

*Anais do III Workshop  
de Pesquisa em Computação  
dos Campos Gerais*



**WPCCG'2019**

Gleifer V. Alves, André P. Borges,  
Leandro M. Zatesko e Sheila M. de Almeida



# Prefácio

Este volume contém os artigos apresentados no WPCCG'2019: III Workshop de Pesquisa em Computação dos Campos Gerais, ocorrido nos dias 19 e 20 de setembro de 2019, na Universidade Tecnológica Federal do Paraná (UTFPR), Ponta Grossa - Paraná, Brasil.

O Workshop de Pesquisa em Computação dos Campos Gerais é um evento criado para divulgar pesquisas em desenvolvimento ou concluídas de alunos e docentes de Instituições de Ensino Superior. Organizado por docentes do DAINF (Departamento Acadêmico de Informática) da UTFPR, o evento está em sua terceira edição.

Nesta edição, pesquisadores foram convidados a submeter resumos de uma página ou artigos com quatro a seis páginas, em português ou inglês. Ao todo, o WPCCG'2019 teve 38 submissões. Cada submissão foi avaliada por ao menos dois revisores da respectiva área da Computação. Dos trabalhos submetidos, 31 foram aceitos para apresentação e publicação neste volume.

As publicações do WPCCG têm periodicidade anual e o acesso a elas é livre. O gerenciamento de toda conferência foi feito por meio do sistema *EasyChair*.

26 de setembro de 2019  
Ponta Grossa

Gleifer Vaz Alves  
André Pinz Borges  
Leandro Miranda Zatesko  
Sheila Moraes de Almeida

# Comitê de Programa

Diana Francisca Adamatti	Universidade Federal do Rio Grande - FURG
Simone Aires	Universidade Tecnológica Federal do Paraná - UTFPR
Luiz Carlos Pessoa Albini	Universidade Federal do Paraná - UFPR
Sheila Almeida	Universidade Tecnológica Federal do Paraná - UTFPR
Gleifer Alves	Universidade Tecnológica Federal do Paraná - UTFPR
Mauren Louise Sguario Coelho De Andrade	Universidade Tecnológica Federal do Paraná - UTFPR
Vinícius Andrade	Universidade Tecnológica Federal do Paraná - UTFPR
Marco Antonio Barbosa	Universidade Tecnológica Federal do Paraná - UTFPR
Jean Paul Barddal	Pontifícia Universidade Católica do Paraná - PUC-PR
Flavia Belintani Blum Haddad	Universidade Tecnológica Federal do Paraná - UTFPR
Samuel Bellido Rodrigues	Universidade Tecnológica Federal do Paraná - UTFPR
Claudio Biazus	Universidade Comunitária da Região de Chapecó - UNOCHAPECÓ
Tarcizio Bini	Universidade Tecnológica Federal do Paraná - UTFPR
André Pinz Borges	Universidade Tecnológica Federal do Paraná - UTFPR
Helyane Borges	Universidade Tecnológica Federal do Paraná - UTFPR
Renato Carmo	Universidade Federal do Paraná - UFPR
Simone de Almeida	Universidade Tecnológica Federal do Paraná - UTFPR
Paulo de Geus	Universidade Estadual de Campinas - UNICAMP
Alane Marie de Lima	Universidade Federal do Paraná - UFPR
Jorge Lima de Oliveira Filho	Universidade Estadual de Santa Cruz - UESC
Pedro De Paula	Universidade Tecnológica Federal do Paraná - UTFPR
Leyza Baldo Dorini	Universidade Tecnológica Federal do Paraná - UTFPR
Augusto Foronda	Universidade Tecnológica Federal do Paraná - UTFPR
Luiz Gomes-Jr	Universidade Tecnológica Federal do Paraná - UTFPR
Luís Cassiano Goularte Rista	Pontifícia Universidade Católica do Rio Grande do Sul - PUC/RS
Marina Groshaus	Universidade Tecnológica Federal do Paraná - UTFPR
André L. P. Guedes	Universidade Federal do Paraná - UFPR
Ariangelo Hauer Dias	Universidade Estadual de Ponta Grossa - UEPG
Sediane Hernandes	Universidade Tecnológica Federal do Paraná - UTFPR
Alessandra Hoffman	Universidade Tecnológica Federal do Paraná - UTFPR
Edna Hoshino	Universidade Federal de Mato Grosso do Sul - UFMS
Ionildo José Sanches	Universidade Tecnológica Federal do Paraná - UTFPR
Nadia Kozevitch	Universidade Tecnológica Federal do Paraná - UTFPR
Evandro Kuszera	Universidade Tecnológica Federal do Paraná - UTFPR
Paulo Leitão	Instituto Politécnico de Bragança - IPB
Raphael Machado	Instituto Nacional de Metrologia, Qualidade e Tecnologia - Inmetro e Clavis

Neumar Malheiros	Universidade Federal de Lavras - UFLA
Luciene Marin	Universidade Tecnológica Federal do Paraná - UTFPR
Priscila Martins	Universidade Federal de Mato Grosso do Sul - UFMS
Simone Matos	Universidade Tecnológica Federal do Paraná - UTFPR
Tania Monteiro	Universidade Tecnológica Federal do Paraná - UTFPR
Erikson Morais	Universidade Tecnológica Federal do Paraná - UTFPR
Carlos Pantoja	Centro Federal de Educação Tecnológica do Rio de Janeiro - CEFET/RJ
Vagner Pedrotti	Universidade Federal de Mato Grosso do Sul - UFMS
Richard Ribeiro	Universidade Tecnológica Federal do Paraná - UTFPR
Andre Roberto Ortoncelli	Universidade Tecnológica Federal do Paraná - UTFPR
Lucio Rocha	Universidade Tecnológica Federal do Paraná - UTFPR
Giancarlo Rodrigues	Universidade Estadual de Ponta Grossa - UEPG
Marilton Sanchotene de Aguiar	Universidade Federal de Pelotas - UFPEL
Celso Santos	Universidade Estadual de Campinas - UNICAMP
Diana Sasaki	Universidade do Estado do Rio de Janeiro - UERJ
Dênis Lucas Silva	Universidade Tecnológica Federal do Paraná - UTFPR
Murilo Silva	Universidade Federal do Paraná - UFPR
Leandro Zatesko	Universidade Tecnológica Federal do Paraná - UTFPR
Alesom Zorzi	Mendelics

# Sumário

## Seção 1: Resumos

Redução de Dimensionalidade em Base de Dados de Microarranjos . . . . .	1
Rafael de Melo e Helyane Borges . . . . .	
Reconhecimento de Fonemas utilizando Redes Neurais Convolucionais para Transcrição Fonética Automática . . . . .	2
Bauke Dijkstra e Ionildo Sanches . . . . .	
Método de Extração de Atributos para Classificação Hierárquica Multirrótulo usando Autoencoders . . . . .	3
Rafael Fernandes Siqueira e Helyane B. Borges . . . . .	
Sistema de gerenciamento visual de múltiplas câmeras IP pan-tilt . . . . .	4
João Eduardo Fornazari e Erikson Freitas de Moraes . . . . .	
A Bioinformatics Approach to Detect Plagiarism in Source Code . . . . .	5
Kaio Pablo Gomes e Simone Matos . . . . .	
Coloração de Arestas em Grafos Split-co-comparabilidade . . . . .	6
Luis Gustavo Da Soledade Gonzaga e Sheila Moraes de Almeida . . . . .	
Detecção de objetos ilícitos em imagens de raio-X de bagagens em scanners de aeroportos . . . . .	7
Gabriel Oliveira Camilo e Simone Bello Kaminski Aires . . . . .	
Refatorção Baseada em Padrões de Projeto Usando Agentes . . . . .	8
Thyago Henrique Pacher e Simone Nasser Matos . . . . .	
Identificação de falhas em linhas de plantio utilizando imagens aéreas obtidas por VANT's . . . . .	9
Enzo Salvadori e Simone Aires . . . . .	

## Seção 2: Artigos

Calibração de câmeras térmicas usando pastilhas Peltier . . . . .	10
Othon Alberto da Silva Briganoé e Erikson Freitas de Moraes . . . . .	
Security in multi agent systems . . . . .	17
Bruno Rafael Alves, André Pinz Borges, Gleifer Vaz Alves e Paulo Leitão . . . . .	
Análise de Perfis de Doenças com Base em Técnicas de Descoberta de Conhecimento em Bases de Dados . . . . .	24
André Pinz Borges, Richardson Ribeiro, Kaique Augusto Moraes da Silva, Marcelo Teixeira, Dalcimar Casanova e Rodrigo Feuser . . . . .	
Mecanismo de Negociação Descentralizado entre Agentes em um Estacionamento Inteligente . . . . .	31
Felipe Felix Ducheiko, Gleifer Alves e André Pinz Borges . . . . .	
Um modelo de rede neural siamesa para re-identificação de pessoas em imagens, utilizando rede neural convolucional e autoencoder . . . . .	37
Fábia Enembreck e Erikson Moraes . . . . .	
Uma proposta de Gerenciamento de Consumo Hídrico para a IoT . . . . .	43
Laryssa Ferreira, Maria Alice Trinta, Carlos Pantoja e Vinicius Souza de Jesus . . . . .	

Desenvolvimento de uma ontologia para representação de objetos e ações do código de trânsito em um veículo autônomo . . . . .	49
Vithor Ferreira e Gleifer Alves . . . . .	
Performance Analysis of SDN Virtualization . . . . .	56
Davi Daniel Gemmer e Augusto Foronda . . . . .	
Abordagem Entrópica de Codificação de Fonte Usando Compressão de Dados . . . . .	62
Kaio Pablo Gomes, Eduardo Sallum El Akkari e Everton Leonardo Skeika . . . . .	
On Subclasses of Circular Arc Bigraphs . . . . .	68
Marina Groshaus, André L. P. Guedes e Fabricio Schiavon Kolberg . . . . .	
Uma Revisão sobre a Relação de BQP com outras Classes de Complexidade Computacional . .	74
Henrique Hepp, Murilo da Silva e Leandro Zatesko . . . . .	
Um Framework para o Teste Automatizado de Sensores . . . . .	80
Darlan Klotz, Marco Tulio Guerreiro, Richardson Ribeiro e Marcelo Teixeira . . . . .	
Estimando Centralidade de Percolação utilizando Amostragem e Teoria da Dimensão Vapnik-Chervonenkis . . . . .	86
Alane Lima, Giovanna Santos, André Vignatti e Murilo da Silva . . . . .	
A Possibility for Handwriting Trajectory Reconstruction with Deep Learning . . . . .	92
Yu Han Lin, Erikson Freitas de Moraes e Simone B. K. Aires . . . . .	
Uma Revisão dos Métodos de Implementação de Sistemas Multiagentes em Ambientes Físicos com Foco em Artefatos . . . . .	98
Fabian Cesar Pereira Brandão Manoel e Carlos Eduardo Pantoja . . . . .	
Aparecimento de cliques em grafos aleatórios lei de potência no modelo de ligação preferencial	104
David Reksidler Júnior e Murilo da Silva . . . . .	
Criticalidade arco-íris em rodas e leques . . . . .	110
Aleffer Rocha e Sheila Almeida . . . . .	
Planejamento de caminhos 3D: Comparação dos algoritmos por grade 3D e grafo de visibilidade	116
Cleiton A. Santos e André L. P. Guedes . . . . .	
Implementação de um Sistema de Reconhecimento de Voz em Protótipos para a Comunicação LIBRAS-Língua Portuguesa . . . . .	122
Maria Alice Trinta, Vinicius Jesus e Carlos Eduardo Pantoja . . . . .	
Implantação de uma arquitetura ciber-física baseada em agentes para um Smart Parking . . . .	128
Pedro Warmling Botelho, Gleifer Vaz Alves e André Pinz Borges . . . . .	
$P = NP$ or 5-Snarks Exist . . . . .	135
Leandro Zatesko e Éverton Vieira . . . . .	

# Índice de Autores

Aires, Simone	7, 9, 92
Alberto da Silva Briganoó, Othon	10
Almeida, Sheila	6, 110
Alves, Bruno Rafael	17
Alves, Gleifer	17, 31, 49, 128
Alves, Gleifer Vaz	17, 31, 49, 128
Augusto Moraes da Silva, Kaique	24
B. Borges, Helyane	3
B. K. Aires, Simone	7, 9, 92
Bello Kaminski Aires, Simone	7, 9, 92
Borges, André Pinz	17, 24, 31, 128
Borges, Helyane	1
Casanova, Dalcimar	24
da Silva, Murilo	74, 86, 104
de Melo, Rafael	1
Dijkstra, Bauke	2
Ducheiko, Felipe Felix	31
Eduardo Pantoja, Carlos	43, 98, 122
El Akkari, Eduardo Sallum	62
Enembreck, Fábila	37
Fernandes Siqueira, Rafael	3
Ferreira, Laryssa	43
Ferreira, Vithor	49
Feuser, Rodrigo	24
Fornazari, João Eduardo	4
Foronda, Augusto	56
Freitas de Moraes, Erikson	4, 10, 37, 92
Gemmer, Davi Daniel	56
Gomes, Kaio Pablo	5, 62
Gonzaga, Luis Gustavo Da Soledade	6
Groshaus, Marina	68
Guedes, André L. P.	68, 116

Hepp, Henrique	74
Jesus, Vinicius	122
Klotz, Darlan	80
Leitão, Paulo	17
Lima, Alane	86
Lin, Yu Han	92
Matos, Simone	5, 8
Matos, Simone Nasser	5, 8
Morais de Almeida, Sheila	6, 110
Morais, Erikson	4, 10, 37, 92
Oliveira Camilo, Gabriel	7
Pacher, Thyago Henrique	8
Pantoja, Carlos	43, 98, 122
Pantoja, Carlos Eduardo	43, 98, 122
Pereira Brandão Manoel, Fabian Cesar	98
Pinz Borges, André	17, 24, 31, 128
Reksidler Júnior, David	104
Ribeiro, Richardson	24, 80
Rocha, Aleffer	110
Salvadori, Enzo	9
Sanches, Ionildo	2
Santos, Cleiton A.	116
Santos, Giovanne	86
Schiavon Kolberg, Fabricio	68
Skeika, Everton Leonardo	62
Souza de Jesus, Vinicius	43
Teixeira, Marcelo	24, 80
Trinta, Maria Alice	43, 122
Tulio Guerreiro, Marco	80
Vaz Alves, Gleifer	17, 31, 49, 128
Vieira, Éverton	135
Vignatti, André	86
Warmling Botelho, Pedro	128
Zatesko, Leandro	74, 135

# Índice de Palavras-chave

Agentes	8, 128
Agricultura de precisão	9
Ambiente inteligente	43
Ambientes Físicos	98
Approximation Algorithm	86
Artefatos	98
Artificial neural networks	37
ASCII	62
Autocodificadores	1
Autoencoders	3
Automatic control	80
Automation	80
Autonomous Vehicles	49
BDI	8
Binarização.	9
Bioinformatics method	5
BPP	74
BQP	74
Caixas d'água	43
Calibração de câmera	10
Certificate Authority	17
Channel Efficiency	62
Characterization	68
Circular arc bigraphs	68
Circular convex bipartite graphs	68
Classificação Hierárquica Multirrótulo	3
Cliques	104
Coloração de arestas	6
Coloração em grafos	6
Colouring of graphs and hypergraphs (MSC 05C15)	135
Complexidade Computacional	74
Computational difficulty of problems (MSC 68Q17)	135
Computação Quântica	74
Computação: Utilizando computadores para produzir e processar imagens.	9
Consumo Hídrico	43
Container	56

Câmera térmica	10
Câmeras IP	4
Câmeras pan-tilt-zoom	4
Data Transmission	62
Deep learning	2
Deep Learning	37, 92
Descoberta de Conhecimento	24
Design Patterns	8
Deteção de Objetos Ilícitos	7
Entropic coding	62
Esqueletização.	9
Estacionamento Inteligente	128
Fan graphs	110
Formal Modeling	80
Gerenciamento de câmeras	4
Grade 3D	116
Grafo de visibilidade 3D	116
Grafos split	6
Graph classes	68
Graph theory in relation to Computer Science (MSC 68R10)	135
Handwriting Reconstruction	92
Handwriting Trajectory Recovery	92
Helly	68
Huffman	62
Image Recovery	92
Internet das Coisas	128
Internet das coisas	43
Javino	43
LIBRAS	122
Local Area Networks	62
luva de aprendizado	122
luva de comunicação	122
língua portuguesa	122
Microarranjos	1
Mineração de Dados	24
Mineração de dados	1
Multi Agent System	17
Multi-Agent Systems	31
Negotiation Protocol	31
NP	74
Ontologies	49

Percolation Centrality	86
Performance	56
Person re-identification	37
Planejamento de caminhos 3D	116
Power Law Graphs	104
Processamento acústico	2
Processamento de imagens.	9
Programming plagiarism	5
Prontuário Eletrônico do Paciente.	24
Rainbow critical graphs	110
Rainbow criticality	110
Raio-X de Bagagens	7
Random Graphs	104
Raspberry	43
Reconhecimento de fonemas	2
Redes Neurais Convolucionais	7
Redução de Dimensionalidade	3
Redução de dimensionalidade	1
Refatoração	8
SDN	56
Shannon-Fano	62
Sistema de gerenciamento	4
Sistema de reconhecimento de voz	122
Sistema Multiagente	98
Sistemas Ciber-Físicos	128
Smart Parking	17, 31
Source code	5
SSL	17
Traffic Rules	49
Transformada de Hough.	9
VANT: Uso de VANTS para produzir imagens aéreas.	9
VC-Dimension	86
Virtual Machine	56
Visão Computacional	10
Wheel graphs	110

# Redução de Dimensionalidade em Base de Dados de Microarranjos

Rafael Felipe Tasaka de Melo<sup>1</sup>, Helyane Bronoski Borges<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR)  
Rua Doutor Washington Subtil Chueire, 330, Jardim Carvalho,  
CEP 84017-220 - Ponta Grossa - PR – Brasil

rafaelftmelo@outlook.com, helyane@utfpr.edu.br

**Palavras-chave:** Mineração de dados, Redução de dimensionalidade, Microarranjos, Autocodificadores

**Resumo.** Algoritmos de Aprendizagem de Máquina vem sendo cada vez mais utilizados pela sua capacidade de aprender a partir de grandes volumes de dados, como, por exemplo, dados de expressão gênica obtidos pela técnica de microarranjo. Uma característica das bases de dados de microarranjos é que, geralmente, ela é formada por grande quantidade de atributos - que correspondem aos genes - e um pequeno número de amostras. Sabe-se que dados com alta dimensionalidade podem possuir atributos redundantes e muitas vezes irrelevantes - podendo atrapalhar o processo de aprendizagem e o desempenho das predições. Métodos de redução de dimensionalidade são utilizados para reduzir a quantidade de atributos das bases de dados. Esses podem ser divididos em duas abordagens principais: a seleção e a extração de atributos. A seleção busca selecionar os atributos mais relevantes e não altera a base de dados original, já a extração possui o mesmo propósito, contudo ela altera os valores dos dados. Redes Neurais Autocodificadoras podem ser adaptadas e utilizadas para a extração de atributos e, conseqüentemente, a redução da dimensionalidade. Esta pesquisa tem como objetivo adaptar uma rede neural autocodificadora para ser utilizada na extração de atributos em bases de dados de microarranjo. Para isso, serão realizados experimentos em cinco bases de dados. Os resultados serão avaliados por meio da taxa de acerto dos classificadores: KNN, SVM, Naive Bayes e Árvore de Decisão e validados estatisticamente.

# Reconhecimento de Fonemas utilizando Redes Neurais Convolucionais para Transcrição Fonética Automática

Bauke Alfredo Dijkstra<sup>1</sup>, Ionildo José Sanches<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR)  
Av. Monteiro Lobato, s/n - Km 04, CEP 84016-210 - Ponta Grossa - PR – Brasil

{baukead, ijsanches}@gmail.com

**Palavras-chave:** Reconhecimento de fonemas; Deep learning; Processamento acústico

**Resumo.** Este trabalho tem como objetivo desenvolver uma técnica de reconhecimento automático de fonemas (RAF) de forma contínua. O reconhecimento de fonemas é a capacidade de extrair características para reconhecer as unidades sonoras das palavras e transcrevê-las. O RAF tem aplicações em diversas áreas como o reconhecimento automático de fala (ASR - *Automatic Speech Recognition*), identificação de locutores, identificação de erros de pronúncia e reconhecimento de emoções. O RAF consiste de três modelos principais de fala: palavras isoladas, palavras concatenadas e da fala contínua. Conforme (Ashwini Kshirsagar, et al. Comparative Study of Phoneme Recognition Techniques. 2012) o reconhecimento de fonemas funciona melhor para palavras isoladas do que em fala contínua. Para realizar o RAF pode-se realizar um pré processamento dos áudios denominado processamento acústico para minimizar ruídos e diferenças entre locutores, e então realizar a classificação dos fonemas utilizando algoritmos de aprendizagem de máquina, com o objetivo de identificar os fonemas. No desenvolvimento desse projeto os testes serão realizados com fonemas extraídos das bases de áudios como a TIMIT *Acoustic-Phonetic Continuous Speech* Corpus que é uma base com fala em inglês e possui transcrições ortográficas, fonéticas e de palavras, e em bases com fala em português como a VoxForge, Sid e LABSM1.4. As bases em português são apenas transcritas em forma ortográficas, portanto torna-se necessário utilizar um programa de grafema para fonema para formar os fonemas em relação ao texto, e por fim validar de forma manual a transcrição de acordo com os áudios. Para realizar o RAF é necessário realizar o processamento acústico dos áudios dessas bases, que podem ser feitas utilizando o janelamento de Hamming e depois aplicar o MFCC (*Mel Frequency Cepstral Coefficients*). Após o processamento acústico é realizada a classificação dos fonemas. Nesse trabalho será utilizado o *framework* Keras para *Deep learning*, uma API (*Application Programming Interface*) de alto nível para o TensorFlow, para realizar os treinamentos com as bases citadas. Por fim, os resultados serão comparados e analisados para avaliar o modelo obtido com a taxas de reconhecimento.

# Método de Extração de Atributos para Classificação Hierárquica Multirrótulo usando Autoencoders

Rafael Fernandes Siqueira<sup>1</sup>, Helyane Bronoski Borges<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR)  
Av Monteiro Lobato, s/n - Km 04, CEP 84016-210 - Ponta Grossa – PR – Brasil

rafaelsiqueira@alunos.utfpr.edu.br, helyane@utfpr.edu.br

**Palavras-chave:** Classificação Hierárquica Multirrótulo; Redução de Dimensionalidade; Extração de Atributos; Autoencoders.

**Resumo.** A predição de proteínas em dados de bioinformática é um exemplo de problema de Classificação Hierárquica Multirrótulo no qual cada instância pode estar associada a múltiplas classes, e estas por sua vez, estão organizadas em uma hierarquia. A alta dimensionalidade dos atributos e das classes influencia no desempenho dos classificadores, tanto no custo computacional quanto na performance, pois prejudica a busca por padrões e extração de conhecimento útil. A extração de atributos é uma das técnicas de redução de dimensionalidade em base de dados, que busca eliminar atributos irrelevantes e/ou redundantes que tendem a confundir o algoritmo de aprendizagem. Nessa técnica, por meio de combinações e/ou transformações dos atributos originais, geram-se novos atributos em um espaço de menor dimensão, que são mais significativos e que melhor representam a base de dados original. Desse modo, propõe-se um método de extração de atributos para classificação hierárquica multirrótulo, FEAE-HMC (*Feature Extraction based on Autoencoders for the Hierarchical Multi-label Classification*), baseado em conceitos de Deep Learning e adaptações em uma rede *Autoencoder* tradicional. Após o treinamento da rede, com o ajuste dos pesos, é possível reduzir a dimensionalidade das amostras da base de dados. O método é dividido em duas etapas principais: a extração de atributos e a avaliação do conjunto de dados reduzido, por meio de um classificador hierárquico multirrótulo e sua medida de desempenho. Nos experimentos são utilizados dados biológicos de 10 bases de dados da Ontologia Gênica, sendo a hierarquia das classes estruturada como um Grafo Acíclico Dirigido (DAG). As bases extraídas pelo FEAE-HMC, quando submetidas em um Classificador Hierárquico Multirrótulo, geram modelos nos quais se obtêm o desempenho preditivo, em relação a medida AUPRC, equivalente e até mesmo superior ao desempenho da base original.

# Sistema de gerenciamento visual de múltiplas câmeras IP pan-tilt-zoom

João Eduardo Kozan Silva Fornazari, Erikson Freitas de Morais

<sup>1</sup>Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR)

{joaofornazari, emorais}@utfpr.edu.br

**Palavras-chave:** Gerenciamento de câmeras, Câmeras pan-tilt-zoom, Câmeras IP, Sistema de gerenciamento

**Resumo.** Em Visão Computacional, o monitoramento preciso de ambientes é crucial. Realizando monitoramento adequadamente, é possível gravar, observar ou realizar os experimentos de Visão Computacional com exatidão, tendo todos os dados do experimento arquivados digitalmente. Contudo, adversidades podem surgir no monitoramento por câmeras, dentre elas o gerenciamento das câmeras. Se o gerenciamento for manual, o pesquisador precisa deslocar-se até a câmera, gastando tempo e, em certos casos, expondo-se a riscos de segurança. Um exemplo é o Laboratório de Computação Aplicada (LaCA) da UTFPR, campus Ponta Grossa, que possui oito câmeras dispostas próximas ao teto do laboratório, a aproximadamente três metros do chão. Estas câmeras são controladas por programas em C, C++ e json respectivos a cada câmera. Isto também atrasa o desenvolvimento das pesquisas, visto que utiliza-se várias câmeras simultaneamente. Com um sistema de controle de várias câmeras, a intervenção manual e individual torna-se desnecessária. Desta forma, propõe-se um modelo de sistema de gerenciamento automático e visual de múltiplas câmeras, onde seu operador pode desligá-las, posicionar seus campos visuais e, em trabalhos futuros, executar tarefas mais complexas, como por exemplo acompanhar objetos em cena. O sistema será desenvolvido em Qt, toolkit de desenvolvimento de interfaces que possui suporte para as linguagens dos programas das câmeras e é compatível com sistemas operacionais MacOS, Windows e Linux. O esquema de monitoramento de Bhan feito em 2012, o sistema integrado de múltiplas câmeras de Kim (2015) e o método de controle automático de múltiplas câmeras de Yang (2017) são exemplos de sistemas que aumentaram a praticidade e agilidade no controle de câmeras. As principais contribuições do trabalho são: o descarte da necessidade de locomoção, a adaptabilidade do desenho do modelo teórico proposto e a visualização e controle de múltiplas câmeras ao mesmo tempo. O ambiente de desenvolvimento do trabalho é o LaCA da UTFPR – Ponta Grossa.

# A Bioinformatics Approach to Detect Plagiarism in Source Code

Kaio P. Gomes<sup>1</sup>, Simone N. Matos<sup>2</sup>

<sup>1</sup>Department of Informatics  
Federal University of Technology - Parana (UTFPR) – Ponta Grossa, PR – Brazil

<sup>2</sup>Department of Informatics  
Federal University of Technology - Parana (UTFPR) – Ponta Grossa, PR – Brazil

kgomes@alunos.utfpr.edu.br, snasser@utfpr.edu.br

**Palavras-chave:** Programming plagiarism; Bioinformatics method, Source code

**Abstract.** The concern about plagiarism is growing in the programming scope, and new solutions are being proposed to handle the detection of plagiarized source code. Such an issue affects not only in the education field but also in the industries dealing with software development. For instance, at the academic level, the students can act with dishonesty on introductory programming course assignments, and at the industry level, the software reuse can be common practice even though using someone else work or ideas without authorization. Nowadays, several higher education institution has been adopting the usage of automatic tools for detecting academic dishonesty on its programming and related courses, and the outcomes reveal that the incident of plagiarism chopped considerably. Thus, New approaches based on different methods and techniques are being developed to address the programming plagiarism problem for the diverse emerging demand. According to the systematic mapping review in the present research, there are different approaches to detect plagiarism in source code. For instance, methods based on token, N-Gram, birthmark, Fuzzy, Bytecode, metric, graph, tree and others. The main solutions try to contemplate the most diverse types of programming plagiarism as well as improving the time complexity of its algorithms. This research proposes a novel approach based on a bioinformatics inspired method to improve plagiarism detection performance and accuracy. The main goal of this proposal is dealing with different types of modifications on plagiarized source code following the classification of programming plagiarism elaborated by [Faidhi and Robinson 1987]. The proposed approach works with the premise of modeling a source code into a synthetic DNA sequence and executing alignment among these types of sequence to identify similarities. The identified similarities correspond to a percentage rate, which indicates how similar source codes can be. This approach seeks to obtain a less computational cost in terms of time complexity comparing to consolidated tools available for this field such as JPLAG. The evaluation of this study considers test scenarios experimentations, which include 223 source codes collected from programming course assignments and others 30 created manually to insert different specific types of plagiarism.

# Coloração de Arestas em Grafos Split-co-comparabilidade

Luis Gustavo da Soledade Gonzaga<sup>1</sup>, Sheila Morais de Almeida<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR)  
Av Monteiro Lobato, s/n - Km 04, CEP 84016-210 - Ponta Grossa - PR – Brasil

**Palavras-chave:** Coloração de arestas; Grafos split; Coloração em grafos

**Resumo.** Uma coloração de arestas própria em um grafo  $G$  é uma atribuição de cores para as arestas de  $G$  tal que arestas incidentes ao mesmo vértice têm cores distintas. O Problema da Coloração de Arestas é responder, dado um grafo  $G$ , qual o menor número de cores para uma coloração de arestas própria de  $G$ . Esse número é chamado de índice cromático de  $G$  e denotado  $\chi'(G)$ . Por definição,  $\chi'(G) \leq \Delta(G)$  para todo grafo simples. Segundo Vizing *On an estimate of the chromatic class of a  $p$ -graph* (1964),  $\chi'(G) \leq \Delta(G) + 1$  para qualquer grafo simples  $G$ . Portanto, quando  $G$  é simples  $\Delta(G) \leq \chi'(G) \leq \Delta(G) + 1$ . Um grafo  $G$  é Classe 1 se  $\chi'(G) = \Delta(G)$  ou Classe 2 se  $\chi'(G) = \Delta(G) + 1$ . O Problema da Classificação é decidir se um grafo simples  $G$  é Classe 1.

Johnson *The NP-completeness column: an ongoing guide* (1985) classificou problemas da teoria dos grafos ainda não solucionados quanto à dificuldade de uma possível solução. Um dos problemas enunciados como possivelmente fácil é o Problema da Classificação para grafos split. Entretanto esse problema permanece em aberto. Um grafo split é um grafo em que os seus vértices podem ser particionados em uma clique e um conjunto independente. Em Chen, Fu e Ko *Total chromatic number and chromatic index of split graphs* (1995) o Problema da Coloração de Arestas foi resolvido quando  $G$  é um grafo split com  $\Delta(G)$  ímpar ou um split completo (aquele em que todo vértice da clique é adjacente a todo vértice do conjunto independente). Almeida *Coloração de arestas em grafos split* (2012) apresentou alguns resultados parciais quando o grafo split  $G$  tem  $\Delta(G)$  par. Cruz et al. *The Overfull Conjecture on Split-Comparability Graphs* (2017) resolveram o Problema da Coloração de Arestas para os split-comparabilidade. Todos esses resultados são acompanhados de algoritmos polinomiais para solução do Problema da Classificação. A classe dos grafos split é fechada para a operação de complemento. O complemento de um grafo de comparabilidade é chamado de grafo de co-comparabilidade. Assim, chamamos o complemento de um grafo split-comparabilidade de split-co-comparabilidade. Para esta família de grafos, o Problema da Classificação permanece em aberto. A classe split-co-comparabilidade é equivalente à classe split-intervalos. Uma das subclasses dos grafos de intervalos é a classe dos grafos indiferença. Ortiz et al. *Characterizing and edge-colouring split-indifference graphs* (1998) apresentam solução polinomial para os problemas da Coloração de Arestas e da Classificação em grafos split-indiferença. Pretende-se verificar se é possível adaptar pelo menos uma das técnicas utilizadas na coloração de arestas dos grafos split-indiferença e dos grafos split-comparabilidade para a solução do Problema da Classificação nos split-co-comparabilidade. A escolha dessas técnicas para estudo deve-se à similaridade da estrutura entre as classes de grafos resolvidas e a classe em estudo.

# Detecção de objetos ilícitos em imagens de raio-X de bagagens em scanners de aeroportos

Gabriel Oliveira Camilo<sup>1</sup>, Simone Bello Kaminski Aires<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR)  
Av Monteiro Lobato, s/n - Km 04, CEP 84016-210 - Ponta Grossa – PR – Brasil

g.oliveiracamilo@gmail.com,  
sbkaminski@utfpr.edu.br

**Palavras-chave:** Redes Neurais Convolucionais; Raio-X de Bagagens; Detecção de Objetos Ilícitos.

**Resumo.** Garantir a segurança de aeroportos, no território brasileiro, é prioridade dentro da Agência Nacional de Aviação Civil (ANAC). Uma das atividades para manutenção da segurança de ambientes internos é a utilização de Raio-X para detectar objetos que possam apresentar algum risco aos passageiros, como armas, facas e explosivos. A detecção de objetos é realizada por operadores humanos, com o auxílio de máquinas computadorizadas que geram as imagens do interior de bagagens. Fatores como habilidade, aptidão, treinamento e falta de atenção podem influenciar os operadores a não realizarem essa tarefa com precisão. A automatização desse trabalho pode se tornar relevante no contexto da detecção, para que fatores humanos não exerçam tanta influência nesse trabalho. Automatizar esse processo apresenta a dificuldade em desenvolver algoritmos capazes de detectar objetos específicos nas imagens geradas através do uso de raio-X. Com o avanço da *deep learning* e o surgimento de novas técnicas voltadas para o reconhecimento de padrões, como as Redes Neurais Convolucionais (RNC), é possível observar, na literatura, ganhos significativos na área de detecção de objetos em imagens. Estudos propõem metodologias compostas por RNCs para detecção de objetos em imagens de raio-X, entretanto existem cenários nos quais os trabalhos já publicados ainda não obtiveram êxito. Esse trabalho tem como objetivo contribuir com os estudos já existentes propondo uma metodologia utilizando RNCs para detectar objetos em imagens de raio-X de bagagens em aeroportos. Os experimentos serão realizados em cima da base SIXray, que contém 1.059.231 imagens de raio-X de diferentes tipos de bagagens, sendo 8.929 compostas por facas, armas, chaves inglesas, alicates, tesouras e martelos. Inicialmente serão necessárias algumas etapas de pré-processamento das imagens, como realce, segmentação e remoção de ruídos. Em seguida, as imagens serão enviadas para uma RNC e na última etapa as características extraídas serão as informações de entrada de um novo classificador que irá determinar a classe do objeto resultante na saída da nova rede.

# Refatoração Baseada em Padrões de Projeto Usando Agentes

Thyago Henrique Pacher<sup>1</sup>, Simone Nasser Matos<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR)  
Av Monteiro Lobato, s/n - Km 04, CEP 84016-210 - Ponta Grossa - PR – Brasil

thyago.pacher@gmail.com, snasser@utfpr.edu.br

**Palavras-chave:** Agentes. Refatoração. Padrões de Projeto.

**Resumo.** O processo de refatoração garante uma qualidade maior no código-fonte aumentando a sua manutenibilidade, legibilidade e flexibilidade. Para Fowler(1999) a refatoração é um processo usado para eliminar *bad smells* e fazer com esta execução a possível melhoria de legibilidade e manutenibilidade de código para facilitar futuras modificações. A refatoração pode ser conduzida com técnicas em geral ou padrões de projeto (*Design Patterns*). Foi criado por Fowler (1999) um catálogo de técnicas de refatoração que é separado nos grupos: *Composing Methods*, *Moving Features between Objects*. Cada grupo possui mais de uma técnica envolvida, por exemplo, o grupo *Composing Methods* tem as técnicas *Extract Method*, *Inline Method*, *Replace Temp with Query*, entre outras. Já a refatoração baseada em padrões de projeto é fundamentada no catálogo de Gamma (1995) e existem na literatura métodos que ajudam a detectar e inserir padrões de projeto tais como de Cinnéide e Nixon (1999), Kerievsky (2009), Ouni et al. (2017), porém não são métodos autônomos. Com a intenção de identificar o que existe na literatura sobre a área de refatoração com agentes foi realizado um mapeamento sistemático focado no uso de agente para o processo de refatoração de software. Como resultado deste mapeamento identificou-se que métodos autônomos de refatoração abrangem somente as técnicas de Fowler (1999) e não contemplam a aplicação de padrões de projeto. Este trabalho propõe um agente para detectar pontos de inserção e aplicação de padrões de projeto em código-fonte orientado a objetos em linguagem de programação Java. Um agente de acordo com Coppin (2015) é um sistema projetado para realizar tarefas em nome do ser humano. Ele pode ser entendido como uma entidade computacional com característica de ser autônomo ou semi-autônomo e ter inteligência para tomada de decisões com base no meio ao qual se encontra. O agente tem as seguintes características: i) escanear o projeto; ii) interpretar o código; e iii) analisar o código requisitando as suas crenças para verificar se por meio de algum método da literatura é possível aplicar algum padrão usando uma lista que contém os passos para seu uso. O agente foi implementado usando *Jadex* e a *IDE Eclipse Oxygen 3a*. O agente ocorre em conjunto a um *plugin* onde este detecta qual projeto esta aberto para poder chamar o agente. O experimento com o agente de refatoração para padrões de projeto foi aplicado em 25 (vinte e cinco) projetos que estavam disponíveis no *GitHub* e a quantidade de classes destes variaram de 1 a 359. Os resultados mostram que o agente conseguiu detectar de 1 a 149 classes candidatas a inserção de padrões de projeto. Conclui-se que o uso de agentes na refatoração pode trazer contribuição para profissionais menos experiente que tem a possibilidade de aplicar padrões de projeto em seu projeto para deixá-lo mais flexível e reusável.

## Identificação de falhas em linhas de plantio utilizando imagens aéreas obtidas por VANT's

Simone Bello Kaminski Aires<sup>1</sup>, Enzo Salvadori<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR)  
Av Monteiro Lobato, s/n - Km 04, CEP 84016-210 - Ponta Grossa – PR – Brasil

sbkaminski@utfpr.edu.br,  
enzosalvadori17@gmail.com

**Palavras-chave:** Computação; Agricultura de precisão; Processamento de imagens; VANT; Binarização; Transformada de Hough; Esqueletização

**Resumo.** Agricultura de precisão é a área da agronomia que trabalha com recursos tecnológicos para gestão dos processos e de técnicas de georreferenciamento, com intuito de melhorar a produtividade geral da plantação e possibilita ao agricultor um monitoramento de sua produção de forma mais precisa e constante. Este trabalho apresenta um método para identificação de locais onde ocorreram falhas de plantio em uma plantação de milho. O processo consiste em utilizar técnicas de processamento de imagem como: binarização, transformada de Hough, esqueletização e outras, para encontrar as linhas de plantio do milho e através desse resultado encontrar os pontos falhos e quantificá los. A base de dados de imagens da plantação serão obtidas por um VANT (Veículo Aéreo Não Tripulado).

A aplicação tem uso prático no campo e está diretamente ligada a produtividade do agricultor, uma falha corresponde a uma área onde deveria existir uma planta mas devido a algum fator externo a mesma não se desenvolveu. Essa perda de plantas pode gerar um grande impacto econômico dependendo da quantidade de plantas perdidas. Com o conhecimento das falhas o agricultor pode contornar a perda caso ainda esteja em uma fase inicial da plantação ou pode buscar identificar o fator externo que afetou o plantio e assim monitorar tal condição para reduzir a perda de produtividade em próximas plantações. Esses fatores externos podem variar desde uma má regulagem da plantadeira, sementes de má qualidade, fatores climáticos entre outros.

# Calibração de câmeras térmicas usando pastilhas Peltier

Othon Alberto da Silva Brigano<sup>1</sup>, Erikson Freitas de Moraes<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática – Universidade Tecnológica Federal do Paraná  
Rua Doutor Washington Subtil Chueire, 330 - 84017-220 – Ponta Grossa – PR – Brasil

othon@alunos.utfpr.edu.br, emorais@utfpr.edu.br

**Abstract.** *Camera calibration is essential in computer vision applications that require real-world metric information such as distance and size. For conventional cameras, there are many methods known. However, thermal cameras capture temperature, not colors, making it impossible to directly use the same methods. In this context, the method of Saponaro is the most used, however for its use is required a period of initial heating and reheating after a certain time of use. We propose a new calibration board for thermal cameras which can be used for a long period of time and allows the use of a classic calibration method for conventional cameras.*

**Keywords:** Computer Vision; Camera Calibration; Thermal Camera.

**Palavras-chave:** Visão Computacional; Calibração de câmera; Câmera térmica.

**Resumo.** *Calibração de câmeras é essencial em aplicações de visão computacional que necessitam de informações métricas do mundo real, como distância e tamanho. Para câmeras convencionais, o processo é conhecido e há uma série de métodos, como o de Zhang. Entretanto, câmeras térmicas capturam temperatura, não cores, impossibilitando o uso direto dos mesmos métodos utilizados nas câmeras convencionais. Nesse contexto, o método de Saponaro é o mais conhecido, entretanto para sua utilização é necessário um período de aquecimento inicial e reaquecimento após um certo tempo de uso. Propomos um novo alvo para calibração de câmeras térmicas que pode ser utilizado por um longo período de tempo e que permite a utilização de um método já conhecido no contexto das câmeras convencionais.*

## 1. Introdução

A calibração de câmeras é um método que permite estimar os parâmetros que definem a projeção de um ponto tridimensional (3D) do mundo em um ponto bidimensional (2D) da imagem [Ellmauthaler et al. 2013]. É um importante processo em aplicações de visão computacional quando há necessidade de informações métricas precisas, mapeadas no mundo 3D, como distância entre objetos, tamanho e profundidade, em aplicações como sistemas para aumentar a segurança em veículos [Bertozzi and Broggi 1998] e reconstrução 3D [Geiger et al. 2011]. Para câmeras convencionais há uma série de métodos conhecidos e amplamente utilizados, como [Zhang 2000] e [Heikkila and Silven 1997]. O mais conhecido é o método de Zhang, detalhado na Seção 2.1.

Câmeras térmicas representam uma opção interessante em diversas áreas, como em uso militar, médico, vigilância e em sistemas inteligentes para carros autônomos, visto que sua efetividade não depende da luminosidade da cena e sim da energia térmica recebida, podendo, assim, ser utilizada em situações de escuridão, neblina, ofuscamento e objetos escondidos, como uma pessoa atrás de uma planta ou um animal em uma rodovia com pouca iluminação. Em circunstâncias como essas, objetos importantes no contexto da cena podem não ser observáveis por uma câmera convencional.

Para calibrar esse tipo de câmera as técnicas tradicionais não são aplicáveis, visto que a imagem é gerada através de sinais elétricos de captação de radiação infravermelha. O que é visto em uma imagem obtida por esse tipo de câmera é a conversão desses valores de temperatura em cores, seguindo uma determinada escala. Como as peças brancas e pretas do alvo tradicional estarão em temperaturas praticamente idênticas não será mantido o padrão do tabuleiro de xadrez, tornando-o invisível. Como possível solução para este problema, em [Saponaro et al. 2015] é desenvolvido um alvo para câmeras térmicas baseado em placas de cerâmica. O padrão de tabuleiro de xadrez é colado em uma peça de cerâmica e então aquecida por uma lâmpada de 250W. Após a obtenção da sequência de imagens, o método aplica um pré-processamento em cada uma delas, que consiste em destacar o alvo de calibração do resto da cena, subtrair o ruído presente na imagem, aplicar *Top Hat Filtering* (uma operação morfológica utilizada para remover iluminação não-uniforme) e, por fim, aplicar correção de gama. No pior cenário, o alvo é aquecido durante 5 minutos e sua utilização é possível por até 10 minutos.

Desta forma, observou-se a necessidade de construir um alvo com temperaturas constantes, utilizável por um período maior de tempo, além de *setup* prático e que tornasse possível aplicação dos métodos já conhecidos para câmeras convencionais em calibração de câmeras térmicas. Para isso, utilizou-se Pastilhas Peltier. Essas pastilhas são quadradas e se baseiam no Efeito Peltier, tendo como comportamento, quando ligadas, um lado esquentar e o outro esfriar. Assim, apresentam características que torna possível serem utilizadas para simular um tabuleiro de xadrez.

O objetivo deste trabalho é apresentar um novo alvo para calibração de câmeras térmicas usando Pastilhas Peltier para que o alvo térmico tenha características similares ao alvo tradicionalmente usado na calibração de câmeras convencionais. A Seção 2 fornece resumidamente o *background* necessário para calibração de câmeras, a Seção 3 detalha a construção do alvo proposto, a Seção 4 demonstra os resultados obtidos em experimentos realizados e a Seção 5 apresenta as conclusões e trabalhos futuros.

## 2. Calibração de câmeras

No contexto de aplicações de Visão Computacional todas as câmeras estão descalibradas. Isso acontece por conta das inevitáveis imperfeições no processo de construção das lentes das câmeras, no qual uma lente nunca é perfeitamente idêntica à outra. Dessa forma, são inseridos pequenos erros de captura nas imagens, o que faz com que o *frame* obtido, em relação às suas dimensões, não seja fidedigno ao objeto real em cena. Por conta dessa diferença, aplicações que exijam medidas precisas, como um veículo autônomo, necessitam da utilização de um método de calibração, pois centímetros de erro podem

causar um problema grave em seus sistemas.

Ao realizar o processo de calibração de uma câmera, o que se deseja conhecer são os parâmetros que mapeiam os pontos 3D da cena com seus pontos 2D correspondentes na imagem. Ou seja, que corrijam o erro causado pela câmera. Os parâmetros obtidos são classificados em dois grupos. O primeiro deles é o conjunto de parâmetros intrínsecos, que representam as características geométricas internas e ópticas da câmera, como distância focal ( $f$ ), fatores de escala ( $s_x$  e  $s_y$ ), ponto principal ( $u_0, v_0$ ) e imperfeições introduzidas pelo sensor da câmera ( $\pi$ ) [Azevedo et al. 2008]. O outro grupo é o de parâmetros extrínsecos, que, em conjunto, informam a posição e orientação da câmera na cena. Para isso utiliza-se uma matriz de rotação  $R$  e um vetor de translação  $t$ , em relação a um certo sistema de coordenadas 3D. É importante ressaltar que câmeras térmicas também podem ser calibradas em relação a valores de temperatura, mas esse tipo de calibração não é abordado neste trabalho.

Por conta do processo de construção da lente dar origem a lentes diferentes, sendo impossível fabricar exatamente igual, é inviável modelar a lente em cada processo de calibração. Por tanto, utiliza-se usualmente um modelo empírico, estimando a distorção radial e tangencial.

### 2.1. Método de Zhang

O método proposto por Zhang, em 2000, é amplamente utilizado como principal método para calibração de câmeras. Esse método requer apenas uma câmera apontada para um alvo de calibração previamente conhecido, como um tabuleiro de xadrez, em uma superfície coplanar. Ao definir que os pontos 3D do alvo de calibração sejam pontos coplanares, o eixo Z é eliminado, simplificando assim o mapeamento entre pontos 3D, do mundo real, e 2D, da imagem. Zhang utiliza-se de conceitos matemáticos como homografia, além de otimização não linear e o conceito de *maximum-likelihood criterion* (em tradução livre, critério de máxima verossimilhança) para resolver as aproximações e estimativas da calibração [Zhang 2000].

De maneira simplificada, o método de Zhang é composto pelas seguintes etapas:

- 1) Obtenção de imagens do alvo coplanar de calibração, em diferentes posições e ângulos;
- 2) Em cada imagem obtida, encontrar os vértices dos quadrados do padrão de calibração;
- 3) Através dos dados obtidos na etapa anterior e do conhecimento prévio do padrão, calcular os parâmetros de calibração [Azevedo et al. 2008].

Ao utilizar um alvo de calibração tradicional para realizar a calibração de uma câmera térmica, não será possível realizar o passo 2, pois a imagem obtida não terá os quadrados, pelos motivos já antes citados. Dessa forma, para ser possível utilizar o método de Zhang com a imagem térmica, deseja-se encontrar uma maneira do padrão de calibração ser visível termicamente.

## 3. Metodologia

Com o problema conhecido, é possível perceber a possibilidade de utilizar os mesmos algoritmos usados em calibração convencionais. Para ser possível encontrar as quinas em uma imagem térmica, é necessário que haja uma diferença de temperatura en-

tre peças equivalentes à peças "brancas" e "pretas" de um alvo convencional, de modo que o encaixe entre elas seja visível em uma imagem de temperaturas. Existem várias maneiras de se atingir o objetivo de tornar as quinas visíveis [Saponaro et al. 2015] [Ellmauthaler et al. 2013][Vidas et al. 2012].

Neste trabalho, para montar o tabuleiro seguindo as especificações acima foram utilizadas pastilhas Peltier. Comumente, são empregadas para resfriar microprocessadores e construir pequenas geladeiras e bebedouros, bem como em sistemas para aquecimento [Sanchez et al. 2015]. Essas pastilhas se baseiam no Efeito Peltier, descoberto pelo físico francês Jean Charles Athanase Peltier, no ano de 1834, e têm como característica absorver calor em um dos lados e o outro lado dissipar calor, quando aplicada corrente elétrica à pastilha. Além disso, possuem o formato quadrado, perfeito para representar uma peça de tabuleiro em um alvo de calibração.

### 3.1. Configuração do Alvo

No desenvolvimento do alvo de calibração que chamaremos de alvo térmico, foram utilizadas 10 pastilhas Peltier, posicionadas com o lado que absorve calor para cima. As pastilhas, cada uma delas medindo  $40\text{mm} \times 40\text{mm}$ , foram fixadas em uma superfície plana de madeira. A nível de circuito, as Pastilhas Peltier foram organizadas em 3 grupos: 2 deles com 3 pastilhas e 1 com 4 pastilhas. Esse circuito é alimentado por 5W, em paralelo, fornecidas por uma fonte ATX 550w Real.

Para simular o tabuleiro de xadrez são necessários dois conjuntos. No tabuleiro convencional, são representados por peças brancas e pretas. No alvo térmico, um dos conjuntos é composto pelas Pastilhas Peltier e o outro é representado por um espaço vazio, de mesmo tamanho, como demonstrado na Figura 1a.

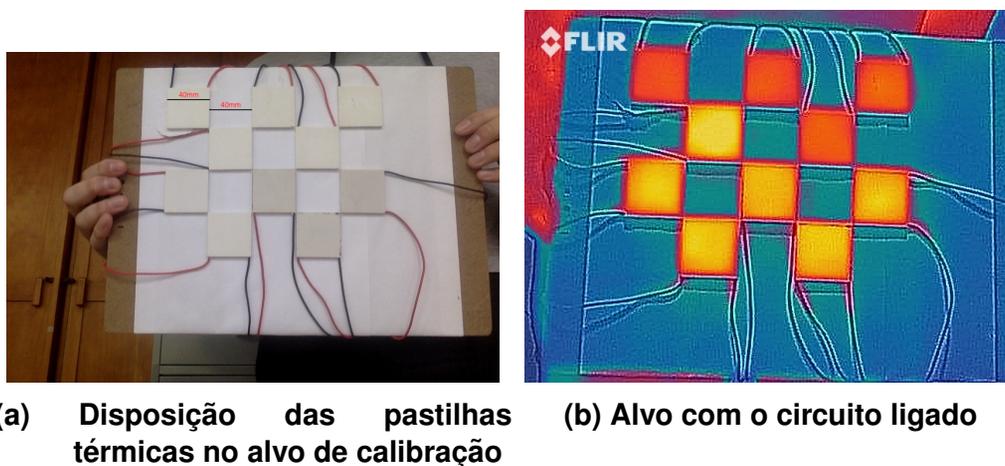


Figura 1. Configuração do alvo térmico de calibração.

Com as 10 pastilhas disponíveis o alvo foi montado em uma configuração  $5 \times 4$ . Sendo assim, existem 12 pontos internos para calibração do alvo. Na implementação do método de Zhang da biblioteca OpenCV são necessários, no mínimo, 2 pontos internos na horizontal e vertical. Portanto, o alvo térmico construído atende esse requisito.

## 4. Resultados

Para a realização dos testes, foram capturadas imagens do alvo de calibração térmico com uma câmera térmica FLIR One, de resolução  $320\text{px} \times 240\text{px}$ , em um ambiente controlado em laboratório. Após a aquisição das imagens para aplicação do método de Zhang, foi executado o mesmo algoritmo de calibração usado para câmeras convencionais, utilizando o OpenCV, principal biblioteca para aplicações de Visão Computacional e que implementa esse método por padrão. O resultado é demonstrado na Figura 2b.



(a) Imagem térmica

(b) Figura 2a com as quinas encontradas

Figura 2. Exemplo de rotina de calibração

A aquisição de pastilhas Peltier para a construção do alvo tem um custo considerável, principalmente em comparação com o alvo desenvolvido em [Saponaro et al. 2015]. Porém, com os bons resultados obtidos, foi constatado que a calibração de câmeras térmicas com essas pastilhas é uma opção viável e prática, considerando que há apenas a necessidade de posicioná-las como em um tabuleiro de xadrez e conectar à uma fonte de energia adequadamente. Além disso, o alvo apresenta pouca variação térmica. Após ligá-lo, as pastilha se aquecem em poucos segundos e então tendem à constância. Não há necessidade de reaquecimento durante o processo. Desse modo, o objetivo de construir um alvo constante e que torna possível utilização por um longo período de tempo foi alcançado.

## 5. Conclusão e Trabalhos Futuros

Por ser tratar de uma pesquisa em desenvolvimento, há melhorias possíveis no futuro. O alvo pode ser aumentado, pois o tamanho atual de  $5 \times 4$  é pequeno em relação aos alvos  $10 \times 7$  comumente utilizados na calibração convencional [OpenCV 2018]. Ao expandir o alvo, eleva-se a quantidade de pontos detectáveis, podendo aumentar a precisão da calibração. Essa expansão não foi possível pela baixa quantidade de placas disponíveis no momento deste estudo.

Outros dois problemas observados em imagens nas quais o alvo de calibração está próximo à câmera podem ser melhorados:

1. A organização da energia para as pastilhas. O grupo de quatro pastilhas fica com temperatura menor e, assim, um tom mais escuro em relação às outras pastilhas. Essa melhoria pode ser feita encontrando uma distribuição igual entre os grupos e uma corrente ideal para que as pastilhas se destaquem, mas não tão alta para não diminuir a vida útil das Pastilhas Peltier, ou via *software*, detectando os locais onde as pastilhas estão na imagem e padronizando todas com a mesma cor.
2. Os cabos da pastilhas precisam ser escondidos para não aparecerem na imagem, pois os cabos apresentam temperatura maior que o fundo do alvo, por conta da corrente que passa por eles. Desse modo, se destacam na imagem, como pode-se observar na Figura 1b, prejudicando o processo de calibração.

Conclui-se que o alvo térmico é uma opção prática para calibração de câmeras térmicas, após a construção inicial. Além disso, o alvo aqui proposto é capaz de ser utilizado por um grande período de tempo, o que permite que em experimentos que necessitem de um longo tempo para obter imagens de calibração não seja necessário parar o processo para configurar novamente o alvo.

## Referências

- Azevedo, T. C., Tavares, J. M. R., and Vaz, M. A. (2008). Análise do método de calibração de câmaras proposto por zhang. In *Proceedings CLME'2008-5º Congresso Luso-Moçambicano de Engenharia, IICEM-2º Congresso de Engenharia de Moçambique*.
- Bertozzi, M. and Broggi, A. (1998). Gold: a parallel real-time stereo vision system for generic obstacle and lane detection. *IEEE Transactions on Image Processing*, 7(1):62–81.
- Ellmauthaler, A., da Silva, E., Pagliari, C., Gois, J., and R. Neves, S. (2013). A novel iterative calibration approach for thermal infrared cameras.
- Geiger, A., Ziegler, J., and Stiller, C. (2011). Stereoscan: Dense 3d reconstruction in real-time. In *2011 IEEE Intelligent Vehicles Symposium (IV)*, pages 963–968.
- Heikkila, J. and Silven, O. (1997). A four-step camera calibration procedure with implicit image correction. In *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)*, CVPR '97, pages 1106–, Washington, DC, USA. IEEE Computer Society.
- OpenCV (2018). Camera calibration with opencv. [https://docs.opencv.org/3.3.1/d4/d94/tutorial\\_camera\\_calibration.html](https://docs.opencv.org/3.3.1/d4/d94/tutorial_camera_calibration.html).
- Sanchez, G., Santana, H. S., Tortola, D., and Taranto, O. P. (2015). Desenvolvimento de um sistema de aquecimento baseado em placa de peltier para microdispositivos. *Blucher Chemical Engineering Proceedings*, 1(3):2457 – 2462.
- Saponaro, P., Sorensen, S., Rhein, S., and Kambhamettu, C. (2015). Improving calibration of thermal stereo cameras using heated calibration board. *2015 IEEE International Conference on Image Processing (ICIP)*, pages 4718–4722.
- Vidas, S., Lakemond, R., Denman, S., Fookes, C., Sridharan, S., and Wark, T. (2012). A mask-based approach for the geometric calibration of thermal-infrared cameras. *IEEE Transactions on Instrumentation and Measurement*, 61(6):1625–1635.

Zhang, Z. (2000). A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:1330–1334.

## Security in multi agent systems

Bruno Rafael Alves<sup>1</sup>, André Pinz Borges<sup>1</sup>, Gleifer Vaz Alves<sup>1</sup>, Paulo Leitão<sup>2</sup>

<sup>1</sup> DAINF - Universidade Tecnológica Federal do Paraná (UTFPR),  
Ponta Grossa-PR, Brasil.

<sup>2</sup>CEDRI - Instituto Politécnico de Bragança (IPB), Bragança, Portugal.

brunoa@alunos.utfpr.edu.br, {apbirges, gleifer}@utfpr.edu.br

pleitao@ipb.pt

**Abstract.** *Smart Cities use technology to improve the citizens' life. A branch of it is the Smart Parkings, that intends to solve the problem of parking cars, making it easier to find a spot to park and consequently decrease the traffic of cars. In this context, a cyber-physical architecture, based on Multi Agent Systems, was developed for a Smart Parking System for cars and bicycles by a previous group. It consists of a community of distributed, intelligent and autonomous agents, representing the parking spots and drivers, which negotiate to reach their objectives. This paper takes the security and the support of failures of the system in consideration, improving the architecture with the implementation of SSL and Certificate Authorities.*

**Keywords:** Multi Agent System; Smart Parking; SSL; Certificate Authority;

### 1. Introduction

Large cities are crowded places with several problems emerging every day, for example, pollution, energy, waste and traffic. A Smart City [Neirottia et al. 2014] aims to make the citizens' daily life more comfortable and convenient by using emergent technologies in order to provide accessibility to public information and services [Anthopoulos and Fitsilis 2010].

When it comes to traffic, the parking problem can be crucial for the improvement of the smart city concept, since 30% of the traffic is generated by drivers trying to find a spot to park their cars [Silva 2015]. Consequently, a parking that uses advanced technologies to improve its management and the provided services can contribute for a reduction of the traffic [Koster et al. 2014].

On a previous work, an architecture for a smart parking has been proposed [Alves 2019]. This architecture aims to develop an agent-based solution to improve the management and the organization of a smart parking, focusing in the agility of the services and the profit. Furthermore, the driver is encouraged to cooperate with the system as a whole, for instance, the driver respects the agreement so he doesn't get to pay tickets.

A Smart Parking System is strongly based on the exchange of messages between agents, these communications become a point of interest in the security. This paper proposes a security architecture for the messages exchanged by the agents in the system

proposed before. Furthermore, this approach can be used in similar scenarios. For that, SSL and certificate authorities are used.

### 2. Smart Parking

Smart cities represents cities that use the technology to improve the well being of the citizens and the efficiency of the city services as a whole [Neirottia et al. 2014]. One branch of it is the Smart Parking. Smart Parking uses technology to solve the parking searching problem and consequently reduce the urban traffic congestion.

As an example of the using technology in a Smart Parking, can be given by means of a mobile application, where the parking spots maybe more accessible and visible to the drivers. Another way to improve the parking is to find the best price for the driver and for the parking manager, making the best agreement for both of them. One way to reach that goal is trough a negotiation, for that, agents can be used in a Multi Agent System (MAS), that is, a system with several agents reacting to each other at the same time. When a driver agent wants to find a place to park, the spot agents can make an offer.

### 3. Architecture

On a previous work, we have developed an agent-based architecture for Smart Parking. Figure 1 presents the main participants of the system. The Driver represents the person that drives the vehicle; the AgentDriver represents the Driver in the system (most likely a phone application); The AdmSpot represents the spot owner (most likely the parking administrator and, it can also have more than one spot); The AgentSpot is the agent that represents a specific spot in the system (most likely a micro controller).

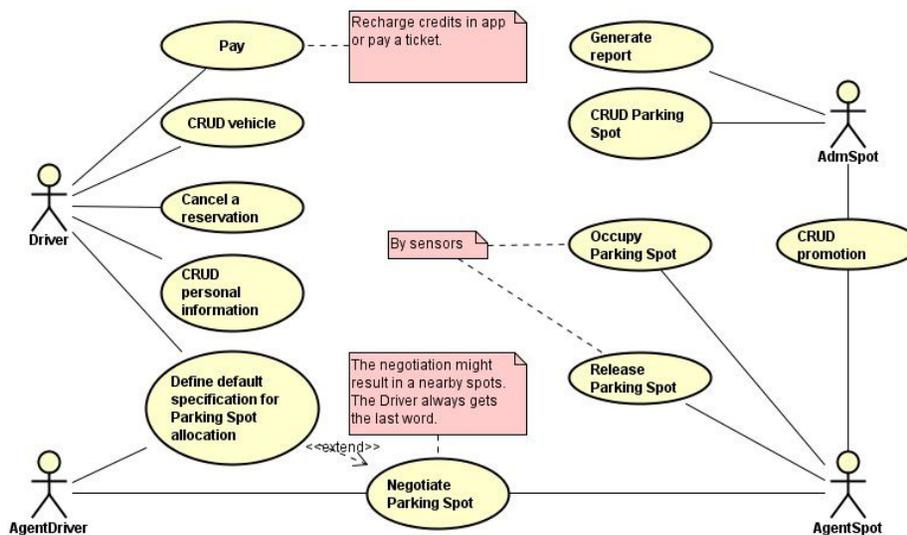


Figure 1. Use Case Diagram Base [Bruno Rafael Alves 2019]

In this architecture, each spot is a module that can be deployed to the system individually. That is, the system can be escalated. Furthermore, the system support failures. If an agent stops to work the system as a whole isn't affected by it. But, it's easy to put an agent with a virus in the system, for example.

The use case 'Negotiation Parking Spot' is a key aspect and will be approached in this paper, since it represents the core of the system. In this use case, the agents communicate with each other to get in agreement and decide the best offer. Moreover, this is the use case that has the higher amount of exchanged messages between agents, becoming one of the possible security breaches.

#### 4. Cryptography

In 1978 Rivest, Shamir, and Adleman discovered the first practical public-key encryption and signature scheme, now referred as RSA. Based on that, in 1991 the first international standard for digital signatures (ISO/IEC 9796) was created [Katz et al. 1996]. This protocol is based on the goals of the cryptography, explained in [Katz et al. 1996]. They are:

1. Confidentiality: A service that does not allow others but the ones previously authorized to see the content of the information. For example, physical protection to mathematical algorithms, which render data unintelligible, can provide confidentiality.
2. Data integrity: A service that prevent the data to be modified by an unauthorized person. To assure that, the detection of an unauthorized change in the data must be identifiable.
3. Authentication: A service related to identification. Both sides must be able to identify the other when a message is received. The information about the data should also be authenticated. Usually divided into two major classes: entity authentication and data origin authentication.
4. Non-repudiation: A service that ensures all the commitments or actions. When an entity tries to deny some previous action, a third trustworthy party is needed to verify an confirm the operations.

#### 5. Certificate Authorities and SSL

In order to guarantee the identity of the agents in the network the Certificate Authority (CA) certificates all the entities in the system. In other words, it's a database of certificates. The system can have more than one CA, each one having a copy of the data base. In that way, the system support failures.

The CAs contain information about the entities, like their version, serial number and public key. The certificate of that entity and all that information are encrypted. All the entities in the system but the CAs send messages using the public key of the destination entity, so only that entity can read the message. On the other hand, the CAs use their own private key to generate the messages, allowing any entity to read the certificates with the public key and making it almost impossible to fake.

For instance, if an entity A wants to communicate with an entity B: first of all, the entity A requests the certificate of the entity B for the entity B and to one CA. Entity A compares them to verify the agent B. After, entity B makes the same verification of the entity A. Once those are verified, they can exchange a symmetric key for a simpler and secure communication called Secure Sockets Layer (SSL) [Elgamal and Hickman 1997].

Now, only the entities that have the symmetric key can join the conversation. In this case, entity A and B.

### 6. Cryptosystem Proposed

As previously mentioned, our goal is to improve the Smart Parking architecture by adding a security layer. This layer comprises SSL and CAs in order to secure the messages in the system.

Initially, the first CA is generate. That CA now can validate the agents (entities) and other CAs. In that way, the system is protected, since, it doesn't rely only on one CA. It will make the system faster, since the response time will be smaller than if there was only one CA answering all the requests. Also, if the system has only one CA, it would probably be far from most of the agents increasing the response time.

The CAs can be the parking lot that already have some structure, for instance a parking with a data base of clients. And, above all, it must be reliable. To define it, the Certificate Manager (a person) is needed. All the agents deployed in the system will have at least one CA listed. Then, when a new agent is deployed it is certified by one CA. After, the CA can inform other CAs in the system.

Another improvement to the system is to have the CAs responsible of a region or a sector, not the entire system. When an agent requests a certificate of an outsider agent, the CA of the region asks others in other regions. Each region might have more than one CA to not compromise the integrity of a region. Also, it will not need a lot of space and processing in the CAs, since each one will handle a small part of the whole system. That architecture is shown in Figures 2 and 3.

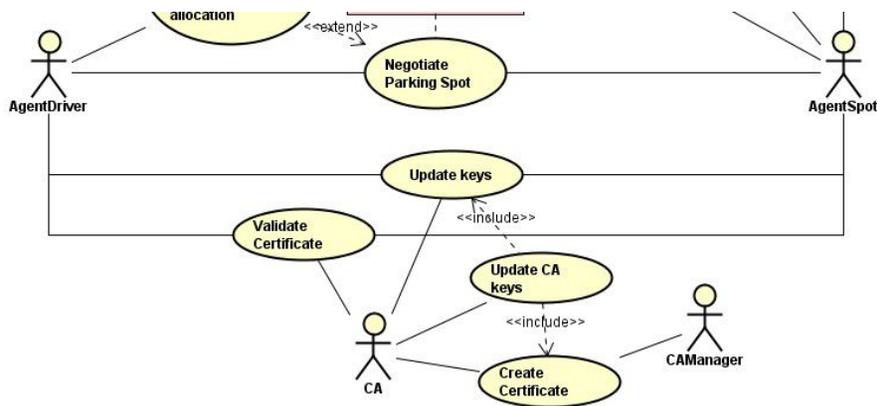


Figure 2. Use Case Diagram Proposal

The Figure 2 refers to the use case diagram of the proposed system. The difference are the additions of two actors and four use cases. The actors represents the person that manage the CA (CAManager) and its agent in the system (CA). With those use cases, all the CA operations can be established and a secure channel can be created (SSL). The use cases are:

- Create Certificate: When a new agent or CA is deployed in the system, one person (CAManager) needs to add it to one CA data base, generating the certificate of it.

- Update CA keys: When the CA update the keys or deploy some agent in the system, it needs to inform the other CAs to update their data base, informing the alterations.
- Validate Certificate: This use case is used when an agent needs to get a certificate about other entity in the system.
- Update keys: When a new agent is created or a key expires the CA create a new pair of keys.

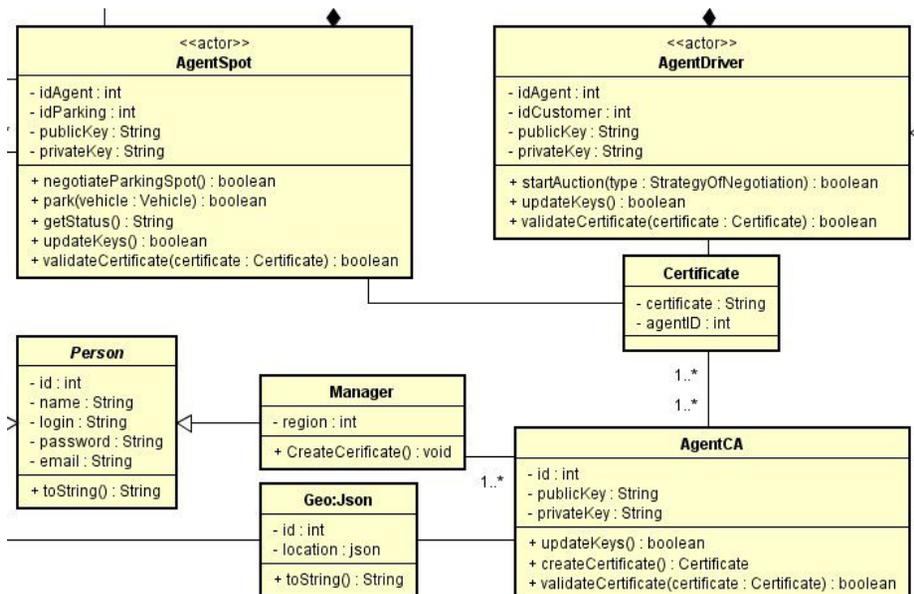


Figure 3. Class Diagram Proposal

Figure 3 shows a part of the class diagram relevant for this propose works. Three main classes are needed, the AgentCA, the Certificate and the Manager of the CA. The first represents the agent of the CA in the system that will store the certificates described in class Certificate. The last one is the previous mentioned CAManager. Those classes are explained bellow:

- AgentCA: This agent must have an unique id, publicKey and PrivateKey, just like any other entity in the system. Moreover, all entities must be capable of update the pair of keys (method "updateKeys"). This agent has two more important methods, the creation of a certificate and the validation of it.
- Certificate: May be the most crucial class in this architecture, it represents the certificates of every entity. Its content is encrypted with the CA private key, but everyone has its public key in order to read it but not fake it.
- Manager: The class that represents the CAManager in the system, capable of create a certificate for a agent that is being deployed in the system. Notice that is the Manager who has the region parameter. When a CA wants to communicate with others, it searches for other managers but its own. In that way, the communication is faster than check all the CAs in the system.

It's possible to the agent stores its public key in the CA aside with its own certificate, because, again, the CAs will not be needed to handle the whole system data, just a small part of it. At the end, the system will have several CAs grouped in regions or sectors holding the public keys and the certificates of the entities of the system. That can guarantee the security and integrity of the system as a whole.

Furthermore, the expiration of the keys is an important issue. From time to time, all the keys must be changed to guarantee the security of the system. That's because the algorithms are not unbreakable, but needs a lot of time to decrypt it. And if the agent itself change its keys from time to time, it makes the decryption of any message harder. The update of the keys can be made automatically by one CA and be distributed to the CAs in the region very fast.

## 7. Conclusion

As mentioned in the Introduction, the previous architecture on Smart Parking does not handle security issues in the exchange of messages between agents. Thus, here we propose an extension in our architecture in order to offer security services. Despite of the initial human need, once that are only agents in the system, it becomes autonomous again. Even when the keys expire, it does not take that much of the system to update itself, since just a few part of the CAs is needed to take care of that update.

Finally, this work proposed an architecture capable to improve the security of a MAS in the case of a Smart Parking. Nonetheless, our approach can be easily adopted in other scenarios, where a MAS is used.

As a future work, we intend to be focused in study other methods for cryptography that were not exposed here and may be better for this case of study, for instance the DSA [A.N.S.I. 1997]. Moreover, an architecture that uses more than one method could add some randomness to the system, improving the security. Furthermore, comparative performance tests about the system with and without this secure module need to be done. Finally, the implementation of Transport Layer Security (TLS) instead of the SSL should be studied, since the TLS is almost a new version of it.

## References

- Alves, B. R. (2019). Smart parking system: Architecture, protocols and prototype. Master's thesis, Instituto Politécnico de Bragança.
- A.N.S.I. (1997). Public key cryptography for the financial services industry: Part 1: the digital signature algorithm (dsa).
- Anthopoulos, L. and Fitsilis, P. (2010). Digital cities: Towards connected citizens and governance. In *Politics, Democracy and E-Government*, pages 275–291.
- Bruno Rafael Alves, Gleifer Vaz Alves, A. P. B. P. L. (2019).
- Elgamal, T. and Hickman, K. E. (1997). Secure socket layer application program apparatus and method. US Patent 5,657,390.
- Katz, J., Menezes, A. J., Van Oorschot, P. C., and Vanstone, S. A. (1996). *Handbook of applied cryptography*. CRC press.

- Koster, A., Koch, F., and Bazzan, A. L. C. (2014). Incentivising Crowdsourced Parking Solutions. In Nin, J. and Villatoro, D., editors, *Citizen in Sensor Networks*, volume 8313 of *Lecture Notes in Computer Science*, pages 36–43. Springer.
- Neirottia, P., Marcob, A. D., Caglianoc, A. C., Manganod, G., and Scorrano, F. (2014). Current trends in smart city initiatives: some stylised facts. *Cities*, 38:25–36.
- Silva, J. (2015). Aplicativo de vagas pode reduzir trânsito nas cidades e facilitar sua vida.

# Análise de Perfis de Doenças com Base em Técnicas de Descoberta de Conhecimento em Bases de Dados

**Kaique Augusto Morais da Silva<sup>1</sup>, Rodrigo Feuser<sup>2</sup>, Richardson Ribeiro<sup>2</sup>, Dalcimar Casanova<sup>2</sup>, Marcelo Teixeira<sup>2</sup>, André Pinz Borges<sup>1</sup>**

<sup>1</sup>Universidade Tecnológica Federal do Paraná (UTFPR), Câmpus Ponta Grossa, Departamento de Informática, Ponta Grossa, Brasil

<sup>2</sup>Universidade Tecnológica Federal do Paraná (UTFPR), Câmpus Pato Branco, Departamento de Informática, Pato Branco, Brasil

{kaiqmo, dalcimar}@gmail.com, rjfeuser@hotmail.com,  
{marceloteixeira, richardsonr, apborges}@utfpr.edu.br

**Abstract.** *In this work, disease profiles are analyzed based on techniques for discovering knowledge of databases of the electronic medical records of the Unified Health System (SUS) patients. The analysis focused on two disease groups: neoplasms and trauma. The process used the C4.5, Bagging and Boosting algorithms to create rules that help identify user profiles in health facilities. Compared to previous work, our approach is superior in terms of interpretation of data by health professionals.*

**Resumo.** *Neste trabalho são analisados perfis de doenças com base em técnicas de descoberta do conhecimento de bases de dados do prontuário eletrônico dos pacientes do Sistema Único de Saúde (SUS). A análise concentrou-se em dois grupos de doenças: neoplasias e traumatismos. O processo utilizou os algoritmos C4.5, Bagging e Boosting para criar regras que auxiliem na identificação de perfis de usuários em unidades de saúde. Em comparação a trabalhos anteriores, nossa abordagem é superior em termos de interpretação dos dados por profissionais da saúde.*

**Keywords:** Knowledge Discovery; Electronic Health Record.

**Palavras-chave:** Descoberta de Conhecimento; Prontuário Eletrônico do Paciente.

## 1. Introdução

Atividades clínicas, como consultas, exames laboratoriais, prescrições médicas diagnósticos, vacinações, entre outras, são realizadas constantemente pelos profissionais da área da saúde. Conseqüentemente, tais atividades geram uma quantidade significativa de dados, documentados geralmente quando ocorrem consultas médicas ou exames laboratoriais. Na área da saúde, esses dados possuem um modo específico para armazenamento, denominado Prontuário Eletrônico do Paciente (PEP) [Witten 2011].

O PEP normalmente armazena dados como nome, idade, profissão, problemas de saúde do paciente, entre outros. O uso destes tem ajudado cada vez mais as entidades de saúde a gerenciar dados, como informações dos pacientes, profissionais, médicos, etc. [Krysztof 2002]. Apesar dos investimentos em sistemas de informação para melhorar a gestão dos PEPs, esforços são necessários para a geração de conhecimento com os dados armazenados.

Um dos meios para explorar dados de PEP é por meio da Descoberta de Conhecimento em Banco de Dados (em inglês, *Knowledge Discovery in Databases* - KDD). Por KDD entende-se o processo, normalmente não trivial, de obter informações implícitas, previamente desconhecidas e potencialmente úteis, a partir dos dados armazenados em um banco de dados [Fayyad, Piatetsky-Shapiro and Smyth 1996]. KDD permite obter novas informações após uma série de etapas, como: seleção dos dados, pré-processamento e limpeza, transformação, mineração de dados e interpretação/avaliação dos resultados obtidos.

Neste trabalho foram aplicadas as etapas do KDD em uma base de dados de PEPs, com dados de pacientes da cidade de Pato Branco, Paraná. Inicialmente, foram levantadas as doenças contidas na base e, após um estudo, foi decidido trabalhar com dois grupos de doenças: neoplasias e traumatismos. Tais doenças foram escolhidas por apresentarem diferentes sub-grupos de doenças e por serem aquelas com maior registros na base. Na etapa de mineração de dados foi aplicado os algoritmos C4.5 [Quinlan 1993], *Bagging* [Breiman 1996] e *Boosting* [Freund and Schapire 1996], com objetivo de criar regras que auxiliem na identificação de perfis de usuários que contenham estas doenças.

## 2. Referencial

A Mineração de Dados é uma das etapas do KDD que, em seu processo, utiliza conceitos de base de dados, métodos estatísticos, ferramentas para visualização e técnicas de Inteligência Artificial (IA). O processo divide-se nas etapas de Seleção, Pré-processamento, Transformação, Mineração de Dados (MD) e Interpretação de dados [Fayyad, *et al.* 1996]. Na etapa de seleção é realizada a escolha da base e dos dados de interesse para o KDD. Na sequência é possível iniciar o pré-processamento, onde dados ruidosos e incompletos são removidos. Na etapa de formatação é feita a transformação dos dados para aplicação dos modelos de MD. E com a MD, algoritmos são aplicados de modo a classificar os dados gerando padrões, os quais podem ser avaliados para a descoberta de conhecimento sobre os dados de interesse. Durante o processo de interpretação é possível voltar as etapas anteriores para que seja melhorado a saída e obtenha-se melhores resultados. Ao completar o processo obtemos conhecimento sobre o banco de dados que foi usado para este processo.

Como dito anteriormente, o processo de KDD é aplicado neste trabalho em PEPs. Os PEPs possuem como vantagens: a agilidade no preenchimento dos documentos; segurança dos dados; atualização em tempo real e a portabilidade. Na literatura, é possível encontrar diversas pesquisas a partir de dados oriundos de PEP. Vilarinho R. (2017) utilizou algoritmos de mineração para obtenção de informações úteis relativas a casos de Dengue nos municípios brasileiros. Trindade (2012) aplicou o KDD para a identificação de padrões de comportamento das Hepatites Virais nas bases de dados do

SINAN (Sistema de Informação de Agravos e Notificações) do Sistema Único de Saúde - Governo Federal do Brasil, objetivando subsidiar ações de controle e prevenção da doença. Martins e Lima (2014) apresentam um estudo com as vantagens e desvantagens de usar um PEP, onde, apesar do custo de implantação, os PEPs apresentam grandes vantagens na sua implantação para gestão hospitalar. Destaca-se o trabalho de Feuser R. (2017), que aplicou os processos do KDD para PEP oriundo do SUS, utilizando algoritmo de associação *a priori* na etapa da mineração de dados, encontrando regras com elevado fator de confiança.

Há ainda na literatura diversos outros trabalhos que fizeram uso de dados oriundos de PEP para: análise de custo-efetividade de vacinas [De Soarez 2009], problemas de fratura ortopédica [Zorman *et al.* 2000], tomadas de decisões clínicas [Bae J-M 2014], tratamento de feridas crônicas [Letourneau and Jensen 2008], etc.

Contudo, os trabalhos já desenvolvidos não abordaram as doenças aqui analisadas: neoplasias e traumatismos. Além disso, os trabalhos não utilizaram de técnicas capazes de gerar conhecimento facilmente interpretáveis por profissionais da saúde, como as árvores geradas pelos algoritmos C4.5, *Bagging* e *Boosting*.

### 3. Desenvolvimento

Os dados utilizados nesta pesquisa foram obtidos por meio de um projeto com a participação das secretarias de Saúde, Ciência e Tecnologia do município de Pato Branco, bem como a participação da Universidade Tecnológica Federal do Paraná (UTFPR) Campus Pato Branco. Os dados foram fornecidos pela empresa responsável por desenvolver o sistema de PEP utilizado no município de Pato Branco. O conjunto de dados contém 43.879 pacientes e 2.296.626 registros de atendimentos. Dados pessoais, como nome, RG, CPF, telefone, e outros, não fazem parte do escopo desse projeto. Portanto, em momento algum, os pesquisadores desse projeto souberam a identificação de pacientes ou de seus responsáveis.

Como etapa de pré-processamento, foram executados os seguintes procedimentos: (i) identificação da faixa etária dos pacientes; (ii) utilização de dados do censo (e-SUS) tais como: bairro, altura, peso, frequência escolar, se frequenta benzedeira, se possui plano de saúde, se é fumante, se possui diabetes, se é gestante, se possui asma, se é alcoólatra, se já teve infarto, AVC ou derrame; (iii) remoção de registros com dados ausentes ou com ruídos foram eliminados, como por exemplo, pacientes que não possuem identificação de doenças ou que possuem apenas informações de exames; (iv) remoção de dados discrepantes relacionados à altura e peso dos pacientes, utilizando para isto valores mínimos e máximos.

A base de dados foi enriquecida com dados do grupo de Classificação Internacional de Doenças e Problemas Relacionados à Saúde (CID) das doenças [OMS, 2019]. Todos estes dados derivados foram também utilizados na geração das árvores de decisão na etapa de MD, cujo objetivo (classe alvo) era o grupo CID.

Uma análise dos grupos de CIDs foi realizada com objetivo de tratar apenas os CIDs referentes às doenças estudadas neste trabalho: neoplasias e traumatismos. As duas doenças foram escolhidas por apresentarem, no conjunto de dados, um maior equilíbrio no número de instâncias entre os diferentes CIDs. Esta filtragem ocorreu devido à falta

de recursos computacionais para processar todo o conjunto de dados. O grupo de traumatismos apresentou um total de 236 instâncias, e o grupo Neoplasias um total de 119 instâncias para a MD (conforme Tabelas 1 e 2).

**Tabela 1: Quantidade de ocorrências de Traumatismos.**

Tipo de Traumatismo	Qtde
Outras causas externas de traumatismos acidentais	16
Sequelas de traumatismos, intoxicações e de outras consequências das causas externas	5
Cabeça	18
Abdome, dorso, coluna lombar e da pelve	4
De localização não especificada do tronco, membro ou outra região do corpo	13
Cotovelo e do antebraço	11
Joelho e da perna	51
Ombro e do braço	7
Pescoço	3
Punho e da mão	49
Quadril e da coxa	7
Tórax	33
Tornozelo e do pé	40
<b>Total Geral</b>	<b>257</b>

**Tabela 2: Quantidade de ocorrências de Neoplasias.**

Tipo de Neoplasia	Qtde
Melanoma e outras(os) neoplasias malignas da pele	2
Neoplasias (tumores) benignas(os)	65
Neoplasias (tumores) de comportamento incerto ou desconhecido	3
Neoplasias (tumores) in situ	20
Neoplasias (tumores) malignas(os)	3
Neoplasias (tumores) malignas(os), declaradas ou presumidas como primárias, dos tecidos linfático, hematopoético e tecidos correlatos	7
Neoplasias malignas da mama	2
Neoplasias malignas do aparelho respiratório e dos órgãos intratorácicos	7
Neoplasias malignas dos órgãos digestivos	4
Neoplasias malignas dos órgãos genitais masculinos	6
<b>Total Geral</b>	<b>119</b>

#### 4. Resultados

Para a etapa de mineração de dados foi utilizada a ferramenta Weka<sup>1</sup> 3.8, onde foram executados os algoritmos C4.5 (algoritmo J48), *Bagging* e *Boosting* (algoritmo *AdaBoostM1*), todos utilizando as configurações padrão do Weka.

Para a execução no WEKA também é escolhido o método de testes de validação cruzada com 10 *folds* para a execução dos três algoritmos. Os resultados obtidos visam analisar os percentuais de acerto dos algoritmos de classificação. A Tabela 3 apresenta os percentuais de acerto, verdadeiros e falsos positivos obtidos dos classificadores C4.5, *Bagging* e *Boosting* referentes aos dois grupos de doenças estudados.

É possível observar que o grupo de neoplasias apresentou uma melhor precisão em todos os casos, sendo o algoritmo de *Boosting* aquele que melhor classificou o conjunto de dados, alcançando 87% de acerto. Ainda em neoplasias, o meta-classificador *Bagging* não foi capaz de melhorar a taxa de acerto do C4.5, algo que aconteceu no grupo de doenças relacionadas aos traumatismos. É interessante observar que nenhum falso positivo foi obtido para Traumatismos, apesar da menor precisão dos classificadores.

<sup>1</sup> Weka: <https://www.cs.waikato.ac.nz/ml/weka/>

**Tabela 3 – Dados comparativos dos algoritmos utilizados.**

	Neoplasias			Traumatismos		
	J48	Bagging	Boosting	J48	Bagging	Boosting
Verdadeiros positivos	84%	85%	87%	8%	7%	7%
Falso positivos	9%	10%	8%	0%	0%	0%
Precisão	70%	70%	87%	60%	61%	65%

No que diz respeito ao conteúdo das árvores obtidas, pôde-se observar, para ambos os conjuntos de doenças, que o principal fator na identificação das doenças foram os atributos bairro, indicação de frequência escolar e faixa etária. Todos os demais atributos utilizados apareceram em regras específicas e em menor quantidade nas árvores geradas.

## 5. Conclusões

A aplicação das etapas do KDD é uma alternativa para a geração do conhecimento em base de dados da área da saúde. É possível perceber o aumento de aplicações que fazem uso do KDD na área da saúde, evidenciando sua necessidade de utilização devido ao aumento exponencialmente de atendimentos clínicos, exames, etc. Uma das dificuldades é o acesso a pesquisa destes bancos de dados. Ainda não se dispõe de legislações sobre a disponibilidade de prontuários eletrônicos padronizados, ou centralizados, o que facilitaria o acesso ao histórico de tratamento do paciente por outras unidades de saúde e médicos.

A relação de atendimentos de unidades de pronto atendimento, são muito peculiares devido ao fato de ser em geral procurado pelos usuários para tratamento de dores agudas, acidentes ou males repentinos. Neste caso as correlações de padrões encontradas pelos algoritmos podem ser afetadas por vários fatores, mas podem inevitavelmente demonstrar fatores ainda não detectados em outras formas de visualizações.

Ao finalizar a aplicação das etapas do KDD para análise de perfis de doenças e dividir a amostra em dois grupos de doenças, um relacionado a neoplasias e o outro relacionado a traumatismos, foi possível determinar os principais atributos a serem analisados para identificação das doenças.

Para trabalhos futuros é possível a validação dos resultados obtidos por profissionais da saúde para definir a veracidade das regras encontradas. Isso poderia expandir este trabalho para outros locais de atendimentos de emergência, como: hospitais, ambulatórios e ou clínicas médicas. Outros trabalhos possíveis envolvem a integração com dados do IBGE, comparação com dados de cidades com mesmas características, aplicações de novos parâmetros e algoritmos.

## Agradecimentos

Esta pesquisa é apoiada por Decit/SCTIE/MS, por intermédio do CNPq, com apoio da Fundação Araucária e da SESA-PR. Programa Pesquisa para o Sistema Único de Saúde: Gestão Compartilhada em Saúde - PPSUS.

## Referências

- Alves, L. (2018) “Prontuário Eletrônico x Prontuário no papel”. Acessado em: <http://meuprontuario.net/prontuario-eletronico-x-prontuario-papel-qual-e-o-melhor/> Disponível em: 24/06/2018.
- Bae, J-M. (2014) “The clinical decision analysis using decision tree”. In: *Epidemiology and Health*. Vol. 36, pg 1-7.
- Breiman, L. (1996) "Bagging predictors". *Machine Learning*. 24 (2): 123–140.
- Carvalho, R. (2018) “Prontuário e registro de enfermagem”. Acessado em: <http://www.ebah.com.br/content/ABAAAASqAAG/prontuario-registro-enfermagem#>. Disponível em: 24/06/2018.
- De Soarez, P. C. (2009) “Use of decision analysis in the programs of vaccination against varicella”. São Paulo: “Faculdade de Medicina da Universidade de São Paulo”.
- Fayyad, U.; Piatetsky-Shapiro, G.; Smyth, P. (1996) “From Data Mining to Knowledge Discovery in Databases”. *AI Magazine*, AAAI, Boston.
- Feuser, R. (2017) “Mineração de Dados com Regras de Associação Aplicada em Dados de Unidade de Saúde de Pronto Atendimento”. UTFPR, Pato Branco (PR).
- Freund Y. and Schapire R.E. (1996) “Experiments with a new boosting algorithm”. In *Proceedings of the Thirteenth International Conference on International Conference on Machine Learning (ICML'96)*, Lorenza Saitta (Ed.). Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 148-156.
- Krysztof, J.C. (2002) “Uniqueness of Medical Data Mining. Artificial Intelligence in Medicine”, mar.
- Letourneau, S., Jensen, L. (2008) "Impact of a decision tree on chronic wound care". *Journal Wound Ostomy Continence Nurs*, vol. 25, pp. 240-247.
- Martins, C. and Lima, S.M. (2014) “Vantagens e desvantagens do prontuário eletrônico para instituição de saúde”. *Revista de Administração em Saúde (RAS)*, v. 16, n. 63.
- Organização Mundial da Saúde (OMS). (2019) “Centro Colaborador da OMS para a Classificação de Doenças em Português (CBCD)”. *Classificação estatística Internacional de Doenças e Problemas Relacionados a Saúde - CID - 10. 2008*. Disponível em <http://www.datasus.gov.br/cid10/V2008/cid10.htm>. Acesso em 02/02/2019.
- Quinlan, J.R. (1993) “C4.5: Programs for Machine Learning”. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- Thomaz, M. (2018) “Tudo que você precisa saber sobre prontuários eletrônicos”. Disponível em: <https://blog.iclinic.com.br/tudo-sobre-prontuario-eletronico/>. Acessado em: 24/06/2018.
- Trindade, C M et al. (2012) “Technology in health: knowledge discovery in public health databases: study of viral hepatitis in the state of Paraná”, Brazil. *Iberoamerican Journal of Applied Computing*, Ponta Grossa, v. 2, n. 2.

Vilarinho R.A. (2017) “Uso de Técnicas de Mineração de Dados para Classificação das Ocorrências de Casos de Dengue nos Municípios Brasileiros”. UFOP – Universidade Federal de Ouro Preto. Março.

Zorman, M., Podgorelec, V., Kokol, P., Peterson and M., Lane, J. (2000) “Decision tree's induction strategies evaluated on a hard real world problem”. In: Proceedings of the 13th IEEE Symposium on Computer-Based Medical Systems CBMS'2000, pp. 19-24.

Witten, I. H. (2011) “Data Mining Practical Machine Learning Tools and Techniques”. 3 ed.

# Mecanismo de Negociação Descentralizado entre Agentes em um Estacionamento Inteligente

Felipe Felix Ducheiko<sup>1</sup>, Gleifer Vaz Alves<sup>1</sup>, André Pinz Borges<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática

Universidade Tecnológica Federal do Paraná (UTFPR)

Rua Doutor Washington Subtil Chueire, 330 - Jardim Carvalho - Ponta Grossa - PR

felipeducheiko@alunos.utfpr.edu.br, {gleifer, apborges}@utfpr.edu.br

**Abstract.** *In a Smart Parking, agents could negotiate in order to obtain a parking spot. Usually, this sort of negotiation is centralized, using a manager agent. Here, we present a decentralized approach, where a given agent (seller) can offer a parking spot for  $n$  agents (buyers). A negotiation mechanism is presented together with some results which show how the Multi-Agent system works considering the number of exchanged messages (between agents), the time to park (when a negotiation is done), and the selling price (of parking spaces).*

**Resumo.** *Em um Smart Parking agentes podem negociar para obter uma vaga de estacionamento. Normalmente, esse tipo de negociação é centralizada, usando um agente de centralizador. No presente trabalho apresentamos uma abordagem descentralizada, em que um determinado agente (seller) pode oferecer uma vaga de estacionamento para  $n$  agentes (buyer). Com isso, é descrito um mecanismo de negociação em conjunto com alguns resultados experimentais, os quais evidenciam o funcionamento do Sistema Multi-Agente, em termos de quantidade de mensagens (entre os agentes), do tempo para estacionar (quando uma negociação é bem sucedida) e o valor de venda das vagas.*

**Keywords:** Smart Parking, Multi-Agent Systems, Negotiation Protocol.

**Palavras-chave:** Estacionamento Inteligente, Sistemas Multi-Agente, Protocolo de Negociação.

## 1. Introdução

O conceito *Smart City* surgiu durante a última década com a fusão de várias ideias, onde a essência do conceito é integrar as tecnologias que até agora têm sido desenvolvidas separadamente, mas que tem ligações claras em seu funcionamento e podem ser desenvolvidas de forma integrada [Batty et al. 2012]. Dentre os desafios a serem enfrentados pelas cidades destacam-se os de mobilidade urbana. Em Nova York 40% dos congestionamentos decorrem de motoristas buscando vagas de estacionamento [Koster et al. 2014].

Quando se percebe que a demanda de vagas de estacionamentos não está sendo satisfeita a solução adotada normalmente é um aumento quantitativo do número de vagas. Porém, a utilização das mesmas vagas de modo mais inteligente pode amenizar ou

até solucionar o problema. *Smart Parkings* são sistemas compostos por dispositivos de *hardware*, capazes de detectar o nível de ocupação do estacionamento e *softwares* integrados, para gerir a atribuição desses espaços de estacionamento [Nocera et al. 2014]. Tais sistemas são concebidos para auxiliar os motoristas na localização de vagas disponíveis, colaborando com a solução de problemas relacionados à mobilidade urbana. Dentre os vários modelos computacionais que podem ser utilizados para implementar um *Smart Parking* destacam-se os Sistemas Multiagentes (SMA).

Segundo [Wooldridge 2009] SMA são sistemas compostos de elementos computacionais que realizam interações ente si, de modo a atingirem seus objetivos, sendo tais elementos conhecidos como agentes. Esses agentes possuem duas características: são capazes de ações autônomas e têm a capacidade de interagir uns com os outros pela interação análoga às interações sociais humanas [Wooldridge 2009].

Por meio das habilidades sociais agentes devem ser capazes de negociar uns com os outros, afim de solucionarem problemas de forma distribuída, como ocorre em sociedades. Negociação é um processo complexo de tomada de decisão em que cada parte representa de forma autônoma seus pontos de vista e interage com as outras para resolver conflitos e chegar a acordos, maximizando os ganhos de todas as partes [Choi et al. 2001].

Para desenvolver um mecanismo de negociação é possível utilizar três configurações: i) *one-to-one*: onde um agente negocia somente com um agente, neste caso os agentes possuem preferências simétricas; ii) *many-to-one* (centralizado): nesta configuração, um único agente negocia com vários agentes, como acontece em um leilão; e iii) *many-to-many* (descentralizado): neste caso, muitos agentes negociam com muitos agentes simultaneamente, como acontece em um mercado [Wooldridge 2009].

Com o objetivo de aplicar métodos e técnicas usados em Sistema Multiagente na criação de soluções para alocação de vagas e gerenciamento de um *Smart Parking* foi concebido um projeto, que possui um SMA com mecanismo de negociação centralizado [Castro et al. 2017]. No sistema todos os agentes *drivers* negociam exclusivamente com o agente administrador. Esta abordagem possui vantagens, como o fato dos agentes *drivers* trocarem mensagem apenas com o agente administrador, diminuindo o custo computacional. Mas também possui desvantagens, visto que o agente administrador pode falhar, deste modo comprometendo todo o SMA. Outro ponto negativo é a questão da autonomia dos agentes, visto que os agentes *drivers* em um sistema com mecanismo de negociação centralizado ficam sujeitos as imposições do agente administrador.

Para o desenvolvimento de agentes autônomos com capacidades sofisticadas e flexíveis de negociação três itens devem ser definidos: i) protocolo de negociação, o qual define as ações que os agentes podem tomar em uma negociação; ii) o problema que a negociação quer solucionar; e iii) modelo de raciocínio, o qual define as ofertas iniciais, a gama de ofertas aceitáveis, as contraofertas, quando a negociação deve ser abandonada e quando um acordo deve ser fechado [O'Hare and Jennings 1996].

O objetivo do presente trabalho é estender o projeto, analisando o SMA para alocação de vagas de estacionamento com um mecanismo de negociação descentralizado, onde não existe a figura de um agente centralizador que administra o sistema e os agentes

*drivers* negociam as vagas entre si. Este artigo apresenta: (i) protocolo de negociação e modelo de raciocínio descentralizados; (ii) brevemente a implementação do SMA descentralizado; e (iii) uma parte dos experimentos realizados para validar o sistema.

## 2. Modelo de Raciocínio e Protocolo de Negociação

O modelo de raciocínio assume a existência de apenas um tipo de agente no SMA, o agente *driver*, o qual pode assumir dois papéis distintos: *seller*, quando o agente está deixando uma vaga e *buyer*, quando o agente está procurando uma vaga. Este modelo de raciocínio é apresentado em maiores detalhes em [Ducheiko et al. 2018].

O modelo de raciocínio obedece a um protocolo de negociação, que é o conjunto de regras as quais especificam a gama de movimentos legais disponíveis para cada agente em qualquer fase de um processo de negociação [Endriss et al. 2006]. A Figura 1 apresenta o protocolo de negociação que é utilizado para implementar o SMA.

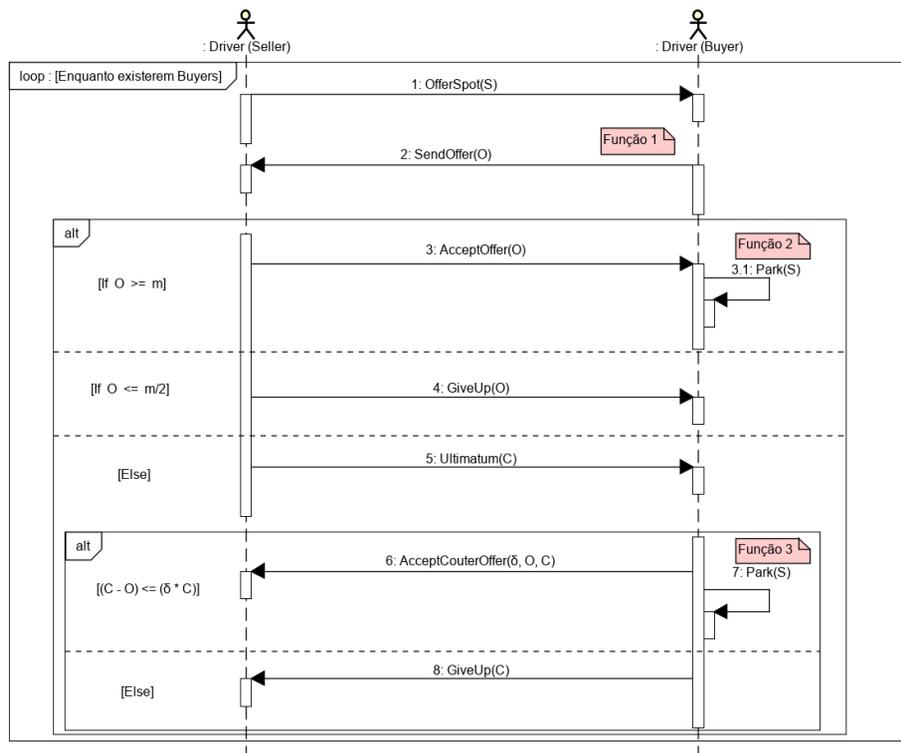


Figura 1. Protocolo de Negociação

Este protocolo define que sempre a negociação deverá ser iniciada pelo agente *driver seller*, informando os outros *drivers* do sistema que possui uma vaga para negociar (na Figura 1, item `1: OfferSpot(S)`). Então, os agentes *drivers* que estão procurando uma vaga (*buyer*) geram uma proposta, utilizando a Função 1, e a enviam para o *driver seller* (na Figura 1, item `2: SendOffer(O)`). Então, o *driver seller* analisa estas propostas utilizando a Função 2, podendo tomar as seguintes decisões: (i) aceitar proposta (na Figura 1, item `3: AcceptOffer(O)`), onde a negociação acaba e o acordo é fechado; (ii) rejeitar proposta (na Figura 1, item `4: GiveUp(O)`), onde a negociação entre estes dois agentes acaba; ou

(iii) enviar contraproposta (na Figura 1, item 5: *Ultatum(C)*), onde o agente *seller* gera uma contraproposta utilizando ainda a Função 2 e envia para o agente *buyer*.

Quando o agente *buyer* recebe esta contraposta ele a analisa utilizando a Função 3 e pode tomar duas ações: (i) aceitar a contraproposta (na Figura 1, item 6: *AcceptCounterOffer( $\delta$ , O, C)*), onde um acordo entre os dois *drivers* é fechado e a negociação acaba; ou (ii) rejeitar contraproposta (na Figura 1, item 8: *GiveUp(C)*), onde a negociação entre estes dois agentes acaba e o agente *seller* inicia uma negociação com outro *driver buyer*, isto se repete até que não haja mais agentes para negociar.

### 3. Implementação

Para implementar o sistema foram utilizadas as três camadas do *framework* JaCaMo. Na camada do *Jason* foram definidos dois tipos de agentes, os agentes *drivers* que utilizam o sistema e negociam vagas entre si e os agentes *parkingSpotController*, que possuem um artefato *Spot* relacionado e controlam o mesmo. Na camada do *Cartago*, foram definidos dois artefatos: o artefato *ParkingSpot*, que é a vaga a qual o agente *parkingSpotController* controla e o artefato *ProposalGenerator*, que é um artefato gerador de proposta, o qual os agentes *drivers* utilizam para calcular as propostas da negociação. Na camada do *Moise*, foi feita a especificação organizacional do sistema, onde foram definidos os dois papéis que o agente *driver* pode assumir, *seller* e *buyer*.

A Figura 2 mostra dois trechos do código da especificação da organização do SMA, utilizando o *Moise*. No primeiro trecho existe a definição dos papéis (definido em *role-definitions*), onde é possível observarmos a especificação dos dois papéis já mencionados: *seller* e *buyer*. No segundo trecho é possível visualizar as normas do sistema.

```

16<role-definitions>
17  <role id="driver" />
18<role id="buyer">
19  <extends role="driver"/>
20 </role>
21<role id="seller">
22 <extends role="driver"/>
23 </role>
24 </role-definitions>
96<normative-specification>
97  <norm id="norm1" type="obligation" role="buyer" mission="arrival"/>
98
99  <norm id="norm2" type="permission" role="driver" mission="negotiate"/>
100
101  <norm id="norm3" type="permission" role="seller" mission="departure"/>
102 </normative-specification>

```

Figura 2. Trechos código *Moise*

A primeira norma é a *norm1* que define que o agente *buyer* execute a missão *arrival*, que consiste em o agente chegar no estacionamento e verificar se há vagas livres que obedeçam os requisitos solicitados. A segunda norma é a *norm2* que define que os agentes iniciem a negociação das vagas. A terceira norma é a *norm3* que define que o agente *buyer* execute a missão *departure* e passe a ser um *seller*. Portanto, os agentes *buyer* que entram no sistema, procurando uma vaga e após conseguirem esta vaga trocam o seu papel e passam a ser *seller*, permanecendo na vaga pelo tempo desejado e depois tentando vender a vaga para outro motorista.

É importante ressaltar que no início da execução do SMA não existem agentes *sellers* e as vagas estão livres, isto é, não pertencem a nenhum agente. Nesta etapa a alocação de vagas é feita sem utilização de negociação. O agente *buyer* envia uma mensagem para os agentes *parkingSpotController* verificando se a vaga que ele é responsável está livre e o agente *buyer* estaciona na primeira vaga que receber a resposta que está livre que obedeça os requisitos solicitados, esta vaga passa a ser sua e o motorista assume o papel de *seller*.

#### 4. Resultados

Nesta seção são apresentados os resultados obtidos por meio da execução de cenários de testes, com o objetivo de validar o funcionamento do SMA e verificar como o mecanismo de negociação se comporta com a alteração do número de *drivers*. Para tanto, foram executados 10 cenários de testes, todos com 160 vagas de estacionamento. O número de *drivers* em cada cenário variou de 50 a 500, sendo acrescido de 50 *drivers* a cada cenário.

O Gráfico 1 (Figura 3) apresenta a variação do número médio de mensagens enviadas e o tempo médio para estacionar por *driver* em cada cenário de teste. No Gráfico 1 os valores médios do tempo para estacionar são multiplicados em 10 vezes, para que a escala fique mais próxima dos valores médios de mensagens enviadas e assim fazendo com que a análise fique mais clara.

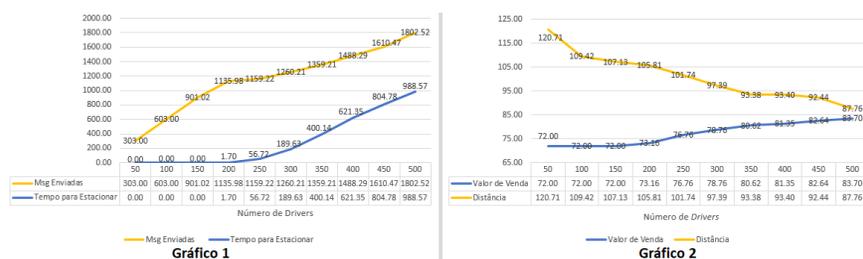


Figura 3. Gráficos de Análise

A partir da análise dos dados coletados da execução dos cenários de testes, foi possível constatar que o tempo para alocação da vaga de estacionamento tende a crescer conforme o número de *drivers* no sistema, como é possível verificar no Gráfico 1. Este fenômeno acontece devido ao fato de que, quanto maior o número de agentes no sistema mais mensagens serão enviadas para realizar a negociação das vagas. A troca de mensagens influi no tempo de alocação das vagas, pois quanto maior o número de mensagens trocadas para fechar uma negociação, mais demorada é esta negociação.

O Gráfico 2 (Figura 3) apresenta a variação da distância média entre o ponto onde o agente deseja a vaga e o ponto onde o agente efetivamente estacionou, juntamente com valor médio de venda das vagas em cada cenário. No Gráfico 2, os valores médios de venda das vagas são multiplicados em 12 vezes, para que a escala fique mais próxima dos valores de distância média e assim fazendo com que a análise fique mais clara.

Por meio da análise do Gráfico 2, foi possível constatar que, a distância média entre o ponto onde o agente deseja a vaga e o ponto onde o agente efetivamente estacionou diminui conforme o número de *drivers* no sistema aumenta. Conseqüentemente, o valor médio de venda das vagas tende a crescer conforme o número de *drivers* cresce, devido ao fato de que, quanto menor a distância entre estes pontos maior será o valor que um *driver* estará disposto a pagar. Este fenômeno acontece pois quanto maior o número de *drivers* no sistema, mais negociações irão acontecer. Portanto, os *drivers buyers* terão mais opções de vagas para negociar e como eles tendem a escolher vagas mais próximas ao ponto onde desejam estacionar, a distância média entre o ponto de desejo e o ponto onde estacionou diminui. Esse comportamento é o esperado conforme fórmulas do modelo de raciocínio (maiores informações podem ser consultadas em [Ducheiko et al. 2018]).

## 5. Conclusão

O desenvolvimento de projetos relacionados com o tema de Cidades Inteligentes é importante, melhorando a qualidade de vida das pessoas e contribuindo no desenvolvimento sustentável e inteligente das cidades. Estacionamentos Inteligentes contribuem neste aspecto, buscando alternativas para automatizar o gerenciamento de estacionamentos.

Com o propósito de validar o funcionamento do SMA e aperfeiçoar o modelo de raciocínio e protocolo de negociação, foram desenvolvidos testes com o Sistema Multiagente. Conforme apresentado, foram feitos experimentos para analisar o funcionamento do sistema, aqui destacam-se as seguintes características: i) existe uma curva de tendência de crescimento de mensagens trocadas entre os agentes conforme aumenta o número de agentes no sistema, ocasionando um maior tempo para a alocação das vagas de estacionamento; e ii) conforme o número de agentes cresce existe uma tendência de que os agentes estacionem em uma vaga mais próxima do ponto de desejo. Como trabalhos futuros podemos elencar: (i) realizar comparações com abordagens centralizadas; (ii) rodar experimentos utilizando dados reais; e (iii) desenvolver outros modelos de raciocínio.

## Referências

- Batty, M., Axhausen, K., Fosca, G., Pozdnoukhov, A., Bazzani, A., Wachowicz, M., Ouzounis, G., and Portugali, Y. (2012). Smart cities of the future.
- Castro, L. F. S. D., Alves, G. V., and Borges, A. P. (2017). Using trust degree for agents in order to assign spots in a Smart Parking. *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal*, 6(2):45–55.
- Choi, S., Liu, J., and Chan, S.-P. (2001). A genetic agent-based negotiation system. *Comput. Netw.*, 37(2):195–204.
- Ducheiko, F. F., André, P. B., and Gleifer, V. A. (2018). Implementação de Modelo de Raciocínio e Protocolo de Negociação para um Estacionamento Inteligente com Mecanismo de Negociação Descentralizado. *Revista Junior de Iniciação Científica em Ciências Exatas e Engenharia*, 1(19):25–32.
- Endriss, U., Maudet, N., Sadri, F., and Toni, F. (2006). Journal of artificial intelligence research. *Negotiating Socially Optimal Allocations of Resources*, 25:315–348.
- Koster, A., Koch, F., and Bazzan, A. (2014). Incentivising crowdsourced parking solutions.
- Nocera, D. D., Napoli, C. D., and Rossi, S. (2014). A social-aware smart parking application. *eur workshop proceedings*. 1260.
- O’Hare, G. M. P. and Jennings, N. R., editors (1996). *Foundations of Distributed Artificial Intelligence*. John Wiley Amp; Sons, Inc., New York, NY, USA.
- Wooldridge, M. (2009). *An Introduction to MultiAgent Systems*. J. Wiley, New York, 2nd edition.

# Um modelo de rede neural siamesa para re-identificação de pessoas em imagens, utilizando rede neural convolucional e *autoencoder*

Fábia Isabella Pires Enembreck<sup>1</sup>, Erikson Freitas de Moraes<sup>1</sup>

<sup>1</sup>Departamento de Informática – Universidade Tecnológica Federal do Paraná (UTFPR)  
Ponta Grossa – PR – Brasil

enembreck@alunos.utfpr.edu.br, emorais@utfpr.edu.br

**Abstract.** *The problem of determining whether a person being watched by a camera has ever been present in an environment is called person re-identification. This problem is considered challenging since the images obtained by cameras are subject to suffer variations, such as illumination, and partial occlusions. In this work, a method was developed for the problem, the method consists of using a siamese neural network architecture, composed of two identical subnets. Each subnet is formed by a convolutional neural network and a denoising autoencoder, responsible for rebuilding the vectors produced by the convolutional neural network, maintaining the most important features for the re-identification.*

**Resumo.** *O problema de determinar se uma pessoa que esta sendo observada por uma câmera já esteve presente em um ambiente é chamado de re-identificação da pessoas. Esse problema é considerado desafiador, pois as imagens estão sujeitas a sofrer variações, como iluminação, além de oclusões parciais. Este trabalho propõe um método para o problema, que consiste em utilizar uma rede neural siamesa, composta por duas sub-redes idênticas. Cada sub-rede é formada por uma rede neural convolucional e um denoizing autoencoder, responsável pela reconstrução dos vetores produzidos pela rede neural convolucional, mantendo as características mais importantes para a re-identificação.*

**Keywords:** Person re-identification; Deep Learning; Artificial neural networks

**Palavras-chave:** Re-identificação de pessoas; Aprendizagem profunda; Redes neurais artificiais

## 1. Introdução

A re-identificação de pessoas consiste em identificar se uma pessoa já esteve em um determinado ambiente antes, de forma que seja atribuído o mesmo rotulo a ela em todas as suas aparições [Ahmed et al. 2015], definida como o processo em que se pretende estabelecer correspondência entre diferentes imagens de uma mesma pessoa [Bedagkar-Gala and Shah 2014]. O problema de re-identificar pessoas em imagens é considerado muito complexo, uma vez que as imagens analisadas estão sujeitas a inúmeras variações de iluminação e pontos de vista, além de oclusões parciais e baixas resoluções,

dentre outros problemas. Dessa forma, duas imagens de uma mesma pessoa tiradas em uma mesma cena podem estar muito diferentes entre si, fazendo com que um re-identificador reconheça as imagens como sendo de duas pessoas distintas.

Vários métodos foram desenvolvidos para tratar esse problema, entre eles destacam-se métodos que utilizam a combinação de histogramas de duas imagens de pessoas, como em [Prosser et al. 2010] e [Gray and Tao 2008]. Além disso, aprendizagem de máquina é uma área que vem sendo utilizada para a re-identificação de pessoas, como em [Yi et al. 2014] e [Ahmed et al. 2015] que implementam variações de uma rede neural siamesa. No entanto, o problema ainda segue sem uma solução definitiva, devido as variações apresentadas pelas imagens.

Este trabalho propõe um método utilizando uma Rede Neural Siamesa para re-identificar pessoas em imagens. A rede é formada por duas sub-redes idênticas compostas por uma Rede Neural Convolutiva (CNN) e um *Autoencoder* (AE). Cada sub-rede produz um vetor de características de uma imagem e no final a Rede Neural Siamesa estima a similaridade entre os vetores para verificar se pertencem a uma mesma pessoa ou não. Para isso, a CNN produz um vetor de características gerais da imagem, que passam pelo AE que produz a saída de cada sub-rede, mantendo as características mais relevantes para a re-identificação, de forma que sejam amenizadas possíveis variações entre as imagens.

O objetivo deste trabalho é o desenvolvimento de um método para re-identificar pessoas em imagens utilizando técnicas de aprendizagem profunda. Para isso, foram determinados os seguintes objetivos específicos: 1) Identificar possíveis técnicas de aprendizagem profundas que possam ser aplicáveis ao problema; 2) Propor e implementar um modelo para re-identificação de pessoas; 3) Selecionar *datasets* públicos para testar o modelo proposto; 4) Realizar experimentos para validação do modelo.

## 2. Metodologia

Esta seção apresenta a metodologia proposta neste trabalho para a re-identificação de pessoas em imagens, obtidas por diferentes câmeras e pontos de vista. A Figura 1 exibe a Rede Neural Siamesa utilizada, que consiste em duas sub-redes, sendo que cada uma delas recebe uma imagem de entrada.

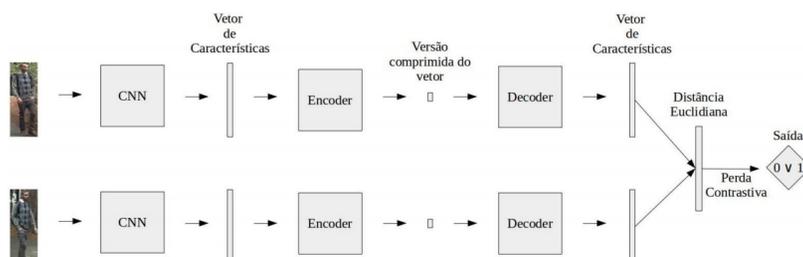


Figura 1. Modelo de Rede Neural Siamesa proposto

### 2.1. Rede Neural Siamesa

Uma Rede Neural Siamesa consiste em duas sub-redes idênticas que compartilham os mesmos parâmetros e são unidas em suas saídas. Cada sub-rede recebe uma entrada

diferente que é mapeada para um descritor de características. Com isso, a rede obtém dois descritores diferentes que são comparados para estimar a similaridade entre eles, resultando na saída da rede [Bromley et al. 1994].

Na Figura 1 é apresentado o modelo de Rede Neural Siamesa proposto. A entrada da rede é composta por um par de imagens e um valor binário  $Y$  que identifica se o par de imagens é de uma mesma pessoa ou não. Com isso, se  $Y = 0$  então as imagens são da mesma pessoa e se  $Y = 1$  as imagens são de pessoas diferentes.

Durante o treinamento, a rede modifica seus parâmetros para que possa diminuir a distância entre um par de imagens de uma mesma pessoa e aumentar entre um par de imagens de pessoas diferentes, de acordo com o rótulo  $Y$  recebido. Considerando que  $X_1$  e  $X_2$  representam o par de imagens de entrada e  $G_W$  é um conjunto de funções de rede, a função  $E_W = |G_W(X_1) - G_W(X_2)|$  é utilizada para calcular a distância e passa por uma Função de *Contrastive Loss*  $L$ .

A Função de *Contrastive Loss*, dada pela Equação 1, é utilizada para estimar a capacidade da rede de encontrar semelhanças entre as imagens. Para isso, a função aprende parâmetros, de forma que exemplos mais semelhantes se aproximem e os diferentes sejam separados.

$$L(W, Y, X_1, X_2) = (1 - Y) \frac{1}{2} (E_W)^2 + Y \frac{1}{2} \{ \max(0, m - E_W) \}^2 \quad (1)$$

onde  $m$  é a margem, se  $m > 0$  e  $(X_1, X_2)$  é um par de imagens positivo e  $(X'_1, X'_2)$  é um par de imagens negativo, então  $E_W(X_1, X_2) + m < E_W(X'_1, X'_2)$  [Bromley et al. 1994].

## 2.2. Rede Neural Convolutacional - CNN

A primeira parte de cada sub-rede da rede proposta consiste em uma CNN, responsável por produzir um vetor de características da imagem de entrada. Esta CNN é formada por 4 camadas de convolução, alternadas entre uma camada de *max-pooling* e outra de normalização.

Uma CNN é caracterizada pela utilização de operações de convolução, em pelo menos uma de suas camadas, para aprender padrões de um determinado conjunto de dados. No caso de um conjunto de dados formado por imagens, a camada de convolução é formada por um conjunto de filtros, produzindo na sua saída mapas de características de uma imagem [LeCun et al. 2015].

Uma camada de agrupamento *pooling* é utilizada com objetivo de reduzir o tamanho dos mapas de características produzidos, de forma que se diminua também o número de parâmetros e custo computacional. O tipo de agrupamento utilizado foi o *max-pooling*, que, para uma região 2x2 dos mapas de características, substitui no *pixel* correspondente do mapa de característica da saída o maior valor dessa região [LeCun et al. 2015].

## 2.3. Denoising Autoencoder - DAE

Uma rede neural do tipo *autoencoder* (AE) é formada por pelo menos 3 camadas: entrada, saída e camada intermediária, sendo que a saída deve ter o mesmo tamanho da entrada,

uma vez que o AE procura reproduzir a entrada da rede na sua saída. Isso é possível através de uma função codificadora, que extrai características da entrada e produz um vetor de características comprimido. A descompressão deste vetor é realizada através de uma função decodificadora que traduz as informações o mais próximo possível da entrada, priorizando as propriedades mais importantes dos dados [Goodfellow et al. 2016].

Um tipo de AE é o *Autoencoder* de Denonização ou *Denoising Autoencoder* (DAE). Este modelo recebe dados corrompidos como entrada e, durante o treinamento, procura prever os dados de entrada não corrompidos em sua saída [Goodfellow et al. 2016]. Com isso, um DAE foi utilizado na rede para, a partir do vetor gerado pela CNN, produzir uma representação de forma que sejam reconstituídas as informações mais relevantes para a re-identificação durante o treinamento da rede.

### 3. Experimentos e Resultados

Esta seção apresenta os resultados obtidos com o treinamento e teste da rede neural siamesa proposta. A rede foi implementada utilizando *Python*, através das bibliotecas para redes neurais *Keras* e *Tensorflow*.

#### 3.1. VIPeR

O *dataset ViewPoint Invariant Pedestrian Recognition* - (ViPER) contém um total de 1264 imagens de 632 pedestres, assim, para cada pedestre, existem duas imagens capturadas por diferentes câmeras, contendo variação entre os pontos de vista e também mudanças na iluminação e nas poses dos indivíduos [Gray and Tao 2008].

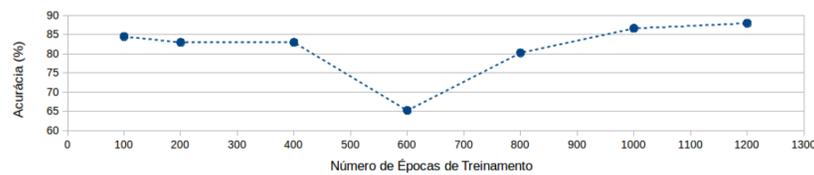
Com objetivo de aumentar o número de imagens de treinamento e reduzir o *overfitting*, foi aplicada a técnica de *data augmentation*. Assim, para cada imagem, foi aplicada 11 transformações, resultando em um total de 15144 imagens. Os resultados obtidos após o treinamento e teste podem ser observados na Tabela 1 e na Figura 2. Neste teste, a maior acurácia obtida foi com 1200 épocas de treinamento, atingindo uma porcentagem de 87.93%. O treinamento com 600 épocas obteve o pior resultado, com uma acurácia de 65.2%.

**Tabela 1. Acurácia em relação ao N° de épocas de treinamento, utilizando o dataset VIPeR**

Épocas	Acurácia (%)	Épocas	Acurácia (%)
100	84,47	800	80,23
200	82,92	1000	86,59
400	82,92	1200	87,93
600	65,2		

#### 3.2. iLIDS-VID

O *dataset iLIDS Video Re-Identification* (iLIDS-VID) contém imagens de 300 pessoas diferentes obtidas através de 2 câmeras não sobrepostas. Esse *dataset* disponibiliza um maior número de imagens do que o ViPeR, uma vez que as imagens foram obtidas



**Figura 2. Acurácia em relação ao N° de épocas de treinamento, utilizando o dataset VIPeR**

por meio de rastreamento de pedestres, contendo de 23 a 192 *frames* para cada pessoa. As imagens tem diversas variações, como iluminação, pontos de vista e oclusões [Wang et al. 2014].

No primeiro experimento com o iLIDS-VID, foram utilizadas 21969 imagens de 319 pessoas diferentes obtidas por duas câmeras, sendo que foram destinadas 14751 imagens para treinamento e 7218 imagens para testes. Os resultados podem ser vistos na Tabela 2 e na Figura 3, na linha representada por "CNN+AE". A melhor acurácia foi 94.26% para 600 épocas de treinamento e o pior foi 89.37% para 1400 épocas de treinamento.

**Tabela 2. Acurácia em relação ao N° de épocas de treinamento, utilizando o dataset iLIDS-VID**

Épocas	Acurácia (%)	Épocas	Acurácia (%)	Épocas	Acurácia (%)
100	93,04	1000	92,4	2000	93,22
200	93,49	1200	89,82	2200	92,06
400	91,2	1400	89,37	2400	92,45
600	94,26	1600	94,09	2600	93,46
800	91,39	1800	90,93	2800	93,82

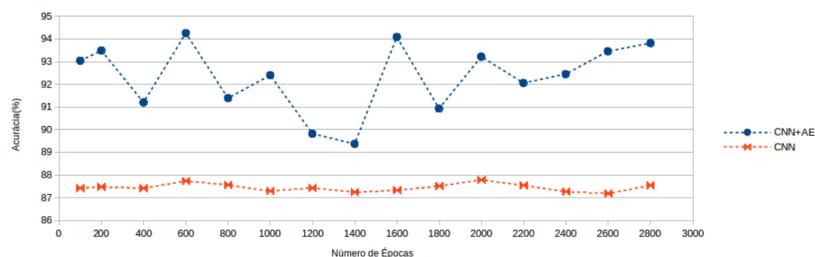
Para comparação, também foi realizado um experimento com sub-redes compostas apenas pela CNN. A Tabela 3 e a Figura 3 na linha representada por "CNN" apresentam os resultados obtidos. A melhor acurácia foi alcançada com 2000 épocas de treinamento, 87, 78% e a pior foi igual a 87, 19% para 2600 épocas de treinamento.

**Tabela 3. Acurácia em relação ao N° de épocas de treinamento, para sub-redes formadas apenas pela CNN, utilizando o dataset iLIDS-VID**

Épocas	Acurácia (%)	Épocas	Acurácia (%)	Épocas	Acurácia (%)
100	87,42	1000	87,30	2000	87,78
200	87,48	1200	87,43	2200	87,54
400	87,42	1400	87,25	2400	87,27
600	87,73	1600	87,33	2600	87,19
800	87,56	1800	87,51	2800	87,54

#### 4. Conclusão e Trabalhos Futuros

Um modelo de aprendizagem profunda foi proposto para o problema de re-identificação de pessoas em imagens, que consiste em uma rede neural siamesa, formada por duas sub-



**Figura 3. Comparação das acurácias em relação ao N<sup>o</sup> de épocas de treinamento, usando o *dataset* iLIDS-VID para as duas redes implementadas.**

redes idênticas, para estimar a similaridade em duas imagens. Cada sub-rede é composta por uma CNN e um DAE. Através de experimentos realizados, verificou-se que a rede siamesa proposta possui grande potencial na re-identificação de pessoas, visto que obteve resultados melhores do que a rede com apenas uma CNN, muito utilizada em outros trabalhos da área, em todos os casos. Como trabalhos futuros, pretende-se realizar mais testes usando outros *datasets* e verificar o desempenho da rede para diferentes qualidades de imagens, condições de iluminação e oclusões parciais.

## Referências

- Ahmed, E., Jones, M., and Marks, T. K. (2015). An improved deep learning architecture for person re-identification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3908–3916.
- Bedagkar-Gala, A. and Shah, S. K. (2014). A survey of approaches and trends in person re-identification. *Image and Vision Computing*, 32(4):270–286.
- Bromley, J., Guyon, I., LeCun, Y., Säckinger, E., and Shah, R. (1994). Signature verification using a “siamese” time delay neural network. In *Advances in neural information processing systems*, pages 737–744.
- Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep learning*. MIT press.
- Gray, D. and Tao, H. (2008). Viewpoint invariant pedestrian recognition with an ensemble of localized features. In *European conference on computer vision*, pages 262–275. Springer.
- LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.
- Prosser, B. J., Zheng, W.-S., Gong, S., Xiang, T., and Mary, Q. (2010). Person re-identification by support vector ranking. In *BMVC*, volume 2, page 6.
- Wang, T., Gong, S., Zhu, X., and Wang, S. (2014). Person re-identification by video ranking. In *European conference on computer vision*, pages 688–703. Springer.
- Yi, D., Lei, Z., Liao, S., and Li, S. Z. (2014). Deep metric learning for person re-identification. In *2014 22nd International Conference on Pattern Recognition*, pages 34–39. IEEE.

# Uma proposta de Gerenciamento de Consumo Hídrico para a IoT

Laryssa A. M. S. Ferreira<sup>1</sup>, Maria A. Trinta<sup>1</sup>, Carlos E. Pantoja<sup>1</sup>, Vinicius S. Jesus<sup>1</sup>

<sup>1</sup>Centro Federal de Educação Tecnológica Celso Suckow (CEFET/RJ)  
20785-220 – Rio de Janeiro – RJ – Brasil

laryssa.afeerreira@gmail.com, atg200210@outlook.com,

pantoja@cefet-rj.br, souza.vdj@gmail.com

**Abstract.** *The main objective of this work is to bring a proposal for monitoring and management of two water tanks (hydric system) done by technologies of a smart hub. The work also searches to help people who have difficulty to reach their water tanks offering virtual access to their water tanks.*

**Resumo.** *Este trabalho tem como objetivo trazer uma proposta de monitoramento e gerenciamento de duas caixas d'água (sistema hídrico), sendo esse monitoramento e/ou gerenciamento realizado através de tecnologias em um ambiente inteligente. Com essa conjuntura, o trabalho busca também suprir a necessidade de acesso das pessoas que, de suas residências, possuem dificuldades de alcance ao sistema hídrico, atendendo essa demanda com a virtualização desse sistema.*

**Keywords:** *Internet of Things; Javino; Raspberry; Water consumption; Water tanks; AmI System.*

**Palavras-chave:** *Internet das coisas; Javino; Raspberry; Consumo Hídrico; Caixas d'água; Ambiente inteligente; Virtualização.*

## 1. Introdução

Automação residencial [Simplício et al. 2018] é um ramo da automação que busca simplificar a vida do indivíduo através da implementação de diversas tecnologias em sua residência, tornando possível o controle, o gerenciamento e a obtenção de dados de diferentes instalações da casa, oferecendo conforto, segurança e economia aos moradores. Nota-se também que projetos como este trazem facilidade de manutenção para pessoas como idosos ou deficientes.

Com a escassez dos recursos naturais, o desperdício de água é um grave problema que vem ganhando foco nos últimos tempos, justamente por ser a água o recurso básico para a existência de vida. Assim, este trabalho propõe uma solução de gerenciamento das atividades hídricas de uma residência que possui a impossibilidade de monitoramento/gerenciamento manual em função da distância entre a residência e o sistema hídrico (Exemplo: Residência em área de mata e a caixa d'água perto da fonte de água).

A solução vem através da virtualização do sistema hídrico que proporciona ao usuário a visualização dos status do nível de volume de água das caixas d'água. Essa visualização possui funcionalidades que possibilitam ao usuário que monitore ou/e gereencie o sistema do modo desejado.

A visualização e obtenção das informações sobre os níveis de volume acontece por meio de sensores presentes nas caixas d'água e o sistema que faz a integração dos componentes disponibilizando os dados através da Internet das coisas (*Internet of Things ou IoT*) [Singer 2012]. A IoT pode ser compreendida como uma rede que interliga diferentes dispositivos, possibilitando um fluxo de dados com a conexão entre eles e um servidor que armazena as informações, construindo um ambiente inteligente e independente.

Este trabalho está organizado da seguinte maneira: a segunda seção apresenta os referenciais teóricos, a terceira seção expõe trabalhos relacionados, a quarta metodologia a quinta seção propõe uma discussão sobre o trabalho, a sexta seção encerra com algumas considerações finais e por fim são exibidas as referencias utilizadas.

## 2. Referencial Teórico

Nesta seção são apresentadas as tecnologias e conceitos necessários para o entendimento do trabalho, auxiliando na compreensão geral do mesmo já que reúne diversas tecnologias diferentes, como A Internet das Coisas, Sistemas Embarcados, Web Services e Javino.

A Internet das Coisas (*Internet of Things (IoT)*) [Singer 2012] é uma rede que interliga múltiplos dispositivos que atuam de forma independente sendo capazes de tomar decisões, com ou sem a supervisão humana. Essa rede permite que haja um tráfego de dados entre os dispositivo e o *software* permitindo a transmissão desses dados para o servidor que atua em conjunto com a IoT [Afonso et al. 2015]. Tendo como uma de suas funcionalidades manter o fluxo de dados no servidor utilizado, esses dados são enviando a uma aplicação na linguagem Java que, por sua vez, as envia para serem armazenadas em um banco de dados afim de oferecer *web services*, seja numa interface gráfica ou em uma página *web*, possibilitando acesso remoto [Eulalio et al. 2017].

Construídos para atender exclusivamente a uma única aplicação, oferecendo ao usuário muitas vezes um dispositivo completo e independente, os sistemas embarcados [de Souza et al. ] são softwares que, diferente dos presentes em computadores, não possuem capacidade ou permissão para a execução de inúmeros aplicativos. Por outro lado, por serem exclusivos de uma única aplicação, oferecem os suficiente para uma execução mais rápida e com menos chances de erros.

Em Arquiteturas que possuem sistemas embarcados distintos, que precisam tanto comunicar-se entre si quanto executar funções independentes, é necessária a presença de um *middleware*, um agente que estabeleça a comunicação entre eles. O Javino [Lazarin and Pantoja 2015] é um *middleware* que foi criado para estabelecer a comunicação entre a linguagem *Arduíno* e a linguagem *JAVA*, oferecendo uma interface de comunicação serial de mão dupla, dispondo também de um protocolo de prevenção de perda de informação que torna a comunicação mais segura.

As informações dos dispositivos e do servidor estão armazenadas em um banco

de dados relacional que possui uma Linguagem de Consulta Estruturada simples e com nível aceitável de segurança, sendo possível acompanhar as atualizações da linguagem sem perder em funcionalidade [Júnior et al. 2018]. Esse armazenamento de informações permite que os dispositivos sejam trocados ou que o sistema inteiro possa mudar de local sem que haja perda dessas informações.

### 3. Trabalhos Relacionados

Nesta seção são abordados os trabalhos relacionados que buscam sanar o problema de monitoramento de Rios, Caixas d'água e do volume de água, no geral, trazendo as diferenças entre eles e a proposta apresentada neste trabalho.

Existe um trabalho que desenvolveu um monitoramento micro-controlado em tempo real de um reservatório residencial de água sendo este monitoramento realizado via *bluetooth* possibilitando ao usuário acesso remoto ao sistema [Oliveira et al. 2014]. Outro trabalho, também, desenvolveu um monitoramento de sistema de níveis de líquidos através da plataforma Arduino que controla a alimentação do tanque de água em função da capacidade máxima e mínima [Gimenes and Pereira 2015].

Entretanto, no trabalho [Oliveira et al. 2014] há um limite na distância visto que o *Bluetooth* tem essa limitação, enquanto, a proposta de gerenciamento permite ao usuário viajar para outras cidades ou lugares possibilitando o gerenciamento/monitoramento do sistema virtualizado em tempo real independente da distância. Já no segundo trabalho [Gimenes and Pereira 2015] restringe-se somente ao monitoramento da alimentação do tanque em função dos níveis de água, enquanto, em nossa proposta o usuário pode ter a escolha de monitorar ou gerenciar independente do nível de água.

### 4. Metodologia

Nesta seção é apresentada a metodologia referente ao desenvolvimento do trabalho através de fluxogramas que exemplificam como cada tecnologia utilizada funciona e como elas se interligam formando a rede de compartilhamento de informações da *IoT*.

#### 4.1. Fluxograma do Dispositivo

O fluxograma de baixo nível (Figura 1) representa o funcionamento do dispositivo e a interação com a caixa d'água em um ambiente externo ao servidor demonstrando que a distância não é um problema.

O sistema embarcado em conjunto com a *Raspberry* envia comandos através do Javino afim de obter informações sobre os níveis de água da caixa d'água presentes no Arduino. O Arduino, por sua vez, realiza a captação dessas informações por meio dos sensores e atuadores conectados a caixa d'água. Esse funcionamento faz com que o Arduino atue em modo escravo, pois o torna incapaz de realizar a tomada de decisões, dependendo sempre da *Raspberry* para realizar algum comandos. O funcionamento do dispositivo com essa arquitetura permite a atuação de forma independente a *IoT*, pois continuará a captar informações mesmo que ela não realize uma solicitação.

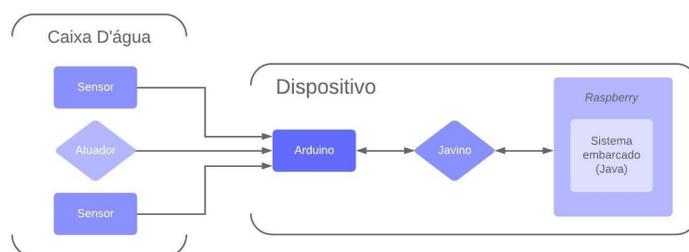


Figura 1. Fluxograma de baixo nível

### 4.2. Fluxograma intermediário

O fluxograma intermediário auxilia na visualização da comunicação entre os polos do dispositivo e da *IoT*. Visto que a passagem de informações entre eles não é realizada de forma direta para que não haja perda de informação.

A comunicação entre o ambiente inteligente (*IoT*) e o dispositivo que faz parte desse ambiente ocorre por meio do *middleware* (*ContextNet*) [Endler et al. 2011] que centraliza as informações impedindo o detrimento delas ao serem enviadas para a *Iot*. Ao serem recebidas pelo *ContextNet*, as informações são enviadas para uma aplicação na linguagem Java responsável por armazená-las no Banco de Dados.

### 4.3. Fluxograma alto nível

O fluxograma abaixo (Figura 2) representa o funcionamento da (*IoT*) e do servidor após a chegada das informações do *ContextNet*. Ao recebe-las, o *ContextNet* as envia para uma aplicação na linguagem Java que realiza o envio para o banco de dados que está instalado no servidor. Esse banco de dados é responsável por armazenar as informações afim de oferecer *web services* em página *Web* e interface gráfica de um dispositivo móvel.

Os *web services* dão ao usuário a possibilidade de consultar o funcionamento do sistema e utilizar as funcionalidades dos serviços tanto na página *web* quanto no dispositivo móvel. Os serviços oferecidos são de acesso remoto e isso permite que o usuário os acesse através de um aplicativo de um aparelho celular ou computador independente da distância em que ele esteja já que o sistema está virtualizado na *IoT*.

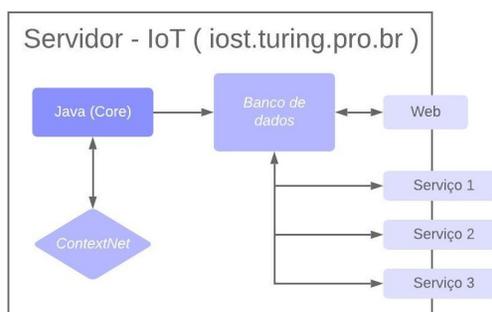
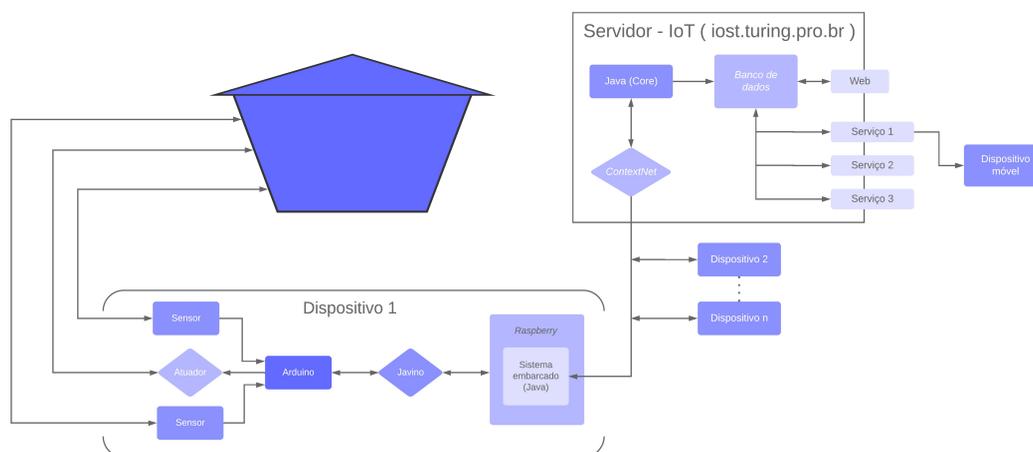


Figura 2. Fluxograma alto nível

### 4.4. Fluxograma final

O fluxograma final (Figura 3) representa o sistema em funcionamento com todas as arquiteturas anteriores reunidas. Nela é possível visualizar os dispositivos interligados na *IoT*,

a caixa d'água conectada ao dispositivo por meio dos sensores e as infinitas possibilidades de dispositivos interligados nessa *IoT*.



**Figura 3. Fluxograma final**

## 5. Discussão

O trabalho foi desenvolvido através da capacitação dos conceitos envolvidos e teste a nível de protótipo. Essa capacitação ocorreu através dos artigos presentes no referencial teórico, pois foi necessário que os conceitos estivessem claros. Enquanto, o teste do projeto foi realizado com dois potes de sorvete, uma bomba d'água de aquário, um Arduino, sensores ultrassônicos, sensor de fluxo de passagem de água e válvulas eletroeletrônicas, sendo esse protótipo uma representação do sistema hídrico proposto.

Os testes do protótipo começaram a ser realizados a partir da escolha do sensor ultrassônico, pois ele vê a água como barreira sendo possível obter dados do volume de água através dele. Logo, foi realizado os testes das válvulas que são do tipo normalmente aberta, ou seja, permitem ao programador em questão a escolha das situações em que ela pode ser aberta/fechada permitindo a passagem/impedimento da água. A partir disso, foi feita uma pré-programação para testar as válvulas e os sensores em conjunto com os potes de sorvete sendo coletado o volume de água.

Com os testes do protótipo, foi possível realizar o monitoramento e gerenciamento hídrico, pensando nas pessoas que moram longe de suas caixas d'água, de forma remota e rápida assim facilitando a identificação de um problema externo ou interno por meio dos sensores, mas vale ressaltar que não é possível ter acesso remoto caso haja ausência de sinal da internet para acessar a nuvem sendo fornecida a opção de verificação manual por meio de um painel de *led* do lado de fora da caixa d'água.

## 6. Considerações Finais

Inspirado no conceito de automação residencial, o trabalho propõe a criação de um mecanismo que simplifique o cotidiano, tornando o gerenciamento das atividades hídricas mais confortável uma vez que ele pode ser feito a qualquer momento por meio de dispositivos ou serviços *web*. A *IoT* torna-se parte indispensável de toda a arquitetura, uma vez que

representa a rede que interliga todos os componentes do trabalho, tornando o processo de controle dos recursos da residência simples e prático.

Para futuros trabalhos pretende-se desenvolver um mecanismo que consiga verificar se há algum problema nas estruturas da caixa d'água, evitando um desperdício de recursos. É considerado também o desenvolvimento de um instrumento integrado ao trabalho que verifique a salubridade da caixa d'água, alertando ao usuário da necessidade uma nova limpeza, bem como armazenamento das datas das limpezas anteriores.

## Referências

- Afonso, B. S., Pereira, R. B. d. O., and Pereira, M. F. L. (2015). Utilização da internet das coisas para o desenvolvimento de miniestação de baixo custo para monitoramento de condições do tempo em áreas agrícolas. *Anais da Escola Regional de Informática da Sociedade Brasileira de Computação (SBC)–Regional de Mato Grosso*, 6:183–189.
- de Souza, M. A. F., Denis, E., and Fernandes, J. C. L. Utilização de um hardware embarcado (raspberry pi) usando programação em blocos (scratch) para ensino de física em escolas secundárias e universidades.
- Endler, M., Baptista, G., Silva, L., Vasconcelos, R., Malcher, M., Pantoja, V., Pinheiro, V., and Viterbo, J. (2011). Contextnet: Context reasoning and sharing middleware for large-scale pervasive collaboration and social networking. In *Proceedings of the Workshop on Posters and Demos Track*, page 2. ACM.
- Eulalio, A. D., Cordeiro, D., and de Souza, R. (2017). Web services: Integração de sistemas orientado a serviços com uma proposta de aplicação na ead. *Revista de Informática Aplicada*, 12(2).
- Gimenes, A. and Pereira, P. (2015). Utilização da plataforma arduino como ferramenta de controle para um sistema de nível de líquidos. *Blucher Chemical Engineering Proceedings*, 1(3):2553–2558.
- Júnior, D. A. B., Paes, R. B., Peixoto, S. C., Santos, A. F., and Maniçoba, R. H. C. (2018). Tetris: Ide visual para desenvolvimento de aplicações java sem codificação manual. *Ciência & Desenvolvimento-Revista Eletrônica da FAINOR*, 11(2).
- Lazarin, N. M. and Pantoja, C. E. (2015). A Robotic-Agent Platform for Embedding Software Agents using Raspberry Pi and Arduino Boards. In *9<sup>th</sup> Software Agents, Environments and Applications School*.
- Oliveira, I. R., Marco, A., and Santos, C. (2014). Desenvolvimento de um aplicativo android para monitoramento microcontrolado do nível de um reservatório de água residencial em tempo real. *XII CEEL, outubro de*.
- Simplício, P. V. G., Lima, B. R., and da Silva, G. S. (2018). Automação residencial: Uma solução social e econômica. *Caderno de Graduação-Ciências Exatas e Tecnológicas-UNIT-ALAGOAS*, 4(3):17.
- Singer, T. (2012). Tudo conectado: conceitos e representações da internet das coisas. *Simpósio em tecnologias digitais e sociabilidade*, 2:1–15.

# Desenvolvimento de uma ontologia para representação de objetos e ações do código de trânsito em um veículo autônomo

Vithor Tozetto Ferreira<sup>1</sup>, Gleifer Vaz Alves<sup>1</sup>

<sup>1</sup>Universidade Tecnológica Federal do Paraná (UTFPR)  
Ponta Grossa – PR – Brazil

vithorferreira@alunos.utfpr.edu.br, gleifer@utfpr.edu.br

**Abstract.** *There are many aspects about the development of autonomous vehicles (AV), among them the interaction of those vehicles with traffic laws. An AV, acting in traffic, will need to follow a set of rules, and its controller will need to be able to utilize these rules in the planning of its actions. The controller of an AV could be modelled as an intelligent agent, still being necessary a representation of the traffic rules, representation which could be done through an ontology. In this paper is presented the proposal of an ontology representing objects and actions referring to traffic rules, which could be utilized as a tool for assisting on the behavior of high-level actions of an AV.*

**Resumo.** *Existem vários aspectos sobre o desenvolvimento de veículos autônomos (VA), entre eles a interação desses veículos com as leis de trânsito. Um VA, agindo no trânsito, deverá seguir um conjunto de regras, e seu controlador deverá ser capaz de utilizar essas regras no planejamento de suas ações. O controlador de um VA pode ser modelado como um agente inteligente, sendo ainda necessária uma representação das regras de trânsito, representação que pode ser feita através de uma ontologia. Neste trabalho é apresentada a proposta de uma ontologia para representação de objetos e ações referentes às regras de trânsito, que poderá ser utilizada como uma ferramenta para auxiliar no comportamento das ações de alto-nível de um VA.*

**Keywords:** Ontologies; Autonomous Vehicles; Traffic Rules

**Palavras-chave:** Ontologias; Veículos Autônomos; Regras de Trânsito

## 1. Introdução

Nos últimos anos, ocorreu um crescimento no desenvolvimento de novas tecnologias automatizadas. Para a indústria automotiva, o lançamento de veículos autônomos (VA's) será um grande salto tecnológico. Com o advento dos VA's, o motorista não precisaria estar no controle de um veículo, e poderia dedicar este tempo de locomoção a realização de outras tarefas [Silberg et al. 2012]. Um dos grandes problemas que impedem a utilização cotidiana de um VA é o de garantir a segurança do funcionamento desses veículos, e no contexto deste artigo, especificamente a segurança do comportamento do VA estar em acordo com as leis de trânsito. Através da união de várias tecnologias, os VA's atuais

conseguem perceber obstáculos em seus ambientes e interagir com segurança e autonomia em certas situações [Gomes 2014]. Porém, esta capacidade de autonomia ainda não é suficiente para garantir a segurança do funcionamento do VA, sendo necessário que o veículo considere em suas ações as regras de trânsito do local onde está trafegando [Vellinga 2017].

Conforme mencionado por Prakken [Prakken 2017], a interação de um VA com as regras de trânsito em geral não é tratada nas etapas de desenvolvimento de um VA. O VA precisa ser capaz de perceber seu ambiente, considerar suas tarefas e planejar ações seguras condizentes com as leis de trânsito locais para atingir seus objetivos de forma autônoma. Portanto, um VA deve ser controlado por um sistema computacional capaz de realizar tal tarefa, e uma das maneiras de representar tal sistema é através de um agente.

De maneira geral, um agente é uma entidade situada em certo ambiente, capaz de realizar ações autônomas para atingir seus objetivos [Wooldridge and Jennings 1995]. A um agente inteligente, são também atribuídas as propriedades de autonomia, habilidade social, reatividade e proatividade. Tais propriedades são interessantes para um VA, já que podem ser utilizadas no contexto de situações reais de trânsito. Mesmo com a utilização de um agente, uma grande dificuldade do desenvolvimento de um VA capaz de agir de forma segura no trânsito ainda permanece: como fazer com que um sistema computacional autônomo utilize as regras de trânsito em seu planejamento de ações.

A adaptação de um conjunto de regras de trânsito para o contexto de um mecanismo autônomo e inteligente é afetada pelas leis estarem descritas em linguagem natural, com a eventual presença de ambiguidades, redundâncias e incoerências [Prakken 2017]. Se faz necessário utilizar uma representação destas leis, que poderá ser utilizada no controle do VA, e tal representação pode ser feita com o auxílio de uma ontologia. Uma ontologia é uma especificação de uma conceitualização, um modelo de dados que possui um domínio, elementos e a relação entre esses elementos e o domínio [Cimiano et al. 2014]. Com uma ontologia, é possível criar uma conexão entre o domínio da linguagem natural e da linguagem artificial, auxiliando o VA a interpretar o ambiente e as leis.

O objetivo específico deste artigo é criar uma representação dos elementos utilizados pelas leis dos Códigos de Trânsito Brasileiro [BRASIL 1997] e Britânico [Department for Transport 2017], visando também mostrar a possibilidade de que um VA se adapte a mais de um conjunto de leis. Este trabalho está diretamente relacionado com o trabalho desenvolvido por Alves et. al [Alves et al. 2018], onde é apresentada uma formalização (inicial) de um conjunto de regras de trânsito do Reino Unido usando lógica temporal, e tais regras são incorporadas em um agente inteligente, modelado com um VA. Especificamente, o artigo proposto aqui descreve objetos e ações através de uma ontologia, que representa elementos das regras de trânsito referentes a cruzamentos de vias urbanas. Tal representação visa facilitar a incorporação das regras no agente inteligente que representa o VA.

## **2. Veículos Autônomos e código de trânsito**

Na literatura da área, é muito comum encontrar em publicações de montadoras de veículos o termo direção automatizada, e raramente o termo direção autônoma. O primeiro termo

envolve um conjunto de ferramentas que auxiliam na direção, enquanto o segundo se refere ao estado final da automação, onde o sistema teria controle total sobre todas as funções de controle do veículo [Herrmann et al. 2018]. Em resumo, um veículo autônomo (VA) é um veículo controlado totalmente por um sistema, sem o auxílio de um motorista. Para garantir a segurança de suas ações no quesito de tráfego urbano, é necessário que o VA leve em consideração as Regras de Trânsito do local onde está situado.

Conforme mencionado por Prakken [Prakken 2017] e reforçado por Alves et. al [Alves et al. 2018], existe uma lacuna no desenvolvimento de VA's no que diz respeito aos seguintes aspectos: i.) A implantação de um VA considera adequadamente as regras de trânsito? ii.) O comportamento de um VA no tráfego urbano dá-se em acordo com as regras de trânsito? iii.) É necessário em alguma instância alterar e adaptar as regras de trânsito para o adequado comportamento de um VA? Esses aspectos reforçam a necessidade de representar o conhecimento das regras de trânsito para que sejam utilizadas por um agente representando um VA.

De acordo com [BRASIL 1997], se configura como trânsito a movimentação e imobilização de veículos, pessoas e animais nas vias terrestres. Cada país, dentro de seus territórios, delimitam um conjunto de regras destinadas a controlar e garantir a segurança do tráfego. É esperado que todo usuário presente no trânsito esteja ciente destas regras, e que as cumpra visando manter a harmonia do trânsito. Portanto, os VA's precisarão se adequar as regras de circulação de seus ambientes de atuação. Além disso, também foi considerada a possibilidade de um VA transitar entre dois territórios que possuam legislações de trânsito distintas, e que o controlador do veículo precise se adaptar ao novo conjunto de regras durante seu percurso. Para o desenvolvimento deste trabalho, são considerados dois fragmentos referentes as regras de cruzamentos em vias urbanas dos Códigos de Trânsito do Brasil e do Reino Unido.

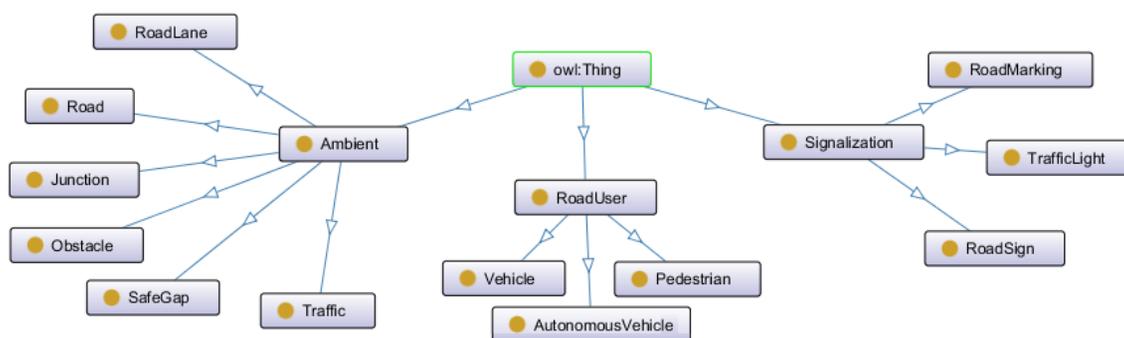
### 3. Ontologia para as regras de cruzamentos de vias

A definição de ontologia pode mudar de acordo com o autor, porém, é comum na grande maioria das definições o termo conceitualização, se referindo a uma visão de mundo, uma forma de descrever um domínio, seus objetos e as relações existentes entre tais objetos [Cimiano et al. 2014]. As ontologias possuem algumas propriedades essenciais: ontologias descrevem um domínio específico; a utilização dos termos deve ser consistente; os conceitos e relações devem ser definidos sem ambiguidades em uma linguagem formal; as relações entre os conceitos determinam a estrutura da ontologia; ontologias podem ser compreendidas e processadas por computadores [Freitas et al. 2017].

A partir destes conceitos foi desenvolvida a *Road Junction Ontology*, demonstrada nas figuras 1 e 2, que visa representar os objetos e ações referentes a um ambiente de trânsito, extraídos a partir de [BRASIL 1997] e [Department for Transport 2017], especificamente aquelas regras referentes a cruzamentos de vias urbanas.

*Road Junction Ontology*: Ontologia que representa os objetos e ações presentes no ambiente de trânsito de acordo com um fragmento de regras dos Códigos de Trânsito do Brasil e Reino Unido.

Na figura 1 são apresentados os objetos presentes em um ambiente de trânsito. Os objetos estão divididos em três grupos, referentes ao ambiente, aos usuários das vias terrestres e aos objetos de sinalização



**Figura 1. Road Junction Ontology - Objetos**

do trânsito. Esses objetos estão descritos da seguinte forma:

**RoadUser:** Os usuários das vias terrestres, presentes no ambiente de trânsito.

**Pedestrian:** Usuário que não utiliza um veículo.

**Vehicle:** Meio de transporte destinado à locomoção de passageiros ou cargas, com alguma forma de propulsão, controlado por um ser humano e que circula nas vias terrestres.

**AutonomousVehicle:** Um meio de transporte destinado à de passageiros ou cargas, com alguma forma de propulsão, controlado por um sistema autônomo inteligente, que pode eventualmente contar com a intervenção de um humano e que circula nas vias terrestres.

**Ambient:** Por onde os usuários das vias terrestres trafegam e interagem com outras entidades.

**Obstacle:** Objeto que interfere na circulação do trânsito, e exige a atenção dos veículos.

**Junction:** Interseção em nível de duas vias, e apresenta regras específicas para a interação dos usuários com sua superfície.

**Road:** Superfície por onde transitam todos os usuários das vias terrestres, composta de faixas (RoadLanes).

**RoadLane:** Subdivisões longitudinais de uma via que delimitam o espaço de circulação dos veículos.

**Traffic:** O conjunto de usuários da via em movimento no ambiente.

**SafeGap:** Espaço seguro para que um usuário execute uma manobra. Uma das grandes dificuldades é representar o conceito abstrato de “segurança” referente a este objeto.

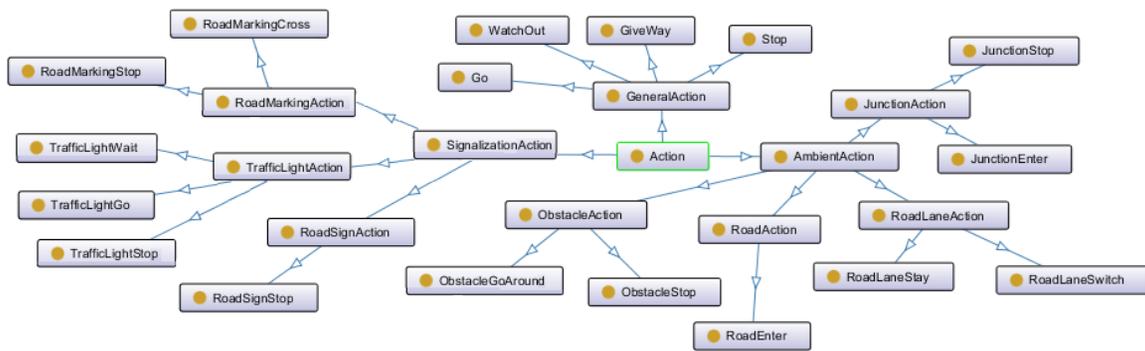
**Signalization:** Objetos que ordenam ou dirigem a circulação no ambiente.

**RoadSign:** Objeto que informa os usuários sobre normas de circulação ou condições do ambiente. Ex: Placas de Trânsito.

**TrafficLight:** Objeto de sinalização luminosa, que indica a prioridade de passagem no local. Ex: Semáforo.

**RoadMarking:** Objeto de sinalização horizontal que informa os usuários sobre normas de circulação ou condições do ambiente. Ex: Faixa de Pedestre.

Na figura 2 são apresentadas as ações presentes em um ambiente de trânsito. As ações foram criadas a partir dos objetos da ontologia, representando as possíveis ações que o agente poderá tomar para interagir com os objetos do trânsito. Essas ações são descritas da seguinte forma:



**Figura 2. Road Junction Ontology - Ações**

**Action:** As ações que o VA possui para interagir com os objetos do trânsito.

**GeneralAction:** Ações generalizadas, aplicáveis em contextos simples e necessárias para representar o possível comportamento do VA em diferentes situações.

**Go:** Ir, mover o veículo.

**Stop:** Parar, imobilizar o veículo.

**WatchOut:** Observar, averiguar a segurança da situação.

**GiveWay:** Dar passagem a outro usuário.

**AmbientAction:** Ações que interagem com o ambiente.

**JunctionAction:** Ações que interagem com cruzamentos.

**JunctionEnter:** Adentrar um cruzamento.

**JunctionStop:** Parar diante de um cruzamento.

**ObstacleAction:** Ações que interagem com obstáculos.

**ObstacleGoAround:** Desviar de um obstáculo.

**ObstacleStop:** Parar diante de um obstáculo.

**RoadLaneAction:** Ações que interagem com as faixas de circulação.

**RoadLaneStay:** Permanecer na faixa.

**RoadLaneSwitch:** Mudar de faixa.

**RoadAction:** Ações que interagem com as vias.

**RoadEnter:** Adentrar uma via.

**SignalizationAction:** Ações que interagem com a sinalização.

**RoadSignAction:** Ações que interagem com placas.

**RoadSignStop:** Parar diante de uma placa.

**TrafficLightAction:** Ações que interagem com os semáforos.

**TrafficLightGo:** Avançar diante de um semáforo.

**TrafficLightStop:** Parar diante de um semáforo.

**TrafficLightWait:** Aguardar a permissão de passagem.

**RoadMarkingAction:** Ações que interagem com a sinalização horizontal.

**RoadMarkingStop:** Parar diante de sinalização horizontal.

**RoadMarkingCross:** Atravessar sinalização horizontal.

## 4. Considerações Finais

Este trabalho visa obter conclusões referentes a questões presentes na literatura de VA's, como aquelas levantadas por Prakken [Prakken 2017] e Vellinga [Vellinga 2017]: quais meios podem ser utilizados para a representação das regras de trânsito para o contexto de um VA; quais são as limitações de um VA no que se refere a compreensão e execução das regras de trânsito; e quais mudanças podem ser necessárias na legislação de trânsito para a adequação do uso de VA's. A partir do trabalho de Alves et. al [Alves et al. 2018], este artigo propôs, de forma generalizada, a representação de conceitos que fossem comuns a dois conjuntos de regras diferentes, considerando a possibilidade de mobilidade internacional de um VA. Para criar esta representação foi necessária a generalização de objetos e ações, tentando adequar ambas as legislações em uma única representação que ainda fizesse sentido no contexto individual das regras de trânsito dos países.

Como trabalho futuro, pretende-se implementar está ontologia, permitindo que um sistema inteligente a utilize como uma base de conhecimento, então incorporar a ontologia em um agente inteligente modelado como um VA utilizando a linguagem de programação de agentes *Gwendolen* [Dennis 2017]. Após a incorporação da ontologia no agente, será possível realizar a verificação formal do comportamento do VA em relação às regras de trânsito, através da ferramenta AJPF (*Agent Java Pathfinder*) [Dennis et al. 2012]. Devido à generalização da ontologia, existe também a possibilidade de estender a representação, de forma a englobar conceitos além daqueles que se referem a cruzamentos urbanos.

## Referências

- Alves, G. V., Dennis, L., and Fisher, M. (2018). Formalisation of the Rules of the Road for embedding into an Autonomous Vehicle Agent. In *International Workshop on Verification and Validation of Autonomous Systems*, pages 1–2, Oxford, UK.
- BRASIL, D. (1997). Código brasileiro de trânsito. [http://www.planalto.gov.br/ccivil\\_03/LEIS/L9503.htm](http://www.planalto.gov.br/ccivil_03/LEIS/L9503.htm).
- Cimiano, P., Unger, C., and McCrae, J. (2014). Ontology-based interpretation of natural language. *Synthesis Lectures on Human Language Technologies*, 7(2):1–178.
- Dennis, L. A. (2017). *Gwendolen Semantics: 2017*. Technical Report ULCS-17-001, University of Liverpool, Department of Computer Science.
- Dennis, L. A., Fisher, M., Webster, M. P., and Bordini, R. H. (2012). Model checking agent programming languages. *Automated Software Engineering*, 19(1):5–63.
- Department for Transport, U. (2017). Using the road (159 to 203) - The Highway Code - Guidance - GOV.UK.
- Freitas, A., Bordini, R. H., and Vieira, R. (2017). Model-driven engineering of multi-agent systems based on ontology. *Applied Ontology*, 12(2):157–188.
- Gomes, L. (2014). Hidden obstacles for google's self-driving cars. *MIT Technology Review*.

- Herrmann, A., Brenner, W., and Stadler, R. (2018). *Autonomous driving: how the driverless revolution will change the world*. Emerald Publishing, Bingley North America Japan India Malaysia China, first edition. OCLC: 1031123857.
- Prakken, H. (2017). On the problem of making autonomous vehicles conform to traffic law. *Artificial Intelligence and Law*, 25(3):341–363.
- Silberg, G., Wallace, R., Matuszak, G., Plessers, J., Brower, C., and Subramanian, D. (2012). Self-driving cars: The next revolution. *White paper, KPMG LLP & Center of Automotive Research*, page 36.
- Vellinga, N. E. (2017). From the testing to the deployment of self-driving cars: legal challenges to policymakers on the road ahead. *Computer Law & Security Review*, 33(6):847–863.
- Wooldridge, M. and Jennings, N. R. (1995). Intelligent agents: Theory and practice. *The knowledge engineering review*, 10(2):115–152.

# Performance Analysis of SDN Virtualization

Davi Daniel Gemmer<sup>1</sup>, Augusto Foronda<sup>1</sup>

<sup>1</sup>Department of Computer Science  
Federal University of Technology - Paraná (UTFPR)  
Post-Graduation Program in Computer Science

gemmer@alunos.utfpr.edu.br, foronda@utfpr.edu.br

**Abstract.** *Software Defined Network (SDN) is an emerging networking paradigm to overcome the limitations of a traditional network infrastructure. SDN can improve the monitoring, management, security and traffic engineering of a network. A solution for SDN architecture is through Virtual Machine (VM) and container. However, there is a lack of performance comparison between VM and container with SDN. This paper analyzes the virtualization of SDN using VM and Docker container in terms of performance. Simulations were done with OpenDaylight SDN controller and an evaluation was done to understand the efficiency and the scalability of the network. The results show that compared with virtual machines, containers can scale a larger number of flows. This paper can help the network designers to make a better SDN architecture in terms of which virtualization method should be used.*

**Keywords –** *SDN, Virtual Machine, container, performance.*

## 1. Introduction

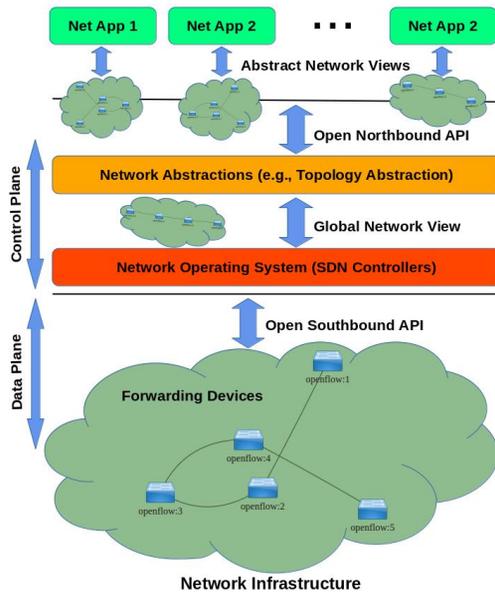
Traditional network technology has inherent problems of rigid structure and complex configuration and cannot meet the requirement of network innovation which demands dynamic and flexible management. Software Defined Network (SDN) is proposed to overcome the problems of a traditional network and presents a new idea with three layers: data plane, control plane and application plane. Data plane is composed by network devices and forwards packets according to a decision made by the control plane, which acts as a mediator for the data plane and the application plane and handles the traffic flow in the network. Application plane achieves customized application, such as network automation, network management and network security [Joy 2015]. SDN architecture involves the usage of virtualization technologies such as virtual machines (VMs) and containers. VMs are extensively used as they permit workloads to be isolated from each other and for the resources to be well controlled. However, containers have the great advantage of allowing applications to run separately from the host infrastructure, such as Docker container. Some papers have shown that Docker container can offer superior performance compared to VM [Felter et al. 2015]. This paper presents an SDN controller built on VMs and Docker containers. We perform experiments to evaluate the memory and number of flows.

This paper is organized as follows: Section II discusses the SDN architecture. In Section III, we present a comparison between VM and container. Section IV describes the related work. Section V present experiments and give the results. Finally, Section VI presents the conclusion of our work.

## 2. SDN Architecture

Figure 1 shows the SDN architecture. There are 3 layers: data layer, control layer and application layer. Data layer can communicate with control layer through Southbound interface and control layer can communicate with application layer through Northbound interface. Data layer is composed of networking equipment's which forms the network infrastructure to forward, modify or drop network traffic according to the instructions given by controller. Control Layer is composed by the SDN controller, which is a logical entity that receives instructions from application layer and relays them to the networking equipment's. SDN controller manages the network and it has logic control for switching, routing, firewall security rules, etc. The SDN controller also extracts information about the network and send it back to the application layer such as statistics and events about the traffic. Application layer is composed by programs that controls a set of resources of one or more SDN controllers via application programming interfaces (APIs) [Joy 2015]. Southbound interfaces are APIs that enables the communication between control layer and data layer. OpenFlow is the first and probably most well-known southbound interface. However, it is not the only one available. Northbound interfaces enable the communication between control layer and application layer and it is configured through REST APIs of SDN controllers, which are used to facilitate automation of the network to align with the needs of different applications with SDN network programmable [Felter et al. 2015].

Figure 1. SDN Architecture



Adapted from [Kreutz et al. 2014]

One of the most famous SDN controllers is OpenDayLight (ODL), which provides a centralized management system that allows to have a programmable network. ODL controller can be used as a platform for configuring different aspects of the network and solving different network challenges. ODL uses open source integration standards and APIs to make the network more programmable, intelligent and adaptable [Bhimani et al. 2017].

### 3. Container vs Virtual Machine (VM)

System virtualization separates the underlying physical device and the upper operating system. A single physical machine can be divided in multiple machines to maximize the resource utilization and flexibility [Peng et al. 2009].

There are some differences between these two technologies: 1) A container share the same OS (operating system) on the same machine with other containers and VMs do not share the OS. Therefore, container is more lightweight than a VM; 2) VM needs an hypervisor to translate an instruction that can be executed by the host because a VM runs in a non-privileged mode. On the contrary. A container does require an extra layer because it communicates with the OS through the system calls; 3) Each VM has its own image file, while different containers may share some of their images. These differences shows some advantages of using containers: the size of a container is smaller than the size of a VM and a container usually takes less hardware resources since it does not need to maintain an OS [Bedhief et al. 2016].

A container packs together all the necessary software that the application needs to run with, such as the libraries and the other dependencies. There are some particular features that enables different containers to run on the same physical machine, such as Cgroups and namespaces [Morabito 2016]. Cgroup allows system administrators to allocate resources such as CPU, memory, network to the running containers, which can be adjusted dynamically. However, it can not use more resources than specified in the cgroups. Namespace guarantees that processes running in different containers will not conflict with each other because it provides IDs, network interfaces and host names for each container [Bedhief et al. 2016].

### 4. RELATED WORK

Many research efforts have been made to explore the advantages of containers as well as to compare the containers with the VMs. Some papers have investigated the container performance. It was showed that a container running in a Raspberry Pi 2 has almost the same performance compared to native execution [Morabito 2016]. Miguel et al demonstrated that a container technology achieves a very low overhead compared to native setups. Other papers have compared VMs and containers. A comparison has been made between Linux containers and virtual machines in terms of the performance and extensibility [Joy 2015]. Performance analysis has been made of virtual machine (VM) application and deployments and compares them with the Linux containers [Felter et al. 2015]. Janki et al. [Bhimani et al. 2017] compares the performance of Spark jobs running in a container cluster and a fix-sized VM cluster with one physical machine. Wes et al. [Felter et al. 2015] showed a better container performance than VM in terms of disk and network I/O. And some papers have demonstrated solutions with SDN and container. IoT networks and applications are extremely complex and an SDN architecture with Docker container has been proposed to manage such networks. The experiments proved the feasibility of the proposed architecture and the communication established between smart devices through an SDN-based network [Bedhief et al. 2016]. Y. Xu et al proposed an SDN-based autodocker framework integrated within the switches for enabling auto-docking/-undocking of applications at the edge switches. In this manner, the Docker framework automatically and effectively manages the storage, computing, and

networking resources of the switch [Xu et al. 2016]. VMs and containers have been used in some architectures, such as radio access network (RAN). A. Gopalasingham et al compared their performance and analyzed the two different architectures. Measurements and comparisons confirmed that the Docker container based architecture could provide a superior performance compared to the VM based architecture [Xingtao et al. 2016].

## 5. PERFORMANCE EVALUATION

Our evaluation considers OpenDayLight SDN controller. The performance metrics considered are: memory use and average flows. The main goal is to investigate which virtualization method gives the best result with these metrics. The evaluation is carried out using WCBench [Farrell 2019], a performance measurement tool to benchmark SDN controllers.

### 5.1. Test Environment

Both WCBench and ODL controller were implemented on the same machine IBM X3850 M2 with (4x Intel® Xeon™ Processor X7350 CPU @ 2.93GHZ (4cores)), 64 GB of memory was available. The system was running CentOS 7  $\times 86\_64$  (1810) and the virtual machine was virtualized through the ESXi 6.0.0 Update 3. Docker was running with centos 7 container image, for tests the OpenDaylight Oxygen 0.8.4 controller was used.

### 5.2. Methodology

WCBench was used to emulate 128 switches and the tests were configured to run a loop of 2000 cycles with a duration of 4 minutes per cycle. For the execution of the tests, a total of 16 virtual processors and 60 GB of RAM were allocated for both virtualization technologies. The table 1 brings the settings used in WCBench.

**Table 1. Parameters for WCBench test and ODL config**

NUM SWITCHES	128
NUM MACS	100000
TESTS PER SWITCH	10
MS PER TEST	10000
CBENCH WARMUP	1
KARAF SHELL PORT	8101
CONTROLLER	OpenDaylight
CONTROLLER IP	localhost
CONTROLLER PORT	6633
SSH HOSTNAME	cbench

Adapted from [Farrell 2019]

### 5.3. Average RAM usage

When comparing the results between the two virtualization technologies, it was noticed that the Docker container had a slightly higher RAM utilization when compared to the virtual machine, with a difference of 131 MB more for the same experiment as can be seen in the figure 2, where the virtual machine used an average of 2224 MB of RAM and the Docker container had an average of 2355 MB of RAM. WCBench analyzes the RAM used throughout the operating system, which has caused a problem due to the need to share host resources for OpenvSwiath (OVS) operation within the Docker container.

Consequently the analysis considered the use of host memory during testing while the hypervisor isolated the VM.

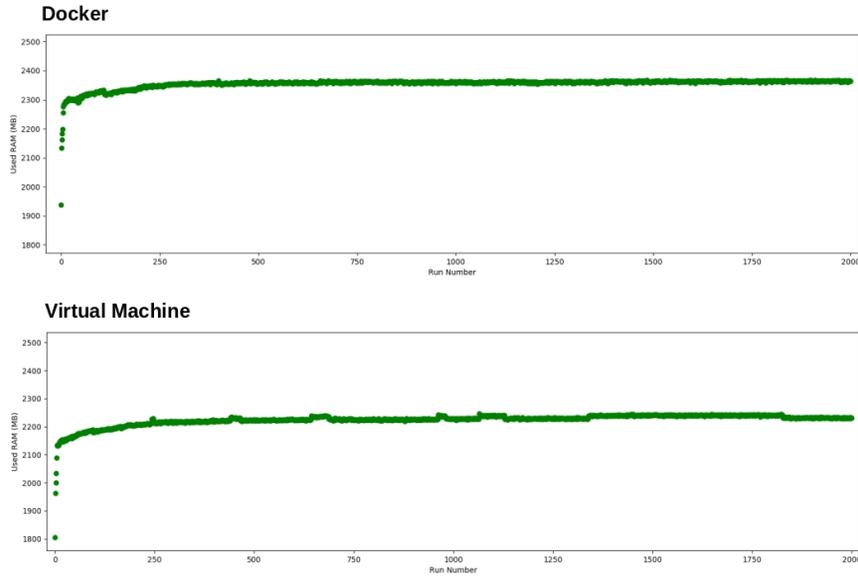


Figure 2. Used RAM

### 5.4. Average Flows per Second

When analyzing the average of transmitted flows, the container Docker obtained an average of 5,287 flows per second where the virtual machine obtained an average of 4,703 flows per second as can be seen in the figure 3. In the same test the container Docker obtained a difference of 584 more flows than the virtual machine.

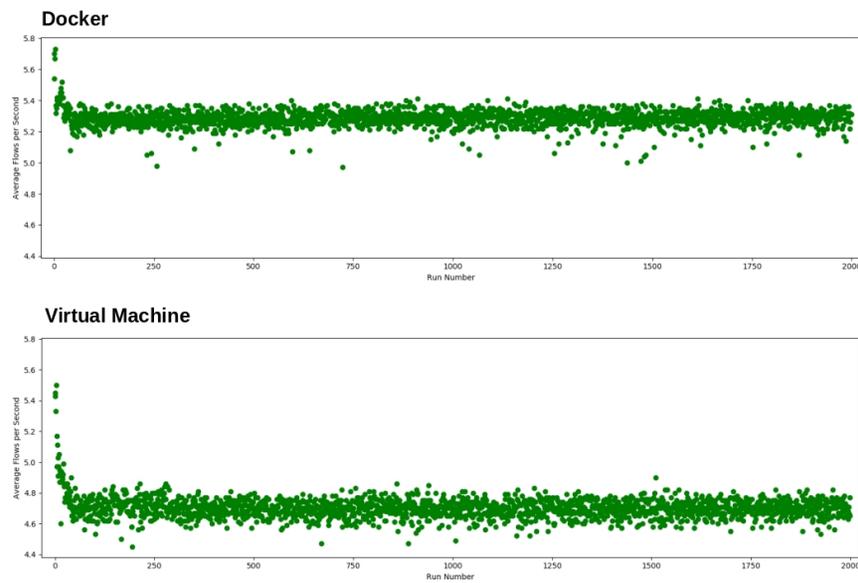


Figure 3. Average Flows

## 6. CONCLUSION

This paper presents a comparison performance evaluation of ODL SDN controller using VM and Docker container. From the performance evaluation, Docker container exhibited the best throughput results showing that it is able to respond to requests more promptly under traffic loads. This work provide users with guidelines towards which virtualization method provides a better performance. The use of RAM was stable and did not obtain much difference in the two cases but the small advantage of VM led us to believe it was due to the way WCBench performs RAM analysis.

## References

- Bedhief, I., Kassar, M., and Aguilu, T. (2016). Sdn-based architecture challenging the iot heterogeneity. In *2016 3rd Smart Cloud Networks & Systems (SCNS)*, pages 1–3. IEEE.
- Bhimani, J., Yang, Z., Leeser, M., and Mi, N. (2017). Accelerating big data applications using lightweight virtualization framework on enterprise cloud. In *2017 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–7. IEEE.
- Farrell, D. (2019). Wcbench. Available in: <https://github.com/dfarrell107/wcbench>. Access in June 2019.
- Felter, W., Ferreira, A., Rajamony, R., and Rubio, J. (2015). An updated performance comparison of virtual machines and linux containers. In *2015 IEEE international symposium on performance analysis of systems and software (ISPASS)*, pages 171–172. IEEE.
- Joy, A. M. (2015). Performance comparison between linux containers and virtual machines. In *2015 International Conference on Advances in Computer Engineering and Applications*, pages 342–346. IEEE.
- Kreutz, D., Ramos, F., Verissimo, P., Rothenberg, C. E., Azodolmolky, S., and Uhlig, S. (2014). Software-defined networking: A comprehensive survey. *arXiv preprint arXiv:1406.0440*.
- Morabito, R. (2016). A performance evaluation of container technologies on internet of things devices. In *2016 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 999–1000. IEEE.
- Peng, J., Zhang, X., Lei, Z., Zhang, B., Zhang, W., and Li, Q. (2009). Comparison of several cloud computing platforms. In *2009 Second international symposium on information science and engineering*, pages 23–27. IEEE.
- Xingtao, L., Yantao, G., Wei, W., Sanyou, Z., and Jiliang, L. (2016). Network virtualization by using software-defined networking controller based docker. In *2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference*, pages 1112–1115. IEEE.
- Xu, Y., Mahendran, V., and Radhakrishnan, S. (2016). Sdn docker: Enabling application auto-docking/undocking in edge switch. In *2016 IEEE conference on computer communications workshops (INFOCOM WKSHPS)*, pages 864–869. IEEE.

## Abordagem Entrópica de Codificação de Fonte usando compressão de dados

Kaio Pablo Gomes<sup>1</sup>, Eduardo El Akkari Sallum<sup>2</sup>, Everton Leonardo Skeika<sup>3</sup>

<sup>1</sup>Departamento de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR) – Ponta Grossa, PR – Brazil

<sup>2</sup>Departamento de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR) – Ponta Grossa, PR – Brazil

<sup>3</sup>Departamento de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR) – Ponta Grossa, PR – Brazil

kaiopablo@live.com, esallum@gmail.com, everton\_skeika@hotmail.com

**Abstract.** *This paper presents an entropic coding approach for the transmission of information between a source and a destination interconnected under a LAN network architecture. In this scenario, an entropic font coding system using Huffman, with variable code size based on unequivocal frequencies of appearance of English letters, was proposed. To make this change in the form of coding, it is suggested to insert pre-coding and pre-decoding steps in the local network data transmission model. Huffman coding results show that in a 1460 textit bytes data packet, it is possible to send up to 2768 symbols, whereas currently with the ASCII system it can only send up to 1460. In the scenario considered, the methodology has 1.89 times more symbols to contribute to channel efficiency. Also, the efficiency of the Huffman coding obtained was superior to an alternative entropic approach evaluated by the Shannon-Fano technique, 99.19% against 99.16%, respectively.*

**Resumo.** *Este trabalho apresenta uma abordagem de codificação entrópica para a transmissão de informação entre uma fonte e um destino interligados sob uma arquitetura de rede LAN. Neste cenário, foi proposto um sistema de codificação entrópica de fonte utilizando Huffman, com tamanho de código variável baseado em frequências não-equiprováveis de aparição das letras da língua inglesa. Para realizar essa mudança na forma de codificar, sugere-se a inserção de etapas de pré-codificação e pré-decodificação no modelo de transmissão de dados em redes locais. Os resultados da codificação de Huffman demonstram que em um pacote de dados de 1460 bytes, é possível enviar até 2768 símbolos, enquanto que atualmente com o sistema ASCII pode ser enviado apenas até 1460. No cenário considerado, a abordagem comporta 1,89 vezes mais símbolos de modo a contribuir com a eficiência do canal. Também, a eficiência da codificação de Huffman obtida foi superior em relação a uma abordagem entrópica alternativa avaliada por meio da técnica de Shannon-Fano, 99,19% contra 99,16%, respectivamente.*

**Keywords:** Entropic coding; Local Area Networks; Huffman; ASCII; Data Transmission; Channel Efficiency; Shannon-Fano.

**Palavras-chave:** Codificação Entrópica; Redes Locais; Huffman; ASCII; Transmissão de Dados; Eficiência do Canal; Shannon-Fano.

## 1. Introdução

Os sistemas de redes locais estão fundamentados no padrão *Ethernet*. Nesse modelo de transmissão de dados, a comunicação é por meio de pacotes de dados denominados quadros ou *frames*, segundo o modelo OSI. Dentre outras especificações da tecnologia *Ethernet*, determina-se a técnica de codificação da fonte e do canal. A codificação da fonte é dada por meio da tabela *ASCII* para a representação dos símbolos transmitidos ou recebidos, enquanto que a do canal é por meio do modelo *Manchester* [Tanenbaum et al. 2003].

A eficiência de um canal está associada com a quantidade de dados que podem ser enviados em uma transmissão de mensagem [Forouzan 2009]. Ao possibilitar maior agregação de informação em um único pacote de dados, aumenta-se a eficiência do canal levando em consideração um cenário ideal. Nesse cenário, não considera outros fatores que afetam a comunicação de dados como: tempo de atraso, colisões de pacotes, dentre outros.

Neste trabalho, para permitir que mais informações possam ser agregadas em um único pacote, propõe-se uma abordagem entrópica de codificação de fonte usando compressão de dados. O objetivo principal é apresentar uma abordagem capaz de proporcionar maior eficiência para o canal considerando a transmissão de dados em redes locais.

## 2. Abordagem Proposta para Transmissão de Dados em Redes Locais

A abordagem proposta neste trabalho visa melhorar a comunicação entre a fonte e o destino utilizando uma codificação entrópica. Analisando o padrão dos sistemas de redes locais, a fonte utiliza a tabela *ASCII* para codificar as mensagens. No entanto, essa técnica de codificação é fundamentada no princípio de geração de códigos de tamanho fixo, sendo qualquer símbolo da tabela equiprovável.

A análise estatística representada pela Tabela 1 mostra a frequência das letras da língua inglesa [Zim 1962]. Conforme os valores identificados, as letras do alfabeto nesse idioma assim como em outros não são equiprováveis. Ao considerar um sistema de codificação em que não conta com a possibilidade de existir diferentes probabilidades, afeta-se a eficiência do processo de transmissão.

A ideia principal da nova abordagem que está sendo proposta é utilizar uma quantidade variável de informação por símbolo. Sendo assim, um processo entrópico de codificação é adotado. No entanto, conhecendo-se apenas as probabilidades de ocorrência das letras de um alfabeto, este trabalho visa codificar apenas tais tipos de caracteres. Ainda, considera-se letras pertencentes ao alfabeto inglês, uma vez que as frequências de utilização das mesmas segue a análise mostrada na Tabela 1.

Para conceder essa melhoria na transmissão de dados, a fonte passa a utilizar um componente adicional chamado pré-codificação. O componente inserido é responsável

Tabela 1. Tabela da Esquerda

Letra	Frequência
e	12.702%
t	9.056%
a	8.167%
o	7.507%
i	6.966%
n	6.749%
s	6.327%
h	6.094%
r	5.987%
d	4.253%
l	4.025%
c	2.782%
u	2.758%

Tabela 2. Tabela da Direita

Letra	Frequência
m	2.406%
w	2.360%
f	2.228%
g	2.015%
y	1.974%
p	1.929%
b	1.492%
v	0.978%
k	0.772%
j	0.153%
x	0.150%
q	0.095%
z	0.074%

por permitir um processo de codificação alternativo para a fonte. Tanto a codificação por meio da tabela ASCII, como a entrópica de Huffman, são possíveis uma vez que verifica-se inicialmente os símbolos que formam a mensagem a ser enviada. Toda vez que uma letra é identificada, utiliza-se o algoritmo de Huffman para codificar tal caractere. Por outro lado, qualquer outro símbolo que não seja uma letra do alfabeto adotado, mapeia-se em código ASCII respeitando a mesma estrutura já existente nos sistemas de redes locais.

Além da fonte, o destino precisa sofrer um ajuste necessário para conseguir lidar com a abordagem entrópica. Por isto, é necessário adicionar um pré-decodificador, o qual é acionado quando uma mensagem composta apenas de letras é recebida. Essa mensagem não chega a ser decodificada pela maneira usual, consultando a tabela ASCII. O pré-decodificador passa a consultar uma nova tabela referente ao mapeamento das letras em códigos Huffman.

As mensagens que antes seriam enviadas por uma única transmissão de dados podem depender de mais de um envio de pacote de dados. Essa relação de dependência está vinculada com o processo entrópico proposto, que modela apenas símbolos referentes a letras. Não é possível misturar a codificação de Huffman e ASCII, uma vez que o processo de decodificação não saberia interpretar os dados transmitidos. Sendo assim, uma mensagem pode ser quebrada em um ou mais envios de dados, dependendo inclusive da taxa de MTU (*Maximum Transmission Unit*) que é seguida no padrão *Ethernet*.

## 2.1. Compressão de Dados

A abordagem entrópica adotada para a fonte, permite uma codificação mais eficiente por meio de compressão de dados. A informação representada em cada símbolo varia de acordo com a probabilidade de ocorrência do mesmo. Para esse trabalho, duas técnicas de compressão de dados foram estudadas e consideradas: algoritmo de Shannon-Fano e algoritmo de Huffman. Apesar de as duas técnicas de compressão poderem apresentar resultados similares em alguns contextos, o funcionamento delas são diferentes na forma como

cada técnica manipula uma árvore de codificação [Kodituwakku and Amarasinghe 2010]. A eficiência da codificação e da fonte utilizando essas técnicas de compressão de dados podem variar de acordo com o contexto da aplicação, inclusive entre ambas as técnicas os resultados podem apresentar uma diferença considerável. A Tabela 3 apresenta a aplicação do algoritmo de Shannon-Fano para esse conjunto de símbolos, respeitando as suas probabilidades de ocorrências identificadas pela Tabela 1.

Para compararmos os resultados obtidos pela codificação de Shannon-Fano, utilizamos a codificação de Huffman. A Tabela 4 mostra o resultado da codificação de Huffman. Nota-se que ambos os algoritmos apresentaram resultados mais eficientes do que o sistema ASCII, em termos de quantidade de *bits*, para representar cada letra do alfabeto inglês.

Tabela 3. Codificação Shannon

Shannon-Fano		
Símbolo	Probabilidade	Codificação
e	12.702%	000
t	9.056%	001
a	8.167%	0100
o	7.507%	0101
i	6.966%	0110
n	6.749%	0111
s	6.327%	1000
h	6.094%	1001
r	5.987%	1010
d	4.253%	1011
l	4.025%	11000
c	2.782%	11001
u	2.758%	11010
m	2.406%	11011
w	2.360%	11100
f	2.228%	111010
g	2.015%	111011
y	1.974%	111100
p	1.929%	111101
b	1.492%	111110
v	0.978%	1111110
k	0.772%	11111110
j	0.153%	1111111100
x	0.150%	1111111101
q	0.095%	1111111110
z	0.074%	1111111111

Tabela 4. Codificação Huffman

Huffman		
Símbolo	Probabilidade	Codificação
e	12.702%	100
t	9.056%	000
a	8.167%	1110
o	7.507%	1101
i	6.966%	1011
n	6.749%	1010
s	6.327%	0111
h	6.094%	0110
r	5.987%	0101
d	4.253%	11111
l	4.025%	11110
c	2.782%	01001
u	2.758%	01000
m	2.406%	00111
w	2.360%	00110
f	2.228%	00101
g	2.015%	110011
y	1.974%	110010
p	1.929%	110001
b	1.492%	110000
v	0.978%	001000
k	0.772%	0010011
j	0.153%	001001011
x	0.150%	001001010
q	0.095%	001001001
z	0.074%	001001000

Ao avaliar as duas codificações geradas, percebe-se que ambos os sistemas de

codificação apresentam resultados diferentes para o contexto da abordagem proposta. Nesse cenário de aplicação, para criar a tabela de Shannon-Fano precisa de 148 *bits*, enquanto que a de Huffman precisa de 143 *bits*. Esse valor é a soma dos *bits* de cada símbolo, nesse caso as letras de a-z. O comprimento médio de *bits* por símbolo é 4,21977 *bits* para a codificação Shannon, enquanto que utiliza-se 4,21854 *bits* para a de Huffman. Considerando que é a entropia da fonte  $H(X)$  é de 4,18 *bits* por mensagem, calcula-se a relação entre a entropia e comprimento médio dos símbolos. A eficiência da codificação de Shannon é de 99,16%, enquanto que a de Huffman é 99,19%. Como a eficiência de Huffman apresentou um melhor desempenho, a abordagem proposta utiliza a codificação Huffman para a sua abordagem entrópica de compressão de dados.

## 2.2. Avaliação da Abordagem Proposta

A abordagem proposta foi avaliada por meio da análise dos resultados implicados pela utilização de compressão de dados pela fonte. É feita uma comparação entre a quantidade de dados enviados por pacote em sistemas de redes locais. A técnica de compressão adotada pela abordagem é a de Huffman. O gráfico ilustrado pela Figura 1 mostra a comparação entre a abordagem proposta e os sistemas de redes locais que utilizam a tabela ASCII.

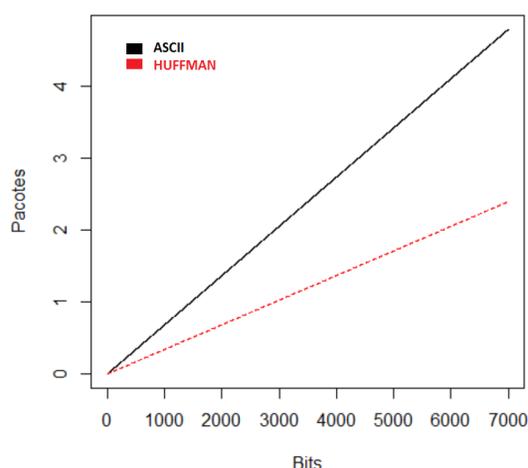


Figura 1. Gráfico comparativo de pacotes por *bits* do Huffman com o ASCII

O gráfico presente na Figura 1 mostra que o uso da abordagem proposta possibilita enviar aproximadamente 1,98 vezes mais símbolos em cada pacote de dados. O comprimento médio dos símbolos com a codificação Huffman é de 4,21854 *bits*, enquanto que o tamanho médio com o código ASCII é de 8 *bits*. Considerando o cenário de redes locais com *Ethernet*, o tamanho máximo de dados que um pacote de dados comporta é 1460 bytes ou 11680 *bits*. Em apenas um único pacote com dados comprimidos pela abordagem proposta é possível enviar aproximadamente até 2768 símbolos. No entanto, ao utilizar dados codificados em ASCII pode-se enviar apenas 1460 símbolos.

Por meio desta avaliação, identifica-se que a aplicação de uma abordagem entrópica de compressão de dados possibilita o aumento da eficiência do canal. Essa

melhoria é possível uma vez que a fonte passa a codificar símbolos com menos dados, e conseqüentemente uma maior quantidade de símbolos podem ser enviados pelo canal. Ressalta-se que os símbolos comprimidos pela abordagem proposta são apenas as letras do alfabeto inglês, qualquer outro caractere tais como números, pontos, dentre outro não são considerados. Para realizar a transmissão desses outros símbolos, a execução de transmissão considerando codificação ASCII é adotada.

### 3. Conclusão

De acordo com os resultados obtidos na avaliação da abordagem sugerida, a compressão de dados pode aumentar a eficiência do canal. Conhecendo a probabilidade de ocorrência dos símbolos codificados pela fonte, torna-se possível uma abordagem entrópica na qual dados podem ser transmitidos de forma comprimida. A escolha da codificação, nesse trabalho, é feita com base na comparação da eficiência entre dois algoritmos de compressão de dados: Huffman e Shannon-Fano.

A abordagem sugere que a fonte realize uma combinação dos sistemas de codificação ASCII e Huffman. No contexto aplicado, considera-se a utilização de Huffman para a transmissão de símbolos referentes as letras do alfabeto inglês, enquanto que o sistema ASCII codifica todos seus caracteres com exceção das letras. Para possibilitar essa mudança nos sistemas de rede local, um processo de pré-codificação e pré-decodificação é proposto.

A principal vantagem associada a aplicação da abordagem proposta é a possibilidade de enviar mais informações em um único pacote de dados. No entanto, outros fatores associados ao impacto de adicionar mecanismos de codificação e decodificação de código Huffman pode afetar a eficiência da transmissão de dados.

A extensão dos estudos da abordagem proposta pode ser feita por meio de um trabalho futuro relacionado a aplicação da abordagem em cenários reais considerando o tempo de atraso dos pacotes, confiabilidade e outros parâmetros avaliativos. A eficiência da codificação e a redução do número de transmissões de dados, são parâmetros que podem ser aprimorados ao utilizar um único sistema de representação de símbolos nas fonte. Um estudo para avaliar a probabilidade de ocorrência de símbolos em uma transmissão pode proporcionar a utilização de um único sistema de codificação.

### Referências

- Forouzan, B. A. (2009). *Comunicação de dados e redes de computadores*. AMGH Editora.
- Kodituwaku, S. and Amarasinghe, U. (2010). Comparison of lossless data compression algorithms for text data. *Indian journal of computer science and engineering*, 1(4):416–425.
- Tanenbaum, A. S. et al. (2003). *Computer networks*, 4-th edition. ed: Prentice Hall.
- Zim, H. S. (1962). *Codes and secret writing (abridged edition)*. Scholastic Book Services, fourth printing.

# On Subclasses of Circular Arc Bigraphs

Marina Groshaus<sup>1</sup>, André Luiz Pires Guedes<sup>2</sup>, Fabricio Schiavon Kolberg<sup>2</sup>

<sup>1</sup>Departamento Academico de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR) – Curitiba – PR – Brazil

<sup>2</sup>Departamento de Informática  
Universidade Federal do Paraná (UFPR) – Curitiba – PR – Brazil

marinagroshaus@yahoo.es, alpguedes@gmail.com, fskolberg@inf.ufpr.br

**Abstract.** *Circular arc (CA) bigraphs, a bipartite variation of circular arc graphs, are bipartite graphs such that, for each of its partite sets, there is a one-to-one correspondence between its elements and a family of arcs such that arcs from the opposing families intersect precisely if the corresponding vertices in the graph are neighbors. In this paper, we present results about different subclasses of circular arc bigraphs, including Helly and proper CA bigraphs, as well as circular convex bipartite (CCB) graphs.*

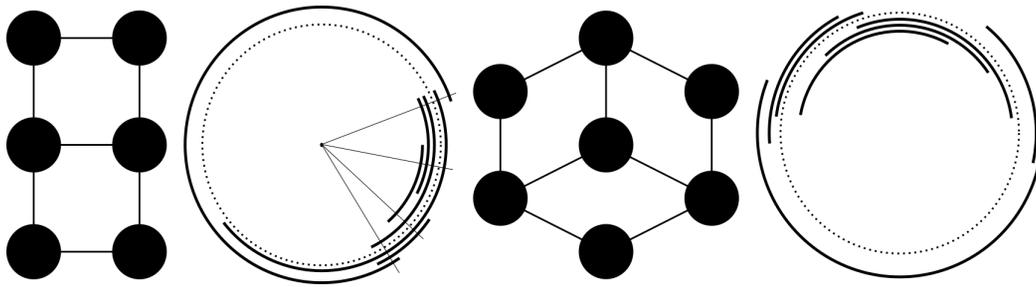
**Keywords:** Graph classes; Characterization; Helly; Circular arc bigraphs; Circular convex bipartite graphs

## 1. Introduction

A *bi-circular-arc model* is a triple  $(C, \mathbb{I}, \mathbb{E})$  such that  $C$  is a circle, and  $\mathbb{I}$  and  $\mathbb{E}$  are families of arcs over  $C$ . The *corresponding graph* of a bi-circular-arc model is constructed by creating a corresponding vertex for every arc in  $\mathbb{I} \cup \mathbb{E}$ , and an edge between the vertices corresponding to two arcs  $I \in \mathbb{I}, E \in \mathbb{E}$  precisely if  $I \cap E \neq \emptyset$  [Basu et al. 2013, Groshaus et al. 2019]. A bipartite graph is a *circular arc (CA) bigraph* if and only if it is the corresponding graph of a bi-circular-arc model. If  $G$  is the corresponding graph of a bi-circular-arc model, we say that  $G$  *admits* said model.

A *proper CA bigraph* is a graph that admits a bi-circular-arc model  $(C, \mathbb{I}, \mathbb{E})$  in which  $\mathbb{I}$  and  $\mathbb{E}$  are *proper families* (i.e. families such that no two elements are properly contained in one another) [Basu et al. 2013]. For a bipartite graph  $G = (V, W, E)$ , a *biclique* is a maximal subset of  $V \cup W$  that induces a bipartite-complete graph. A *Helly CA bigraph* is a graph that admits a bi-circular-arc model  $(C, \mathbb{I}, \mathbb{E})$  such that, for every biclique  $K$  in the graph, there is a point  $p_K \in C$  such that every arc in  $\mathbb{I} \cup \mathbb{E}$  that corresponds to a vertex of  $K$  contains  $p_K$ . We call a model with that property a *Helly bi-circular-arc model*.

Figure 1 contains examples of Helly and Proper CA bigraphs, and their respective bi-circular-arc models. When representing a model  $(C, \mathbb{I}, \mathbb{E})$  graphically, circle  $C$  is denoted as a dotted circle, and the arcs of  $\mathbb{I}$  and  $\mathbb{E}$  are then drawn inside and outside the circle, respectively.



**Figure 1. A Helly CA bigraph alongside its Helly bi-circular-arc model (left), and a proper CA bigraph alongside its proper bi-circular-arc model (right). In the Helly model, the points where the bicliques meet are denoted by a line segment starting at the circle’s center.**

The class of CA bigraphs is a bipartite variation of circular arc graphs [Klee 1969, Lin and Szwarcfiter 2009], and characterizations for the class and its proper, Helly, and unit subclasses exist [Basu et al. 2013, Das and Chakraborty 2015, Groshaus et al. 2019].

A bipartite graph  $G = (V, W, E)$  is a *circular convex bipartite (CCB) graph* if  $V$  (or  $W$ ) admits a circular order such that, for every vertex  $w \in W$  (or  $v \in V$ ), the neighborhood of  $w$  (or  $v$ ) is consecutive in the given order [Liang and Blum 1995]. We call such an order a *circular convex bipartite order*. The class has been extensively studied for its computational properties, since many hard computational problems can be solved in polynomial time within it [Liang and Blum 1995, Liu et al. 2014]. However, the recognition problem for the class, as well as its relationship with CA bigraphs, have not been extensively studied at the time of this writing. In this paper, we show that CCB graphs are indeed CA bigraphs.

We call a CCB graph  $G = (V, W, E)$  *doubly CCB* if both  $V$  and  $W$  admit CCB orders.

In this article, we present results on CCB graphs, doubly CCB graphs, and proper and Helly CA bigraphs, including containment relations and characterizations. Due to space constraints, we limit ourselves to presenting short ideas of the proofs of every claim, rather than their full proofs.

Given a vertex  $v$ , denote by  $a(v)$  the arc corresponding to  $v$  in a bi-circular-arc model, and for any arc  $A$ , denote by  $s(A)$  and  $t(A)$  its counter-clockwise and clockwise endpoints, respectively. All classes presented remain the same regardless of whether arcs are considered closed or open, but in order to simplify our proofs, we consider every arc to be open unless otherwise stated.

## 2. Results

We start by presenting a characterization of CCB graphs which we use in some proofs. For that, we introduce what we call *arc-point models*. An arc-point model is a triple  $(C, \mathbb{A}, P)$  such that  $C$  is a circle,  $\mathbb{A}$  is a family of arcs on  $C$  and  $P$  is a set of points in  $C$ . The corresponding graph of an arc-point model is constructed by creating one vertex for every arc in  $\mathbb{A}$  and point in  $P$ , and for  $A \in \mathbb{A}$  and  $p \in P$ , an edge between the

corresponding vertices of  $A$  and  $p$  exists precisely if  $p \in A$ .

**Lemma 1.** *A graph is a CCB graph if and only if it is the corresponding graph of an arc-point model.*

*Proof Idea.* ( $\Leftarrow$ ) The clockwise order of the points of  $P$  around  $C$  is a circular convex bipartite order. ( $\Rightarrow$ ) For the part that admits a CCB order, distribute points around  $C$  according to the order. For the opposite part, it is easy to build a family of arcs.  $\square$

Applying Lemma 1, we can prove the following theorem.

**Theorem 1.** *Every CCB graph is a CA bigraph.*

*Proof Idea.* By replacing the set  $P$  of points in an arc-point model with a family of  $|P|$  sufficiently small arcs in  $C$ , we obtain a bi-circular-arc model with the same corresponding graph. Therefore, CCB graphs are CA bigraphs.  $\square$

The following is also an important preliminary result. We call vertices with the same open neighborhood *twins*, and the *twin-free version* of a graph the graph that results from removing one vertex from every pair of twins until no twins remain.

**Lemma 2.** *A graph is a CCB (doubly CCB) graph if and only if its twin-free version is.*

In the following theorem, we present a pair of containment relations between CCB graphs and subclasses of CA bigraphs.

**Theorem 2.** *If  $G$  is a Helly or proper CA bigraph, then  $G$  is a doubly CCB graph.*

*Proof Idea.* For the Helly case, consider a twin-free graph  $G$  that admits a Helly model  $(C, \mathbb{I}, \mathbb{E})$ . For every vertex  $v$  in the part represented by  $\mathbb{E}$ , there is a biclique  $K_v$  containing its closed neighborhood, and since the model is Helly, there is a point  $P_{K_v} \in C$  contained in every arc corresponding to elements of  $K_v$ . Therefore, we can build an arc-point model  $(C, \mathbb{I}, P)$  where  $P$  contains the points  $P_{K_v}$  for every  $v$  in the part represented by  $\mathbb{E}$ .

For the proper case, consider a model  $(C, \mathbb{I}, \mathbb{E})$ , with  $\mathbb{I}$  and  $\mathbb{E}$  proper, corresponding to twin-free graph  $G$ . Consider the circular order in which the counter-clockwise endpoints of  $\mathbb{I}$  are encountered in a clockwise traversal of the circle. That order is a CCB order of the part represented by  $\mathbb{I}$ .

Both classes are doubly CCB, since the proofs can be applied equally to both partite sets.  $\square$

The converse of Theorem 2 is not true, and Figure 2 contains counter-examples for both cases.

The following result presents a characterization of proper CA bigraphs within the class of CCB graphs. In a bipartite graph  $G = (V, W, E)$ , we call a vertex  $v \in V$  ( $w \in W$ ) *bi-universal* if  $N(v) = W$  ( $N(w) = V$ ).

**Lemma 3.** *A bipartite graph  $G = (V, W, E)$  without bi-universal vertices is a proper CA bigraph if and only if  $V$  admits a CCB order such that, for every pair of elements  $w_1, w_2 \in W$  satisfying  $N(w_1) \subset N(w_2)$ , the intervals of the CCB order corresponding to the vertices in  $N(w_1)$  and  $N(w_2)$  either begin or end in the same vertex.*

*Proof Idea.* Theorem 3.6 from [Basu et al. 2013] states that a bipartite graph is a proper CA bigraph if and only if it admits a biadjacency matrix with a *monotone circular arrangement* (MCA). The order of the columns on a matrix with an MCA is a CCB order with the properties stated in the lemma.

Conversely, if a graph admits a CCB order with the properties stated in the lemma, it is easy to show that there exists a biadjacency matrix with an MCA such that the order of columns is given by the CCB order.  $\square$

We close our results section presenting a containment relation between a subclass of Helly CA bigraphs and proper CA bigraphs. A bipartite graph is *bichordal* if it does not contain an induced cycle of order greater than 4. The following characterization, presented without proof due to space constraints, is applied in the containment relation.

**Lemma 4.** Consider graph  $G$ , with  $V(G)$  being partitioned in the following subsets, for  $k \geq 6, n_1, \dots, n_k \geq 0$ :

- $C = \{c_1, c_2, \dots, c_k\}$ .
- $V = \{v_1, v_2, \dots, v_k\}$ .
- $W_i = \{w_{i,1}, \dots, w_{i,n_i}\}$  for all  $1 \leq i \leq k$ .
- $X_i = \{x_{i,1}, \dots, x_{i,n_i}\}$  for all  $1 \leq i \leq k$ .

Let the neighborhoods of the vertices in  $V(G)$  be the following, for all  $1 \leq i \leq k$ . Consider cyclic summation (e.g.  $1 - 1 = k, k + 1 = 1$ ) for vertex and set indices:

- $N(c_i) = \{c_{i-1}, c_{i+1}, v_i\} \cup W_i \cup X_{i-1}$ .
- $N(v_i) = \{c_i\}$ .
- $N(w_{i,j}) = \{c_i\} \cup \{x_{i,l} \in X_i | l \leq j\}$ , for all  $1 \leq j \leq n_i$ .
- $N(x_{i,j}) = \{c_{i+1}\} \cup \{w_{i,l} \in W_i | l \geq j\}$ , for all  $1 \leq j \leq n_i$ .

A bipartite graph that is not bichordal is a Helly CA bigraph if and only if its twin-free version is an induced subgraph of  $G$ , for certain values of  $k, n_1, \dots, n_k$ .

**Theorem 3.** If  $G$  is a Helly CA bigraph that is not bichordal, then  $G$  is a proper CA bigraph.

*Proof Idea.* It suffices to prove that graph  $G$  from Lemma 4 is a proper CA bigraph.

Note that  $G$  has no bi-universal vertices, so we can apply Lemma 3 to prove that  $G$  is a proper CA bigraph.

The partite sets of  $G$  are the following:

- $X = C_1 \cup V_2 \cup \bigcup_{i=2, i+2}^k W_i \cup \bigcup_{i=1, i+2}^{k-1} X_i$ .
- $Y = C_2 \cup V_1 \cup \bigcup_{i=2, i+2}^k X_i \cup \bigcup_{i=1, i+2}^{k-1} W_i$ .

Consider, then, the following circular ordering of  $X$ :

$$(c_1, x_{1,1}, \dots, x_{1,n_1}, v_2, w_{2,1}, \dots, w_{2,n_2}, c_3, \dots, c_{k-1}, x_{k-1,1}, \dots, x_{k-1,n_{k-1}}, v_k, w_{k,1}, \dots, w_{k,n_k})$$

The above order, in simple terms, is built as follows: for every  $i$  (starting at  $i = 1$ ), add vertex  $c_i$ , followed by the elements of  $X_i$  in ascending order, followed by  $v_{i+1}$ , followed by  $W_{i+1}$  in ascending order, and then the loop restarts with  $i + 2$  until  $i > k$ .

Note that the presented order is a CCB order. Furthermore, the only proper containments between neighborhoods of  $Y$  are the following, for  $i \leq k, l_1, l_2 \leq n_i$ :

- Between pairs of the form  $x_{i,l_1}, x_{i,l_2}$ .
- Between pairs of the form  $w_{i,l_1}, w_{i,l_2}$ .
- Between vertices of  $W_i$  and  $c_{i+1}$ .
- Between vertices of  $X_i$  and  $c_i$ .
- Between  $v_i$  and any vertex from  $W_i, X_{i-1}$ , or  $\{c_{i-1}, c_{i+1}\}$ .

In the given CCB order of  $X$ , all pairs listed above are such that the intervals containing their neighborhoods have an endpoint in common. Therefore,  $G$  is a proper CA bigraph.

Note that, if a graph is a proper CA bigraph, the addition of twins to it makes it remain a proper CA bigraph. To add a twin arc to a family without creating proper containments, just copy the original arc and shift the endpoints a significantly small amount. Therefore, we conclude that every Helly CA bigraph that is not bichordal is a proper CA bigraph.  $\square$

Figure 2 is a Venn diagram of the classes mentioned in the paper. The regions marked as  $CAB$ ,  $CCB$ , and  $D-CCB$  represent, respectively, CA bigraphs, CCB graphs and doubly CCB graphs, while the regions marked as  $P$ ,  $H$  and  $NBH$  represent proper and Helly CA bigraphs, and Helly CA bigraphs that are not bichordal, respectively.

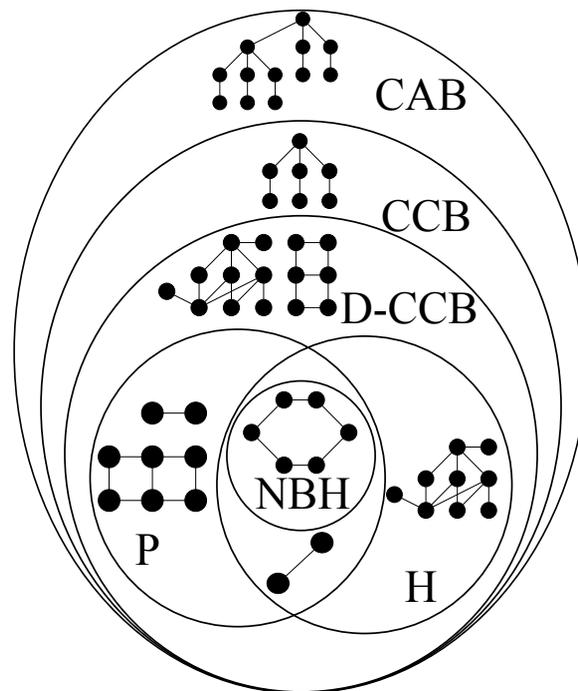
### 3. Conclusion

We presented a characterization of circular convex bipartite graphs using the concept of arc-point models. We showed that doubly CCB graphs, a proper subclass of circular convex bipartite graphs, are a proper superclass of both Helly and proper circular arc bigraphs. We also showed a result that proves that non-bichordal Helly circular arc bigraphs are a proper subclass of proper circular arc bigraphs, and presented a characterization of proper CA bigraphs without bi-universal vertices within the class of CCB graphs.

Future work includes searching for recognition algorithms for circular convex bipartite graphs, and studying other potential characterizations of proper and Helly CA bigraphs with relation to the class of CCB graphs.

### References

- Basu, A., Das, S., Ghosh, S., and Sen, M. (2013). Circular-arc bigraphs and its subclasses. *Journal of Graph Theory*, 73(4):361–376.
- Das, A. and Chakraborty, R. (2015). New characterizations of proper interval bigraphs and proper circular arc bigraphs. In Ganguly, S. and Krishnamurti, R., editors, *Algorithms and Discrete Applied Mathematics*, volume 8959 of *Lecture Notes in Computer Science*, pages 117–125. Springer International Publishing.



**Figure 2. A Venn diagram of the classes mentioned in the paper, with an example graph in each region.**

Groshaus, M., Guedes, A. L. P., and Kolberg, F. S. (2019). Subclasses of circular-arc bigraphs: Helly, normal and proper. In *Proc. X Latin & American Algorithms, Graphs, and Optimization Symposium (LAGOS '19)*. Accepted.

Klee, V. (1969). Research problems: What are the intersection graphs of arcs in a circle? *American Mathematics Monthly*, 76(7):810–813.

Liang, Y. and Blum, N. (1995). Circular convex bipartite graphs: Maximum matching and hamiltonian circuits. *Information Processing Letters*, 56(4):215 – 219.

Lin, M. C. and Szwarcfiter, J. L. (2009). Characterizations and recognition of circular-arc graphs and subclasses: A survey. *Discrete Mathematics*, 309(18):5618–5635. *Combinatorics 2006, A Meeting in Celebration of Pavol Hell's 60th Birthday (May 1–5, 2006)*.

Liu, T., Lu, M., Lu, Z., and Xu, K. (2014). Circular convex bipartite graphs: Feedback vertex sets. *Theoretical Computer Science*, 556:55 – 62. *Combinatorial Optimization and Applications*.

# Uma Revisão sobre a Relação de BQP com outras Classes de Complexidade Computacional

Henrique Hepp<sup>1</sup>, Murilo V. G. da Silva<sup>1</sup>, Leandro M. Zatesko<sup>2</sup>

<sup>1</sup>Departamento de Informática, Universidade Federal do Paraná

<sup>2</sup>Departamento de Informática, Universidade Tecnológica Federal do Paraná

{hhepp,murilo}@inf.ufpr.br, zatesko@utfpr.edu.br

**Abstract.** *The class of problems for which there are efficient quantum algorithms, denoted by BQP (Bounded-error Quantum Polynomial), is still not well understood. A common approach in theoretical computer science for understanding the limits of a class of complexity is to clarify its relations with other classes. This article presents a brief survey of the relations of BQP with the most known classical and quantum complexity classes.*

**Resumo.** *A classe de problemas para os quais existem algoritmos quânticos eficientes, denotada por BQP (Bounded-error Quantum Polynomial), ainda é pouco compreendida. Uma abordagem comum em teoria da computação para a compreensão dos limites de uma classe de complexidade é esclarecer relações desta com outras classes. Este artigo apresenta um breve survey das relações de BQP com as classes de complexidade computacional clássicas e quânticas mais conhecidas.*

**Keywords:** Quantum Computation; Computational Complexity; BQP; BPP

**Palavras-chave:** Computação Quântica; Complexidade Computacional; BQP; BPP

## 1. Introdução

Os modelos clássico e quântico de computação definem duas classes diferentes de problemas computacionais que podem ser resolvidos em tempo polinomial. Em computação clássica, denotamos tal classe por P e em computação quântica por BQP, ou *Bounded-error Quantum Polynomial*.

Em computação clássica, uma outra classe importante é a classe dos problemas decididos em tempo polinomial por algoritmos aleatorizados, denominada BPP, *Bounded-error Probabilistic Polynomial time*, que claramente generaliza P. A classe BPP está contida na classe BQP, por outro lado, acredita-se que a classe BQP não seja igual à classe BPP, pois existem algoritmos polinomiais quânticos para problemas que conjecturam-se não estar em BPP, como por exemplo, o algoritmo de Shor, que resolve o problema de fatoração em números primos.

Classes de complexidade que contêm BQP têm sido alvo recente de investigação em diversos trabalhos [Aaronson 2005, Aaronson et al. 2016,

Morimae and Nishimura 2017]. Pois os estudos de classes mais poderosas que BQP ajudam a compreender a estrutura matemática de diversos problemas computacionais, em particular, problemas na fronteira entre o que pode e o que não pode ser revolvido por algoritmos quânticos de maneira eficiente.

Nessa revisão apresentamos as classes de complexidade mais relevantes no contexto de computação quântica e suas relações com a classe BQP. O artigo está organizado da seguinte maneira. A Seção 2 apresenta uma introdução às classes de complexidade quântica. A Seção 3 apresenta as principais classes de complexidade relacionadas com BQP e a Seção 4 conclui o artigo.

## 2. Classes de Complexidade Quânticas

Nesse artigo assumimos que o leitor tenha familiaridade com classes de complexidade computacional em geral. Para maiores detalhes indicamos o livro [Arora and Barak 2009]. Nessa seção apresentamos apenas os preliminares matemáticos para definição das classes de complexidade quânticas. Para maiores detalhes sobre o modelo de computação quântica indicamos o livro [Nielsen and Chuang 2011].

O estado de um sistema quântico é um vetor unitário de números complexos chamado de *vetor de estado* representado por  $|\psi\rangle$ . De modo geral, um vetor de estado  $|\psi\rangle$  em um espaço com  $N$  dimensões  $\mathbb{C}^N$  pode ser descrito pela combinação linear de  $N$  estados linearmente independentes:  $|\psi\rangle = \alpha_1 |1\rangle + \alpha_2 |2\rangle + \alpha_3 |3\rangle + \dots + \alpha_N |N\rangle$ , onde  $\alpha_1, \dots, \alpha_N$  são números complexos. Como  $|\psi\rangle$  é um vetor unitário, temos que  $\sum_i |\alpha_i|^2 = 1$ .

Para a computação quântica usa-se o vetor de estado mais simples, que é o *qubit*, uma generalização do bit. Enquanto o bit pode ser representado por  $\{0, 1\}$ , o qubit é um estado quântico  $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$ , onde  $\alpha, \beta \in \mathbb{C}$  e  $|\alpha|^2 + |\beta|^2 = 1$ . De forma mais geral, o estado quântico que descreve  $n$  qubits é produto tensorial entre os  $n$  qubits  $|i\rangle$ , dado por  $|\Psi\rangle = \sum_{i=0}^{2^n-1} \alpha_i |i\rangle$  onde  $\alpha_0, \dots, \alpha_{2^n-1} \in \mathbb{C}$  e  $\sum_{i=0}^{2^n-1} |\alpha_i|^2 = 1$ .

*Transformações Unitárias* são transformações dos estados quânticos de modo que eles continuem sendo vetores unitários. Em outras palavras, seja  $U$  uma transformação unitária,  $|\Psi'\rangle = U |\Psi\rangle$ . Uma *porta quântica* que age sobre qubits é uma transformação unitária agindo sobre o vetor correspondente aos qubits em questão.

Dado o estado  $|\Psi\rangle = \sum_{i=0}^{2^n-1} \alpha_i |i\rangle$ , podemos fazer uma *medição quântica* de  $|\Psi\rangle$  com relação à base  $\{|0\rangle, |1\rangle, \dots, |2^n-1\rangle\}$ . Ao fazermos a medição obtemos um dos estados  $|i\rangle$  com probabilidade  $|\alpha_i|^2$  e perdemos todas as informações do estado original. Em outras palavras, ao se medir  $|\Psi\rangle$ , o estado *colapsa* em  $|i\rangle$  com probabilidade  $|\alpha_i|^2$ .

Um *circuito quântico* é um grafo direcionado acíclico onde cada vértice pode ser uma porta quântica ou uma medição agindo sobre qubits. O tamanho do circuito é o número de vértices do grafo. Um *algoritmo quântico* é uma família infinita de circuitos quânticos  $\{C_n\}$ ,  $n \in \mathbb{N}^*$ , sendo que deve ser possível obter a descrição de  $C_n$  por meio de uma máquina de Turing de tempo polinomial em  $n$ .

A classe BQP, ou *Bounded-error Quantum Polynomial Time*, é a classe de problemas de decisão resolvidos por uma família uniforme de circuitos quânticos com tamanho polinomial. Isto é, uma instância positiva é aceita com probabilidade pelo menos maior

ou igual a  $2/3$ , uma instância negativa é aceita com probabilidade menor que  $1/3$ . Chamamos os problemas dessa classe por problemas que admitem um algoritmo BQP.

A classe QMA, ou *Quantum Merlin-Arthur*, é a classe de problemas cujas instâncias verdadeiras admitem certificados quânticos (i.e., estados quânticos) que podem ser verificadas em tempo polinomial quântico. Ou seja, QMA se relaciona com BQP assim como NP se relaciona com P (assim como MA com relação à BPP, quando permitimos aleatoriedade).

A classe CQP, ou *Collapse-free Quantum Polynomial Time*, é a classe de problemas resolvidos por um algoritmo BQP contendo medições que não provoquem o colapso dos estados. A classe naCQP, “non-adaptive Collapse-free Quantum Polynomial Time”, é a classe CQP com a restrição de que as operações quânticas independem dos resultados das medições que não colapsam. Essas duas classes não são realizáveis fisicamente, pois admitiriam a possibilidade de sinais mais rápidos do que a velocidade da luz, mas fornecem um ferramental útil no estudo da complexidade computacional [Aaronson et al. 2016].

A classe clássica PP, ou *Probabilistic Polynomial-Time*, é equivalente à classe quântica PostBQP, ou *Postselected Bounded-Error Quantum Polynomial-Time*. PostBQP é a classe de problemas resolvidos por um algoritmo BQP com a funcionalidade extra de se poder *assumir* que a medição um determinado qubit sempre será igual a  $|1\rangle$  quando essa probabilidade for diferente de zero [Aaronson 2005].

Em complexidade também é muito comum usar-se classes com um oráculo. Um oráculo é uma espécie de “caixa preta” que retorna uma resposta para qualquer pergunta. Se  $\mathcal{A}$  é uma classe de complexidade,  $\mathcal{A}^B$  denota a classe dos problemas resolvidos por algoritmos em  $\mathcal{A}$  com oráculo para problemas completos de  $B$ .

### 3. Relações de BQP com outras Classes de Complexidade

Nas Seções 3.1 e 3.2 são apresentadas respectivamente as relações conhecidas e conjecturadas de BQP com outras classes de complexidade conforme vistas na Figura 1.

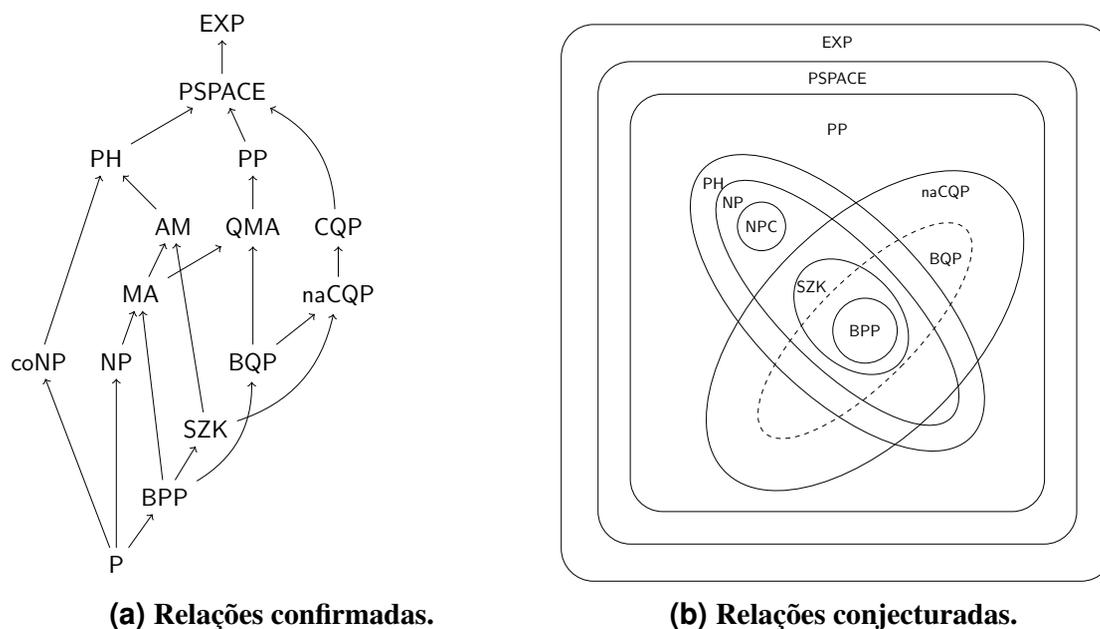
#### 3.1. Relações Conhecidas

**Teorema 1.** [Gill 1977, Bernstein and Vazirani 1997]  $P \subseteq BPP \subseteq BQP$

BPP contém P, pois um algoritmo BPP possui a funcionalidade extra de produzir bits aleatórios. BQP contém BPP, pois além de simular qualquer circuito clássico também tem a habilidade de gerar bits aleatórios usando uma porta quântica chamada de Porta de Hadamard [Nielsen and Chuang 2011].

**Teorema 2.** [Watrous 2000, Marriott and Watrous 2005]  $BQP \subseteq QMA \subseteq PP$

A prova de que BQP está contida em QMA é análoga à prova de que P está contida em NP, generalizando bits para qubits e verificadores polinomiais para verificadores BQP. A classe QMA, por sua vez, está contida em PP. Esse resultado com demonstração bastante técnica foi obtido por Marriott e Watrous. É possível obter-se uma demonstração alternativa usando-se o fato de que  $PP = PostBQP$  e que QMA é um caso particular de PostBQP [Aaronson 2005].



(a) Relações confirmadas.

(b) Relações conjecturadas.

Figura 1. Relações entre algumas das principais classes de complexidade.

**Teorema 3.** [Aaronson et al. 2016]  $BQP \subseteq naCQP \subseteq CQP \subseteq PSPACE$

As duas primeiras inclusões procedem das definições das classes. Para a inclusão de CQP em PSPACE, mostra-se que uma sequência de “queries” para um oráculo que resolva problemas de  $BPP^{\#P}$  pode ser usado para simular um circuito CQP. A inclusão segue do fato que  $BPP^{\#P} = BPP^{PP} \subseteq PSPACE$ .

**Teorema 4.** [Goldwasser et al. 1989, Aaronson et al. 2016]  $BPP \subseteq SZK \subseteq naCQP$

A classe SZK contém BPP, pois na definição de SZK usa-se um verificador BPP. Para a prova da inclusão de SZK em naCQP usa-se a redução do problema de determinar a diferença estatística de distribuições, completo para SZK.

### 3.2. Relações Conjecturadas

**Conjectura 1.** [Aaronson et al. 2016]  $NP \not\subseteq naCQP$

No modelo de computação “caixa-preta”, provou-se que para encontrar um elemento dentre  $2^n = N$  itens, um algoritmo naCQP necessita tempo  $\Omega(N^{1/4})$ . Isso significa que um algoritmo naCQP não conseguiria resolver por “força-bruta” o problema 3SAT em tempo polinomial, pois no modelo “caixa-preta”, a estrutura dos problemas é abstraída. Por isso, acredita-se que a naCQP não contém NP, o que tornaria naCQP apenas um pouco maior que BQP.

**Conjectura 2.** [Aaronson et al. 2016]  $naCQP \subset PP$

Considerando-se que (provavelmente) naCQP seja uma classe pouco maior que BQP, suspeita-se que naCQP esteja contido em PP. Atualmente o limite superior conhecido de naCQP é  $BPP^{PP}$ .

**Conjectura 3.**  $SZK \not\subseteq BQP$

Suspeita-se que BQP não contém SZK pois foi encontrado um oráculo  $A$  tal que

$SZK^A \not\subseteq BQP^A$  [Aaronson 2002]. Ademais, até então não se encontrou um algoritmo quântico para vários problemas SZK, como o de Isomorfismo de Grafos.

**Conjectura 4.**  $SZK \subsetneq NP$

Acredita-se que a SZK seja NP-intermediária, pois a hierarquia polinomial colapsa caso vários problemas importantes em SZK sejam provados como NP-completos [Aiello and Hastad 1991].

**Conjectura 5.** [Bernstein and Vazirani 1997]  $BPP \subsetneq BQP$

Acredita-se que a BQP não seja igual à BPP, pois existem algoritmos polinômiais quânticos que conjectura-se não estar em BPP. Tal é o caso do Algoritmo de Shor [Shor 1999] que resolve o problema de fatoração em números primos.

**Conjectura 6.**  $NP \neq BQP$ . Em particular  $\exists L \in NP$  tal que  $L \notin BQP$  e  $\exists L \in BQP$  tal que  $L \notin NP$ .

Acredita-se que BQP não contém NP pelas seguintes razões. Primeiro, até então não se encontrou nenhum algoritmo quântico que resolva um problema NP-completo em tempo polinomial. Segundo, no modelo de computação “caixa-preta”, provou-se que para encontrar um elemento dentre  $2^n = N$  itens, um algoritmo BQP necessita tempo  $\Omega(N^{1/2})$  [Bennett et al. 1997]. Isso significa que um algoritmo BQP não conseguiria resolver por “força-bruta” o problema 3SAT em tempo polinomial.

Também se acredita que NP não contém BQP. Primeiro, porque os problemas BQP-completos conhecidos, até onde se sabe, não estão em NP. Segundo, por que existe um oráculo  $A$  tal que  $BQP^A \not\subseteq NP^A$  [Watrous 2000].

**Conjectura 7.**  $PH \neq BQP$ . Em particular  $\exists L \in PH$  tal que  $L \notin BQP$  e  $\exists L \in BQP$  tal que  $L \notin PH$ .

Essa conjectura é uma generalização da Conjectura 6, pois PH é uma classe que generaliza as classes NP e coNP. Conjectura-se que  $BQP \not\subseteq PH$  pois encontrou-se recentemente um oráculo  $A$  tal que  $BQP^A \not\subseteq BH^A$  [Raz and Tal 2018].

**Conjectura 8.**  $PH \subseteq PP$

De acordo com o Teorema de Toda,  $PH \subseteq BP \cdot PP$  [Toda 1991, Toda and Ogiwara 1992], sendo que BP é um operador que adiciona uma aleatorização do mesmo modo como é adicionada de P para BPP, ou seja,  $BP \cdot P = BPP$ . Assumindo hipóteses de desaleatorização, pode-se mostrar que  $PH \subseteq BP \cdot PP = PP$  [Hitchcock and Pavan 2004].

## 4. Conclusão

Neste artigo foram vistas algumas das relações comprovadas e conjecturadas entre BQP e outras classes de complexidade, juntamente com uma breve ideia das prova ou das razões para as conjecturas feitas. De modo geral, acredita-se que BQP contém restritamente BPP, não contém NP e resolve problemas fora de PH.

## Referências

Aaronson, S. (2002). Quantum lower bound for the collision problem. In *Proceedings of the thirty-fourth annual ACM symposium on Theory of computing*, pages 635–642. ACM.

- Aaronson, S. (2005). Quantum computing, postselection, and probabilistic polynomial-time. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 461(2063):3473–3482.
- Aaronson, S., Bouland, A., Fitzsimons, J., and Lee, M. (2016). The space just above BQP. In *Proceedings of the 2016 ACM Conference on Innovations in Theoretical Computer Science*, pages 271–280. ACM.
- Aiello, W. and Hastad, J. (1991). Statistical zero-knowledge languages can be recognized in two rounds. *Journal of Computer and System Sciences*, 42(3):327–345.
- Arora, S. and Barak, B. (2009). *Computational Complexity: a Modern Approach*. Cambridge University Press.
- Bennett, C. H., Bernstein, E., Brassard, G., and Vazirani, U. (1997). Strengths and weaknesses of quantum computing. *SIAM journal on Computing*, 26(5):1510–1523.
- Bernstein, E. and Vazirani, U. (1997). Quantum complexity theory. *SIAM Journal on computing*, 26(5):1411–1473.
- Gill, J. (1977). Computational complexity of probabilistic turing machines. *SIAM Journal on Computing*, 6(4):675–695.
- Goldwasser, S., Micali, S., and Rackoff, C. (1989). The knowledge complexity of interactive proof systems. *SIAM Journal on computing*, 18(1):186–208.
- Hitchcock, J. M. and Pavan, A. (2004). Hardness hypotheses, derandomization, and circuit complexity. In *International Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 336–347. Springer.
- Marriott, C. and Watrous, J. (2005). Quantum arthur–merlin games. *Computational Complexity*, 14(2):122–152.
- Morimae, T. and Nishimura, H. (2017). Merlinization of complexity classes above BQP. *Quantum Information & Computation*, 17(11-12):959–972.
- Nielsen, M. A. and Chuang, I. L. (2011). *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, New York, NY, USA, 10th edition.
- Raz, R. and Tal, A. (2018). Oracle separation of BQP and PH. In *Electronic Colloquium on Computational Complexity (ECCC)*, volume 25, page 107.
- Shor, P. W. (1999). Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM review*, 41(2):303–332.
- Toda, S. (1991). PP is as hard as the polynomial-time hierarchy. *SIAM Journal on Computing*, 20(5):865–877.
- Toda, S. and Ogiwara, M. (1992). Counting classes are at least as hard as the polynomial-time hierarchy. *SIAM Journal on Computing*, 21(2):316–328.
- Watrous, J. (2000). Succinct quantum proofs for properties of finite groups. In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, pages 537–546. IEEE.

# Um Framework para o Teste Automatizado de Sensores

Darlan Felipe Klotz<sup>1</sup>, Marco Tulio Guerreiro<sup>1</sup>, Marcelo Teixeira<sup>1</sup>, Richardson Ribeiro<sup>1</sup>

<sup>1</sup>Programa de Pós-Graduação em Engenharia Elétrica – Universidade Tecnológica Federal do Paraná (UTFPR) – Pato Branco – PR – Brasil

darlan.klotz@ifc.edu.br, marcotulio.guerreiro@gmail.com,  
marceloteixeira@utfpr.edu.br, prof.richard@gmail.com

**Abstract.** *This paper proposes an automated procedure to test the operation of capacitive soil moisture sensors. The test procedure is modeled using Finite Automata and Language concepts. From the monolithic synthesis, we obtained a controller that was converted to Ladder language and implemented in a Programmable Logic Controller. A SCADA application monitors and communicates with the process via Modbus RTU protocol and RS 485 serial.*

**Resumo.** *Este trabalho propõe um procedimento automatizado para testar a operação de sensores capacitivos de umidade do solo. O procedimento de teste é modelado utilizando Autômatos Finitos e conceitos de Linguagens. A partir da síntese monolítica, foi obtido um controlador que foi convertido em linguagem Ladder e implementado em um Controlador Lógico Programável. Uma aplicação SCADA monitora e se comunica com o processo através do protocolo Modbus RTU e do padrão serial RS 485.*

## 1. Introdução

Procedimentos de testes possuem um papel fundamental na indústria [Tahera, Earl e Eckert 2012]. A partir de rotinas específicas de testes, é possível verificar o funcionamento de um produto, em sua forma final ou parcial. Um exemplo que é de interesse particular desse trabalho, envolve a fabricação de sensores capacitivos para a medição de umidade de solo. Nesse caso, um teste manual de validação apresenta potencial para alterar o resultado do teste em função de erros causados pela ação humana [Caetano 2008].

O presente artigo propõe um método automático para a execução de uma rotina de teste para sensores capacitivos de umidade de solo, visto que, atualmente, observa-se que essa prática é realizada manualmente. O procedimento consiste em inserir o sensor em um tanque com água e medir sua resposta em dois níveis diferentes, a fim de verificar, com maior confiabilidade, o correto funcionamento do sensor. Em cada nível, uma resposta pré-definida é esperada. O sensor é aprovado no teste caso as medidas nos dois níveis de água sejam iguais às medidas esperadas e reprovado caso contrário.

Inicialmente, o processo de teste foi observado como um Sistema a Eventos Discretos (SED) [Cassandras e Lafortune 2009]. Essa abordagem permite modelar a dinâmica do sistema por meio de um Autômato Finito (AF) tal que as sequências de evolução do sistema correspondem à evolução desse modelo em seu espaço de estados. A partir do modelo do sistema exposto como um AF, foram construídas especificações

de comportamento que, em conjunção, viabilizam o cálculo de um controlador para o processo. Esse controlador inclui todas e somente as sequências de operação que são permitidas para o sistema sob controle, ou seja, que levam a um comportamento seguro, robusto, minimamente restritivo e não bloqueante. O processo de cálculo desse controlador é denominado de síntese, e a sua concepção matemática é estruturada pela Teoria de Controle Supervisório (TCS) de SEDs [Wonhan 2002].

O controlador foi convertido em código implementável, nesse artigo em um Controlador Lógico Programável (CLP), usando a linguagem Ladder [Bryan e Bryan 1997]. Para fins de supervisão, foi desenvolvida e implementada uma aplicação SCADA que se comunica com o CLP através de uma rede industrial RS 485 regida pelo protocolo Modbus® RTU.

## 2. Descrição do Processo de testes

O dispositivo, cuja sequência de testes é automatizada nesse trabalho, é um sensor capacitivo que detecta a umidade do solo, como mostra a Figura 1. Suas hastes detectam o contato com a matéria, alterando a resposta do sensor conforme o meio é alterado.



**Figura 1. Sensor capacitivo de umidade de solo a ser testado.**

A rotina de teste dos sensores é assim descrita. Após a montagem do circuito, os sensores são postos em uma caixa ou recipiente, denominada de “testes”. Em seguida, um operador retira um sensor por vez da caixa de testes, conectando-o a um cabo para energizá-lo. Então, o sensor é inserido em um tanque com água, sendo que, inicialmente, 90 % do comprimento das hastes do sensor é submerso.

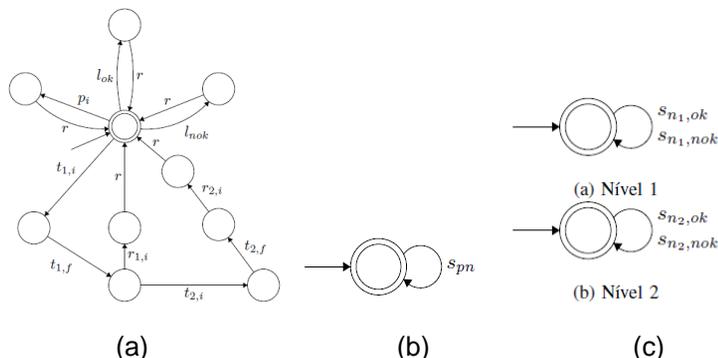
A resposta do sensor é lida por um circuito eletrônico e convertida para valores de tensão, sendo que o circuito final deve apresentar uma tensão de cerca de 3V. Caso o valor de leitura seja diferente de  $3V \pm 0,1 V$ , o sensor é reprovado no teste e descartado em uma caixa denominada de “nok”. Do contrário, i.e., caso o sensor seja aprovado no teste, o próximo passo consiste em elevar o sensor para o segundo nível de medição, de forma que as hastes se encontrem 50% submersas. Nesse caso, a tensão esperada é de  $2.5V \pm 0,1 V$ . Caso a tensão não esteja dentro da faixa aceitável, o sensor é reprovado e, igualmente, depositado em “nok”. Do contrário, o sensor é aprovado no teste e colocado em uma caixa de sensores aprovados, denominada de “ok”.

Para o caso em que uma máquina executa a rotina de testes, um braço robótico pode ser usado para buscar um novo sensor na caixa de testes, executar na ordem correta e segura as rotinas para verificação das medidas de tensão e, após o resultado, depositá-lo no recipiente de sensores aprovados ou reprovados [Rosário 2010].

## 3. Modelagem do processo de testes

Para a modelagem do sistema, foi considerada a inserção de um braço robótico, para a manipulação dos componentes a serem testados, e um conjunto de sensores que

identificam os componentes e auxiliam o braço. O manipulador robótico foi modelado com base em todos os movimentos possíveis para realizar o procedimento de teste e os sensores são modelados por auto laços, como mostra a Figura 2.



**Figura 2. Modelos que compõem a planta do sistema.**

Para iniciar o procedimento de testes, o manipulador realiza o movimento de submergir o sensor até nível 1. O comando  $t_{1,i}$  inicia o movimento do braço robótico para levar o sensor até o nível 1, emitindo um sinal  $t_{1,f}$  para indicar que o manipulador está posicionado com o sensor no nível 1. Então, o manipulador pode retornar, a partir do comando  $r_{1,i}$  que inicia o retorno à posição de descanso caso o sensor seja reprovado, ou avançar ao próximo nível a partir do comando  $t_{2,i}$  caso seja aprovado. Ao atingir o nível 2, o manipulador emite um sinal  $t_{2,f}$  indicando estar em posição de teste. A partir do nível 2, a única possibilidade para o manipulador é voltar para a posição de descanso, que é realizado a partir do comando  $r_{2,i}$ . Quando o manipulador alcançar a posição de descanso, o sinal  $r$  é disparado acusando tal evento. Esse modelo é apresentado na Figura 2a.

Para evitar que o manipulador busque um novo sensor com a caixa “teste” vazia, há um sensor que detecta a presença de peças na caixa por um evento  $s_{pn}$ , visto na Figura 2b. Outros dois sensores, modelados conforme a Figura 2c, indicam se o teste retornou verdadeiro ou falso, ou seja, se as medidas de tensão estão dentro dos valores esperados ou não. No caso de sucesso, os sensores emitem os eventos  $s_{n1,ok}$  e  $s_{n2,ok}$  para o nível 1 e nível dois, respectivamente. Em caso de falha, os sensores emitem os sinais  $s_{n1,nok}$  e  $s_{n2,nok}$ .

### 3.1. Especificações

Para o controle do sistema, quatro especificações foram modeladas. A primeira especificação, apresentada na Figura 3a, define que o manipulador busque uma peça na caixa “teste” se existe de fato um sensor esperando para ser testado.

A segunda especificação, apresentada na Figura 3b, é um complemento da anterior. Nesse caso, a especificação determina uma condição de manipulador ocupado, permitindo a busca de uma nova peça após o teste ser iniciado (evento  $t_{1,i}$ ) e finalizado (evento  $l_{ok}$  ou  $l_{nok}$ ). Dessa forma, essa especificação indica que o braço só pode testar novos sensores caso tenha terminado um procedimento.

Na terceira especificação, apresentada na Figura 3c, determina o procedimento para o primeiro teste. Após posicionado no nível 1 (ocorrência do evento  $t_{1,f}$ ), o manipulador deve esperar o resultado do teste 1 (ocorrência do evento  $s_{n1,ok}$  ou  $s_{n1,nok}$ ). Se o sensor passar no teste, a especificação bloqueia o comando para o manipulador retornar à posição de descanso e permite apenas o movimento para o próximo nível  $t_{2,i}$ . Se o sensor

reprovar no teste, somente o comando para retornar ao descanso é habilitado  $r_{1,i}$ , não permitindo que o teste prossiga. Além disso, após retornar ao descanso, o manipulador deve deixar o sensor na caixa “nok” ( $l_{nok}$ ).

A quarta e última especificação, apresentada na Figura 3d, define o procedimento para o teste no nível 2. Após posicionado neste nível (evento  $t_{2,f}$ ), o manipulador deve aguardar o resultado do teste. Dessa forma, o comando de retorno ao descanso é desabilitado até que a rotina de teste é completada (evento  $s_{n2,ok}$  ou  $s_{n2,nok}$ ). Após o teste, o manipulador deve retornar ao descanso, uma vez que não há outros níveis a serem testados e, com isso, o comando  $r_{2,i}$  para iniciar o retorno é habilitado independente do resultado do teste. No entanto, se o teste falhar ( $s_{n2,nok}$ ), apenas o comando  $l_{nok}$  para deixar o sensor na caixa “nok” é habilitado. Por outro lado, caso o teste passar  $s_{n2,ok}$ , apenas o comando  $l_{ok}$  para deixar o sensor na caixa “ok” é habilitado.

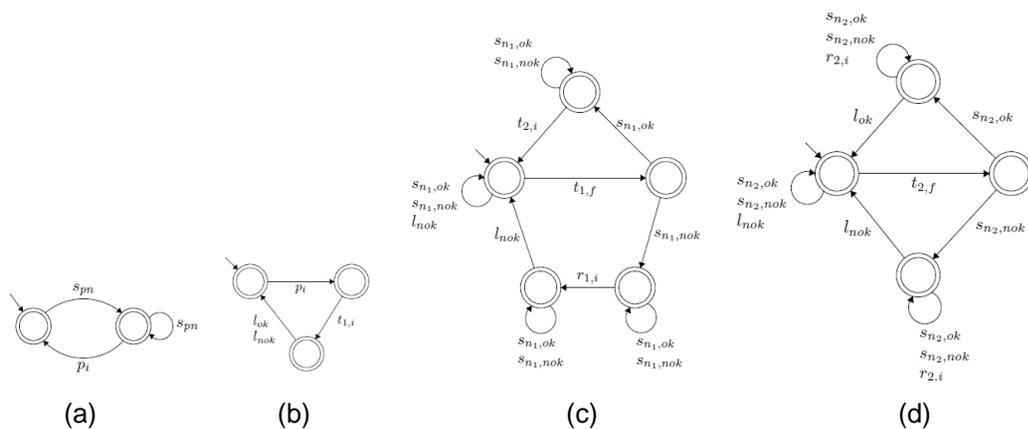


Figura 3. Especificações para o sistema.

### 3.2. Controlabilidade de eventos

Para fins de controle, é importante saber se um evento é de fato controlável, ou seja, se ele é passível de intervenção por via controlador, ou se, do contrário, ele não pode ser diretamente proibido. Esse conceito é um dos fundamentos da TCS de SEDs [Wonhan, 2002]. Para a aplicação em questão, apenas  $\{p_i, l_{ok}, l_{nok}, t_{1,i}, t_{2,i}, r_{1,i}, r_{2,i}\}$  são controláveis.

## 4. Síntese, Implementação e supervisão do sistema de controle

Essa seção apresenta detalhes sobre a síntese do controlador para a aplicação proposta, bem como detalhes sobre a implementação física do sistema.

### 4.1. Composição e Síntese

O modelo do sistema e suas especificações foram construídos na ferramenta *Supremica*® [Akesson, Fabian, Flordal e Malik 2006]. Essa ferramenta incorpora as funcionalidades de modelar, compor e sintetizar o controlador. Para a aplicação proposta, a planta, denotada por  $G$ , é derivada da composição síncrona dos modelos apresentados na Figura 2. Essa composição possui 10 estados. Ainda, o modelo da especificação, denotado por  $E$ , resulta da composição das especificações da Figura 3. O autômato que representa  $E$  possui 120 estados.

A partir de  $G$  e  $E$ , é possível se obter o modelo  $K = G \parallel E$ , representando o comportamento desejado para o sistema  $G$  quando sob influência de  $E$ . O modelo  $K$  pode então ser usado como entrada para a síntese de um controlador, denominado  $S$ , que restringe  $G$  reconhecendo a impossibilidade de desabilitar diretamente eventos não controláveis. O comportamento modelado por  $S$  é, por construção da TCS, considerado minimamente restritivo em relação à planta e aos requisitos, não bloqueante, e alinhado com as especificações do problema.

## 4.2. Implementação e supervisão

A implementação física do sistema de controle foi realizada a partir de um CLP TPW03-30HRA®. A escolha do CLP foi feita visando a comunicação com um sistema supervísório, utilizando o protocolo Modbus RTU [Goldenberg e Wool 2013]. A linguagem *Ladder* foi utilizada para programar o CLP. O controlador  $S$  foi convertido para a linguagem *Ladder*, criando-se blocos que são ativados/desativados conforme o modelo formal obtido na síntese monolítica.

Para supervisionar a rotina de testes automatizada, uma aplicação SCADA® (Supervisory Control and Data Acquisition) foi desenvolvida utilizando o software Elipse E3® para desenvolvimento da interface [Dnaeels e Slater 1999], [Becker, Pardi e Pereira 1999]. A comunicação entre o software supervísório e o CLP é feita a partir do protocolo Modbus® RTU. Na camada física, o CLP utiliza o protocolo serial RS 485. Dessa forma, há um conversor RS485/USB para a conexão serial entre o CLP e o computador utilizado na supervisão. A Figura 4 ilustra o procedimento de supervisão do sistema.

Essa interface permite, além da monitoração on-line, a contagem de sensores aprovados, descartados e totais. Também é possível visualizar o *layout* da planta fabril, com o manipulador robótico, tanque aquático de teste, esteiras de entrada e saída de peças (sensores a serem testados). Por fim, também é possível intervir via SCADA sobre o processo físico, uma vez que ele implementa um formato de comunicação dual.

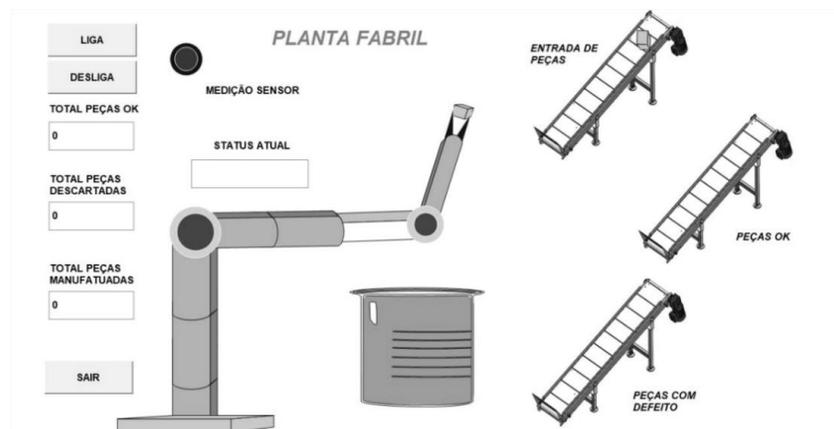


Figura 4. Interface de Controle Supervísório do sistema.

## 4.3. Práticas de testes com o framework de controle e supervisão

Foram realizados ensaios para ilustrar o comportamento do sistema sob controle do CLP. Usando botões para simular os sensores de entrada e observando as saídas do CLP, foi possível observar as ações permitidas e bloqueadas do manipulador através de possíveis sequências de eventos. A primeira sequência testada foi  $\{p_i, r\}$  resultando no evento  $p_i$

bloqueado pois necessita da antecedência do evento  $s_{pn}$  indicando presença de nova peça na caixa “teste”. A segunda sequência testada foi  $\{s_{pn}, p_i, r, s_{pn}, p_i\}$  em que o evento  $p_i$  foi novamente desabilitado pelo controlador após a primeira transição desta cadeia, restringindo a busca por uma nova peça antes da atual ser processada. Outras cadeias de eventos foram aplicadas (como o teste em cada nível do tanque) e nenhum comportamento excedeu às condições esperadas sob controle.

## 5. Conclusão

A aplicação desse framework traz benefícios imediatos, como redução significativa no tempo do teste, redução de erros na validação, monitoramento on-line do sistema, maior confiabilidade nos resultados, dinamismo na confecção de relatórios com número de sensores aprovados e rejeitados, entre outros.

Trabalhos futuros pretendem implementar a comunicação do sistema via Modbus® TCP, na qual possibilitará o armazenamento de dados em rede e contribuirá com a tendência da indústria 4.0.

## Referências

- K. Tahera, C. Earl e C. Eckert, “The role of testing in the engineering product development process”, Proceedings of TMCE 2012, Maio, 2012.
- K. Karhu, T. Repo, O. Taipale e K. Smolander, ”Empirical Observations on Software Testing Automation,”2009 International Conference on Software Testing Verification and Validation, Denver, CO, 2009, pp. 201- 209.
- C.G. Cassandras e S. Lafortune. “Introduction to Discrete Event Systems”, 2 ed, 2009.
- W. Wonhan. “Notes on Discrete Event Systems”, Universidade de Toronto, 2002.
- L.A. Bryan e E.A. Bryan. “Programmable Controllers: Theory and Implementation”. Industrial Text Company, 2 ed, 1997.
- K. Akesson, M. Fabian, H. Flordal, e R. Malik, “Supremica—na integrated environment for verification, synthesis and simulation of discrete event systems”, 8th Int. Workshop on Discrete Event Systems, 384–385, 2006.
- N. Goldenberg e A. Wool, “Accurate modeling of Modbus/TCP for intrusion detection in SCADA systems”. Journal of Critical Infrastructure Protection, pp. 63-75, 2013.
- A. Dnaeels e W. Slater, “What is SCADA?”. International Conference on Accelerator and Large Experimental Physics Control Systems, It’alia, 1999.
- L. Becker, W. Pardi, C. Pereira, “Integrated Object- Oriented Environment for the Design of Supervisory Software for Real-Time Industrial Automation Systems”. Workshop on Object-Oriented Real-Time Dependable Systems, 1999.
- C. Caetano, "Melhores Práticas na Automação de Testes." Revista Engenharia de Software Magazine, 5ª edição, p42-47, 2008.
- J.M. Rosário, “Robótica Industrial I Modelagem, Utilização e Programação”, Vol. 1, Editora Baraúna, 2010.

# Estimando Centralidade de Percolação utilizando Amostragem e Teoria da Dimensão Vapnik-Chervonenkis

Alane Marie de Lima<sup>1</sup>, Giovanne Marcelo dos Santos<sup>2</sup>, André Luis Vignatti<sup>1</sup>,  
Murilo Vicente Gonçalves da Silva<sup>1</sup>

<sup>1</sup>Departamento de Informática – Universidade Federal do Paraná (UFPR)  
Caixa Postal 19.031 – 81531-980 – Curitiba – PR – Brasil

<sup>2</sup>Instituto de Matemática e Estatística – Universidade de São Paulo (USP)

{amlima, vignatti, murilo}@inf.ufpr.br, gsantos@ime.usp.br

**Resumo.** *Medidas de centralidade em redes quantificam a importância relativa de seus nós e conexões, e podem variar de acordo com o contexto em que se aplicam. A centralidade de percolação determina a importância de um nó em aplicações em que há um processo de infestação na rede. O melhor algoritmo exato conhecido para computá-la de maneira exata em uma rede com  $n$  nós e  $m$  conexões executa em tempo  $\mathcal{O}(n^3)$ . Neste trabalho apresentamos um algoritmo aleatorizado para estimar a centralidade de percolação de um dado nó da rede. O algoritmo proposto utiliza resultados da Teoria da Dimensão VC e Teorema da  $\epsilon$ -amostra, e executa em tempo  $\mathcal{O}(\max(n^2, (n+m)\frac{c}{\epsilon^2} \ln \frac{1}{\delta}))$  para grafos sem peso e  $\mathcal{O}(\max(n^2, (m \log n)\frac{c}{\epsilon^2} \ln \frac{1}{\delta}))$  para grafos com peso, apresentando um erro de no máximo  $\epsilon$  com probabilidade  $1 - \delta$ , para constante  $c > 0$ .*

**Abstract.** *In network analysis, a variety of centrality measures are used to quantify the relative importance of entities and connections of a system. Percolation centrality refers to the importance of a node in a network going through a contagious process. The fastest algorithm for computing this measure in a network of  $n$  nodes and  $m$  edges runs in  $\mathcal{O}(n^3)$ . In this work, we propose a randomized algorithm to approximate the percolation centrality of a given vertex in a network. Our algorithm relies on VC-Dimension Theory and  $\epsilon$ -samples, and have running time  $\mathcal{O}(\max(n^2, (n+m)\frac{c}{\epsilon^2} \ln \frac{1}{\delta}))$  for unweighted graphs and  $\mathcal{O}(\max(n^2, (m \log n)\frac{c}{\epsilon^2} \ln \frac{1}{\delta}))$  for weighted graphs, within a factor of  $\epsilon$  for the error and with probability  $1 - \delta$ , for a constant  $c > 0$ .*

**Keywords:** Percolation Centrality, Approximation Algorithm, VC-Dimension

**Palavras-chave:** Centralidade de Percolação, Algoritmo de Aproximação, Dimensão VC

## 1. Introdução

No estudo de redes complexas é comum quantificar a centralidade de nós e de conexões de uma rede a fim de determinar a importância relativa que estas entidades têm no sistema. Em particular, a centralidade de um nó pode ser definida em termos de características locais, como *grau*, ou globais, como *intermediação* ou *percolação*. O foco deste trabalho

é a medida de *centralidade de percolação*, proposta por [Piraveenan et al. 2013], que mede a importância de um nó no contexto em que há um processo de infestação da rede (e.g., transmissão de doenças ou divulgação de notícias falsas).

O estudo do fenômeno de percolação em um sistema foi introduzido por [Broadbent e Hammersley 1957], que atribuíram a aleatoriedade ao processo de passagem de um dado fluido, ou seja, cada parte do “meio” onde passa o fluido possui uma probabilidade de transmiti-lo ou não (estado de percolação). No caso de redes, a centralidade de percolação de um nó generaliza a centralidade de intermediação do mesmo. Enquanto esta última se refere a quantidade de caminhos mínimos que passam pelo nó, a centralidade de percolação, além desta quantidade, leva em consideração o estado de percolação dos nós da rede.

Os melhores algoritmos para o cálculo exato da centralidade de intermediação dos nós da rede dependem da computação de todos os seus caminhos mínimos [Riondato e Kornaropoulos 2016] e, conseqüentemente, o mesmo vale para o cálculo da centralidade de percolação. Até onde os autores sabem, isso se aplica mesmo ao caso de se calcular a centralidade de percolação de um único nó da rede. O melhor algoritmo conhecido para o problema possui complexidade de tempo  $\mathcal{O}(n^3)$ , e pode ser reduzida para  $\mathcal{O}(nm)$  por meio de uma adaptação do algoritmo proposto por [Brandes 2001] em uma versão mais simples do problema que não é a que lidamos aqui. Segundo [Riondato e Kornaropoulos 2016], em redes de larga escala, este algoritmo é custoso e, além disso, resultados aproximados respeitando parâmetros de qualidade e confiança podem ser suficientes na prática. Em nosso trabalho, propomos um algoritmo para *estimar* a centralidade de percolação de um dado vértice, satisfazendo parâmetros de qualidade e confiança arbitrários fornecidos juntos com a entrada. Este trabalho se baseia na abordagem de [Riondato e Kornaropoulos 2016] para estimar a centralidade de intermediação usando resultados da teoria da Dimensão Vapnik-Chervonenkis (VC) e  $\epsilon$ -amostra. O algoritmo proposto executa em tempo  $\mathcal{O}(\max(n^2, (n + m)\frac{c}{\epsilon^2} \ln \frac{1}{\delta}))$  para grafos sem peso e  $\mathcal{O}(\max(n^2, (m \log n)\frac{c}{\epsilon^2} \ln \frac{1}{\delta}))$  para grafos com peso, apresentando um erro de no máximo  $\epsilon$  com probabilidade  $1 - \delta$ , para constante  $c > 0$ . Na prática os parâmetros  $\epsilon$  e  $\delta$  podem ser tratados como constantes, sendo que tipicamente fazendo  $\delta = 0.1$  e  $\epsilon = 0.015$ , obtém-se resultados bastante satisfatórios [Riondato e Kornaropoulos 2016].

## 2. Preliminares Matemáticos

Nesta seção, apresentamos as definições, notação e resultados utilizados neste trabalho.

### 2.1. Grafos e Centralidade de Percolação

Dado um grafo  $G = (V, E)$  (direcionado ou não), os estados de percolação  $x_v, \forall v \in V$ , e  $(u, w) \in V^2$ , seja  $S_{uw}$  o conjunto de todos os caminhos mínimos entre  $u$  e  $w$ , e  $\sigma_{uw} = |S_{uw}|$ . A quantidade de caminhos mínimos entre  $u$  e  $w$  em que  $v \in V$  é vértice interno é denotada por  $\sigma_{uw}(v)$ , para  $u \neq v \neq w$ . O conjunto de vértices internos de um caminho mínimo  $p_{uw} \in S_{uw}$  é denotado por  $Int(p_{uw})$ . O conjunto de *predecessores* de  $w$  em  $p_{uw}$  é definido como  $P_u(w) = \{s | s \in V \text{ e } (s, w) \in E_{p_{uw}}\}$ , onde  $E_{p_{uw}}$  denota o conjunto de arestas de  $p_{uw}$ . Seja  $0 \leq x_v \leq 1$  o estado de percolação do vértice  $v \in V$ . Dizemos que

$v$  está totalmente percolado se  $x_v = 1$ , não percolado se  $x_v = 0$  e parcialmente percolado se  $0 < x < 1$ . A centralidade de percolação  $p(v)$  é definida abaixo.

**Definição 1.** Seja  $R(x) = \max\{x, 0\}$ . Dado grafo  $G = (V, E)$  e estados de percolação  $x_v, \forall v \in V$ , a *centralidade de percolação*  $p(v)$  do vértice  $v \in V$  é

$$p(v) = \sum_{\substack{(u,w) \in V^2 \\ u \neq v \neq w}} \frac{\sigma_{uw}(v)}{\sigma_{uw}} \frac{R(x_u - x_w)}{\sum_{\substack{(f,d) \in V^2 \\ f \neq v \neq d}} R(x_f - x_d)}.$$

A Definição 1 segue a notação original. Para simplificar, seja  $r_{uw} = R(x_u - x_w)$  e  $S(v) = \sum_{\substack{(f,d) \in V^2 \\ f \neq v \neq d}} R(x_f - x_d)$ . Assim,  $p(v) = \frac{1}{S(v)} \sum_{\substack{(u,w) \in V^2 \\ u \neq v \neq w}} \frac{\sigma_{uw}(v)r_{uw}}{\sigma_{uw}}$ .

## 2.2. Dimensão Vapnik-Chervonenkis

Em algoritmos que utilizam amostragem, a análise de complexidade de amostra relaciona o tamanho mínimo necessário que uma amostra aleatória deve ter para que os resultados aproximados obtidos estejam consistentes com os parâmetros de confiança e qualidade desejados [Mitzenmacher e Upfal 2017]. Limitantes superiores para a Dimensão VC do espaço de intervalos (definido abaixo) modelado para um determinado problema estabelecem limitantes sobre o tamanho da amostra com essas características. A definição de  $\epsilon$ -amostra formaliza isso (Definição 3). Mais detalhes acerca das definições e resultados apresentados a seguir podem ser encontrados em [Shalev-Shwartz e Ben-David 2014].

Um *espaço de intervalos* é um par  $R = (X, \mathcal{I})$ , onde  $X$  é um domínio (finito ou infinito) e  $\mathcal{I}$  é uma coleção de subconjuntos de  $X$ , denominados *intervalos*. Dado  $S \subseteq X$ , a *projeção* de  $\mathcal{I}$  em  $S$  é o conjunto  $\mathcal{I}_S = \{S \cap I \mid I \in \mathcal{I}\}$ . Se  $|\mathcal{I}_S| = 2^{|S|}$  então dizemos que  $S$  é *despedaçado* por  $\mathcal{I}$ .

**Definição 2.** A Dimensão VC de um espaço de intervalos  $R = (X, \mathcal{I})$ , denotada por  $VCDim(R)$ , é  $VCDim(R) = \max\{d : \exists S \subseteq X \text{ tal que } |S| = d \text{ e } |\mathcal{I}_S| = 2^d\}$ .

**Definição 3.** Sejam  $R = (X, \mathcal{I})$  um espaço de intervalos, e  $\mathcal{D}$  uma distribuição de probabilidade em  $X$ . Um conjunto  $S \subseteq X$  é uma  $\epsilon$ -amostra para  $X$  com respeito a  $\mathcal{D}$  se para todos os conjuntos  $I \in \mathcal{I}$ ,  $|\Pr_{\mathcal{D}}(I) - |S \cap I|/|S|| \leq \epsilon$ , onde  $\Pr_{\mathcal{D}}(I)$  é a probabilidade de um elemento pertencer ao intervalo  $I$  de acordo com a distribuição  $\mathcal{D}$ .

**Teorema 1.** (prova em [Li et al. 2001]) Sejam  $R = (X, \mathcal{I})$  um espaço de intervalos tal que  $VCDim(R) \leq d$  e  $\mathcal{D}$  uma distribuição de probabilidade em  $X$ . Para qualquer  $\epsilon, \delta \in [0, 1]$ , existe  $m = \frac{c}{\epsilon^2} (d + \ln \frac{1}{\delta})$  tal que uma amostra aleatória com respeito a  $\mathcal{D}$  de tamanho  $m$  é uma  $\epsilon$ -amostra para  $X$  com probabilidade  $1 - \delta$ , para constante  $c > 0$ .

Segundo [Löffler e Phillips 2009], a constante  $c$  é aproximadamente 0.5. A seguir, demonstramos que  $VCDim(R) = 0$  se  $|\mathcal{I}| = 1$ .

**Teorema 2.** Um espaço de intervalos  $R = (X, \mathcal{I})$  com  $|\mathcal{I}| = 1$  possui  $VCDim(R) = 0$ . *Demonstração.* Seja  $VCDim(R) = k$ , onde  $k \in \mathbb{N}$ . Então existe um conjunto  $S \subseteq X$  de tamanho  $k$  que é despedaçado por  $\mathcal{I}$ , isto é, existem  $2^{|S|}$  intervalos distintos em  $\mathcal{I}$  tais que  $|\mathcal{I}_S| = 2^k$ . Por outro lado,  $|\mathcal{I}| = 1$ , e então, como  $2^k = 1$ , temos que  $k = 0$ .  $\square$

### 3. Algoritmo Proposto

Nesta seção, modelamos o problema em termos de um espaço de intervalos e apresentamos um algoritmo cuja corretude e tempo de execução sustentam-se no Teorema 1.

#### 3.1. Espaço de Intervalos sobre Caminhos Mínimos

Seja  $S_G$  o conjunto de todos os caminhos mínimos em  $G$ , isto é,  $S_G = \bigcup_{(u,w) \in V^2: u \neq w} S_{uw}$ .

Dado  $v \in V$ , cada caminho  $p_{uw} \in S_G$  é amostrado com probabilidade  $\mathcal{D}^v(p_{uw}) = \frac{r_{uw}}{S(v)\sigma_{uw}}$ . Para cada vértice  $v$ , seja  $\tau_v$  o conjunto de caminhos mínimos  $p_{uw} \in S_G$  em que  $v$  é interno, isto é,  $\tau_v = \{p_{uw} | p_{uw} \in S_G, v \in \text{Int}(p_{uw}) \text{ e } u, w \in V\}$ . Seja também  $R^v = (S_G, \mathcal{I})$  o espaço de intervalos definido para  $v$  onde  $\mathcal{I} = \{\tau_v\}$ . Pelo Teorema 2, como  $|\mathcal{I}| = 1$ , então  $VCDim(R^v) = 0$ .

#### 3.2. Algoritmo

Inicialmente, a matriz de diferenças  $R$ , de dimensão  $(n+2) \times (n+1)$ , é pré-computada no Algoritmo 1. Cada posição  $R[u][w]$  corresponde ao valor  $r_{uw}$ , para  $(u, w) \in V^2$ , como descrito na Definição 1. As posições  $R[n+1][w]$  e  $R[w][n+1]$  contêm os valores  $\sum_{u \in V} r_{uw}$  e  $\sum_{u \in V} r_{wu}$ , respectivamente. A posição  $R[n+1][n+1]$  contém a soma  $\sum_{(u,w) \in V^2: u \neq w} r_{uw}$ , enquanto  $R[n+2][v]$  contém o valor  $S(v)$ .

---

#### Algoritmo 1: PRECALCULADIFERENCAS( $x$ )

---

**Entrada:** O estado de percolação  $x_v$  para todo vértice  $v \in V$ .

**Saída:** Matriz  $R$  de tamanho  $(n+2) \times (n+1)$ .

- 1  $R[i][j] \leftarrow 0, \forall (i, j) \in (n+2) \times (n+1)$
  - 2 **para**  $i \leftarrow 1$  até  $n+1$  **faça**
  - 3     **para**  $j \leftarrow 1$  até  $n+1$  **faça**
  - 4          $\text{dif} \leftarrow (x[i] - x[j]);$
  - 5         **se**  $\text{dif} > 0$  **então**
  - 6              $R[i][n+1] \leftarrow R[i][n+1] + \text{dif}, \quad R[n+1][j] \leftarrow R[n+1][j] + \text{dif};$
  - 7              $R[i][j] \leftarrow \text{dif}, \quad R[n+1][n+1] \leftarrow R[n+1][n+1] + \text{dif};$
  - 8 **para**  $i \leftarrow 1$  até  $n$  **faça**
  - 9      $R[n+2][i] \leftarrow R[n+1][n+1] - R[n+1][i] - R[i][n+1];$
  - 10 **retorna**  $R$
- 

No Algoritmo 2 é apresentado o método proposto, dado um grafo com um estado de percolação  $x_v$ , para todo  $v \in V$ , e os parâmetros de erro e confiabilidade  $\epsilon, \delta \in [0, 1]$ , respectivamente. Em linhas gerais, dois vértices  $u$  e  $w$  são amostrados de acordo com os valores de  $R$  (linha 3), e em seguida um caminho  $p_{uw} \in S_{uw}$  é amostrado uniformemente nas linhas 5–10. Se o vértice  $v$  é interno a  $p_{uw}$ , então  $\tilde{p}(v)$  é incrementado em  $1/r$  (passo que corresponde a média amostral, demonstrada no Teorema 3). O algoritmo utilizado na linha 4 calcula  $\sigma_{uz}, \sigma_{ut}$  e  $P_u(t)$ , necessários para o passo da linha 8.

**Teorema 3.** Para uma amostra  $S$  de tamanho  $r = \frac{c}{\epsilon^2} \ln \frac{1}{\delta}$  e constante  $c > 0$ , a saída do Algoritmo 2 possui erro de no máximo  $\epsilon$  tal que  $\Pr(|p(v) - \tilde{p}(v)| \leq \epsilon) \geq$

**Algoritmo 2:** CENTRALIDADEDEPERCOLACAO( $G, x, v, \epsilon, \delta$ )

**Entrada:** Grafo  $G = (V, E)$  com  $n = |V|$ , estados de percolação  $x$ , vértice  $v \in V$ , erro  $\epsilon$ , confiança  $\delta$ .

**Saída:** Aproximação  $\tilde{p}(v)$  para a centralidade de percolação do vértice  $v$ .

```

1  $R \leftarrow \text{preCalculaDiferencas}(x), \quad r \leftarrow \frac{c}{\epsilon^2} \ln \frac{1}{\delta}, \quad \tilde{p}(v) \leftarrow 0;$ 
2 para cada  $i \leftarrow 1$  até  $r$  faça
3   amostre  $w$  com probabilidade  $\frac{R[n+1][w]}{R[n+2][w]}$  e  $u$  com probabilidade  $\frac{R[u][w]}{R[n+1][w]}$ ;
4    $S_{uw} \leftarrow \text{todosCaminhosMínimos}(u, w);$ 
5   se  $S_{uw} \neq \emptyset$  então
6      $t \leftarrow w;$ 
7     enquanto  $t \neq u$  faca
8       amostre  $z \in P_u(t)$  com probabilidade  $\frac{\sigma_{uz}}{\sigma_{ut}};$ 
9       se  $z \neq u$  e  $z = v$  então  $\tilde{p}(z) \leftarrow \tilde{p}(z) + 1/r;$ 
10       $t \leftarrow z;$ 
11 retorna  $\tilde{p}(v)$ 

```

$1 - \delta$ . *Demonstração.* O Algoritmo 2 amostra um caminho mínimo  $p_{uw}$  com probabilidade  $\mathcal{D}^v(p_{uw})$ , dada a maneira como os vértices  $u$  e  $w$  são selecionados e que o laço das linhas 7–10 amostra um caminho mínimo  $p_{uw}$  de maneira uniforme sobre o conjunto  $S_{uw}$  (Lema 5 em [Riondato e Kornaropoulos 2016]). Seja  $\tilde{p}(v)$  a estimativa para a centralidade de percolação do vértice  $v$  e seja  $S = \{p_1, \dots, p_r\}$  o conjunto de  $r$  elementos de  $S_G$  amostrados pelo Algoritmo 2. Temos que  $\Pr_{\mathcal{D}}(\tau_v)$  é igual a  $\sum_{p_{uw} \in \tau_v} \frac{r_{uw}}{S(v)} \frac{1}{\sigma_{uw}} = \sum_{\substack{(u,w) \in V^2 \\ u \neq v \neq w}} \sum_{p \in S_{uw}} \frac{r_{uw}}{S(v)} \frac{1}{\sigma_{uw}} = \sum_{\substack{(u,w) \in V^2 \\ u \neq v \neq w}} \frac{r_{uw}}{S(v)} \frac{\sigma_{uw}(v)}{\sigma_{uw}} = p(v)$ . Temos também

que  $\tilde{p}(v)$  corresponde a média amostral dada por  $\tilde{p}(v) = \frac{1}{r} \sum_{p \in S} 1_{\tau_v(p)} = \frac{|S \cap \tau_v|}{|S|}$ , onde  $1_{\tau_v(p)}$  é a função indicadora para o conjunto  $\tau_v$  (o valor de  $1_{\tau_v(p)}$  é 1 se  $p \in \tau_v$  e 0 caso contrário). Como  $\left| \Pr_{\mathcal{D}}(\tau_v) - \frac{|S \cap \tau_v|}{|S|} \right| = |p(v) - \tilde{p}(v)|$ , então podemos aplicar os Teoremas 1 e 2 para obter  $d = VCDim(R^v) = 0$  e obter que  $S$  é uma  $\epsilon$ -amostra de tamanho  $r = \frac{c}{\epsilon^2} \ln \frac{1}{\delta}$  tal que  $\Pr(|p(v) - \tilde{p}(v)| \leq \epsilon) \geq 1 - \delta$ .  $\square$

**Teorema 4.** O Algoritmo 2 executa em tempo  $\mathcal{O}(\max(n^2, (n + m) \frac{c}{\epsilon^2} \ln \frac{1}{\delta}))$  para grafos sem peso e em tempo  $\mathcal{O}(\max(n^2, (m \log n) \frac{c}{\epsilon^2} \ln \frac{1}{\delta}))$  para grafos com peso. *Demonstração.* Seja  $T_R$  o tempo de execução do Algoritmo 1. Note que  $T_R = \Theta(n^2)$ . Para a amostragem utilizada nas linhas 3 e 8 utilizamos o algoritmo de [Vose 1991], de tempo linear. Sejam  $T_{su}$  e  $T_{sw}$  os tempos de execução da amostragem do vértice  $u$  e do vértice  $w$ , respectivamente. Então  $T_{su} = T_{sw} = \mathcal{O}(n)$ . Sejam  $T_p$  o tempo de execução do trecho das linhas 7–10 e  $T_{S_{uw}}$  o tempo de execução da linha 4. Como  $|P_u(w)| \leq d_G(w)$ , onde  $d_G(w)$  denota o grau do vértice  $w$  em  $G$ , e o laço do trecho das linhas 7–10 é executado no máximo  $n$  vezes caso o caminho amostrado passe por todos os vértices do grafo, então  $T_p = \sum_{v \in V} d_G(v) = 2m = \mathcal{O}(m)$ . O algoritmo executado na linha 4 é Dijkstra ou BFS, dependendo do grafo ter ou não pesos. Portanto  $T_{S_{uw}}$  é, respectivamente,  $\mathcal{O}(m \log n)$  ou  $\mathcal{O}(n + m)$ . Como o laço principal

das linhas 2–10 é executado  $r$  vezes, e  $r \in \mathcal{O}(\frac{1}{\epsilon^2} \ln \frac{1}{\delta})$ , então o custo total do Algoritmo 2 é  $\mathcal{O}(\max(T_R, T_{su}, T_p, T_{S_{uw}})) = \mathcal{O}(\max(n^2, r(n, m, T_{S_{uw}})))$ , que corresponde a  $\mathcal{O}(\max(n^2, \frac{c}{\epsilon^2} \ln \frac{1}{\delta}(n + m)))$  em grafos sem peso e  $\mathcal{O}(\max(n^2, \frac{c}{\epsilon^2} \ln \frac{1}{\delta}(m \log n)))$  em grafos com peso.  $\square$

#### 4. Conclusão

Apresentamos um algoritmo baseado em amostragem e conceitos da teoria da Dimensão VC para o cálculo da centralidade de percolação de um vértice que permite a escolha entre baixo tempo de execução ou alta precisão na estimativa por meio parâmetros  $\epsilon$  e  $\delta$ . Para o caso de  $\epsilon$  e  $\delta$  constantes, este é o algoritmo mais rápido conhecido para o problema. Observamos que o algoritmo proposto neste trabalho, quando modificado para o problema mais geral de estimar a centralidade de percolação de todos os vértices do grafo, torna-se menos eficiente que o melhor algoritmo exato conhecido. Dessa forma, como trabalhos futuros, outras técnicas e resultados da área de complexidade de amostra como médias de Rademacher poderiam ser utilizadas para reduzir o número de amostras necessárias, servindo como base no projeto de um algoritmo de tempo  $o(n^3)$  para calcular a centralidade em todos os vértices do grafo. Finalmente, uma análise experimental comparativa dos algoritmos seria útil para verificar o comportamento do algoritmo proposto na prática e como usá-lo para equilibrar tempo de execução com os parâmetros de qualidade e confiança desejados.

#### Referências

- Brandes, U. (2001). A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25(163):163–177.
- Broadbent, S. e Hammersley, J. M. (1957). Percolation processes: I. crystals and mazes. *Math. Proc. of the Cambridge Philosophical Society*, 53(3):629–641.
- Li, Y., Long, P. M., e Srinivasan, A. (2001). Improved bounds on the sample complexity of learning. *Journal of Computer and System Sciences*, 62(3):516–527.
- Löffler, M. e Phillips, J. M. (2009). Shape fitting on point sets with probability distributions. In *European Symposium on Algorithms*, pages 313–324. Springer.
- Mitzenmacher, M. e Upfal, E. (2017). *Probability and computing: Randomization and Probabilistic Techniques in Algorithms and Data Analysis*. Cambridge University Press.
- Piraveenan, M., Prokopenko, M., e Hossain, L. (2013). Percolation centrality: Quantifying graph-theoretic impact of nodes during percolation in networks. *PLOS ONE*, 8(1):1–14.
- Riondato, M. e Kornaropoulos, E. M. (2016). Fast approximation of betweenness centrality through sampling. *Data Mining and Knowledge Discovery*, 30(2):438–475.
- Shalev-Shwartz, S. e Ben-David, S. (2014). *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press.
- Vose, M. D. (1991). A linear algorithm for generating random numbers with a given distribution. *IEEE Transactions on software engineering*, 17(9):972–975.

# A Possibility for Handwriting Trajectory Reconstruction with Deep Learning

Yu Han Lin<sup>1</sup>, Erikson F. de Moraes<sup>1</sup>, Simone B. K. Aires<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática – Universidade Tecnológica Federal do Paraná (UTFPR)  
Av Monteiro Lobato s/n, Km 04 – Ponta Grossa – PR – Brasil

nilyuhan@gmail.com, {emorais, sbkaminski}@utfpr.edu.br

**Abstract.** *Studies on handwriting trajectory recovery have gained space in the offline recognition of manuscript researches. The reason is the use of online recognition resources, creating techniques to simulate the writing of the handwritten word and inserting the simulated coordinates of the pixels in online handwriting recognition systems. The present work aims to present the possibility to perform the reconstruction of the trajectory of handwritten characters using an existing deep learning architecture. This reconstruction involves recovery of the coordinate sequence of the trajectory of the character and, if needed, the retrieval of missing parts of the characters. First of all, the state of art about handwriting trajectory is presented. Experiments of handwriting characters trajectory recovery using deep learning model is performed and shows that the actual deep models of handwriting trajectory recovery architecture can recover some missing parts of the characters. In the end, presents the conclusion of the work.*

**Keywords:** Deep Learning, Handwriting Trajectory Recovery, Handwriting Reconstruction, Image Recovery

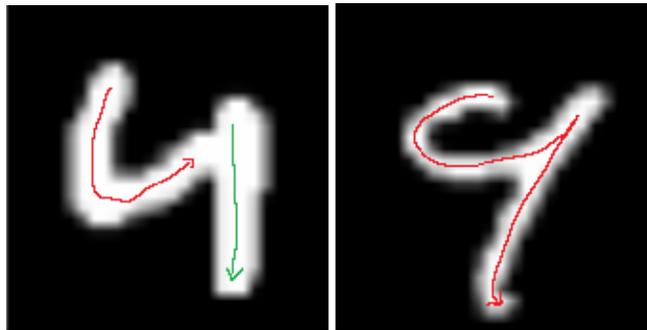
## 1. Introduction

The ability to transmit a message or idea through specific handwriting language is unique to each person. Handwriting recognition consists to transform the language in graphics marks into digital signals [Plamondon and Srihari 2000].

The handwriting analysis and recognition can be classified into two categories: Offline and Online. The online recognition is to transcript the handwriting from some special input device, which provides dynamic and temporal information, like the sequence coordinate points and pen pressure; and the offline recognition is the process that extracts and identifies the handwriting from static handwriting images [Plamondon and Srihari 2000].

By the limited information of offline images, offline recognition is performed with less accuracy than online recognition [Zhang et al. 2017]. Based on that limitation, handwriting trajectory recovery (HTR) system is developed, recovering the temporal order of the handwriting strokes from statics images.

The main task of HTR is to find an oriented path of the strokes of the image similar to that done during the writing operation. With the dynamics information recovered, handwriting images that have similar geometric forms can be distinguished [Noubigh and Kherallah 2017]. The example of those situations is shown in Figure 1, in which both image numbers (four and nine) have similarities in their geometric shapes.



**Figure 1. The example of number four (on the left) and nine (on the right), which is written with two and one stroke, respectively.**

To recover the online data, [KumarBhunja et al. 2018] uses the recent successful deep learning model to propose a novel technique for HTR by using a merged of Convolutional Neural Network (CNN) with Long Short-Term Memory (LSTM) to predict coordinates points from a given offline character.

Following the idea of recovering data, the reconstruction handwriting process also can be explored. This paper explores the model proposed by [KumarBhunja et al. 2018] to perform the reconstruction of handwriting trajectory. In this work, the reconstruction consists of the recovering of handwriting trajectory from the character image and also the retrieve of missing parts of those images when it went through (e.g., natural causes of disintegration). To simulate the disintegration of an offline image, we artificially remove some parts of the image to perform the reconstruction process.

The organization of this paper is described as follows. Firstly, we present a bibliography of a trajectory recovery based handwriting recognition system. Then, the methodology and implementation of the experiment are presented in section 3. Experiment results and analysis are discussed in section 4. Section 5 brings up the conclusion and future works.

## 2. Related Work

The works [Noubigh and Kherallah 2017] and [Nguyen and Blumenstein 2010] have reported that the techniques have adhered to the hypothesis of [Rousseau et al. 2006] trying to recovery dynamic information from static offline character images.

The two major approaches found in the literature to recovery handwriting trajectory are heuristic rules algorithm to find the optimal graph path and artificial intelligence or machine learning-based algorithm. The first consists of preprocessing the character offline image using a technique called skeletonization, then convert the character skeleton

into a graph and develop an algorithm to find the optimal graph path. The second consists of apply artificial intelligence or deep learning to approximate the probability of the trained model to reproduce the handwriting.

The work present by [Sharma 2013] recover the handwriting trajectory by using two algorithms that use the code chain from a skeleton character image. In the sequence [Sharma 2015] use the dynamic features from the previous work [Sharma 2013] and the statics features from static handwriting images to perform the handwriting character recognition, showing that the combination of both types of features can achieve better results than using only one type of feature.

By using a genetic algorithm, [Elbaati et al. 2009] optimizes the best trajectory from segments obtained from a character skeleton image. Dynamic information is restored from the stroke chronology and is used to perform the Arabic handwriting recognition.

The deep learning was a recent technique applied by [KumarBhunia et al. 2018]. The principle is using the feature extraction from convolutional neural networks to extract the feature sequence from the offline handwriting image, then use the encoder-decoder LSTM in seq2seq model to compute the feature codified by the convolutional neural network and produce the final coordinates sequence.

### 3. Methodology

In this section, we detail the offline to online data translation process, introducing the dataset and the handwriting reconstruction process. The deep neural network is from the work of [KumarBhunia et al. 2018], which is used as a start point experiment to perform the handwriting reconstruction and recover missing parts of the characters.

#### 3.1. Reconstruction Process

With the predicted points, we can perform the possibility of reconstruction process. To check if the network can recover some missing parts, four regions are delimited to be withdrawn: upper left, upper right, down left and downright. The Figure 2 shows the delimited region from a offline Telugu character.

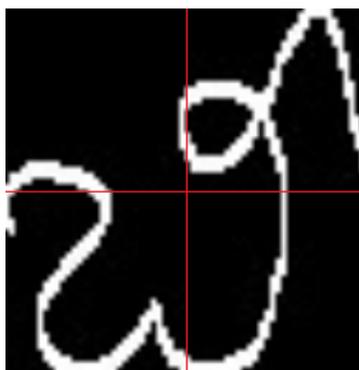


Figure 2. Regions delimited to be withdrawn.

This occlusion test was performed in samples in which the sequence coordinate points were predicted correctly by the network.

### 3.2. Implementation Details

The length of each layer input and output of the CNN is described in Table 1.

**Table 1. The Input and Output of each layer from CNN**

<i># layer</i>	<i>Input</i>	<i>Output</i>
1	4096	64
2	64	128
3	128	256
4 - 8	256	256

The pooling method 1x2 [Shi et al. ] is adopted and is applied after the layers 1, 2, 4, 5 and 7. Batch normalization is applied after 3<sup>th</sup>, 6<sup>th</sup> and 8<sup>th</sup> convolutional layer, speeding up the training process. ReLu activation function is applied after the layers 3, 6 and 8.

The two bidirectional LSTM with 256 cells each are stacked to have higher abstraction ability [KumarBhunia et al. 2018]. To obtain each point from the decoder network at a particular time-step  $t$ , a fully connected layer is used (with 256 input length and 2 output), and the previous time-step is the input at the next time-step of the decoder.

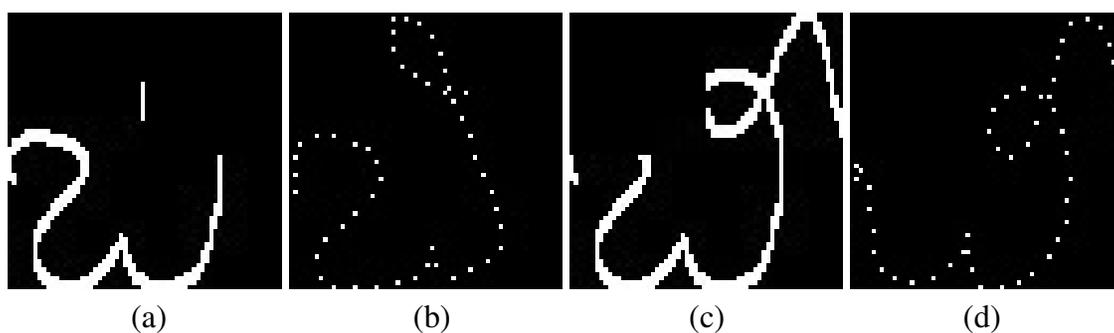
## 4. Experiments and Results

For our experiment, [KumarBhunia et al. 2018] provides a dataset with 7875 samples of Telugu script. It is a type of Indic script which contains the character level of both online and offline data. We train our model with those samples to predict the coordinate points of the characters.

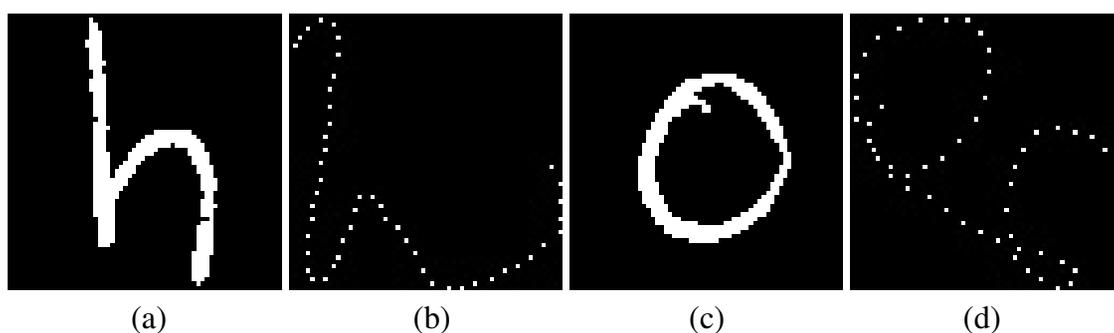
The predicted coordinate points are translated to the nearest point on the skeleton of the offline image as a post-processing step. For the characters that the shape format which achieved a satisfactory result, the occlusion test was performed as described previously. The Figure 3 shown the offline images used to check if the network can reconstruct the shape of the original character when it's provided an offline character with missing parts. For each sample showed in Figure 3, the original image is the one on the left (with a full traced path of the Telugu character) and the predicted coordinates image is on the right.

Figure 3b shows that the network can recover partially the missing part of the offline character. The Figure 3d also shows the network behavior to recover the curve from the missing upper left region, producing a little curve at the same region in the output.

As part of the experiments, some samples of Latin character from IRONOFF dataset is used to check the behavior of network when it is trained with Telugu characters and fed with Latin characters, showed in Figure. 4. This process can provide an initial experiment for the technique called Transfer Learning [Pan and Yang 2009].



**Figure 3. Reconstruction of offline characters with online coordinates data translated to offline.**



**Figure 4. Reconstruction of offline Latin characters with online coordinates data translated to offline when the network was trained with Telugu characters.**

It is to be expected that the network will not recover Latin characters. However, the Figure 4b shows the output of network seems similar from the input Figure 4a. In the case shown by Figure 4d, the network can reproduce the complete loop from the original character Figure 4c, but output some additional parts. The network could interpret that the character has some missing part, compared to some Telugu character learned. Those examples show the capability of the network to generalize even be trained with another dataset of another language.

## 5. Conclusion and Future Works

In this work, experiments using deep learning aiming to reconstruct handwriting trajectory was performed. Those experiments include the use of the framework proposed by [KumarBhunja et al. 2018] and have show a possibility for handwriting trajectory reconstruction when the handwriting character image has hidden parts.

Even without a large amount of data to perform the training, the deep neural network could partially reconstruct a few handwriting images, showing that there are indications that deep neural networks may have the ability to predict pixels in regions that have been removed or lost.

That prediction ability brings up possibilities to continue this research. In the future works, we will train the network with Latin characters from IRONOFF database. Transfer Learning from a Telugu trained network to train the Latin network will be on

the line. Also, use the zoning proposed by [Freitas et al. 2007] to perform the withdraw regions, so we can have a zoning mechanism guide to simulate missing parts of an offline character image.

## References

- Elbaati, A., Kherallah, M., Ennaji, A., and Alimi, A. M. (2009). Temporal order recovery of the scanned handwriting. In *2009 10th International Conference on Document Analysis and Recognition*, pages 1116–1120. IEEE.
- Freitas, C. O., Oliveira, L. S., Bortolozzi, F., and Aires, S. B. (2007). Handwritten character recognition using nonsymmetrical perceptual zoning. *International Journal of Pattern Recognition and Artificial Intelligence*, 21(01):135–155.
- KumarBhunia, A., Bhowmick, A., Bhunia, A. K., Konwer, A., Banerjee, P., Roy, P. P., and Pal, U. (2018). Handwriting trajectory recovery using end-to-end deep encoder-decoder network. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3639–3644. IEEE.
- Nguyen, V. and Blumenstein, M. (2010). Techniques for static handwriting trajectory recovery: a survey. In *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems*, pages 463–470. ACM.
- Noubigh, Z. and Kherallah, M. (2017). A survey on handwriting recognition based on the trajectory recovery technique. In *2017 1st International Workshop on Arabic Script Analysis and Recognition (ASAR)*, pages 69–73. IEEE.
- Pan, S. J. and Yang, Q. (2009). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359.
- Plamondon, R. and Srihari, S. N. (2000). Online and off-line handwriting recognition: a comprehensive survey. *IEEE Transactions on pattern analysis and machine intelligence*, 22(1):63–84.
- Rousseau, L., Camillerapp, J., and Anquetil, E. (2006). What knowledge about handwritten letters can be used to recover their drawing order? In *Tenth international workshop on frontiers in handwriting recognition*. Suvisoft.
- Sharma, A. (2013). Recovery of drawing order in handwritten digit images. In *2013 IEEE Second International Conference on Image Information Processing (ICIIP-2013)*, pages 437–441. IEEE.
- Sharma, A. (2015). A combined static and dynamic feature extraction technique to recognize handwritten digits. *Vietnam Journal of Computer Science*, 2(3):133–142.
- Shi, B., Bai, X., and Yao, C. An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016.
- Zhang, X.-Y., Bengio, Y., and Liu, C.-L. (2017). Online and offline handwritten chinese character recognition: A comprehensive study and new benchmark. *Pattern Recognition*, 61:348–360.

# Uma Revisão dos Métodos de Implementação de Sistemas Multiagentes em Ambientes Físicos com Foco em Artefatos

Fabian Cesar Pereira Brandão Manoel<sup>1</sup>, Carlos Eduardo Pantoja<sup>1</sup>

<sup>1</sup>Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (CEFET/RJ)  
Rio de Janeiro, Brasil

fabiancpbm@gmail.com.br, pantoja@cefet-rj.br

**Abstract.** *In the Multiagent Systems research community, it is possible to find areas of study of agents, environments with artifacts, and organization of groups and society. However, the focus in these areas has been on development in simulated environments, making it difficult for developers to migrate to physical environments. Although approaches can already be found that allows agents the ability to perceive and act in physical environments, there is still little exploration for artifacts to interact with these scenarios as well. Besides, the emergence of paradigms such as Intelligent Environment has increased interest in SMA applications in physical environments. This paper presents a Systematic Review of SMA applied in physical environments, focusing on the artifact dimension as an important approach to be considered in the development of these SMA.*

**Resumo.** *Na comunidade de pesquisa de Sistemas Multiagentes, é possível encontrar áreas de estudo de agentes, ambientes com artefatos, e organização de grupos e sociedade. No entanto, o foco nessas áreas tem voltado para o desenvolvimento em ambientes simulados, dificultando a migração dos programadores para os ambientes físicos. Embora já seja possível encontrar abordagens que permitam aos agentes a capacidade de perceber e agir em ambientes físicos, ainda não há muita exploração para que artefatos também interajam com estes cenários. Além disso, o surgimento de paradigmas como o Ambiente Inteligente aumentaram o interesse pelas aplicações SMA em ambientes físicos. Este artigo traz uma Revisão Sistemática do SMA aplicada em ambientes físicos, focando na dimensão de artefatos como uma importante abordagem a ser considerada no desenvolvimento desses SMA.*

**Keywords:** Sistema Multiagente; Artefatos; Ambientes Físicos;

## 1. Introdução

A Teoria da Atividade (TA) [Vygotzky 1980] é um estudo da psicologia soviética que busca entender como o ser humano aprende e como a sociedade evolui. Este tema tem servido como base para fundamentar pesquisas em ciência da computação que focam na interação humana-computacional e no trabalho coletivo. Em Sistemas Multi-Agente (SMA), a TA teve um papel fundamental ao enfatizar o papel do fator ambiental/social no desenvolvimento cognitivo de atividades colaborativas. De acordo com essas teorias,

grande partes das atividades humanas são assistidas por algum tipo de artefato, cujo design e o uso dessas ferramentas possui papel fundamental no desenvolvimento de atividades. Além disso, o desenvolvimento das sociedades humanas está estritamente relacionado ao desenvolvimento das ferramentas construídas e usadas nessas sociedades. Portanto, esta teoria reforça a importância de Artefatos, que são entidades que representam dispositivos computacionais que populam o ambiente de trabalho do agente e possuem funções que auxiliam nas atividades destes agentes [Ricci et al. 2006b, Ricci et al. 2007]. Os agentes podem se relacionar com artefatos para realizar **(i) seleção**, **(ii) uso** ou **(iii) reconstrução/manipulação** - que se trata da criação ou obtenção de um novo artefato caso as tentativas (i) e (ii) tenham falhado [Omicini et al. 2005]. Além disso, artefatos são compostos por 4 elementos [Ricci et al. 2006b]: *usage interface (UI)*, que é um conjunto de operações que agentes podem invocar para usar o artefato; *Operating Instructions (OI)*, que é uma descrição de como o artefato deve ser usado para que as suas funcionalidades possam ser acessadas; *Function*, que é o propósito de existência do artefato; e *Structure and Behaviour*, que são as características de implementação do artefato.

Com a crescente demanda para tornar ambientes e processos inteligentes e autônomos, como pode ser visto em Ambiente Inteligente (AmI) (ambientes que usam tecnologias autônomas para tornar possível a presença de dispositivos inteligentes de forma onipresente e imperceptível ao usuário [Aarts and De Ruyter 2009]) cientistas na área de SMA têm buscado criar métodos para utilizar estes SMA em ambientes físicos [Brödner 2018]. Além disso, a TA surge na área de SMA para embasar a importância da inclusão ambiental/social para o desenvolvimento cognitivo de um agente. A contribuição da TA para SMA mostrou ser mais eficiente do que a antiga ideia baseada na forte noção de agência - ao tratar agentes como entidades individuais, a implementação do sistema poderia gerar inconsistências e desorganização, dependendo da complexidade do problema abordado [Ricci et al. 2006b].

Sendo assim, este trabalho tem como objetivo realizar uma revisão sistemática sobre a implementação de SMA em ambientes físicos focando na abstração de artefatos e, com isso, gerar fundamentação teórica para trabalhos futuros. Alguns trabalhos já descrevem o papel dos ambientes em SMA [Weyns and Holvoet 2005, Andrea Omicini et al. 2007] e fornecem discussões sobre este tipo de abordagem no cenário atual [Brödner 2018]. Este artigo está organizado da seguinte forma: metodologia de pesquisa na seção 2, discussão dos trabalhos relacionados ao tema na seção 3 e a uma conclusão da revisão na seção 4.

## 2. Metodologia da Revisão

O objetivo desta revisão sistemática da literatura é o de conhecer o estado da arte sobre a implementação de SMA em ambientes físicos focando na utilização de artefatos. Para isso, as seguintes questões de pesquisa foram levantadas: (1) Qual a relevância de ter um ambiente físico controlado por um SMA? (2) Como artefatos são implementados para ambientes físicos? (3) Quais os problemas existentes ao implementar artefatos de um SMA para ambientes físicos? (4) É possível que SMA sejam usados para controlar ambientes físicos com altos requisitos de confiabilidade?

Para encontrar os trabalhos relacionados que ajudarão com as questões de pesquisa, uma *string* foi formada com base nos seguintes itens que definem a busca: (1) expressões para a base primária de pesquisa (SMA), como "*multiagent*" e "*multi-agent*"; (2) termo que representa ambientes em SMA, como "*artifact*"; e (3) termos relacionados a ambientes físicos na área de industrial, petrolífera, residencial, robótica e veicular. Cada item foi escrito de forma restritiva na *string* de busca, ou seja, utilizando a palavra *AND*. Já os sinônimos encontrados em cada item foram escritos utilizando a palavra *OR* entre eles. A partir deste processo, os termos de busca ficaram definidas na seguinte *string*: ("*multiagent*"*OR* "*multi-agent*") *AND* (*artifact*\*) *AND* (*industr*\* *OR* "*home*"*OR* *resident*\* *OR* "*oil*"*OR* "*automation*"*OR* "*physical environment*"*OR* "*real world*"*OR* "*internet of things*"*OR* "*device*"*OR* *robot*\* *OR* "*car*"*OR* "*automobile*"*OR* "*vehicle*").

Esta *string* de busca foi executada no *Scopus*, *ACM* e *IEEE Xplore*, onde foram retornados, respectivamente, 104, 12 e 24 trabalhos, totalizando 140 resultados. Para a seleção dos trabalhos mais relevantes, foram adotados os seguintes critérios de inclusão e exclusão:

1. Inclusão: (a) Trabalhos que respondam a pelo menos uma das questões de pesquisa; (b) Trabalhos que tragam discussões sobre a relação entre SMA e ambientes físicos; (c) Trabalhos que utilizem um ambiente simulado que represente cenários industriais, residenciais, petrolíferos, robóticos ou veiculares.
2. Exclusão: (a) Trabalhos que não simulem ou implementem ambientes reais, ou seja, aqueles que, mesmo falando de artefatos e ambientes, não demonstram o cenário no trabalho; (b) Trabalhos similares entre si, onde neste caso, a inclusão deste trabalho se faria redundante para esta revisão; (c) Proceedings.

Primeiramente, foi feita a leitura de todos os 140 *abstracts*. Utilizando os critérios adotados, foram selecionados 16 trabalhos para leitura completa, sendo [Takeda et al. 2000] e [Andrea Omicini et al. 2007] encontrados na base *IEEE Xplore* e os demais na base *Scopus*. Além disso, quatro trabalhos foram aproveitados como referencial teórico [Ricci et al. 2006b, Ricci et al. 2007, Weyns and Holvoet 2005, Andrea Omicini et al. 2007].

### 3. Discussão dos Trabalhos

Esta seção apresenta uma discussão dos trabalhos selecionados, buscando apresentar a necessidade, vantagens e desvantagens de aplicar SMA em ambientes físicos com foco em artefatos e comparar os trabalhos relacionados.

Uma discussão em [Brödner 2018] indica que SMA tem sido um tema de interesse em busca de otimização computacional para a inteligência artificial, uma vez que é remontada a ideia de que a capacidade do SMA de realizar tarefas simples, coordenadas e distribuídas entre vários agentes produza inteligência artificial mais leve do que a tradicional. Apesar dessa ideia ter sido considerada incorreta (já que agentes também possuem restrições computacionais), SMA ainda devem ser considerados para formar sistemas físicos devido à capacidade que seus agentes têm de coletar dados do ambiente, processar informações de forma adaptativa e orientada a objetivos, e gerar respostas ao

ambiente ou outros agentes. Um cenário onde a forte noção de agência pode ser crítica pode ser visto em [Amaral et al. 2019].

Em [Issicaba et al. 2018], são apresentados experimentos de laboratório para verificar o raciocínio de agentes BDI para o Controle Web of Cell (WoC) - que é uma abordagem de rede para mapear células do sistema de energia elétrica e gerenciá-los de forma distribuída. Para a aplicação do paradigma de agentes, utilizou-se o JASON, e para paradigma de ambiente, utilizou-se o framework *CARtAgO* [Ricci et al. 2006a]. Porém, embora este trabalho tenha sido aplicado para ambientes físicos e seus agentes, sua solução foi específica para esta aplicação, sendo difícil utilizar esta solução em outro projeto.

Em [Amaral et al. 2019], o componente *Apache Camel* para artefatos (*CamelArtifact*) é demonstrado no cenário da indústria 4.0. Para isso, um SMA com a abordagem de Agentes e Artefatos (A&A) foi implementado e, por ter sido feito para um ambiente que demanda alta performance, a implementação com artefatos é necessária por ser computacionalmente mais leve. Além disso, o uso do *Apache Camel* possibilita um roteamento de mensagens leve e a integração com diversos dispositivos e tecnologias de rede, como o OPC, REST, e-mail, MQTT e Telegram. Apesar da solução tornar possível implementações em ambientes industriais físicos, é preciso ficar atento quanto à dependência na tecnologia *Apache Camel*. Trabalhos futuros podem focar em aplicações de artefatos físicos que mantenham o caráter heterogêneo independente do *Apache Camel*. Além destes trabalhos, outros encontrados na revisão possuem características semelhantes, e portanto podem ser agrupados de acordo com estas características, como está sendo representado na Tabela 1.

#### 4. Conclusão

Este artigo apresentou uma revisão sobre a aplicação SMA aplicados a ambientes físicos, buscando entender e explorar a relevância dos artefatos como facilitadores no desenvolvimento de sistemas deste tipo. Para isso, questões de pesquisa foram levantadas e os resultados de busca foram adaptados para garantir tanto o embasamento teórico quanto a exploração de características divergentes entre os trabalhos selecionados.

Sobre a relevância de ter um ambiente físico controlado por um SMA, os agentes de um SMA podem receber percepções do ambiente, realizar planos orientado a objetivos, gerando respostas ao ambiente ou outros agentes. Isso torna um SMA sensível às modificações de um ambiente físico. Já sobre a implementação de artefatos para ambientes físicos, foi visto que, atualmente, alguns SMA são implementados sem utilizar artefatos; já outras aplicações implementam artefatos por meio de simulações; e por último, artefatos aplicados em ambientes físicos costumam ser específicos para a abordagem do trabalho, dificultando a reprodutibilidade da aplicação. Artefato apresenta grande vantagem para sistemas físicos, porém, não deve ser considerado como uma solução livre de limitações computacionais. Sobre a capacidade de um SMA de ser usado para controlar ambientes físicos com altos requisitos de confiabilidade, foram apresentadas implementações na área de petróleo, indústria 4.0, etc. que mostraram que SMA são capazes de atuar nestes cenários. Porém, deve-se destacar que o nível de confiabilidade tende a aumentar a medida que novas abordagens surgem para ambientes físicos.

**Tabela 1. Tabela Comparativa dos Trabalhos**

<b>Trabalho</b>	<b>Característica</b>
[Molfino et al. 2008, De Carolis et al. 2005, Locatelli et al. 2010, Pecora and Cesta 2007, Rincon et al. 2018, Fahad et al. 2014]	Representa um cenário real, porém, implementado em um ambiente simulado, e com isso, não realiza conexões com o meio físico.
[Takeda et al. 2000, Issicaba et al. 2018]	Utiliza abordagem de SMA ou similar conectado a um ambiente físico, porém a aplicação é específica para o trabalho, não podendo reutilizar com facilidade a abordagem implementada em outros trabalhos.
[Amaral et al. 2019], [Sanchez-Pi et al. 2015]	Realiza testes em ambientes críticos e que, portanto, demandam alto grau de confiabilidade.
[Sanchez-Pi et al. 2015]	Considerando apenas este trabalho, a abordagem de agentes é usada para representar objetos como sensores e atuadores, podendo resultar em uma aplicação pesada para agentes dependendo do tamanho do projeto.

Por fim, esta revisão mostrou que a implementação ambientes físicos através de SMA é uma abordagem que vem ganhando cada vez mais importância e que ainda deve ser melhorada para suportar cenários críticos. Além disso, novas ferramentas que ajudem o desenvolvedor a implementar SMA sem preocupações sobre detalhes de tecnologia já estão sendo implementadas. Com isso, SMA com uso de artefatos tem ganhado a característica de um sistema robusto e adaptativo, e portanto, estão se tornando cada vez mais preparados para aplicações em ambientes físicos e complexos.

## Referências

- Aarts, E. and De Ruyter, B. (2009). New research perspectives on ambient intelligence. *Journal of Ambient Intelligence and Smart Environments*, 1(1):5–14.
- Amaral, C., Cranefield, S., Hübner, J., and Roloff, M. (2019). Giving camel to artifacts for industry 4.0 integration challenges. 11523 LNAI:232–236.
- Andrea Omicini, Alessandro Ricci, and Giuseppe Vizzari (2007). Building smart environments as agent workspaces.
- Brödner, P. (2018). “super-intelligent” machine: technological exuberance or the road to subjection. 33(3):335–346.

- De Carolis, B., Cozzolongo, G., Pizzutilo, S., and Plantamura, V. (2005). Agent-based home simulation and control. 3488 LNAI:404–412.
- Fahad, M., Boissier, O., Maret, P., Moalla, N., and Gravier, C. (2014). Smart places: Multi-agent based smart mobile virtual community management system. 41(4):1024–1042.
- Issicaba, D., Rosa, M., Prostejovsky, A., and Bindner, H. (2018). Experimental validation of BDI agents for distributed control of electric power grids. In *2017 IEEE PES Innovative Smart Grid Technologies Conference Europe, ISGT-Europe 2017 - Proceedings*, volume 2018-January, pages 1–6.
- Locatelli, M., Loregian, M., and Vizzari, G. (2010). Artificial societies in a community-based approach to ambient intelligence. 53(8):1152–1168.
- Molfino, R., Carca, E., Zoppi, M., Bonsignorio, F., Callegari, M., Gabrielli, A., and Principi, M. (2008). A multi-agent 3d simulation environment for clothing industry. 5325 LNAI:53–64.
- Omicini, A., Ricci, A., and Viroli, M. (2005). Coordination artifacts as first-class abstractions for mas engineering: State of the research. In *International Workshop on Software Engineering for Large-Scale Multi-agent Systems*, pages 71–90. Springer.
- Pecora, F. and Cesta, A. (2007). DCOP for smart homes: a case study. 23(4):395–419.
- Ricci, A., Viroli, M., and Omicini, A. (2006a). Cartago: A framework for prototyping artifact-based environments in mas. In *International Workshop on Environments for Multi-Agent Systems*, pages 67–86. Springer.
- Ricci, A., Viroli, M., and Omicini, A. (2006b). Programming MAS with artifacts. 3862 LNAI:206–221.
- Ricci, A., Viroli, M., and Omicini, A. (2007). "give agents their artifacts: The a&a approach for engineering working environments in MAS. In *Proceedings of the International Conference on Autonomous Agents*, pages 613–615.
- Rincon, J., Garcia, E., Julian, V., and Carrascosa, C. (2018). The JaCalIVE framework for MAS in IVE: A case study in evolving modular robotics. 275:608–617.
- Sanchez-Pi, N., Leme, L., and Garcia, A. (2015). Intelligent agents for alarm management in petroleum ambient. 28(1):43–53.
- Takeda, H., Ueno, A., Saji, M., Nakano, T., and Miyamoto, K. (2000). A robot recognizing everyday objects-towards robot as autonomous knowledge media. 2:1107–1112.
- Vygotsky, L. S. (1980). *Mind in society: The development of higher psychological processes*. Harvard university press.
- Weyns, D. and Holvoet, T. (2005). On the role of environments in multiagent systems. 29(4):409–421.

# Aparecimento de cliques em grafos aleatórios lei de potência no modelo de ligação preferencial

David Reksidler Jr.<sup>1</sup>, Murilo V. G. da Silva<sup>1</sup>

<sup>1</sup>DINF – Universidade Federal do Paraná (UFPR)  
Curitiba, Brazil

juniorrek@ufpr.br, murilo@inf.ufpr.br

**Abstract.** *In this paper we present preliminary results on the occurrence of cliques of a given size in power law graphs sampled from a distribution of random graphs based on the preferential attachment paradigm. We derive explicitly a formula for the probability of a clique of a prescribed size to appear in a graph sampled from the referred distribution. These results are part of a more general project aiming to investigate the performance of algorithms for the maximum clique problem in power law graphs.*

**Resumo.** *Neste artigo, apresentamos estudos preliminares sobre o surgimento de cliques de diferentes tamanhos em grafos aleatórios lei de potência amostrados segundo o modelo de ligação preferencial. Derivamos explicitamente a fórmula para a probabilidade de uma clique de tamanho prescrito aparecer na amostragem. Estes resultados estão inseridos em um projeto em andamento mais amplo que visa investigar o desempenho de algoritmos para o problema de clique máxima em grafos lei de potência.*

**Keywords:** Cliques; Power Law Graphs; Random Graphs

**Palavras-chave:** Cliques; Grafos lei de potência; Grafos Aleatórios

## 1. Introdução

Com o avanço na capacidade de processamento e armazenamento de grandes quantidades de dados, foi observado em diversos estudos [Faloutsos et al. 1999, Guelzim et al. 2002, Eubank et al. 2004] que muitas redes complexas de grande porte, advindas de diversas situações práticas, desde a *World Wide Web* até redes sociais e biológicas, apresentam uma distribuição lei de potência nos graus de seus vértices. Tais redes, chamadas neste artigo de *grafos lei de potência*, de maneira simplificada, apresentam poucos vértices com grau alto e muitos vértices com grau baixo. Formalmente, em um grafo lei de potência, o número de vértices com um determinado grau  $k$  é  $k^{-\beta}$ , onde  $\beta > 0$  é o expoente que caracteriza a lei de potência e  $k$  é uma constante de proporcionalidade.

Do ponto de vista algorítmico, a relevância deste fato é a evidência crescente de que alguns problemas de otimização parecem admitir soluções mais eficientes em grafos lei de potência do que em outros grafos em geral. Estudos empíricos

[Park and Lee 2001, Silva et al. 2013, Rogiski 2016, Pinto and da Silva 2017] mostram que algoritmos gulosos atingem soluções muito próximas da ótima em tais grafos, em particular, melhores que as garantias de aproximação que se aplicam a grafos no caso geral para problemas como o da coloração de vértices, clique máxima e o da cobertura mínima por vértices. Isso explica em parte porque problemas computacionalmente difíceis em algumas circunstâncias admitem soluções eficientes na prática. Recentemente alguns estudos analíticos [Shen et al. 2012, Gast and Hauptmann 2014, Gast et al. 2015, Vignatti and da Silva 2016] têm procurado esclarecer este fato e dar sustentação matemática a tal fenômeno.

O presente trabalho se insere neste contexto, investigando o problema da clique máxima em grafos lei de potência, dado a falta de análise teórica presente na literatura acerca de tal problema. O primeiro passo para entender o comportamento de algoritmos para clique máxima é compreender o fenômeno do aparecimento de cliques em modelos de grafos aleatórios que seguem a lei de potência. Neste estudo preliminar investigamos o aparecimento de cliques de diversos tamanhos em um modelo conhecido como *modelo de ligação preferencial*. O objetivo disso é estabelecer as bases para o prosseguimento deste projeto que visa o esclarecimento de quão rápido decai a quantidades de cliques nestes grafos à medida que o tamanho requerido para tal clique cresce e como isso impacta no desempenho de algoritmos para o problema da clique máxima.

## 2. Modelos de Grafos Aleatórios

O estudo de modelos de grafos aleatórios iniciou-se na década de 60 com [Erdős and Rényi 1960]. Os pesquisadores buscavam analisar redes complexas, propondo modelos matemáticos capazes de prever algumas características de tais redes muito antes que dados massivos de redes de grande escala estivessem disponíveis, como é o caso hoje. Tais modelos podem não capturar com exatidão fenômenos de muitos grafos advindos de aplicações práticas, pelo fato de induzirem uma distribuição normal (ao invés de, por exemplo, lei de potência) nos graus dos vértices. Ainda assim, a simplicidade dos modelos de Erdos e Rényi resultou na ascensão de um grande interesse de estudo de grafos aleatórios estabelecendo as bases da pesquisa na área. [Aiello et al. 2001, Bollobás et al. 2001, Mitzenmacher and Upfal 2017, Hofstad 2010].

Atualmente, os modelos propostos no estudo de redes de grande escala e que acomodam o fenômeno da lei de potência, são comumente divididos em três grupos: *grafos aleatório generalizado (GRG)*, *modelo de configuração* e *modelo de ligação preferencial*.

O *modelo de ligação preferencial*, que é o modelo utilizado neste trabalho, é inspirado em uma das hipóteses para o aparecimento do fenômeno lei de potência. A hipótese é que redes na natureza tendem a começar pequenas e crescem com o tempo seguindo um processo denominado *ligação preferencial* [Barabási and Albert 1999]. A regra define que novos vértices que vão sendo adicionados ao grafo, se conectam preferencialmente a vértices que já possuam bastantes conexões. De maneira mais precisa, neste trabalho é utilizado o modelo proposto por [Hofstad 2010] para estudar o fenômeno do aparecimento de cliques de diversos tamanhos em grafos lei de potência.

### 3. O Modelo de Ligação Preferencial

Em modelos de ligação preferencial os vértices são adicionados sequencialmente com uma ou mais arestas, proporcionando o crescimento do grafo. Essas arestas são conectadas a vértices pré-existentes no grafo com probabilidade proporcional ao grau desses vértices, favorecendo ainda mais o crescimento dos graus dos vértices que já tem grau elevado, levando o fenômeno da lei de potência a emergir.

**Definição 3.1** (Modelo de Ligação Preferencial [Hofstad 2010]) O modelo produz uma sequência de grafos não direcionados denotada por  $(PA_t^{(m,\delta)})_{t>1}$ , onde para cada tempo  $t$  é gerado um grafo com  $t$  vértices, onde cada vértice possui  $m$  arestas resultando em  $mt$  arestas no total. O parâmetro  $\delta$  do modelo controla a proeminência da lei de potência nos grafos gerados.

Inicialmente, o modelo é definido em termos de  $m = 1$ , onde os grafos gerados consistem em uma coleção de árvores. O conjunto de vértices do modelo  $PA_t^{(1,\delta)}$  é denotado como  $\{v_1^{(1)}, v_2^{(1)}, \dots, v_t^{(1)}\}$ . Seja  $D_i(t)$  o grau do vértice  $v_i^{(1)}$  no tempo  $t$ , onde um laço aumenta o grau em 2. Por definição,  $PA_1^{(1,\delta)}$  consiste em um grafo de um único vértice com um laço. A regra para se obter o grafo  $PA_{t+1}^{(1,\delta)}$  a partir de  $PA_t^{(1,\delta)}$  é descrita abaixo usando a seguinte notação:  $v_i^{(m)} \rightarrow v_j^{(m)}$  denota o surgimento de uma aresta do vértice  $v_i^{(m)}$  para o vértice  $v_j^{(m)}$ . Segue a regra define o modelo aleatório:

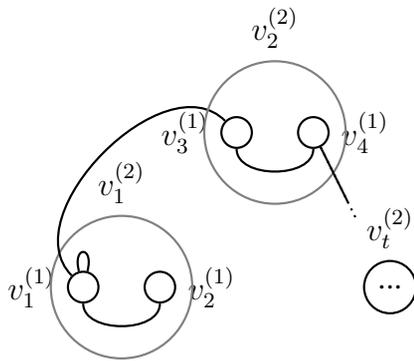
1. Adicione um vértice  $v_{t+1}$  contendo uma aresta em  $PA_{t+1}^{(1,\delta)}$ .
2. A aresta é conectada a outro vértice com probabilidade:

$$Pr(v_{t+1}^{(1)} \rightarrow v_i^{(1)}) = \begin{cases} \frac{1+\delta}{(2+\delta)t+1+\delta}, & \text{se } i = t + 1 \\ \frac{D_i(t)+\delta}{(2+\delta)t+1+\delta}, & \text{se } 0 < i < t + 1 \end{cases}, \forall \delta \geq -1$$

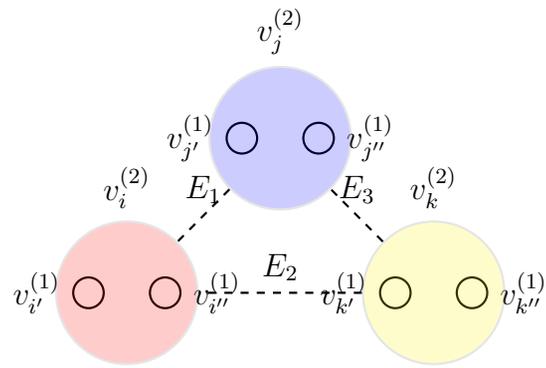
Para valores de  $m > 1$ , o modelo é definido em termos do modelo para  $m = 1$ , da seguinte forma: Fixa-se  $\delta \geq -m$ ; Amostra-se um grafo do modelo  $PA_{mt}^{(1,\delta/m)}$  e denota seus vértices como  $\{v_1^{(1)}, v_2^{(1)}, \dots, v_{mt}^{(1)}\}$ ; Colapsa-se os  $m$  primeiros vértices de  $PA_{mt}^{(1,\delta/m)}$  em um único vértice que será o primeiro vértice de  $PA_t^{(m,\delta)}$ , ou seja,  $v_1^{(m)} = \{v_1^{(1)}, v_2^{(1)}, \dots, v_m^{(1)}\}$ . Ao fazer isso, todas as arestas que são incidentes em qualquer um dos  $m$  primeiros vértices de  $PA_{mt}^{(1,\delta/m)}$  serão incidentes no novo vértice  $v_1^{(m)}$ .

A seguir colapsa-se os próximos  $m$  vértices para se tornarem  $v_2^{(m)} = \{v_{m+1}^{(1)}, v_{m+2}^{(1)}, \dots, v_{2m}^{(1)}\}$ , e assim consecutivamente até que  $v_t^{(m)} = \{v_{(t-1)m+1}^{(1)}, v_{(t-1)m+2}^{(1)}, \dots, v_{mt}^{(1)}\}$ . Ou seja, para  $m > 1$  o modelo gera um grafo em termos de  $m = 1$  e logo depois converte-o para o valor de  $m$  desejado. A Figura 1 ilustra o processo de geração do grafo para um valor de  $m = 2$ .

Para entender  $PA_t^{(m,\delta)}$  em termos de  $PA_{mt}^{(1,\delta/m)}$  é necessário observar que uma aresta em  $PA_{mt}^{(1,\delta/m)}$  é conectada a um determinado vértice  $v_k^{(1)}$  com probabilidade proporcional ao peso do vértice, onde o peso é igual ao grau do vértice somado com  $\delta/m$ . Como os vértices em  $PA_{mt}^{(1,\delta/m)}$  são colapsados para formarem os vértices de  $PA_t^{(m,\delta)}$ ,



**Figura 1.** Exemplo com  $m = 2$ . As arestas presentes na figura foram criadas no passo inicial do processo, em que  $m = 1$ .



**Figura 2.** Eventos para o aparecimento de um  $K_3$ . Os três vértices finais da clique são os coloridos com as cores vermelha, azul e amarela.

então uma aresta em  $PA_t^{(m,\delta)}$  é conectada a um determinado vértice  $v_k^{(m)}$  com probabilidade proporcional à soma dos pesos dos  $m$  vértices em  $PA_{mt}^{(1,\delta/m)}$ . Dado que a soma dos graus dos  $m$  vértices em  $PA_{mt}^{(1,\delta/m)}$  é igual ao grau do vértice  $v_k^{(m)}$  em  $PA_t^{(m,\delta)}$ , então a probabilidade da aresta conectar no vértice  $v_k^{(m)}$  é igual ao seu grau somado com  $\delta$ . É importante ressaltar que durante a geração do grafo, para valores de  $m > 1$ , os graus são atualizados após cada aresta ser adicionada ao grafo.

#### 4. Aparecimento de cliques em grafos aleatórios lei de potência

Agora é apresentado o cálculo da probabilidade do aparecimento de cliques em grafos amostrados do modelo apresentado na Seção 3. Para facilitar o entendimento, inicialmente realizamos o cálculo para cliques de tamanho 3 e, em seguida generalizamos para cliques arbitrárias.

Para que haja ciclos no grafo (e conseqüentemente cliques de tamanho maior que 2), precisa-se de  $m > 1$  e  $t > 2$ . Seja  $G$  um grafo amostrado do modelo  $PA_3^{(2,\delta)}$ . Como o modelo gera grafos inicialmente em termos de  $m = 1$ , primeiramente tem-se um grafo com  $mt$  vértices onde cada vértice liga-se com apenas um outro vértice. Logo, para que apareça uma clique, dados três vértices  $v_i^{(m)}, v_j^{(m)}, v_k^{(m)}$  distintos, é necessário que aconteçam os seguintes eventos:  $E_1 =$  pelo menos um par de vértices conectado entre  $\{v_{j'}^{(1)}, v_{j''}^{(1)}\}$  e  $\{v_{i'}^{(1)}, v_{i''}^{(1)}\}$ ;  $E_2 =$  pelo menos um par de vértices conectado entre  $\{v_{k'}^{(1)}, v_{k''}^{(1)}\}$  e  $\{v_{i'}^{(1)}, v_{i''}^{(1)}\}$ ;  $E_3 =$  pelo menos um par de vértices conectado entre  $\{v_{k'}^{(1)}, v_{k''}^{(1)}\}$  e  $\{v_{j'}^{(1)}, v_{j''}^{(1)}\}$ . A Figura 2 ilustra tais eventos.

Como mencionado anteriormente, a probabilidade de um vértice  $v_i^{(2)}$  ter uma aresta com o vértice  $v_j^{(2)}$  é proporcional à soma dos pesos desses vértices se conectarem para  $m = 1$ . Em outras palavras,  $Pr(E_1) = Pr(v_j^{(2)} \rightarrow v_i^{(2)})$  é igual a

$Pr(v_{j'}^{(1)} \rightarrow v_{i'}^{(1)}) + Pr(v_{j''}^{(1)} \rightarrow v_{i''}^{(1)}) + Pr(v_{j'''}^{(1)} \rightarrow v_{i'''}^{(1)}) + Pr(v_{j''''}^{(1)} \rightarrow v_{i''''}^{(1)})$ . O Lema a seguir generaliza a probabilidade de existir uma aresta entre um par qualquer de vértices para  $m > 1$ .

**Lema 4.1** A probabilidade de  $v_i^{(m)}$  ser adjacente a  $v_j^{(m)}$  é dada por

$$Pr(v_i^{(m)} \rightarrow v_j^{(m)}) = \sum_{l=(i-1)m+1}^{im} \sum_{k=(j-1)m+1}^{jm} \frac{D_k(l-1) + \delta}{(l-1)(2+\delta) + (1+\delta)}$$

A prova do Lema 4.1 vem da formalização do argumento esboçado nos parágrafos anteriores. A seguir o resultado principal deste trabalho:

**Teorema 4.1** Seja  $K_x \subseteq G$  o evento de uma clique de tamanho  $x$  aparecer em um grafo no modelo  $(PA_i^{(m,\delta)})_{t>1}$ . Temos que

$$Pr(K_x \subseteq G) = \sum_{\substack{S_i \in [t] \\ |S|=t-x}} \prod_{\substack{0 < j,k \\ j,k \in [t] \setminus S_i}} Pr(v_j^{(m)} \rightarrow v_k^{(m)})$$

**Prova:** Começaremos calculando a probabilidade para uma clique de tamanho 3 dados 3 vértices distintos  $v_i^{(m)}, v_j^{(m)}, v_k^{(m)}$ . Como mencionado anteriormente, para a clique aparecer é necessário que aconteçam os seguintes eventos  $Pr(E_1) \cdot Pr(E_2) \cdot Pr(E_3)$ . Utilizando o Lema 4.1, temos que a probabilidade dos 3 vértices formarem uma clique é igual a  $Pr(v_j^{(m)} \rightarrow v_i^{(m)}) \cdot Pr(v_k^{(m)} \rightarrow v_i^{(m)}) \cdot Pr(v_k^{(m)} \rightarrow v_j^{(m)})$ . Porém essa é a probabilidade para uma clique considerando somente os 3 vértices fixos. Vamos agora considerar todos os possíveis conjuntos de 3 vértices do grafo.

A equação generalizada para cliques de tamanho 3 em qualquer grafo e para qualquer conjunto de vértices é igual a  $Pr(K_3 \subseteq G) = \sum_{\substack{S_i \in [t] \\ |S|=t-3}} \prod_{\substack{0 < j,k \\ j,k \in [t] \setminus S_i}} Pr(v_j^{(m)} \rightarrow v_k^{(m)})$ ,

onde o índice do produto percorre os vértices de um determinado subconjunto de vértices, onde esse subconjunto é dado pelo somatório, que itera sobre todas as possíveis combinações de vértices do grafo. Isso nos fornece a probabilidade do aparecimento de cliques de tamanho 3. Para generalizar esta fórmula para cliques de qualquer tamanho, o tamanho do conjunto  $S$  é igual a  $|S| = t - x$ , onde  $x$  é o tamanho da clique desejada.  $\square$

### 5. Conclusões e Trabalhos Futuros

Ainda que preliminares, os resultados obtidos contribuem para analisar o surgimento de cliques de diversos tamanhos em grafos aleatórios lei de potência. Pode-se observar que é necessário aumentar muito o número de arestas de um dado grafo para possivelmente surgir uma clique, dado que a probabilidade vai decrescendo a medida que o tamanho da clique aumenta. Na opinião dos autores, isso deve ajudar no desempenho de algoritmos de otimização em tais grafos lei de potência. Para confirmar ou refutar esta hipótese, o passo seguinte neste projeto será obter resultados analíticos a respeito do tamanho da maior clique do grafo e realizar experimentos com o problema da clique máxima.

## Referências

- Aiello, W., Chung, F., and Lu, L. (2001). A random graph model for power law graphs. *Experimental Mathematics*, 10(1):53–66.
- Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286(5439):509–512.
- Bollobás, B., Riordan, O., Spencer, J., and Tusnády, G. (2001). The degree sequence of a scale-free random graph process. *Random Structures & Algorithms*, 18(3):279–290.
- Erdős, P. and Rényi, A. (1960). On the evolution of random graphs. *Publ. Math. Inst. Hung. Acad. Sci.*, 5(1):17–60.
- Eubank, S., Kumar, V., Marathe, M. V., Srinivasan, A., and Wang, N. (2004). Structural and algorithmic aspects of massive social networks. *Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 718–727.
- Faloutsos, M., Faloutsos, P., and Faloutsos, C. (1999). On power-law relationships of the internet topology. *ACM SIGCOMM computer communication review*, 29(4):251–262.
- Gast, M. and Hauptmann, M. (2014). Approximability of the vertex cover problem in power-law graphs. *Theoretical Computer Science*, 516:60–70.
- Gast, M., Hauptmann, M., and Karpinski, M. (2015). Inapproximability of dominating set on power law graphs. *Theoretical Computer Science*, 562:436–452.
- Guelzim, N., Bottani, S., Bourguine, P., and Képès, F. (2002). Topological and causal structure of the yeast transcriptional regulatory network. *Nature genetics*, 31(1):60.
- Hofstad, R. V. D. (2010). *Random Graphs and Complex Networks*, pages 543–555. Cambridge University Press.
- Mitzenmacher, M. and Upfal, E. (2017). *Probability and Computing Randomization and Probabilistic Techniques in Algorithms and Data Analysis*. Cambridge University Press.
- Park, K. and Lee, H. (2001). On the effectiveness of route-based packet filtering for distributed dos attack prevention in power-law internets. 31(4):15–26.
- Pinto, G. P. and da Silva, L. A. (2017). Algoritmos de clique máximo em redes complexas. Technical report, Universidade Tecnológica Federal do Paraná.
- Rogiski, R. (2016). Desempenho do algoritmo guloso na coloração de vértices em grafos de sistemas complexos. B.S. thesis, Universidade Tecnológica Federal do Paraná.
- Shen, Y., Nguyen, D. T., Xuan, Y., and Thai, M. T. (2012). New techniques for approximating optimal substructure problems in power-law graphs. *Theoretical Computer Science*, 447:107–119.
- Silva, M. O. D., Gimenez-Lugo, G. A., and Silva, M. V. D. (2013). Vertex cover in complex networks. *International Journal of Modern Physics C*, 24(11):1350078.
- Vignatti, A. L. and da Silva, M. V. G. (2016). Minimum vertex cover in generalized random graphs with power law degree distribution. *Theoretical Computer Science*, 647(1):101–111.

# Criticalidade arco-íris em rodas e leques

Aleffer Rocha<sup>1</sup>, Sheila Morais de Almeida<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná (UTFPR)  
Av Monteiro Lobato, s/n - Km 04, CEP 84016-210 - Ponta Grossa - PR – Brasil

aleffer@alunos.utfpr.edu.br, sheilaalmeida@utfpr.edu.br

**Resumo.** *Uma coloração arco-íris de um grafo conexo  $G$  é uma coloração de arestas, não necessariamente própria, tal que entre qualquer par de vértices de  $G$  existe um caminho cujas cores das arestas são duas a duas distintas. O número de conexão arco-íris de um grafo  $G$ , denotado por  $rc(G)$ , é o menor número de cores necessárias para se obter uma coloração arco-íris de  $G$ . Um grafo  $G$  é arco-íris crítico se a remoção de uma aresta qualquer de  $G$  aumenta o seu número de conexão arco-íris. Nesse trabalho, provamos que grafos rodas e leques são arco-íris críticos somente quando esses grafos tem poucos vértices.*

**Abstract.** *A rainbow coloring of a connected graph  $G$  is an edge coloring that is not necessarily proper such that there is a path between any pair of vertices of  $G$  whose edge colors are pairwise distinct. The rainbow connection number of a graph  $G$ , denoted by  $rc(G)$ , is the least number of colors for which there is a rainbow coloring of  $G$ . A graph  $G$  is rainbow critical if its rainbow connection number increases when we remove any edge from  $G$ . In this work, we prove wheel and fan graphs are rainbow critical only when these graphs have few vertices.*

**Palavras-chave:** Grafos arco-íris críticos, grafos rodas, grafos leques

**Keywords:** Rainbow critical graphs; wheel graphs; fan graphs

## 1. Introdução

Uma *coloração de arestas* em um grafo  $G$  é uma atribuição de cores para as arestas de  $G$ . Neste trabalho, representamos as cores através de números inteiros e os grafos considerados são simples e conexos. Os conjuntos de vértices e arestas de qualquer grafo  $G$  serão denotados respectivamente por  $V(G)$  e  $E(G)$ . Uma coloração é própria se as arestas incidentes em um mesmo vértice possuem cores distintas. Dado um grafo  $G$  com uma coloração de arestas não necessariamente própria, um caminho entre um par de vértices  $u$  e  $v$  em  $G$  é arco-íris se as cores de quaisquer duas arestas do caminho são distintas. Uma coloração de arestas de  $G$  é uma coloração arco-íris se entre qualquer par de vértices de  $G$  existe um caminho arco-íris [Chartrand et al. 2008]. O *número de conexão arco-íris* de um grafo  $G$  é o menor inteiro  $k$  possível para se obter uma coloração

arco-íris em  $G$  com  $k$  cores e é denotado por  $rc(G)$ . O Problema da Coloração Arco-Íris consiste em determinar  $rc(G)$  para um grafo  $G$  simples e conexo qualquer. É importante ressaltar que se  $G$  é desconexo então existe pelo menos um par de vértices,  $u$  e  $v$ , entre os quais não existe caminho. Portanto, não há cores suficientes para se ter um caminho arco-íris entre  $u$  e  $v$ , e, conseqüentemente podemos concluir que o número de conexão arco-íris de  $G$  é infinito. O Problema da Coloração Arco-íris é de interesse da Agência de Segurança Nacional (em inglês: *National Security Agency - NSA*), que tem financiado projetos sobre o tema [Lo 2012, Dudek et al. 2015]. Mais detalhes sobre essa aplicação podem ser vistos em [Li and Sun 2012].

Para um  $k$  fixo,  $k \geq 2$ , determinar se o  $rc(G) \leq k$  é NP-completo [Chakraborty et al. 2011, Ananth et al. 2011, Li and Li 2011]. Considerando a dificuldade em se calcular o  $rc(G)$ , diversos autores empreenderam esforços na busca por limitantes inferiores e superiores para o número de conexão arco-íris [Chakraborty et al. 2011, Chandran and Rajendraprasad 2012, Schiermeyer 2009].

Um grafo  $G$  é arco-íris crítico se, ao remover uma aresta qualquer de  $G$ , o número de conexão arco-íris de  $G$  aumenta [Rao and Murali 2014]. Em [Rao and Murali 2014] os autores também mostram que os grafos resultantes do produto cartesiano de dois caminhos,  $P_m$  e  $P_n$ , são arco-íris críticos quando  $m, n \geq 2$  e que o produto cartesiano de ciclos pares com caminhos,  $C_m$  e  $P_n$ , são arco-íris críticos quando  $m \geq 4$  par e  $n \geq 2$ . Esses resultados foram refutados em [Rocha and Almeida 2017], ficando provado que o produto cartesiano de dois caminhos é arco-íris crítico se, e somente se,  $m = 1$  ou  $m = n = 2$  e que o produto cartesiano de ciclos com caminhos não é arco-íris crítico. Este trabalho apresenta resultados sobre a criticalidade arco-íris dos grafos rodas e leques.

## 2. Arcabouço teórico

Um grafo  $H$  é *subgrafo* de um grafo  $G$  se  $V(H) \subseteq V(G)$  e  $E(H) \subseteq E(G)$ . Um subgrafo  $H$  de um grafo  $G$  é *gerador* se  $V(H) = V(G)$ . Uma *árvore geradora* de um grafo  $G$  é subgrafo gerador de  $G$  conexo e sem ciclos. Chartrand et. al [Chartrand et al. 2008] apresentam os primeiros limitantes superiores para o número de conexão arco-íris, explorando a quantidade de arestas presentes nas árvores geradoras do grafo. Como as árvores são grafos conexos, possuem caminho entre quaisquer pares de vértices. Além disso, qualquer subgrafo conexo de um grafo  $G$  que contenha  $|V(G)|$  vértices tem pelo menos tantas arestas quanto uma árvore geradora de  $G$ . Então, para obter uma coloração arco-íris de qualquer grafo simples e conexo, é suficiente atribuir cores distintas para todas as arestas de uma das suas árvores geradoras. Sabe-se que qualquer árvore com  $n$  vértices tem exatamente  $n - 1$  arestas. Então, uma coloração arco-íris de uma árvore geradora de um grafo  $G$  usa  $|V(G)| - 1$  cores. Portanto,  $rc(G) \leq |V(G)| - 1$ , para qualquer grafo conexo  $G$ .

A *distância* entre dois vértices  $u$  e  $v$  em um grafo  $G$  é o número de arestas de um caminho mínimo entre  $u$  e  $v$ . A *excentricidade* de um vértice  $v$  é a maior das distâncias entre  $v$  e os demais vértices do grafo  $G$ . O *diâmetro* de  $G$ , denotado por  $diam(G)$ , é a maior excentricidade de um vértice de  $G$ . Assim como a árvore geradora provê um limitante superior para o  $rc(G)$ , o diâmetro do grafo provê um limitante

inferior [Chartrand et al. 2008]. Note que se existe um par de vértices em  $G$  entre os quais não há caminho de tamanho menor que  $diam(G)$ , não pode haver uma coloração arco-íris para  $G$  com menos que  $diam(G)$  cores. Logo,  $rc(G) \geq diam(G)$ , para qualquer grafo  $G$ . Pode-se concluir então que, se  $G$  é um grafo conexo e não trivial com  $|V(G)|$  vértices, então  $diam(G) \leq rc(G) \leq |V(G)| - 1$ .

Dados dois grafos  $G$  e  $H$ , o grafo junção  $G + H$  tem conjunto de vértices  $V(G + H) = V(G) \cup V(H)$  e conjunto de arestas  $E(G + H) = \{vu : v \in V(G) \text{ e } u \in V(H)\} \cup E(G) \cup E(H)$ .

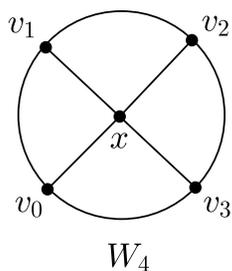
Os primeiros resultados sobre coloração arco-íris foram publicados em [Chartrand et al. 2008] e apresentam o número de conexão arco-íris de árvores, ciclos, grafos completos, rodas, dentre outros. Para um grafo  $G$  com  $n$  vértices,  $rc(G) = n - 1$  se, e somente se,  $G$  é uma árvore. O ciclo  $C_n$  tem  $rc(C_n) = \lceil \frac{n}{2} \rceil$  e o grafo completo  $K_n$  tem  $rc(K_n) = 1$ . Como o número de conexão arco-íris de qualquer grafo é limitado inferiormente por seu diâmetro, não existem grafos além dos completos com número de conexão arco-íris igual a 1.

Nesse trabalho, vamos tratar da criticalidade dos grafos rodas e leques. Um grafo roda,  $W_n$ , consiste da junção de um ciclo  $C_n$  com um grafo completo  $K_1$ , como mostra a Figura 1a. O Teorema 1 apresenta o número de conexão arco-íris dos grafos roda.

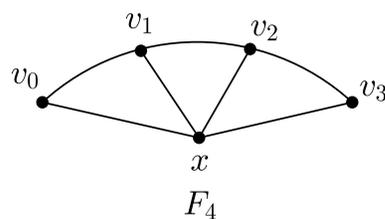
**Teorema 1.** [Chartrand et al. 2008] *Seja  $W_n$  um grafo roda com  $n \geq 3$ .*

$$rc(W_n) = \begin{cases} 1, & \text{se } n = 3, \\ 2, & \text{se } 4 \leq n \leq 6, \\ 3, & \text{se } n \geq 7. \end{cases}$$

Considere o grafo roda  $W_n = C_n + K_1$  e sejam os vértices de  $C_n$  rotulados  $v_0, v_1, v_2, \dots, v_{n-1}$  tal que  $v_i$  é adjacente a  $v_{(i+1) \bmod n}$ ,  $0 \leq i \leq n - 1$ . O grafo leque  $F_n$  é obtido pela remoção da aresta  $v_0v_{n-1}$  do grafo  $W_n$ , como apresenta a Figura 1b. O Teorema 2 apresenta o número de conexão arco-íris dos grafos leque.



(a) Um grafo roda  $W_4$ .



(b) Um grafo leque  $F_4$ .

**Teorema 2.** [Sy et al. 2013] *Seja  $F_n$  um grafo leque.*

$$rc(F_n) = \begin{cases} 1, & \text{se } n = 2, \\ 2, & \text{se } 3 \leq n \leq 6, \\ 3, & \text{se } n \geq 7. \end{cases}$$

Para provar que um grafo  $G$  não é arco-íris crítico, é suficiente mostrar que  $rc(G - e) = rc(G)$ , para alguma aresta  $e$  em  $G$ . Por outro lado, para provar que um grafo  $G$  é crítico, é necessário mostrar que a remoção de qualquer aresta do grafo  $G$  aumenta o seu número de conexão arco-íris. Na próxima seção é feita a discussão sobre a criticalidade das classes roda e leque.

### 3. Resultados

Os resultados desta seção estão divididos em dois teoremas. O primeiro mostra que a roda  $W_n$  é arco-íris crítica se e somente se  $n = 3$ . O segundo mostra que o leque  $F_n$  é arco-íris crítico se e somente se  $n = 2$ .

**Teorema 3.** *O grafo  $W_n$  é arco-íris crítico se e somente se  $n = 3$ .*

*Demonstração.* Primeiro observe que o menor grafo roda  $W_n$  tem  $n = 3$ . Além disso,  $W_3$  é isomorfo ao grafo completo  $K_4$ . Como  $rc(G) = 1$  se e só se  $G$  é completo, a remoção de uma aresta de  $W_3$  aumenta o seu número de conexão arco-íris. Logo,  $W_3$  é arco-íris crítico.

Para os demais casos, considere  $W_n = C_n + K_1$  e rotule os vértices de  $C_n$ :  $v_0, v_1, \dots, v_{n-1}$  tal que  $v_i$  é adjacente a  $v_{(i+1) \bmod n}$ ,  $0 \leq i \leq n - 1$ . Seja  $e = v_{n-1}v_0$ . Observe que  $W_n - e$  é isomorfo ao leque  $F_n$ . Portanto,  $rc(W_n - e) = rc(F_n)$ . Quando  $n \geq 4$ , pelos Teoremas 1 e 2, tem-se  $rc(F_n) = rc(W_n)$ . Portanto,  $rc(W_n - e) = rc(W_n)$  quando  $n \geq 4$  e conclui-se que  $W_n$  não é arco-íris crítico.  $\square$

**Teorema 4.** *O grafo  $F_n$  é arco-íris crítico se e somente se  $n = 2$ .*

*Demonstração.* Observe que  $F_n$  é isomorfo ao grafo junção  $P_n + K_1$ . Considere um grafo leque  $F_n$  e rotule os vértices de  $P_n$ :  $v_0, v_1, v_2, \dots, v_{n-1}$  tal que  $v_i$  é adjacente a  $v_{i+1}$ ,  $0 \leq i < n - 1$ . Seja  $x$  o vértice do grafo  $K_1$ .

O grafo  $F_2$  é isomorfo ao grafo  $K_3$  e, portanto,  $rc(F_2) = 1$ . Seja  $e$  uma aresta qualquer de  $F_2$ . Então, o grafo  $F_2 - e$  é uma árvore com três vértices e, pelos resultados apresentados em [Chartrand et al. 2008], tem  $rc(F_2 - e) = 2$ . Logo,  $F_2$  é um grafo arco-íris crítico.

Sejam  $n = 3$  e  $e = v_1x$ . Então,  $F_3 - e$  é isomorfo ao grafo  $C_4$ . Pelos resultados apresentados em [Chartrand et al. 2008],  $rc(C_4) = 2$ . Pelo Teorema 2,  $rc(F_3) = 2$ . Logo,  $rc(F_3 - e) = rc(F_3)$  e o grafo  $F_3$  não é arco-íris crítico.

Considere o caso  $4 \leq n \leq 6$ . Pelo Teorema 2,  $rc(F_n) = 2$ . É suficiente mostrar que existe uma aresta em  $F_n$  cuja remoção não aumenta o seu número de conexão arco-íris. Seja  $e = v_{\lfloor \frac{n}{2} \rfloor - 1}v_{\lfloor \frac{n}{2} \rfloor}$ . O grafo  $F_n - e$  é a união de dois subgrafos leques, um isomorfo ao  $F_{\lfloor \frac{n}{2} \rfloor}$  e outro isomorfo ao  $F_{\lceil \frac{n}{2} \rceil}$ , com apenas o vértice  $x$  em comum. Uma coloração arco-íris utilizando 2 cores para estes grafos é apresentada na Figura 2. Portanto, quando  $4 \leq n \leq 6$ , tem-se  $rc(F_n - e) = rc(F_n) = 2$  e  $F_n$  não é arco-íris crítico.

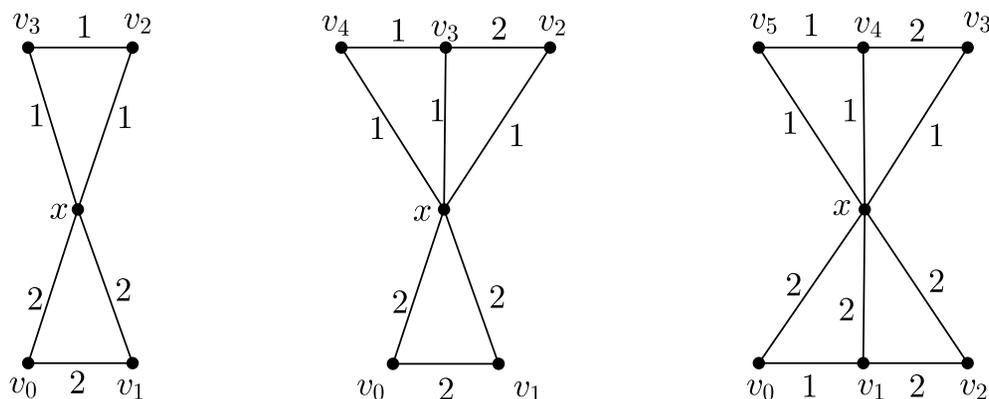


Figura 2. Leques  $F_4, F_5$  e  $F_6$  não arco-íris críticos.

O grafo  $F_7$  tem  $rc(F_7) = 3$ , pelo Teorema 2. Seja  $e = v_0v_1$ . Note que  $F_7 - e$  tem como subgrafo induzido um grafo  $F_6$ , onde  $V(F_6) = \{v_1, v_2, \dots, v_6, x\}$ . Sabemos que  $F_6$  tem uma coloração arco-íris com 2 cores, pelo Teorema 2. Como  $v_0$  é um vértice pendente em  $F_7 - e$ , é necessário atribuir uma nova cor para a aresta  $v_0x$  de forma que se obtenha uma coloração arco-íris para  $F_7 - e$ . Logo,  $rc(F_7 - e) = rc(F_7) = 3$ . Portanto  $F_7$  não é arco-íris crítico.

Resta considerar os casos em que  $F_n$  tem  $n \geq 8$ . Pelo Teorema 2, sabemos que  $rc(F_n) = 3$  nestes casos. Seja  $e = v_3v_4$ . Vamos mostrar que  $rc(F_n - e) = 3$ . Para  $0 \leq i < n$ , pinte a aresta  $v_i v_{i+1}$  com cor 1 quando  $i$  é par e com cor 3 quando  $i$  é ímpar. Para  $0 \leq i < n$ , pinte a aresta  $v_i x$  com cor 3 se  $i$  é ímpar e com cor 2 se  $i$  é par. Note que existe um caminho de tamanho 1 entre  $x$  e qualquer outro vértice. Se  $i$  é ímpar e  $j$  é par, existe um caminho arco-íris  $v_i x v_j$ . Se  $i$  e  $j$  são ímpares, então o caminho arco-íris é  $v_i v_{i-1} x v_j$ . Se  $i$  e  $j$  são pares, então o caminho arco-íris é  $v_i v_{i+1} x v_j$ . Já que existe caminho arco-íris entre todo par de vértices,  $rc(F_n - e) = rc(F_n) = 3$ . Portanto,  $F_n$  não é crítico.  $\square$

#### 4. Conclusão

É interessante observar que, mesmo os ciclos e os caminhos sendo grafos arco-íris críticos, os grafos  $P_n + K_1$  com  $n \geq 3$  e  $C_n + K_1$  com  $n \geq 4$  não são arco-íris críticos. Levantamos a seguinte questão: Seja  $G$  um grafo simples, conexo com  $n$  vértices,  $n \geq 3$ . Quando o grafo  $G + K_1$  é arco-íris crítico? Sabe-se que se  $G$  é completo, então  $G + K_1$  é arco-íris crítico. Um caminho hamiltoniano em um grafo  $G$  é um caminho que contém todos os vértices de  $G$ . Note que se  $G$  tem um caminho hamiltoniano, então  $G + K_1$  tem um leque  $F_{|V(G)|}$  como subgrafo e  $rc(G + K_1) \leq 3$ . Pode-se concluir que se  $G$  tem um caminho hamiltoniano e  $rc(G + K_1) = 3$ , então  $|V(G)| \geq 7$  e  $G + K_1$  não é arco-íris crítico pelo Teorema 4. Se  $G$  tem caminho hamiltoniano e  $rc(G) = 2$ , então  $rc(G + K_1) = 2$ , basta manter a mesma coloração arco-íris nas arestas do grafo  $G$  e reutilizar uma das duas cores para colorir as arestas da junção. Suponha que  $G$  tem uma coloração arco-íris com as cores 1 e 2 e que as arestas da junção foram coloridas com cor 1. Sejam  $v_1$  e  $v_2$  dois vértices consecutivos em um caminho hamiltoniano em  $G$  e  $x$  o vértice do grafo  $K_1$ . Ao

remover a aresta  $v_1x$ ,  $G + K_1$  ainda tem uma coloração arco-íris com duas cores. Observe que é suficiente garantir um caminho arco-íris entre  $v_1$  e  $x$  já que os caminhos arco-íris entre os demais pares de vértices estão preservados. Para tanto, se a aresta  $v_1v_2$  estiver colorida com cor 1, pinte  $v_2x$  com cor 2 e vice-versa. Portanto  $G + K_1$  não é arco-íris crítico.

Como trabalhos futuros sugere-se o estudo da criticalidade arco-íris de grafos junção  $G + K_1$  nos casos em que  $G$  tem um caminho hamiltoniano (ou seja,  $G + K_1$  tem um leque como subgrafo),  $rc(G) = 3$  e  $rc(G + K_1) = 2$ .

## Referências

- Ananth, P., Nasre, M., and Sarpatwar, K. K. (2011). Rainbow connectivity: Hardness and tractability. In Chakraborty, S. and Kumar, A., editors, *IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2011)*, volume 13 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 241–251, Dagstuhl, Germany. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- Chakraborty, S., Fischer, E., Matsliah, A., and Yuster, R. (2011). Hardness and algorithms for rainbow connectivity. *Journal of Combinatorial Optimization*, 21:330–347.
- Chandran, L. S. and Rajendraprasad, D. (2012). Rainbow colouring of split and threshold graphs. In *International Computing and Combinatorics Conference*, pages 181–192. Springer.
- Chartrand, G., Johns, G. L., McKeon, K. A., and Zhang, P. (2008). Rainbow connection in graphs. *Mathematica Bohemica*, 133(1):85–98.
- Dudek, A., Frieze, A. M., and Tsourakakis, C. E. (2015). Rainbow connection of random regular graphs. *SIAM Journal on Discrete Mathematics*, 29(4):2255–2266.
- Li, S. and Li, X. (2011). Note on the complexity of deciding the rainbow connectedness for bipartite graphs. *arXiv preprint arXiv:1109.5534*.
- Li, X. and Sun, Y. (2012). *Rainbow connections of graphs*. Springer Science & Business Media.
- Lo, I. Y. (2012). Some bounds on the rainbow connection number of 3-, 4- and 5-connected graphs. *arXiv preprint arXiv:1212.5934*.
- Rao, K. S. and Murali, R. (2014). Rainbow critical graphs. *International Journal of Computer Application*, 4(4):252–259.
- Rocha, A. and Almeida, S. M. (2017). Criticalidade arco-íris dos grafos resultantes de produto cartesiano de ciclos e caminhos. In *Anais do II Workshop de Pesquisa em Computação dos Campos Gerais*, Ponta Grossa, PR.
- Schiermeyer, I. (2009). Rainbow connection in graphs with minimum degree three. In *International Workshop on Combinatorial Algorithms*, pages 432–437. Springer.
- Sy, S., Medika, G. H., and Yulianti, L. (2013). The rainbow connection of fan and sun. *Applied Mathematical Sciences*, 7(64):3155–3159.

# Planejamento de caminhos 3D: Comparação dos algoritmos por grade 3D e grafo de visibilidade \*

Cleiton A. Santos <sup>1</sup>, André L. P. Guedes <sup>1</sup>

<sup>1</sup>Departamento de Informática – Universidade Federal do Paraná (UFPR)  
Centro politécnico – Curitiba – PR – Brazil

{casantos, andre}@inf.ufpr.br

**Resumo.** Este artigo descreve o andamento de uma pesquisa focada em planejamento de caminhos 3D. O objetivo da pesquisa é encontrar algoritmos de baixa complexidade computacional, capazes de gerar um caminho próximo do ótimo em um ambiente controlado. Os algoritmos em estudo possuem como entrada a descrição do ambiente, e como saída o melhor caminho entre os pontos de origem e de destino. As abordagens por visibilidade e por grade são comparadas em ambientes simples. Também são propostas melhorias.

**Abstract.** This paper describes the progress of a research focused on 3D path planning. The goal of the research is to find algorithms of low computational complexity capable of generating a near-optimal path in a controlled environment. The algorithms under study have as input the description of the environment, and as output, the best path between the source point and the target point. We compare the approaches by visibility and by grid in simple environments. We also propose improvements.

**Palavras-chave:** Planejamento de caminhos 3D; Grafo de visibilidade 3D; Grade 3D.

## 1. Introdução

Para atender o recente aumento da demanda por aplicações computacionais relacionadas a ambientes tridimensionais, tais como: veículos autônomos, drones, jogos digitais, simulação de multidões, robótica, impressoras 3D, entre outros são necessários algoritmos eficientes. A pesquisa em andamento, em planejamento de caminhos, busca criar e refinar algoritmos para suporte à movimentação de agentes em um ambiente 3D. Entenda-se por agente um personagem, uma ferramenta, um veículo ou algum objeto, que precisa se movimentar com segurança no ambiente escolhido.

Em um ambiente  $W$  com obstáculos convexos  $W_o$  e espaço livre  $W_f = W \setminus W_o$  dados os pontos origem  $s$  e destino  $t$ , contidos no espaço livre ( $s, t \in W_f$ ), uma curva  $\tilde{st}$ , também contida no espaço livre ( $\tilde{st} \in W_f$ ) é chamada de *caminho*. Em  $W$  podem existir vários caminhos de  $s$  a  $t$ . O melhor caminho, aquele de menor custo, é chamado de *caminho ótimo* e denotado por  $\tilde{st}^*$ . *Planejamento de caminhos* é o processo para encontrar os caminhos e identificar o caminho ótimo. Em um ambiente 3D o conjunto de caminhos

\*Com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES).

é infinito e não computável. Achar o caminho ótimo é provado por [Canny 1988] ser NP-Difícil. Portanto, é adotada uma variação do planejamento de caminhos que encontra alguns caminhos e identifica o *melhor caminho discretizado* que é o mais próximo do caminho ótimo que a representação do ambiente permite alcançar em tempo polinomial.

A pesquisa em andamento tem por objetivo encontrar algoritmos para planejamento de caminhos em um ambiente 3D controlado, que gerem caminhos próximos do ótimo e com baixa complexidade computacional. Em [Yang et al. 2016] são apresentadas abordagens de algoritmos para planejamento de caminho. Os algoritmos de interesse precisam atender aos requisitos de obter os melhores resultados em tempo de execução previsível e o algoritmo precisa ser aplicável a diferentes cenários. Por não atender estes requisitos, as abordagens bioinspiradas (ACO, GA, PSO, etc.), de exploração aleatória (PRM, RRT, etc.), baseadas em modelos matemáticos (MILP, BIP) e outras como campos potenciais artificiais (APF) foram descartadas.

A fase inicial da pesquisa descrita em [Santos e Guedes 2019] consistiu em identificar as abordagens que poderiam ser empregadas, escolher algumas consideradas promissoras, implementar e avaliar os resultados. O critério de comparação é a qualidade dos caminhos gerados, ou seja, caminhos de menor custo. Foi objetivo na implementação inicial avaliar a qualidade das saídas dos algoritmos. Embora o potencial de reduzir a complexidade computacional dos métodos tenha sido relevante na escolha. As abordagens trabalhadas inicialmente foram algoritmos baseados em grade e visibilidade. Para eficiente busca de caminho mínimo em grafos, tanto densos como esparsos, foram implementados os métodos *Dijkstra* [Dijkstra, 1959] e *Floyd-Warshall* [Ingerman, 1962].

O presente artigo tem por objetivo apresentar as análises e resultados preliminares, bem como, descrever o que será objeto de estudo a seguir. Na Seção 2 são detalhados os algoritmos implementados. A Seção 3 é apresenta uma comparação dos resultados gerados pelos algoritmos. Na Seção 4 são descritas possíveis melhorias nos algoritmos. Finalmente na Seção 5 estão as considerações finais.

## 2. Os algoritmos implementados

Os algoritmos de planejamento de caminhos baseados em grade e visibilidade, são conhecidos e aplicados em ambientes 2D [Andayesh e Sadeghpour 2014, Berg et al. 2008]. Na presente pesquisa, é considerada a expansão para ambientes 3D. Os algoritmos implementados são detalhados a seguir.

### 2.1. Algoritmos baseados em grade 3D

A ideia base dos algoritmos por grade 3D é particionar o ambiente em uma grade uniforme de pequenos volumes cúbicos, também chamados de *voxels*. A primeira etapa do algoritmo é marcar cada elemento da grade como livre ou ocupado. São considerados *voxels* ocupados aqueles que pelo menos um obstáculo ocupa pelo menos uma parte do seu volume. Apenas os cubos livres interessam. A partir dos livres é criado um grafo onde cada *voxel* é representado por um vértice no grafo. As arestas do grafo são baseadas na vizinhança de cada cubo. Os critérios de vizinhança podem ser faces, arestas ou vértices em comum entre dois *voxels*. Ao grafo gerado, a partir da grade, são adicionados

dois vértices e duas arestas. Os vértices adicionados representam os pontos de origem  $s$  e destino  $t$ , e as arestas adicionadas representam a ligação dos novos vértices com os representantes dos *voxels* que contém  $s$  e  $t$ . Com o grafo gerado, são utilizados algoritmos de busca em grafos para encontrar o melhor caminho. Nas Figuras 1 e 2 podem ser visualizadas as etapas do algoritmo baseado em grade para obstáculos com formatos de cubo e esfera<sup>1</sup> respectivamente, na Figura 1 o espaço foi subdividido em 27 *voxels* e na Figura 2 o espaço foi subdividido em 729 *voxels*. É intencional o cubo possa ser inscrito na esfera, estejam na mesma posição e ocupem os mesmos *voxels*. Se trocar o cubo pela esfera na Figura 1 ou a esfera pelo cubo na Figura 2 os caminhos gerados respectivamente pelo algoritmo de grade serão o mesmo.

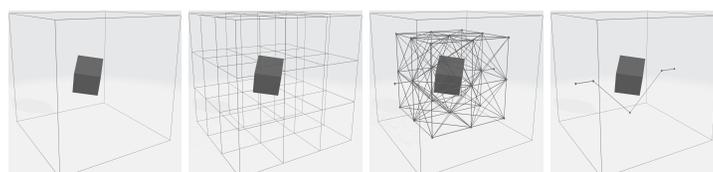


Figura 1. Grade 3D - Cubo - 27 *voxels*

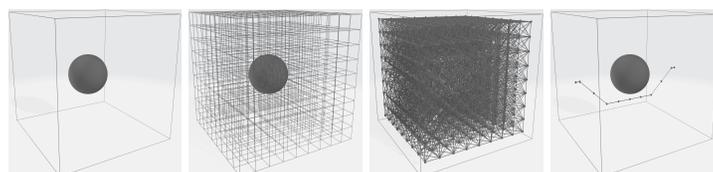


Figura 2. Grade 3d - Esfera - 729 *voxels*

## 2.2. Algoritmos baseados em visibilidade

A abordagem dos algoritmos por grafo de visibilidade consiste em construir um grafo a partir dos vértices dos obstáculos, sendo cada um representado por um vértice no grafo. São acrescentados ao grafo, também, 2 vértices para representar os pontos de origem  $s$  e destino  $t$ . Se dois vértices do grafo são visíveis entre si é criada uma aresta entre eles no grafo. Ser visível significa que um segmento de reta entre o par de vértices não atravessa obstáculos. Uma vez gerado o grafo, são aplicados algoritmos de busca de caminhos em grafos para encontrar o melhor caminho. Nas Figuras 3 e 4 podem ser visualizadas as etapas do algoritmo baseado em visibilidade para cubo e esfera respectivamente.

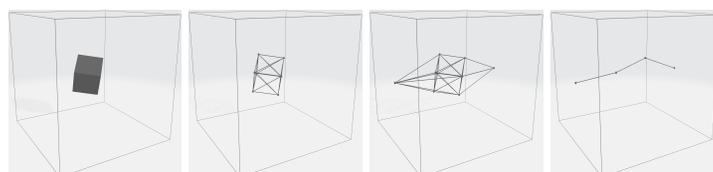


Figura 3. Grafo de visibilidade - Cubo

<sup>1</sup>O termo *esfera* se refere a um poliedro regular com muitas faces e aparência de uma esfera.

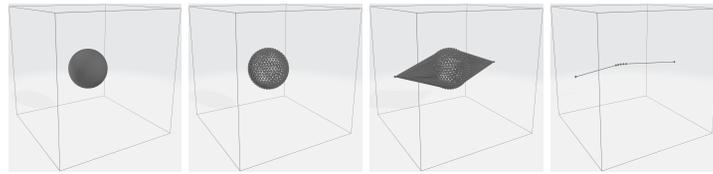


Figura 4. Grafo de visibilidade - Esfera

### 3. Comparação dos métodos

A abordagem por grade 3D, sofre de problema semelhante a “maldição de dimensionalidade”. Quanto menores os *voxels* melhor será caminho obtido. Mas reduzir os *voxels* provoca aumento no volume de dados. O caminho gerado por grade possui muitas arestas e aparência serrilhada. Por outro lado, se um obstáculo for substituído por outro mas ambos ocupam os mesmos *voxels*, não importando a complexidade deles, o caminhos gerado, pelo algoritmo por grade 3D, será o mesmo. Já a abordagem por grafo de visibilidade se baseia nos vértices dos obstáculos e sofre degradação de desempenho quando os obstáculos são poliedros complexos. O caminho gerado por visibilidade costuma ser menor em custo, com arestas mais longas e em menor quantidade se comparado com o caminho gerado a partir da Grade 3D.

Tabela 1. Comparação entre as abordagens. Grade 3 x 3 x 3 (27 voxels, com 26 livres) e 9 x 9 x 9 (729 voxels, com 702 livres); Visibilidade Cubo (8 vértices, 18 arestas, 12 faces) e Esfera (642 vértices, 1.920 arestas, 1.280 faces)

		Grade		Visibilidade Cubo		Visibilidade poliedro	
		3x3x3	9x9x9	Vértice	Aresta	Vértice	Aresta
<b>Grafo</b>	<b>Vértices</b>	28	704	10	20	644	1.922
	<b>Arestas</b>	268	13.812	68	148	4.768	10.264
<b>Caminho</b>	<b>Vértices</b>	5	11	4	3	7	12
	<b>Arestas</b>	8	20	6	4	12	22
	<b>Tamanho</b>	1.108,5	1.025,7	906,3	895,4	912,8	912,6

A Tabela 1 compara os resultados obtidos. Os obstáculos são um cubo e uma esfera. As colunas a esquerda representam experimentos com grade, e a direita por visibilidade. Por grade o ambiente de dimensão 900 x 900 x 900, foi dividido nos experimentos em 27 *voxels* de 300 x 300 x 300, e em 729 *voxels* de 100 x 100 x 100. Os obstáculos estão posicionados no centro do ambiente e o diâmetro de ambos é 300, ocupando um *voxel* e 27 *voxels* respectivamente. Por visibilidade as colunas da tabela, foram subdivididas em visibilidade por vértices (ver. Subseção 2.2) e por arestas (ver Subseção 4.2). Na tabela as linhas superiores apresentam as dimensões do grafo que representa o ambiente, e as linhas inferiores apresentam o melhor caminho obtido entre *s* e *t*. Por grade, os vértices do grafo representam os *voxels* livres e as arestas do grafo representam a vizinhança dos *voxels*. Por visibilidade, os vértices do grafo representam pontos do ambiente, que são, os vértices ou as arestas dos obstáculos e as arestas do grafo indicam a visibilidade entre os pares de vértices. O grafo é orientado, portanto, existem duas arestas para cada par de vértices conectado. Ao grafo são acrescentados dois vértices extra para representar *s* e *t*.

Por grade são acrescentadas quatro arestas extra para a ligação de  $s$  e  $t$  com os *voxels* que os contém. Por visibilidade são adicionadas as arestas extra para a ligação dos vértices extra aos demais vértices visíveis. Na grade a relação de vizinhança adotada é 26 (faces, arestas e vértices) e o grau máximo é 26. O grau máximo por visibilidade é, no pior caso, o número de vértices menos um. O melhor caminho é indicado pelos os vértices, arestas e o custo entre  $s$  e  $t$ .

## 4. Melhorias nos algoritmos

Em ambas as abordagens, existem formas de melhorar tanto os resultados como o desempenho dos algoritmos. Nas subseções a seguir são abordadas algumas alternativas.

### 4.1. Otimização de algoritmo baseado em grade 3D

Para otimizar a abordagem por grade é necessário buscar formas de reduzir o número de vértices do grafo mantendo os mais significativos. Uma maneira de obter este efeito seria utilizar *voxels* de tamanho variável. Um critério de redução seria adotar perto de obstáculos *voxels* pequenos e longe de obstáculos *voxels* grandes. Para se obter esta variação de tamanho é possível utilizar uma *octree*, que é uma estrutura de dados em forma de árvore, em que cada nó-pai possui 8 nós-filhos, que representam a subdivisão de um volume maior em 8 volumes menores [Meagher 1982]. Para reduzir o efeito serrihado, seria necessário fazer um pós-processamento para suavizar e simplificar o caminho, eliminando vértices e unindo arestas adjacentes. Para 3 vértices consecutivos no caminho gerado, verificar se é possível ligar diretamente, por uma aresta, o primeiro vértice ao último vértice, eliminando o vértice do meio.

### 4.2. Otimização de algoritmo baseado em visibilidade

Para otimizar os grafos de visibilidade existe um número maior de hipóteses. É importante observar que em ambientes 2D o melhor caminho passa pelos vértices dos obstáculos porém em ambientes 3D o melhor caminho passa por algum ponto desconhecido das arestas dos obstáculos [Berg et al. 2008]. Construir o grafo a partir do ponto médio das arestas dos obstáculos deve melhorar o caminho. Porém para obter melhora significativa seria necessário adicionar um algoritmo de pós-processamento iterativo no caminho gerado fazendo uma espécie de busca binária sobre as arestas dos obstáculos até que o ganho obtido comparado com a iteração anterior seja pequeno. Essa hipótese otimiza uma solução inicial, encontrando o melhor ponto de passagem nas arestas de obstáculos previamente escolhidas. Pois o caminho ótimo pode passar por arestas de obstáculos não escolhidas inicialmente. Como este pós-processamento é sobre o caminho gerado e não sobre o grafo inteiro o custo computacional não é tão elevado.

Outra otimização a ser considerada é que um par de vértices, quanto a visibilidade, pode ser classificado em 4 grupos:

- **Ocluso** - Há um obstáculo entre os vértices do par bloqueando a visibilidade.
- **Invisível** - Pelo menos um dos vértices do par é adjacente a somente faces ocultas.
- **Central** - Pelo menos um dos vértices do par é adjacente a somente faces visíveis.
- **Tangente** ambos os vértices estão na fronteira entre as faces visíveis e ocultas.

Somente os pares **tangentes** são candidatos a participar do melhor caminho. Os pares **centrais** geram caminhos mais longos e incluí-los infla o grafo de arestas irrelevantes tornando a busca de caminhos mais lenta. Os pares **invisíveis** sobrecarregam o teste de visibilidade que é o trecho de maior complexidade computacional do algoritmo. A classificação dos pares como **invisível**, **central** ou **tangente** pode ser feita pelas normais das faces adjacentes. É necessário verificar a visibilidade para descartar os pares **occlusos**. Se a verificação de visibilidade for ordenada em uma direção do espaço é possível detectar a oclusão com menor custo computacional. Os vértices tangentes de um obstáculo próximo formam um fecho convexo e todos os pares de vértices que intersectam este fecho convexo são pares oclusos.

## 5. Considerações finais

Pelos aspectos citados, avaliando os resultados apresentados na Tabela 1 e pelo potencial de otimização, a abordagem por grafo de visibilidade tem demonstrado ser mais promissora que a abordagem por grade. Porém, como já comentado, em cenários complexos a abordagem por grade leva vantagem. Combinar as abordagens permitiria aproveitar o melhor de cada uma. Existe uma ordem topológica no ambiente que ainda precisa ser melhor compreendida e aproveitada. Encontrar uma estrutura de dados adequada é uma meta e existem outras abordagens, citadas por [Yang et al. 2016] como alternativas, que precisam ser verificadas, por exemplo: particionamento binário do espaço (BSP), tetraedralização de Delaunay e/ou diagrama de Voronoi e fecho convexo. Outra perspectiva que existe é que sendo encontrada uma solução satisfatória na terceira dimensão, avançar para dimensões mais altas será menos custoso. Os problemas do mundo real, que são os mais interessantes, necessitam de muitos graus de liberdade, recaindo nas dimensões mais elevadas.

## Referências

- Andayesh, M. e Sadeghpour, F. (2014). A comparative study of different approaches for finding the shortest path on construction sites. *Procedia Engineering*, 85:33–41.
- Berg, M. d., Cheong, O., Kreveld, M. v., e Overmars, M. (2008). *Computational geometry : algorithms and applications*. Springer, Germany, 3rd ed edition.
- Canny, J. (1988). *The complexity of robot motion planning*. MIT press.
- Dijkstra, E. W. (1959). A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1):269–271.
- Ingerman, P. Z. (1962). Algorithm 141: path matrix. *Communications of the ACM*, 5(11):556.
- Meagher, D. (1982). Geometric modeling using octree encoding. *Computer graphics and image processing*, 19(2):129–147.
- Santos, C. A. e Guedes, A. L. P. (2019). Algoritmos de planejamento de caminhos 3D. *Anais do Computer on the Beach*, pages 774–775.
- Yang, L., Qi, J., Song, D., Xiao, J., Han, J., e Xia, Y. (2016). Survey of robot 3D path planning algorithms. *Journal of Control Science and Engineering*, 2016:5.

# Implementação de um Sistema de Reconhecimento de Voz em Protótipos para a Comunicação LIBRAS-Língua Portuguesa

Maria Alice Trinta<sup>1</sup>, Vinicius Souza de Jesus<sup>1</sup>, Carlos Eduardo Pantoja<sup>1</sup>

<sup>1</sup> Centro Federal de Educação Tecnológica Celso Suckow (CEFET/RJ)  
20785-220 – Rio de Janeiro – RJ – Brasil

atg200210@outlook.com, souza.vdj@gmail.com, pantoja@cefet-rj.br

**Abstract.** *This work presents a voice recognition system that was implemented to LuBRAS: a prototype that performs the translation between LIBRAS and the Portuguese language. The original architecture of LuBRAS was extended to add a mechanism to capture a spoken word and transform it into a written word, giving the signs that will be interpreted by the learning glove.*

**Resumo.** *Este trabalho apresenta um sistema de reconhecimento de voz que trabalha em conjunto com os protótipos de comunicação Língua Brasileira de Sinais (Libras)-Língua portuguesa conhecido como LuBRAS. A arquitetura original do LuBRAS foi estendida para que fosse possível adicionar um mecanismo de captura da palavra falada e a transformar em sinais que serão interpretados pela luva de aprendizado.*

**Keywords:** LIBRAS; portuguese language; communication glove; learning glove; Voice recognition system.

**Palavras-chave:** LIBRAS; língua portuguesa; luva de comunicação; luva de aprendizado; Sistema de reconhecimento de voz.

## 1. Introdução

A Tecnologia assistiva [Bersch 2008] é um termo referente ao conjunto de serviços e recursos que contribuem para propiciar ou aperfeiçoar habilidades funcionais a pessoas com deficiência, que ajuda a promover tanto independência, quanto a inclusão social. Na literatura é possível encontrar diversos trabalhos que visam melhorar a comunicação entre surdos e ouvintes, como por exemplo o *SensorLibras* [Tavares et al. 2010] que faz uma tradução da Língua Brasileira de Sinais (LIBRAS) para o português, e o *Hand-Talk* [Tenório et al. 2019] que traduz textos do português para a LIBRAS. Entretanto, estes trabalhos são unidirecionais funcionando apenas de ouvinte para surdo ou de surdo para ouvinte.

Para tratar tais limitações, o trabalho *LuBRAS* [Jesus et al. 2016], que foca na melhora da comunicação entre surdos e ouvintes, e também possibilita a promoção do ensino da LIBRAS, trabalhando assim de forma bidirecional. Porém, o protótipo era pouco uma vez que suas interfaces de interação destinadas ao ouvinte permitiam apenas inserção de sentença através de digitação, porém o protótipo não era prático uma vez que

suas interfaces de interação destinadas ao ouvinte permitiam apenas inserção de sentença através de digitação, podendo deixar a comunicação lenta. Logo, foi identificado que a implementação de um mecanismo de reconhecimento de voz nas interfaces trará mais praticidade e usabilidade aos protótipos.

Portanto, o intuito deste trabalho é implementar ao *LuBRAS* um sistema de reconhecimento voz que atua como uma alternativa de entrada na interface destinada ao ouvinte, tornando possível inserir sentenças tanto oralmente, quanto por meio da digitação, fazendo com que a tradução se torne mais prática, adaptável e eficiente. Para isso, a arquitetura original do fluxo de comunicação do *LuBRAS* foi estendida para permitir que a funcionalidade de reconhecimento de voz seja possível através de um sistema que identifique uma sentença a partir da produção sonora do usuário utilizando um captador de áudio juntamente de um *software* que fara o reconhecimento da fala. Como avaliação experimental, uma validação por prototipação a partir de um cenário e prova de conceito foram apresentados.

Este trabalho está estruturado da seguinte maneira: Na segunda seção estão presentes os referenciais teóricos; Na terceira seção é discutida a metodologia; A quarta seção traz a avaliação experimental; A quinta seção apresenta os trabalhos relacionados; A sexta seção apresenta a conclusão e os futuros trabalhos; Por fim são declaradas as referências utilizadas.

## 2. Referencial Teórico

Nesta seção são apresentados os conceitos e tecnologias que fazem parte do desenvolvimento deste trabalho e auxiliam na compreensão do protótipo de forma geral. Nos protótipos do *LuBRAS* são utilizados *softwares* que fazem a interface humano-hardware e estão embarcados em placas microcontroladas e em computadores com tamanho reduzido com capacidade para manter um sistema operacional e executar *software* com determinada robustez.

Esses sistemas que independem de periféricos são chamados de Sistemas Embarcados [Cunha 2007] que são construídos para atender a uma determinada aplicação, muitas vezes oferecido ao usuário como o responsável por um dispositivo completo e independente. Diferentemente dos computadores, que possuem capacidade de executar inúmeros aplicativos, os sistemas embarcados executam isoladamente, não tendo permissão ou capacidade, na maioria das vezes, para a execução de outras funcionalidades se não aquelas pré-determinadas em seu projeto.

Dessa forma, o *LuBRAS* utiliza uma abordagem híbrida onde o microcontrolador precisa tanto executar suas funções independentes quanto comunicar-se com o microprocessador, vice-versa, estabelecendo uma comunicação serial. E para estabelecer essa interface de comunicação foi utilizado o Javino [Lazarin and Pantoja 2015], uma biblioteca criada inicialmente para fazer a comunicação entre a linguagem de programação Java e a placa microcontrolada Arduíno, oferecendo uma uma comunicação de mão dupla, tanto do Java para o Arduíno quanto do Arduíno para o Java de maneira segura.

Os sistemas de reconhecimento de fala [Ishizaki 2018], ou sistemas de reconhecimento de voz, utilizados neste trabalho funcionam de maneira a converter um si-

nal acústico, capturado por um transdutor como um microfone, para um conjunto de palavras. O sinal de fala é obtido através da conversão de ondas sonoras em sinais elétricos. O Sistema de reconhecimento de voz que foi implementado ao dispositivo foi construído a partir de uma *Application Interface Program* (API) chamada *Java Speech* [Marangoni and Precipito 2017].

A *Java Speech* API (JSAPI) possui como principal característica ser totalmente proveniente da linguagem Java, apresentando duas funções: o reconhecimento da fala e a síntese da fala. O reconhecimento da fala permite ao computador detectar o áudio, processá-lo e transportá-lo de áudio para texto o que foi dito pelo usuário. Já a síntese da fala gera uma fala sintética que se baseia no texto fornecido por um usuário, sendo assim o processo inverso do reconhecimento da fala, chamado de *Text-to-speech* (Texto para fala).

### 3. Trabalhos relacionados

O trabalho *LuBRAS* [Jesus et al. 2016] é composto por duas luvas, a luva de aprendizado que é utilizada pelo ouvinte, e a luva de comunicação que é utilizada pelo surdo, e duas interfaces gráficas didáticas, uma destinada ao surdo e outra ao ouvinte, com o objetivo de melhorar a comunicação entre LIBRAS e língua-portuguesa.

Portanto, nesta seção são apresentados alguns trabalhos que também focam na comunicação entre LIBRAS e língua portuguesa. O *SensorLibras* [Tavares et al. 2010] possui uma arquitetura organizada em três elementos principais: o primeiro são duas luvas responsáveis pelo sensoriamento dos sinais, o segundo é um dispositivo que recebe via frequência os dados das luvas, e o terceiro é um cabo *USB* ligando o dispositivo anterior à um computador onde serão feitas as leituras de dados e a tradução propriamente dita. Porém, a tradução não é prática uma vez que o sistema, por não ser embarcado, precisa de periféricos como um computador à disposição. Já o *LuBRAS*.

Outro trabalho é o *HandTalk* [Tenório et al. 2019], um aplicativo que faz a tradução de um texto para a LIBRAS, possuindo tanto um módulo para a tradução de páginas *Web* quanto uma aplicação para celulares. Entretanto, o trabalho realiza somente a tradução de textos para LIBRAS e, além disso, é dependente de uma conexão com a internet para realizar a tradução de modo ideal, enquanto o *LuBRAS* faz traduções de ambas as vias e não depende de conexões externas para obtenção de dados.

O *LuBRAS* ao permitir interações somente através do *display*, tornava a comunicação lenta e limitada. Problema que foi solucionado com a implementação de um módulo de reconhecimento de voz, que tornou a comunicação mais natural e a interação com o dispositivo mais dinâmica.

### 4. Metodologia

Nesta seção será apresentada a metodologia referente à criação do sistema de reconhecimento de voz, suas estruturas e tecnologias, bem como sua atuação em conjunto das interfaces destinadas ao ouvinte presentes no protótipo.

O *LuBRAS* dispõe de duas interfaces gráficas, uma dedicada ao surdo e outra ao ouvinte, e dois protótipos de luvas. O primeiro protótipo de luva é chamado luva de

comunicação, é utilizada pelo surdo como um meio para interpretar seus movimentos em LIBRAS e traduzi-los para a língua portuguesa. A luva de comunicação pode ser controlada através da interface dedicada aos surdos. O segundo protótipo é chamado luva de aprendizado, é usada pelo ouvinte com a intenção de que a tradução da língua portuguesa para LIBRAS seja didática, ensinando ao ouvinte os movimentos necessários a se fazer para executar a sentença desejada em LIBRAS. A luva de aprendizado pode ser controlada a partir da interface dedicada ao ouvinte.

Existem também meios às interfaces gráficas, duas telas de interação que são utilizadas para fazer a tradução entre LIBRAS e língua portuguesa, sendo que uma delas faz a tradução de LIBRAS para língua portuguesa, montando as sentenças a partir de imagens dos sinais que representam letras. Já a outra interface gráfica faz a tradução de língua portuguesa para LIBRAS, inserindo uma sentença que, através de imagens, terá letra por letra demonstrada em LIBRAS no *display*. O dispositivo é totalmente vestível e dispensa qualquer conexão externa para obtenção de dados, como a *internet*, o que torna o trabalho totalmente móvel e independente.

Um fluxograma que vai do módulo de reconhecimento até as interfaces gráficas da luva será apresentado a fim de auxiliar na visualização do funcionamento do novo método de entrada.

#### 4.1. O sistema de Reconhecimento de voz

Com o objetivo de tornar o *LuBRAS* mais diversificado e usual, foram implementados ao trabalho um dispositivo de captura de fala juntamente de um *software* auxiliar que fará o reconhecimento da fala, para que o áudio reconhecido se transforme em uma sentença que será traduzida para LIBRAS. O novo módulo é uma alternativa de entrada que funcionará juntamente da interface de controle da luva de aprendizado e da interface de tradução Língua portuguesa-LIBRAS.

A extensão da arquitetura consiste na adição da JSAPI juntamente de um captador de áudio, nesse caso um microfone, trabalhando com as interfaces destinadas aos ouvintes como uma alternativa para a entrada de informações que serão encaminhadas pelo resto da arquitetura como mostra o Fluxograma de Funcionamento (figura 1). A JSAPI foi escolhida principalmente por ser inteiramente proveniente da linguagem JAVA, pois o *LuBRAS* foi majoritariamente desenvolvido em JAVA. Em segundo, pois a JSAPI é totalmente embarcável e não necessita de conexão externa com internet para obtenção de dados.

Em funcionamento, o programa do reconhecimento de voz grava em uma *String* a palavra falada pelo usuário e disponibiliza através do *display* o que foi reconhecido, para que o usuário confirme e a arquitetura do *LuBRAS* faça a tradução. O novo método de entrada poderá ser utilizado tanto pela interface destinada ao controle da luva de aprendizado ou pela interface de tradução por inserção de texto, ambas interfaces dedicadas ao ouvinte. O *software* de reconhecimento foi embarcado juntamente da aplicação pré-existente do *LuBRAS*, assim como o microfone. Sendo assim, o módulo de reconhecimento de voz age juntamente das interfaces do dispositivo *LuBRAS* para uma usabilidade diversificada e interativa, tornando a comunicação mais fluida e natural.

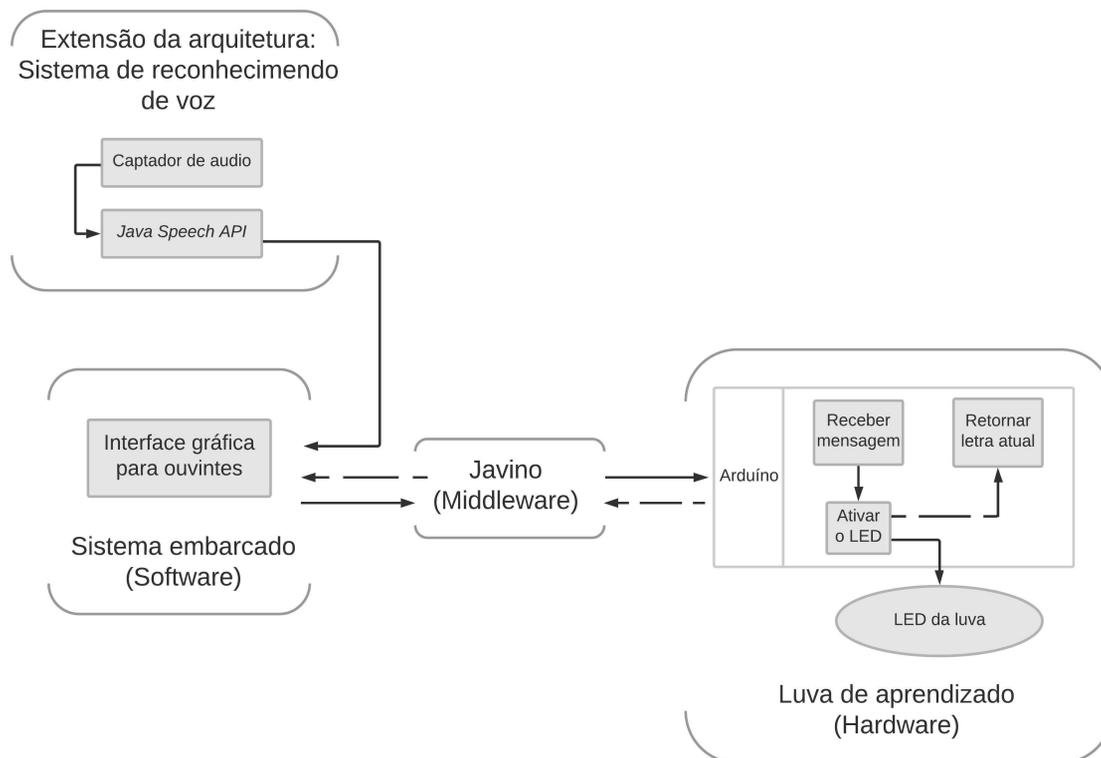


Figura 1. Fluxograma de Funcionamento da Extensão da Arquitetura

## 5. Prova de conceito

Esta seção tem como objetivo esclarecer o funcionamento do sistema de reconhecimento de voz, juntamente da luva de aprendizado, no meio prático como método de avaliação. Para exemplificar o funcionamento da luva de aprendizado e do sistema de reconhecimento de fala é apresentada uma prova de conceito.

A prova de conceito é apresentada através de uma situação presente em inúmeros estabelecimentos: O número limitado de funcionários que, frente a uma grande demanda, perde em qualidade de atendimento. Sem o comando de voz, utilizar somente as entradas para texto do *LuBRAS*, atrasaria o atendimento uma vez que seria preciso parar todas as ações para digitar as sentenças na interface gráfica da luva para os ouvintes.

Sendo assim, o usuário ouvinte com seu dispositivo *LuBRAS*, aciona o comando de voz através da interface gráfica para ouvintes e, em seguida, o usuário fala uma palavra em língua portuguesa. Essa palavra é reconhecida pelo sistema de voz e é retornada na tela do usuário para que haja a confirmação se a palavra identificada está correta ou não. Não havendo confirmação, o sistema de reconhecimento de voz é acionado novamente para uma nova identificação.

Assim que a palavra for confirmada pelo ouvinte, ela é transformada em um vetor de caracteres que é enviado através do *middleware* Javino para o microcontrolador Arduino onde é feita as associações entre os caracteres em língua portuguesa com os sinais em LIBRAS. Está associação é dada, transformando cada carácter em língua portuguesa

em sinais luminosos que indicam os movimentos necessários para fazer o sinal em LIBRAS referente a cada um destes caracteres.

## 6. Considerações Finais

O trabalho parte da ideia de inclusão, vindo como necessidade a melhora da comunicação entre surdos e ouvintes, promovendo também o ensino da LIBRAS. O *LuBRAS* busca ser um dispositivo de uso comum no dia-a-dia, por isso, nota-se a mobilidade do trabalho, a não necessidade de obtenção de dados externos via internet por exemplo, e a importância em facilitar o uso do dispositivo através de um *display* otimizado, simples e intuitivo. O objetivo da adição de um sistema reconhecimento de voz é justamente tornar o *LuBRAS* ainda mais prático, uma vez que possibilitou ao usuário uma interação mais flexível, rápida e natural com o dispositivo.

Para trabalhos futuros o *LuBRAS* pretende-se embarcar junto à interface de comunicação um sintetizador de voz, de modo que a palavra traduzida de LIBRAS para língua-portuguesa possa ser escutada pelo ouvinte. Além disso, é desejado desenvolver um protótipo mais robusto e resistente, pois iremos trocar a placa de arduíno por um circuito impresso próprio para cada protótipo com toda a parte de *hardware* embutida no circuito impresso.

## Referências

- Bersch, R. (2008). Introdução à tecnologia assistiva. *Porto Alegre: CEDI*.
- Cunha, A. F. (2007). O que são sistemas embarcados. *Saber Eletrônica*, 43(414):1–6.
- Ishizaki, M. Y. (2018). Reconhecimento automático de palavras. B.S. thesis, Universidade Tecnológica Federal do Paraná.
- Jesus, V. S., Manoel, F. C. P. B., Pantoja, C. E., and Samyn, L. M. (2016). Lubras: Uma arquitetura de um dispositivo eletrônico para a comunicação libras-língua portuguesa utilizando o javino. *I Workshop de Pesquisa em Computação dos Campos Gerais (WPCCG)*.
- Lazarin, N. M. and Pantoja, C. E. (2015). A Robotic-Agent Platform for Embedding Software Agents using Raspberry Pi and Arduino Boards. In *9<sup>th</sup> Software Agents, Environments and Applications School*.
- Marangoni, J. B. and Precipito, W. B. (2017). Reconhecimento e sintetização de voz usando java speech. *Revista Científica Eletrônica de Sistemas de Informação (ISSN 1807-1872)*. Ano, 2.
- Tavares, J. E. d. R., Barbosa, J. L., and Leithardt, V. R. (2010). Sensorlibras: Tradução automática libras-português através da computação ubíqua. In *II Congresso Nacional de Pesquisas em Tradução e Interpretação de Libras e Língua Portuguesa, Universidade Federal de Santa Catarina-UFSC*.
- Tenório, R., Wanderlan, C., and Luz, T. (2019). Hand talk. *handtalk.com*. Accessed: 27 February 2019.

# Implantação de uma arquitetura ciber-física baseada em agentes para um Smart Parking

Pedro W. Botelho<sup>1</sup>, Gleifer V. Alves<sup>1</sup>, André P. Borges<sup>1</sup>

<sup>1</sup>Departamento Acadêmico de Informática – Universidade Tecnológica Federal do Paraná  
Rua Doutor Washington Subtil Chueire, 330 - 84017-220 – Ponta Grossa – PR – Brasil

pbotelho@alunos.utfpr.edu.br, {gleifer, apborges}@utfpr.edu.br

**Abstract.** *Smart Parking systems are being used to manage and allocate parking lots in several cities. These systems aim to reduce urban traffic by helping drivers to find a place to park. The inclusion of new technologies such as Cyber-Physical Systems and artificial intelligence techniques such as Multi-Agent Systems are considered a suitable alternative for solving this problem. In this context, the interconnection of the physical part with the agents is a fundamental issue for the correct functioning of the system. Thus, this paper presents the implementation of an agent-based cyber-physical architecture that focuses on connecting agents with physical controllers using IoT technologies.*

**Resumo.** *Sistemas para estacionamentos inteligentes estão sendo utilizados no gerenciamento e alocação de vagas em diversas cidades. Estes sistemas visam a redução do tráfego urbano auxiliando os motoristas na busca por vagas. A inclusão de novas tecnologias, como Sistemas Ciber-Físicos e técnicas de inteligência artificial, como Sistemas Multi-Agentes são consideradas uma alternativa considerável para a resolução deste problema. Neste contexto, a conexão da parte física com os agentes é uma questão fundamental para o correto funcionamento do sistema. Assim, este trabalho apresenta a implantação de uma arquitetura ciber-física baseada em agentes que tem como foco a conexão dos agentes com controladores físicos utilizando tecnologias de Internet das Coisas.*

**Keywords:** Agents; Smart Parking; Cyber-Physical Systems; Internet of Things

**Palavras-chave:** Agentes; Estacionamento Inteligente; Sistemas Ciber-Físicos; Internet das Coisas

## 1. Introdução

As Cidades Inteligentes usam tecnologias de comunicação e informação como sensores para coletar dados e então, usa-se estes dados para gerenciar seus recursos de forma eficiente [Caragliu et al. 2011]. Para gerenciar os dados de tráfego de maneira eficiente, deve-se também lidar com uma das causas de engarrafamento nas cidades, a busca por vagas de estacionamento [Lin et al. 2017].

A divergência entre a falta de espaço nas grandes cidades e o número de carros presentes nelas faz com que encontrar vagas seja um problema para os motoristas. Dirigir

à procura de uma vaga faz com que o trânsito se torne mais lento, causando congestionamentos, gasto de tempo e gasolina dos veículos e ainda emitindo no meio ambiente gases poluentes que não seriam emitidos se o carro estivesse estacionado [Lin et al. 2017].

Para tentar lidar com esse problema surgem os Estacionamentos Inteligentes. Estacionamento Inteligente é um sistema de estacionamento que auxilia os motoristas a encontrar vagas utilizando sensores que detectam se há ou não um veículo e então o direcionam para a vaga [Hassoune et al. 2016]. O objetivo desses sistemas é garantir que o motorista encontre uma vaga para estacionar da maneira mais rápida possível levando em consideração o local desejado. Assim, o motorista pode iniciar uma negociação para reservar uma vaga antes mesmo de entrar no veículo [Castro et al. 2017].

Por ser uma solução complexa, estacionamentos inteligentes precisam adotar tecnologias que atendam a demanda da implementação, como os Sistemas Ciber-Físicos (SCF) que oferecem comunicação entre elementos computacionais e entidades físicas utilizando plataformas de hardware [Kaithan and McCalley 2015]. Porém, esta alternativa não apresenta uma solução inteligente contendo gerenciamento de vagas e negociação, assim, é proposto utilizar agentes embarcados nestas entidades físicas. Um agente é um sistema computacional capaz de ações autônomas no ambiente que está situado visando atingir seus objetivos propostos [Wooldridge 2009]. Desta forma, é possível fazer com que agentes façam a negociação de uma vaga enquanto outros agentes gerenciam as vagas do estacionamento.

Em trabalhos relacionados foram desenvolvidos mecanismos de negociação entre agentes [Castro et al. 2017] [Ducheiko et al. 2018]. Esses trabalhos realizam simulações computacionais para avaliar sua aplicabilidade em um estacionamento inteligente. Logo, há necessidade de realizar tais simulações em ambientes físicos, o que possibilitará uma avaliação precisa para incorporar uma arquitetura baseada em agentes em um estacionamento inteligente.

Em [Sakurada et al. 2019], é justamente descrita uma arquitetura ciber-física para estacionamentos inteligentes, porém tal arquitetura considera um estacionamento para bicicletas. O trabalho descrito aqui propõe a implantação de uma arquitetura ciber-física baseada em agentes, para veículos. Tal arquitetura visa obter informações a respeito da vaga de estacionamento, para que a partir destas informações, seja possível ao agente interagir com o sistema de estacionamento. Assim, esta abordagem permitirá criar um meio de se realizar testes de protocolos de negociação em ambientes mais próximos de um estacionamento real, materializando assim os trabalhos anteriormente citados.

O restante do artigo está estruturado da seguinte forma. Na seção 2, é descrito o desenvolvimento da arquitetura, contendo uma maior descrição das tecnologias utilizadas e como cada uma delas atua dentro do sistema. Na seção 3, é descrito a implantação da arquitetura, como cada dispositivo foi programado para funcionar corretamente. E por fim, a seção 4 apresenta as considerações finais a respeito do desenvolvimento do trabalho e os trabalhos futuros.

## 2. Desenvolvimento da arquitetura

Em um SCF, a parte física é capaz de interagir com o mundo real por meio de sensores e a parte ciber é o processo de obter dados destes sensores [Jazdi 2014]. Assim, a arquitetura é baseada em [Sakurada et al. 2019] sendo a parte ciber constituída de placas Raspberry Pi contendo um agente embarcado e implementado no framework JADE para receber as informações da parte física e a parte física composta por módulos ESP-12e conectados a sensores de presença para monitorar o estado da vaga (por exemplo, livre ou ocupada).

A comunicação entre estes dois dispositivos dá-se-a através de um broker MQTT que também está embarcado no Raspberry Pi. Este broker segue o princípio *Publish-Subscribe*, que tem seu funcionamento por meio de tópicos. Um dispositivo publica em um tópico no broker e o mesmo replica esta mensagem a todos os dispositivos que estão inscritos neste tópico. Assim, é possível que dois dispositivos troquem informações, o que atende as requisições necessárias para o funcionamento do sistema.

A Figura 1 ilustra a arquitetura, que é composta por  $n$  setores distribuídos em um estacionamento e cada setor possui  $m$  vagas que são gerenciadas pelo agente responsável pelo setor. A parte da negociação ocorre em outra camada da arquitetura pois o presente artigo tem como escopo apenas o ambiente físico do estacionamento. Porém, basicamente o motorista faz a requisição de uma vaga e um Sistema Multi-Agente (SMA) é responsável por receber esta requisição, negociar a vaga com o motorista e atualizar uma base de dados que contém as informações sobre o motorista e a vaga, como o número do setor e da vaga, o id do motorista, horário da reserva, etc.

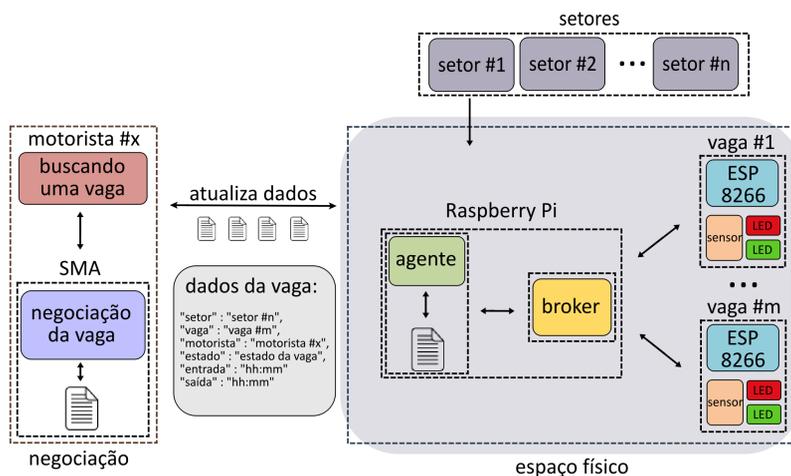


Figura 1. Implementação da comunicação ciber-física

O agente, desenvolvido utilizando o framework JADE e embarcado em um Raspberry Pi 3B+, é responsável por receber uma mensagem do controlador físico, verificar se a vaga está reservada consultando a base de dados e atualizar o estado da vaga. O controlador físico é composto por um microcontrolador ESP8266, um sensor ultrassônico de presença e dois LED's, um verde e outro vermelho. Juntos, estes dispositivos são responsáveis por enviar ao agente uma mensagem através do broker contendo o status correspondente da vaga para que possa ser feita a verificação.

Cada vaga possui um ESP8266 conectado a um sensor de presença para medir a distância entre o dispositivo e um obstáculo a sua frente, no caso de estacionamentos fechados, a medida pode ser entre o chão e o teto. Quando um carro estaciona na vaga, o sensor medirá a distância dele até o carro, portanto, a distância anterior e a atual não é a mesma. Como resultado, é possível reconhecer que há um carro estacionado e então enviar uma mensagem para o agente. O mesmo procedimento é seguido quando o carro sai da vaga. Assim, o microcontrolador é programado para monitorar o sensor e enviar uma mensagem ao broker toda vez que o sensor detectar que a distância foi alterada.

Para que o sistema funcione corretamente, o agente correspondente do setor deve gerenciar as mensagens recebidas do broker e atualizar o estado da vaga, enquanto o SMA é o responsável pela negociação das vagas livres com o motorista. Como consequência, o agente é programado para receber as mensagens do controlador da vaga, consultar a base de dados para verificar o estado dela, e então enviar uma mensagem de volta ao controlador com qual LED deve acender-se. Especificamente, o sistema aqui proposto funciona da seguinte maneira:

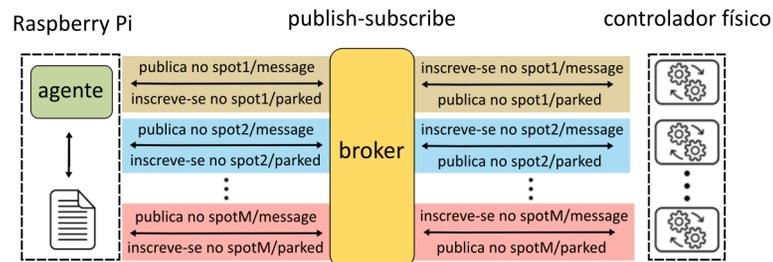
1. O motorista requer uma vaga de estacionamento por meio de um aplicativo móvel ou de um sistema web;
2. O SMA recebe esta requisição, negocia uma vaga baseado nas vagas livres e atribui uma vaga ao motorista. Ex: vaga#15;
3. O SMA atualiza a base de dados mudando o estado da vaga e atualizando os demais campos. Assim, a vaga#15 não está mais livre e sim reservada;
4. Neste momento, o agente no Raspberry Pi recebe a informação que a base de dados foi atualizada, então o mesmo fica a espera de uma notificação;
5. O motorista estaciona na vaga#15;
6. O sensor no ESP8266 detecta que um carro estacionou na vaga#15 e envia uma mensagem notificando o agente;
7. O agente então deve verificar a base de dados daquela vaga: a vaga#15 está reservada para este horário?
8. Se a vaga está reservada, o agente atualiza o estado da vaga na base de dados de reservado para ocupado;
9. O agente envia uma mensagem para o ESP8266 que a vaga#15 agora está ocupada;
10. O LED vermelho deve ser aceso indicando que a vaga está ocupada.

Note que se a reserva não for confirmada (não existe reserva para aquela vaga na base da dados), o agente não vai publicar a confirmação da vaga e assim o ESP8266 não irá acendar o LED vermelho, mantendo o LED verde aceso. Portanto, assume-se que o motorista irá respeitar a política do sistema e apenas estacionar em uma vaga corretamente atribuída ao motorista.

### **3. Implantação da arquitetura**

Como um primeiro experimento, foi desenvolvido um simples protótipo ciber-físico contendo duas vagas contendo um controlador físico em cada. Com isso, foi possível testar a troca de mensagens entre o agente e o controlador físico por meio do broker. Esta troca

de mensagens só é possível por meio de tópicos, assim, para receber as mensagens publicadas em um tópico, o dispositivo deve estar inscrito no mesmo. A Figura 2 ilustra os tópicos seguidos para a troca de mensagens e ainda em qual tópico cada dispositivo está inscrito e publica mensagens.



**Figura 2. Tópicos do broker para o suporte de comunicação publish/subscribe**

Neste experimento não é feito o uso do mecanismo de negociação, portanto, para simular um carro estacionado é necessário utilizar um objeto (representando o carro) dentro do raio de alcance do sensor ultrassônico. Dessa maneira, é possível determinar quando o controlador deve notificar o agente que um carro está estacionado. O agente implementado nos experimentos executa passos simples: recebe uma mensagem por meio do broker e envia de volta uma mensagem de acordo com as especificações abaixo.

O microcontrolador ESP8266 é capaz de conectar a Internet, portanto, pode ser programado para conectar-se no broker que está no Raspberry Pi por meio do endereço de IP. O sensor ultrassônico mede a distância pela duração em microsegundos de quando uma onda ultrassônica é emitida para quando ela bate no objeto a sua frente e volta. Assim, a distância deve ser ajustada para que haja uma distância padrão a que possa ser comparada com a distância atual. Deve-se também manter a distância anterior para evitar o envio mensagens repetidas para o broker. Portanto, quando um carro deixa a vaga, apenas uma mensagem é enviada, e isso é apenas possível comparando a distância anterior com a distância atual.

Quando a conexão com a Internet é estabelecida e a distância padrão é ajustada, o LED verde é acendido e a distância atual é obtida e comparada com a distância padrão. Se a distância atual é menor que a distância padrão significa que um carro está estacionado na vaga, portanto, os seguintes passos são seguidos:

- O ESP8266 publica no tópico 'spot1/parked' que um carro estacionou;
- O agente verifica a base de dados da vaga e atualiza o estado dela para ocupado;
- O agente então publica no tópico 'spot1/message' que a reserva está confirmada;
- O ESP8266 recebe a mensagem e acende o LED vermelho.

Quando o carro sai da vaga, uma nova mudança na distância é detectada pelo sensor, sendo necessário executar os seguintes passos:

- O ESP8266 publica no tópico 'spot1/parked' que um carro não está mais estacionado;
- O agente recebe a mensagem e atualiza o estado da vaga para livre;

- O agente então publica no tópico ‘spot1/message’ que o estado da vaga foi atualizado;
- O ESP8266 recebe a mensagem e acende o LED verde.

Se a vaga está livre, o sensor continua medindo a distância a cada cinco segundos até que uma mudança na distância é detectada. Quando isso ocorre, os passos anteriores são reproduzidos.

Neste contexto, um experimento foi realizado para identificar o correto funcionamento dos dispositivos escolhidos e também o processo de troca de mensagens entre o agente e o ESP8266 por meio do broker. Este experimento foi realizado durante quatro horas e os controladores físicos das duas vagas foram capazes de detectar o carro estacionado as 25 vezes que um objeto simulando o carro foi colocado a frente de cada, assim como foi capaz de enviar a mensagem correspondente ao agente 100 vezes, 50 informando que o carro estacionou e 50 que o carro saiu da vaga. Além disso, o agente apresentou o comportamento esperado gerenciando as 100 mensagens recebidas de forma correta e enviou de volta uma mensagem informando qual LED deveria ser aceso.

#### 4. Conclusão

Este artigo apresenta a implantação de uma arquitetura baseada em agentes e SCF. A arquitetura possui como uma de suas características, a escalabilidade, visto que é possível replicar a estrutura em  $n$  setores e  $m$  vagas, dependendo da necessidade do ambiente de implantação. Outra característica diz respeito a modularização, já que facilmente módulos podem ser trocados de forma que seja implantado em outro ambiente, por exemplo um estacionamento inteligente de bicicletas.

Tendo como base os experimentos iniciais é possível evidenciar que o princípio *Publish-Subscribe* apresenta resultados satisfatórios na comunicação entre o agente e os controladores físicos. Além disso, o sensor ultrassônico é capaz de detectar um carro todas as vezes, enviando corretamente a mensagem para o agente o qual lidou satisfatoriamente com o gerenciamento das mensagens.

Como continuidade imediata deste trabalho, deseja-se implantar uma API REST para conectar a arquitetura com um SMA responsável pela negociação entre os agentes do sistema. A partir disto será implantando um simples protótipo de um estacionamento inteligente, onde serão executados diversos testes para evidenciar a aplicabilidade desta arquitetura em um cenário com componentes de um ambiente físico.

#### Referências

- Caragliu, A., Bo, C. D., and Nijkamp, P. (2011). Smart Cities in Europe. *Journal of Urban Technology*, 18(2):65–82.
- Castro, L. F. S. D., Alves, G. V., and Borges, A. P. (2017). Using trust degree for agents in order to assign spots in a Smart Parking. *ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal*, 6(2):45–55.
- Ducheiko, F. F., Alves, G. V., and Borges, A. P. (2018). Implantação de um modelo de raciocínio e protocolo de negociação para um estacionamento inteligente com mecanismo de negociação descentralizado. In *Revista Junior – ICCEE*, pages 25–32.

- Hassoune, K., Dachry, W., Moutaouakkil, F., and Medromi, H. (2016). Smart parking systems: A survey. In *2016 11th International Conference on Intelligent Systems: Theories and Applications (SITA)*, pages 1–6. IEEE.
- Jazdi, N. (2014). Cyber physical systems in the context of industry 4.0. *Proceedings of the IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR'14)*, pages 1–4.
- Kaithan, S. K. and McCalley, J. D. (2015). Design techniques and applications of cyberphysical systems: A survey. *IEEE Systems Journal*, pages 350–365.
- Lin, T., Rivano, H., and Le Mouël, F. (2017). A survey of smart parking solutions. *IEEE Transactions on Intelligent Transportation Systems*, 18(12):3229–3253.
- Sakurada, L., Barbosa, J., Leitão, P., Botelho, P. W., Alves, G. V., and Borges, A. P. (2019). Development of agent-based cps for smartparking systems. In *Aguardando publicação*.
- Wooldridge, M. (2009). *An introduction to multiagent systems*. John Wiley & Sons.

# $\mathcal{P} = \mathcal{NP}$ or 5-Snarks Exist

Leandro M. Zatesko<sup>1</sup>, Éverton de A. Vieira<sup>2</sup>

<sup>1</sup>Academic Department of Informatics, Federal University of Technology — Paraná,  
Campus Curitiba, Brazil

<sup>2</sup>Federal University of Fronteira Sul, Campus Chapecó, Brazil

**Abstract.** *Similar to the way snarks are defined for 3-regular graphs, we present the definition of 5-snarks for 5-regular graphs. Although we do not know any such graph yet, we prove that they must exist, unless  $\mathcal{P} = \mathcal{NP}$ . We also prove that, unless  $\mathcal{P} = \mathcal{NP}$ , the number of 5-snarks of a given order cannot be polynomially bounded.*

**Resumo.** *De modo similar a como snarks são definidos para grafos 3- regulares, apresentamos a definição de 5-snarks para grafos 5- regulares. Embora não conheçamos nenhum destes grafos ainda, provamos que eles precisam existir, a menos que  $\mathcal{P} = \mathcal{NP}$ . Também provamos que, a menos que  $\mathcal{P} = \mathcal{NP}$ , o número de 5-snarks de uma dada ordem não pode ser limitado polinomialmente.*

**Keywords:** Graph theory in relation to Computer Science (MSC 68R10); Computational difficulty of problems (MSC 68Q17); Colouring of graphs and hypergraphs (MSC 05C15).

## 1. Introduction

The *Four Colour Theorem* is a major and well-known result in the history of Mathematics which states that *no more than four colours suffice to colour any map in such a way that no two adjacent regions are coloured the same*. This statement was first conjectured in 1852 and it took more than 120 years to be proved [Appel and Haken 1977, Appel et al. 1977]. During the 20<sup>th</sup> century, many research branches in Combinatorics and Computer Science have been developed from the efforts on trying to prove the Four Colour Theorem, specially the studies on graph colouring problems and on graph planarity and embeddings, leading to several important applications [Fu and Ma 2013, Lewis 2016].

Amongst the earliest results on the Four Colour Problem is Tait's Theorem, which states that the Four Colour Theorem (Conjecture, by that time) is equivalent to the statement that there is no cubic (i.e. 3-regular) 2-edge-connected non-3-edge-colourable graph<sup>1</sup> which is planar [Tait 1880]. Cubic 2-edge-connected non-3-edge-colourable graphs have very peculiar properties and play an important role concerning many graph problems [Chladný and Škoviera 2010]. These graphs were named *snarks* by [Gardner 1976] after the mysterious thing hunted in the poem *The Hunting of the Snark*, by Lewis Carroll.

The first snark to be known was the Petersen graph (Fig. 1a), which, although credited to Petersen [Petersen 1898], had already appeared earlier [Kempe 1886]. It took

<sup>1</sup>We refer the reader to [Diestel 2010] for graph-theoretical definitions, whose notation we follow. For definitions on Computational Complexity, we refer the reader to [Arora and Borak 2007].

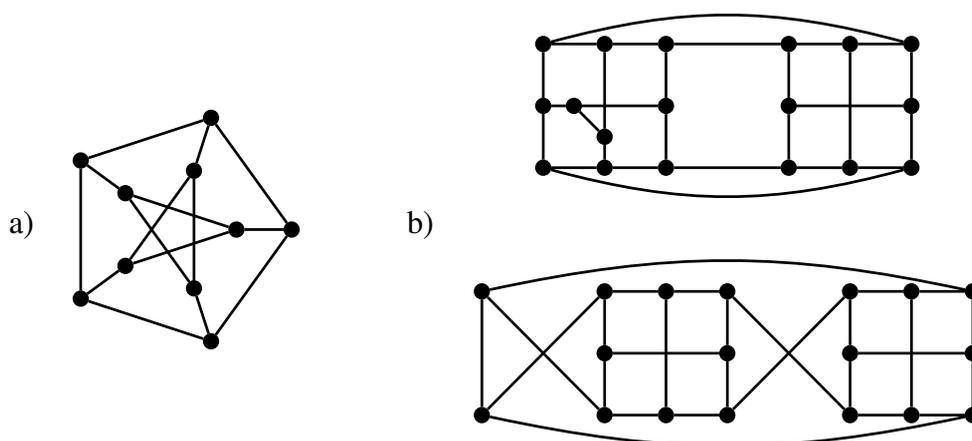


Figure 1. The Petersen graph (a) and the Blanuša snarks (b)

half a century for other three snarks to be found: the two 18-vertex Blanuša snarks [Blanuša 1946] (Fig. 1b) and the 210-vertex Descartes snark [Descartes 1948]. The first infinite family of snarks to be shown are the *Flower snarks* [Isaacs 1975]. Now it is known that the number of snarks of sufficiently large even order  $n$  is at least  $2^{(n-84)/18}$  [Skupień 2007] (recall that there is no cubic graph of odd order). However, despite this exponential lower bound, snarks can be still regarded as *rare* graphs, since, for every  $d \geq 3$ , the proportion of  $d$ -regular non- $d$ -edge-colourable graphs of even order  $n$  over all  $n$ -order  $d$ -regular graphs goes to 0 as  $n/2$  goes to  $\infty$  [Robinson and Wormald 1994].

In this paper, we introduce the *5-snarks*, defined below for 5-regular graphs similarly as snarks are defined for cubic graphs. As much is known about the relation between snarks and important graph problems and conjectures when restricted to cubic graphs, the aim of investigating 5-snarks is to improve the state of the art of these problems and conjectures concerning graphs of higher degree.

**Definition.** A *5-snark* is a 5-regular 4-edge-connected non-5-edge-colourable graph.

Although we do not know yet any graph fitting this definition, we prove that:

**Theorem 1.** *If 5-snarks do not exist, then  $\mathcal{P} = \mathcal{NP}$ .*

Not only these graphs *must* exist, but there cannot be *too few* of them<sup>2</sup>.

**Corollary 2.** *There cannot be a polynomial  $p(n)$  which bounds above the number of 5-snarks of order  $n$ , unless  $\mathcal{P} = \mathcal{NP}$ .*

Amongst the important open graph problems related to the snarks are the *Overfull* and the *1-Factorisation Conjectures* in edge-colouring<sup>3</sup>. A graph  $G$  is said to be *subgraph-overfull* (SO) if has a subgraph  $H$  with  $\Delta(H) = \Delta(G) =: \Delta$  which is *overfull*, i.e.  $|E(H)| > \Delta \lfloor |V(H)|/2 \rfloor$ . Clearly, no SO graph can be  $\Delta$ -edge-colourable, but the converse does not hold, and in the same manner that the snarks are examples of cubic graphs which are neither 3-edge-colourable nor SO, the 5-snarks are examples of 5-regular

<sup>2</sup>We remark that there cannot also be *too many* 5-snarks, again from [Robinson and Wormald 1994].

<sup>3</sup>We refer the reader to [Zatesko 2018, Ch. 2] for references and details on these important conjectures.

graphs which are neither 5-edge-colourable nor *SO* (see Observation 4). This implies that 5-snarks have at least 16 vertices, unless the *Overfull Conjecture* does not hold, since this conjecture states that all non- $\Delta$ -edge-colourable graphs with less than  $3\Delta$  vertices are *SO*.

The proofs for our results are presented next. We remark that, in this paper, graphs are assumed to be undirected and loopless, but they are allowed to have parallel edges.

## 2. Proofs

Consider the following decision problem.

EDGE-COLOURING:

*Instance:* a graph  $G$ ;

*Question:* is  $G$  a  $\Delta(G)$ -edge-colourable<sup>4</sup> graph?

This problem is  $\mathcal{NP}$ -complete [Holyer 1981] even restricted to  $d$ -regular graphs for any constant  $d \geq 3$  [Leven and Galil 1983].

Now, for any constant  $d \geq 3$ , consider this restriction of EDGE-COLOURING.

EDGE-COLOURING( $d$ -regular,  $(d - 1)$ -edge-connected):

*Instance:* a  $d$ -regular  $(d - 1)$ -edge-connected graph  $G$ ;

*Question:* is  $G$  a  $d$ -edge-colourable graph?

In order to prove Theorem 1, we shall first prove the following.

**Theorem 3.** EDGE-COLOURING(5-regular, 4-edge-connected) is  $\mathcal{NP}$ -complete.

The proof of Theorem 3 goes by reducing EDGE-COLOURING(5-regular) to its restriction EDGE-COLOURING(5-regular, 4-edge-connected). In order to do so using a traditional Karp reduction [Karp 1972], we should present a polynomial-time algorithm which, receiving a 5-regular graph  $G$ , outputs a 5-regular 4-edge-connected graph  $G^\ddagger$  which is 5-edge-colourable if and only if  $G$  is 5-edge-colourable. However, our proof uses a Turing oracle reduction<sup>5</sup> (as the reduction used in Cook's proof of the  $\mathcal{NP}$ -completeness of the Boolean satisfiability problem [Cook 1971]) instead of a Karp reduction. In the context of our problem, such reduction can be viewed as a polynomial-time algorithm which, receiving a 5-regular graph  $G$ , outputs not only one, but possibly several 5-regular 4-edge-connected graphs which are *all* 5-edge-colourable if and only if  $G$  is 5-edge-colourable (or, equivalently, outputs a possibly disconnected 5-regular graph  $G^\ddagger$  whose connected components are 4-edge-connected in a manner that  $G^\ddagger$  is 5-edge-colourable if and only if  $G$  is 5-edge-colourable). Observe that the construction of such reduction still implies that if EDGE-COLOURING(5-regular, 4-edge-connected) is in  $\mathcal{P}$ , then  $\mathcal{P} = \mathcal{NP}$ .

In the construction of our reduction, we assume that the input of the reduction, i.e. the instance of EDGE-COLOURING(5-regular), has no cut with 1 or 3 edges, in view of an argument on edge-colouring regular graphs known as the *Parity Lemma* [Isaacs 1975].

<sup>4</sup>No graph admits edge-colourings using less than  $\Delta$  colours. Since every simple graph is  $(\Delta + 1)$ -edge-colourable [Vizing 1964],  $\Delta$ -edge-colourable graphs and non- $\Delta$ -edge-colourable graphs are often referred to as *Class 1* and to as *Class 2* graphs in the literature, respectively. However, we avoid these terms in this paper as the graphs dealt may not be simple (and hence may require more than  $\Delta + 1$  colours).

<sup>5</sup>We refer the reader to [Arora and Borak 2007] for a more detailed discussion on types of reductions.

This lemma states that if  $G$  is a  $d$ -regular  $d$ -edge-colourable graph and  $F \subseteq E(G)$  is a cut in  $G$ , then, for any  $d$ -edge-colouring of  $G$  with colour set  $\{1, \dots, d\}$ ,

$$f_1 \equiv f_2 \equiv \dots \equiv f_d \pmod{2},$$

being  $f_\alpha$  the number of edges in  $F$  coloured  $\alpha \in \{1, \dots, d\}$ . From the Parity Lemma follows that if  $G$  is a  $d$ -regular graph and has a cut  $F \subseteq E(G)$  such that  $|F|$  is odd and strictly less than  $d$ , then  $G$  cannot be  $d$ -edge-colourable. Hence, the restriction of EDGE-COLOURING(5-regular) to graphs with no cut with 1 or 3 edges is as hard as EDGE-COLOURING(5-regular) itself, since for graphs having a cut with 1 or 3 edges the answer of the problem is known — it is *no*. Recall that it can be decided in polynomial time if  $G$  has a cut with 1 or 3 edges [Ford and Fulkerson 1956, Edmonds and Karp 1972].

*Proof of Theorem 3.* Let  $G$  be a 5-regular graph having no cut with 1 or 3 edges. We shall show how to construct in polynomial time a 5-regular graph  $G^\ddagger$  whose connected components are 4-edge-connected in a manner that  $G^\ddagger$  is 5-edge-colourable if and only if  $G$  is 5-edge-colourable. If  $G$  is already 4-edge-connected, then our reduction simply outputs  $G^\ddagger := G$ . If  $G$  is not 4-edge-connected, then all its cuts of cardinality smaller than 4 have 0 or 2 edges. While some connected component of  $G$  still has a cut  $F$  with 2 edges  $uv$  and  $u'v'$ , being  $u$  and  $u'$  on the same side of the cut, we replace the edges  $uv$  and  $u'v'$  with the edges  $uu'$  and  $vv'$  (parallel edges can be created this way, but they are not a problem), as Fig. 2 illustrates. In the end, the result is a 5-regular graph  $G^\ddagger$  whose connected components are 4-edge-connected. Notice that iteratively finding cuts with 2 edges in  $G$  until it has none can be done in polynomial time. Also, since we only consider cuts with 2 edges connecting vertices in the same connected component, the number of such cuts in the graph  $G$  is decremented at each cut considered, so our reduction halts.

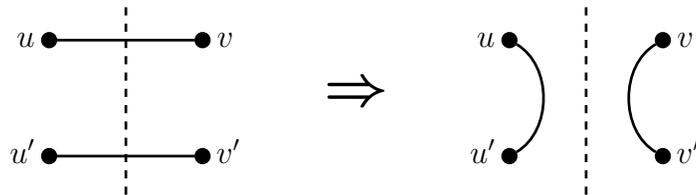


Figure 2. The acting of the reduction over a cut with two edges

We shall prove that  $G$  is 5-edge-colourable if and only if  $G^\ddagger$  is 5-edge-colourable.

- If  $G$  admits a 5-edge-colouring, then, for every cut  $F = \{uv, u'v'\}$  considered in an iteration of the reduction, the Parity Lemma guarantees that  $uv$  and  $u'v'$  are coloured the same. Hence, we can transfer this colour to the edges  $uu'$  and  $vv'$  and this leads to the construction of a 5-edge-colouring of  $G^\ddagger$  once all cuts with 2 edges have been considered.
- Conversely, if  $G^\ddagger$  admits a 5-edge-colouring, then we consider, one at a time, every cut  $F = \{uv, u'v'\}$  taken in an iteration of the reduction, transferring the colour of the edges  $uu'$  and  $vv'$  created by the reduction to the edges  $uv$  and  $u'v'$ , thus yielding the construction of a 5-edge-colouring of  $G$ . Clearly, this works if  $uu'$  and  $vv'$  are coloured the same, but if they are not, we recall that both sides of the

cut have been disconnected by the reduction, so we can permute the names of the colours in one of the sides in order to get the same colour on both  $uu'$  and  $vv'$ .  $\square$

*Proof of Theorem 1.* Since 5-snarks are by definition the negative instances of the problem EDGE-COLOURING(5-regular, 4-edge-connected), if they do not exist then this problem admits an  $O(1)$ -time algorithm, which by Theorem 3 implies  $\mathcal{P} = \mathcal{NP}$ .  $\square$

Corollary 2 follows from the classical Fortune–Mahaney’s Theorem on the computational complexity of sparse languages [Fortune 1979, Mahaney 1982]. A language  $L$  is said to be *sparse* if there is a polynomial  $p(n)$  such that the number of words of length  $n$  belonging to  $L$  is bounded above by  $p(n)$  for any  $n$ . The Fortune–Mahaney’s Theorem states that if any sparse language is  $\mathcal{NP}$ -complete or  $\text{co}\mathcal{NP}$ -complete, then  $\mathcal{P} = \mathcal{NP}$ .

*Proof of Corollary 2.* Follows by observing that the proof of Theorem 1 immediately implies that the language of the 5-snarks is  $\text{co}\mathcal{NP}$ -complete, so it cannot be sparse.  $\square$

We conclude observing the importance of 5-snarks in the context of the Overfull and the 1-Factorisation Conjectures, as briefly discussed in Section 1, and encouraging further investigation on these graphs and other interesting questions which arise: *Do 5-snarks contain 3-snarks as subgraphs? Do they allow infinite family constructions?*

**Observation 4.** *No 5-snark can be SO.*

*Proof.* A graph  $G$  on  $n$  vertices is *overfull* if and only if  $\sum_{u \in V(G)} (\Delta(G) - d_G(u)) \leq \Delta(G) - 2$  and  $n$  is odd [Niessen 1994]. Ergo, if a 5-snark is *SO*, then it has an overfull subgraph  $H$  with  $\Delta(H) = 5$  and, since 5-regular graphs have even order,  $V(H) \neq V(G)$ . Further assuming, without loss of generality, that  $H$  is induced by  $V(H) =: U$ , we have that  $s := \sum_{u \in V(H)} (\Delta(H) - d_H(u))$  is the number of edges  $uv \in E(G)$  with  $u \in U$  and  $v \notin U$ , so  $s \leq 3$  by the overfullness of  $H$ , but  $s \geq 4$  by Definition 1, a contradiction.  $\square$

## Acknowledgements

We heartily express our gratitude to the anonymous referee for the suggestions given, and to CNPq (through grant 428941/2016-8) for the partial support.

## References

- Appel, K. and Haken, W. (1977). Every planar map is four colorable. Part I: discharging. *Illinois J. Math.*, 21:429–490.
- Appel, K., Haken, W., and Koch, J. (1977). Every planar map is four colorable. Part II: reducibility. *Illinois J. Math.*, 21:491–567.
- Arora, S. and Borak, B. (2007). *Computational Complexity: A Modern Approach*. Princeton.
- Blanuša, D. (1946). Problem četiriju boja. *Glasnik. Mat. Fiz. Astr. Ser. II*, 1:31–42.
- Chladný, M. and Škoviera, M. (2010). Factorisation of snarks. *Electron. J. Combin.*, 17(#R32).
- Cook, S. A. (1971). The complexity of theorem-proving procedures. In *Proc. 3<sup>rd</sup> Annual ACM Symposium on Theory of Computing (STOC '71)*, pages 151–158, New York. ACM.

- Descartes, B. (1948). Network-colourings. *The Mathematical Gazette*, 32:67–69.
- Diestel, R. (2010). *Graph Theory*. Springer, 4<sup>th</sup> edition.
- Edmonds, J. and Karp, R. M. (1972). Theoretical improvements in algorithmic efficiency for network flow problems. *J. ACM*, 19:248–264.
- Ford, L. R. and Fulkerson, D. R. (1956). Maximal flow through a network. *Can. J. Math.*, 8:399–404.
- Fortune, S. (1979). A note on sparse complete sets. *SIAM J. Comput.*, 5:431–433.
- Fu, Y. and Ma, Y. (2013). *Graph Embedding for Pattern Analysis*. Springer.
- Gardner, M. (1976). Mathematical games. *Scientific American*, 4(234):126–130.
- Holyer, I. (1981). The  $\mathcal{NP}$ -completeness of edge-colouring. *SIAM J. Comput.*, 10(4):718–720.
- Isaacs, R. (1975). Infinite families of non-trivial trivalent graphs which are not Tait-colorable. *Amer. Math. Monthly*, 82:221–239.
- Karp, R. M. (1972). Reducibility among combinatorial problems. In Miller, R. E. and Thatcher, J. W., editors, *Complexity of Computer Computations*, pages 85–103. Plenum.
- Kempe, A. B. (1886). A memoir on the theory of mathematical form. *Philos. Trans. Royal Soc.*, 177:1–70.
- Leven, D. and Galil, Z. (1983).  $\mathcal{NP}$ -completeness of finding the chromatic index of regular graphs. *J. Algorithms*, 4:35–44.
- Lewis, R. M. R. (2016). *A Guide to Graph Colouring*. Springer.
- Mahaney, S. R. (1982). Sparse complete sets of  $\mathcal{NP}$ : Solution of a conjecture by Berman and Hartmanis. *J. Comput. Syst. Sci.*, 25:130–143.
- Niessen, T. (1994). How to find overfull subgraphs in graphs with large maximum degree. *Discrete Appl. Math.*, 51:117–125.
- Petersen, J. (1898). Sur le théorème de Tait. *L'Intermédiaire des Mathématiciens*, 5:225–227.
- Robinson, R. W. and Wormald, N. C. (1994). Almost all regular graphs are hamiltonian. *Random Struct. Algor.*, 5:363–374.
- Skupień, Z. (2007). Exponentially many hypohamiltonian snarks. *Electron. Notes Discrete Math.*, 28:417–424.
- Tait, P. G. (1878–1880). On the colouring of maps. *Proc. Roy. Soc. Edinburgh Sect. A*, 10:501–503, 729.
- Vizing, V. G. (1964). On an estimate of the chromatic class of a  $p$ -graph (in Russian). *Diskret. Analiz.*, 3:25–30.
- Zatesko, L. M. (2018). *Novel Procedures for Graph Edge-colouring*. PhD thesis, Federal University of Paraná.





Departamento Acadêmico de Informática  
Universidade Tecnológica Federal do Paraná