militares, mas a quantidade de pontos mapeados foi relativamente grande, especialmente em regiões estratégicas como a Amazônia Legal e áreas próximas a fronteiras.

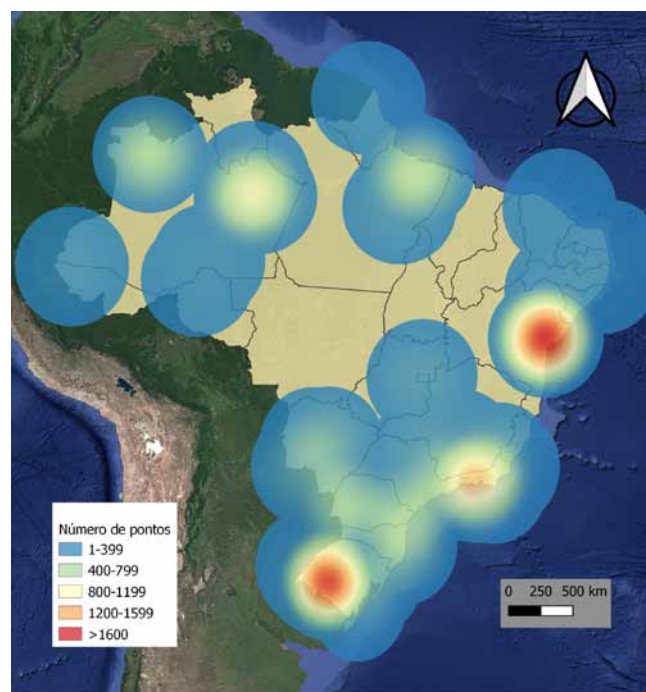


Fig. 10 – Distribuição dos militares que realizaram contribuições ao sistema considerando a quantidade de pontos como Peso.

Ao se analisar a faixa etária dos participantes, verificou-se que os participantes de todas as idades realizaram contribuições significativas. Considerou-se que a faixa etária é correspondente ao Posto ou Graduação do militar, pois não foi coletada a idade dos participantes no experimento. A **tabela 4** ilustra a quantidade de contribuições no mapa nos Postos e Graduações.

Tab. 4 – Quantidade de pontos mapeados por Postos e Graduações.

Posto/Graduação	Nº de militares	Nº de pontos mapeados
Gen e Cap QAO	4	1255
Cel e Ten QAO	7	1370
TC e STen	7	3539
Maj e 1º Sgt	8	1504
Cap e 2º Sgt	10	847
Ten, Asp e 3º Sgt	7	303
P/G não identificado	2	24

Ao examinar a contribuição individual dos participantes, verificou-se a ocorrência do Princípio de Pareto, o qual estabelece que 20% das causas são responsáveis por 80% dos resultados. No presente caso, 20% dos participantes foram responsáveis por

aproximadamente 80% das contribuições no mapa, conforme ilustra a **figura 11**.

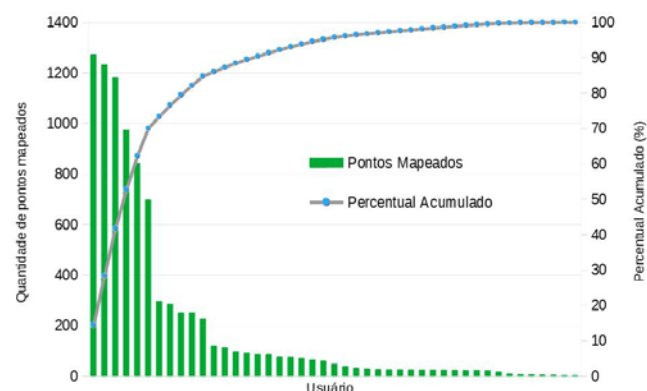


Fig. 11 – Diagrama de Pareto.

Ao observar o grupo dos 20% dos participantes que foram responsáveis por 80% das contribuições, verificou-se que, em sua maioria, são de faixas etárias mais elevadas, conforme ilustrado pela **tabela 5**. Desse modo, é possível suprir o gestor militar de informações que permitam concentrar seus esforços motivacionais em um público-alvo mais específico, otimizando o emprego dos recursos humanos e materiais.

Tab. 5 – Postos e Graduações que realizaram 80% das contribuições.

Posto/Graduação	Nº de militares
Gen e Cap QAO	2
Cel e Ten QAO	1
TC e STen	3
Maj e 1º Sgt	2
Cap e 2º Sgt	1
Ten, Asp e 3º Sgt	0
P/G não identificado	0

5. Conclusão

O presente trabalho teve por objetivo verificar a influência da gamificação em um sistema de mapeamento colaborativo restrito aos militares do Exército Brasileiro. Além disso, buscou entender a motivação dos participantes do experimento, observando diferentes características como faixa etária e distribuição geográfica.

Foi verificado que os elementos de *design* de jogos escolhidos não foram eficazes em aumentar a motivação dos militares em contribuir com um sistema de mapeamento colaborativo. O grupo que não experimentou a gamificação apresentou um desempenho 44% superior.

Uma possível explicação para o fenômeno observado é que o próprio sistema de mapeamento colaborativo no âmbito do Exército Brasileiro se mostrou uma novidade interessante com diversas aplicações possíveis, de modo que a gamificação inserida dentro do sistema ficou em segundo plano. Caso o experimento tivesse sido realizado em um momento no qual um sistema de mapeamento colaborativo já estivesse consolidado e em operação regular na Força Terrestre, a gamificação poderia ter sido notada com mais intensidade.

Verificou-se que a distribuição geográfica e faixa etária não enviesaram a análise. Os participantes estavam distribuídos de maneira homogênea em relação ao território nacional e à faixa etária. Foi verificado que os militares de faixas etárias mais elevadas realizaram mais contribuições que os militares mais novos.

O experimento realizado permitiu concluir que estímulos periódicos influenciam significativamente na motivação dos militares em contribuir com um mapeamento colaborativo.

Referências bibliográficas

- [1] BRASIL, Lista de Tarefas Funcionais, 2016.
- [2] BRITO, L. A. T. Mapas em Meio Analógico Para Emprego Militar: Necessidades e Possibilidades. Rio de Janeiro, RJ. Dissertação de Mestrado. Instituto Militar de Engenharia. Departamento de Engenharia Cartográfica, 134p. 2012.
- [3] SENARATNE, H., MOBASHERI, A., ALI, A. L., CRISTINA, C. A review of volunteered geographic information quality assessment methods. *International Journal of Geographical Information Science*. v. 31. pp. 139-167. 2017.
- [4] GOODCHILD, M. F. Citizens as sensors: the world of volunteered geography. *Geojournal* 69 (4), pp.211-221. 2007.
- [5] MENDONÇA JÚNIOR, M. G. Estratégias para geração semi-automática de modelos digitais do terreno visando seu emprego no mapeamento sistemático. Rio de Janeiro, RJ. Dissertação de Mestrado. Instituto Militar de Engenharia. Departamento de Engenharia Cartográfica, 155p. 2002.
- [6] PRANDI, C., VALENTINA, N., SALOMONI, P., NUNES, N. J. From gamification to pervasive game in mapping urban accessibility. *CHIItaly* pp 126-129. 2015.
- [7] BAJPAI, V., YADAV, D. K. Management and validation of VGI. 1st International Conference on Emerging Trends and Applications in Computer Science, ICETACS 2013. pp 57-61. 2013.
- [8] DE MENEZES, P. M. L., FERNANDES, M. C. Roteiro de Cartografia. Oficina de Textos. 2016.
- [9] MILLER, K., DECI, E., RYAN, R. Intrinsic Motivation and Self-Determination in Human Behavior. *Contemporary Sociolog.* 1988.
- [10] RYAN, R., DECI, E. Intrinsic and Extrinsic Motivations: Classic Definitions and New Directions. *Contemporary Educational Psychology*. vol 25 (1) pp. 54-67. 2000.
- [11] COMPOSTO, S., INGENSAND, J., NAPPEZ, M., ERTZ, O., RAPPO, D., BOVARD, R., WIDMER, I., JOOST, S. How to recruit and motivate users to utilize VGI-systems? *International Conference on Geographic Information Science*. pp. 2-5. 2016.
- [12] MARTELLA, R., KRAY, C., CLEMENTINI, E. A Gamification Framework for Volunteered Geographic Information. *Lecture Notes in Geoinformation and Cartography*. pp. 73-89. 2015.
- [13] DETERDING, S., DIXON, D., KHALED, R., NACKE, L. From Game Design Elements to Gamefulness: Defining "Gamification". *International Academic MindTrek Conference: Envisioning Future Media Environments*. pp. 9-15. 2011.
- [14] WERBACH, K., HUNTER, D. For the Win: How Game Thinking Can Revolutionize Your Business. 144p. 2012.
- [15] EL-KHUFFASH, A. Gamification. Ryerson University, Toronto, Canada. 40p. 2013.
- [16] COLEMAN, D., GEORGIADOU, Y., LABONTE, J. Volunteered Geographic Information: The Nature and Motivation of Producers. *International Journal of Spatial Data Infrastructures Research*. vol. 4. pp. 332-358. 2009.
- [17] MARTI, I., RODRIGUEZ, L., BENEDITO, M., TRILLES, S., BELTRAN, A., DIAZ, L., HUERTA, J. Mobile Application for Noise Pollution Monitoring through Gamification Techniques. *International Conference on Entertainment Computing (ICEC)*. pp. 562-571. 2012.
- [18] BRITO, L. A. T. O Uso da Inteligência Coletiva do EB na Cartografia de Interesse para a Força Terrestre. Rio de Janeiro, RJ. Dissertação de Mestrado. Instituto Militar de Engenharia. Departamento de Engenharia de Defesa, 129p. 2017.
- [19] BRASIL. Especificações Técnicas para Aquisição de Dados Geoespaciais Vetoriais. v. 3.0. 2020.
- [20] SHEEHAN, K. B. E-mail Survey Response Rates: A Review. *Journal of Computer-Mediated Communication*. Vol. 6 (2). 2006.
- [21] BRASIL. O Exército em Números. 2020.

Um algoritmo de aprendizado iterativo baseado no ciclo de marcha humana para a locomoção de um robô humanoide

Fábio Suim Chagas^a, Luis David Peregrino de Farias^b, Matheus Bozza^b, Paulo Fernando Ferreira Rosa^a

^aInstituto Militar de Engenharia (IME),

Praça General Tibúrcio, 80, 22290-270,

Praia Vermelha, Rio de Janeiro, RJ, Brasil.

^bCentro Tecnológico do Exército (CTEx)

Av. das Américas - Guaratiba, Rio de Janeiro - RJ, 23032-050

*fsuim@yahoo.com.br

RESUMO: Este artigo apresenta um algoritmo de aprendizado iterativo que prioriza a estabilidade estática em vez da estabilidade dinâmica, com a finalidade de diminuir a complexidade da marcha de um robô humanoide. Inicialmente, há um conjunto de posições almejadas que o algoritmo deve atingir durante o ciclo de marcha. Um processo de tentativa e erro conduz o método de aprendizado. Cada tentativa de movimento do robô fica disponível em uma lista de ações. Sempre que o robô alcança um objetivo, o algoritmo armazena a sequência de movimentos em uma memória. Se há uma falha, a lista de ações fornece a posição anterior à queda – e o processo recomeça a partir deste ponto. Com o objetivo de testar o algoritmo, foi desenvolvido um simulador usando o Matlab Simulink, juntamente com a biblioteca de forças de contato do Simscape Multibody. Os dados da simulação são apresentados através de gráficos que descrevem o comportamento das articulações durante o processo de aprendizado para um ciclo de marcha.

PALAVRAS-CHAVE: Robô Humanoide. Algoritmo de Aprendizagem. Cinemática. Simulador Humanoide.

ABSTRACT: This paper shows an iterative learning algorithm that prioritizes static over dynamic stability to decrease gait complexity for an articulated humanoid robot. Initially, we have an array of goal positions, which the algorithm must achieve during the gait cycle. A trial and error process leads to the learning approach. Each robot motion attempt is available on an action list. At the moment the robot achieves a goal, the algorithm stores the sequence of movements in memory. When a failure happens, the action list provides the position just before the fall - and the process starts from that point onwards. In order to test the algorithm, we developed a simulator using Matlab Simulink, together with the Simscape Multibody contact forces library. We present the simulation data through graphs that describe the behavior of the joints during the learning process of a gait cycle.

KEYWORDS: Humanoid robot. Learning Algorithm. Kinematic. Humanoid Simulator.

1. Introdução

Desde que o primeiro robô humanoide [1] foi apresentado pelo laboratório do professor Kato, na Universidade de Waseda nos anos 1970, o número de pesquisas relacionadas ao assunto cresce a cada dia. Nos dias de hoje, plataformas cada vez mais elaboradas e amigáveis estão sendo desenvolvidas em uma escala bem acima daquela década [2]-[5]. Isto pode ser explicado pela evolução e queda nos preços dos circuitos integrados (CIs) e outros componentes como os atuadores, além do fato do uso amplamente difundido da impressão 3D a um baixo custo.

Com relação ao uso de humanoides em aplicações militares, eles poderiam ser úteis em tarefas de resgate e desarmamento de bombas [6]-[7]. A locomoção similar aos seres humanos permitiria a passagem por terrenos irregulares como os rochosos ou montanhosos, substituindo robôs dotados de rodas ou esteiras. Em ambientes urbanos, poderiam se locomover por ruelas, subir escadas, escalar edifícios durante uma perseguição, agir contra um oponente de forma não letal, visto que não necessita preservar sua própria vida, como no caso de um soldado humano.

Um fator que ainda apresenta desafios no que diz respeito aos humanoides é a locomoção. Andar

com equilíbrio, ritmo e eficiência energética não é uma tarefa fácil [8]-[9]. Por mais de três décadas, esse assunto atraiu a atenção de pesquisadores das universidades e dos institutos mais prestigiados ao redor do mundo [10]-[12].

Até o momento, não há um modelo matemático completo que represente a marcha humana, mas formas alternativas de resolução do problema da locomoção são abordadas por modelos e algoritmos heurísticos, tais como: o pêndulo linear invertido [13] e o pêndulo linear invertido 3D [14]; os quais são de baixa ordem, e adicionalmente, quando combinadas com outras técnicas como o Zero Moment Point (ZMP) [15]-[16] melhoram substancialmente a solução do problema. Apesar da evolução nessa área ao longo dos anos, a dinâmica da marcha humana ainda demanda desenvolvimento. Locomoção com sincronismo, equilíbrio e economia de energia é ainda um campo a ser explorado.

O principal objetivo deste artigo é apresentar um algoritmo de aprendizagem para o problema do caminhar em robôs bípedes baseado na marcha humana. A principal vantagem está no fato da solução ser iterativa o que simplifica a obtenção das equações cinemáticas da marcha.

O artigo está organizado da seguinte forma: a Seção 2 apresenta uma revisão de literatura; a Seção 3 mostra o plano de fundo teórico; a Seção 4 mostra o algoritmo proposto; a Seção 5 mostra os experimentos e resultados; e finalmente, a Seção 6 apresenta a conclusão e os próximos passos.

2. Revisão de literatura

No estudo da locomoção robótica, a marcha pode ser classificada como estaticamente e dinamicamente estável. No caminhar estático, a projeção do centro de massa do robô está na região do pé: se o robô se movimenta lentamente para garantir isso, o caminhar é sempre estável. O problema é o alto consumo de energia e perda dos movimentos dinâmicos [17][18].

No caminhar dinâmico, o centro de massa do robô assume a posição fora do ponto de suporte e não é, portanto, dinamicamente estável. McGeer [19]

mostrou que certos sistemas mecânicos possuem uma tendência natural para caminhar e chamou isso de caminhada passiva dinâmica. Nesta caminhada, o dispositivo usa as forças gravitacionais para se mover usando uma pequena inclinação [21], [22]. Como resultado, possui as vantagens da eficiência energética e de dinamismo do movimento. Em contrapartida, ela se restringe ao plano sagital e não obterá sucesso sobre um piso desnivelado.

Outro fator preponderante na marcha dinamicamente passiva é a alta não-linearidade do modelo, o que dificulta o projeto dos robôs caminhantes. A observação biológica dos seres vivos, juntamente com ajustes por experimentação, auxilia a modelagem da tarefa de caminhar, como mostrado em [22]-[23]. Devido a não-linearidade da marcha e a dificuldade de obter um modelo, aproximações por algoritmos heurísticos são desejáveis na solução do problema.

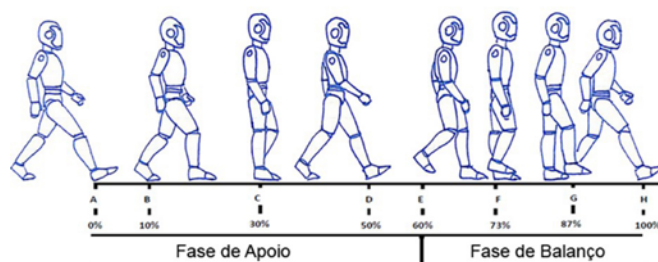


Fig. 1 - Ciclo de marcha (Adaptado de Magee [27]).

Nesse cenário, o aprendizado por reforço emerge como uma opção para resolver o problema. Pesquisas, como as feitas por Yamaguchi *et al* [24], investigaram o algoritmo Q-Learning para o controle de um robô 50 GDL. Eles compararam uma abordagem por aprendizado por reforço hierárquico e uma função aproximada. Como resultado, eles obtiveram um aprendizado apropriado em um intervalo de 20 a 30 episódios. Kulk e Welsh [25] propuseram um algoritmo baseado em funções que se opõem e em funções redundantes de adequação, focando em três componentes principais: velocidade, eficiência e estabilidade. O algoritmo demonstrou uma melhora na velocidade e eficiência para realizar o ciclo de marcha aprendido.

Outros fatores preponderantes são os simuladores. O desenvolvimento do aprendizado por reforço baseado no algoritmo de locomoção humanoide exige uma gama de experimentações e os simuladores aceleram o resultado. Para muitas situações, o algoritmo é incompatível com o hardware. Em Oliveira *et al* [26], há uma tentativa de criar uma função para transferir rotinas de um ambiente simulado para o robô real. Infelizmente, algumas funções simuladas não funcionam no sistema real e ainda precisam de melhorias.

Neste artigo, é proposto um algoritmo iterativo de inteligência de máquina, o qual não requer o uso de um modelo matemático de ordem superior. Foi adotada a estratégia de aprendizagem observacional. Um conjunto de pontos sobre o corpo humano, formado por posições angulares capturadas durante a marcha, compõe uma matriz de objetivos que o robô deve alcançar.

3. Tópico tutorial

Esta sessão descreve brevemente o ciclo da marcha humana e o comportamento da mesma nos planos sagital, frontal e transversal.

3.1 O ciclo de marcha

O ciclo de marcha é uma sequência de posições que o corpo toma para se mover de um ponto a outro. Em um caminhar normal, isso ocorre suave e ciclicamente. O começo do processo ocorre quando o calcanhar da perna que está à frente toca o chão. A sequência termina quando o mesmo calcanhar, que começou o ciclo, toca o solo novamente.

O ciclo ocorre em duas fases, sete períodos e oito eventos. Duas fases que dividem o ciclo de marcha são a fase de apoio (ou suporte) e a fase de oscilação (ou balanço), sendo que a fase de oscilação é mais rápida que as anteriores. A fase de suporte inclui os eventos de contato inicial e calcanhar livre, além das fases de apoio médio e de apoio final. O contato inicial oposto acontece em duas fases de transição. O evento dedos dos pés livres, pés adjacentes, tibia vertical, e próximo contato inicial estão na fase de balanço. Pré-balanço, balanço inicial, balanço médio e balanço final são os períodos da fase de balanço, conforme a **figura 1**.

Inicialmente, o movimento começa com o contato do

calcanhar com o solo (**figura 1 A**). Na sequência, o pé fica estável no solo para receber a carga de resposta do peso e corresponde a do ciclo (**figura 1 B**). Quando o calcanhar oposto ao que começou o movimento está fora do solo, atinge-se do ciclo e no evento de calcanhar livre (**figura 1 C**). A perna oposta se move para frente, e o calcanhar oposto toca o chão, do ciclo de marcha é alcançado. O evento correspondente é o contato inicial oposto (**figura 1 D**). Daqui em diante, é iniciada a fase de pré-balanço, quando o calcanhar da perna que iniciou o ciclo está fora do solo é atingido do ciclo (**figura 1 E**). Quando os pés estão adjacentes o ciclo, encontra-se em (**figura 1 F**) e quando a tibia está vertical (**figura 1 G**) e o movimento desacelera. O que determina o fim do ciclo é o início de um novo contato do calcanhar da perna que iniciou o movimento, (**figura 1 H**). Vale a pena mencionar que a fase de balanço é mais rápida que a fase de apoio, e dura de todo o ciclo.

4. Algoritmo de aprendizado

Durante o ciclo de marcha as articulações assumem várias posições angulares e isso pode ser utilizado para construir a matriz de objetivo do algoritmo. As posições acessadas por captura são: os tornozelos direito e esquerdo, o quadril no início do fêmur, direito e esquerdo; o centro de massa e os ombros direito e esquerdo. A **figura 2** ilustra as articulações a serem analisadas, onde as letras de a, b, c, d, e, f, g, h e i indicam a posição dos sensores no corpo.

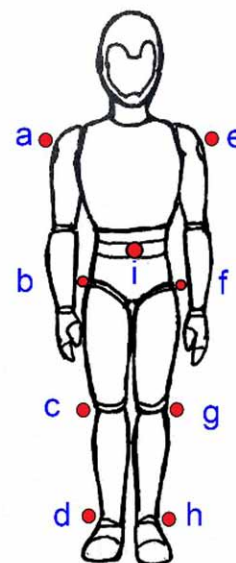


Fig. 2 - Distribuição dos sensores no corpo humano.

Tab. 1 – Descritores analisados durante o ciclo de marcha.

	Tornozelo		Joelho		Quadril		CG	Ombro	
	Esquerdo	Direito	Esquerdo	Direito	Esquerdo	Direito		Esquerdo	Direito
Evento A	$al(\theta_x, \theta_y, \theta_z)$	$ar(\theta_x, \theta_y, \theta_z)$	$kl(\theta_x, \theta_y, \theta_z)$	$kr(\theta_x, \theta_y, \theta_z)$	$hl(\theta_x, \theta_y, \theta_z)$	$hr(\theta_x, \theta_y, \theta_z)$	$c(\theta_x, \theta_y, \theta_z)$	$sl(\theta_x, \theta_y, \theta_z)$	$sr(\theta_x, \theta_y, \theta_z)$
Evento B									
Evento C
Evento D
Evento E
Evento F									
Evento G									
Evento H	$al(\theta_x, \theta_y, \theta_z)$	$ar(\theta_x, \theta_y, \theta_z)$	$kl(\theta_x, \theta_y, \theta_z)$	$kr(\theta_x, \theta_y, \theta_z)$	$hl(\theta_x, \theta_y, \theta_z)$	$hr(\theta_x, \theta_y, \theta_z)$	$c(\theta_x, \theta_y, \theta_z)$	$sl(\theta_x, \theta_y, \theta_z)$	$sr(\theta_x, \theta_y, \theta_z)$

A **tabela 1** apresenta os descritores analisados durante o ciclo de marcha. A sequência é dividida em oito eventos que vão de A até H. Em cada um dos quais é possível obter os valores dos ângulos θ_x , θ_y e θ_z para cada uma das articulações como mostrado na **figura 1**.

A nomenclatura *al* é para o tornozelo esquerdo, *ar* é para o tornozelo direito, *kl* é para o joelho esquerdo, *kr* é para o joelho direito, *c* é para o centro de massa (CG), *sl* ombro esquerdo e *sr* ombro direito. A terna $(\theta_x, \theta_y, \theta_z)$ representa a variação angular nos eixos *x*, *y* e *z*. O conjunto nomeado como *q* representa as articulações *al*, *ar*, *kl*, *kr*, *c*, *hl*, *hr*, *sl*, *sr*.

Seja $[\psi]_{t_i}$ o conjunto de todas as combinações angulares de todas as articulações *q*, mostrado em .

$$[\psi]_{t_i} = \{q\}_{t_i} \quad (1)$$

Durante o ciclo de caminhada obteve-se uma matriz ξ (2), na qual as colunas representam as variações angulares das articulações no instante. As linhas representam os eventos, períodos e fases do ciclo de caminhada.

$$\xi = \{[\psi]_{t_1}, [\psi]_{t_2}, \dots, [\psi]_{t_{n-1}}, [\psi]_{t_n}\} \quad (2)$$

Durante um evento, é possível obter tantos conjuntos ξ quanto o sistema de captura é capaz de processar, fazendo o número de conjuntos ξ estar diretamente associado ao poder computacional

disponível.

É importante perceber que nem todos os estados alcançados são válidos. Portanto, é necessário encontrar um subconjunto de posições dentro de ξ que leva às posições de equilíbrio. Nesse caso, busca-se por um subconjunto como descrito na **equação 3**.

$$\exists \Lambda \in \xi, \Lambda = (\lambda_1, \lambda_2, \dots, \lambda_{n-1}, \lambda_n) \quad (3)$$

onde, Λ é o subconjunto de estados estáveis de ξ e esses estados são representados por λ .

As posições λ_i são responsáveis por suprir os valores da matriz de objetivo *G* do algoritmo . Seja *n*(G) o número de linhas da matriz de objetivos e *n*(Λ) o número de estados válidos ou posições de equilíbrio que o corpo assume durante o ciclo de marcha, onde $n(G) \leq n(\Lambda)$.

4.1 Fluxograma do algoritmo de aprendizado

A **figura 3** mostra o fluxograma do algoritmo proposto. Inicialmente, é definido o tamanho da passada e a discretização angular para atingir o objetivo. O bloco 1 mostra a matriz objetivo *G* com as posições desejadas. *G* é composta pela seleção de algumas posições das fases do ciclo de marcha.

No Bloco II, a função recebe o primeiro objetivo *e*, a partir desse ponto, o sistema cria uma lista de ações. Depois, o programa checa se o valor coincide com o último valor de . Se o teste falhar, o programa executa o próximo passo.

 Algoritmo 1: Pseudocódigo para o algoritmo de aprendizagem

 1 Aprendizagem-por-iteração (q ; G ; inc ; deg)

Dados:
Entradas: Número de ciclos, tamanho do passo,
 q : Conjunto de juntas; G : Matriz de objetivos; inc : tempo gasto nas interações; deg : verifica se q alcançou objetivo;

Saída: M é a memória do robô.

2 Inicialização;

 3 **repita**

 4 Pega o objetivo ($Get_Goal(G(i))$);

5 Coloca a informação na Lista de Ações;

 6 **Enquanto** $l < deg$ **Faça**

 7 **Para** $i = 2$ até o número de juntas do robô **Faça**

 8 **Se** $q(i) < G(i)$ **Então**

 9 q é decrementado por (5);

 10 $checkforfall()$;

 11 **Se** $q(i) > G(i)$ **Então**

 12 q é incrementado por (6);

 13 $checkforfall()$;

 14 **Senão**

 15 $M = q$;

 16 **Fim**

 17 **Fim**

 18 **Até** G seja terminal;

Procedimento $checkforfall()$

Saída: Lista de Ações atualizada

 1 **Se** Condição de Equilíbrio Falhou **Então**

 2 $q \leftarrow q(i-1)$;

 3 $M \leftarrow q$;

4 Termina a Aplicação;

 5 **Senão**

6 Atualiza a lista de Ações;

 7 **Fim**

O valor deg é o número de vezes que o algoritmo executa o teste para verificar que o valor de q alcançou o valor alvo. Se o valor de $q(i)$ excede o valor de $G(i)$, isto é, o valor calculado excede o valor alvo, o valor de $q(i)$ é decrementado obedecendo à regra vista em (5). Por outro lado, se o valor de $q(i) < G(i)$, então o valor de $q(i)$ é incrementado pela regra dada por (6). A função aleatória k garante que os valores de q não sejam repetidos no evento de uma queda. Ela pode assumir o valor 0 ou 1.

Inc (4) é um conjunto contendo o tempo que cada articulação $q(i)$ vai permanecer durante sua atualização. σ_i , em uma situação real, é a frequência de operação do atuador na articulação do robô.

$$inc = \{\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_{n(G)}\} \quad (4)$$

$$q(i) = q - \left[\left(\frac{|G - inc|}{deg} \right) * k \right] \quad (5)$$

$$q(i) = q + \left[\left(\frac{|G - inc|}{deg} \right) * k \right] \quad (6)$$

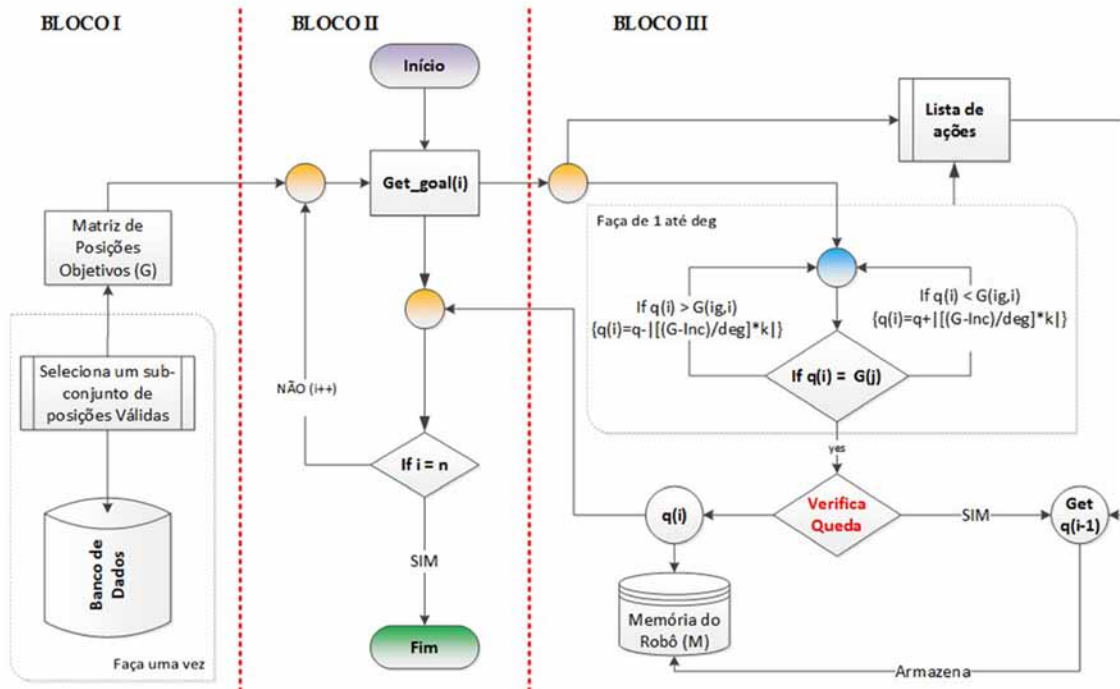


Fig. 3 - Fluxograma do algoritmo de aprendizado.

Durante esse processo, o algoritmo checa se o robô cai. Se o alvo é atingido, a matriz M armazena os valores de q e a função *Get_goal* recebe o próximo objetivo e repete o processo.

Se o objetivo não for alcançado, o algoritmo recebe o valor de $q(i - 1)$ da lista de ações e o armazena na matriz M . Então ele tenta alcançar o objetivo a partir do ponto de queda, repetidamente até alcançar sucesso. Após atingir o objetivo, a matriz M é atualizada. Quando o objetivo final é alcançado, o processo para e o aprendizado para um ciclo de caminhada está completo.

5. Experimentos e resultados

Com o objetivo de testar o algoritmo, fez-se necessário o desenvolvimento de um Dispositivo de Captura de Dados de Movimento Humano (DCMH) para montar a matriz objetivo e um simulador. O DCMH consiste de um conjunto de nove sensores inerciais de seis eixos (MPU-6050), conectados a uma placa Raspberry pi 3.

Foram medidos os ângulos dos tornozelos, esquerdo e direito no plano sagital (lateral) e frontal. Os ângulos dos joelhos, esquerdo e direito. Os ângulos, esquerdo e direito, dos quadris nos planos sagital, frontal e Centro de Gravidade. A **figura 4** mostra a configuração feita para o experimento com o propósito de coletar dados durante o ciclo de marcha.



Fig. 4 – Esquerda marcadores da posição dos sensores; direita dispositivo montado no corpo humano.

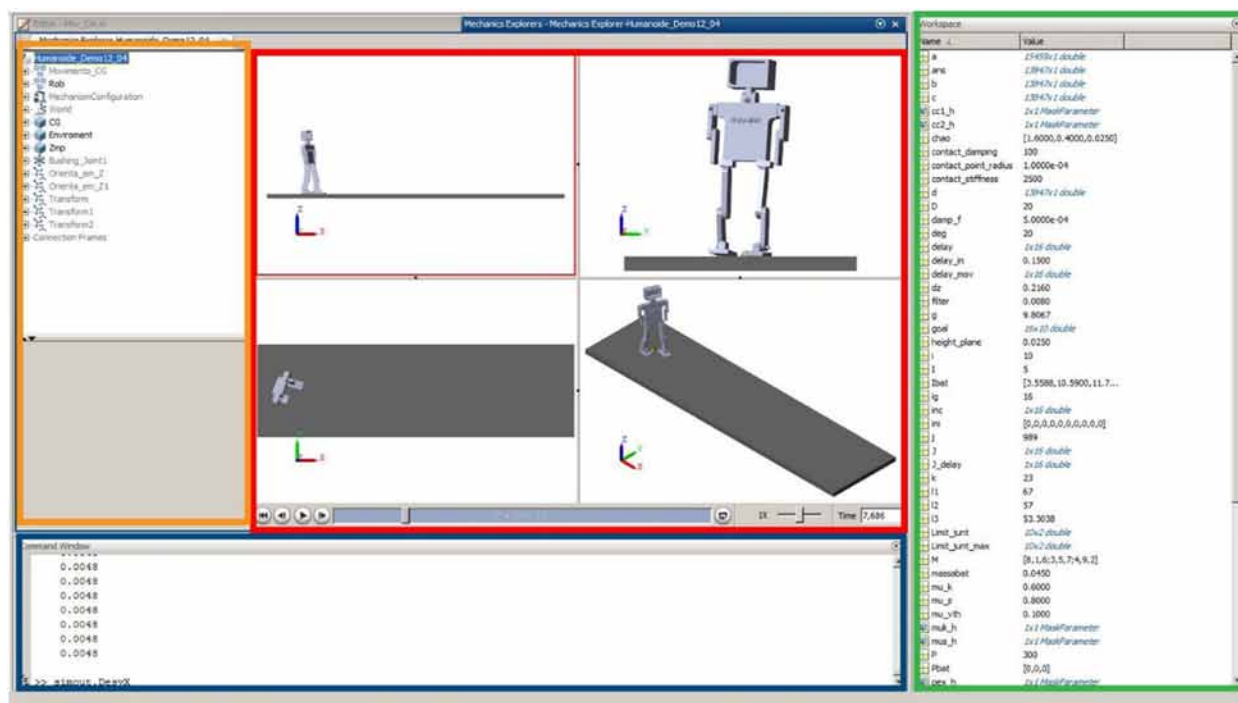


Fig. 5 - A interface gráfica do simulador: O retângulo laranja indica os objetos do robô. O retângulo vermelho mostra as vistas Lateral, Frontal, Superior e Isométrica. O retângulo verde disponibiliza as informações variáveis após a execução do simulador e o retângulo azul mostra a janela de comandos.

5.1 Simulador

A **figura 5** mostra o simulador desenvolvido usando o *Simulink* na versão 2018 do *MatLab* e a biblioteca de forças de contato do *Simscape Multibody*. Para modelar o robô, foi usado o *SolidWorks* 2017. A simulação exigiu um processador Intel(R) Xeon(R) CPU E7- 4870, com *clock* de 2.4 GHz, 32 núcleos, 32 GB de RAM, e uma placa de vídeo Cirrus Logic GD 5446. A **tabela 2** mostra a disposição das juntas do robô.

Tab. 2 – Disposição das juntas no corpo do robô

Membro	Graus de liberdade	Movimento
Pé direito	2	Flexão/dordiflexão, inversão/eversão.
Pé esquerdo	2	Flexão/dordiflexão, inversão/eversão.
Joelho direito	1	Flexão, extensão.
Joelho esquerdo	1	Flexão, extensão.
Quadril direito	2	Flexão, rotação.
Quadril esquerdo	2	Flexão, rotação.
Ombro direito	2	Flexão e rotação externa.
Ombro esquerdo	2	Flexão e rotação externa.
Cotovelo direito	1	Flexão do antebraço e extensão
Cotovelo esquerdo	1	Flexão do antebraço e extensão
Pescoço esquerdo	1	Rotação esquerda e direita.

Para a simulação, foram analisados somente os ângulos dos membros inferiores. Os dados foram colhidos de um modelo humano e adaptados para gerar a matriz objetivo para a caminhada do robô. A **tabela 3** mostra a matriz objetivo. Ao todo são 16 objetivos a serem cumpridos ao longo de um passo. Os valores negativos para o pé significam uma inversão, já os positivos uma eversão; para o tornozelo, valores negativos representam uma flexão plantar e os positivos uma dorsiflexão. A flexão do joelho é sinalizada por um valor negativo, já a extensão por um valor positivo; para a coxa valores negativos significam flexão e os positivos uma extensão. Valores positivos e negativos para o quadril representam uma rotação interna e externa respectivamente.

Tab. 3 – Matriz de objetivos – entrada de dados para o algoritmo

Eventos	Lado direito					Lado esquerdo				
	Pé	Tornozelo	Joelho	Coxa	Quadril	Pé	Tornozelo	Joelho	Coxa	Quadril
1	-11	0	0	-16	0	13	12	0	-16	0
2	-11	0	6	4	0	2	25	0	-32	-4
3	-8	-2	6	-16	8	9	19	6	-30	-8
4	0	-10	4	8	8	0	10	4	-8	-8
5	0	-13	4	-2	8	0	7	4	-18	-8
6	13	0	0	-4	-6	-12	10	4	-18	6
7	13	8	0	-20	-6	-12	10	4	-34	6
8	13	12	0	-16	0	-12	0	0	-16	0
9	2	25	0	-32	-4	-11	0	6	4	0
10	9	19	6	-30	-8	-8	-2	6	-16	8
11	0	8	4	-8	-8	0	-8	4	8	8
12	0	6	4	-18	-8	0	-10	4	-2	8
13	-12	10	4	-18	6	13	0	0	-4	-6
14	-12	10	4	-34	6	13	8	0	-20	-6
15	-12	0	0	-16	0	13	2	0	-16	0
16	0	0	0	0	0	0	0	0	0	0

5.2 Resultados

A **figura 6** mostra o resultado obtido durante o treinamento do robô para um ciclo de marcha, no qual foram selecionados 16 objetivos. Os eixos representam a informação do tempo de viagem e a variação angular das articulações, respectivamente. As linhas laranja e azul representam as articulações nos lados direito e esquerdo do humanoide. As linhas verde, violeta e preta indicam a variação do centro de massa nos eixos.

Inicialmente o robô está com as duas pernas paralelas e inicia o movimento com a perna direita à frente. As linhas azuis e laranjas representam os membros inferiores do lado direito e esquerdo respectivamente. A **figura 6 (a)** mostra o comportamento dos tornozelos no plano sagital. No início do movimento, ambos os tornozelos sofrem uma variação, devido ao ajuste de todo o corpo do robô. Quando ambos estão paralelos eles estão na posição 0°. A **figura 6 (b)** mostra os movimentos de inversão e eversão sofridas durante a caminhada.

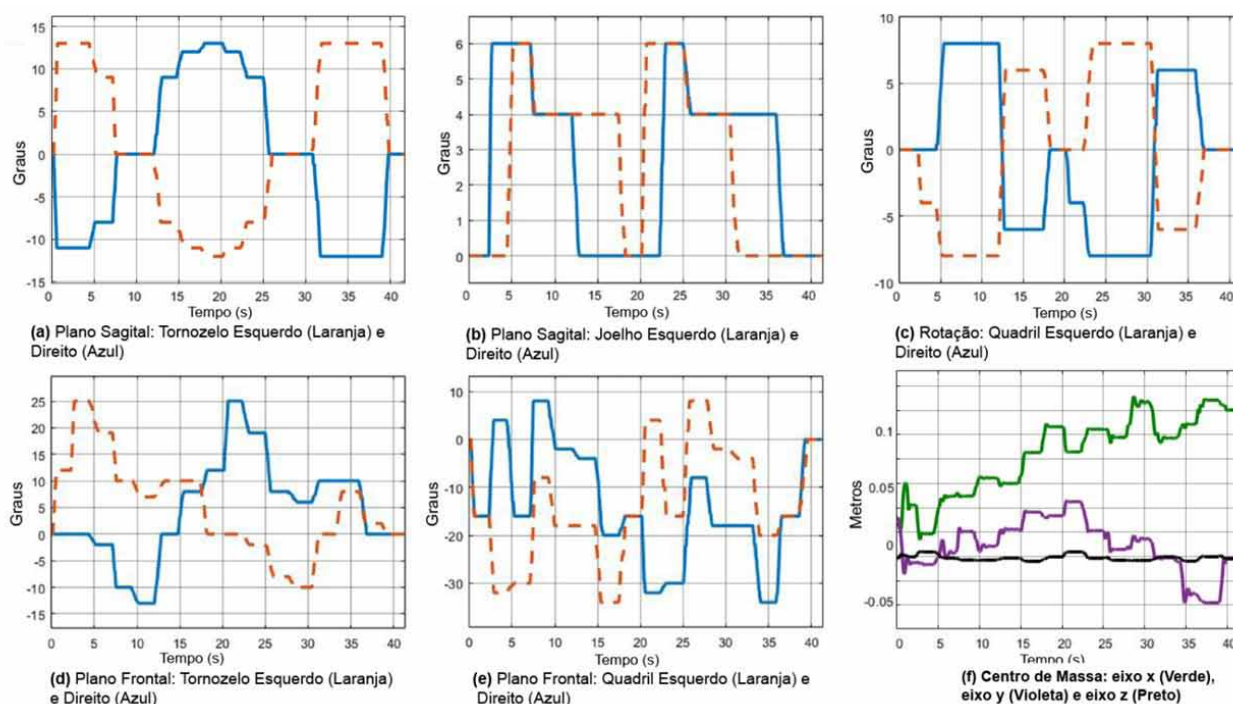


Fig. 6 - Representação do comportamento das articulações durante o processo de aprendizado.

A **figura 6 (c)** mostra o comportamento do quadril no plano sagital. A amplitude máxima atingida é de 8° para ambos os lados, o comportamento dos lados direito e esquerdo são espelhados em quase todo ciclo. Já a **figura 6 (d)** mostra a análise do quadril no plano frontal, inicialmente as duas curvas variam de 0° a -15° isto indica o momento de ajuste que o robô fez no início do ciclo de caminhada; no decorrer do percurso as curvas oscilam de forma espelhada, e no final da caminhada retornam a posição zero.

A **figura 6 (e)** mostra a análise dos joelhos. Inicialmente, eles estão paralelos na posição zero, à amplitude máxima atingida são 6 graus, as curvas apresentam um comportamento simétrico. O segmento de curva que o joelho direito faz no intervalo de 4 a 15 segundos é idêntico ao movimento do joelho esquerdo no intervalo de 20 a 35 segundos.

A **figura 6 (f)** diz respeito ao comportamento do Centro de Massa do robô nos eixos x, y e z que são representados pelas linhas verde, violeta e preta

respectivamente. As variações sofridas pelo centro de massa no eixo z o robô são sutis. No eixo y as variações máximas são de aproximadamente 0,04 metros para ambos os lados, isso acontece devido à necessidade de compensar o movimento de rotação do quadril. No eixo x o gráfico apresenta um desvio de 0,15 m do objetivo final que era caminhar em linha reta.

6. Conclusão

Neste artigo foi apresentada uma solução para o problema da caminhada baseada em um método iterativo. O algoritmo se inspira no andar humano, em que determinados pontos das posições de equilíbrio estático do ciclo de marcha, norteiam a construção de uma matriz de objetivos, os quais o robô deve atingir.

Inicialmente, desenvolveu-se um dispositivo de captura de movimentos do corpo humano (DCMH) através de sensores de inerciais de medidas, esses sensores foram dispostos na parte inferior dos membros para capturar as informações dos ângulos

articulares dos tornozelos, joelhos e quadril. Inspirados nas curvas obtidas com o DCMH, foi possível estimar posições para preencher uma matriz de objetivos, onde as colunas da matriz representam os valores angulares das juntas e as linhas a posição no ciclo de marcha.

O algoritmo segue a premissa de que há pontos que levam o robô a uma posição de equilíbrio durante o ciclo de marcha. Esses pontos são dispostos na matriz de objetivos. O algoritmo parte da posição inicial onde o robô está na posição ereta e com os pés paralelos; o valor da posição de referência das juntas é zero. A seguir, uma função pega a primeira posição da matriz e através de um processo incremental tenta chegar na posição alvo. Durante o processo pode haver quedas; caso aconteça, o robô armazena a última posição de sucesso e reinicia o processo a partir do ponto de queda.

Um simulador desenvolvido no Matlab possibilitou a execução da simulação computacional. O algoritmo convergiu para um ciclo de marcha após 6 tentativas utilizando uma matriz de dezesseis objetivos.

A principal vantagem do algoritmo é a de que não foi necessário resolver nenhuma equação diferencial,

o que simplifica a modelagem matemática do problema. A desvantagem é que a solução apresentada pertence ao conjunto das soluções numéricas, e estas estão sujeitas à singularidades, que resultam em uma falha na obtenção de uma solução, mesmo que exista. Além disso, a solução iterativa não é adequada para a execução em tempo real; o número de iterações é limitado e, portanto, elas podem não convergir. A análise desses fatores sugere uma mudança na tratativa do problema. A direção da pesquisa segue para uma solução analítica, que será abordada em trabalhos futuros.

Agradecimentos

Este estudo foi financiado em parte pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código Financeiro 001. O presente trabalho utilizou a infraestrutura computacional do Laboratório de Computação de Alto Desempenho para Defesa Cibernética, financiado com recursos dos Projetos Estratégicos do Exército Brasileiro. Os autores agradecem João Carlos José dos Santos e o Cel Ney Bruno pelo seu suporte com os experimentos.

Referências bibliográficas

- [1] B. Siciliano and O. Khatib, Springer Handbook of Robotics, 2nd ed. Springer Publishing Company, Incorporated, 2016.
- [2] D. Gouaillier et al., "Mechatronic design of NAO humanoid," 2009 IEEE International Conference on Robotics and Automation, 2009, pp. 769-774, doi: 10.1109/ROBOT.2009.5152516.
- [3] K. Kaneko et al., «Design of prototype humanoid robotics platform for HRP,» IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002, pp. 2431-2436 vol.3, doi: 10.1109/IRDS.2002.1041632.
- [4] Ill-Woo Park, Jung-Yup Kim, Jungho Lee and Jun-Ho Oh, «Mechanical design of humanoid robot platform KHR-3 (KAIST Humanoid Robot 3: HUBO),» 5th IEEE-RAS International Conference on Humanoid Robots, 2005., 2005, pp. 321-326, doi: 10.1109/ICHR.2005.1573587.
- [5] J. Chestnutt, P. Michel, J. Kuffner and Takeo Kanade, «Locomotion among dynamic obstacles for the honda ASIMO,» 2007 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007, pp. 2572-2573, doi: 10.1109/IROS.2007.4399431.
- [6] J. Lim et al., «Robotic software system for the disaster circumstances: System of team KAIST in the DARPA Robotics Challenge Finals,» 2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), 2015, pp. 1161-1166, doi: 10.1109/HUMANOIDS.2015.7363509.
- [7] L. Pineda, T. Takahashi, H. Jung, S. Zilberstein and R. Grupen, «Continual planning for search and rescue robots,» 2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), 2015, pp. 243-248, doi: 10.1109/HUMANOIDS.2015.7363542.
- [8] N. Banerjee et al., «Human-supervised control of the ATLAS humanoid robot for traversing doors,» 2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids), 2015, pp. 722-729, doi: 10.1109/

HUMANOIDS.2015.7363442.

- [9] H. Jeong, O. Sim, H. Bae, K. Lee, J. Oh and J. Oh, «Biped walking stabilization based on foot placement control using capture point feedback,» 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017, pp. 5263-5269, doi: 10.1109/IROS.2017.8206418.
- [10] Y. Minami Shiguematsu, M. Brandao, K. Hashimoto and A. Takanishi, «Effects of Biped Humanoid Robot Walking Gaits on Sparse Visual Odometry Algorithms,» 2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids), 2018, pp. 160-165, doi: 10.1109/HUMANOIDS.2018.8625015.
- [11] S. Kajita et al., «Position-Based Lateral Balance Control for Knee-Stretched Biped Robot,» 2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids), 2019, pp. 17-24, doi: 10.1109/Humanoids43949.2019.9035077.
- [12] G. Nava et al., «Modeling and control of humanoid robots in dynamic environments: ICub balancing on a seesaw,» 2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids), 2017, pp. 263-270, doi: 10.1109/HUMANOIDS.2017.8246884.
- [13] S. Kajita and K. Tani, «Study of dynamic biped locomotion on rugged terrain-theory and basic experiment,» in *Advanced Robotics, 1991. Robots in Unstructured Environments, 91 ICAR., Fifth International Conference on.* IEEE, 1991, pp. 741-746.
- [14] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, «The 3d linear inverted pendulum mode: a simple modeling for a biped walking pattern generation,» in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, vol. 1, 2001, pp. 239–246 vol.1.
- [15] M. Vukobratović and B. Borovac, «Zero-moment point – thirty five years of its life,» *International Journal of Humanoid Robotics*, vol. 1, no. 1, pp. 157–173, 2005. [Online]. Available: <http://www.cs.cmu.edu/~cga/legs/vukobratovic.pdf>
- [16] S. Kajita et al., «Biped Gait Control Based on Spatially Quantized Dynamics,» 2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids), Beijing, China, 2018, pp. 75-81, doi: 10.1109/HUMANOIDS.2018.8624942.
- [17] A. S. Baskoro and M. G. Priyono, «Design of humanoid robot stable walking using inverse kinematics and zero moment point,» 2016 International Electronics Symposium (IES), 2016, pp. 335-339, doi: 10.1109/ELECSYM.2016.7861027.
- [18] T. K. Hooi and M. N. Mahyuddin, «A study of walking gait stability and gait efficiency of a cost-effective small humanoid bipedal robot: Analysis, simulation and implementation,» 2017 IEEE International Symposium on Robotics and Intelligent Sensors (IRIS), 2017, pp. 125-129, doi: 10.1109/IRIS.2017.8250109.
- [19] T. McGeer, «Passive dynamic walking,» *the international journal of robotics research*, vol. 9, no. 2, pp. 62–82, 1990.
- [20] T. Aoyama, K. Sekiyama, Z. Lu, T. Kobayashi, Y. Hasegawa, and T. Fukuda, «Stability enhancement of 3-d biped walking based on passive dynamic autonomous control,» in 2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012), Nov 2012, pp. 443–448.
- [21] H. Suzuki, J. H. Lee and S. Okamoto, «Development of semi-passive biped walking robot embedded with CPG-based locomotion control,» 2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI), Jeju, 2017, pp. 75-78, doi: 10.1109/URAI.2017.7992889.
- [22] K. Hashimoto, Y. Takezaki, H. Motohashi, T. Otani, T. Kishi, H. o. Lim, and A. Takanishi, «Biped walking stabilization based on gait analysis,» in 2012 IEEE International Conference on Robotics and Automation, May 2012, pp. 154–159.
- [23] C. Piazza et al., «Toward an adaptive foot for natural walking,» 2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids), Cancun, 2016, pp. 1204-1210, doi: 10.1109/HUMANOIDS.2016.7803423.
- [24] A. Yamaguchi, S. Hyon, and T. Ogasawara, «Reinforcement learning for balancer embedded humanoid locomotion,» in 2010 10th IEEE-RAS International Conference on Humanoid Robots, Dec 2010, pp. 308–313.
- [25] J. Kulk and J. S. Welsh, «Using redundant fitness functions to improve optimisers for humanoid robot walking,» in 2011 11th IEEE-RAS International Conference on Humanoid Robots, Oct 2011, pp. 312-317.
- [26] M. Oliveira, S. Doncieux, J. Mouret, and C. P. Santos, «Optimization of humanoid walking controller: Crossing the reality gap,» in 2013 13th IEEE-RAS International Conference on Humanoid Robots (Humanoids), Oct 2013, pp. 106-111.
- [27] D. Magee, *Orthopedic Physical Assessment*, 6th ed., ser. *Orthopedic Physical Assessment*. Saunders Elsevier, 2013, no. v. 1.

Data Modeling for Cassandra

Thiago B Leão*, Maria Claudia R Cavalcanti, Raquel C G Pinto
 Instituto Militar de Engenharia (IME),
 Praça General Tibúrcio, 80, 22290-270,
 Praia Vermelha, Rio de Janeiro, RJ, Brasil.
 * thiagobleao@gmail.com

RESUMO: Devido à crescente demanda por escalabilidade e distribuição de dados, uma variedade de sistemas gerenciadores de banco de dados NOSQL surgiram e vêm sendo usados com diferentes objetivos. Entre as abordagens mais populares podemos citar a orientada a documentos, a orientada a colunas e a chave-valor. Embora já existam vários sistemas de banco de dados que adotam essas abordagens, até onde foi possível investigar, quase não há diretrizes de modelagem de dados para eles. O algoritmo proposto neste artigo analisa um conjunto de consultas pré-definido e, baseado nas cláusulas de filtro dessas consultas, ele define as chaves primárias e de clustering para um conjunto de visões materializadas. Adicionalmente, ele define um conjunto pares <consulta, visão materializada> indicando quais consultas cada visão materializada atende. Para avaliar o algoritmo, foi realizado um experimento que compara o desempenho entre usar diversas tabelas para cada consulta e usar as visões materializadas sugeridas pelo algoritmo. Os resultados mostraram-se promissores e apontam para novas direções com relação à modelagem de dados para sistemas NOSQL.

PALAVRAS-CHAVE: modelagem de dados, NOSQL, visões materializadas.

ABSTRACT: Due to the increasing demand for scalability and distribution of data, a variety of NOSQL database management systems have emerged and are being used for different purposes. Some of the most popular systems are: Document-oriented, Column-oriented and Key-value. Although there are many of these systems, to the best of our knowledge, there are barely any guidelines for data modeling for them. The proposed algorithm analyzes a predefined set of queries and, based on their filter clauses, it defines the composition of the primary and clustering keys for a set of materialized views. In addition, it defines a set of <query, materialized view> pairs, indicating which queries each materialized view addresses. To evaluate the algorithm, we report on an experiment that compares the performance of having different tables for each query and of having the materialized views suggested by the algorithm. It shows promising results and points to new directions on data modeling for NOSQL systems.

KEYWORDS: data modeling, NOSQL, materialized views.

1. Introduction

NOSQL came as an alternative to supply a demand for scalability, availability, and fault tolerance. It emerged as an alternative to Relational database management systems. Although there are many different NOSQL solutions already available, to the best of our knowledge, there are barely any guidelines for data modeling for such solutions. Moreover, modeling guidelines for Relational DBMS are not appropriate for NOSQL approaches. While Relational modeling focuses on creating distinct relations and references between them, avoiding data redundancy, NOSQL approaches stimulate data aggregation and duplication [1].

NOSQL database systems are classified according to the way they store and access data. For each of these approaches, there may be a different way to model the data. There are already some initiatives to fill the gap of data modeling for these different NOSQL

approaches [2][3][4]. In this last work, the authors propose data modeling for the HBase system, which is classified as a Column-oriented system. Cassandra is also classified as a column-oriented system, but after running some initial tests, we have seen that Cassandra and HBase systems work differently and the data modeling process for them cannot be the same.

Therefore, in this paper, we propose some guidelines to help the user with data modeling for the Cassandra NOSQL system. We assume that there is an initial schema that consists of a single table, which results from a denormalized relational schema. Additionally, we also consider a set of pre-defined queries on this schema. The main contribution of the proposed guidelines consists of a heuristic. It is formalized as an algorithm that ranks possibilities of primary keys and materialized views based on query demands, aiming at the reduction of the number of materialized views to be created and maintained.

In order to evaluate the proposed algorithm, we took a well-known benchmark for OLAP applications. The idea was to evaluate performance gains among different data modelings for the same relational schema, including the one generated by the proposed algorithm.

This paper is organized as follows. The second section presents some concepts and technological details that were used to develop the proposed method. Section 3 describes some related work, highlighting the contribution of our approach, which is presented in section 4. Sections 5 and 6 present the method evaluation scenario, the results, and their discussion. Finally, section 7 concludes the work by indicating future directions.

2. NOSQL databases

One of the main characteristics of NOSQL DBMS is the ease of dealing with data scalability while maintaining a good performance [1]. Some of these systems use resources such as parallel architectures, data sharding, and replication in order to gain performance. On the other hand, in these systems, maintaining consistency may be an issue and thus many of them do not provide the ACID properties. They usually provide what is called eventual consistency, which allows the replicas and/or shards not to be fully consistent all the time, but at some future point in time. This disadvantage is acceptable to obtain benefits such as availability and performance.

The main characteristic of column-oriented databases is that they store tables in columns instead of rows. In a relational database, each tuple (with all its attribute values) is stored together. Thus, to retrieve the values of part of these attributes, it is necessary to retrieve the entire tuple, directly affecting the query execution time [5]. Differently, in a columnar database, the attribute values of a tuple may be stored separately in columns. For instance, a column may store all values of a single attribute and their corresponding identifying keys. It also may be organized in families of columns, where each family

may store a subset of attributes that compose the tuple stored in the database. With this approach, retrieving some attributes does not bring the whole tuple, resulting in a better performance if compared to relational databases.

Therefore, column-oriented databases tend to perform better than relational databases, especially when executing aggregation queries over some attributes. Cassandra and HBase are examples of column-oriented databases. In this work, we chose to work with Cassandra because of its popularity among the column-oriented databases (according to the DB-Engines site ¹).

In Cassandra, data is distributed over all nodes of a cluster, according to the partition keys defined on each table. When a node is added or removed, all its data is automatically distributed over the other available nodes. If a node fails, it will be replaced instantly. Because of this, it is no longer necessary to calculate and assign data to each node. Cassandra's architecture is known to be peer-to-peer (it partitions tasks or workloads among peers equally) and overcomes master-slave limitations by providing high availability and massive scalability. Data is replicated over multiple nodes in the cluster. Failed nodes are detected by gossip protocols (peer-to-peer communication protocol in which nodes periodically exchange state information about themselves, and about other nodes they know about), and those nodes can be replaced instantly [6].

In Cassandra, data is indexed by the primary key, which is composed of a partition key and clustering keys. The partition key is used by Cassandra to define how data will be partitioned over the nodes. The clustering keys define how data will be ordered on the partition. The primary key leads to the row where the data is stored, and in each row, the data is divided into columns and column families. Each column in Cassandra has a name, a value, and a timestamp. Both the value and the timestamp are provided by the client application when data is inserted.

Recently, in Cassandra 3.0, the concept of Column Family is also called Table. Unlike columns, the Tables

¹ <https://db-engines.com/en/ranking>

are not dynamic and must be previously declared in a configuration file. They are the unit of abstraction containing keyed rows that group together columns of highly structured data. Tables have no defined schema of column names and types supported. Lastly, tables are grouped into Keyspaces. These Keyspaces can be compared to Schemas in a relational database.

In Cassandra's peer-to-peer model, each node exchanges information across the cluster every second. A sequentially written commit log on each node captures write activity to ensure data durability[6]. Data is then indexed and written to an in-memory structure called MemTable, which resembles a write-back cache. Once the memory structure is full, the data is written to disk in an SSTable (sorted string table) data file (a file of key/value string pairs, sorted by keys). All writes are automatically partitioned and replicated throughout the cluster. When a read or write request is made, any node in the cluster can handle it. Through the key, the node that answered the request can know which node possesses data information.

Cassandra also enables the creation of materialized views. The concept is the same as in relational databases. The idea is to store the data according to some predefined query, aiming at improving performance. Each table may have one or more materialized views.

Typically, the disadvantages of the usage of materialized views are: an extra storage cost and the time cost for the maintenance of consistent materialized views, as updates occur in the base table. In Cassandra, when the user updates the base table, the materialized views will be updated automatically, generating a lower maintenance cost at the user level.

3. Related work

This section summarizes and compares papers presented as shown in Table 1. In order to fill the data modeling gap for NOSQL databases, some works chose to focus on a specific performance demanding application: the OLAP application [7] which has a heavy use of queries to retrieve

large volumes of data. Typically, it is based on the multidimensional model, which includes the fact and the dimension concepts. These concepts are represented in the relational model as a star schema, where the fact corresponds to a table as well as each dimension. Each fact tuple refers to tuples in each dimension.

The transformation of the multi-dimensional conceptual model directly to the NOSQL logical model is proposed by Chevalier *et al* [8]: each star schema is mapped into a single table. The fact is transformed into a column family, in which every measure is a column. Each dimension is transformed into a column family, in which every attribute is a column. In addition, all aggregation possibilities for that schema are also similarly mapped into a separate table, as materialized views. However, in this work, there is no intention in selecting a subset of those materialized views, which implies high costs concerning storage and materialized view maintenance.

A complementary study over NOSQL Multidimensional Modeling [9] presents three different ways of logical modeling in a NOSQL columnar database. The first one, named normalized logical approach (NLA), adopts a vertical fragmentation of a denormalized star schema and stores the fact and each dimension into different tables. The denormalized logical approach (DLA) maps the star schema into a single table, which stores the fact and dimensions all together. The third one, called denormalized logical approach using column families (DLA-CF), is similar to the DLA approach, but the dimensions and the fact are mapped, each one, to a different column family.

Another study [10] proposes a Cassandra data modeling based on the queries. It also defines modeling principles, mapping rules, and mapping patterns. This methodology prioritizes the applications workflow and its access patterns. The normalization is removed, implying data redundancy and materialized views usage over joins. Because of those differences, it is necessary a paradigm shift from modeling based purely on

entities and relationships to modeling based on queries.

On Scabora *et al* [4], three modelings are used over HBase to evaluate the performance of OLAP Queries. In addition to the DLA e DLA-CF modelings presented on Dehdouh *et al* [9], which are denominated SameCF e CNSSB respectively, the authors propose the FactDate modeling as the third alternative. It follows the same idea of the DLA-CF alternative, but it gathers the fact and the date dimension in the same column family. Experiments with those three modeling alternatives showed that the FactDate alternative has better performance on queries that use few dimensions, i.e., the Date dimension and one other. On the other hand, the SameCF alternative has better performance on queries that use a higher number of dimensions.

These related works were reported on modeling performance of column-oriented databases, but they do not approach how to deal with the data distribution nor how to select materialized views to get the best performance of the database. This is a crucial factor to execute queries properly over Cassandra. In this work, we present a set of guidelines and a heuristic to help the modeler on selecting the best distribution keys (partition and clustering keys) and a set of materialized views for the Cassandra database system.

4. Initial experiments

As previously mentioned [8], in order to perform data modeling, it is a good practice to start with a conceptual schema of the data and then proceed to the data modeling, where the conceptual schema elements are mapped into a logical/physical schema of a specific DBMS.

Once a logical schema is designed, it is important to know the typical/critical queries that should be supported by the application. From these, it is possible to define logical/physical schema alternatives (candidate schemas) to the database. In the case of a column-oriented DBMS, the choice of such schema is not trivial. A careful analysis is necessary to

identify the most appropriate logical/physical schema according to the application demand.

In short, data modeling consists of two steps. The first step is concerned with conceiving the first version of a logical schema. Then, the second one focuses on performance issues and on attending to application demands, such as addressing a set of critical queries. In this work, we address just the second step for the Cassandra DBMS. We assume that an initial logical schema is already available, and then we apply a set of heuristics in the form of an algorithm.

Tab. 1 – related work comparison.

Work	Modeling	Materialized Views	DBMS
Chevalier <i>et al</i> [8]	Conceptual/ Logical	-	MongoDB and HBase
Dehdouh <i>et al</i> [9]	Conceptual/ Logical	-	HBase
Chebotko <i>et al</i> [10]	Logical	-	Cassandra
Scabora <i>et al</i> [4]	Logical	Materialized Views with an external application	HBase

In order to develop such an algorithm, we performed some initial experiments for a typical OLAP application, which are detailed in this Section. To guarantee that we would start with the best initial logical schema, we explored the CNSSB datasets [11] and queries, considering three logical schemas, as proposed in [9] and [4]: SameCF, DLA-CF, and FactDate. Based on the results of such experiments detailed in Section 4.2, we found out that the best initial logical schema was the SameCF schema.

Then, assuming the SameCF schema as the initial schema, we observed the performance gains while using partition and clustering keys alternatives and while querying on materialized views. This discussion is presented in Section 4.2. Then, in Section 5, we identify some heuristics for choosing those keys and materialized views to address most of the queries and reduce the set of materialized views. Finally, these heuristics were formalized in the algorithm presented in Section 5.

4.1 Initial logical schema definition

All three models (SameCF, DLA-CR, and FactDate)² were populated with data generated from

the DBGen application of the CNSSB³ [11]. After populating the models, the thirteen CNSSB queries, organized in 4 typical sets, Q1, Q2, Q3, and Q4 [11], were executed five times on each model to measure their performance average. Experimentally, it was noticed that there was not a large variation within a few executions. Thus, five executions seemed to be sufficient to characterize the performance.

As it can be seen in the graph of **figure 1**, the queries had similar behavior in the DLA-CF and FactDate models, except for the Q1 set of queries that benefits from the fact it does not require joins to perform the queries. The SameCF model can obtain better performance in sets Q2, Q3, and Q4, showing that the use of joins in Cassandra implies a worse performance of the query. Analyzing the query behavior, we can note the influence of the Partition Key and Clustering Keys. Cassandra not only partitions the data and distributes them among the nodes according to the Partition Key but also orders the data in the partition according to the Clustering Key. Queries that use Partition Key attributes as filters usually perform better. However, those queries performance may be reduced if they also use other attributes as filters. Therefore, query performance is highly dependent on the attributes used and on the fact that they are part of the Partition or Clustering Keys.

Since the SameCF model had an average superior performance to the DLA-CF and FactDate, this model was chosen as the initial logical model. From this model, an evaluation of the use of materialized views is presented in the following section. Then, a heuristic is defined based on the use of materialized views according to the SameCF model.

4.2 Experiments results

Analyzing the SameCF model, we can see that the way the attributes are arranged in the PrimaryKey directly influences queries performance. A query that filters on an attribute that belongs to the Partition

Key will perform well as opposed to a query that filters on an attribute that is positioned at the end of the Clustering Key [6]. The Clustering Key sorts the records of a partition according to each attribute defined in it, that is, it is an ordered list of attributes that determines the order of the records in the disk. From the graph of **figure 2**, in the set of queries QG1, there was a significant variation in the performance of the queries. Queries 1.1 and 1.3 performed well because their filter attributes were those used for the formation of the Partition Key (year) and the Clustering Key (discount, quantity), in this case, the first attributes. On the other hand, query 1.2 had the worst performance for two reasons: first, since it does not include the Partition Key attribute as a filter, and second, because the attribute yearmonthnum is an attribute unfavorably positioned in the Clustering Key, that is, it is not an attribute that is positioned right at the beginning of the Clustering Key, impairing filter performance.

Regarding the QG2 set, queries showed the best average performance, with times very close to each other. On the other hand, the QG3 was the worst set. Interestingly, query 3.1 had one of the best performances concerning all queries of all sets. This is explained by the fact that the attributes of the filters in this query are either an attribute of the Partition Key (year) or belong to the first positions of the Clustering Key (supregion, region). Although queries 3.2 and 3.3 used a filter based on the Partition Key attribute (year), the other attributes used as filters belong to unfavorable positions in the Clustering Key, which explains their bad performances. Query 3.4 does not use the Partition Key (year) attribute as a filter but as an ordering/grouping attribute and, in addition, it also filters data by unfavorably positioned Clustering Key attributes.

Finally, when analyzing the QG4 set, we note that queries 4.1 and 4.2 had a good performance. This is probably due to the fact that those queries filter using the first attribute of the Clustering Key. The use of

²<https://db-engines.com/en/ranking>

³https://github.com/thiagobleao/cnssb_dataset

the Partition Key (year) in query 4.2 may have led it to perform better. Query 4.3 had the worst performance because it used only unfavorable positioned attributes of the Clustering Key. However, interestingly, it uses the Partition Key (year) as filter, which shows that only using the Partition Key will not necessarily ensure the best performance for a query.

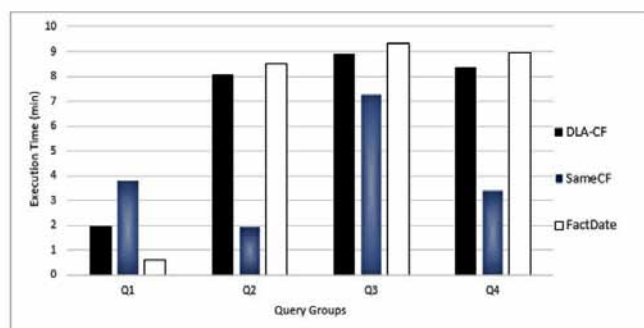


Fig. 1 – Query performance on each model

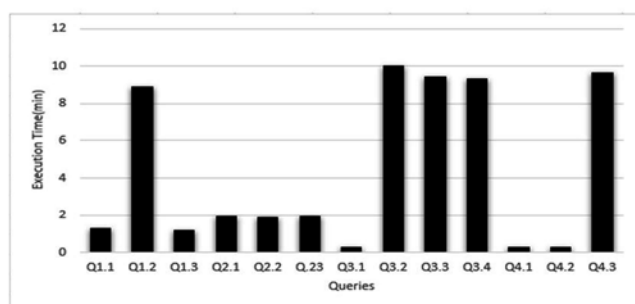


Fig. 2 – Queries performance in SameCF model

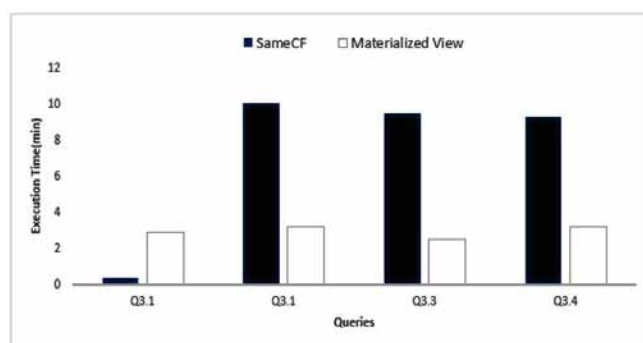


Fig. 3 – Materialized view performance

To improve the performance of the queries, it is possible to create materialized views from a table in Cassandra. This allows the choice of attributes that will be part of each materialized view and the reorganization

of the Partition and Clustering Keys for that materialized view. Using the criterion proposed by Baralis *et al* [12], we chose the worst performance set of queries (Q3) as the basis for the creation of the materialized view. The graph of **figure 3** shows the performance of these queries on the materialized view and compares them to the base table. When performing the four queries of group Q3 on the created materialized view, we can observe that query 3.1 loses performance. This is explained using the modified Clustering Key. In addition, this is also the reason for the reduction of the execution time of the other queries. That is, query 3.1 can continue to be executed directly on the base table and the other queries would do better if executed on the materialized view.

We can conclude that using joins through applications is not ideal since there is a loss of performance. That said, the ideal for modeling in Cassandra is to denormalize the data and store them in a single table.

5. Guidelines for cassandra

Usually, at the logical modeling phase of a database design, the idea is to depart from a conceptual DBMS independent view of the application and arrive at a DBMS dependent schema of it. In the context of a relational database, it consists of designing a set of tables and attributes, whereas, in the context of a columnar database approach, this means choosing which column families should be created. Assuming an initial columnar logical schema is already chosen, the next phase of a database design is the physical schema design, which focuses on the performance of the database for a given set of applications, queries, and activities. In the Cassandra case, this includes defining materialized views.

Based on the results of the initial experiments, reported in subsection 4.2, it was possible to devise some initial guidelines, concerning the choice of a logical/physical schema for Cassandra DBMS.

Guideline 1. Denormalize the logical schema.

Considering an initial conceptual-logical mapping as the logical initial schema, it is recommended to denormalize this schema, in such a way that it is able to answer a set of critical queries or demands.

Guideline 2. Define the primary key attributes.

Based on the set of critical queries, analyze the most frequently used attributes on the query constraints and then choose the ones to form the primary key.

Guideline 3. Define a set of materialized views.

The idea is to group queries, according to their common attributes, i.e., attributes that are present in selecting and filtering clauses. These groups are the basis for the choice of the materialized views. Once defined, these materialized views may be created.

Guideline 4. Define a query redirection policy.

Given a set of materialized views, a query redirection mechanism could benefit from those materialized views, by rewriting and redirecting the query to the materialized view that can probably provide the best performance for that query.

In order to help the designer with guidelines 2 and 3, in this section we present Algorithm 1. It is based on Cassandra query execution constraints, previously presented in Section 2. It takes as input an initial denormalized logical schema, as suggested by guideline 1, and a set of pre-defined critical/typical queries, and its output may be used as the input for guideline 4.

Cassandra supports query execution only directly into a partition, that is, it demands an equality constraint over the partition key. This is the first premise adopted to suggest a materialized view. Another premise based on Cassandra's limitations is the use of inequality constraints, which can only appear once in each query. In the case of constraints with the IN clause, it may appear along with another inequality constraint, and this must be the last one to be applied in the query expression, and only in the Clustering Key.

Therefore, based on these initial premises, Algorithm 1 finds possible combinations of attributes to form the primary key for each materialized view. The main idea is to identify groups of queries, of which constraints use attributes in common, and for each group, the algorithm suggests a reduced number of materialized views, with their respective primary keys. In addition, for each materialized view, it suggests the set of queries that are associated with it, i.e., it indicates to which materialized view each query should be redirected to (or rewritten to).

The following variables are used in Algorithm 1

(figure 4):

- Q : set of critical/typical queries to be executed on Cassandra;
- v : index of the materialized view under construction, $1 \leq v \leq |Q|$;
- AE_x : set of equality attributes, i.e., attributes that are involved in equality-based constraints on a query qx expression, which will compose the set of attributes AE_v of the primary key of the associated materialized view v ;
- AE_v : set of equality-based attributes of materialized view v ;
- ANE_x : set of non-equality attributes, i.e., attributes that are involved in inequality-based constraints on a query qx expression, which will compose the set of attributes ANE_v primary key of the associated materialized view v ;
- ANE_v : set of inequality-based attributes of materialized view v ;
- A_x : set of attributes of query qx , where $A_x = AE_x \cup ANE_x$;
- A_v : set of attributes that will compose the primary key of the associated materialized view, where $A_v = AE_v \cup ANE_v$;
- Q_v : set of queries that will be addressed by the materialized view v ;
- V : set of materialized views to be generated, formed by a set of pairs (A_v, Q_v) ;

Q' : set of queries that are not supported by none of the existing materialized view in V ;

In step I, Q' is initialized with the complete set of queries, then the algorithm iterates over the Q' set until it becomes empty. Each iteration on Q' (step I) aims at building a new materialized view v and its corresponding primary key. Every query q_x from Q' is analyzed by the algorithm concerning its attributes involved in equality and inequality constraints until the key for the materialized view under construction is formed. Depending on the evolution of the key under construction for the materialized view of the moment (v), query q_x may be treated in steps II, III, and IV. If it satisfies the constraints for one of these steps, it is included in the set of queries (Q_v) that will be addressed by materialized view v and removed from the Q' set.

The first query of each iteration on Q' is always treated within step II, where the sets of equality/inequality-based attributes are initialized for the materialized view of the moment. Once the first attributes for a materialized view are defined, then the next Q' queries are treated by the following steps, depending on if they have attributes in common with the materialized view under construction. Step III will add queries that bring to the materialized view one attribute involved in an inequality constraint at the most, and those queries have attributes in common with the set of attributes of the materialized view ($A_v \subseteq A_x$ or $A_x \subseteq A_v$). Step IV deals with queries that bring to the materialized view more than one attribute involved in an inequality constraint. In this case, the query under analysis needs to have the same equality constraint attributes. Queries that did not fit within any of the previous steps remain in Q' , and they will be analyzed again for a new iteration, i.e., for the creation of a new materialized view.

Algorithm 1 Cassandra Modeling Heuristic

```

Input:  $Q$ 
output:  $V$ 
function modeling_views( $Q$ )
 $Q' \leftarrow Q; V \leftarrow \{\}$ ;  $v = 0$ ;
(I) While  $Q' \neq \{\}$ , do:
    counter = 0;
     $v = v + 1$ ;
     $AE_v = \{\}$ ;
     $ANE_v = \{\}$ ;
    For each  $q_x \in Q'$ , do:
         $AE_x \leftarrow getEqAttributes(q_x)$ ;
         $ANE_x \leftarrow getIneqAttributes(q_x)$ ;
(II) If (counter = 0):
     $AE_v \leftarrow AE_x$ ;
     $ANE_v \leftarrow ANE_x$ ;
     $Q' \leftarrow Q' - \{q_x\}$ ;
     $Q_v \leftarrow Q_v \cup \{q_x\}$ ;
(III) Else If (( $A_v \subseteq A_x$  and  $ANE_v = \{\}$  and  $Size(ANE_x) < 2$ ) or
    ( $A_x \subseteq A_v$  and  $ANE_x = \{\}$  and  $Size(ANE_v) < 2$ )):
     $AE_v \leftarrow AE_x \cup AE_v$ ;
     $ANE_v \leftarrow ANE_x \cup ANE_v$ ;
    sort_by_equality_in_AE( $ANE_v$ );
     $AE_v \leftarrow AE_v - ANE_v$ ;
     $Q' \leftarrow Q' - \{q_x\}$ ;
     $Q_v \leftarrow Q_v \cup \{q_x\}$ ;
(IV) Else If ( $AE_v = AE_x$  and
    ( $ANE_v \subseteq ANE_x$ ) or
    ( $ANE_x \subseteq ANE_v$ )):
     $ANE_v \leftarrow ANE_x \cup ANE_v$ ;
    sort_by_equality_in_AE( $ANE_v$ );
     $AE_v \leftarrow AE_v - ANE_v$ ;
     $Q' \leftarrow Q' - \{q_x\}$ ;
     $Q_v \leftarrow Q_v \cup \{q_x\}$ ;
    counter = counter + 1;
(V)  $A_v \leftarrow AE_v \cup ANE_v$ ;
     $V \leftarrow V \cup \{(A_v, Q_v)\}$ ;

```

Fig. 4 – Algorithm 1

Finally, at the end of each while iteration, in step (V), the set of attributes that compose the materialized view A_v is created and the set of materialized views V is incremented with the new pair (materialized view (A_v), query (Q_v)).

6. Experiments with materialized views in cassandra

In this section, we present experiments' results to evaluate the heuristics proposed in the previous section. The purpose of these experiments is to compare the performance of three logical models with different input rates of reading and writing operations, one of which uses the proposed heuristics.

The experiments were carried out in a computational cluster with four nodes, each with 158 GB of RAM, 64 CPUs of 2.4 GHz. The operating system of each node is CentOS, with the DBMS Cassandra (version 3.0). The dataset used during the experiments was generated by the dbgen of the CNSSB [11] without any post-treatment since it is already generated on a single denormalized table in a CSV file.

6.1 Heuristic application

Using the CNSSB schema with its thirteen queries⁴, the proposed heuristic created a set of nine materialized views. Each materialized view has a different primary key that will allow the execution of one or more queries. None of these queries could be performed by more than one materialized view. Therefore, a minimum number of materialized views was generated to meet all the specified queries.

Next, it is exemplified the generation of a materialized view formed by the following two queries:

1. select year, nation, revenue, supplycost from cnssb.nlineorder where region = 'AMERICA' and suppreion = 'AMERICA' and mfgr in ('MFGR#1','MFGR#2')
2. select year, nation, revenue, supplycost from cnssb.nlineorder where region = 'AMERICA' and

⁴ https://github.com/thiagobleao/rmct_appendix/blob/main/rmct_appendix.pdf

suppregeion = 'AMERICA' and year in (1997,1998) and mfg in ('MFG#1','MFG#2')

Both queries have three attributes in common in equality filters: region, suppregeion, and mfg. These attributes will initially compose the primary key, one of them as the partition key, and the other attributes will begin the clustering key. The query (2) also has an equality filter over the year attribute. This attribute must also be included in the clustering key since it is not used by the query (1). The composition of the primary key of the materialized view that fits the two queries is:

- Partition Key: region
- Clustering Key: suppregeion, mfg, year

The next example is based on a group of queries that use equality and inequality filters:

1. select discount,quantity from cnsb.nlineorder where year = 1993 and quantity < 25 and discount between 1 and 3
2. select extendedprice,discount as revenue from cnsb.nlineorder where year = 1994 and yearmonth = 'Jan1994' and quantity between 26 and 35 and discount between 4 and 6
3. select extendedprice, discount as revenue from cnsb.nlineorder where year = 1994 and weeknuminyear = 6 and quantity between 26 and 35 and discount between 5 and 7

If executed as presented, these queries will not run in Cassandra since this DBMS restricts comparisons of "greater than" and "less than" types only to the last field of the key to be filtered. Because of this, these three queries were adapted by changing the "quantity <" to "quantity in". According to Cassandra's constraints, the filter on the discount attribute could be made using "greater than" or "less than" since it is the last attribute of the clustering key of the materialized view. But in order not to assume that this will necessarily be done by the heuristic, it was decided to adapt this filter also to use the "in" clause. After these adaptations, the queries became as follows:

1. select discount,quantity from cnsb.nlineorder where year = 1993 and yearmonth in ('Jan1993', 'Feb1993', 'Mar1993', 'Apr1993', 'May1993', 'Jun1993', 'Jul1993', 'Aug1993', 'Sep1993',

'Oct1993', 'Nov1993', 'Dec1993') and quantity in (0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24) and discount in (1, 2, 3)

2. select extendedprice,discount as revenue from cnsb.nlineorder where year = 1994 and yearmonth = 'Jan1994' and quantity in (26, 27, 28, 29, 30, 31, 32, 33, 34, 35) and discount in (4, 5, 6)
3. select extendedprice, discount as revenue from cnsb.nlineorder where year = 1994 and weeknuminyear = 6 and quantity in (26, 27, 28, 29, 30, 31, 32, 33, 34, 35) and discount in (5,6,7)

Regarding the application of the heuristic on these three queries, another adaptation was made in the query (1) so that it could be served by the same materialized view of the query (2). Originally, query (1) does not have the filter on the attribute yearmonth. However, it could be included in query (1) to achieve the same result and to add a few restrictions to query (1). The same cannot be done to query (3) on the weeknuminyear filter. While the yearmonth filter represents the months of one year (maximum of twelve restrictions), the weeknuminyear filter represents the number of the week of a year (approximately fifty-two restrictions). The last filter has a very high number of restrictions, escaping much from a scenario closer to a real-world application. In this way, the heuristic generated two materialized views, one to answer queries (1) and (2) and another to answer the query (3).

The queries (1) and (2), after the aforementioned adaptations, have equality filters over the following attributes: year and yearmonth. These attributes will compose the primary key of the materialized view, with one being the partition key and the other being the clustering key. These queries also have two non-equality filters: quantity and discount. Both attributes will be included in the clustering key, right after the equality attribute that was included in it. The composition of the primary key of the vision that will support these queries is:

- Partition Key: year
- Clustering Key: yearmonth, quantity, discount

Query (3) should correspond to a materialized

view that will only attend to that query. The key of this materialized view will be composed of the attributes “year” and “weeknuminyear”. One of them will be chosen for the partition key. The other one will be the first attribute in the clustering key of that view, and it will be followed by the attributes quantity and discount (involved in inequality conditions). The composition of the primary key of the view that will support this query is:

- Partition Key: year
- Clustering Key: weeknuminyear, quantity, discount

Like the examples cited above, nine materialized views were created from the application of the proposed heuristic on all of the thirteen queries, as shown in table 2. After the implementation of the three models in Cassandra, the five workloads were executed in each modeling mode to evaluate the proposed heuristic and verify its performance compared with the other models.

6.2 Evaluation scenarios

To evaluate the performance of the use of the heuristic in Cassandra, three different scenarios were used. For each scenario, a keyspace was created with a replication equal to 2, that is, each record was replicated twice in the cluster.

1. Scenario 1: Heuristic generated model, with a minimum of materialized views that can handle all the queries.
2. Scenario 2: Modeling with a materialized view for each query.
3. Scenario 3: Modeling with a table for each query.

Five workloads were developed to evaluate different aspects of each scenario. Each workload represents a combination of reading and writing operations. The objective is to analyze the performance of the evaluated models considering different possibilities of operations that are made over a database.

All workloads are based on the same dataset (CNSSB) to ensure that they are executed under the same conditions. In this way, it is expected to identify which scenario provides the best performance for each workload.

The following workloads were used:

- Only read operations.
- Most read operations: 75% read and 25% write operations.
- Read and write operations equal: 50% read and 50% write operations.
- Most write operations: 25% read and 75% write operations. —Only write operations.

These workloads were developed using the CNSSB dataset file as a basis for writing operations and CNSSB queries of reading operations. The execution of each workload was performed through a program written in the Python language that performs all operations in parallel.

6.3 Performance analysis

Initially, the performance of the three scenarios is verified in a workload with reading operations only. The execution performance of the three scenarios is very close and took around 10 minutes. Considering that each scenario is modeled to attend all the proposed queries and Cassandra itself ensures that a query will be executed only if it has a good performance, model variations do not affect the execution performance of the queries. However, read operations concurs with the update and insert operations and, in this case, there is a usual performance loss.

Tab 2: materialized views created from the heuristic

View	Partition Key	Clustering Key	Supported Queries
V1	suppreigion	region,mfgr,year	4.1, 4.2
V2	suppcity	city,year,yearmonth	3.3, 3.4
V3	suppreigion	brand1	2.2, 2.3
V4	year	yearmonth,quantity,discount	1.1, 1.2
V5	year	weeknuminyear,quantity,discount	1.3
V6	suppreigion	category	2.1
V7	suppreigion	region,year	3.1
V8	year	suppnation,region,category	4.3
V9	suppnation	nation,year	3.2

Considering the workloads that include writing operations, there are two different situations: write operations that result from existing records (updates) and write operations that are inserts of new registries (inserts). Due to the fact that Cassandra does not

read the existing values when running an update [6] when the number of updates is greater than the number of inserts, the table-based modeling tends to offer better performance, even when it runs more operations. However, the use of materialized views makes Cassandra lose this feature [13], leading to a worse performance. Due to these differences, two comparisons were made with write operations in workloads, one performing only updates and another only inserting new records.

Figure 5 shows the results of the execution of the workloads with inserts. This graph uses a logarithmic scale, but the absolute values of the runtime in minutes are highlighted in each bar. From these results, it was observed that scenario 3 (table-based modeling) is clearly the most affected by the increase of writing operations. This scenario has a table for each query and these tables are completely independent of each other. Therefore, each writing to be done in the database must be done thirteen times in order to update all tables. This fact explains the significant increase of the execution time in this scenario, as the number of write operations increases.

Still considering **figure 5**, it was verified that scenario 1 (modeling generated by the heuristic) performs better than scenario 2 (modeling with a materialized view per query). This improvement in performance ranges from 13% to 35%, and it is justified by the fact that in scenario 2 there is a greater number of materialized views. This does not imply the execution of a greater number of write operations, as it occurs in scenario 3. The replication of the updates is done automatically by Cassandra internally. Although this replication is much more efficient than writing separately in each table, it has a running cost. As scenario 2 has more materialized views than scenario 1, this cost ends up directly affecting the performance in scenario 2.

Figure 6 shows the results of the executions of the workloads with updates. This graph also uses a logarithmic scale, and the values of the runtime in minutes are highlighted in each bar. In the case of existing records updates, there is a change of behavior considering the operations of new records

insertions. It was observed that scenario 3 performed better compared to other scenarios. This is because Cassandra implements updates on materialized views as follows: it performs reading the data already present in the materialized views, updates it, removes the old record and inserts the new one. While without the use of materialized views, Cassandra simply inserts the record with a most recent timestamp, and this is the record that will be returned to the queries.

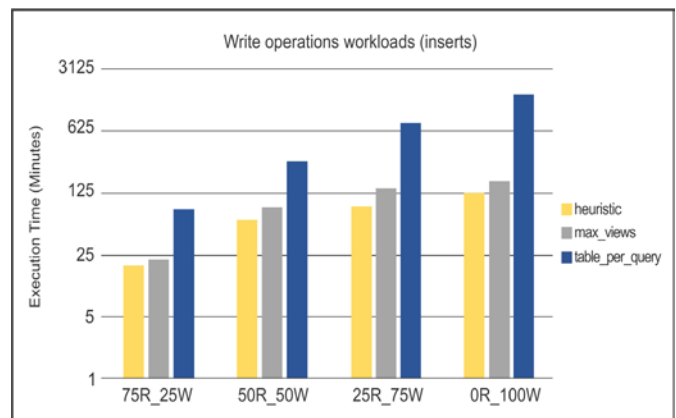


Fig. 5 – Graph of workloads with new registries insert operations

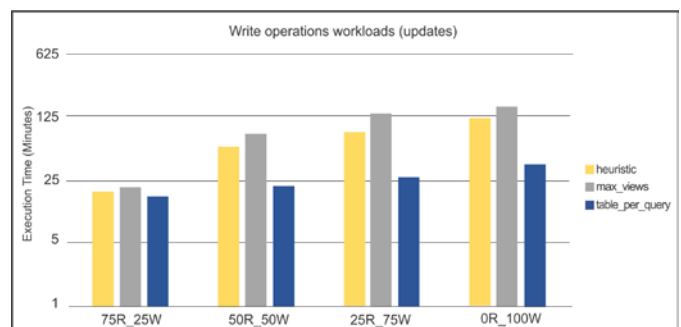


Fig. 6 – Graph of workloads with existing registries update operations

Comparing scenarios 1 and 2, it can be observed in **figure 6** that scenario 1 presents a better performance in all situations, ranging from 14% to 31%, performance similar to the use of inserts. In addition, there is an increase in performance gain as the number of updates increases.

Although scenario 3 performs better than scenarios 1 and 2, the use of the table-based modeling leads to a higher cost of maintaining the data since a change in one table should be reflected

in all other tables created, losing Cassandra's feature of syncing materialized views.

We can observe that a factor that influences the number of materialized views suggested by the heuristic is the variety of attributes in the query filters. The smaller the variety of these attributes, the smaller the number of materialized views generated by the heuristic, ensuring a more significant performance of scenario 1 compared to scenario 2, especially when there are a lot of write operations.

7. Conclusion

This work presents a set of guidelines to support the logical/physical design of database schemas for Cassandra DBMS. It includes a heuristic for data modeling based on specific queries to define a set of materialized views and their corresponding primary keys.

The CNSSB benchmark dataset and its queries were used to evaluate the proposed heuristic. The experiments used workloads varying the rate of read/write operations. The results showed that the more insert operations the better was the performance of the heuristic. On the other hand, when most of the operations are updated, the use of a table for each query performs better. However, it is worth

saying that the reduced number of materialized views (heuristic scenario) is still a better choice if compared to the use of all possible materialized views. Therefore, the proposed guidelines bring light to the data modeling for Cassandra DBMS. Moreover, in the case of analytical applications, where write operations are usually a large set of inserts, the heuristic is particularly useful.

For future work, we plan to apply the heuristic with different datasets, sets of queries, and applications like OLTP. Also, we intend to investigate the application of the heuristic with adjustments, over different NOSQL DBMS that behave similarly to Cassandra. Additionally, we intend to evaluate the impact of Cassandra replication factor and how its variation may affect query performance and memory usage.

Acknowledgments

The authors would like to thank the Brazilian agency CAPES for the financial support and Cristina Dutra de Aguiar Ciferri for her careful review of this work. In addition, thanks to the Brazilian Army for the use of the computational infrastructure of the High-Performance Computing Laboratory for Cyber Defense, funded with resources from the Brazilian Army Strategic Project.

References

- [1] Pramod J. Sadalage and Martin Fowler. 2012. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence (1st ed.). Addison-Wesley Professional
- [2] R. A. S. N. Soransso and Maria Cláudia Cavalcanti. 2018. Data modeling for analytical queries on document-oriented DBMS. In Proceedings of the 33rd Annual ACM Symposium on Applied Computing, SAC 2018, Pau, France, April 09-13, 2018. 541–548.
- [3] Silas P. Lima Filho, Maria Cláudia Cavalcanti, and Cláudia Marcela Justel. 2018. Managing Graph Modeling Alternatives for Link Prediction. In Proceedings of the 20th International Conference on Enterprise Information Systems, ICEIS 2018, Funchal, Madeira, Portugal, March 21-24, 2018, Volume 2. 71–80.
- [4] Lucas C. Scabora, Jaqueline Joice Brito, Ricardo Rodrigues Ciferri, and Cristina Dutra de Aguiar Ciferri. 2016. Physical Data Warehouse Design on NoSQL Databases - OLAP Query Processing over HBase. In Proc. of the 18th Int. Conf. on Enterprise Information Syst. (ICEIS) (1 ed.). 111–118.
- [5] Gheorghe Matei. 2010. Column-Oriented Databases, an Alternative for Analytical Environment. Database Systems Journal 1 (2010), 3–16.
- [6] DataStax. 2018. How Cassandra reads and writes data. <https://docs.datastax.com/en//cassandra/3.0/cassandra/dml/dmlHowDataWritten.html> 13 jul. de 2018.
- [7] R. Kimball and M. Ross. The Data Warehouse Toolkit: the Definitive Guide to Dimensional Modeling. John Wiley & Sons, New York, USA, 3 edition, 2013.

- [8] Max Chevalier, Mohammed El Malki, Arlind Kopliku, Olivier Teste, and Ronan Tournier. 2015. Implementing Multidimensional Data Warehouses into NoSQL. In ICEIS 2015 - Proceedings of the 17th International Conference on Enterprise Information Systems, Volume 1, Barcelona, Spain, 27-30 April, 2015 (1 ed.). 172–183.
- [9] Khaled Dehdouh, Fadila Bentayeb, Omar Boussaid, and Nadia Kabachi. 2015. Using the column oriented NoSQL model for implementing big data warehouses. In Proceedings of the International Conference on Parallel and Distributed Processing Techniques and Applications (PDPTA) (1 ed.). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), 469.
- [10] A. Chebotko, A. Kashlev, and S. Lu. 2015. A Big Data Modeling Methodology for Apache Cassandra. In 2015 IEEE International Congress on Big Data (1 ed.). 238–245. <https://doi.org/10.1109/BigDataCongress.2015.41>
- [11] Khaled Dehdouh, Fadila Bentayeb, and Omar Boussaid. 2014. Columnar NoSQL Star Schema Benchmark. In Model and Data Engineering (1 ed.). The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp), Springer International Publishing, 281–288.
- [12] Elena Baralis, Stefano Paraboschi, and Ernest Teniente. 1997. Materialized Views Selection in a Multidimensional Database. In VLDB'97, Proceedings of 23rd International Conference on Very Large Data Bases, August 25-29, 1997, Athens, Greece (1 ed.). 156–165.
- [13] Jonathan Ellis. 2018. Materialized View Performance in Cassandra 3.x. <https://www.datastax.com/dev/blog/materialized-view-performance-in-cassandra-3-x> 13 jul. de 2018.

Uma proposta de ferramenta para a experimentação do aprendizado profundo no reconhecimento da fala para o idioma Português

Julio Cesar Duarte*, Matheus Rodrigues Affonso, Rebeca Pinheiro dos Reis
Instituto Militar de Engenharia (IME),
Praça General Tibúrcio, 80, 22290-270,
Praia Vermelha, Rio de Janeiro, RJ, Brasil.
* duarte@ime.eb.br

RESUMO: O Processamento de Linguagem Natural já faz parte hoje da rotina de grande parte da sociedade, em gestos simples como comandos de voz pelo celular ou instruções faladas a um assistente virtual. Compreender o que é dito verbalmente pode ser uma tarefa simples para um falante nativo de um idioma, e considerada uma das mais essenciais para o convívio interpessoal. No contexto atual, marcado pela grande busca por integração entre sistemas, Internet das Coisas e ambientes dotados de dispositivos inteligentes e robôs, tal interação entre homens e máquinas fica em evidência, e uma abordagem utilizando aprendizado de máquina para atuar nesse relacionamento torna-se uma solução possível. O presente trabalho tem o objetivo de apresentar um ambiente que possibilite a experimentação de modelos na tarefa de Reconhecimento de Fala, bem como implementar modelos de classificação utilizando redes neurais de aprendizado profundo. Ao final, é feita uma análise comparativa entre a acurácia de possíveis modelos sendo experimentados. Como contribuições deste trabalho, é apresentada uma metodologia que permite a experimentação de reconhecimento de fala, bem como uma proposta de classificador treinado em uma base de dados de áudios em Português do Brasil.

PALAVRAS-CHAVE: Aprendizado Profundo. Redes Neurais recorrentes. Memória de longo e curto prazo. Reconhecimento de Fala.

ABSTRACT: Natural Language Processing is now part of the routine of a large part of society, in simple gestures such as voice commands mobile phones or spoken instructions to a virtual assistant. Understanding what is said verbally can be a simple task for a native speaker of a certain language, and it is considered one of the most essential for interpersonal interaction. In the current context, marked by the great search for integration between systems, Internet of Things and environments equipped with intelligent devices and robots, such interaction between men and machines is in evidence. An approach using machine learning to act in this relationship becomes a possible solution. The present work aims at presenting an environment that allows the experimentation of models for the Speech Recognition task as well as implementing classification models using deep learning neural networks. At the end, a comparative analysis between the accuracy of possible models being tested is made. As contributions of this work, a methodology that allows the experimentation of speech recognition and the proposal of a trained classifier using a Brazilian Portuguese Audio dataset are presented.

KEYWORDS: Deep Learning. Recurrent Neural Networks. Long short-term memory. Speech Recognition.

1. Introdução

Compreender o que é dito verbalmente por intermédio da fala é uma tarefa corriqueira para um falante nativo de um certo idioma. Atualmente, em um mundo marcado pela grande busca por integração entre sistemas, uma abordagem utilizando aprendizado de máquina que possa compreender facilmente uma linguagem, torna-se uma solução possível e desejável nessa interação entre

pessoas e máquinas.

Um problema muito importante nessa área é o denominado Reconhecimento de Fala, ou, do inglês, *Speech Recognition*, e existem na literatura diversos trabalhos que procuram resolvê-lo utilizando aprendizado profundo. Por exemplo, Hannun et al. [1] propõe uma solução completa utilizando uma rede neural recorrente bidirecional para a classificação de arquivos de áudio apresentando, além da arquitetura da rede, um modelo de linguagem para reduzir

erros fonéticos e soluções para paralelismo da fase de treinamento.

Já em um outro estudo, Amodei *et al.* [2] trabalha com uma arquitetura similar, entretanto mais complexa que a do modelo proposto por Hannun *et al.* [1], e que se mostrou muito eficiente tanto para a língua inglesa quanto para o Mandarim, dois idiomas extremamente diferentes no que diz respeito à construção linguística e escrita.

Existem também estudos sobre reconhecimento de fala aplicado à Língua Portuguesa do Brasil. Por exemplo, Quintanilha [3] desenvolve um modelo de aprendizado profundo com erro por rótulo de 25,13%, mostrando ser viável desenvolver um sistema de ponta a ponta para o português. Em termos de ferramentas disponíveis para tal tarefa, a equipe da Mozilla possui um projeto de código aberto com uma implementação baseada na arquitetura do grupo de pesquisa Baidu [4] que utiliza a biblioteca Python do Google para aprendizado profundo denominada *Tensorflow*.

Entretanto, poucos são os trabalhos encontrados que tentam resolver o problema do reconhecimento de fala para o Português e que possuam uma ferramenta de código aberto que permita realizar a experimentação de diferentes arquiteturas e hiperparâmetros.

O presente trabalho tem como finalidade apresentar uma ferramenta de experimentação que possibilite a realização de tais experimentos, utilizando modelos de aprendizado profundo baseados em redes recorrentes para a tarefa de reconhecimento da fala em língua portuguesa. Tal implementação utilizou como base a ferramenta gerenciada pela equipe do Mozilla [4].

Além disso, também são implementados modelos de classificação para a tarefa, a partir de dados de áudio brutos, de forma a definir uma metodologia que permita não só a escolha do melhor modelo para o Português do Brasil, mas que também possibilite a realização de experimentos com outros idiomas. A grande contribuição do trabalho, em relação aos resultados práticos, é apresentar a flexibilidade que a ferramenta possui na inspeção do melhor conjunto de hiperparâmetros a serem utilizados em um problema

particular. Nesse sentido, foram apresentados diversos experimentos, com diferentes hiperparâmetros, que permitem mostrar a evolução dos resultados em um possível ambiente de desenvolvimento de um reconhecedor de fala para o Português. Dessa forma, são apresentados uma metodologia que permite a experimentação de reconhecimento de fala e um classificador treinado em uma base de dados de áudios em Português do Brasil. Além disso, todos os códigos e bases de dados utilizados estão disponibilizados publicamente [5], permitindo a reprodução dos experimentos realizados e a continuação do trabalho para outros possíveis idiomas.

2. A ferramenta Deepspeech

Devido à repercussão positiva das técnicas e resultados obtidos com o sistema de reconhecimento de fala proposto por Hannun *et al.* [1], a equipe da Mozilla decidiu desenvolver um sistema de código fonte aberto [6] para reconhecimento de fala completo, baseado não só nas técnicas presentes no artigo do Baidu em questão, mas em outros estudos, tais como os realizados por Park *et al.* [7] para aumentar a capacidade de generalização dos modelos através de técnicas como *Data Augmentation* dos dados brutos de áudio (sinais e espectrogramas), que consiste na complementação da base com dados artificiais gerados a partir dos dados disponíveis.

A ferramenta *Deepspeech* utiliza o *Tensorflow* desenvolvido pelo Google e, além de oferecer modelos pré-treinados para o inglês e uma lista de APIs em diversas linguagens (C, C#, Java, JavaScript e Python) para facilitar o uso dos modelos em sistemas já em produção, permite que o modelo seja treinado com uma base de dados própria, bem como a configuração de diversos hiperparâmetros do sistema, tais como o número de células em cada camada, taxa de aprendizado e até mesmo o alfabeto e um modelo de linguagem para aumentar a precisão das transcrições.

O *Deepspeech*, por ser uma plataforma customizável para a construção de reconhecedores de fala, que utilizam aprendizado profundo, permite não só o treinamento de modelos totalmente novos, mas também o aproveitamento de modelos pré-treinados,

que possibilitam a construção de reconhecedores em ambientes de vocabulário limitado, por exemplo, melhorando o seu desempenho.

2.1 Redes neurais recorrentes

Redes neurais são um conjunto de modelos matemáticos desenvolvidos tomando como inspiração a estrutura e o funcionamento celular de um neurônio biológico. A origem matemática por trás das redes neurais data por volta de 1940, porém, apenas recentemente, com o grande volume de dados e o avanço do poder computacional, as redes neurais adquiriram um desempenho elevado em tarefas de regressão e classificação [8].

A estrutura básica de uma rede neural começa a partir da definição de um neurônio que possui uma tarefa de implementação simples, ao receber um vetor de entrada x , retornar um valor escalar y . Cada neurônio é definido por três parâmetros: (1) um vetor de pesos w ; (2) um escalar b chamado bias; e (3) uma função de ativação f . A operação realizada em um neurônio é a seguinte:

$$y = f(w \cdot x + b)$$

onde $w \cdot x$ representa o produto escalar entre o vetor de entrada x e os pesos w .

Existem algumas funções de ativação [9] que são mais comuns no desenvolvimento de redes neurais, sendo elas a ReLU ($f(x) = \max\{0, x\}$), a Sigmoid ($\sigma(x) = 1 / (1 + e^{-x})$) e a Tangente Hiperbólica ($\tanh(x) = (e^{2x} - 1) / (e^{2x} + 1)$).

As redes neurais surgem da organização de dezenas, centenas, ou até milhares de neurônios simples em camadas. As redes neurais possuem três tipos de camadas básicas: (1) camada de entrada, responsável por receber os dados de entrada e repassar às camadas seguintes; (2) camada(s) intermediária(s), composta(s) por diversos neurônios, cada um processando os dados de entrada individualmente e repassando os resultados para as camadas seguintes; (3) camada de saída, responsável por redimensionar os dados para a dimensão da saída.

Redes neurais com uma camada escondida são consideradas redes neurais simples e redes com diversas camadas escondidas em sequência são chamadas de redes neurais profundas, ou mais comumente, *deep*

neural networks. Todos os neurônios de uma camada devem possuir a mesma função de ativação, mas camadas diferentes podem ter funções de ativação diferentes.

Durante as rodadas de treinamento de uma rede neural, que normalmente é realizado utilizando o algoritmo *backpropagation*, por intermédio de uma certa taxa de aprendizado, é possível que, por conta de o ajuste dos parâmetros do modelo, ao se buscar minimizar os resultados da função de erro sobre o conjunto de treinamento, o modelo se ajuste tanto a esse conjunto de dados, que não seja capaz de generalizar para dados fora do conjunto de treinamento. Os passos do *backpropagation* são realizados em frações consecutivas do conjunto de treinamento, denominados lotes.

Esse fenômeno de ajuste excessivo é chamado de sobreajuste, ou *overfitting*, sendo o objetivo do modelo, na verdade, ter um bom desempenho sobre os dados do conjunto de teste, e não só sobre o conjunto de treino. Uma das técnicas para prevenir o sobre-ajuste é o *dropout*, que consiste em, durante a etapa de treinamento, escolher uma fração aleatória dos neurônios de cada camada a serem ignorados [10].

Redes neurais recorrentes são um tipo específico de redes neurais, no qual as unidades intermediárias alimentam não apenas as camadas seguintes, mas também a mesma camada ou as anteriores. Ao se alimentar uma célula de uma rede neural recorrente simples, é aplicada uma função de ativação, implicando que, quanto mais a informação flui pela camada intermediária, maior a quantidade de composições de funções de ativação [11].

Tal arquitetura, entretanto, apresenta um problema ao calcular o gradiente descendente devido à regra da cadeia, pois quanto maior a sequência de composição de funções de ativação, menor se torna o gradiente. Ou seja, entradas de iterações mais antigas acabam perdendo o impacto em iterações mais recentes. Esse fenômeno é chamado de esvanecimento do gradiente, do inglês, *vanishing gradient*. Para solucionar esse problema, foi desenvolvida a arquitetura *long short-term memory* (LSTM), que tem o objetivo de manter as informações importantes aprendidas pela rede neural com o passar do tempo, o que é realizado por meio da célula de memória que pode ser observada na **figura 1** [13].

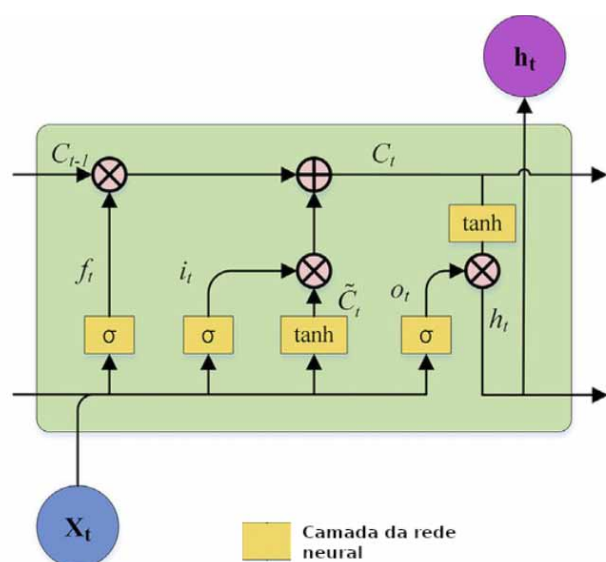


Fig. 1 – Célula LSTM. Fonte: [13].

Simplificadamente, a arquitetura LSTM apresenta três tipos de portas: entrada (i), esquecimento (f) e saída (o). A porta de entrada é responsável pela influência do estado anterior no estado atual, ao passo que a porta de esquecimento controla a parte da informação que deve ser atualizada ou esquecida. Finalmente, a porta de saída controla quais partes da informação devem ser utilizadas na saída da célula [11].

3. Uma proposta para o reconhecimento de fala em português

A proposta deste trabalho é produzir um sistema que seja capaz de permitir experimentos que utilizam modelos de aprendizado profundo baseados em redes recorrentes na área de reconhecimento de fala em língua portuguesa. As etapas seguidas ao longo do trabalho estão retratadas no esquema da **figura 1**. Inicialmente, são definidas as três arquiteturas de rede a serem utilizadas, onde são realizados diversos treinamentos variando seus hiperparâmetros, como forma de compreender como cada configuração pode influenciar nos resultados e para quais valores são observados um melhor desempenho na tarefa. Os hiperparâmetros incluem, por exemplo, número de neurônios, taxa de aprendizado, tamanho do lote, dentre outros.

A experimentação dos hiperparâmetros para cada topologia indica qual a melhor configuração do modelo de classificação correspondente deveria ser utilizada. Em seguida, os três modelos são treinados, variando-se apenas a arquitetura de rede, para que seus resultados possam ser comparados dentro de cada configuração definida. Para cada modelo, as mesmas etapas, de (1) a (5), são executadas. Tais modelos foram treinados e validados a partir da base de dados gratuita Common Voice [14], disponibilizada pela Mozilla. A versão utilizada da base de dados foi a 5.1, que contém aproximadamente 48 horas de áudio validados pela comunidade em arquivos de no máximo dez segundos cada, com 744 diferentes locutores.

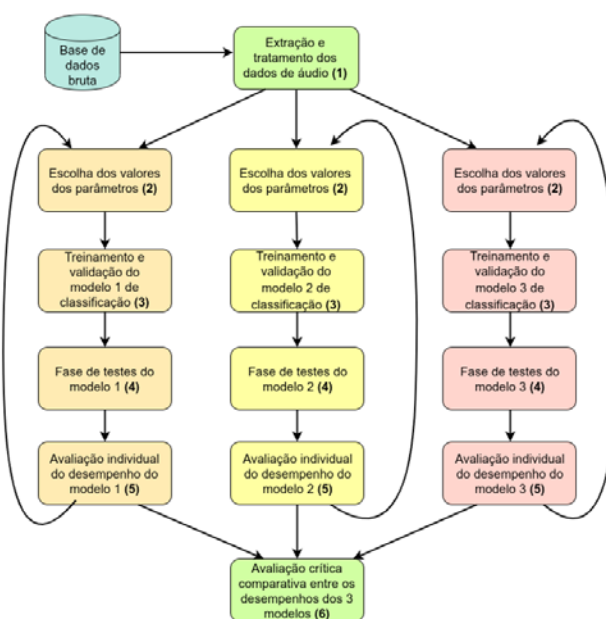


Fig. 2 – Processo de desenvolvimento iterativo do projeto.

As fases do projeto podem ser melhor descritas nas seis seguintes etapas: (1) extração e tratamento dos arquivos de áudio que são utilizados no treinamento e avaliação de cada modelo; (2) escolha dos possíveis valores para os hiperparâmetros dos algoritmos; (3) execução do treinamento e validação do modelo desenvolvido utilizando a ferramenta; (4) testes do modelo utilizando um conjunto de dados diferente; (5) avaliação individual do desempenho de cada modelo;

e (6) realização de uma avaliação crítica e comparativa do desempenho de cada modelo desenvolvido.

As etapas de (2) a (5) se repetem a cada novo modelo proposto, ou seja, para os modelos com topologias diferentes. Já as etapas (1) e (6) são realizadas apenas uma vez, respectivamente, no início do projeto, como um pré-processamento, e ao final, como uma análise dos resultados de forma global. Em cada ramo representado na **figura 2**, o ciclo de execução das etapas (2) a (5) ocorre até que sejam definidos os hiperparâmetros mais adequados para a configuração do modelo.

4. Experimentação na escolha dos hiperparâmetros para a tarefa de reconhecimento de fala

Esta seção relata a realização de diversos experimentos com configurações distintas de hiperparâmetros, para efeitos de comparação dos resultados obtidos e teste do ambiente desenvolvido.

Diversas abordagens podem ser utilizadas para a otimização dos melhores hiperparâmetros para um algoritmo. Buscas manuais e por grids são as técnicas comumente utilizadas na busca de otimização de hiperparâmetros [15]. Neste trabalho, em função do objetivo ser apresentar o ambiente de experimentação, optou-se por realizar os experimentos por intermédio de uma busca gulosa dos hiperparâmetros onde, em cada momento, um deles era otimizado, enquanto os demais se mantinham fixados.

Dessa maneira, pretende-se explorar ao máximo o grau de customização que pode ser aplicado na implementação e arquitetura das redes, de forma a analisar o desempenho sobre a base de dados da língua portuguesa, já que a ferramenta foi desenvolvida com recursos nativos que a princípio facilitam a tarefa para o idioma Inglês. Com isso, espera-se compreender de que forma as alterações nos hiperparâmetros influenciam nos erros, e quais são as melhores configurações para obtenção de melhorias nos resultados.

São realizados experimentos com três modelos distintos, variando-se as topologias da rede. O

primeiro modelo possui 3 camadas totalmente conectadas, 1 camada recorrente utilizando células LSTM e 1 camada totalmente conectada ao fim (3-1-1). Os outros dois modelos apresentam arquiteturas semelhantes, incrementando apenas a quantidade de camadas recorrentes de um para o outro: o segundo modelo com 3 camadas recorrentes (3-3-1) e o terceiro modelo com 5 camadas recorrentes (3-5-1).

Com o objetivo de comparar as estratégias experimentadas, são adotadas três medidas comumente utilizadas nesse contexto: a função de custo, *Word Error Rate* (WER) e o *Character Error Rate* (CER).

A função de custo representa o valor cuja arquitetura de rede neural implementada dentro da ferramenta *DeepSpeech* busca minimizar. Nesse caso, a função de custo consiste, em escala logarítmica, calcular eficientemente a probabilidade inversa de um certo áudio $X = [X_1, X_2, \dots, X_T]$, ser transcrito em uma sequência $Y = [Y_1, Y_2, \dots, Y_U]$. Tal medida está ligada diretamente ao treinamento da rede e seu processo de validação.

A medida WER é uma medida de erro a nível de palavras que considera as suas alterações, inserções e supressões ao se comparar com a saída correta. O WER normalmente penaliza muito os erros, uma vez que apenas um caractere errado em uma palavra a invalida completamente. Nesse caso, também foi considerada a medida CER, que considera os erros a nível de caractere isoladamente.

4.1 Modelo de linguagem

A ferramenta *DeepSpeech* apresenta a implementação de um modelo de linguagem em inglês que pode ser utilizado para refinar os resultados das inferências realizadas, mas que não interfere no treinamento. Para garantir a não interferência do modelo de linguagem nativo do inglês, as *flags*, *lm_alpha* e *lm_beta*, podem ser configuradas para 0, de forma a se manterem nulos os valores de α e β , que regulam o peso da aplicação do modelo de linguagem.

Para os experimentos realizados com o modelo 3-1-1, as flags em questão são zeradas, de modo que nenhum

modelo de linguagem é aplicado à inferência. Para os experimentos com o segundo (3-3-1) e terceiro (3-5-1) modelos de classificação, entretanto, um modelo de linguagem em português é aplicado para que possam ser observadas as diferenças nos resultados obtidos com tal modelo. Espera-se que o modelo de linguagem melhore as previsões dos classificadores. O modelo de linguagem, entretanto, é aplicado apenas na etapa final do reconhecimento e, portanto, não interfere no treinamento dos modelos.

O modelo de linguagem em português foi construído a partir das próprias transcrições contidas na base de dados do Common Voice, utilizando a ferramenta chamada *KenLM* [16], sugerida pela equipe da *DeepSpeech*.

Em relação às flags *lm_alpha*, *lm_alpha_max*, *lm_beta* e *lm_beta_max*, também é feito um processo de otimização, conhecido como ajuste dos hiperparâmetros, com o objetivo de encontrar os melhores valores de α e β que se adequem a cada modelo treinado. Para tal otimização, utiliza-se um script fornecido pela própria ferramenta *DeepSpeech*. Tal código busca os valores das flags, a partir de uma biblioteca chamada Optuna [17].

4.2 Experimentos realizados com o modelo 3-1-1

Aqui são descritos os 9 experimentos realizados para a primeira arquitetura, composta de 3 camadas totalmente conectadas, 1 camada recorrente com células do tipo LSTM e 1 camada totalmente conectada. Ao fim de

cada treinamento, é executada uma inferência com um mesmo subconjunto de transcrições de áudio já vistas pelos modelos, como forma de comparar o desempenho da rede para cada configuração de hiperparâmetros utilizada e observar qualitativamente algum grau de sobre-ajuste.

Os experimentos estão separados em 4 conjuntos, cada um responsável pela análise dos impactos causados pela variação de um dos hiperparâmetros: número de neurônios (a mesma quantidade foi utilizada em todas as camadas), tamanho do lote, taxa de aprendizado ou tamanho da base de dados. Dentro de um mesmo grupo de experimentos, apenas um dos hiperparâmetros varia, mantendo as outras configurações fixas, como pode ser observado no esquema da **tabela 1**. Por fim, foi possível tirar conclusões acerca dos hiperparâmetros que mais favoreceram o desempenho na tarefa de reconhecimento de fala para este modelo de classificação 3-1-1. O último conjunto possui apenas uma configuração I, pois o seu objetivo é realizar a avaliação dos modelos com um tamanho de base maior, utilizando-se os melhores hiperparâmetros encontrados nos conjuntos anteriores.

Após a realização de todos os experimentos e comparação das métricas obtidas, o eleito é aquele que obtiver o melhor desempenho dentre os outros, sendo este o experimento denominado I. Tal experimento é configurado com 512 neurônios, tamanho do lote 10, taxa de aprendizado 0.0005 e base de dados em 50% do seu tamanho.

Tab. 1 – Esquema representativo dos 4 grupos de experimentos realizados, especificando os valores dos hiperparâmetros para cada treinamento do modelo 3-1-1.

Análises	Experimento	Número de neurônios					Tamanho do lote				Taxa de Aprendizado		Tamanho da Base de Dados			
		64	128	256	512	1024	1	5	10	15	0.001	0.0005	1%	10%	25%	50%
Número de neurônios	A	■					■				■			■		
	B		■													
	C			■												
	D				■											
Taxa de Aprendizado	D*				■		■				■			■		
	E											■				
Tamanho do lote	F					■		■			■			■		
	G								■							
	H									■						
Tamanho da Base	I				■				■			■		■		■

A **tabela 2** contém o valor médio do CER, WER e da função de custo obtidos para o experimento I.

Tab. 2 – Resultados obtidos a respeito do CER, WER e da função de custo sobre o conjunto de teste do experimento I no modelo 3-1-1.

Custo	CER	WER
98,55	0,64	1,13

O gráfico da **figura 3** retrata o valor da função de custo ao longo do experimento I, tanto para o conjunto de treino quanto para o conjunto de validação. Observa-se, entretanto, um comportamento não desejável para o erro de validação.

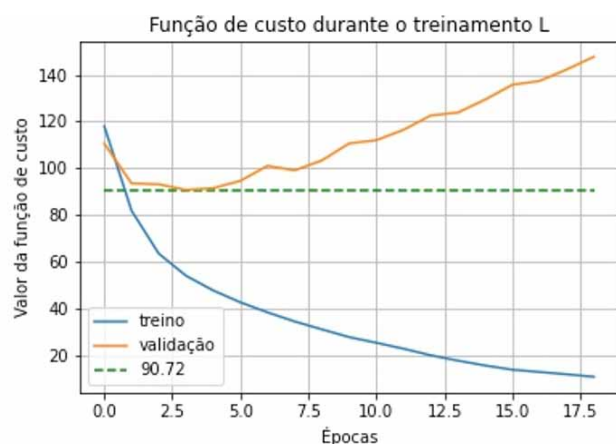


Fig. 3 – Evolução da função de custo para o conjunto de treino e validação ao longo do experimento I no modelo 3-1-1.

4.3 Experimentos realizados com o modelo 3-3-1

Esta subseção descreve os experimentos realizados com o modelo 3-3-1, composto de 3 camadas totalmente conectadas, 3 camadas recorrentes e 1 totalmente conectada, apresentando a discussão teórica a respeito dos resultados obtidos. Os experimentos com esse modelo utilizaram a técnica de *Data Augmentation*, com inserção de ruídos aditivos e multiplicativos no sinal de áudio para aumento do conjunto de treino. Já as inferências, realizadas com os hiperparâmetros mais favoráveis escolhidos ao final, foram executadas com e sem o modelo de linguagem construído em português, para efeitos de visualização dos impactos desse refinamento.

Diferente da metodologia seguida na experimentação com o modelo 3-1-1, os testes a seguir seguem a mesma abordagem gulosa. Os experimentos estão separados em 3 conjuntos, cada um responsável pela análise dos impactos

causados pela variação de um dos hiperparâmetros: número de neurônios, tamanho do lote e taxa de aprendizado. A base de dados completa foi utilizada em todos os experimentos. Dentro de um mesmo grupo de experimentos, apenas um dos hiperparâmetros varia, mantendo-se as outras configurações. O tamanho da base foi fixo em 50%.

Os grupos de experimentos seguem a seguinte ordem: número de neurônios, variando entre 200, 400 e 600; taxa de aprendizado, variando entre 0.001, 0.0005 e 0.0001; tamanho do lote, variando entre 20, 60 e 100. Para o primeiro conjunto de experimentos, quando ainda não se conhecem os melhores hiperparâmetros, foram fixados os valores da taxa de aprendizado em 0.0005 e tamanho do lote 100, fazendo variar apenas a quantidade de neurônios, como pode ser observado na **tabela 3**.

Tab. 3 – Esquema representativo do grupo de experimentos a serem realizados para determinação da quantidade ótima de neurônios, especificando os valores dos hiperparâmetros para cada treinamento no modelo 3-3-1.

		Número de neurónios			Taxa de Aprendizizado			Tamanho do lote		
Análise	Experimento	200	400	600	0.001	0.0005	0.0001	20	60	100
Número de Neurónios A										
	B									
	C									

Após a análise das métricas obtidas, conclui-se que o modelo apresenta seu melhor desempenho com 600 neurônios. Descoberto tal valor para esse hiperparâmetro, segue-se para o segundo grupo de experimentos, aqueles em que a taxa de aprendizado varia nos 3 valores possíveis, fixando o número de neurônios em 600 e o tamanho do lote novamente em 100, conforme a **tabela 4**.

Tab. 4 – Esquema representativo do grupo de experimentos a serem realizados para determinação da melhor taxa de aprendizado, especificando os valores dos hiperparâmetros para cada treinamento no modelo 3-3-1.

Análise	Experimento	Número de neurônios			Taxa de Aprendizado			Tamanho do lote		
		200	400	600	0.001	0.0005	0.0001	20	60	100
Taxa de Aprendizado	D									
	E									

Após a análise das métricas obtidas, conclui-se que o modelo tem seu melhor desempenho com a taxa de

aprendizado 0.001. Tendo em vista o resultado obtido, os experimentos que se seguem foram fixados com 600 neurônios e taxa de aprendizado 0.001, variando o lote entre 20, 60 e 100, conforme o esquema da **tabela 5**.

Tab. 5 – Esquema representativo do grupo de experimentos a serem realizados para determinação do tamanho do lote, especificando os valores dos hiperparâmetros para cada treinamento no modelo 3-3-1.

		Número de neurônios			Taxa de Aprendizado			Tamanho do lote		
Análise	Experimento	200	400	600	0.001	0.0005	0.0001	20	60	100
Tamanho do lote	F									
	G									
	H ou D									

Após a análise das métricas obtidas, conclui-se que o modelo teve seu melhor desempenho com tamanho do lote 100, equivalendo à configuração de hiperparâmetros do experimento D. A **tabela 6** contém o valor médio do CER, WER e da função de custo obtidos para o experimento D.

Tab. 6 – Resultados obtidos a respeito do CER, WER e da função de custo sobre o conjunto de teste do experimento D no modelo 3-3-1.

Custo	CER	WER
93,50	0,56	1,05

O gráfico da **figura 4** retrata o valor da função de custo ao longo do experimento D, tanto para o conjunto de treino quanto para o conjunto de validação. Observa-se, nesse gráfico, o comportamento desejável onde tanto o erro de treinamento quanto o de validação diminuem à medida que o algoritmo realiza o seu treinamento.

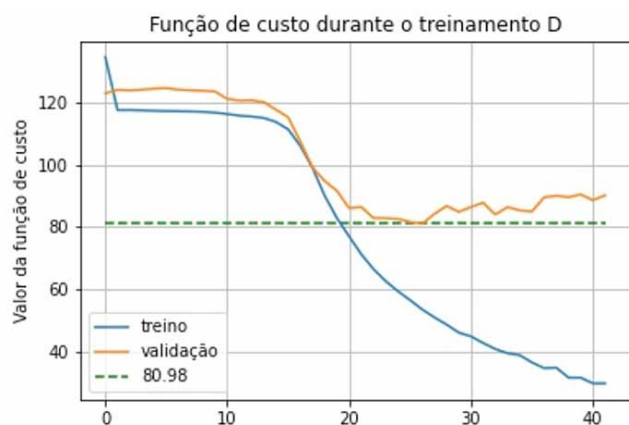


Fig. 4 – Evolução da função de custo para o conjunto de treino e validação ao longo do experimento D no modelo 3-3-1.

Observando os gráficos, nota-se que o experimento D atinge o menor valor de função de custo dentre os comparados, com a curva de validação chegando a 80,98. Isso pode ser confirmado pelas métricas da **tabela 6**, que revelaram o menor valor médio para a função de custo no experimento D, com tamanho de lote 100. De maneira geral, também se percebe um menor valor para o CER no caso do experimento D, reforçando que essa configuração de hiperparâmetros foi a mais favorável para o desempenho do modelo 3-3-1.

Para demonstrar os resultados obtidos nessas condições, foram realizadas inferências com e sem o modelo de linguagem construído em português, como forma de observar os impactos deste para a predição. Vale destacar que tais inferências foram realizadas com um subconjunto de treino. Após a otimização do modelo de linguagem para a arquitetura atual, foram encontrados os valores 0.00483 para a flag α e 0.038042 para a flag β . A **tabela 7** mostra os valores das métricas obtidas nessas inferências, com e sem o modelo de linguagem, para efeitos de comparação.

Tab. 7 – Resultados obtidos a respeito do CER, WER e da função de custo do modelo 3-3-1 para o experimento D, com e sem modelo de linguagem.

Experimento	Custo	CER	WER
D	93,50	0,56	1,05
D+ML		0,59	1,02

Pode ser observada uma leve melhora no WER após aplicação do modelo de linguagem. Outro detalhe é que o valor da função de custo de fato não é alterado, já que o modelo de linguagem não afeta o treinamento, apenas é aplicado na decodificação após a saída do modelo. Um possível fator que pode contribuir para a influência do modelo de linguagem ter sido pouco significativo reside na forma como o modelo foi construído, utilizando transcrições da própria base de dados de entrada, ou seja, um conjunto pequeno. Dessa forma, deve-se levar em consideração que tal base de dados é pequena para o universo do modelo de linguagem, ou seja, a construção de um bom modelo requer muito mais informações para melhorar seu desempenho. Além disso, o tamanho das transcrições pode ser um problema, pois são

frases relativamente curtas, enquanto seria mais adequado utilizar sentenças longas para a construção de um modelo mais preciso, por questões inerentes à forma como as probabilidades são calculadas.

Vale destacar também que o otimizador, em função dos motivos apresentados, encontrou valores muito baixos para as flags α e β , indicando que o modelo de linguagem influencia pouco na saída final do classificador. Caso esse impacto fosse mais efetivo, a otimização retornaria valores mais altos, justamente para dar maior peso ao modelo de linguagem.

4.4 Experimentos realizados com o modelo 3-5-1

Esta subseção descreve os experimentos realizados com o modelo 3-5-1, composto de 3 camadas totalmente conectadas, 5 camadas recorrentes e 1 totalmente conectada, apresentando a discussão a respeito dos resultados obtidos. Assim como no anterior, os experimentos com esse modelo utilizaram a mesma técnica de *Data Augmentation*.

Inicialmente, pretendia-se seguir a mesma abordagem gulosa utilizada para o modelo 3-3-1, separando as análises nos 3 conjuntos, cada um responsável por verificar os impactos causados pela variação de um dos hiperparâmetros: número de neurônios, tamanho do lote e taxa de aprendizado. Os grupos de experimentos seguiriam a mesma ordem: número de neurônios, variando entre 200, 400 e 600; taxa de aprendizado, variando entre

0.001, 0.0005 e 0.0001; tamanho do lote, variando entre 20, 60 e 100. O tamanho da base, como no experimento anterior, foi fixado em 50%.

Entretanto, logo na primeira análise, do número de neurônios, nota-se que o modelo não consegue aprender de maneira satisfatória e apresentar bons resultados, de forma que os treinamentos terminam após 15 épocas sem apresentar melhora com as curvas de treino e validação por volta de 120 em termos de função de custo. De forma visual, é como se existisse uma barreira nesse patamar, que não consegue ser superada, diferente do modelo anterior, em que as curvas ultrapassam esse valor após algumas épocas estáveis naquele nível, como pode ser visto no gráfico da **figura 4**, por exemplo.

Tendo em vista as dificuldades encontradas pelo modelo, para esta terceira arquitetura seguiu-se uma abordagem distinta das anteriores: buscou-se encontrar, a princípio, alguma combinação de número de neurônios e taxa de aprendizado que fosse capaz de melhorar o desempenho do modelo 3-5-1. Para o tamanho do lote, em função do melhor resultado obtido pelo modelo anterior, e pensando também em favorecer os tempos de execução do treinamento, foi mantido seu valor constante em 100. Os experimentos seguiram uma tendência de diminuição do número de neurônios, variando entre 600, 400, 200 e 150. De forma equivalente, as taxas de aprendizado exploradas foram 0.005, 0.001, 0.0005 e 0.0001. A configurações de cada experimento podem ser observadas na **tabela 8**.

Tab. 8 – Esquema representativo do grupo de experimentos a serem realizados para determinação do número de neurônios e taxa de aprendizado, especificando os valores dos hiperparâmetros para cada treinamento no modelo 3-5-1.

Análise	Experimento	Número de neurônios				Taxa de Aprendizado				Tamanho do lote		
		600	400	200	150	0.005	0.001	0.0005	0.0001	20	60	100
Análise Global	A											
	B											
	C											
	D											
	E											
	F											
	G											
	H											

De maneira geral, os gráficos obtidos a respeito da evolução da função de custo para o conjunto de treinamento e validação ao longo dos experimentos foram muito semelhantes, todos mantendo a curva de validação em torno de 120, sem conseguir ultrapassar esta barreira. Como exemplo, a **figura 5** apresenta um desses gráficos para o experimento C.

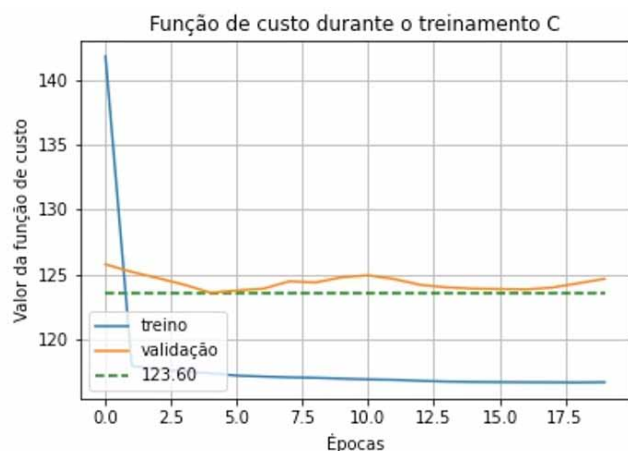


Fig. 5 – Evolução da função de custo para o conjunto de treino e validação ao longo do experimento C, no modelo 3-5-1.

Além disso, a **tabela 9** contém o valor médio do CER, WER e da função de custo para os oito experimentos realizados, como forma de visualização geral dos resultados e comparação.

Tab. 9 – Resultados obtidos a respeito do CER, WER e da função de custo para a análise do número de neurônios e taxa de aprendizado referente ao modelo 3-5-1.

Experimento	Custo	CER	WER
A	128,47	0,77	1,05
B	127,98	0,81	1,02
C	128,27	0,79	1,03
D	127,77	0,78	1,02
E	127,91	0,79	1,04
F	127,75	0,79	1,03
G	128,55	0,78	1,07
H	127,53	0,81	1,01

Comparando-se as métricas obtidas, observa-se que os resultados foram muito semelhantes, de forma que todas as configurações realmente tiveram um desempenho equivalente. Isso pode ser também confirmado pelos gráficos, uma vez que todos seguiram essa mesma tendência e ficaram estacionados com a curva de validação em torno de 120. Diferente do modelo 3-3-

1, que também apresentava inicialmente tal barreira no aprendizado, o modelo 3-5-1 não conseguiu superar este patamar e ampliar sua capacidade de generalização.

Esse comportamento pode estar associado ao fato de o modelo 3-5-1 precisar de uma quantidade muito maior de dados e tempo para treinamento, em função da quantidade de parâmetros a serem aprendidos. Outro detalhe a ser levado em consideração é em relação à base de dados utilizada. Vale destacar que o modelo fica restrito às características e nuances disponíveis na amostra fornecida, de forma que alguns comportamentos são inerentes a esses dados, como por exemplo os patamares observados em ambos os modelos. De maneira geral, então, notou-se que o modelo 3-5-1 não se adaptou bem a essa base, talvez por questões de insuficiência de dados disponíveis para uma arquitetura mais robusta como esta, ou por razões de a topologia utilizada não ter sido a mais adequada para esta tarefa, em razão da presença de muitas camadas recorrentes ou até mesmo do tipo de célula aplicado (LSTM).

4.5 Comparação entre os modelos

Concluídos os experimentos de cada modelo, pode-se realizar uma comparação entre seus desempenhos utilizando os hiperparâmetros mais favoráveis de cada um. Para o modelo 3-1-1, já que foi seguida uma metodologia distinta dos outros, pode-se olhar para todos os experimentos de uma maneira geral. Observando os gráficos, verifica-se que o valor mínimo para a função de custo atingido pelas curvas de validação manteve-se na faixa de 105 para grande parte dos experimentos, alcançando o melhor resultado no experimento I, em 90,72. Quanto às outras métricas, este mesmo experimento também se destaca, com o menor valor médio para a função de custo, valendo 98,55, e para o CER, valendo 0,64. Dessa maneira, como forma de comparar o desempenho do modelo 3-1-1 com os outros, pode-se adotar o experimento I como referência.

Em relação ao modelo 3-3-1, foi eleito o experimento D como o mais eficiente, alcançando

um valor mínimo de 80,98 no gráfico da evolução da curva de validação. As outras métricas obtidas para este também foram muito positivas: 93,5 de Custo, 0,56 de CER e 1,05 de WER.

A experimentação com o modelo 3-5-1, como visto na subseção anterior, não apresenta resultados satisfatórios, de forma que todos os experimentos são equivalentes em seus resultados, impossibilitando a eleição de um melhor. De maneira geral, todos os gráficos estacionam na faixa de 120, e a média da função de custo ficou em 128,03, com o CER em torno de 0,79 e o WER em 1,03.

Sendo assim, os resultados apontam que o segundo modelo, 3-3-1, teve melhor desempenho dentre os demais, apresentando as menores métricas de custo, WER e CER, e revelando em seus gráficos que tal rede foi realmente capaz de aprender mais e superar as barreiras observadas, como aquela no nível 120. Vale destacar que, em testes realizados com a ferramenta DeepSpeech para o modelo 3-1-1 no idioma inglês por Ardila *et al.* [18], por exemplo, verifica-se um alto desempenho do classificador, obtido a partir de uma base de dados em inglês com tamanho 50 vezes maior que a utilizada [14], o que indica que o desempenho do modelo é diretamente relacionado com a quantidade de dados disponibilizados durante o seu treinamento.

Tais resultados apontam, pelo menos inicialmente que utilizar uma arquitetura com mais camadas recorrentes leva a melhores resultados no reconhecimento do Português, diferentemente do que foi reportado para o Inglês na ferramenta.

5. Conclusão

O momento atual revela um crescente uso de inteligência artificial no cotidiano dos indivíduos. Espera-se que, em breve, as pessoas estejam em grande parte inseridas em ambientes dotados de dispositivos inteligentes e robôs, de maneira que a interação entre homens e máquinas fique em evidência. Analisando esse contexto, verifica-se a aplicação do Processamento de Linguagem Natural e sua importância para desenvolver essas interações entre humanos e máquinas. Este trabalho, que foca no reconhecimento

de fala a partir de áudios da língua portuguesa, é uma etapa essencial que pode ser empregada em qualquer projeto que busque melhorar tal relacionamento entre pessoas e dispositivos inteligentes, por exemplo em tarefas que envolvam comandos de voz.

O presente trabalho apresentou o desenvolvimento de um ambiente que possibilita a experimentação de modelos na tarefa de Reconhecimento de Fala em português brasileiro, bem como a implementação de modelos de classificação utilizando redes neurais de aprendizado profundo, para que no fim possa ser feita uma análise comparativa entre seus desempenhos. Em sua primeira fase, foram realizados diversos experimentos variando-se hiperparâmetros de configuração do ambiente, de forma a analisar e compreender como esses valores impactam no desempenho da rede.

Como etapa seguinte, realizaram-se alterações na topologia da rede, sendo necessário não apenas variar hiperparâmetros, mas também realizar modificações no código base dos modelos. Dadas as definições das melhores configurações de hiperparâmetros para cada arquitetura, foi possível avaliar o desempenho de cada um dos modelos e compará-los. Além disso, foi possível estudar o desempenho da ferramenta DeepSpeech sobre a base de dados da língua portuguesa, uma vez que esta foi implementada com recursos nativos que facilitam tal tarefa para a língua inglesa.

A realização dos experimentos permitiu comprovar a eficácia do ambiente que permitiu a realização de diferentes modelagens com configurações de hiperparâmetros para as topologias das redes testadas, bem como a avaliação do desempenho dos classificadores gerados. A dificuldade em se encontrar rapidamente uma configuração que tenha um desempenho ótimo, demonstra as inúmeras possibilidades de emprego do ambiente em conjunto com outros algoritmos ou sendo aplicado na tarefa para outros idiomas.

Como contribuições deste trabalho, um sistema de experimentação de modelos para a tarefa de reconhecimento de fala foi desenvolvido, mostrando

ser possível utilizar tal ferramenta para realização de experimentos em busca de um modelo mais adequado para a língua portuguesa. Além disso, foi apresentado um classificador treinado em uma base de dados de áudios em português do Brasil, sendo o modelo escolhido aquele que obteve melhor desempenho dentre os demais desenvolvidos. O projeto contribuiu também para enriquecer o conhecimento acerca do problema em questão, aumentando a experiência com as ferramentas de aprendizado profundo, manipulação de redes neurais recorrentes, treino do modelo e configuração dos hiperparâmetros.

Esse trabalho permitiu a construção de um modelo base de comparação para o reconhecimento da fala no idioma do Português utilizando a ferramenta DeepSpeech. Futuros trabalhos poderão utilizar não só o modelo, mas a metodologia de experimentação para melhorar os resultados e, com uma base de dados significativa, construir um reconhecedor que possa ser utilizado em tarefas que lidem com o idioma

Português.

No tocante a trabalhos futuros, pretende-se buscar uma base de dados maior em Português para testar as mesmas arquiteturas, e comparar os desempenhos. Além disso, é importante pensar na construção de um modelo de linguagem mais efetivo, utilizando para isso uma base de dados maior que priorize sentenças longas. Mediante a obtenção de resultados melhores, pode-se pensar também na possibilidade de experimentar novas arquiteturas, diferentes tipos de células e outros idiomas. Além disso, pretende-se aplicar o treinamento de tais classificadores em ambientes com muito ruído, criando reconhecedores particulares para certos ambientes ruidosos.

Agradecimentos

Este material é baseado no trabalho apoiado pela Pesquisa Científica do Escritório da Força Aérea sob o número de concessão FA9550-19-1-0020.

Referências bibliográficas

- [1] HANNUN, A.; CASE, C.; CASPER, J.; CATANZARO, B.; DIAMOS, G.; ELSSEN, E.; PRENGER, R.; SATHEESH, S.; SENGUPTA, S.; COATES, A.; NG, A. Y. Deep speech: Scaling up end-to-end speech recognition. ArXiv, v. 1412.5567, 2014. Disponível em: <<https://arxiv.org/abs/1412.5567>>.
- [2] AMODEI, D.; ANANTHANARAYANAN, S.; ANUBHAI, R.; BAI, J.; BATTENBERG, E.; CASE, C.; CASPER, J.; CATANZARO, B.; CHEN, J.; CHRZANOWSKI, M.; COATES, A.; DIAMOS, G.; ELSSEN, E.; ENGEL, J.; FAN, L.; FOGNER, C.; HANNUN, A. Y.; JUN, B.; HAN, T.; LEGRESLEY, P.; LI, X.; LIN, L.; NARANG, S.; NG, A. Y.; OZAIR, S.; PRENGER, R.; QIAN, S.; RAIMAN, J.; SATHEESH, S.; SEETAPUN, D.; SENGUPTA, S.; SRIRAM, A.; WANG, C.; WANG, Y.; WANG, Z.; XIAO, B. S.; XIE, Y.; YOGATAMA, D.; ZHAN, J.; ZHU, Z. Deep speech 2 : End-to-end speech recognition in english and mandarin. ArXiv, abs/1512.02595, 2015. Disponível em: <<http://proceedings.mlr.press/v48/amodei16.pdf>>.
- [3] QUINTANILHA, I. M. End-to-end speech recognition applied to brazilian portuguese using deep learning. Ph. D. dissertation, MSc dissertation, PEE/COPPE, Federal University of Rio de Janeiro, 2017.
- [4] BATTENBERG, E.; CHEN, J.; CHILD, R.; COATES, A.; GAUR, Y.; LI, Y.; LIU, H.; SATHEESH, S.; SEETAPUN, D.; SRIRAM, A.; ZHU, Z. Exploring neural transducers for end-to-end speech recognition. IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), 2017.
- [5] CORPORATION, M. PFC DeepSpeech. 2020. 19 out. de 2020. Disponível em: <<https://github.com/M-Rodrigues/pfc-deepspeech>>.
- [6] MOZILLA. Welcome to DeepSpeech's documentation. 2021. Disponível em: <<https://deepspeech.readthedocs.io/en/r0.9/>>.
- [7] PARK, D. S.; CHAN, W.; ZHANG, Y.; CHIU, C.-C.; ZOPH, B.; CUBUK, E. D.; LE, Q. V. SpecAugment: A simple data augmentation method for automatic speech recognition. Interspeech, ISCA, v. 1st, 2019.
- [8] KROSE, B.; SMAGT, P. van der. An introduction to Neural Network, 1993.
- [9] PROGRAMMERSOUGHT. Activation function. 2018. 18 abr 2020. Disponível em: <<https://www.programmersought.com/article/92601995466/>>.
- [10] SRIVASTAVA GEOFFREY HINTON, e. a. N. Dropout: A simple way to prevent neural networks from overfit-

- ting. Journal of Machine Learning Research 15, v. 6, 2014.
- [11] OLAH, C. Understanding LSTM Networks. 2015. 10 mai. de 2019. Disponível em: <<https://colah.github.io/posts/2015-08-Understanding-LSTMs/>>.
 - [12] EREMENKO, K. Deep Learning A-ZTM: Recurrent Neural Networks (RNN) - Module 3. 2018. Disponível em: <<https://www.slideshare.net/KirillEremenko/deep-learning-az-recurrent-neural-networks-rnn-module-3>>.
 - [13] LI, X.; PENG, L.; YAO, X.; CUI, S.; HU, Y.; YOU, C.; CHI, T. Long short-term memory neural network for air pollutant concentration predictions: Method development and evaluation. Environmental pollution (Barking, Essex: 1987), v. 231, 2017.
 - [14] MOZILLA. Common Voice. 2017. Disponível em: <<https://commonvoice.mozilla.org/pt/datasets>>.
 - [15] Bergstra, J.; Bengio, Y. Random Search for Hyper-Parameter Optimization. Journal of Machine Learning Research, 13, 2012.
 - [16] HEAFIELD, K. KenLM Language Model Toolkit. 2020. Disponível em: <<https://kheafield.com/code/kenlm/>>.
 - [17] OPTUNA. A hyperparameter optimization framework. 2019. Disponível em <<https://github.com/optuna/optuna>>.
 - [18] ARDILA, R.; BRANSON, M.; DAVIS, K.; HENRETTY, M.; KOHLER, M.; MEYER, J.; MORAIS, R.; SAUNDERS, L.; TYERS, F. M.; WEBER, G. Common Voice: A Massively-Multilingual Speech Corpus. 2020.

Agora, você tem mais facilidade em suas mãos.

Acesse. Simule. Contrate.

Sujeito a alteração sem aviso prévio.
Consulte as normas e condições vigentes.



Correção pela
TR, pelo **IPCA**
ou juros
Prefixados

Juros ainda
menores

#fiqueemcasa



www.poupex.com.br

0800 61 3040



PÓS-GRADUAÇÃO NO IME

Bolsas: CAPES, CNPq e FAPERJ

Mestrado

- Engenharia Nuclear
- Engenharia Elétrica
- Engenharia de Transportes
- Sistemas e Computação

Mestrado e Doutorado

- Química
- Ciência dos Materiais
- Engenharia de Defesa



Apoio:



CREA-RJ
Conselho Regional de Engenharia, Arquitetura e Agronomia do Rio de Janeiro

Matrículas: Fluxo Contínuo

Tel.: (21) 2546-7114 – Fax: (21) 2546-7089 – www.ime.eb.br – sd1@ime.eb.br