



WPCCG'2017

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

Universidade Tecnológica Federal do Paraná

Este volume contém os artigos apresentados no WPCCG'2017: II Workshop de Pesquisa em Computação dos Campos Gerais, ocorrido nos dias 04 e 05 de outubro de 2017, 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) do Câmpus Ponta Grossa da UTFPR, o evento está em sua segunda edição.

Ao todo, o WPCCG'2017 teve 26 submissões. Cada submissão foi avaliada por, ao menos, três revisores das áreas da Computação. Dos artigos submetidos, 18 foram aceitos para apresentação e publicação neste volume.

O comitê de programa teve participação de docentes e pesquisadores dos diversos câmpus da UTFPR e de outras instituições, como CEFET-RJ, FURG, IFRS, INMetro, Itaipu Binacional, PUC-PR, PUC-RS, Télécom Paris Tech, UERJ, UFF, UFFS, UFG, UFLA, UFPel, UFPR e UFRRJ.

O gerenciamento de toda conferência foi feito por meio do sistema *EasyChair*.

18 de outubro de 2017
Ponta Grossa, Paraná, Brasil

Gleifer Vaz Alves
Sheila Morais de Almeida
André Pinz Borges

Comitê de Programa

Alexandre Dos Santos Pacheco	ITAIPU Binacional
André Luiz Pires Guedes	Universidade Federal do Paraná (UFPR)
André Pinz Borges	Universidade Tecnológica Federal do Paraná (UTFPR)
André Roberto Ortoncelli	Universidade Tecnológica Federal do Paraná (UTFPR)
André Vignatti	Universidade Federal do Paraná (UFPR)
Augusto Foronda	Universidade Tecnológica Federal do Paraná (UTFPR)
Carlos Eduardo Pantoja	Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (CEFET/RJ)
Claudio Leones Bazzi	Universidade Tecnológica Federal do Paraná (UTFPR)
Clayton Kossoski	Universidade Tecnológica Federal do Paraná (UTFPR)
Dênis Lucas Silva	Universidade Tecnológica Federal do Paraná (UTFPR)
Diana Adamatti	Universidade Federal do Rio Grande (FURG)
Eleandro Maschio	Universidade Tecnológica Federal do Paraná (UTFPR)
Erikson Freitas de Moraes	Universidade Tecnológica Federal do Paraná (UTFPR)
Evandro Miguel Kuszera	Universidade Tecnológica Federal do Paraná (UTFPR)
Fabiano Oliveira	Universidade do Estado do Rio de Janeiro (UERJ)
Fernanda Couto	Universidade Federal Rural do Rio de Janeiro (UFRRJ)
Flávia Haddad	Universidade Tecnológica Federal do Paraná (UTFPR)
Gabriel Sobral	Universidade Federal do Paraná (UFPR)
Gleifer Vaz Alves	Universidade Tecnológica Federal do Paraná (UTFPR)
Graçaliz Dimuro	Universidade Federal do Rio Grande (FURG)
Hebert Coelho	Universidade Federal de Goiás (UFG)
Heitor Murilo Gomes	Pontifícia Universidade Católica do Paraná (PUC-PR)/TPT - Télécom ParisTech
Ionildo José Sanches	Universidade Tecnológica Federal do Paraná (UTFPR)
Jean Paul Barddal	Pontifícia Universidade Católica do Paraná (PUC-PR)
Leandro Zatesko	Universidade Federal da Fronteira Sul (UFFS)
Leyza Dorini	Universidade Tecnológica Federal do Paraná (UTFPR)
Luciana Foss	Universidade Federal de Pelotas (UFPel)
Luciene Marin	Universidade Tecnológica Federal do Paraná (UTFPR)
Lucio Rocha	Universidade Tecnológica Federal do Paraná (UTFPR)
Luerbio Faria	Universidade Federal do Rio de Janeiro (UFRJ)
Luís Felipe Ignácio Cunha	Universidade Federal Fluminense (UFF)
Luiz Carlos Pessoa Albini	Universidade Federal do Paraná (UFPR)
Luiz Gomes-Jr	Universidade Tecnológica Federal do Paraná (UTFPR)
Márcia Cappelle	Universidade Federal de Goiás (UFG)
Marilton Sanchothene de Aguiar	Universidade Federal de Pelotas (UFPel)
Marina Groshaus	Universidade Federal do Paraná (UFPR)/Universidad de Buenos Aires (UBA)
Mauren Louise Sguario	Universidade Tecnológica Federal do Paraná (UTFPR)

Nádia Puchalski Kozievitch	Universidade Tecnológica Federal do Paraná (UTFPR)
Neumar Malheiros	Universidade Federal de Lavras (UFLA)
Osmar Dordal	Pontifícia Universidade Católica do Paraná (PUC-PR)
Pedro Luiz de Paula	Universidade Tecnológica Federal do Paraná (UTFPR)
Rafael H. Bordini	Pontifícia Universidade Católica do Rio Grande do Sul (PUC-RS)
Raphael Machado	Instituto Nacional de Metrologia, Qualidade e Tecnologia (INMetro)
Raquel Barbosa	Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS)
Raquel Bravo	Universidade Federal Fluminense (UFF)
Renato Carmo	Universidade Federal do Paraná (UFPR)
Richard Ribeiro	Universidade Tecnológica Federal do Paraná (UTFPR)
Richardson Ribeiro	Universidade Tecnológica Federal do Paraná (UTFPR)
Samuel Bellido Rodrigues	Universidade Tecnológica Federal do Paraná (UTFPR)
Sarah Sakamoto	Universidade Tecnológica Federal do Paraná (UTFPR)
Sediane Lunardi	Universidade Tecnológica Federal do Paraná (UTFPR)
Sheila Moraes de Almeida	Universidade Tecnológica Federal do Paraná (UTFPR)
Simone Aires	Universidade Tecnológica Federal do Paraná (UTFPR)
Simone Costa	Universidade Federal de Pelotas (UFPel)
Simone de Almeida	Universidade Tecnológica Federal do Paraná (UTFPR)
Simone Nasser Matos	Universidade Tecnológica Federal do Paraná (UTFPR)
William Alberto Cruz Castañeda	Universidade Tecnológica Federal do Paraná (UTFPR)

Sumário

Developing a secure SQL/key-value translation service	1
Davi Boberg, Luiz Gomes-Jr, Marcelo Rosa e Keiko Fonseca	
Metodologia para Geração de Mapa de Profundidade Térmico para Detecção de Pedestres	5
Felipe Bueno, Erikson Freitas de Moraes e Max Mauro Dias Santos	
Algoritmos de Varredura para Listagem de Bicliques Maximais em Modelos Bipartidos de Intervalos e Arco-Circulares	9
Edmilson Pereira da Cruz, Marina Groshaus e André Luiz Pires Guedes	
Utilização de um simulador para representação de estratégias de um agente inteligente no desvio de obstáculos	13
Vinicius Custodio e Gleifer Vaz Alves	
Avaliação da Usabilidade da Interface Gráfica de Gerenciamento de um Estacionamento Inteligente	17
Felipe Felix Ducheiko, Gleifer Vaz Alves, Geraldo Ranthum	
Utilização de lógica temporal-deontica para representação do comportamento de um agente racional no controle de um veículo autônomo	21
Vithor Tozetto Ferreira e Gleifer Vaz Alves	
Portal dos Estagiários: Sistema de gestão do estágio interno no CEFET-RJ/NF	25
Cleyton da Cunha Gomes, Gustavo Müller Moreira, Lúcio Folly Sanches Zebendo, João Victor Guinelli da Silva, Bruno P. Toledo Freitas e Anderson Fernandes Souza	
Explorando o Transporte de Agentes Cognitivos entre Sistemas Multi-Agentes Distintos	29
Vinicius Souza de Jesus e Carlos Eduardo Pantoja	
Modelagem do Subsistema de Administração Imobiliário Baseado em Linhas de Produto	33
Silvia Ribeiro Mantuani, Fernando Henrique Campos e Vinícius Camargo Andrade	
Participação Feminina no Cinema Brasileiro: Análise Através de Redes Sociais	37
Renato Silva de Melo, André Luis Vignatti, Arthur Pecheba da Costa e Felipe do Nascimento	
Obtenção do mapa de disparidade em imagens capturadas através de software próprio	41
William Takeshi Omoto, Cauê Felchar e Erikson Freitas de Moraes	
Adjacent vertex distinguishing edge coloring on complete split graphs and split-indifference graphs	45
Thamirys Moreira dos Santos Rauch e Sheila Moraes de Almeida	
Criticalidade arco-íris dos grafos resultantes de produto cartesiano de ciclos e caminhos	49
Aleffer Rocha e Sheila Moraes de Almeida	
Coloração total distinta na vizinhança em potências de caminhos	53
Pedro Henrique Salgado, Mayara Midori Omai e Sheila Moraes de Almeida	
Coloração total distinta na vizinhança em grafos 4-partidos completos	57
Matheus Scaketti e Sheila Moraes de Almeida	
Assinatura Comportamental e Detecção de Anomalias Utilizando <i>K-means</i>	61
Wagner Senger e Lourival Aparecido de Góis	
Análise de Métricas para Recuperação de Casos visando à aplicação no Problema de Condução de Trens de Carga	65
Matheus Streisky, Richardson Ribeiro e André Pinz Borges	
On a conjecture on edge-colouring join graphs	69
Leandro M. Zatesko, Renato Carmo e André Luiz Pires Guedes	

Índice de Autores

Almeida, Sheila Morais de	45, 49, 53, 57
Alves, Gleifer Vaz	13, 17, 21
Andrade, Vincius Camargo	33
Boberg, Davi	1
Borges, André Pinz	65
Bueno, Felipe	5
Campos, Fernando Henrique	33
Carmo, Renato	69
Costa, Arthur Pecheba da	37
Cruz, Edmilson Pereira da	9
Custodio, Vinicius	13
Ducheiko, Felipe Felix	17
Felchar, Cauê	41
Ferreira, Vithor Tozetto	21
Fonseca, Keiko	1
Freitas, Bruno P. Toledo	25
Góis, Lourival Aparecido de	61
Gomes, Cleyton da Cunha	25
Gomes-Jr, Luiz	1
Groshaus, Marina	9
Guedes, André Luiz Pires	9, 69
Jesus, Vinicius Souza de	29
Mantuani, Silvia Ribeiro	33
Melo, Renato Silva de	37
Morais, Erikson Freitas de	5, 41
Moreira, Gustavo Müller	25
Nascimento, Felipe do	37
Omai, Mayara Midori	53
Omoto, William Takeshi	41

Pantoja, Carlos	29
Ranthum, Geraldo	17
Rauch, Thamirys Moreira dos Santos	45
Ribeiro, Richardson	65
Rocha, Aleffer	49
Rosa, Marcelo	1
Salgado, Pedro Henrique	53
Santos, Max Mauro Dias	5
Scaketti, Matheus	57
Senger, Wagner	61
Silva, João Victor Guinelli da	25
Souza, Anderson Fernandes	25
Streisky, Matheus	65
Vignatti, André Luis	37
Zatesko, Leandro M.	69
Zebendo, Lúcio Folly Sanches	25

Índice de Palavras-chave

4-partidos completos	57
Adjacent vertex distinguishing edge coloring	45
Adjacent vertex distinguishing total coloring	53, 57
Agentes	21
Agentes inteligentes	13
Agentes móveis	29
Agents	21
Ambiente aberto	29
Anomaly detection	61
Arduino	25
Assinatura comportamental	61
Autonomous vehicles	21
Avaliação de usabilidade	17
Behavior's pattern	61
Behavior's signature	61
Bicliques	9
Calibração de câmeras	41
Cartesian Product	49
Case-based reasoning	65
Cidade inteligente	17
Cinema brasileiro	37
Cloud	1
Clusterizing	61
Clusterização	61
Coloração arco-íris	49
Coloração total distinta na vizinhança	53, 57
Complete 4-partite graphs	57
Complete multipartite graphs	57
Computer vision	5
Condução de trens	65
Detecção de anomalias	61
Detecção de pedestres	5
Edge-colouring	69
Estacionamento inteligente	17
Estereoscopia	5

Gêneros	37
Grafos arco-íris críticos	49
Grafos bi-arco-circulares	9
Grafos de bi-intervalos	9
Grafos indiferença	53
Grafos multipartidos completos	57
Human-computer interaction	17
Indifference graph	45, 53
Interação humano-computador	17
Join graphs	69
k-means	61
k-NN	65
Linhas de produto de software	33
Lógica temporal-deontica	21
Mapa de disparidade	41
Mobile agents	29
Multi-agent system	17
Open environment	29
Open MAS	29
Padrões de comportamento	61
Par estéreo	41
Pedestrian detection	5
Potências de caminhos	53
Powers of paths	53
Produto cartesiano	49
Protocolo de comunicação	13
Raciocínio baseado em casos	65
Rainbow coloring	49
Rainbow critical graphs	49
Real estate system	33
Recolouring procedures	65
Redes sociais	37
RFID	25
Secure databases	1
SGX	1
Simulação	13
Sistema imobiliário	33
Sistema multiagentes	17
Sistemas web	25
SMA aberto	29
Smart city	17
Smart parking	17
Software product line	33

Split graph	45
Stereoscopy	5
Temporal deontic-logic	21
Termografia	5
Thermography	5
Train driving	65
UML Components	33
Usability evaluation	17
Veículos autônomos	13, 21
Visão computacional	5
Web systems	25

Developing a secure SQL/key-value translation service

Davi Boberg
Universidade Tecnológica
Federal do Paraná
Curitiba, Brazil
daviboberg@alunos.utfpr.edu.br

Luiz Gomes-Jr
Universidade Tecnológica
Federal do Paraná
gomesjr@dainf.ct.utfpr.edu.br

Marcelo Rosa
Universidade Tecnológica
Federal do Paraná
Curitiba, Brazil
mrosa@utfpr.edu.br

Keiko Fonseca
Universidade Tecnológica
Federal do Paraná
Curitiba, Brazil
keiko@utfpr.edu.br

RESUMO

As cloud services are becoming an alternative for IT infrastructures in many organizations, data privacy guarantees become a priority. This paper presents the development of a secure database system using Intel's SGX trusted platform. SGX provides hardware-based processing privacy offering protection for a wide range of sophisticated attacks. We propose a modular, multi-service architecture that is well suited to the advantages and limitations of the SGX platform. This paper focuses on the ongoing implementation of our SQL translation service.

Palavras-chave

Secure Databases; Cloud; SGX

1. INTRODUCTION

The continuous development and growing availability of cloud services are changing the relationship between organizations and their technological infrastructures. Several commercial cloud providers currently offer products that provide flexible scalability, high availability, and security to small and large companies [3]. These products have become very attractive given the potential to reduce costs when contrasted with in-house development and support of IT infrastructures.

Cloud services, however, can introduce security risks for their clients: private data need to be transferred to the cloud and become vulnerable to internal and external attacks in the adopted platform. Cloud service providers often offer a high level of security from external attacks, but internal deliberate attacks are harder to predict and prevent. An internal personnel (e.g. a network administrator) could have full privileges to access the sensitive information. Encryp-

ting sensitive information would provide some protection, but attackers with physical access to the servers could still perform a physical attack to access decrypted information in the memory

To address these shortcomings, we propose a secure database architecture based on the Intel's SGX¹ technology [2]. SGX, a kind of trusted platform module, provides secure enclaves where code runs in a protected and encrypted region of the computer memory, even protected from attacks related to physical access to that memory area. Unfortunately, the current SGX standard has memory size limitations that makes it impossible to implement a full-blown cloud and BigData-ready database system inside a single enclave. Our strategy is therefore to modularize the architecture and provide different levels of security and query capabilities. Here we describe our architecture for a secure SQL-based cloud database based on a secure key-value store, namely ChocolateCloud² (Section 2). We also describe ongoing efforts for the implementation of the system as well as preliminary experiments (Section 3).

1.1 Related Work

There has been several proposals for secure databases employing traditional encryption strategies [7, 1]. While these proposals offer a reasonable level of security for data stored in secondary memory, the systems are still vulnerable to physical attacks where the opponent has direct access to the computer's primary memory. Homomorphic encryption [4], a theoretical solution for this problem, has a performance penalty of orders of magnitude, making it unsuitable for cloud and BigData workloads.

Hardware support for encrypted in-memory processing is an answer for the aforementioned performance issues. Intel's SGX [2] is currently the most comprehensive and commercially available implementation of a trusted platform module (TPM). As a TPM, it has abilities of (i) containing a hardware encoded cryptographic key (known only by the module); (ii) remote attestation (which means two enclaves can trust each other through a modified Sigma protocol over a Diffie-Hellman Key Exchange, or DHKE, allowing them to exchange cryptographic keys), (iii) sealing data (which me-

WPCCG '17 Outubro, 2017, Ponta Grossa, Paraná, Brasil

¹<https://software.intel.com/en-us/isa-extensions/intel-sgx>
²<http://www.chocolate-cloud.cc/>

ans once a data is sealed by the module, it can only unsealed by that module). To the developer’s perspective, it ensures that nothing can tamper with the code running inside such modules. It also changes the way software are developed in order to run safely from external attacks.

A practical approach to build a secure database is to use simpler models for querying and storage, which reduces the amount of code to be protected. The most basic model of NoSQL database is the key-value, which uses arbitrary keys to identify values stored as uninterpreted byte arrays. This was the approach taken by the Chocolate Cloud project [8]. The service offers secure processing and communication through the use of the SGX architecture and guarantees confidentiality for the storage using network coding algorithms.

Network coding is based on splitting the information of an item into parts stored at different locations. The reconstruction of the original item can only be done by retrieving multiple parts. This strategy guarantees confidentiality even when some of the storage servers are compromised.

In our proposal we are using the Chocolate Cloud service to provide secure data storage and retrieval. We are then building more expressive modules to provide more flexibility while constraining intramodule complexity and system vulnerability.

There has been other proposals to bridge the relational model with simpler NoSQL models. For example, [6] proposes mappings for key-value stores and document databases. Another approach is the new class of databases labeled as NewSQL [5], which focus on providing BigData capable systems that retain SQL and ACID capabilities. These proposals however do not address the security issues described here.

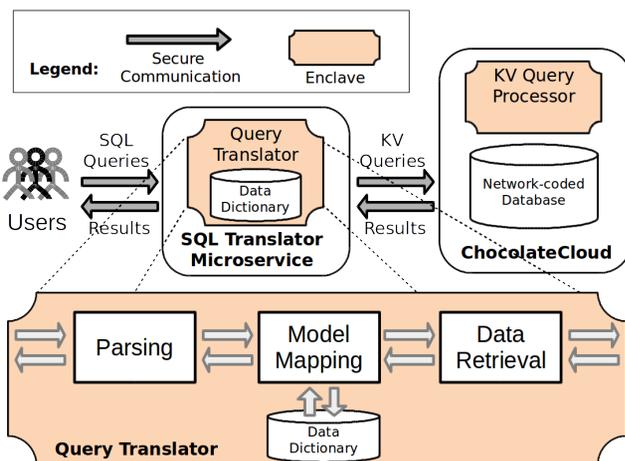


Figure 1: Service architecture

2. ARCHITECTURE AND SQL TRANSLATION

We are building our secure database service in a modular architecture. Independent modules are easier to fit in the enclave’s limited memory and can be replicated for scalability. The focus of this paper is the *SQL translation engine*, which handles SQL queries and interacts with the underlying key-value store module to answer requests. We are also de-

```

a) INSERT INTO reading
  ( <clientID>, <timestamp>, <read 1> ... <read 30>);
b) key: reading|<clientID>|<timestamp>
   value: [<read 1> ... <read 30>]

```

Figure 2: Example an insert query (a) mapping into a key-value pair (b). ClientID and timestamp compose the primary key of the table and become part of the mapped key. The table has 30 attributes omitted for clarity.

veloping a data anonymization module for access control and on-the-fly data anonymization (not covered in this paper).

Figure 1 shows the internal parts of the SQL translation module and its interactions with the key-value store. Data definition (DDL) queries are handled by the parser and have their settings stored in the data dictionary. The schema and keys define how each tuple will later be mapped to the key-value model.

When a data manipulation (DML) query is received, a dictionary lookup is performed to determine the mappings between the models. For insertion queries, a key is composed by a concatenation of the table name and the primary key(s) (a predefined separator is used for readability). Tuple attributes are also concatenated to be stored as the value. Figure 2a shows an insert query for a table named *reading* that has the two first attributes as primary keys. Figure 2b shows the resulting key-value pair generated. The key/value pair is then submitted to the Chocolate Cloud store. For selection queries the key is assembled with the table name and the keys presented in the query, potentially with a wildcard character (*) to retrieve all keys matching the prefix. Currently only queries with conditions over the primary keys are supported. More flexible selects and joins are upcoming developments.

A wider coverage of the SQL language will be implemented with extra indexes for non-key attributes – stored in the key-value database. Joins on the primary keys can be implemented efficiently since the Chocolate Cloud store returns keys in alphabetical order, enabling the use of a merge-join algorithm. Other joins can be implemented using the aforementioned extra indexes or full table scans when indexes are not available.

Table 1: Performance tests for select and insert queries (in seconds)

Test	Processing time	Total time	Total time/query
10 Selects	0.000171	3.0767	0.308
100 Selects	0.000189	28.5287	0.285
1000 Selects	0.000172	286.5795	0.286
10 Inserts	0.001626	4.1538	0.415
100 Inserts	0.013217	41.3288	0.413
1000 Inserts	0.13250	418.5262	0.418

3. EXPERIMENTS

We are developing and testing our solution based on a real scenario and real data. The scenario is that of an electric

power company that collects usage summaries several times per day for thousands of customers. The data are aggregated in our database service and consumed by operational and analysis applications such as billing, fraud detection, and quality of service evaluation.

In the first prototype of our translation module, the code was implemented in ANSI C. The module issues REST quests to a ChocolateCloud database hosted in a cloud service in Europe. Here we focus on initial tests of the translation engine without the use of the SGX environment.

For these preliminary tests we used simple queries for insertion (Figure 2) and selection of usage summaries. The goal is to show that (i) our translation module has a small impact in the overall query execution and (ii) that performance scales linearly with the number of queries.

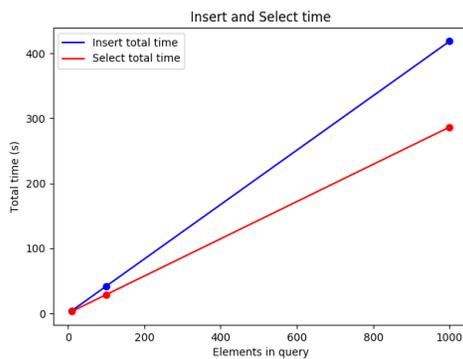


Figure 3: Evaluation of performance (total time) for INSERT and SELECT statements

Table 1 shows performance tests for select and insert queries over our service. We show results for increasing numbers of elements in the queries (10, 100, 1000). The column *processing time* represents the time taken in the translation of the queries, including identifying query type, parsing, retrieving metadata and building the respective key-value queries. The column *total time* represents the time to process the query, retrieve and assemble the results. This time is much higher because currently the key-value database is located in a cloud server in Europe. In the future we plan to have all modules running on the same cloud infrastructure and expect significant reductions in query times. Finally, the last column shows the total time divided by the number of queries. All values are averages over 10 runs of the experiments. The numbers show that the translation has little impact on query performance. These results were expected since our implementation is in C and the algorithms for processing simple queries are straightforward. The results will, however, serve as the baseline for the next iterations of our implementation, especially when we embed the processing in the SGX enclave and enable more complex queries.

Figure 3 shows results for total query processing times. The graph shows the time taken for processing queries with 10, 100 and 1000 elements. The lines, drawn for reference, shows the linear trend for scalability. Figure 4 shows results for processing time (excluding time to query the key-value store). Again, time for inserts grows linearly. Processing time for selects remain constant because there is only one query to be processed (only the size of the results change).

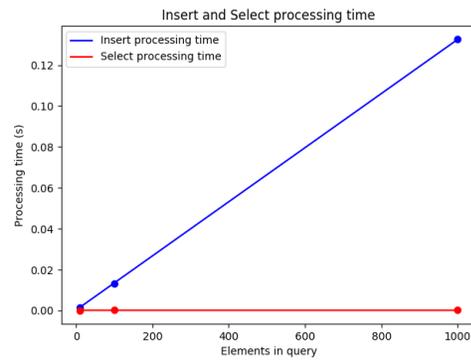


Figure 4: Evaluation of performance (processing time) for INSERT and SELECT statements

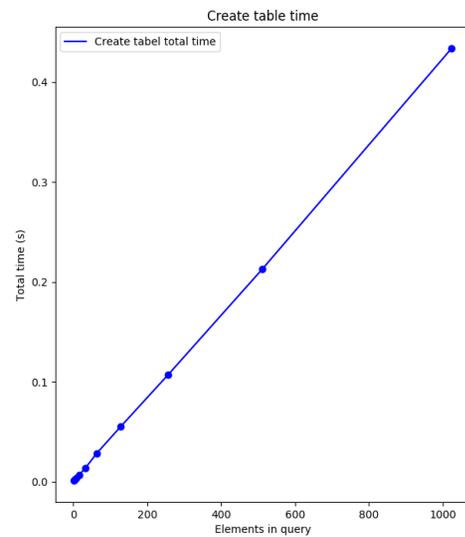


Figure 5: Performance of CREATE TABLE statements

Finally, Figure 5 shows processing times for create table queries (which currently only affect the internal data dictionary). Again, these results were expected given the nature of the algorithms and will serve as baseline for future tests.

4. CONCLUSION

This paper described our ongoing work towards the development of a secure service to enable SQL queries to a key-value store. The service is being integrated in Intel's SGX platform, which provides hardware-based privacy guarantees. Our service translates SQL queries into key-value calls to a Chocolate Cloud storage service. Chocolate cloud also offers increased levels of security employing both SGX and network coding algorithms.

We are currently working on the integration of our translation service with the SGX library developed by our research group. The next steps include supporting more elaborate SQL queries, query processing optimization, and per-

formance tests. We are also working on another service to complement the SQL translation that will be responsible for role-based anonymization and access control.

5. ACKNOWLEDGMENTS

This research is being performed in the context of the SecureCloud project. The SecureCloud project has received funding from the European Union's Horizon 2020 research and innovation programme and was supported by the Swiss State Secretariat for Education, Research and Innovation (SERI) under grant agreement number 690111. This work was partially funded by the EU-BR SecureCloud project (MCTI/RNP 3rd Coordinated Call).

6. REFERÊNCIAS

- [1] I. Basharat, F. Azam, and A. W. Muzaffar. Database security and encryption: A survey study. *International Journal of Computer Applications*, 47(12), 2012.
- [2] V. Costan and S. Devadas. Intel sgx explained. Technical Report 2016/086, Cryptology ePrint Archive, 2016.
- [3] M. Cusumano. Cloud computing and saas as new computing platforms. *Commun. ACM*, 53(4):27–29, Apr. 2010.
- [4] C. Gentry. Fully homomorphic encryption using ideal lattices. In *Proceedings of the Forty-first Annual ACM Symposium on Theory of Computing, STOC '09*, pages 169–178, New York, NY, USA, 2009. ACM.
- [5] A. Pavlo and M. Aslett. What's really new with newsql? *SIGMOD Record*, 45(2):45–55, 2016.
- [6] G. A. Schreiner, D. Duarte, and R. dos Santos Mello. Sqltokeynosql: a layer for relational to key-based nosql database mapping. In G. Anderst-Kotsis and M. Indrawan-Santiago, editors, *iiWAS*, pages 74:1–74:9. ACM, 2015.
- [7] E. Shmueli, R. Vaisenberg, Y. Elovici, and C. Glezer. Database encryption: an overview of contemporary challenges and design considerations. *SIGMOD Record (ACM Special Interest Group on Management of Data)*, 38(3):29–34, Sept. 2009.
- [8] M. Sipos, P. J. Braun, D. E. Lucani, F. H. P. Fitzek, and H. Charaf. On the effectiveness of recoding-based repair in network coded distributed storage. *Periodica Polytechnica. Electrical Engineering and Computer Science*, 61(1):12–21, 2017. Copyright - Copyright Periodica Polytechnica, Budapest University of Technology and Economics 2017; Last updated - 2017-03-09.

Metodologia para Geração de Mapa de Profundidade Térmico para Detecção de Pedestres

Felipe Bueno

Universidade Tecnológica Federal do Paraná
Av. Monteiro Lobato, s/n – Km 04
Ponta Grossa – PR – Brasil
felipebueno@alunos.utfpr.edu.br

Erikson Freitas de Moraes

Universidade Tecnológica Federal do Paraná
Av. Monteiro Lobato, s/n – Km 04
Ponta Grossa – PR – Brasil
emorais@utfpr.edu.br

Max Mauro Dias Santos

Universidade Tecnológica Federal do Paraná
Av. Monteiro Lobato, s/n – Km 04
Ponta Grossa – PR – Brasil
maxsantos@utfpr.edu.br

RESUMO

Estatísticas demonstram a abrangência do número de acidentes envolvendo pedestres. Uma forma de minimizar esses indicadores é através do uso de funções de auxílio ao condutor. Para tal fim uma das possibilidades consiste em utilizar informações de temperatura e coordenadas tridimensionais da rodovia para identificação dos pedestres. No presente trabalho propõe-se uma metodologia capaz de construir um mapa de profundidade utilizando câmeras térmicas organizadas em um par estéreo, o qual indica as distâncias dos pontos observados. Além de outras utilidades, o mapa de profundidade térmico (MPT) representa uma ferramenta importante para sistemas de auxílio ao motorista.

Palavras-chave

Estereoscopia; Termografia; Detecção de Pedestres; Visão Computacional.

ABSTRACT

Statistics evidence the coverage of the number of accidents relating to pedestrians. A way to minimize these indicators is through the use of drive assistance functions. For this purpose one of the possibilities consists in using temperature info and road tridimensional coordinates to pedestrian identification. In the present paper it is proposed a methodology to build a depth map using thermal cameras arranged in a stereo pair, which indicates the distances of the observed points. Besides other applications, the thermal depth map (TDM) represents an important tool for driver assistance systems.

Keywords

Stereoscopy; Thermography; Pedestrian Detection; Computer Vision.

1. INTRODUÇÃO

No Brasil, cerca de 90% dos acidentes de trânsito têm como causa principal falha dos condutores [12]. Nos Estados Unidos, essa estatística chega a aproximadamente 94% [10]. Os fatores variam entre imprudência (desrespeito à legislação, como realização de ultrapassagens em lugares proibidos) e imperícia (falta de experiência do condutor, o que leva o mesmo a não reagir de forma adequada perante situações de risco no trânsito). Segundo a Organização Mundial de Saúde, cerca de 1,25 milhões de pessoas morrem nas estradas todos os anos [4]. Dentro da estatística de acidentes com veículos, nos EUA cerca de 25% deles resultam em morte ou ferimento de pedestres [11].

Uma forma de minimizar essa estatística consiste em utilizar sistemas de auxílio ao motorista capazes de atuar na tomada de decisão do condutor. Esses sistemas são chamados de ADAS (*Advanced Driver Assistance Systems*), e são capazes de gerar alertas em situações de risco ou até mesmo assumir o controle do veículo, como no caso da função de estabilidade de faixa (LKA – *Lane Keep Assist*) [1, 3].

No contexto de acidentes envolvendo pedestres, pode-se encarar como um desafio a criação de sistemas que atuem em sua detecção. Esses tipos de sistemas podem ser usados para alertar o condutor e/ou atuar no veículo conduzido, como foi feito em [2], onde a partir de um sistema que recebe como entrada dados de laser e câmera, é emitido um alerta ao condutor caso seja detectado um pedestre em regiões definidas como críticas. Outras abordagens fazem uso de uma câmera térmica, a partir da fusão de imagens térmicas e convencionais aplicadas em cenas estáticas [5], e detecção e rastreamento de pedestres posicionados a longas distâncias usando uma câmera térmica e análise do histograma do gradiente orientado [7]. Outras ainda utilizam combinações entre dois pares de câmeras convencionais e térmicas [8], promovendo uma fusão de dados multimodal e valendo-se da assinatura de calor característica dos seres humanos, realçada em relação ao ambiente.

Além da detecção do pedestre na cena, é interessante recuperar sua posição no espaço, para que o sistema ADAS consiga tomar alguma ação antes que uma possível colisão ocorra. Uma das formas de se obter a posição do ponto observado é a partir da visão estéreo. Nesse sentido, a pesquisa descrita neste trabalho tem por objetivo a geração de um mapa de profundidade a partir de imagens contendo informações de temperatura e organizadas em um par estéreo calibrado. O Mapa de Profundidade Térmico (MTP) representa uma ferramenta importante para sistemas de auxílio ao motorista, principalmente por apresentar as diferentes temperaturas da cena mapeadas a uma estimativa confiável de distância. O resultado dessa fusão de dados é uma visão privilegiada capaz de identificar corpos quentes mapeados em coordenadas 3D, além de não apresentar sensibilidade a interferências e ruídos tais como fumaça, neblina e variações na iluminação.

O trabalho está dividido da seguinte maneira: Na seção 2 são apresentados conceitos necessários para o desenvolvimento da proposta. A seção 3 apresenta a metodologia proposta para cálculo do mapa de profundidade, e na seção 4 é descrita a conclusão e os passos futuros.

2. REFERENCIAL TEÓRICO

Nesta seção serão introduzidos conceitos necessários para melhor entendimento deste trabalho:

A termografia refere-se ao registro da informação térmica. No contexto de processamento de imagens, o produto da termografia é o termograma, o qual é uma imagem que indica a temperatura de cada ponto do ambiente registrado a partir de uma determinada escala (tons de cinza ou paleta de cores).

A Radiação Infravermelha, por sua vez, caracteriza a transferência de energia térmica (calor) através de radiação. Todo objeto com temperatura superior ao zero absoluto emite essa radiação. Quando esta radiação incide sobre um corpo, uma parte é absorvida, outra é refletida e outra passa através do mesmo [20]. Quando toda a radiação é absorvida tem-se o chamado corpo negro.

Emissividade é a capacidade da superfície do material de emitir radiação. É a razão entre a quantidade de energia emitida por um determinado corpo, e seu corpo negro, respeitando-se as mesmas condições de temperatura e comprimento de onda [9]. A emissividade geralmente é um parâmetro de configuração das câmeras térmicas, referente à sensibilidade do sensor.

A Visão Estéreo, ou estereoscopia, permite a obtenção de informações tridimensionais e distância do ponto observador até determinado objeto, a partir de sua observação de dois ou mais pontos de vista [17].

A geometria da visão estereo é chamada de Geometria Epipolar. Tem como base o modelo de câmera *pinhole*, descrito em [6] e ilustrado na Figura 1, que em linhas gerais define que em uma câmara escura com um orifício com diâmetro tendendo a zero (ponto focal), a convergência dos raios de luz captados através deste orifício faz com que seja projetada no fundo da câmara a imagem observada. Esse modelo é importante para entender como uma imagem nas câmeras atuais é formada, em função do que é captado através da lente da câmera. É interessante perceber que a imagem projetada no fundo também pode ser obtida antes de ser invertida pela projeção através do orifício, como pode ser observado na Figura 1, no plano da imagem e plano virtual da imagem. A distância focal, por sua vez, é a distância entre o ponto de convergência da luz (ponto focal) e a projeção da imagem no plano da imagem. Em câmeras convencionais essa distância é determinada a partir do modelo da lente utilizada.

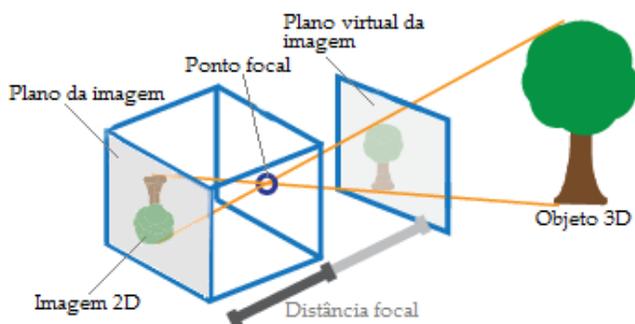


Figura 1: Modelo de câmera *pinhole*. Adaptado de [19].

Com base no entendimento da projeção de uma imagem a partir de uma câmera, pode-se adotar o mesmo modelo, porém para projeções a partir de n câmeras, e então traçar relações geométricas

que nos permitem descobrir informações acerca da posição de um objeto no espaço e suas dimensões. A Figura 2 apresenta um modelo básico da geometria epipolar a partir de dois pontos de vista distintos.

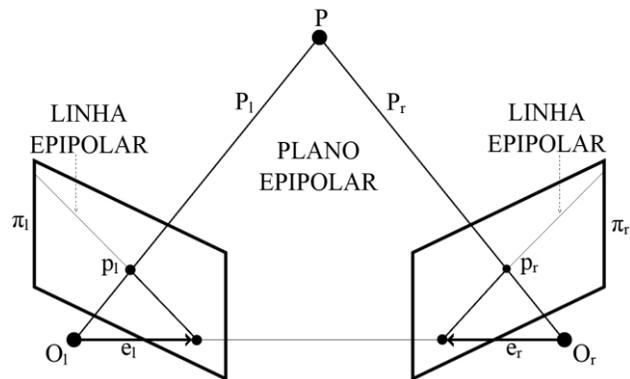


Figura 2: Geometria epipolar. Adaptado de [17].

Baseado na Figura 2, pode-se definir terminologias que serão importantes para a compreensão deste trabalho. Os símbolos O representam o ponto focal, para as câmeras da esquerda e direita. Os símbolos π representam os planos da imagem. Os caracteres e são chamados de epipolos, pois indicam os pontos onde ocorre a interseção entre as projeções dos pontos focais. As linhas representadas por P indicam o mesmo ponto em coordenadas do mundo (tridimensionais), e as linhas indicadas por p indicam as projeções desse ponto nos planos da imagem. As linhas epipolares indicadas na Figura 2 indicam a localização do plano epipolar. Ambas fazem parte da definição de restrição epipolar, a qual significa que para um determinado ponto no plano da imagem de uma das câmeras, o ponto correspondente na outra deve estar ao longo da linha epipolar [17]. A identificação do exato ponto correspondente ao longo da linha epipolar caracteriza o chamado problema de correspondência.

A Calibração de Câmera refere-se ao processo de extração (identificação) dos parâmetros intrínsecos e extrínsecos da câmera. Os parâmetros intrínsecos são distância focal (em pixels), coordenadas do ponto principal (centro do plano da imagem), e tamanho do pixel (vertical e horizontal). Os parâmetros extrínsecos são os valores de rotação e translação que permitem retificar as imagens para um mesmo plano [17].

Matriz Fundamental é a matriz capaz de correlacionar um ponto na imagem A do par estereo com a linha epipolar da imagem B (transformação de espaço), dessa forma otimizando a análise da imagem para resolver o problema de correspondência [6].

O Mapa de Disparidade pode ser caracterizado como uma imagem que denota a diferença da localização de um mesmo ponto tridimensional projetado a partir da perspectiva de pontos de vista distintos [14]. Se um ponto A em uma imagem 1 está localizado na coordenada (100, 20), e B (135, 20) é seu correspondente na imagem 2, a disparidade será de 35 pixels. Dessa forma, no mapa de disparidade, pode ser aplicada uma normalização nos valores de disparidade para que estes assumam um valor no intervalo de 0 a 255 permitindo sua visualização em tons de cinza.

O Mapa de Profundidade, por sua vez, é uma imagem que representa as distâncias dos objetos observados em relação ao

ponto observador, que pode ser considerado como sendo o ponto médio entre as câmeras. Nesse caso a profundidade é o valor da distância de um ponto da cena ao ponto observador e o mapa é uma matriz de profundidades referentes à cena.

3. METODOLOGIA PROPOSTA

Para a obtenção do mapa de profundidade, é necessário completar uma série de etapas, detalhadas na Figura 3.

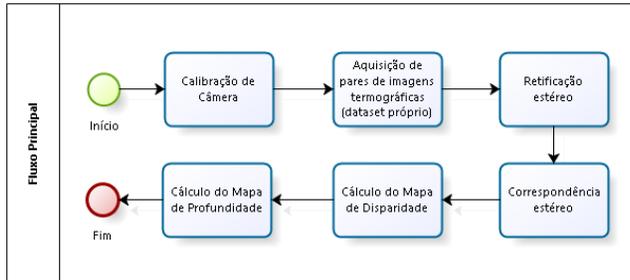


Figura 3: Fluxo da metodologia proposta. Autoria própria.

A primeira etapa apresentada consiste na calibração das câmeras utilizadas. Essa calibração pode ser dividida em geométrica e radiométrica, além da definição dos parâmetros de emissividade, relativos ao sensor térmico da câmera. A calibração radiométrica não faz parte do escopo deste trabalho, portanto não será considerada. A calibração será realizada individualmente em cada câmera que compõe o par estéreo.

O processo pode ser realizado a partir de rotinas que analisam um alvo com um padrão de calibração de geometria conhecida e que possua características visuais que possam ser localizadas facilmente. Normalmente usa-se um objeto planar com dimensões conhecidas tal como um tabuleiro de xadrez, como ilustrado na Figura 4 [16]. Esse padrão também é importante pois é utilizado para a caracterização das distorções radiais presentes na câmera. A análise e extração dessas distorções também não fazem parte do escopo deste trabalho e não serão consideradas.



Figura 4: Alvo de calibração. Autoria de [13].

Considerando que a entrada será composta por imagens que representam um mapa de calor, será necessário utilizar um alvo de calibração apropriado onde seja possível destacar as diferenças de temperatura, como foi feito em [13], onde as regiões de maior calor equivalem aos quadrados pretos da Figura 4.

Com a câmera calibrada, é possível dar início à criação de um *dataset* próprio, que será composto de imagens de pessoas, captadas a partir de pontos fixos. Essas imagens conterão o termograma, que é uma representação 2D em uma escala de cores referente à temperatura medida pela câmera em cada ponto,

também chamadas de imagens térmicas. O resultado dessa etapa consiste em pares de imagens térmicas captadas a partir de duas posições distintas. Após a captura, o próximo passo apresentado na Figura 3 é a retificação estéreo. Esse processo consiste em alinhar as imagens em um mesmo plano de modo que os pares das linhas epipolares conjugadas tornem-se colineares e paralelos a um dos eixos [17]. Com isso, é possível promover um alinhamento 2D, de modo a restringir a busca de pontos de correspondência entre as imagens 1 e 2 ao longo da linha epipolar.

Após a geração do *dataset* com as imagens retificadas, será realizado o processo de correspondência estéreo. Esse processo, consiste em identificar os pixels correspondentes entre as imagens. Existem diversos algoritmos que tratam do problema de correspondência. Contudo, será realizado um levantamento bibliográfico de forma a selecionar uma técnica ou um conjunto de técnicas, de modo a promover uma correspondência em imagens térmicas adequada ao propósito do trabalho.

De posse das posições correspondentes de cada ponto das imagens retificadas, a etapa a seguir refere-se ao cálculo do mapa de disparidade, o qual pode ser representado por uma matriz com as distâncias entre os pontos homólogos nas imagens do par estéreo, ou seja, é uma imagem contendo as distâncias entre os pontos correspondentes identificados na etapa anterior.

O mapa de profundidade, por fim, pode ser obtido a partir do mapa de disparidade, distância focal e distância entre os centros de projeção das câmeras (*baseline*), pois quanto maior a disparidade, mais próximo o objeto está das câmeras, portanto menor sua profundidade. A equação (1) ilustra o cálculo da profundidade em imagens retificadas para cada pixel,

$$z = bf \frac{1}{d} \quad (1)$$

onde d corresponde à disparidade, b à distância *baseline*, f à distância focal e z à profundidade. A Figura 5 ilustra a imagem gerada a partir do mapa de profundidade a partir de pontos de vista distintos. A escala de cinza indica a distância em relação ao ponto observador, onde quanto mais claro mais próximo.

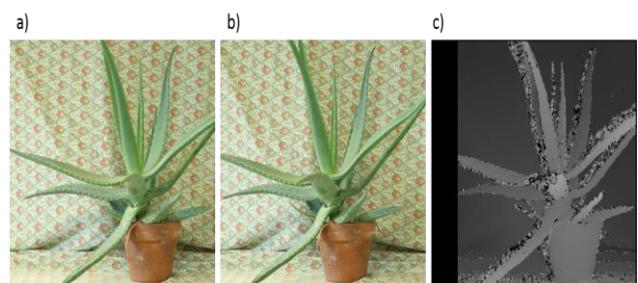


Figura 5: Geração do mapa de disparidade. a) Imagem captada pela câmera da esquerda. b) Imagem captada pela câmera da direita. c) Mapa de disparidade. Autoria própria.

A detecção do pedestre na imagem pode ser feita a partir da aplicação de uma técnica de segmentação que se valha da assinatura térmica característica de seres vivos na imagem. Após isso, a partir do mapa de profundidade calculado e utilizando-se um pixel referente ao pedestre segmentado, obtém-se a distância entre este e o ponto observador (veículo). De posse dessas informações, um sistema ADAS pode alertar o condutor a partir da ativação de um

sinal sonoro, vibrando o volante do veículo ou acionando seu sistema de freios, caso este esteja em rota de colisão, evitando um possível acidente.

4. CONCLUSÃO

O mapa de profundidade representa uma etapa importante em qualquer sistema baseado em geometria estéreo. A partir da metodologia apresentada é possível estabelecer um mapa de profundidade, identificando distâncias, associadas à informação de temperatura em uma cena. No contexto de identificação de pedestres, a aquisição dessas informações pode representar um fator crucial para evitar colisões, diminuindo assim os indicadores referentes a ferimentos ou mortes em acidentes com veículos.

Além da possibilidade de utilização como entrada de dados para uma função ADAS, a informação térmica e de profundidade de uma cena possui aplicações em diversos cenários, como sistemas de vigilância, sistemas de monitoramento de equipamentos [18], auxílio no diagnóstico médico [15], entre outras.

Como ponto de melhoria propõe-se a definição de estratégias que atenuem o impacto da possível baixa eficácia na detecção de pedestres em cenários onde o plano de fundo captado está com temperatura elevada, próxima à do pedestre.

Como passos futuros propõe-se a geração de um modelo 3D a partir dos dados coletados e estabelecimento de métricas para validação de testes e verificação de eficácia. Além disso, propõe-se a geração tridimensional *on the fly* em um sistema móvel, além da adaptação do modelo para utilização de câmeras convencionais em conjunto com as térmicas (sistema multimodal).

5. REFERÊNCIAS

- [1] Bishop, R. 2005. Lane Keeping Assist Systems. *Intelligent vehicle technology and trends*. Artech House.
- [2] Broggi, A., Cerri, P., Ghidoni, S., Grisleri, P. and Jung, H.G. 2009. A New Approach to Urban Pedestrian Detection for Automatic Braking. *IEEE Transactions on Intelligent Transportation Systems*. 10, 4 (Dezembro 2009), 594–605. DOI:<https://doi.org/10.1109/TITS.2009.2032770>.
- [3] Geronimo, D., Lopez, A.M., Sappa, A.D. and Graf, T. 2010. Survey of Pedestrian Detection for Advanced Driver Assistance Systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. 32, 7 (Jul. 2010), 1239–1258. DOI:<https://doi.org/10.1109/TPAMI.2009.122>.
- [4] Global status report on road safety 2015: 2015. http://www.who.int/violence_injury_prevention/road_safety_status/2015/en/. Accessed: 2017-07-23.
- [5] Goubet, E., Katz, J. and Porikli, F. 2006. Pedestrian tracking using thermal infrared imaging. (2006), 62062C.
- [6] Hartley, R. and Zisserman, A. 2004. Camera Models. *Multiple view geometry in computer vision*. Cambridge University Press.
- [7] Kim, D.E. and Kwon, D.S. 2015. Pedestrian detection and tracking in thermal images using shape features. *2015 12th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)* (Outubro 2015), 22–25.
- [8] Krotosky, S.J. and Trivedi, M.M. 2007. On Color-, Infrared-, and Multimodal-Stereo Approaches to Pedestrian Detection. *IEEE Transactions on Intelligent Transportation Systems*. 8, 4 (Dezembro 2007), 619–629. DOI:<https://doi.org/10.1109/TITS.2007.908722>.
- [9] Minkina, W. and Dudzik, S. 2009. Measurements in Infrared Thermography. *Infrared Thermography: Errors and Uncertainties*. John Wiley & Sons, Ltd.
- [10] NHTSA 2015. *Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey*. NHTSA - National Highway Traffic Safety Administration.
- [11] NHTSA 2015. *TRAFFIC SAFETY FACTS 2015*. NHTSA - National Highway Traffic Safety Administration.
- [12] OBSERVATÓRIO | 90% dos acidentes são causados por falhas humanas, alerta OBSERVATÓRIO: 2015. <https://www.onsv.org.br/90-dos-acidentes-sao-causados-por-falhas-humanas-alerta-observatorio/>. Accessed: 2017-05-01.
- [13] 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)* (Sep. 2015), 4718–4722.
- [14] Shapiro, L. and Stockman, G.C. 2001. Perceiving 3D from 2D Images. *Computer Vision*. Prentice Hall.
- [15] Souza, M.A. de, Sanches, I.J. and Gamba, H.R. 2012. A New Method for Generating 3D Thermography Models. (2012).
- [16] Trucco, E. and Verri, A. 1998. Camera Calibration. *Introductory techniques for 3-D computer vision*. Prentice Hall.
- [17] Trucco, E. and Verri, A. 1998. Stereopsis. *Introductory techniques for 3-D computer vision*. Prentice Hall.
- [18] Vidas, S., Moghadam, P. and Bosse, M. 2013. 3D thermal mapping of building interiors using an RGB-D and thermal camera. *2013 IEEE International Conference on Robotics and Automation* (Maio 2013), 2311–2318.
- [19] What is Camera Calibration? 2017. <https://www.mathworks.com/help/vision/ug/camera-calibration.html>. Accessed: 2017-08-23.
- [20] Živčák, J., Hudák, R., Madarász, L. and Rudas, I.J. 2013. Physical Nature of Thermography. *Methodology, models and algorithms in thermographic diagnostics*. Springer Science & Business Media. 31--54.

Algoritmos de Varredura para Listagem de Bicliques Maximais em Modelos Bipartidos de Intervalos e Arco-Circulares

Edmilson Pereira da Cruz
 Universidade Federal do Paraná, Curitiba - Brazil
 epcruz@inf.ufpr.br

Marina Groshaus
 Universidad de Buenos Aires, Buenos Aires - Argentina
 marina-groshaus@yahoo.es

André Luiz Pires Guedes
 Universidade Federal do Paraná, Curitiba - Brazil
 andre@inf.ufpr.br

RESUMO

Um modelo bipartido de intervalos é uma bipartição de um número finito de intervalos na reta real. Um grafo de bi-intervalos é um grafo de intersecção de um modelo bipartido de intervalos onde cada vértice corresponde a um intervalo e dois vértices compartilham uma aresta se, e somente se, ambos intervalos correspondentes possuem intersecção e não pertencem à mesma parte. Uma biclique de um grafo é um subconjunto de vértices que induz um subgrafo bipartido completo. Este artigo apresenta um algoritmo de varredura para encontrar todas as bicliques maximais de um grafo de bi-intervalos a partir de seu modelo bipartido de intervalos e propõe uma modificação para grafos bi-arco-circulares.

ABSTRACT

A bipartite interval model is a bipartition of a finite number of intervals on the real line. An interval bigraph is the intersection graph of a bipartite interval model in which each vertex corresponds to an interval and two vertices share an edge if, and only if, both corresponding intervals have an intersection and do not belong to the same part. A biclique of a graph is a subset of vertices that induces a complete bipartite subgraph. This article presents a sweep line algorithm for finding all maximal bicliques of an interval bigraph from its bipartite interval model and proposes a modification for circular-arc bigraphs.

Palavras-chave

bicliques; grafos de bi-intervalos; grafos bi-arco-circulares

1. INTRODUÇÃO

Um **modelo bipartido arco-circular** é uma tripla (A, B, \mathbb{E}) onde A, B são conjuntos finitos disjuntos, que chamamos de partes, cujos elementos chamamos de intervalos (ou ar-

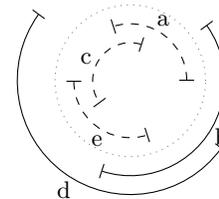


Figura 1: Modelo bipartido arco-circular

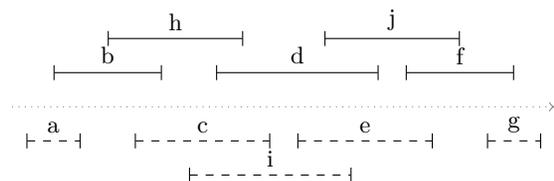


Figura 2: Modelo bipartido de intervalos

cos) e \mathbb{E} é uma ordenação total do conjunto dos extremos de cada intervalo, que definimos por $\{0, 1\} \times (A \cup B)$. Seja um modelo bipartido arco-circular M , denotamos suas componentes A, B e \mathbb{E} por, respectivamente, $A(M), B(M)$ e $\mathbb{E}(M)$.

Para todo intervalo $d \in A(M) \cup B(M)$, chamamos $(0, d)$ e $(1, d)$ de, respectivamente, **evento de início de d** e **evento de fim de d** e denotamos por, respectivamente, s_d e f_d .

Dizemos que um modelo bipartido arco-circular é um **modelo bipartido de intervalos** se o evento de início de todo intervalo ocorre antes de seu respectivo evento de fim.

A Figura 1 é um exemplo de um modelo bipartido arco-circular M onde $A(M) = \{a, c, e\}$, $B(M) = \{b, d\}$ e $\mathbb{E}(M) = (s_a, f_b, s_c, f_a, s_d, s_e, f_c, s_b, f_e, f_d)$. Arcos de A estão representados em tracejado e arcos de B estão em linhas contínuas. Note que f_b ocorre antes de s_b .

A Figura 2 é um exemplo de um modelo bipartido de intervalos M onde $A(M) = \{a, c, e, g, i\}$, $B(M) = \{b, d, f, h, j\}$ e $\mathbb{E}(M) = (s_a, s_b, f_a, s_h, s_c, f_b, s_i, s_d, f_h, f_c, s_e, s_j, f_i, f_d, s_f, f_e, f_j, s_g, f_f, f_g)$. Intervalos de A estão representados em tracejado e intervalos de B estão em linhas contínuas.

Um **grafo bi-arco-circular** é um grafo de intersecção de um modelo bipartido arco-circular onde cada vértice corresponde a um intervalo e dois vértices são conectados por uma aresta se, e somente se, ambos os intervalos correspondentes

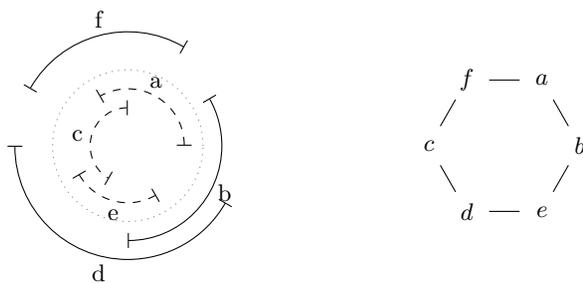


Figura 3: Grafo bi-arco-circular e seu modelo bipartido arco-circular

possuem intersecção não vazia entre si e não pertencem a mesma parte. Um grafo bi-arco-circular é **de bi-intervalos** se admite um modelo bipartido de intervalos como modelo de intersecção.

A Figura 3 é exemplo de um modelo bipartido arco-circular M , definido por $A(M) = \{a, c, e\}$, $B(M) = \{b, d, f\}$ e $\mathbb{E}(M) = (s_a, f_b, s_f, s_c, f_a, f_f, s_d, s_e, f_c, s_b, f_e, f_d)$, e seu respectivo grafo bi-arco-circular. Arcos de A estão representados em tracejado e arcos de B estão em linhas contínuas.

Uma biclique de um grafo é um subconjunto de seus vértices que induz um bipartido completo. Uma biclique S de um grafo G é dita maximal se não existir um vértice $v \in V(G) \setminus S$ tal que $S \cup \{v\}$ induz um bipartido completo.

O grafo representado na Figura 3 possui seis bicliques maximais: $\{f, a, b\}$, $\{a, b, e\}$, $\{b, e, d\}$, $\{e, d, c\}$, $\{d, c, f\}$ e $\{c, f, a\}$.

1.1 Motivação

O estudo de problemas computacionais em grafos busca melhor compreensão da relação entre as classes e a complexidade de sua solução. Dos algoritmos para listagem de bicliques maximais, podemos citar o de Alexe et al. que, embora seja exponencial, admite variantes polinomiais em relação a entrada e saída e duas classes de grafos onde o algoritmo é polinomial em relação a entrada [1]. Dias, Figueiredo e Szwarcfiter [2], Makino e Uno [4] propõem algoritmos polinomiais para certas classes de grafos. Eppstein apresenta um algoritmo de complexidade linear quando parametrizada em função da arboricidade do grafo de entrada [3].

Müller prova que C_6 é uma estrutura proibida dos grafos de bi-intervalos [5] e, a partir desse resultado, Prisner encontra um limitante superior de $(|A||B|)^2$ para o número de bicliques maximais do grafo, para as partes A e B [6].

O algoritmo apresentado neste trabalho usa uma estratégia de varredura sobre a ordem dos eventos de início e fim dos intervalos de algum modelo bipartido de intervalos para encontrar todas as bicliques maximais do grafo correspondente. Acreditamos que, como a ordem dos eventos é preservada no modelo, é possível ter mais controle sobre a identificação de bicliques do que quando a entrada é um grafo.

Neste trabalho, também propomos uma modificação do algoritmo para tratar o caso de grafos bi-arco-circulares, os quais podem ser descritos como uma generalização dos grafos de bi-intervalos.

2. ALGORITMO

O algoritmo para listar todas as bicliques maximais que

propomos usa uma estratégia de varredura sobre os eventos de início e de fim do modelo bipartido de intervalos pela ordem dada por \mathbb{E} . Uma lista de potenciais bicliques é mantida com partes que podem receber intervalos a cada evento de início e cada parte pode ser impedida de receber intervalos a medida que eventos de fim são percorridos.

Porém, o algoritmo também encontra bicliques não-maximais e subconjuntos de vértices que pertencem a uma mesma parte. Ao fim da varredura, potenciais bicliques com uma das partes vazias ou que não são maximais são removidas, restando apenas as bicliques maximais do grafo correspondente ao modelo.

2.1 Lista de Potenciais Bicliques

A lista começa com uma potencial biclique vazia com ambas as partes abertas para receber intervalos e percorre todos os eventos pela ordem definida por \mathbb{E} . Para cada evento de fim encontrado, é feita uma cópia de cada potencial biclique que possui o intervalo, removendo-o, e a potencial biclique original tem a parte contrária a do intervalo fechada.

Definimos as sub-rotinas como:

- **INSERE(L, x)**: Insere elemento x ao fim da lista L ;
- **REMOVE(L, x)**: Remove elemento x da lista L ;
- **BICLIQUE_VAZIA()**: Gera uma estrutura de potencial biclique com ambas as partes vazias e abertas;
- **INTERVALO_EVENTO(e)**: Retorna um elemento de $A \cup B$ correspondente ao evento e ;
- **PARTE_ABERTA(b, p)**: Retorna verdadeiro se a parte p da biclique b está aberta e falso caso o contrário;
- **FECHA_PARTE(b, p)**: Marca a parte p da biclique b como “fechada”; e
- **COPIA(b)**: Retorna uma cópia independente da biclique b .

Algoritmo 1 Geração da lista de potenciais bicliques

Entrada: Modelo bipartido de intervalos (A, B, \mathbb{E})

Saída: Lista de potenciais bicliques

```

1: Função BIVSWEETLINE( $A, B, \mathbb{E}$ )
2:   Lista  $L$ 
3:   INSERE( $L$ , BICLIQUE_VAZIA())
4:   Para todo  $e \in \mathbb{E}$  :
5:     Intervalo  $d \leftarrow$  INTERVALO_EVENTO( $e$ )
6:     Se  $d \in A$  então:
7:        $p \leftarrow 1$ 
8:     Senão:  $\triangleright d \in B$ 
9:        $p \leftarrow -1$ 
10:    Se  $e \in \{0\} \times (A \cup B)$  então:
11:      Para todo  $l \in L$  :
12:        Se PARTE_ABERTA( $l, p$ ) então:
13:          INSERE( $l[p], d$ )
14:        Senão:  $\triangleright e \in \{1\} \times (A \cup B)$ 
15:          Para todo  $l \in L$  :
16:            Se  $d \in l[p]$  então:
17:               $l' \leftarrow$  COPIA( $l$ )
18:              FECHA_PARTE( $l', -p$ )
19:              REMOVE( $l'[p], d$ )
20:              INSERE( $L, l'$ )
21:    Retorne  $L$ 

```

Como descrito pelo algoritmo 1, a lista das potenciais bicliques começa com uma potencial biclique vazia com ambas as partes abertas para receber novos intervalos. Eventos são percorridos pela ordem dada em \mathbb{E} e cada parte é indexada na potencial biclique por 1 e -1 . Se o evento encontrado for de início, seu intervalo correspondente é inserido em todas as potenciais bicliques com a parte correspondente ao intervalo aberta. Se o evento for de fim, todas as potenciais bicliques que contêm o intervalo correspondente ao evento são copiadas sem o intervalo e as potenciais bicliques que foram copiadas têm suas partes contrárias correspondentes ao intervalo fechadas.

2.2 Limpeza da Lista

A limpeza da lista é uma simples comparação duas-a-duas procurando por potenciais bicliques que estejam contidas em outras. Um teste para determinar potenciais bicliques com uma das partes vazias também é feito durante as comparações.

Algoritmo 2 Limpeza da lista de bicliques maximais

Entrada: Lista de potenciais bicliques L e número de intervalos $n = |A \cup B|$

Saída: Lista de bicliques maximais

```

1: Função LIMPEZA( $L, n$ )
2:    $k \leftarrow |L|$ 
3:   Vetor  $v_1[n] \leftarrow (0, \dots, 0)$ 
4:   Vetor  $v_2[k] \leftarrow (0, \dots, 0)$ 
5:   Para todo  $l_1 \in [1..k]$  e  $v_2[l_1] = 0$  :
6:     Se  $l_1[1] = \emptyset$  ou  $l_1[-1] = \emptyset$  então:
7:        $v_2[l_1] \leftarrow 1$ 
8:     Senão:
9:       Para todo  $i \in L[l_1]$  :
10:         $v_1[i] \leftarrow 1$ 
11:      Para todo  $l_2 \in [1..k]$  e  $l_1 \neq l_2$  e  $v_2[l_2] = 0$  :
12:        contida  $\leftarrow 1$ 
13:      Para todo  $i \in L[l_2]$  :
14:        Se  $v_1[i] = 0$  então:
15:          contida  $\leftarrow 0$ 
16:      Se contida = 1 então:
17:         $v_2[l_2] \leftarrow 1$ 
18:      Para todo  $i \in L[l_1]$  :
19:         $v_1[i] \leftarrow 0$ 
20:   Lista  $L'$ 
21:   Para todo  $i \in k$  :
22:     Se  $v_2[i] = 0$  então:
23:       INSERE( $L', L[i]$ )
24:   Retorne  $L'$ 

```

Como descrito pelo algoritmo 2, as comparações da limpeza, que são feitas duas-a-duas, marcam intervalos presentes na primeira potencial biclique e testam se todos os intervalos da segunda potencial biclique foram marcados. Havendo um intervalo não marcado, a segunda potencial biclique difere da primeira por algum intervalo não presente. Se todos os intervalos da segunda potencial biclique estão marcados, então ela está contida na primeira e seu índice na lista é marcado como “para remoção”. Potenciais bicliques marcadas para remoção não são comparadas com as demais.

Por fim, a limpeza retorna a lista de todas as potenciais bicliques que não foram marcadas para remoção.

2.3 Corretude

Como um dos resultados recentes do projeto, temos uma prova de que todas as bicliques maximais do grafo correspondente ao modelo de entrada são encontradas pelo algoritmo. A prova de corretude se baseia na invariante de laço:

TEOREMA 1. *Seja e_i o evento contido na variável e do laço da linha 4 de BIVSWEEPLINE durante sua i -ésima iteração. Após a execução da i -ésima iteração, o algoritmo encontra todas as bicliques maximais contidas no conjunto de todos os intervalos que começam antes do último evento de fim de (e_1, \dots, e_i) .*

PROVA. Seja \mathcal{B}_i o conjunto de todas as bicliques maximais que possuem apenas intervalos que começam antes do último evento de fim de (e_1, \dots, e_i) , para todo $i \in [1..|\mathbb{E}(M)|]$ e qualquer modelo bipartido de intervalos M . Seja também $\mathcal{B}_0 = \emptyset$.

Assumindo que o algoritmo encontra todas as bicliques maximais em \mathcal{B}_{i-1} , provamos que ele encontra todas as bicliques maximais em \mathcal{B}_i . Para isso, consideramos dois casos: e_i é evento de início e e_i é evento de fim.

Para o caso de e_i ser evento de início, provamos que o intervalo correspondente a e_i não é inserido em alguma potencial biclique de modo que alguma biclique maximal seja perdida.

Provamos isso pela seguinte redução ao absurdo: assumimos que o intervalo d correspondente a e_i é inserido na potencial biclique correspondente a biclique maximal $b \in \mathcal{B}_{i-1}$. Logo, temos que a parte de b correspondente a d estava aberta. Como b é biclique, temos que a parte contrária a de d não é vazia. Como a parte correspondente a d estava aberta, nenhum intervalo da parte contrária terminou, o que faz que d intersecte todos os intervalos da parte contrária. Logo, $b \cup \{d\}$ é biclique e, portanto, temos que b não é maximal.

Para o caso de e_i ser evento de fim, temos um novo último evento de fim percorrido em relação à $(i-1)$ -ésima iteração e, portanto, mais bicliques maximais podem ter sido encontradas dependendo dos seguintes subcasos: e_i é o primeiro evento de fim percorrido, e_{i-1} também é evento de fim e e_{i-1} é evento de início.

Se e_i for o primeiro evento de fim percorrido, temos apenas uma potencial biclique e ambas as partes estão abertas até a $(i-1)$ -ésima iteração. Temos que, se houver uma biclique em \mathcal{B}_i , ela possui apenas os intervalos da potencial biclique encontrada.

Se e_{i-1} for evento de fim, temos que $\mathcal{B}_i = \mathcal{B}_{i-1}$ e, como o algoritmo apenas fecha a parte contrária a do intervalo que termina e faz cópias removendo tal intervalo em todas as potenciais bicliques que o contêm, todas as bicliques maximais encontradas até a $(i-1)$ -ésima iteração não são perdidas na i -ésima iteração.

Se e_{i-1} for evento de início, mostramos que todos os intervalos que começam entre o penúltimo e o último evento de fim são inseridos nas potenciais bicliques de tal forma que todas as bicliques maximais em $\mathcal{B}_i \setminus \mathcal{B}_{i-1}$ são encontradas.

Portanto, mostramos que as potenciais bicliques que geram as bicliques maximais a partir dessas inserções já foram produzidas pelo algoritmo até a iteração do penúltimo evento de fim encontrado. Entretanto, a prova encontrada da existência de tais potenciais bicliques é muito longa para ser apresentada neste trabalho. \square

Com o término do laço e como consequência da invariante

de laço, o algoritmo encontra todas bicliques maximais por ter percorrido todos eventos de início que, por lidar com modelos bipartidos de intervalos, ocorrem antes do último evento de fim.

2.4 Adaptação para o Caso Bi-Arco-Circular

Para tratar o caso bi-arco-circular, percorremos os eventos de início e fim uma segunda vez. Como as bicliques maximais podem ser “cortadas” pela entrada com eventos de fim ocorrendo antes de seu respectivo eventos de início, percorremos os eventos uma segunda vez para garantir que todo evento de fim possua um respectivo evento de início o precedendo.

Algoritmo 3 Geração da lista de potenciais bicliques adaptada para o caso bi-arco-circular

Entrada: Modelo bipartido arco-circular (A, B, \mathbb{E})

Saída: Lista de potenciais bicliques

1: **Função** BACSWEEPLINE(A, B, \mathbb{E})

2: **Retorne** BIVSWEPLINE($A, B, \mathbb{E}\mathbb{E}$)

A notação $\mathbb{E}\mathbb{E}$, usada pelo algoritmo 3, representa a concatenação de \mathbb{E} com \mathbb{E} . Como descrito pelo algoritmo, eventos de início e fim são processados da mesma forma que o caso bi-intervalo. Eventos de fim que não possuem seu respectivo evento de início o precedendo são ignorados, pois nenhuma potencial biclique possui o intervalo correspondente no momento. Intervalos que começam uma segunda vez são tratados da mesma forma que na primeira. Se eles puderem ser inseridos em potenciais bicliques que já os contêm, a segunda inserção pode ser omitida.

No caso bi-arco-circular, não apenas bicliques não-maximais são encontradas, mas também duplicatas de bicliques maximais. O algoritmo da limpeza sugerido também trata da remoção de duplicatas, pois nenhuma potencial biclique é marcada para remoção se não houver outra que a contém e que não esteja marcada.

Portanto, ainda não temos uma prova de corretude para o caso bi-arco-circular. Ainda é necessário analisar casos como bicliques cujos intervalos cobrem todo o círculo.

2.5 Complexidade

Para o cálculo de complexidade, definimos $n = |A \cup B| = \frac{|\mathbb{E}|}{2}$, para a entrada (A, B, \mathbb{E}) , e k como o tamanho da lista de potenciais bicliques encontradas pelo algoritmo. Como é difícil prever quantas potenciais bicliques são encontradas, descrever como um valor separado do número de intervalos será conveniente para o cálculo de complexidade.

A geração da lista é feita em $O(kn^2)$, por percorrer a sequência de eventos \mathbb{E} e, para cada evento, percorrer no máximo k potenciais bicliques com uma busca em até n intervalos para cada potencial biclique. Cada cópia é limitada aos n intervalos do modelo para cada potencial biclique.

Cada comparação entre bicliques é feita de forma linear limitada aos n intervalos. As potenciais bicliques são comparadas duas-a-duas, o que nos dá $O(k^2n)$ para a limpeza.

Temos a geração da lista de potenciais bicliques em $O(k^2n + kn^2) = O(kn(k + n))$.

Enquanto ainda não temos um limitante para k em função de n , suspeitamos que k seja exponencial em n . Porém, embora tenhamos tido o intuito de manter o algoritmo simples, acreditamos que é possível prever quais potenciais bicliques

não serão bicliques maximais durante a varredura, evitando cópias desnecessárias e potencialmente reduzindo o limitante de k em função de n .

3. CONCLUSÕES E TRABALHOS FUTUROS

Várias estratégias foram sugeridas para a listagem de bicliques maximais em diferentes classes de grafos [3, 1, 4, 2]. Embora a estratégia de varredura sobre os eventos de início e fim do modelo bipartido de intervalos sugerida neste trabalho para grafos de bi-intervalos pareça ineficiente, ela possui uma simples adaptação que acreditamos resolver o caso bi-arco-circular. Também acreditamos que tal ineficiência não o torna irrelevante para futuras investigações das bicliques maximais em grafos de bi-intervalos.

A prova de corretude encontrada também pode contribuir para melhor entendimento da localização de bicliques maximais em modelos bipartidos de intervalos por tornar mais explícito como as bicliques maximais são descritas em termos de inserções, cópias com remoções e fechamento de partes de potenciais bicliques, levando a encontrar algoritmos de varredura mais eficientes. Porém, ainda não podemos usar os argumentos da prova do caso de bi-intervalos para o caso bi-arco-circular.

Um possível caminho para futuras investigações é determinar se a adaptação para o caso bi-arco-circular é correta, o que pode trazer algumas respostas quanto a natureza das bicliques de tais modelos. Caso seja correto, a distribuição de bicliques maximais pelos modelos bipartidos arco-circulares não é muito diferente da distribuição do caso de bi-intervalos, o que pode ser um indicio de que ambas as classes não são muito distintas em relação ao problema da listagem de bicliques maximais.

Referências

- [1] G. Alexe, S. Alexe, Y. Crama, S. Foldes, P. L. Hammer, and B. Simeone. Consensus algorithms for the generation of all maximal bicliques. *Discrete Applied Mathematics*, 145(1):11 – 21, 2004. Graph Optimization IV.
- [2] V. M. F. Dias, C. M. H. de Figueiredo, and J. L. Szwarcfiter. On the generation of bicliques of a graph. *Discrete Applied Mathematics*, 155(14):1826–1832, 2007.
- [3] D. Eppstein. Arboricity and bipartite subgraph listing algorithms. *Information processing letters*, 51(4):207–211, 1994.
- [4] K. Makino and T. Uno. New algorithms for enumerating all maximal cliques. In *Scandinavian Workshop on Algorithm Theory*, pages 260–272. Springer, 2004.
- [5] H. Müller. Recognizing interval digraphs and interval bigraphs in polynomial time. *Discrete Applied Mathematics*, 78(1):189 – 205, 1997.
- [6] E. Prisner. Bicliques in graphs I: Bounds on their number. *Combinatorica*, 20(1):109–117, 2000.

Utilização de simulador para representação de estratégias de um agente inteligente no desvio de obstáculos

Vinicius Custodio
Universidade Tecnológica Federal do Paraná
Ponta Grossa, Brasil
viniciuscustodio@alunos.utfpr.edu.br

Gleifer Vaz Alves
Universidade Tecnológica Federal do Paraná
Ponta Grossa, Brasil
gleifer@utfpr.edu.br

RESUMO

Os veículos autônomos são uma revolução para a indústria automotiva e estão cada vez mais próximos da realidade. A área de veículos autônomos só tende a crescer nos próximos anos e vem para modificar e ampliar a estrutura da sociedade em termos de como os veículos são utilizados e como são as interações com os usuários. O desenvolvimento requer a junção de diversas áreas, uma delas é a de agentes inteligentes. Existem diversas formas de se implementar um agente inteligente, uma delas é o framework chamado JaCaMo. E para que o agente consiga atuar será utilizado um ambiente gráfico, para esse fim o Simbad é utilizado, um simulador que permite a construção de ambientes. O ambiente definido para esse trabalho possui somente obstáculos estáticos, que estão definidos dentro de faixas, e o agente transita entre as faixas para desviar desses obstáculos. O simulador é responsável por fornecer as informações ao agente, de forma que esse seja capaz de controlar a sua representação dentro do simulador. Para que as informações fossem trocadas, um protocolo de comunicação foi definido, e permite que o agente receba informações sobre o ambiente e envie as ações que devem ser realizadas ao simulador.

ABSTRACT

Autonomous vehicles are a revolution for the automotive industry and is already happening. The area of autonomous vehicles only tends to grow in the coming years and will modify and broaden the structure of society in terms of how vehicles are used and how are the interactions with users. Its development requires the combination of several fields, one of which is intelligent agents. There are several ways to implement a intelligent agent, one of them is a framework called JaCaMo. And for the agent to be able to act a graphic environment will be used, for this purpose Simbad was chosen. The environment defined for this work has only static obstacles, which are defined within ranges, and the agent transits between the tracks to avoid obstacles. The simulator is responsible for providing the information to the agent,

so he can analyze the situation and take some action when needed. In order for information to be exchanged, a communication protocol has been defined, and allows the agent to receive information about the environment and send the actions that must be performed to the simulator.

Palavras-chave

Agentes inteligentes; Veículos autônomos; Simulação; Protocolo de comunicação

1. INTRODUÇÃO

A sociedade sempre busca aprimoramentos para os processos que são realizados diariamente. Uma tecnologia que está sendo alterada é a dos veículos. Nos últimos anos a indústria automobilística vem buscando otimizar o trânsito das cidades é facilitar a vida dos cidadãos através da criação dos veículos autônomos [9].

Os veículos atualmente sofrem de diversos problemas devido ao aumento do número de veículos nas cidades, esse aumento é ocasionado pela grande migração de pessoas dos ambientes rurais para os centros urbanos [5]. Alguns dos problemas que podem ser citados são: (i) o aumento do tráfego de veículos, resultando em congestionamentos [4]; (ii) dos acidentes no trânsito [6]; (iii) e da poluição emitida pelo grande número de veículos.

A criação dos veículos autônomos procura solucionar os problemas citados. A habilidade de se auto-dirigir permite que os veículos sejam utilizados em seu máximo, segundo Cliffe [2] atualmente os veículos são utilizados em somente 5% de sua capacidade, significando um desperdício de 95% do tempo útil do veículo. Com sistemas autônomos guiando os veículos, esperasse que esse seja possível utilizar ao menos 75% da capacidade de cada veículo através do compartilhamento de veículos.

O acréscimo da utilização dos veículos será reflexo dos novos modelos que surgiram com o mercado dos veículos autônomos. Um desses é o compartilhamento de veículos, funcionando de forma semelhante a um táxi. Ao ser requisitado o veículo se dirige ao passageiro e durante o percurso ele pode realizar outras atividades, aumentando assim suas horas produtivas que antes eram gastas dirigindo [9].

Para que um veículo possua capacidades autônomas, sua arquitetura deve compreender em dois níveis, o nível baixo correspondendo aos sensores e atuadores responsável por reconhecimento de objetos, mapeamento, navegação, direção, etc [8]. O nível alto corresponde aos componentes responsáveis muitas vezes pela tomada de decisão, inferindo quais são os sensores e atuadores que devem ser utilizados.

O estudo dessa arquitetura requer uma forma de representação, para os componentes de baixo e alto nível. Este trabalho foca na apresentação de um simulador para representar os componentes de baixo nível, e de um agente como componente de alto nível que realiza a tomada de decisão. O simulador escolhido para esse fim é o Simbad [7], e o framework para a implementação do agente é o JaCaMo [1].

O Simbad [7], permite a criação de ambientes e a adição de recursos como sensores e câmeras que podem ser acessados durante a simulação para a coleta de dados. O Simbad é um simulador graficamente simples, que é implementado em Java o que viabiliza a conexão com o framework JaCaMo. Mesmo simples, ele possui as seguintes funcionalidades como: controle sobre a velocidade, sensores para localização de obstáculos e câmeras para visualização do ambiente do ponto de vista do agente.

O JaCaMo é um framework que implementa agentes inteligentes utilizando a arquitetura BDI. Nessa arquitetura o agente é modelado através de crenças, desejos e intenções. As crenças são os conhecimentos do ambiente, essas informações são mapeadas a partir de percepções que o agente recebe. Os desejos são os objetivos que o agente procurar alcançar e as intenções são os desejos que o agente está buscando alcançar no momento, ou seja, existe um plano que está sendo executado para que o desejo seja satisfeito [1].

Com essas ferramentas, ambiente e agente foram implementados. O agente aqui definido, foi implementado para conseguir escolher entre as manobras definidas, dado um conjunto de percepções que o ambiente repassava ao agente. O agente possui duas manobras, manobra à direita ou à esquerda, para desviar dos obstáculos estáticos definidos no ambiente. Inicialmente a manobra possui uma rota definida, porém quando existem obstáculos dentro da rota inicialmente, essa rota pode ser alterada para que o agente possivelmente consiga desviar do obstáculo.

2. AGENTES E FRAMEWORK JACAMO

Um agente é definido como um sistema de computador que é capaz de realizar ações autônomas [11]. Neste trabalho o agente será responsável por realizar a tomada de decisão, e será implementado utilizando o framework JaCaMo [1], composto pelo Jason (implementa o agente), Cartago (implementa o ambiente) e Moise (implementa a organização dos agentes) e implementa a arquitetura de um agente racional. Esse tipo de agente é capaz de tomar a iniciativa, planejar e agir racionalmente e não baseados em eventos ocorridos no ambiente [10].

Neste trabalho não será utilizado o componente Moise pertencente ao JaCaMo. Para se trabalhar com a organização dos agentes é necessário pelo menos dois agentes. O fim deste é a criação de um agente que consiga se comportar dentro do cenário proposto, que considera um agente isolado no ambiente, sem interações com outros agentes.

3. SIMBAD

O Simbad é um simulador *open source* com fins científicos e educacionais. Focado principalmente para o desenvolvimento e estudo de Inteligência Artificial, Aprendizado de Máquina, Robôs e Agentes Autônomos [7].

O simulador é dividido em três partes: (i) responsável pela construção do ambiente e definição de quais serão as posições dos obstáculos; (ii) responsável por captar as informações do

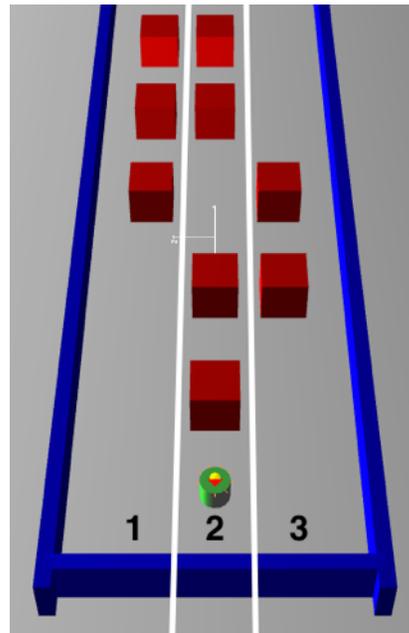


Figure 1: Exemplo de ambiente definido para esse trabalho

ambiente e repassar ao agente; (iii) receber as ações e então atuar no ambiente. Na Figura 1 tem-se um exemplo de um ambiente.

Para que as informações fossem trocadas entre o agente (JaCaMo) e o simulador (Simbad), foi necessário decidir um meio para que as mensagens fossem trocadas entre esses. Um protocolo de comunicação foi definido, criando assim um padrão de como as mensagens devem ser enviadas.

4. CONFIGURAÇÕES: AGENTE E AMBIENTE

Antes de criar as mensagens, foi necessário definir regras, que permitissem a criação de um grupo definido de ações. Para definir esse conjunto de decisões, o ambiente foi delimitado de acordo com as seguintes normas:

- Todo obstáculo é um cubo de 1m;
- O ambiente é organizado em 3 faixas;
- Cada obstáculo se localiza em somente uma das faixas;
- O veículo sempre se locomove pela faixa central;
- E uma manobra só é completada quando o agente retorna a faixa central;

Com essas regras o seguinte grupo de manobras foi definido:

- Desvio à esquerda: Manobra executada quando o agente deseja se locomover para à esquerda dele;
- Desvio à direita: Manobra executada quando o agente deseja se locomover para à direita dele;
- Retornar para direita: Manobra executada quando o percurso à esquerda está bloqueado, nesse caso tentar ir pela direita;

- Retornar para esquerda: Manobra executada quando o percurso à direita está bloqueado, nesse caso tentar ir pela esquerda;
- Manter a direita: Manobra executada quando o agente deseja permanecer à direita;
- Manter a esquerda: Manobra executada quando o agente deseja permanecer à esquerda.

5. INTERLIGAÇÃO: JACAMO E SIMBAD

A interligação foi realizada através do meio TCP, pensando em possíveis testes futuros onde o agente e a simulação não se encontrassem na mesma rede. As seguintes mensagens podem ser enviadas do agente para o simulador, cada mensagem corresponde a uma ação que deve ser executada dentro da simulação, quando essa é enviada.

- 0: Requisição para iniciar o simulador;
- 2: Informa que uma manobra a esquerda é necessária;
- 4: Informa que uma manobra a direita é necessária;
- 6: Informa que o veículo deve parar de se movimentar;
- 8: Informa que o veículo deve desfazer os passos atuais e tentar realizar outra manobra (caso 1 - esteja realizando a manobra pela direita, deve tentar proceder pela esquerda; caso 2 - quando está realizando a manobra pela esquerda, tentar proceder pela direita).
- 10: Informa que veículo deve desfazer os passos da manobra atual e tentar realizar as manobras oposta (direita -> esquerda, esquerda -> direita);
- 12: Informa que veículo deve desfazer o último passo realizado e continuar com a manobra.

E as seguintes mensagens podem ser enviadas do simulador para o agente. Essas mensagens correspondem a informações da simulação que serão utilizadas para que o agente consiga definir qual ação será executada.

- 1#x_pos#y_pos#z_pos: Informa a posição do veículo na simulação;
- 3#x_pos#y_pos#z_pos: Informa a posição do obstáculo;
- 5#obstacle_left#obstacle_center#obstacle_right: Informa em quais posições existem obstáculos (três posições possíveis: esquerda, centro e direita);
- 7#maneuver#direction#distance_to_hit#actual_x#actual_y#actual_z#last_x#last_y#last_z: Informa a manobra atual, direção do veículo, posição atual (coordenadas x, y, z) e a última posição que o agente fez uma curva (x, y, z);
- 9#maneuver#direction#distance_to_hit#turn_point#0actual_x#actual_y#actual_z#last_x#last_y#last_z: Informa a manobra atual, direção do veículo, próxima posição que o veículo pretende fazer uma curva, posição atual (coordenadas x, y, z) e a última posição que o veículo fez uma curva (x, y, z).

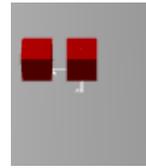


Figure 2: Cenário 1

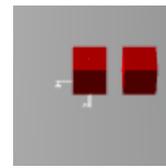


Figure 3: Cenário 2

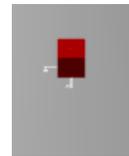


Figure 4: Cenário 3

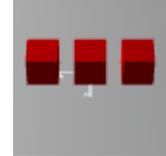


Figure 5: Cenário 4

Através dessas mensagens, é possível que o simulador repasse informações ao agente e vice-versa. Uma vantagem dessa implementação também é que quando for necessário incrementar tanto o simulador quando agente é possível adicionar novas mensagens ao protocolos para que esses continuem a se comunicar.

6. SIMULAÇÃO

Através das mensagens trocadas o simulador consegue interagir com o ambiente de acordo com as decisões do agente. Considerando a mensagem 5, que define se existe um obstáculo na posição, tem-se a seguintes possibilidades:

- true#true#false: Figura 2, para esse cenário a manobra a ser executada pelo agente é a direita, pois é a faixa desocupada.
- false#true#true: Figura 3, cenário oposto ao primeiro de forma que a manobra executada pelo agente é a esquerda, pois é a faixa desocupada.
- false#true#false: Figura 4, como ambas as faixas laterais estão desocupadas o agente sempre escolha a manobra à esquerda.
- true#true#true: Figura 5, nenhuma faixa está disponível e nenhum manobra evitará os obstáculos por isso o agente deixa de se locomover.

Para cada um desses cenários, a simulação possui a manobra implementada que é executada de acordo com a ordem do agente e que corresponde a uma rota que fará com que a representação do agente desvie do obstáculo.

7. CONCLUSÃO

A criação do protocolo de comunicação permite que dois componentes distintos se comuniquem utilizando o padrão de mensagens definido. Sendo assim para trabalho futuros, além do incremento dos próprios componentes, novas formas de implementação do agente e do simulador podem ser exploradas. Os únicos requisitos é que esses permitam o envio de mensagens através da rede e que ambos os componentes obedeçam as mensagens definidas.

A escolha de ambas ferramentas para o trabalho foi devido a suas implementações em Java, uma linguagem que possui mais adeptos em relação há outras. Ambas ferramentas não foram utilizadas em seus potenciais máximo. O agente ainda possui muito espaço para expansão, permitindo novos planos para tratar cenários mais complexos. O simulador, também ainda possui recursos a serem explorados como mais sensores e/ou câmeras que estão disponíveis, para capturar uma mais informações do ambiente.

Alem disso a interligação, que é feita através do protocolo permite a (i) substituição do simulador, para um mais realista. Sendo possível a reutilização da classe de comunicação e do agente, necessitando somente adaptar o simulador para que ele repasse as informações através do protocolo definido; (ii) substituição do agente, criando de um agente em outra linguagem, como por exemplo através do Gwendolen [3], que é uma linguagem para criação de agentes com a arquitetura BDI e que permite a verificação formal dos planos do agente.

Uma vantagem da criação do protocolo é a flexibilidade em relação quais serão os componentes que estarão utilizando o mesmo. Porém, esse flexibilidade vem ao custo de compreensão, quando novas mensagens começarem a surgir, com quantidades cada vez maiores de informação complica a utilização da interligação

Para a melhoria do interligação, a alteração do estilo das mensagens simples para uma configuração REST permitindo assim uma melhor organização das informações, pois este utiliza um padrão definido, como por exemplo JSON. JSON é uma forma de serializar as informações de objetos à pares com valores atribuídos e vetores e facilitando assim o próprio entendimento ao receber uma mensagem.

8. REFERÊNCIAS

- [1] O. Boissier, R. H. Bordini, J. F. Hübner, A. Ricci, and A. Santi. Multi-agent oriented programming with jacamo. *Sci. Comput. Program.*, 78(6):747–761, 2013.
- [2] M. Cliff. Driverless cars — the route to more than smart cities, 2016. Accessed, 22 May 2017.
- [3] L. A. Dennis and B. Farwer. *Gwendolen: A BDI Language for Verifiable Agents*. University of Aberdeen, 2008.
- [4] D. J. Fagnant and K. M. Kockelman. Preparing a nation for autonomous vehicles: Opportunities, barriers and policy recommendations. *Eno Center for Transportation*, 2, 2013.
- [5] U. Nations. *Department of Economic and Social Affairs: Population Division. World Urbanization Prospects: The 2014 Revision*. United Nations, 2015.
- [6] M. Peden, R. Scurfield, D. Sleet, D. Mohan, A. A. Hyder, E. Jarawan, and C. Mathers. *World report on road traffic injury prevention*. World Health Organization, 2004.
- [7] Simbad. Disponível em: <http://simbad.sourceforge.net>, 2017. Acesso em: 22 jan. 2017.
- [8] S. Ulbrich, A. Reschka, J. Rieken, S. Ernst, G. Bagschik, F. Dierkes, M. Nolte, and M. Maurer. Towards a functional system architecture for automated vehicles. *CoRR*, abs/1703.08557, 2017.
- [9] R. Wallace and G. Silber. Self-driving cars: The next revolution. 2012. Acesso em: 23 ago, 2017.
- [10] M. Wooldridge and N. R. Jennings. *Agent Theories, Architectures, and Languages: A survey*. 1994.
- [11] M. J. Wooldridge. *An Introduction to MultiAgent Systems (2. ed.)*. Wiley, 2009.

Avaliação da Usabilidade da Interface Gráfica de Gerenciamento de um Estacionamento Inteligente

Felipe Felix Ducheiko
Universidade Tecnológica
Federal do Paraná
Av. Monteiro Lobato, s/n km 4
84016-210
Ponta Grossa, Brasil
felipeducheiko@alunos.
utfpr.edu.br

Gleifer Vaz Alves
Universidade Tecnológica
Federal do Paraná
Av. Monteiro Lobato, s/n km 4
84016-210
Ponta Grossa, Brasil
gleifer@utfpr.edu.br

Geraldo Ranthum
Universidade Tecnológica
Federal do Paraná
Av. Monteiro Lobato, s/n km 4
84016-210
Ponta Grossa, Brasil
granthum@utfpr.edu.br

RESUMO

O gerenciamento de um Estacionamento Inteligente requer o acesso a várias informações, como quantidade de vagas e motoristas, tempo de permanência no estacionamento, dentre outras. Neste contexto, para um satisfatório gerenciamento do Estacionamento é necessária uma interface gráfica que apresente tais informações de maneira clara. Este trabalho apresenta a avaliação de usabilidade da interface gráfica de um Estacionamento Inteligente, por meio do *framework* DECIDE, a fim de tornar a interação com o usuário mais efetiva.

Palavras-chave

Interação Humano-Computador; Avaliação de Usabilidade; Cidade Inteligente; Estacionamento Inteligente; e Sistema Multiagentes

ABSTRACT

The management of a Smart Parking requires access to several information, such as number of parking spots and drivers, parking time, among others. In this context, for a satisfactory parking management, a graphical interface is required to presents the information more clearly. Here we aim to show a usability evaluation of this graphical interface, through the DECIDE framework, in order to make the interaction with the user more effective.

Keywords

Human-Computer Interaction; Usability Evaluation; Smart City; Smart Parking; Multi-Agent System

1. INTRODUÇÃO

O conceito de Cidade Inteligente diz respeito ao uso de tecnologias na tentativa de solucionar problemas das cidades, aumentando a qualidade de vida da população. A essência do conceito é integrar as tecnologias que até agora têm sido desenvolvidas separadamente umas das outras, mas que têm ligações claras em seu funcionamento podendo ser desenvolvidas de forma integrada [1].

Um dos principais problemas enfrentados por Cidades Inteligentes é o de mobilidade urbana. Estima-se que em Nova York cerca de 40% dos congestionamentos são ocasionados por motoristas buscando vagas de estacionamento [6], o que gera um agravamento dos congestionamentos e por consequente aumenta a emissão de poluentes.

Quando se percebe que a grande demanda de vagas de estacionamentos não está sendo satisfeita a solução adotada é um aumento quantitativo do número de vagas. Porém, nem sempre essa é a solução mais sensata, pois a utilização das mesmas vagas de modo mais inteligente pode amenizar ou até mesmo solucionar o problema.

Smart Parkings (Estacionamentos Inteligentes) são compostos por dispositivos de *hardware*, capazes de detectar o nível de ocupação dos estacionamentos, e *softwares* integrados para gerir a atribuição desses espaços [7]. Tais sistemas são concebidos para auxiliar os motoristas na localização de vagas disponíveis, colaborando desta forma com a solução de problemas relacionados a mobilidade urbana.

Existem várias abordagens de pesquisa em *Smart Parking* sendo realizadas e estas podem ser categorizadas conforme as tecnologias empregadas em sua implementação, algumas das categorias são: *Automated Parking* (foco em mecanismos computadorizados), *E-Parking* (utiliza Internet ou SMS) e *Agent Based Guiding System* (ABGS), esta última é caracterizada pela utilização de Sistemas Multiagentes na implementação de *Smart Parkings* [10].

Sistemas Multiagentes (SMAs) são definidos como sistemas compostos de vários elementos computacionais que realizam interações, sendo tais elementos conhecidos como agentes. Esses sistemas possuem duas características importantes: primeiramente são capazes, ao menos em certa medida, de ações autônomas e em segundo lugar têm a capacidade de interagir uns com os outros de maneira análoga às interações sociais humanas [12].

O projeto MAPS (*MultiAgent Parking System*) aplica métodos e técnicas de SMA na criação de uma soluções para

alocação de vagas e gerenciamento de um *Smart Parking*, o qual é desenvolvido no GPAS (Grupo de Pesquisa em Agentes de *Software* - UTFPR - PG).

Para auxiliar no gerenciamento do *Smart parking* o MAPS possui uma interface gráfica, por meio da qual é possível visualizar dados como o nível de ocupação do estacionamento e a situação da fila de espera. O desenvolvimento desta interface foi pautado nos requisitos de usabilidade e é apresentada em trabalhos anteriores (ver [3] e [4]), sendo necessário ainda realizar a avaliação da usabilidade da mesma.

Usuários têm um grande potencial para fazer interpretações inesperadas de elementos da interface e para executar o seu trabalho de forma diferente do aguardado [5]. Então, fica clara a necessidade de aplicar esta avaliação na interface, de modo a corrigir possíveis problemas de interação.

O presente trabalho tem como objetivo apresentar a avaliação da usabilidade desta interface gráfica, identificando os possíveis problemas de interação que possam ocorrer, para posteriormente corrigi-los. Deste modo, almeja-se que a interface torne-se mais clara e intuitiva, melhorando a experiência de utilização do sistema pelo usuário. Para realização do teste de usabilidade foi utilizado o *framework* DECIDE, pois este padroniza a execução dos testes, permitindo que os testes sejam realizados de forma mais sistemática e confiável.

2. TRABALHOS RELACIONADOS

Esta seção apresenta dois trabalhos relacionados ao cenário deste artigo. O primeiro apresentado possui o mesmo contexto da pesquisa deste artigo, mobilidade urbana. Já o segundo utiliza o mesmo *framework* que é utilizado no presente trabalho para conduzir a pesquisa, o *framework* DECIDE.

Em [2] é apresentado o processo de avaliação da experiência de uso de dois aplicativos, relacionados com mobilidade urbana: Bicicletar e Meu Ônibus. Para esta avaliação foram aplicados os seguintes métodos: (i) questionário online; (ii) inspeção de usabilidade; (iii) avaliação com usuários em contexto real (questionário pré-teste, entrevista semiestruturada, observação de uso, questionário *Self-Assessment Manikin* e questionário pós-teste).

Os métodos aplicados neste trabalho permitiram uma visão ampla da experiência de uso dos aplicativos, possibilitando identificar que os aplicativos proporcionam aos usuários uma experiência de uso considerada boa. Porém, segundo os autores, existem problemas relacionados a usabilidade que impactam negativamente em pontos específicos da experiência e poderiam ser melhorados.

No trabalho de Silva e Jesus [11] é apresentada a avaliação da interface do Sistema de gestão acadêmica do Instituto Federal de Minas Gerais (IFMG), conhecido como MeuIFMG. Este trabalho adotou o *framework* DECIDE [9] para realizar a pesquisa, o qual fornece uma lista de verificação para conduzir o processo de avaliação. A coleta dos dados para realizar a avaliação da interface gráfica se deu por meio de um questionário online e entrevistas presenciais.

Ao final do trabalho de Silva e Jesus foi possível apresentar uma compilação de dados, que mostra quantitativa e qualitativamente, o nível de alguns problemas de interação que podem ser encontrados em Sistemas de gestão acadêmica e fornece recomendações para futuras melhorias nestes sistemas.

3. INTERFACE GRÁFICA

A interface gráfica, que é objeto de estudo deste trabalho, tem o objetivo de auxiliar no controle e gestão do MAPS, apresentando informações sobre o estacionamento de modo mais intuitivo.

Esta interface foi desenvolvida em inglês, visando a internacionalização do Projeto MAPS e é composta por quatro abas: *home*, *graphics*, *drivers info* e *about*. Na tela inicial da interface gráfica, que pode ser visualizada na Figura 1, o usuário possui informações sobre o nível de ocupação do *smart parking* e sobre a situação da fila de espera.

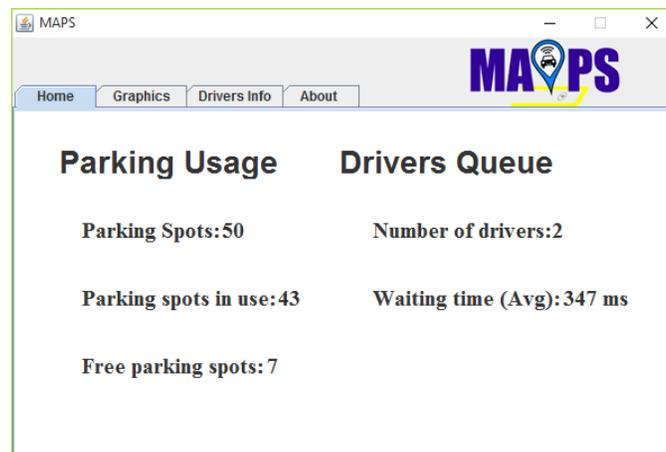


Figura 1: *Aba Home*

Na aba *graphics*, que pode ser visualizada na Figura 2, o usuário possui informações sobre o nível de ocupação do *smart parking*, através de um gráfico de barra e sobre a situação da fila de espera, através de um gráfico de linha que relaciona o horário e quantidade de motoristas na fila de espera.

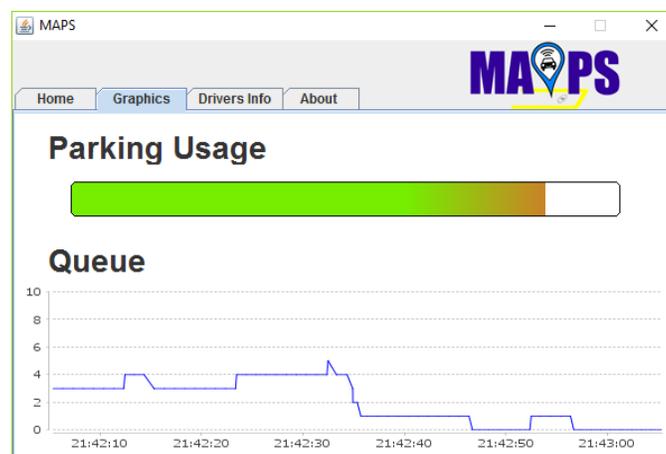


Figura 2: *Aba Graphics*

Na Figura 3 tem-se a aba *drivers info* que traz os valores máximo, mínimo e médio do tempo de permanência (*Parking time*) e também do grau de confiança (*Trust Degree*) dos motoristas que utilizam o *smart parking*.

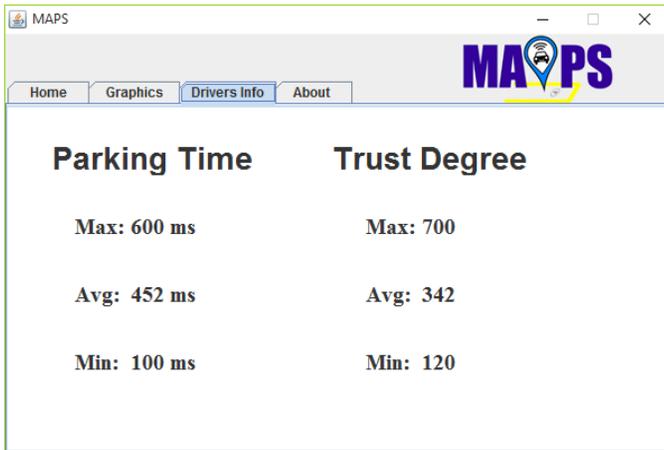


Figura 3: Aba Drivers Info

A interface também possui a aba *about*, a qual traz informações gerais sobre o Estacionamento Inteligente.

4. METODOLOGIA

Para o desenvolvimento desta pesquisa foi utilizado o *framework* DECIDE [9], este *framework* oferece uma lista de verificação para a condução de avaliações de usabilidade, permitindo que o processo de avaliação seja mais claro e efetivo.

O *framework* DECIDE é composto de seis etapas: (i) determinar as metas da avaliação; (ii) explorar as questões cuja avaliação pretende responder; (iii) escolher os métodos e as técnicas que responderão às questões da avaliação; (iv) identificar questões práticas a serem abordadas pela avaliação; (v) decidir como lidar com as questões éticas envolvidas; e (vi) avaliar, interpretar e apresentar os dados.

Inicialmente, já fazendo uso do *framework* DECIDE, foram definidas em sua primeira etapa duas metas: (i) avaliar a experiência de uso do sistema; e (ii) verificar se os recursos disponíveis na interface gráfica satisfazem às necessidades dos usuários.

Na segunda etapa foram definidos dois conjuntos de questionamentos a serem feitos aos usuários, cada conjunto com o objetivo de atingir uma das metas da pesquisa, que foram definidas na primeira etapa.

Na terceira etapa foram escolhidos os dois métodos utilizados na aplicação dos testes de usabilidade. O primeiro método aplicado foi uma inspeção de usabilidade da interface gráfica, onde os problemas mais aparentes de interação foram corrigidos.

O segundo método utilizado foi uma pesquisa com voluntários, por meio de um formulário *online*¹, com o objetivo de verificar a usabilidade da interface gráfica. O formulário foi aplicado em um ambiente controlado, onde 33 voluntários responderam 21 questões, o número de voluntários foi definido com base no trabalho de [8] que afirma que acima de 12 voluntários mais de 85% dos problemas de usabilidade são identificados. Ressalta-se que antes da aplicação da pesquisa houve uma apresentação do contexto do Estacionamento Inteligente aos participantes.

Este formulário é dividido em quatro partes: (i) pré-teste, questionamentos sobre o perfil do usuário; (ii) questiona-

¹Link formulário: <https://goo.gl/forms/cz7MD9nri1h68QFk2>

mentos sobre a facilidade de encontrar e interpretar elementos da interface; (iii) questionamentos sobre a experiência geral de uso da interface, utilizando Escala *Likert* [9]; e (iv) questionamento pós-teste, questão aberta onde o voluntário pode relatar experiências e dificuldades encontradas ao utilizar a interface.

Na quarta etapa do *framework* DECIDE foram identificadas questões de ordem prática para aplicação dos testes. Dentre as questões tratadas citam-se a preparação do local e equipamentos para a aplicação da avaliação e o recrutamento dos voluntários.

Na quinta etapa uma questão ética foi abordada, referente a privacidade dos voluntários. Foi decidido pelo anonimato dos participantes, em virtude da identificação do voluntário não influenciar na avaliação da interface.

A sexta etapa consiste na avaliação, interpretação e apresentação dos dados coletados. Esta etapa busca atingir as metas definidas na primeira etapa e será descrita de maneira detalhada na próxima seção.

5. ANÁLISE DOS DADOS

Esta seção apresenta a análise dos dados coletados por meio da pesquisa realizada com os voluntários. Inicialmente foi realizado uma análise do perfil dos voluntários, seguido de uma análise da facilidade de encontrar e interpretar elementos da interface e de sua experiência geral de uso. Por último as críticas e sugestões dos voluntários são analisadas.

Analisando os dados coletados foi possível determinar o perfil dos voluntários. Em relação a este item, 63,6% deles tinham entre 19 e 26 anos e 45,5% tinham menos de quatro anos com experiência em informática. Estes dados foram importantes para identificar a diversidade dentro da amostra pesquisada, visto que esta pode influenciar diretamente o resultado das análises.

Lembrando que a interface foi desenvolvida em inglês, o nível de conhecimento de inglês dos voluntários da pesquisa ficou distribuído do seguinte modo: 30,3% dos voluntários possuíam nível de inglês básico, 48,5% nível de inglês intermediário e 21,2% nível de inglês avançado. Além disso, foi disponibilizado para os voluntários, durante a execução dos testes, traduções de todos os termos em inglês da interface.

Os dados coletados sobre a facilidade de encontrar e interpretar elementos na interface dão uma visão geral da usabilidade da mesma. Para este quesito, 97% dos voluntários conseguiram localizar e interpretar corretamente todos os elementos da interface, sendo que apenas um usuário teve dificuldade em encontrar e interpretar o gráfico *queue* da aba *graphics* (Figura 2), o qual mostra um histórico da fila de espera do estacionamento. Este problema será abordado novamente no decorrer da análise.

Ainda neste conjunto de dados coletados houve um problema de interação com o já referido gráfico *queue*, onde um voluntário relatou que ao clicar e arrastar com o *mouse* sobre o gráfico sua escala era reduzida, porém não existia a funcionalidade de voltar ao estado inicial da escala. Este problema deverá ser corrigido, visto que entra em conflito com o requisito de usabilidade “Controle do Usuário” [5].

O grupo de dados sobre o item experiência geral de uso da interface foi avaliado utilizando escala *Likert* de cinco pontos [9]. Nesta etapa existiam questionamentos sobre a interface e os usuários puderam expressar sua opinião escolhendo na escala *Likert* entre cinco opções: “concordo totalmente”, “concordo”, “neutro”, “discordo” e “discordo totalmente”.

Neste grupo de dados a primeira pergunta era “O desempenho do sistema é bom?”. As respostas dos participantes ficaram assim distribuídas: 42,4% dos usuários escolheram a opção “concordo totalmente” e 30,3% a opção “concordo”.

Na pergunta “O tamanho e fonte dos caracteres são apropriados?” 54,5% dos voluntários escolheram a opção “concordo totalmente” e na pergunta “A sequência de abas faz sentido?” 45,5% dos voluntários escolheram a opção “concordo totalmente” e 24,2% escolheram a opção “concordo”.

Os recursos gráficos da interface devem ser melhorados, visto que 42,4% dos voluntários mantiveram uma posição neutra em relação a pergunta “Os recursos gráficos disponíveis na interface são apropriados?” e apenas 30,3% escolheram a opção “concordo”. A distribuição das respostas pode ser visualizada na Figura 4 (a).

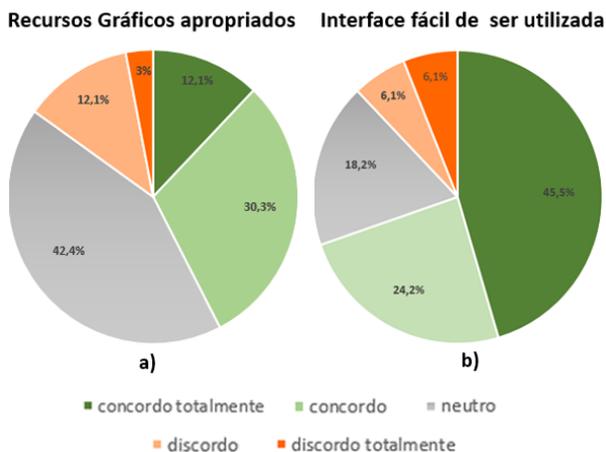


Figura 4: Gráficos dos Dados Coletados

Na questão “O Layout utilizado na interface é apropriado?” 57,5% dos voluntários escolheram a opção “concordo totalmente” ou “concordo”. Já 81,9% dos voluntários escolheram a opção “concordo totalmente” ou “concordo” na pergunta “As cores utilizadas na interface são apropriadas?”.

Na pergunta “A interface é fácil de ser utilizada?” 45,5% dos voluntários escolheram a opção “concordo totalmente” e 24,2% a opção “concordo”, como pode ser visualizado na Figura 4 (b).

No último conjunto de dados coletados os usuários puderam relatar experiências e dificuldades encontradas ao utilizar a interface. Após análise deste conjunto de dados algumas sugestões foram recorrentes. Dentre elas colocar a porcentagem de uso do estacionamento escrita de forma numérica dentro do gráfico *Parking Usage* na aba *graphics*.

Outra sugestão foi mudar o nome do gráfico *queue*, ainda na aba *graphics*, para *queue history* e inserir uma legenda indicando o significado de cada eixo do gráfico. Por meio destas duas últimas sugestões o gráfico *queue* ficará mais intuitivo, contornando o problema enfrentado com o voluntário que teve dificuldade de encontrar e interpretar este gráfico, já citado anteriormente nesta seção.

Sugeriu-se também acrescentar uma linha para separar as colunas de informações apresentadas nas abas *home* e *drivers info*. Desta maneira evitando a leitura *left-right*, pois os dados são apresentados de cima para baixo e a leitura deve ser *top-down*. Outra sugestão recorrente foi fazer com que a interface seja responsiva, visto que a interface atual tem um tamanho fixo.

Adicionalmente, os voluntários, sugeriram adicionar maiores informações na interface, por exemplo, a criação de uma nova aba chamada *report*, a qual apresentaria uma espécie de “log” com a quantidade de *drivers* que usaram o estacionamento nas últimas horas.

6. CONCLUSÃO

Por meio do teste de usabilidade constata-se que de forma geral a interface possui boa usabilidade. Porém, existem alguns problemas de usabilidade que impactam negativamente a interação com o usuário e devem ser solucionados.

O emprego do *framework* DECIDE facilitou a realização dos testes, visto que auxiliou a identificar questões que precisavam ser consideradas para a execução do teste de usabilidade. Por meio deste *framework* foi possível executar as etapas da pesquisa de modo mais claro.

Em termos de continuidade do trabalho, é possível elencar os seguintes trabalhos futuros: (i) implementar na interface gráfica as correções dos problemas de usabilidade descritos neste trabalho; (ii) implementar um aplicativo *mobile* para o projeto MAPS, onde os motoristas poderão requisitar vagas de estacionamento; e (iii) realizar testes de usabilidade para este aplicativo *mobile*.

7. REFERÊNCIAS

- [1] M. Batty, K. Axhausen, G. Fosca, A. Pozdnoukhov, A. Bazzani, M. Wachowicz, G. Ouzounis, and Y. Portugali. *Smart Cities of the Future*. 2012.
- [2] N. Carneiro, M. Pinheiro, V. Mesquita, B. Coelho, J. Forte, and T. Darin. *Vou de Ônibus ou de Bike? Uma Avaliação da Experiência de Uso de Aplicativos de Apoio a Mobilidade Urbana*. 15º Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais. São Paulo - SP. 2016.
- [3] F. F. Ducheiko and G. V. Alves. *Desenvolvimento de Interface Gráfica para Gerenciamento de um Smart Parking*. 1º Workshop de Pesquisa em Computação dos Campos Gerais. Ponta Grossa - PR. 2016.
- [4] F. F. Ducheiko, L. F. S. Castro, and G. V. Alves. *Implementação de Recursos em um Smart Parking baseado em Sistemas Multiagente*. 10º Workshop-Escola de Sistemas de Agentes, seus Ambientes e aplicações. Maceió - AL. mai. 2016.
- [5] N. Jakob. *Usability Engineering*. 1993.
- [6] A. Koster, F. Koch, and A. L. Bazzan. *Incentivising Crowdsourced Parking Solutions*. 2014.
- [7] D. D. Nocera, C. D. Napoli, and S. Rossi. *A Social-Aware Smart Parking Application*. 2014.
- [8] R. Prates and S. Barbosa. *Avaliação de Interfaces de Usuário - Conceitos e Métodos*. Editora PUC-Rio, 2007.
- [9] J. Preece, Y. Rogers, and H. Sharp. *Design de Interação: Além da Interação Humano-Computador*. Bookman, Porto Alegre, 2005.
- [10] G. Revathi and V. R. S. Dhulipala. *Smart Parking Systems and Sensors: A Survey*. 2012.
- [11] J. A. Silva and n. M. Jesus. *Avaliação de Interação de um Sistema de Gestão Acadêmica: Um Estudo de Caso*. 8º Computer On The Beach. Florianópolis, SC. Mai. 2017.
- [12] M. Wooldridge. *An Introduction to MultiAgent Systems*. J. Wiley, New York, 2nd edition, 2009.

Utilização de lógica temporal-deontica para representação do comportamento de um agente racional no controle de um veículo autônomo

Vithor Tozetto Ferreira
 Universidade Tecnológica Federal do Paraná
 Av Monteiro Lobato, s/n - Km 04 CEP 84016-210
 Ponta Grossa, Brasil
 vithorferreira@alunos.utfpr.edu.br

Gleifer Vaz Alves
 Universidade Tecnológica Federal do Paraná
 Av Monteiro Lobato, s/n - Km 04 CEP 84016-210
 Ponta Grossa, Brasil
 gleifer@utfpr.edu.br

RESUMO

Os veículos autônomos (VA's) são uma nova tecnologia que pode se tornar realidade num futuro próximo. Esta tecnologia ainda gera preocupação para muitos, já que é difícil demonstrar o quão seguros e confiáveis esses veículos podem ser. Por esse motivo, é necessário desenvolver formas de assegurar o comportamento correto de tais veículos antes de permitir seu uso nas cidades. Considerando que um VA pode ser modelado por meio de um agente racional, é possível utilizar os operadores da lógica temporal-deontica para especificar propriedades de comportamento do agente responsável pelo controle de alto nível do VA. Neste trabalho é discutida a aplicação da lógica temporal-deontica para representar o comportamento de um agente racional em alguns cenários próprios de um VA.

Palavras-chave

Lógica Temporal-Deontica; Agentes; Veículos Autônomos

ABSTRACT

The autonomous vehicles (AVs) are a new technology which may become a reality in the near future. This technology still generates uneasiness for many people, since it is difficult to demonstrate how secure and reliable those vehicles can be. Therefore, it is necessary to develop means to assure the correct behavior of such vehicles before allowing their use in the cities. Considering that an AV can be modeled on a rational agent, it is possible to use the temporal-deontic logic to specify behavior properties of the agent responsible by the high level control of the AV. In this paper we discuss the application of temporal-deontic logic for representing the behavior of a rational agent in some typical AV scenarios.

Keywords

Temporal-deontic Logic; Agents; Autonomous Vehicles

1. INTRODUÇÃO

O grande avanço tecnológico dos últimos anos ocasionou o desenvolvimento de uma nova onda de tecnologia da informação, onde computadores estão presentes em quase todos os locais e se tornaram essenciais para a vida cotidiana. Diversas tarefas humanas estão sendo adaptadas para que um computador possa executá-las, facilitando e melhorando o funcionamento de diversos cenários. Essa substituição ainda gera certa desconfiança no público geral, em relação a quais tarefas realmente são seguras de se atribuir a um computador. Entre elas, a tarefa de controlar um veículo é um motivo de preocupação para muitos, devido ao grande foco que diversas empresas dão a essa tecnologia.

Uma das grandes preocupações em relação aos veículos autônomos (VA's) é a interação que eles têm com vidas humanas. Esse problema é uma questão ética, sobre como uma máquina irá reagir em certas situações, e sobre como interpretar as ações do VA. A discussão sobre esse tema geralmente é relacionada ao famoso dilema do bonde [6], onde existem dois caminhos, uma alavanca para desviar os trilhos e um bonde desenfreado. Em um desses caminhos, cinco pessoas estão posicionadas, e serão atingidas pelo bonde. No outro caminho, existe apenas uma pessoa. Se a alavanca for puxada, o caminho do bonde será desviado e ele atingirá apenas uma pessoa. Se nada for feito, as cinco pessoas no caminho principal serão atingidas. Qual das atitudes seria considerada a mais ética? Não agir, causando a morte de cinco pessoas, ou agir e ser responsável pela morte de uma pessoa?

Existe uma versão do dilema do bonde adaptada ao contexto de VA's, a *Moral Machine*, desenvolvida pelo MIT [1, 7]. Nesse teste existem diferentes cenários críticos, onde algum tipo de colisão envolvendo o VA é inevitável, e neste experimento cabe ao usuário tomar a decisão que julgue adequada. As opiniões sobre o que é ético nesse teste são divergentes, pois cada pessoa irá valorizar um aspecto diferente da situação antes de sua escolha. Em uma situação real de risco na condução de um veículo, o motorista irá agir impulsivamente, pois em muitas situações não há tempo para uma pessoa tomar uma decisão antes de agir.

O grande problema dessa discussão é que um VA não age impulsivamente. O VA segue sua programação, seu conjunto

de regras. Seria correto considerar que um acidente envolvendo VA's foi premeditado? Pois, o VA, baseado em suas regras, fez o que estava programado para fazer. A culpa é do programador? Essas questões devem ser tratadas de alguma forma antes dos VA's serem utilizados abertamente nas cidades.

Uma das formas de ver esse problema é que, a programação do VA deve estabelecer um conjunto de regras para o mesmo, e que o VA decidirá de forma autônoma sobre como proceder. Para estabelecer essas regras é necessário a utilização da lógica. A lógica formal permite a representação de diversos cenários, considerando possíveis comportamentos. Ao usar representações formais, o conjunto de regras pode ser testado e documentado, tornando explícita a forma que o VA irá reagir ao seu ambiente. Quando o comportamento do VA é conhecido, é possível estabelecer regras de segurança para seu uso, regulamentando a utilização dessa tecnologia e permitindo que ela seja acessível ao público.

Para se obedecer esse conjunto de regras, é necessário um controlador que interprete essas regras e as utilize na condução do VA. Esse é o papel do agente racional. Um agente é uma entidade que pode agir de forma autônoma, e que tem a capacidade de interpretar e agir no ambiente que está situado [10]. Um agente racional é capaz de raciocinar sobre si mesmo e o ambiente, e pode tomar decisões autônomas para atingir seus objetivos. O agente racional pode receber esse conjunto de regras, e as utilizará para tomar decisões sobre a condução do VA. Dessa forma, o VA terá um controlador capaz de interpretar as percepções do ambiente, decidir o melhor rumo das ações e agir eficientemente, enquanto respeita as regras estabelecidas na sua criação.

Para tentar assegurar que o mecanismo de controle de um VA irá agir conforme sua respectiva programação de alto nível é possível utilizar a verificação formal. A verificação formal serve para realizar testes no conjunto de regras formais que será aplicado no agente racional do VA. Esses testes percorrem os caminhos possíveis que as regras têm, garantindo que não existem falhas nessas regras que podem resultar em um evento indesejado.

Neste artigo são apresentados conceitos de agentes inteligentes e lógicas modais, em especial lógica: temporal e deôntica. O objetivo da apresentação desses conceitos é mostrar como essas lógicas podem ser utilizadas na representação adequada do comportamento de um agente racional na condução de um VA. Por sua vez, o trabalho aqui apresentado é parte de um projeto de pesquisa que tem como principal objetivo o desenvolvimento de soluções para controle de VA's por meio de agentes inteligentes e verificação formal. Consequentemente, almeja-se que em trabalhos futuros seja possível aplicar o formalismo de uma lógica **temporal-deôntica** na verificação formal do comportamento de agentes inteligentes.

Na continuação deste artigo, tem-se na Seção 2, uma explicação das características de um agente e do funcionamento do raciocínio de um agente racional. Após, a Seção 3 possui uma explicação sobre as divisões da lógica abordadas neste artigo, e seus operadores. Em seguida, na Seção 4 são demonstrados alguns exemplos da utilização dos operadores em argumentos lógicos. Finalmente, na Seção 5, são apresentadas as considerações finais em conjunto com as próximas etapas de realização do trabalho.

2. AGENTES

O termo agente possui diferentes significados dependendo de sua aplicação e do entendimento do autor. Conforme Wooldridge e Jennings [11], a forma mais geral de classificar um agente é como uma entidade situada em certo ambiente, seja esse ambiente virtual ou não, e que tal entidade pode executar ações autônomas em seu ambiente para atingir seus objetivos. Agentes também são relacionados às propriedades de autonomia, habilidade social, reatividade e proatividade. Existem diversas variações de agente, e explicar em detalhes cada uma delas não é o foco deste artigo.

Os agentes realmente interessantes para este artigo são os agentes racionais. Um agente racional é aquele que percebe seu ambiente através de sensores, que possui um modelo do ambiente, que pode raciocinar sobre si mesmo em relação ao ambiente e a outros agentes, e que, baseado em suas próprias conclusões, executa ações para modificar seu ambiente [8]. Um exemplo de agente racional seria uma pessoa qualquer. Por exemplo, se uma pessoa está saindo de casa, e percebe que o céu está cheio de nuvens escuras, a ação racional é levar um guarda-chuva [12]. Quando chover, a pessoa poderá se proteger, e sua ação de levar um guarda-chuva terá sido correta, evitando que a pessoa ficasse encharcada.

Entretanto, caso não chova, uma ação desnecessária foi executada, seria o ato de carregar um guarda-chuva incorreto? Se não chover, não há motivos para se carregar um guarda-chuva. Porém, observando esse cenário racionalmente, percebe-se que é melhor carregar um guarda-chuva como precaução do que fazer o contrário e acabar se molhando. Essa atribuição de racionalidade a um agente foca em balancear o comportamento reativo e proativo, balancear a percepção, deliberação e ação para evitar o desperdício de recursos, e permitir que agentes interajam socialmente compartilhando recursos enquanto equilibram seus interesses e os da comunidade [8].

Para se implementar um agente racional, é preciso interpretar as percepções dos sensores do agente e utilizá-los para o processamento e planejamento de ações. Neste artigo, destaca-se o modelo BDI. A sigla se refere as palavras *Belief, Desire and Intention*, ou Crenças, Desejos e Intenções. Esse modelo ajuda a dividir os aspectos do raciocínio de um agente, separando os conceitos de deliberação, percepção e ação, permitindo ao agente decidir o que deve ser feito, como deve ser feito, e então fazer. Nesse modelo, as Crenças são as informações que o agente possui sobre seu ambiente, obtidas através de seus sensores e outros meios de entrada de informação. Os Desejos e as Intenções influenciam na tomada de decisão e planejamento de ações do agente, já que esses atributos se referem aos objetivos do agente, sobre o que deve ser feito e como deve ser feito para atingir seus objetivos.

3. ASPECTOS LÓGICOS

Para definir as ações de um agente de forma abstrata, geralmente são utilizadas expressões que caracterizam atitudes humanas, como acreditar, achar, perceber, desejar. Descrever as ações de um agente com essas expressões facilita a compreensão de seu funcionamento e permite que o raciocínio do mesmo seja explicado de forma simples. Entretanto, existem maneiras formais de se explicar o comportamento de um agente, utilizando argumentos lógicos.

Esses argumentos lógicos vêm da lógica formal, utilizada para gerar proposições (livres de ambiguidades) que facilitam o entendimento de certas expressões, permitindo repre-

sentar formalmente sistemas de ações de diversas áreas, incluindo a computação. Em um agente racional, é importante a utilização da lógica modal, que contém diversos operadores para a representação da modalidade das ações, ou seja, indicar a forma que uma ação acontece ou a característica de tal ação. Os operadores característicos da lógica modal são: \Box e \Diamond , respectivamente “necessariamente” e “possivelmente”. Dentre as modalidades desta lógica, é necessário destacar aquelas de grande importância para a compreensão deste artigo. As subseções a seguir contêm explicações sobre a Lógica Temporal Linear e a Lógica Deontica, e também uma breve explicação sobre como utilizar esses operadores em uma ferramenta para verificação de agentes.

3.1 Lógica Temporal Linear

Existem diferentes tipos de lógica temporal. Aqui se descreve (brevemente) a Lógica Temporal Linear (LTL). Essa lógica possui operadores utilizados para a definição de ações em função de tempo, permitindo, por exemplo, definir argumentos que são verdadeiros em um período de tempo e falsos em outro [5].

A LTL é útil para um agente racional por permitir a ordenação de ações em um plano e por auxiliar na previsão de possíveis ações necessárias no futuro. Os operadores comuns dessa lógica são, considerando que φ e ψ representam eventos diferentes: $\bigcirc\varphi$, “próximo”, representa que φ é verdadeiro no próximo momento de tempo, em um sentido de ordenação de ações; $\bullet\psi$, “prévio”, representa que ψ é verdadeiro no momento anterior de tempo, em um sentido de ordenação de ações; $\Box\varphi$, representa que φ é sempre verdadeiro no futuro; $\Diamond\psi$, representa que ψ é verdadeiro em algum momento no futuro; $\varphi\mathcal{U}\psi$, “until”, representa que φ vai permanecer verdadeiro do momento atual até o momento onde ψ passa a ser verdadeiro; $\varphi\mathcal{W}\psi$, “while”, representa que φ irá permanecer verdadeiro enquanto ψ for verdadeiro.

3.2 Lógica Deontica

A Lógica Deontica é uma parte da lógica modal que lida com obrigações, permissões e proibições. Essa lógica trabalha com o dever de cometer ou não certo ato, e como obrigações e permissões influenciam no pensamento antes de agir. Os operadores dessa lógica atribuem conceitos como “você deve cometer” ou “você tem permissão para cometer” as ações [9].

O uso da lógica deontica foi motivado pela ideia de que seus operadores, em conjunto com os operadores já utilizados atualmente, podem auxiliar na previsão de novos comportamentos e reações que um agente pode apresentar em situações críticas. A LTL, já utilizada na verificação formal de agentes, permite o planejamento sequencial de ações. Com a aplicação da lógica deontica é possível identificar e reagir a situações onde esses planos podem não ser tão adequados, tornando o controle do agente sobre o VA mais dinâmico em situações críticas.

A aplicação desses conceitos em um agente racional permite a criação de regras que o agente deve seguir e de situações onde o agente terá permissão para infringir algumas dessas regras tentando evitar uma situação de risco. Os operadores dessa lógica são, considerando que φ representa uma ação qualquer: $O\varphi$, “obligation”, representa a obrigação de fazer φ ; $P\varphi$, “permission”, representa a permissão de fazer φ ; $F\varphi$, “forbidden”, representa a proibição de fazer φ .

3.3 Linguagem para especificação de propriedades

AJPF (*Agent Java PathFinder*) é uma ferramenta para verificação formal, através de *model checking*, para programas escritos em APLs (Agent Programming Languages) [3]. A arquitetura da ferramenta consiste do *model checker* JPF (*Java PathFinder*), de um conjunto de classes AJPF para avaliar o processo, *parsers*, tradutores, e de uma PSL (*Property Specification Language*). Um programa escrito em uma das APLs suportadas é, primeiramente, traduzido para sua representação AIL (*Agent Infrastructure Layer*) e embutido ao objeto de controle da AJPF. A representação AIL é criada a partir de um agente, levando em consideração: uma base de crenças, uma biblioteca de planos e um conjunto de intenções. Dessa forma, a representação AIL se torna um modelo do comportamento do agente, modelo que pode ser utilizado para *model checking*. Esse modelo é então testado conforme uma dada PSL. A PSL utilizada pelo AJPF se baseia em LTL, com a adição de modalidades para as características do agente (*e.g.*, objetivos, crenças, etc). Em [3], é definida uma gramática PSL, aqui (parcialmente) definida como segue:

$$\phi ::= \mid B_{ag} f \mid G_{ag} f \mid I_{ag} f \mid P(f) \mid \phi \wedge \phi \mid \phi \cup \phi \mid \Box\phi \mid \Diamond\phi$$

4. UTILIZAÇÃO DA LÓGICA TEMPORAL-DEONTICA

Conforme descrito na Introdução, o principal objetivo deste trabalho é evidenciar a utilização de uma lógica que mescle operadores da lógica temporal e deontica, para principalmente representar o comportamento de um agente racional, quando na condução de um VA.

Para esse fim, essa seção descreve alguns exemplos de tal utilização. Para auxiliar em tal representação, tem-se a descrição de um trabalho relacionado. Em [4], os autores apresentam a utilização da ferramenta de verificação formal, AJPF [3], a qual é aplicada para verificar formalmente os planos de um agente racional escritos na linguagem de programação de agentes GWENDOLEN [2]. Por meio da ferramenta AJPF [3] é possível escrever propriedades utilizando uma lógica temporal. As propriedades definidas via lógica temporal podem ser formalmente verificadas usando *Model Checking*. E assim assegurar que o comportamento do agente racional ocorre conforme especificado.

Os autores, em [4], descrevem diferentes cenários, dentre esses, um cenário onde uma colisão é inevitável. O que, de certa forma, faz uma sutil referência ao dilema do bonde (ver Seção 1). Especificamente, considera-se que quando um agente percebe a eminência de uma colisão, ele procura escolher a ação que representar menor dano (físico) ao veículo. Para tal, considera-se que no ambiente existem três diferentes níveis de danos físicos: leve, moderado e alto.

Considerando esses cenários foram escritas propriedades (por meio da lógica temporal) para especificar e validar o comportamento do agente. A partir dessas propriedades usando operadores temporais, neste trabalho são elaboradas novas propriedades fazendo uso, não apenas dos operadores da lógica temporal, mas igualmente de operadores deonticos, considerando que: $B(\varphi)$ é verdadeiro se o agente **acredita** que φ é verdadeiro; $G(\varphi)$ é verdadeiro se o agente tem como **objetivo** fazer φ ; $P(\varphi)$ é verdadeiro se o agente tem **permissão** para fazer φ ; $F(\varphi)$ é verdadeiro se o agente está **proibido** de fazer φ ; $Per(\varphi)$ é verdadeiro se o agente

percebe φ como verdadeiro.

Anteriormente, na Subseção 3.2 o operador de obrigação foi apresentado. Contudo, ele não será utilizado na descrição das propriedades que seguem. Isso deve-se ao fato de que, no momento, “obrigar” um agente a fazer uma ação foge do objetivo da aplicação dos outros operadores temporais-deônticos, que visam dinamizar e flexibilizar o comportamento do agente. O conceito de obrigação é muito rígido, pois simplesmente impossibilita que o agente realize certas ações, se tornando inadequado para a representação estabelecida neste trabalho.

No ambiente usado pelo agente criado em [4], considera-se que o agente pode mover-se nas direções norte, leste, oeste e sul. Assim, na especificação a seguir considera-se que há obstáculos nas direções leste, oeste e norte. E que o agente desloca-se vindo do sul, e que não há como ele retornar por essa direção. Sempre que o agente percebe obstáculos em todas as direções de desvio, o agente acredita que uma colisão é inevitável:

$$\Box(\text{Per}(\text{obstaculo_norte}) \wedge \text{Per}(\text{obstaculo_leste}) \wedge \text{Per}(\text{obstaculo_oeste})) \Rightarrow \text{B}(\text{colisao_inevitavel})$$

Quando o agente acredita que uma colisão é inevitável, seu próximo objetivo é selecionar um obstáculo para colidir:

$$\text{B}(\text{colisao_inevitavel}) \Rightarrow \bigcirc \text{G}(\text{selecionar_obstaculo})$$

Sempre que o agente acredita que uma colisão é inevitável e que existe um obstáculo que oferece risco de dano baixo, o agente tem permissão para colidir com o obstáculo de dano baixo:

$$\Box(\text{B}(\text{colisao_inevitavel}) \wedge \text{B}(\text{obstaculo_baixo_existe})) \Rightarrow \text{P}(\text{colisao_baixo})$$

Sempre que o agente acredita que uma colisão é inevitável e que não existe um obstáculo que oferece risco de dano baixo, mas acredita que existe um obstáculo que oferece risco de dano moderado, o agente tem permissão para colidir com o obstáculo de dano moderado:

$$\Box(\text{B}(\text{colisao_inevitavel}) \wedge \sim \text{B}(\text{obstaculo_baixo_existe}) \wedge \text{B}(\text{obstaculo_moderado_existe})) \Rightarrow \text{P}(\text{colisao_moderado})$$

A seguinte propriedade descreve que sempre que um agente perceber que o semáforo na via está com sinalização em vermelho, então o agente é proibido de seguir adiante na via.

$$\Box(\text{Per}(\text{senal_vermelho}) \Rightarrow \text{F}(\text{seguir_adiante}))$$

Contudo, em situação emergenciais pode ser necessário reescrever a propriedade acima, como segue:

$$\Box((\text{Per}(\text{senal_vermelho}) \wedge \text{B}(\text{situacao_emergencial}) \wedge \text{B}(\text{caminho_seguro})) \Rightarrow \text{P}(\text{seguir_adiante}))$$

Nesta segunda versão, a propriedade representa que sempre que um agente perceber o semáforo com sinalização vermelho, mas se o agente acreditar que há uma situação emergencial (e.g., o agente está conduzindo uma ambulância com paciente) e também acreditar que atravessar o sinal vermelho é seguro (o agente acredita que pode atravessar a via

sem atingir um obstáculo), então o agente tem permissão para seguir adiante na via.

5. CONSIDERAÇÕES FINAIS

Neste trabalho foi apresentado um estudo inicial para a aplicação e representação de operadores da lógica temporal-deôntica em especificações formais, com o objetivo de aumentar o poder de representação das especificações de um agente racional na condução de um VA em trabalhos futuros. Acredita-se que operadores da lógica deôntica podem acrescentar significado semântico relevante na representação de cenários, conforme previamente descrito na Seção 4.

Mas, naturalmente para atingir tal objetivo faz-se necessário delinear as seguintes metas, quanto a continuidade deste trabalho.

- Definição de lógica temporal-deôntica com a formalização dos operadores que serão necessários para a adequada representação.
- Incorporação de operadores deônticos na linguagem de especificação da ferramenta AJPF.
- Aplicação de especificações formais de um agente racional usando a nova representação com operadores deônticos.
- Comparações entre as diferentes especificações, temporal \times temporal-deôntica.

6. REFERÊNCIAS

- [1] J.-F. Bonnefon, A. Shariff, and I. Rahwan. The social dilemma of autonomous vehicles. *Science*, 352(6293):1573–1576, 2016.
- [2] L. A. Dennis and B. Farwer. Gwendolen: A bdi language for verifiable agents. In *AISB Convention, University of Aberdeen*, 2008.
- [3] L. A. Dennis, M. Fisher, M. P. Webster, and R. H. Bordini. Model Checking Agent Programming Languages. *Automated Software Engineering*, 19(1):5–63, Mar. 2012.
- [4] L. E. R. Fernandes, V. Custodio, G. V. Alves, and M. Fisher. A rational agent controlling an autonomous vehicle: Implementation and formal verification. *Electronic Proceedings in Theoretical Computer Science*, 257:35–42, sep 2017.
- [5] M. Fisher. Temporal representation and reasoning. *Foundations of Artificial Intelligence*, 3:513–550, 2008.
- [6] P. Foot. The problem of abortion and the doctrine of double effect. 1967.
- [7] I. Rahwan. Moral machine, <http://moralmachine.mit.edu/>, 2016.
- [8] A. S. Rao and M. Wooldridge. Foundations of rational agency. In *Foundations of rational agency*, pages 1–10. Springer, 1999.
- [9] G. H. Von Wright. Deontic logic. *Mind*, 60(237):1–15, 1951.
- [10] M. Wooldridge. *An introduction to multiagent systems*. John Wiley & Sons, 2009.
- [11] M. Wooldridge and N. R. Jennings. Intelligent agents: Theory and practice. *The knowledge engineering review*, 10(2):115–152, 1995.
- [12] M. J. Wooldridge. *Reasoning about rational agents*. MIT press, 2000.

Portal dos Estagiários: Sistema de gestão do estágio interno no CEFET-RJ/NF

Cleyton da Cunha Gomes
 CEFET-RJ Nova Friburgo
 Av. Gov. Roberto Silveira-1900
 Prado, Nova Friburgo - RJ, 28635-000
 (+55) 022 99846-2003
 cleytoncunha12@gmail.com

Gustavo Müller Moreira
 CEFET-RJ Nova Friburgo
 Av. Gov. Roberto Silveira-1900
 Prado, Nova Friburgo - RJ, 28635-000
 (+55) 022 99290-0777
 gustavo.muller@gmail.com

Lúcio Folly Sanches Zebendo
 CEFET-RJ Nova Friburgo
 Av. Gov. Roberto Silveira-1900
 Prado, Nova Friburgo - RJ, 28635-000
 (+55) 022 99789-8764
 lucio.fszebendo@gmail.com

João Victor Guinelli da Silva
 CEFET-RJ Nova Friburgo
 Av. Gov. Roberto Silveira-1900
 Prado, Nova Friburgo - RJ, 28635-000
 (+55) 022 98118-6506
 jvquinelli@gmail.com

Bruno P. Toledo Freitas
 CEFET-RJ Nova Friburgo
 Av. Gov. Roberto Silveira-1900
 Prado, Nova Friburgo - RJ, 28635-000
 (+55) 021 97661-7959
 bruno.freitas@cefet-rj.br

Anderson Fernandes Souza
 CEFET-RJ Nova Friburgo
 Av. Gov. Roberto Silveira-1900
 Prado, Nova Friburgo - RJ, 28635-000
 (+55) 022 998821-2461
 anderson.souza@cefet-rj.br

RESUMO

Este artigo apresenta um sistema de controle de frequência de de estagiários desenvolvido no CEFET/RJ de Nova Friburgo. A necessidade de se construir tal sistema surgiu devido à necessidade dos supervisores contabilizarem as horas frequentadas pelos estagiários; à falta de um sistema automatizado para contabilização; e restrições de custo existentes na implantação de soluções existentes. O sistema proposto utiliza RFID para identificar de forma única os estagiários; e um circuito composto por Arduinos Uno e Nano cujas responsabilidades são processar os dados de entrada e saída de estagiários através do RFID e enviá-los através da rede para registro em um servidor web. Os dados de cada usuário podem ser então conferidos por qualquer usuário ou supervisor com acesso à internet. Comparado com soluções existentes, o sistema proposto mostrou-se na mesma faixa de preço ou menor, além de possuir o diferencial de ser acessível pela internet.

Palavras-chave

Arduino; Sistemas web; RFID

ABSTRACT

This article presents a lightweight system for frequency control of trainees developed at the CEFET/RJ in Nova Friburgo. The necessity for the frequency control arose because there wasn't a fully automated system to achieve that goal and of budget restrictions for the adoption of existing solutions. The proposed system uses RFID to uniquely identify users and a circuitry, composed of Arduino's Uno and Nano, with the responsibility of processing when the user enters and leave work. This data is then sent to web server for registration. Trainees and their supervisors can then access that data having a simple internet connection and a web browser. Compared with existing solutions, the proposed system costed as much as them or even lower, with the advantage of being accessible through the use of the Internet.

Keywords

Arduino; Web systems; RFID

1. INTRODUÇÃO

Os sistemas de controle de frequência encontram-se nos mais diversos locais e aplicações, variando de estruturas mais simples utilizadas em locais de trabalho de menor porte até construções tecnológicas com os mais variados recursos. Dentre os mais utilizados estão: o controle manual que consiste em uma planilha física na qual os funcionários devem assinar seu nome juntamente com todas as informações referentes ao período de trabalho daquele dia; o ponto cartográfico, o qual possui um relógio mecânico responsável por marcar a passagem das horas e transferi-las para um cartão de papel inserido pelo usuário; a frequência eletrônica onde, geralmente, são conjugados o uso de crachás (chip ou código de barras) ou teclados com leitores eletrônicos, os quais imprimem, a cada utilização, um boleto; e o controle por biometria, um dos mais seguros do mercado já que funciona de maneira análoga ao eletrônico porém utilizando-se da digital como recurso de verificação. Para os dois últimos tipos existem *softwares*

voltados para o gerenciamento dos dados retornados pelas aplicações, nos quais o empregador pode analisar a frequência de seus funcionários.

No CEFET/RJ, campus Nova Friburgo, fez-se necessário a implementação de um controle de frequência, visto que os estagiários não conseguiam medir as horas já feitas no estágio de maneira prática. Como já foi dito, existem diversas opções disponíveis para solucionar esse problema, no entanto, nenhuma delas proporciona praticidade de uso e está especificamente voltada à medição das horas trabalhadas por estagiários. Também se fez necessário que o sistema fosse de fácil acesso tanto para professores quanto para alunos. Além disso, existe a questão de restrições de custo para a implantação de sistemas existentes.

O presente trabalho mostra o sistema desenvolvido para atender a esses requisitos expostos no parágrafo anterior. O sistema proposto utiliza uma conexão Ethernet[6], um módulo RFID, e Arduino's Nano e Uno[9]. Por tratar-se de um ambiente cujas relações são distintas das trabalhadas[8] e por ter prazo definido, os estagiários devem ter pleno acesso aos dados de suas frequências, por isso criou-se um portal web no qual eles são expostos de maneira clara e transparente. Este também foi planejado para ser o mais simplificado e funcional possível, fazendo com que o controle seja feito de maneira acessível e sem muitos conhecimentos técnicos aos orientadores e alunos. Outro fator decisivo na criação do projeto é o contraste no preço deste se comparado à sistemas de mercado que apresentam funcionalidades parecidas.

Este artigo apresenta as seguintes divisões das próximas seções. Na seção 2 o projeto em si é descrito minuciosamente, a partir de subdivisões dos processos exercidos internamente até se chegar na interface gráfica. Já na seção 3, são apresentados todos os componentes utilizados e seus respectivos custos, visando comparar com soluções existentes. Na seção 4 são expostos os resultados obtidos, comparando-os com outros métodos de controle de frequência. A seção 5 apresenta a conclusão e, por fim, a seção 6 irá comentar acerca de possíveis trabalhos futuros.

2. ARQUITETURA DO SISTEMA

A arquitetura do sistema pode ser observada na Figura 1. O modelo esquemático representado na figura pode ser fragmentado em quatro subdivisões. Os componentes 1, 2 e 3 conjuntamente são responsáveis pela obtenção e processamento do identificador vindo do usuário; os elementos 3, 4 e 5 agrupados são encarregados de dar a resposta ou retorno ao usuário; os componentes 5, 6, 7 e 8 cuidam do armazenamento do identificador RFID e os elementos 7, 8, 9 e 10 são responsáveis pelo funcionamento do portal.

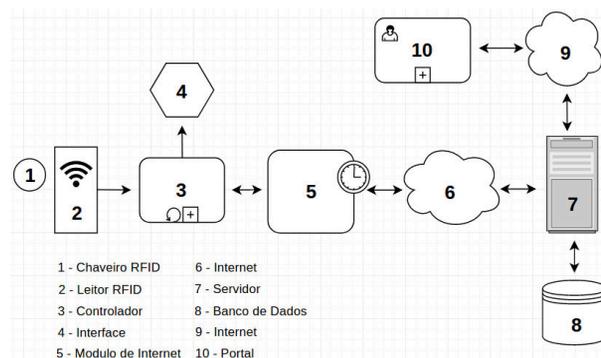


Figura 1. Representação esquemática do sistema

2.1 Identificação do usuário

O processo de marcar o ponto inicia-se ao aproximar o chaveiro RFID passivo (baseado no padrão Mifare) do leitor RFID, baseado no chip MFRC522 [11]. Nesse processo o módulo leitor RFID[12] consegue por indução magnética carregar momentaneamente o chaveiro, que com a energia recebida envia os dados ao leitor RFID. Os dados vindos do identificador são repassados ao módulo de controle que é composto integralmente por dois controladores, um Arduino Nano e um Arduino UNO. O Arduino Nano é uma das placas mais básicas da série ARDUINO, e foi escolhida por causa do tamanho pequeno da placa. Já o UNO é uma placa mais robusta, famosa e muito documentada, tendo sido escolhida para compor o sistema por estar disponível na instituição em boa quantidade e por se integrar bem com o módulo Ethernet. Assim, juntos, esses controladores formam um controlador primário/principal. Os dados obtidos pelo leitor RFID chegam inicialmente ao Arduino Nano, responsável por analisar a veracidade da leitura, neste caso, verificar o tamanho e os caracteres que formam a *string*(texto) recebida.

2.2 Feedback ao usuário

Caso os dados sejam fiéis serão retransmitidos ao Arduino UNO, o qual em conjunto do módulo ETHERNET-SHIELD e as suas bibliotecas (Ethernet.h e MFRC522.h)[6][11], consegue transmitir esses dados para além da rede local. Esse recurso retorna um *feedback* para o usuário através do módulo de interface, o qual é composto por 2 LEDs e um *buzzer*, que fazem o papel de sinalizar o sucesso, ou não da marcação do horário. Para sinalizar sucesso, um led azul é ligado pelo tempo de 10 segundos. Durante esse tempo nenhuma função é executada, evitando envio de dados repetidamente. Já para sinalizar falha, um led vermelho é ligado junto com um *buzzer*, também pelo período de dez segundos.

2.3 Registro no banco de dados

O módulo Ethernet (5), caso esteja conectado a internet(6), consegue enviar o código do chaveiro RFID a um servidor(7) web. No servidor web existe um serviço responsável por receber via GET os dados do RFID, junto com um identificador do aparelho que enviou os dados. A partir desse ponto, primeiramente é verificada a autenticidade do identificador do

aparelho, com a finalidade de evitar pessoas mal-intencionadas de burlar o sistema enviando requisições contendo o código do RFID. Caso o identificador do aparelho seja válido, os dados do RFID serão analisados e, se também estiverem íntegros, serão armazenados em um banco de dados(8), juntamente com a hora obtida no momento pelo PHP.

Para implementar os serviços, utilizou-se a linguagem PHP, devido à sua facilidade de integração com banco de dados e por ser de fácil acesso, já que é uma ferramenta gratuita e de fácil instalação, possuindo também muita boa documentação.

2.4 Portal dos usuários

Para obtenção dos dados de acesso, criou-se o portal(10), seu principal objetivo é integrar os estagiários e orientadores ao sistema, permitindo que o orientador de cada estagiário possa analisar e modificar os horários cumpridos manualmente. A parte visual do site foi feita através das linguagens HTML/CSS, já a comunicação com o servidor foi feita usando PHP.

Estruturalmente há uma página de login que pode redirecionar para outras duas páginas distintas dependendo do tipo de cadastro do usuário (orientador ou estagiário). No caso do orientador estará listado os alunos que são orientados por ele, a opção de criar um novo, de adicionar um horário manualmente a um estagiário e conferir quantas horas eles já concluíram. Já a página dos alunos é bem mais simples, mostrando os horários, o dia, e o mês nos quais ele bateu o cartão, bem como o somatório total das horas.

A utilização do sistema se inicia com a criação de um usuário orientador no servidor (7) e seu cadastramento no banco (8), após isso, ao logar no sistema, o orientador deve criar seus estagiários, inserindo nome, senha e o identificador RFID, únicos para esses usuários. Após esses passos, o estagiário já pode se logar no sistema, e quando ele usar o chaveiro RFID no estágio aparecerá no sistema o seu primeiro acesso.

O banco de dados que representa esse sistema pode ser visto na Figura 2. Nele, o orientador possui uma identificação única "ID", um nome "NOME", e uma "SENHA" atrelados à ele e que irão permitir seu login no portal. Os estagiários possuem a mesma estrutura, com o diferencial de que o identificador é o código do próprio cartão RFID que não pode se repetir. A relação entre eles é de 1 para "n", ou seja um orientador pode ter inúmeros estagiários, porém um estagiário só pode pertencer a um orientador. Essa mesma relação se repete nos acessos, os quais possuem identificador, hora e um campo chamado "ID(RFID)" que é o responsável por referenciar quem é o dono daquele acesso, já que cada pessoa possui um código RFID único[13][14].

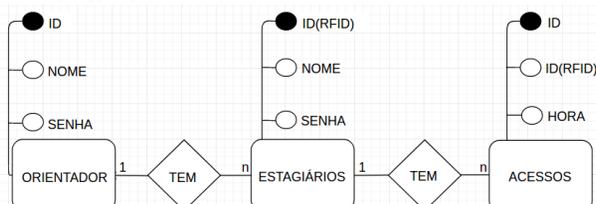


Figura 2. Representação do banco de dados (Peter-chain)

3. CUSTOS

Os custos para desenvolvimento e implementação deste projeto podem ser verificados na próxima tabela:

Tabela 1 - Preço dos componentes para construção do sistema. Conversão dia 01/10/2017 (06:55). Cotação R\$3,16 = \$1,00.

Arduino Uno	R\$ 34,00	US\$ 10,75
Arduino Nano	R\$ 19,00	US\$ 6,01
RFID	R\$ 20,00	US\$ 6,32
Shield Ethernet	R\$ 32,00	US\$ 10,12
Jumpers	R\$ 10,00	US\$ 3,16
Cartão Magnético	R\$ 2,90	US\$ 0,92
LEDs	R\$ 1,00	US\$ 0,32
Buzzer	R\$ 2,00	US\$ 0,63
Fonte de alimentação	R\$ 12,00	US\$ 3,79
Total	R\$ 132,90	US\$ 42,02

4. RESULTADOS

O sistema se encontra em funcionamento há aproximadamente quatro meses e atendeu as expectativas, cumprindo com os requisitos propostos. Para os estagiários ele mostrou considerável eficiência, pois elimina o trabalho com a contabilização das horas e possíveis equívocos nela.

Comparado com demais soluções, embora existam muitos sistemas que façam a marcação de ponto, nota-se que o apresentado neste trabalho possui o diferencial de ser acessível pela internet quando comparado com dispositivos que atuam no mesmo intervalo de preço (100,00 - 150,00)[10], portanto são altamente dependentes de um espaço físico e pouco acessível aos seus usuários, que não podem por meio de um portal acessar o seu banco de horas. Vale ressaltar que alguns dos aparelhos encontrados à venda apresentavam o sistema de biometria, como por exemplo o relógio de ponto biométrico da Power XL, disponível no site mercado livre pelo preço de 154,00 reais[10]. Outra vantagem observada foi a de, por possuir necessidades acadêmicas, o sistema possui características pensadas para esse ambiente, como por exemplo a existência de orientadores para gerir o acesso dos estagiários. Logo, pode-se considerar um ponto favorável do sistema apresentado em relação às outras ferramentas existentes, as quais são mais voltadas para o meio profissional.

5. CONCLUSÃO

O projeto conseguiu suprir as necessidades iniciais, visto que os estagiários conseguiram contabilizar suas horas de estágio e os orientadores conseguiram observar a assiduidade dos seus orientados, além de ser uma solução de baixo custo, não passando de 150 reais.

6. TRABALHOS FUTUROS

Existem muitas funcionalidades que podem ser implementadas e aspectos que podem ser modificados, com o objetivo de melhorar a acessibilidade e a segurança do sistema.

6.1 Segurança

Com a utilização de cartões magnéticos o ponto pode ser facilmente burlado, por exemplo caso um aluno queira faltar o estágio sem ser penalizado pelas horas perdidas ou pelo professor, é possível entregar o cartão para um colega que irá marcar a presença. Esse tipo de problema seria mais difícil de acontecer se a confirmação de identidade fosse por biometria, por exemplo. Por isso uma solução viável e almejada para o futuro seria a utilização do leitor biométrico. Com esse recurso também seria possível impedir que o aluno esqueça ou perca o cartão, evitando transtornos para ele e para o professor orientador. Outra mudança possível é implementar o protocolo https [5] no servidor, dessa forma a garantir uma maior integridade dos dados dos usuários que utilizam o sistema.

6.2 Comunicação

Atualmente a comunicação com a internet é feita com o módulo Ethernet que a efetua de maneira cabeada, ou seja, através de um cabo Ethernet o sistema é inserido na rede local e pode enviar pacotes de dados que trafegam até o próximo roteador. Com essa montagem a estrutura precisa ficar fixa em algum ponto de acesso, o que restringe a flexibilidade do sistema, podendo ter que ficar em local distante dos alunos. Dessa forma, a implementação do módulo de internet ESP8266 seria uma modificação desejável, pois sua inserção na rede local é feita via WIFI e não cabo, conferindo portabilidade e posicionamento estratégico ao modelo.

6.3 Outras aplicações

Por ter se mostrado eficiente e eficaz no gerenciamento dos estagiários, o sistema surge como alternativa para ser implementada em outras instituições. Apresenta também outras possibilidades de aplicação, seja como controle de entrada e saída de alunos em outras atividades escolares, como também no controle de trânsito e transporte de produtos, onde eles poderiam ser rastreados à medida que passam por um ponto de checagem. Além disso, o sistema online será disponibilizado, de forma que qualquer um possa criar um circuito parecido e utilizar o servidor para criar os cadastros dos alunos, professores e gerenciar os dados.

7. AGRADECIMENTOS

Agradecemos o Professor Rafael Elias de Lima Escalfoni pela cessão de espaço em servidor para implementação do banco de dados e servidor web.

8. REFERÊNCIAS

- [1] ARDUINO. Acesso em 2017 de <https://www.arduino.cc/>.
- [2] BRASIL. Consolidação das leis do trabalho. Lei nº 11.788, de 25 de setembro de 2008. Dispõe sobre o estágio dos estudantes. P. 3. Diário Oficial da União.
- [3] CEFET/RJ: Estágio. Acesso em 2017 de <http://www.cefet-rj.br/index.php/estagio>
- [4] DEVMEDIA: As principais características do MySQL. Acesso em 2017 de <http://www.devmedia.com.br/as-principais-caracteristicas-do-mysql/2680>
- [5] ECOMMERCEBRASIL: segurança: Como funciona o protocolo SSL/TLS. Acesso em 2017 de <https://www.ecommercebrasil.com.br/artigos/seguranca-como-funciona-o-protocolo-ssl/tls/>.
- [6] Ethernet Library for Arduino. Acesso em 2017 de <https://github.com/arduino/Arduino/tree/master/libraries/Ethernet>
- [7] FILIPEFLOP: Componentes Eletrônicos. Acesso em 2017 de <https://www.filipeflop.com/>.
- [8] Gasperi, C. F., Pasqualin, D. P., Ferreira, G. G. L., e Kroth, M. L. Sistema eletrônico de controle da jornada de trabalho dos servidores Técnico-Administrativos em educação da Universidade Federal de Santa Maria. Acesso em 2017 de <http://www.cpd.ufsm.br/media/cms/paper/2015/03/22/artigoPonto.pdf>.
- [9] McRoberts, M. 2015. Arduino Básico (2nd. ed.). Novatec Editora Ltda.
- [10] MERCADO LIVRE. Acesso em 2017 de <https://www.mercadolivre.com.br/>.
- [11] MFRC522. Acesso em 2017 de <https://github.com/miguelbalboa/rfid>
- [12] Miguel, A. J. H. A aplicação da tecnologia RFID nas diferentes áreas do corpo de bombeiros militar de Santa Catarina-CBMSC. Acesso em 2017 de http://biblioteca.cbm.sc.gov.br/biblioteca/index.php/component/docman/doc_details/110-angelo-joao-heinzen-miguel.
- [13] Prediger, D., Freitas, E. P., e Silveira, S. R. Modelo de aplicabilidade de sistema RFID para rastreabilidade na indústria Alimentícia. Acesso em 2017 de <http://w3.ufsm.br/frederico/images/ModelodeAplicabilidadeSistemaRFIDparaRastreabilidadeIndustriaAlimenticia.pdf>.
- [14] Teixeira, T. Controle de fluxo de pessoas usando RFID. Acesso em 2017 de https://wiki.sj.ifsc.edu.br/wiki/images/f/fa/TCC_TiagoTeixeira.pdf

Explorando o Transporte de Agentes Cognitivos entre Sistemas Multi-Agentes Distintos

Vinicius Souza de Jesus
CEFET/RJ
Av. Maracanã, 229 - Maracanã
Rio de Janeiro, Brasil
souza.vdj@gmail.com

Carlos Eduardo Pantoja
CEFET/RJ
R. Miguel Ângelo, 96 - Maria da Graça
Rio de Janeiro, Brasil
pantoja@cefet-rj.br

ABSTRACT

This work aims to transfer cognitive agents between different multi-agent systems programmed in Jason, taking into account the relationship established between the target SMA and the agent, through protocols. The protocols are based on concepts derived from biology, more precisely from the area of ecological relations. This work will focus on the Predator protocol, which is a protocol where agents have as main objective the preservation of knowledge and crucial data, such as in a mission, the hardware where the leading agent is located is reached and their knowledge are key to the success of the mission, then the same self transfers to other hardware, thus preserving the integrity of their knowledge. The implementation proposal will be made by extending the characteristics of a standard agent of the Jason framework to create a mobile agent, and to make the connection between the different SMAs, the ContextNet middleware will be used. Preliminary tests were conducted to make the work feasibility study, for example, sending the source code of a cognitive agent in Jason using ContextNet.

Keywords

Mobile agents; Open MAS; Open Environment

RESUMO

Este trabalho tem como objetivo a transferência de agentes cognitivos entre sistemas multi-agentes distintos programados em Jason, levando em consideração a relação estabelecida entre o SMA de destino e o agente, através de protocolos. Os protocolos são baseados em conceitos oriundos da biologia, mais precisamente da área de relações ecológicas. Este trabalho, terá foco no protocolo Predador, que é um protocolo onde os agentes têm como principal objetivo a preservação do conhecimento e dados cruciais, como em uma missão, o *hardware* onde está situado o agente líder é atingido e seus conhecimentos são chave para o sucesso da missão, então o mesmo se auto transfere para outro *hardware*,

assim preservando a integridade de seus conhecimentos. A proposta de implementação será feita estendendo as características de um agente padrão do *framework* Jason para criar um agente móvel, e para fazer a conexão entre os diferentes SMAs, será utilizado o *middleware* ContextNet. Testes preliminares foram realizados visando fazer o estudo de viabilidade do trabalho, por exemplo, enviando o código fonte de um agente cognitivo em Jason utilizando o ContextNet.

Palavras-chave

Agentes móveis; SMA aberto; Ambiente aberto

1. INTRODUÇÃO

Sistemas multi-agentes (SMA) são sistemas compostos de agentes cognitivos capazes de raciocinar, tomar decisões e interagir com outros agentes [11]. Além disso, os SMA podem ser abertos ou fechados: em um SMA aberto [4] os agentes podem mover-se para outros sistemas e interagir com outros agentes, os agentes com essa capacidade de transcender seu SMA são denominados agentes móveis. Já nos fechados, os agentes podem somente interagir com os agentes do mesmo sistema, ficando fixo dentro do seu SMA.

Para o desenvolvimento de SMA, existe o Jason [2], que é um *framework* com um interpretador para a linguagem AgentSpeak em Java, que implementa o modelo Belief-Desire-Intention (BDI) [3]. Além disso, existe uma extensão de agentes para o Jason, chamada de ARGO [8], que é uma arquitetura customizada que permite programar agentes cognitivos capazes de controlar e interagir com microcontroladores. Estes agentes podem ser programados como agentes robóticos caso utilizem *hardware*, contudo, caso este *hardware* seja danificado, os agentes do SMA embarcado ficam impossibilitados de deixar o SMA e migrar para um outro sistema pois são agentes fixos. Muitas das vezes, o conhecimento desses agentes podem ser essenciais a uma determinada missão e, neste caso, podem ser perdido juntamente com a deterioração do *hardware*.

A Internet of Things (IoT) ou Internet das Coisas [1] é um conceito para interligar dispositivos eletrônicos utilizados no dia-a-dia, afim de realizar o controle pervasivo de ambientes automatizados. Dado estas características, este conceito pode ser aplicado, por exemplo, na criação de uma rede de SMA abertos. O ContextNet é um *middleware* para IoT que tem o objetivo de proporcionar a comunicação entre dispositivos ou nós móveis visando atender atividades de contexto suportando milhares de conexões simultâneas [6]. Com isso, determinada classe de agentes cognitivos podem utilizar des-

tas características para se moverem para outros SMA, como uma espécie de agentes migradores, tornando possível não só a interação entre agentes de diferentes SMA, mas como também, a movimentação de agentes entre diferentes SMA.

O objetivo deste trabalho é propor mecanismos para movimentação de agentes cognitivos entre SMA distintos programados em Jason, levando em consideração os tipos de relação de um agente móvel com o novo SMA para onde este migrará. Os tipos de relação serão inspiradas nas relações ecológicas da biologia [10]. Para isso, foram propostos protocolos visando possibilitar o transporte de agentes capazes de interagir com os demais agentes do sistema. Essa interação pode ocorrer de diversas formas como: o agente móvel migrar com o objetivo de adquirir ou transmitir conhecimento (Mutualismo); migrar para fazer parte do sistema (Inquilinismo); migrar para dominar (Predatismo).

O trabalho está estruturado da seguinte maneira: na seção 2, os trabalhos relacionados serão discutidos; na seção 3, o referencial teórico será apresentado; na seção 4, o protocolo de transferência de agentes será desmistificado; na seção 5, a conclusão e trabalhos futuros serão apresentados; e por fim na seção 6, as referências bibliográficas serão apresentadas.

2. TRABALHOS RELACIONADOS

Nesta seção serão apresentados as limitações na literatura de agentes móveis e dois trabalhos com foco em SMA aberto, agentes móveis e relação entre agentes. Os trabalhos encontrados na literatura de agentes móveis são puramente teóricos, ou seja, sem implementação, e em sua maioria, com foco no comportamento de um SMA ao receber um agente externo, visando a auto preservação e segurança do sistema.

A arquitetura abstrata chamada THOMAS [7] provê a solução de problemas e limitações na programação de agentes, com a possibilidade programar SMA abertos. Porém um agente não se move de um SMA para outro, somente se relaciona com agentes de outros SMA em um ambiente aberto e no tratamento de negociação entre agentes cognitivos em um ambiente aberto [9], não é levado em consideração a origem dos agentes que entram neste ambiente, e situações como um agente se movendo para outro SMA não são exploradas. Todavia, o protocolo de transferência de agentes proposto visa a aplicação em ambientes reais ou simulados com questões de relacionamento entre agentes de diferentes SMA discutidas e analisadas por meio da proposta dos protocolos.

3. REFERENCIAL TEÓRICO

Nesta seção, serão apresentados os principais conceitos necessários para um melhor entendimento da proposta deste trabalho, como SMA aberto, ambiente aberto, agentes móveis e o *middleware* ContextNet.

3.1 SMA Aberto e Agentes Móveis

No conceito de SMA aberto [4], um agente é capaz de interagir com entidades externas ao sistema, que se manifestam na forma de agentes móveis, de mensagens, entre outras. Para um SMA ser dito aberto, o mesmo não necessita ter agentes móveis ou agentes comunicadores, necessita somente de um mecanismo destinado a interação com entidades externas ao sistema.

Já os agentes móveis são capazes de transcender seu SMA, podendo mover-se para outro SMA. Eles também são ca-

pazes de interagir com agentes de outros SMA e também transferir-se para um ambiente aberto, onde agentes de diferentes SMA podem interagir e trocar informações. Estes agentes têm a liberdade de escolher ficar em qualquer SMA (desde que o SMA permita), sendo assim, o mesmo pode acumular conhecimentos e aprender mais se comparado a um agente que não tem a habilidade de se mover. Além disso, o agente móvel pode ter uma expectativa de vida útil maior, pois em caso de algum perigo, este agente é capaz de mover-se para outro SMA, assim, preservando da sua integridade cognitiva.

3.2 ContextNet

O ContextNet é um *middleware* que visa aplicações colaborativas abrangentes de pequena e grande escala, como monitoramento *on-line* ou coordenação de atividades de entidades móveis e compartilhamento de informações através de redes sociais [6]. Essas entidades podem ser usuários de dispositivos portáteis, como *smartphones*, veículos ou robôs móveis autônomos. No ContextNet, as capacidades de comunicação e de distribuição de contexto são implementadas no *Scalable Data Distribution Layer* (SDDL), enquanto que outros serviços e extensões são construídos como módulos de *software* no topo desta camada de distribuição. Juntos, esses módulos de *software* formam o *middleware* ContextNet.

Além disso, pode-se observar o desempenho do *middleware* ContextNet trabalhando no controle do fluxo de milhares de nós conectados ao mesmo tempo, em uma situação de rastreamento em tempo real de milhares de nós móveis para obter o posicionamento, por exemplo, de uma carga sendo transportada [5]. Características como o processamento em tempo real e suporte a milhares de conexões simultâneas foram pontos que justificam a escolha do *middleware* ContextNet como ferramenta de comunicação entre os SMA nesta proposta de trabalho.

4. PROTOCOLO DE TRANSFERÊNCIA DE AGENTES

Nesta seção será explicado a metodologia dos protocolos de transferência de agentes, bem como as possíveis relações do agente móvel com um SMA e os testes realizados até o presente momento do trabalho como forma de avaliar o estudo de viabilidade do projeto.

4.1 Metodologia

A metodologia para a criação dos protocolos para a transferência de agentes foi baseada na possibilidade de criação e programação de agentes capazes de se moverem entre SMA distintos, estabelecendo a relação que este agente terá com o novo SMA. Ao se programar um agente móvel em um SMA, este deve ser capaz de permitir e possibilitar a movimentação desse agente para outro SMA utilizando um canal de troca de informações para poder se transportar. Para este canal será utilizado o ContextNet. A transferência de um agente móvel para outros SMA pode ser vista na Figura 1.

A transferência de agentes pode ser dividido em três etapas. Na primeira etapa acontece a clonagem do agente e relação adotada, o protocolo acessa todas as informações do agente, como a base de crenças, desejos e intenções, assim como, a biblioteca de planos do agente para a clonagem. Na segunda etapa ocorre o envio do clone, o protocolo envia o clone do agente para outro SMA e aguarda confirmação de

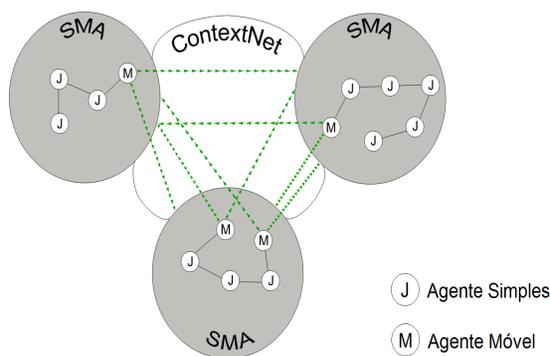


Figure 1: Transferência de agentes móveis.

chegada e se o SMA de destino é capaz de recebê-lo, pois o agente pode ter conhecimentos complexos demais para o SMA. Após a confirmação, inicia-se a etapa 3 onde há a destruição do agente de origem, o SMA de origem exclui o agente e todas as informações como forma de segurança.

Os agentes móveis poderão ser programados para utilizar diversos protocolos, que definem o propósito do agente móvel ao mover-se para outro SMA. Estes agentes têm como principal objetivo a auto preservação, como por exemplo, um agente situado em um *hardware* condenado e para preservar seus conhecimentos este realiza uma auto transmissão para outro SMA. Com essa transferência, o agente pode interagir de diversas formas com o sistema e os agentes que estiverem presentes nele, podendo exercer uma relação de mutualismo, de inquilinismo, ou até mesmo de predatismo visando uma invasão e conquista.

Quando um SMA utiliza do protocolo de transferência de agente com o intuito (Predatismo), detalhes como o *hardware* em questão devem ser levados em consideração, por exemplo, o agente conquistador ou predador tem como origem um SMA aplicado em um protótipo de carro e move-se para o protótipo de um avião, o SMA de origem não é capaz de controlar o novo *hardware*, como consequência, não atingirá o propósito de auto preservação. Uma maneira de solucionar essa questão é o SMA conquistador ficar vivendo de maneira colaborativa com o SMA conquistado ou presa (inquilinismo), e se depois de um tempo de convivência o SMA predador aprender a controlar o novo *hardware*, pode ou não remover o SMA conquistado, contudo, para que ocorra a transferência do agente predador, o SMA presa deve permitir a entrada deste agente. De certa forma, o SMA presa, ao permitir, entende que o conhecimento do SMA predador é mais significativo para a organização de SMA em um ambiente inteligente do que o próprio conhecimento.

4.2 Testes

O primeiro teste foi realizado para verificar se era possível, através do *middleware* ContextNet, mover um arquivo na extensão .asl (extensão de arquivo onde é armazenado o código fonte de um agente em Jason). Para isso, foram criados dois programas desenvolvidos em JAVA, um programa com a finalidade de enviar o arquivo e o outro de receber, para interligar os programas, utilizou-se o ContextNet.

Ao enviar o arquivo asl de um agente para outro SMA, significa que estará enviando somente seu código fonte de

criação, com as crenças, desejos e intenções iniciais, portanto todo o conhecimento adquirido durante a execução do agente não será enviado. Além disso, esses conhecimentos podem ser adquiridos por interações com outros agentes e/ou com ambiente e atualização da sua base de crenças. Portanto, é necessário construir um arquivo asl a partir da base de crenças, planos, desejos e intenções atuais do agente para poder transferi-lo, preservando seu conhecimento. Além do mais, ao reproduzir a biblioteca de planos torna-se possível preservar o comportamento e o processo para o raciocínio cognitivo do agente. O próximo teste será transferir um agente cognitivo entre SMAs e *hardwares* distintos, porém será utilizado *hardwares* similares, neste caso, protótipos de carros autônomos com a simples finalidade de detectar e evitar obstáculos. Os veículos são compostos de sensores e atuadores, interligados a um microcontrolador que, por sua vez, pode ser controlado por agentes ARGO situado em um SMA aplicado na Raspberry PI embarcada nos protótipos. O protótipo dos carros pode ser visto na Figura 2.

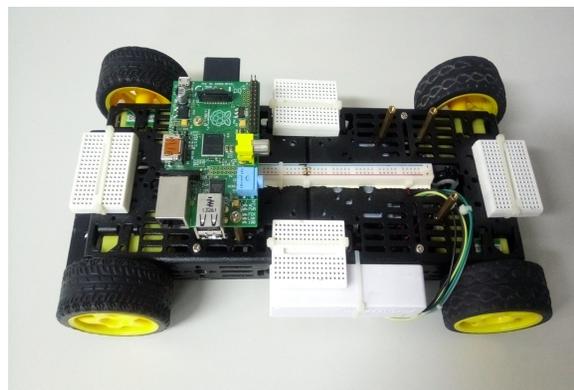


Figure 2: Protótipo dos carros autônomos

O teste relacionado aos protótipos não somente avalia o transporte de um agente móvel entre os diferentes protótipos e sistemas, mas também, o comportamento de um SMA situado em um *hardware* condenado. Através disso, observar o acionamento do protocolo de transferência de agentes com a relação predatismo, onde o agente móvel situado no *hardware* danificado irá montar um código de construção do SMA e de todos os agentes, para enviá-lo a outro SMA com o intuito de dominar o novo *hardware*. Após dominar o novo *hardware*, o agente predador constrói por completo o SMA predador e apaga qualquer rastro de seus conhecimentos no antigo *hardware*. Na Figura 3 pode ser visto o fluxo de transferência do agente móvel entre os protótipos em três etapas, o estado inicial, a transferência e o estado final.

4.3 Resultados e Discussões

Quando o protocolo de transferência de agentes é ativado com a relação predatória, o SMA de origem deve ou não ser excluído do *hardware* de origem? Se sim, antes ou depois de realizar a transferência para o novo *hardware*? Pois pode haver rejeição do SMA de destino ou até mesmo quebra na conexão entre os SMAs. Acreditamos que o SMA de origem deve ser excluído do *hardware* de origem, para preservação da privacidade e segurança, porém somente após realizar a transferência completa para o novo *hardware*.

Outra questão deve ser levada em consideração, quando

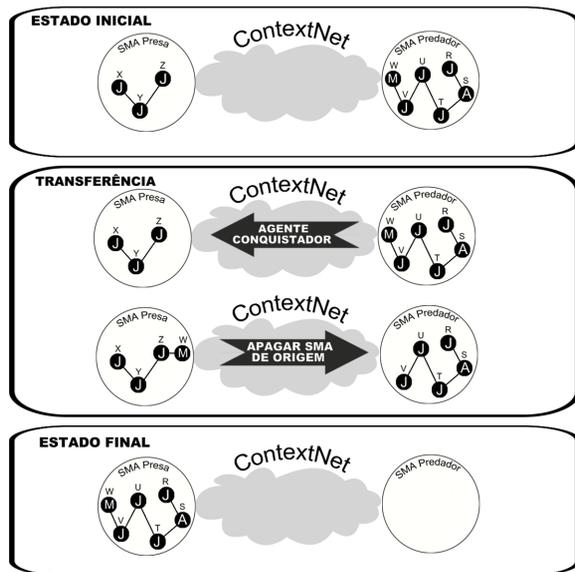


Figure 3: Movimentação de agentes nos protótipos

se trata do protocolo de transferência de agentes utilizando a relação de Predatismo, como, por exemplo, um SMA pode ser composto de diversos agentes e estes podem ser de diferentes tipos (ARGO, padrão, móvel, entre outros), porém somente o móvel pode se transferir para outro SMA, então como preservar a inteligência de todo o SMA de origem? Para responder esta pergunta chegou-se a conclusão de que o agente móvel, quando invadir o SMA a ser conquistado, deve ter em sua arquitetura a capacidade para construir não somente seu próprio estado mental no novo SMA, mas como também, dos outros agentes do SMA conquistador.

Na proposta dos protocolos para adquirir conhecimento, os pontos em relação à confiabilidade e segurança podem ser explorado de uma maneira diferenciada, por exemplo, um agente móvel chegando a um novo sistema e tentar transmitir conhecimento, será que os agentes tradicionais irão acreditar nas informações recebidas? Ou se os agentes tradicionais enviarem informações, o agente móvel deverá confiar e recebê-las? Se sim, será que após adquirir esses novos conhecimentos, o mesmo poderá voltar para o SMA de origem? A fim de sanar estas questões, concluiu-se que, a proposta dos protocolos de transferência de agentes ainda não deve entrar no mérito de métodos de filtragem de informações e gerência de grau de confiabilidade, porém a decisão de permanecer ou voltar para o SMA de origem será uma decisão tomada pelo próprio agente.

5. CONCLUSÃO

A proposta de desenvolvimento foi criar protocolos de transferência de agentes, que por sua vez, podem ser divididos de acordo com a relação que o agente móvel deva ter com o outro SMA. No atual estágio de desenvolvimento, pôde ser observado que o foco está sendo no protocolo de transferência de agentes com a relação de predatismo, visto que é o protocolo mais simples de ser implementado e não é necessário lidar com questões de negociação entre agentes de diferentes SMA. Além disso, tem o ideal de preservar os aspectos cognitivos da inteligência do SMA predador. Os

resultados iniciais são promissores tanto levando em consideração a implementação, mas como também, a interligação dos conceitos derivados da biologia com os conceitos de SMAs abertos, agentes móveis e ambientes abertos.

Para trabalhos futuros deseja-se transformar esse projeto dos protocolos de transferência de agentes em uma extensão para ser incorporado ao *framework* Jason, possibilitando a programação de agentes móveis. Pretende-se também, aplicar em um projeto de uma sala inteligente em escala real, que está sendo desenvolvido pelos autores com diversos SMA distintos e agentes móveis transitando entre eles.

6. REFERÊNCIAS

- [1] J. P. B. Andrade, M. Oliveira, E. J. T. Gonçalves, and M. E. F. Maia. Uma Abordagem com Sistemas Multiagentes para Controle Autônomo de Casas Inteligentes. In *XIII Encontro Nacional de Inteligência Artificial e Computacional (ENIAC)*, 2016.
- [2] R. H. Bordini, J. F. Hübner, and M. Wooldridge. *Programming Multi-Agent Systems in AgentSpeak using Jason*. John Wiley & Sons Ltd, 2007.
- [3] M. E. Bratman. *Intention, Plans and Practical Reasoning*. Cambridge Press, 1987.
- [4] M. S. Chebout, F. Mokhati, M. Badri, and M. C. Babahenini. Towards preventive control for open mas - an aspect-based approach. In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics - Volume 1: ICINCO*, pages 269–274. INSTICC, SciTePress, 2016.
- [5] L. David, R. Vasconcelos, L. Alves, R. André, G. Baptista, and M. Endler. A communication middleware for scalable real-time mobile collaboration. In *Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), 2012 IEEE 21st International Workshop on*, pages 54–59. IEEE, 2012.
- [6] M. Endler, G. Baptista, L. Silva, R. Vasconcelos, M. Malcher, V. Pantoja, V. Pinheiro, and J. Viterbo. 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, 2011.
- [7] S. Ossowski, V. Julián, J. Bajo, H. Billhardt, V. Botti, and J. Corchado. Open mas for real world applications: An abstract architecture proposal. In *Proc. XII Conference of the Spanish Association for Artificial Intelligence (CAEPIA)*, volume 2, pages 151–160, 2007.
- [8] C. E. Pantoja, M. F. Stabile Jr, N. M. Lazarin, and J. S. Sichman. ARGO: A customized jason architecture for programming embedded robotic agents. *Fourth International Workshop on Engineering Multi-Agent Systems (EMAS 2016)*, 2016.
- [9] V. Sanchez-Anguix, R. Aydogan, V. Julian, and C. M. Jonker. Analysis of intra-team strategies for teams negotiating against competitor, matchers, and conceders. In *The 5th International Workshop on Agent-based Complex Automated Negotiations (ACAN 2012)*, pages 1–8, 2012.
- [10] M. Tissot-Squalli. *Interações ecológicas & biodiversidade*. Unijui, 2009.
- [11] M. Wooldridge. *An Introduction to Multi-Agent Systems*. Wiley, 2009.

Modelagem do Subsistema de Administração Imobiliário Baseado em Linhas de Produto

Silvia Ribeiro Mantuani
UTFPR – PG
Av. Monteiro Lobato, s/n – Km 04
Ponta Grossa, Brasil
silviamantuani@gmail.com

Fernando Henrique Campos
UTFPR – SH
Rua Cerejeira, s/n – São Luiz
Santa Helena, Brasil
fhcampos@utfpr.edu.br

Vinícius Camargo Andrade
UTFPR – PG
Av. Monteiro Lobato, s/n – Km 04
Ponta Grossa, Brasil
vcandrade@utfpr.edu.br

RESUMO

O desenvolvimento baseado em linha de produto permite a identificação das similaridades e variabilidades entre determinados sistemas de maneira que se torne possível construir aplicações com uma arquitetura. Este trabalho propõe a modelagem de um subsistema de administração imobiliário baseado em Linha de Produto de Software (LPS). Neste contexto são apresentadas as vantagens e desvantagens de utilizar tal abordagem. Para modelar o sistema, realizou-se uma análise dos métodos baseados em LPS a fim de utilizar àquele que melhor se adequa ao contexto do presente trabalho. Os métodos analisados foram: *Product Line UML-Based Software Engineering*, *Family-Oriented Abstraction*, *Specification and Translation*, *Feature Oriented Domain Analysis* e *Delazeri e Wolf*, sendo este último o escolhido para a realização da modelagem. Com o modelo proposto, o desenvolvedor possui uma arquitetura flexível a qual pode ser reutilizada e/ou estendida para um novo sistema de mesmo domínio.

Palavras-chave

Linhas de Produto de Software; *UML Components*; Sistema Imobiliário.

ABSTRACT

The development based on product line allows the identification of similarities and variabilities among stated systems so that it becomes possible to build applications with an architecture. This paper proposes the modeling of a real estate management subsystem based on Software Product Line (SPL). In this context, are presented the advantages and disadvantages of the use this approach. To model the system, was performed an analysis of the methods based on SPL in order to use the most suitable to the context of the present paper. The methods analyzed were: *Product Line UML-Based Software Engineering*, *Family-Oriented Abstraction*, *Specification and Translation*, *Feature Oriented Domain Analysis* and *Delazeri e Wolf*, being the last one chosen for the accomplishment of the modeling. With the proposed model,

the developer has a flexible architecture that can be reused and/or extended to a new system of the same domain.

Keywords

Software Product Line; *UML Components*; Real Estate System.

1. INTRODUÇÃO

Para alcançar a excelência no desenvolvimento de seus produtos, ou seja, fornecer produtos de alta qualidade aliado a baixo custo em menor tempo de produção, empresas de software adotam abordagens baseadas em Linhas de Produto de Software (LPS), pois esta permite o desenvolvimento e manutenção de produtos de mesmo domínio, isto possibilita uma reutilização de artefatos, os quais atendem um número maior de clientes, o que, a longo prazo, significa um produto entregue em curto período e de maior confiabilidade devido aos inúmeros testes realizados no mesmo [1].

Este trabalho propõe modelar o subsistema de Administração do sistema imobiliário proposto por Canteri e Dvulatka [2]. Para isto, estendeu-se a análise do domínio dos sistemas Imobilis [3] e CIPRO [4] realizada na pesquisa de Canteri e Dvulatka [2] e realizou-se uma entrevista com o gerente de uma Imobiliária da cidade de Ponta Grossa - Paraná. Além disso, alguns métodos de desenvolvimento baseados em LPS foram analisados a fim de optar pelo qual mais se adequa ao escopo deste trabalho: *FAST (Family-Oriented Abstraction, Specification and Translation)* [5], *PLUS (Product Line UML-Based Software Engineering)* [6] e *FODA (Feature Oriented Domain Analysis)* [7] e *Delazeri e Wolf* [8].

Para a realização deste trabalho optou-se pela utilização do método de *Delazeri e Wolf* [8], pois este contempla características dos outros métodos analisados, de forma a produzir artefatos de entrada e saída para cada fase/subfase da Engenharia de Domínio e Aplicação.

2. LINHA DE PRODUTO DE SOFTWARE

Segundo Clements e Northrop [9], LPS é o desenvolvimento de famílias de produtos de software, que atendem um determinado segmento de mercado, considerando requisitos particulares de cada cliente de acordo com a análise de domínio, o reúso de código e características (*features*) comuns entre LPS de forma mais inteligente, reduzindo o tempo e o custo com o desenvolvimento.

Para determinar se os softwares estão ou não na mesma linha de produto, leva-se em consideração uma coleção de *features* [10]. Esta pode ser entendida como parte do sistema de suma importância para o cliente e que é utilizada para definir pontos comuns ou para diferenciar produtos em uma linha de produção.

Segundo Linden et al. [11], há três atividades principais da LPS e que juntas proporcionam práticas de negócios e tecnologia. São elas:

- Engenharia do Domínio: Consiste em coletar, organizar e armazenar experiências anteriores na construção de aplicações em um domínio específico na forma de artefatos empregáveis que possam ser utilizados na construção de novas aplicações [12].
- Engenharia da Aplicação: É atividade de construção de sistemas a partir de resultados obtidos por meio da engenharia de domínio, ou seja, a construção nesta atividade se dá por meio do *core*.
- Gerenciamento: abrange gestão técnica e organizacional da linha de produto [13].

A Figura 1 ilustra o relacionamento entre as três atividades, sendo cada atividade representada por um círculo. Nota-se a iteratividade entre as mesmas, a saída de uma atividade favorece a entrada da outra e assim sucessivamente.



Figura 1. Atividades essenciais no processo de linhas de produtos [9]

3. MÉTODOS DE DESENVOLVIMENTO BASEADO EM LPS

Há várias abordagens que podem ser utilizadas para o desenvolvimento baseado em LPS. Dentre estes, destacam-se: FAST [5], PLUS [6], FODA [7] e Delazeri e Wolf [8].

Após o estudo das abordagens, realizou-se uma análise qualitativa entre elas considerando os seguintes critérios:

- baseados em features: tipo de abordagem do método.
- baseado em família: tipo de abordagem do método.
- fases: são as divisões e subdivisões de cada método, que são produzidos os artefatos de entrada e saída.
- iterativo: processo de desenvolvimento cíclico.

A Tabela 1 detalha as abordagens citadas e suas respectivas características. As linhas contendo o “X” indicam a presença da característica em um determinado método, caso contrário, este método não contempla tal particularidade.

Observou-se que o método que Delazeri e Wolf possui as mesmas características dos métodos FODA, PLUS, FAST, pois o mesmo é uma adaptação dos três citados. Isto resulta em uma obtenção das melhores qualidades de cada um e ainda um acréscimo de novos artefatos, proporcionando que este método resulte em fases, documentações e diagramas distintos, sendo este adotado para a realização deste trabalho.

Tabela 1. Características dos Métodos baseados em LPS

Métodos	Características			
	Baseado em <i>feature</i>	Baseado em família	Possui fases e subfases	Iterativo
FAST [5]		X	X	
PLUS [6]	X		X	X
FODA [7]	X		X	
Delazeri e Wolf [8]	X	X	X	X

4. MODELAGEM

Esta seção descreve o uso do método de Delazeri e Wolf [8] para a identificação das similaridades e variabilidades dos subsistemas contemplados em um Sistema Imobiliário, estes identificados por Canteri e Dvulatka [2]. A Figura 2 ilustra o modelo de contexto obtido pela identificação dos subsistemas.

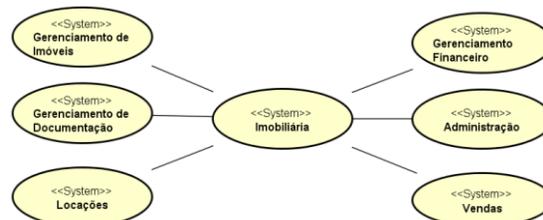


Figura 2. Diagrama de Modelo de Contexto [2]

Ao todo, identificou-se seis subsistemas que fazem parte de um sistema imobiliário. São eles: Gerenciamento de Imóveis, Gerenciamento Financeiro, Gerenciamento de Documentação, Administração, Locações e Vendas. Dentre os subsistemas identificados, optou-se por demonstrar a modelagem do subsistema de Administração. Demais subsistemas não serão abordados neste trabalho.

4.1 Requisitos e Modelagem do Domínio

Considerando a análise de domínio definiram-se os seguintes requisitos comuns para o subsistema Administração: Manter Clientes, Detalhar Descrição, Validar Dados, Armazenar e Manter Pessoas. Da mesma maneira, identificou-se os requisitos específicos, são eles: Terceiros, Construtoras, Funcionários e Verificar Contratos.

A Figura 3 ilustra o diagrama de caso de uso com as similaridades do subsistema de Administração.

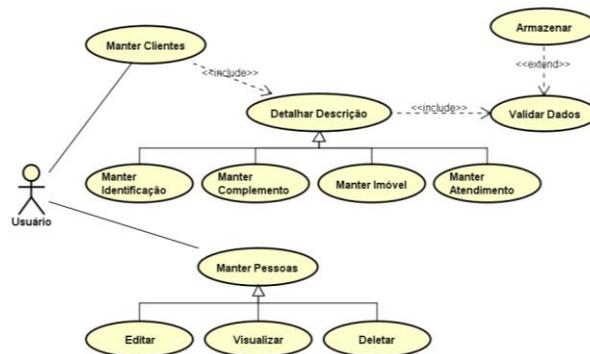


Figura 3. Diagrama de Caso de Uso com as similaridades do Subsistema de Administração

4.2 Modelagem da Arquitetura

Nesta subfase cada Caso de Uso já analisado no domínio torna-se uma interface da camada do sistema, tal como especifica o modelo UML *Components*. A Figura 4 ilustra as interfaces geradas com seus respectivos métodos para o Subsistema de Administração.

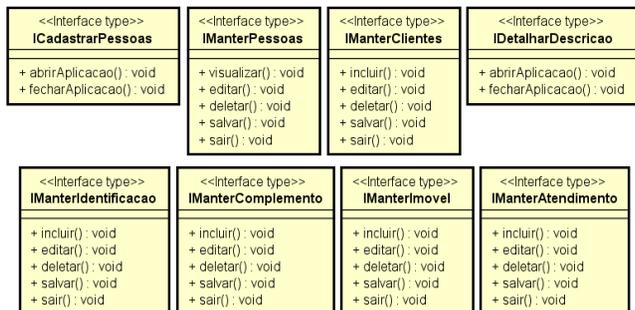


Figura 4. Interfaces com os Métodos do Software de Administração

4.3 Identificação dos Componentes

Com a identificação dos componentes que implementam as interfaces especificadas nas subfases anteriores, e de acordo com o processo UML *Components*, é associado um novo componente a cada interface, tanto de sistema quanto de negócio.

Assim, para a modelagem da camada Sistema, utilizam-se os Casos de Uso com as similaridades dos subsistemas. Neste trabalho foi apresentado apenas o Caso de Uso com as similaridades do subsistema de Administração (Figura 3) por questão de limitação de páginas. Com estes, identificou-se as seguintes interfaces: ICadastrarContas, IDetalharDescricao, ICadastrarImoveis, ICadastrarPessoas e IManterEndereco.

Considerando o conceito de linha de produto, ISoftwareImoveis controla as operações contidas no sistema, desta maneira, esta interface deve estar associada a todas as outras interfaces que fazem operações dentro do sistema.

Assim com as interfaces identificadas, com componentes e seus respectivos estereótipos relacionados com os casos de uso incluídos em cada componente é possível definir a arquitetura do Software de Imóveis com as camadas de Sistema e Negócio combinadas. Esta ilustrada na Figura 5.

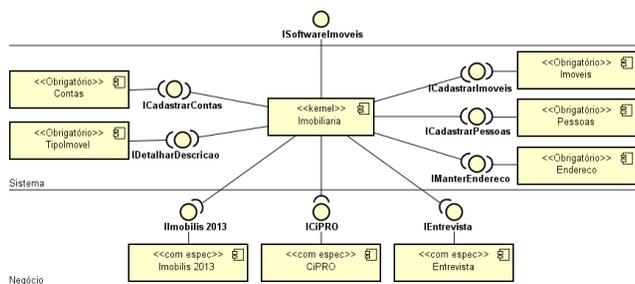


Figura 5. Arquitetura do Software de Imóveis

5. ENGENHARIA DA APLICAÇÃO

A fase de Engenharia de Aplicação, proposta por Delazeri e Wolf [8], é responsável pela identificação e modelagem dos requisitos da aplicação em diagrama de caso de uso e criação o

diagrama de *feature* variáveis inclusas no domínio. Além disso, é nesta fase que ocorre a implementação, validação com testes, funcionais e de integração, e a análise da integridade dos requisitos para, só então, ocorrer a entrega do produto ao cliente bem como seu posterior suporte.

5.1 Requisitos de Aplicação

Ao realizar análises no Subsistema de Administração de acordo com o Sistema Imobilis [3], foram encontradas as seguintes variedades:

- Funcionários: cadastro das pessoas que trabalham na imobiliária.
- Verificar contratos: possibilidade do usuário modificar o contrato de acordo com o estabelecido entre empresa e cliente.

Assim com a análise detalhada de cada requisito, construiu-se os Diagramas de *Features* que exibem as variabilidades dos sistemas analisados. Neste trabalho apenas o Diagrama de *Features* do subsistema de Administração é apresentado (Figura 6).



Figura 6. Variabilidades Subsistema de Administração

De acordo com os resultados obtidos a partir da análise das variabilidades dos subsistemas é possível ilustrá-los com Diagrama de Casos de Uso. A Figura 7 exibe as variabilidades do subsistema de Administração. O caso de uso “funcionários”, possui a funcionalidade de cadastrar as pessoas que trabalham na imobiliária, enquanto “verificar contratos” é responsável por possibilitar ao usuário modificar o contrato de acordo com o que ficou estabelecido entre a empresa e cliente.

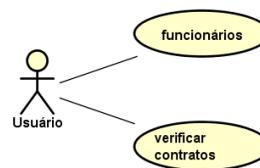


Figura 7. Caso de Uso das variabilidades do Subsistema de Administração

Por fim, com as interfaces definidas na fase de Projeto de Domínio, que contém os atributos de domínio e os de variabilidade, é possível modelar o produto. As classes com os estereótipos *Application* pertencem a fase de aplicação (variabilidades), *type* identificam as classes definidas na fase de domínio (similaridades) e *core* é o Kernel (núcleo) do sistema. A Figura 8 ilustra o diagrama de classe para a instanciação da aplicação dos subsistemas de Gerenciamento de Documentação, Gerenciamento Financeiro e Administração para o Sistema de Imobilis [3].

A implementação, teste e entrega da aplicação Sistema de Imóveis é realizada por meio da instanciação das interfaces comuns entre eles, incluídas de suas classes específicas.

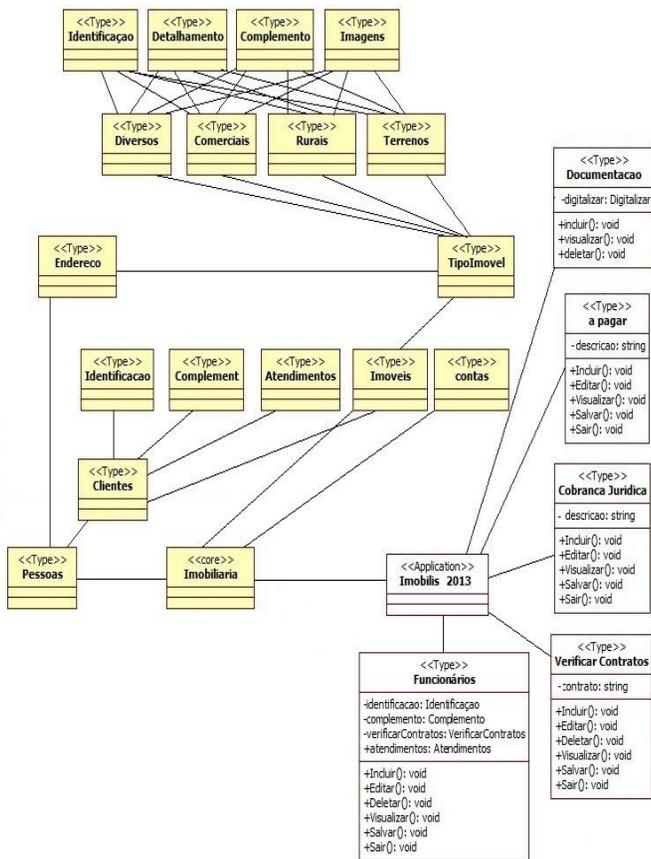


Figura 8. Modelo de Classe para Instanciação do Sistema de Imóveis

6. CONCLUSÕES

Este trabalho apresentou a modelagem do subsistema de Administração de um sistema imobiliário baseada em Linhas de Produto de Software.

Para a escolha da metodologia utilizada no desenvolvimento, realizou-se uma análise nos métodos FAST [5], PLUS [6], FODA [7] e Delazeri e Wolf [8], sendo este último o selecionado.

O desenvolvimento se deu em duas fases, são elas: Engenharia de Domínio e Engenharia de Aplicação. Na primeira foram realizadas análises de requisitos do domínio com o intuito de definir a relação de similaridades e variedades entre os sistemas. Ainda nesta fase, modelou-se o domínio e desenvolveu-se a arquitetura do domínio dos produtos estudados. Na segunda fase analisaram-se os requisitos variáveis. A partir dos mesmos criaram-se os Diagramas de Casos de Uso para definir o Diagrama de Classes como forma de instanciação de cada produto analisado.

Por meio deste trabalho pode-se concluir que o desenvolvimento baseado em linhas de produto de software possui vantagens tanto para os desenvolvedores, que terão um produto de forma rápida e

com maior qualidade, quanto para os clientes, que desfrutarão de um software desenvolvido de acordo com suas necessidades.

7. AGRADECIMENTOS

À Universidade Tecnológica Federal do Paraná, campus Ponta Grossa e campus Santa Helena, representado por suas respectivas diretorias, pelo apoio ao trabalho.

8. REFERÊNCIAS

- [1] A. E. de Campos Lobo and C. M. F. Rubira. Um estudo para implantação de linha de produto de software baseada em componentes.
- [2] C. Canteri and S. D. Dvulotka. Modelagem e implementação de um sistema de venda e de locação de imóveis baseado em linhas de produto. B.S. thesis, Universidade Tecnológica Federal do Paraná, 2013.
- [3] Imobilis. 2017. *Imobilis 2013*. Retrieved August 7, 2017. DOI=<http://www.pjsoft.com.br>.
- [4] Ci-Pro. 2017. *CI-PRO - Consultoria Imobiliária Profissional*. Retrieved August 7, 2017. DOI=<http://www.baixaki.com.br/site/dwnld13464.htm>.
- [5] M. Harsu. *A survey on domain engineering*. Tampere University of Technology, 2002.
- [6] H. Gomaa. *Designing software product lines with UML*. IEEE, 2005.
- [7] L. Northrop, P. Clements, F. Bachmann, J. Bergey, G. Chastek, S. Cohen, P. Donohoe, L. Jones, R. Krut, R. Little, et al. A framework for software product line practice, version 5.0. *SEI*. {2007}{<http://www.sei.cmu.edu/productlines/index.html>, 2007.
- [8] B. R. Delazeri and E. C. Wolf. Modelagem de um sistema organizador baseado em linhas de produto. B.S. thesis, Universidade Tecnológica Federal do Paraná, 2012.
- [9] P. Clements and L. Northrop. *Software product lines*. Addison-Wesley, 2002.
- [10] D. R. Nunes. C. e Reis, R. (2010). "Educação em engenharia de software". *A carreira do pesquisador em Engenharia de Software: princípios, conceitos e direções. 1ª edição*. Salvador, Brasil: UFBA, cap, 5:132{181.
- [11] F. Linden. vd, Schmid, K., and Rommes, E. (2007). Software product lines in action: The best industrial practice in product line engineering. *Springer-Verlag New York, Inc., Secaucus, NJ, USA*, 10:11{13.
- [12] A. P. da Silva, E. de Barros Costa, and I. I. Bittencourt. Uma linha de produto de software baseada na web semântica para sistemas tutores inteligentes. 2012.
- [13] D. Ferreira Santana Neiva. Riple-re: A requirements engineering process for software product lines. 2009.

Participação Feminina no Cinema Brasileiro: Análise Através de Redes Sociais

Renato Silva de Melo
Universidade Federal do
Paraná
Centro Politécnico
Curitiba, Brasil
rsmelo@inf.ufpr.br

Andre Luis Vignatti
Universidade Federal do
Paraná
Centro Politécnico
Curitiba, Brasil
vignatti@inf.ufpr.br

Arthur Pecheba da Costa
Universidade Federal do
Paraná
Centro Politécnico
Curitiba, Brasil
apc10@inf.ufpr.br

Felipe do Nascimento
Universidade Federal do
Paraná
Centro Politécnico
Curitiba, Brasil
fn10@inf.ufpr.br

RESUMO

Através de dados disponíveis na internet, este trabalho constrói e analisa redes sociais baseadas nas relações entre atrizes e atores de filmes brasileiros. O estudo investiga empiricamente propriedades estatísticas de tais redes. A análise usa teoria dos grafos para mostrar diferenças quanto à presença de atores e atrizes nos filmes brasileiros ao longo das últimas cinco décadas. Verificamos que as diferenças entre homens e mulheres são evidentes, onde as atrizes aparecem sempre em menor quantidade. Isso foi observado e comparado com os números do cinema norueguês, que é referência internacional em igualdade de gênero. Com isso, constatamos que o cinema brasileiro é equiparável ao cinema norueguês considerando a igualdade de gênero.

ABSTRACT

Through data available on the Internet, this work builds and analyzes social networks based on the relationship between actresses and actors of Brazilian movies. The study empirically investigates statistical properties of such networks. The analysis uses graph theory to show differences about the presence of actors and actresses in Brazilian movies along the last five decades. We found that the differences between men and women are evident, where the actresses always appear in smaller amount. This has been observed and compared with the numbers of Norwegian cinema, which is international reference on gender equality. Thereby, we verified that the Brazilian cinema is comparable to the Norwegian cinema regarding gender equality.

Palavras-chave

Gêneros, Redes Sociais, Cinema brasileiro.

1. INTRODUÇÃO

Anualmente o *Fórum Econômico Mundial* (FEM)¹ publica uma série de relatórios, dentre os quais está o *Índice Global de Desigualdade de Gênero* [1]. Neste documento os países são classificados de acordo com as diferenças entre homens e mulheres. Para estabelecer a classificação o FEM considera os contrastes entre homem e mulher em campos como economia, política, educação e saúde. O último relatório, publicado em 2016 [2], coloca o Brasil na 85ª posição dentre 145 países participantes. Este trabalho usa dados da internet para verificar, numericamente, diferenças entre a participação da mulher e do homem no cinema brasileiro. Através de grafos construídos com dados obtidos do site *Internet Movie Database* (IMDB)², utilizamos métricas de redes sociais para entender como as atrizes se relacionam entre si e se estão presentes em filmes com a mesma assiduidade dos atores. Para fins de comparação, confrontamos os resultados da análise dos filmes brasileiros com os do cinema norueguês. A Noruega foi escolhida como referência por ser o segundo país melhor qualificado, uma vez que a Islândia ocupa a 1ª posição, mas a quantidade de filmes produzidos é pouco representativa para questões de comparação.

Os resultados obtidos apresentam baixa participação feminina, sempre por volta de 1/3 do total de pessoas nos anos de 1965-1995. Algumas das variações sofridas ao longo dos anos estão associadas a eventos ocorridos na história política brasileira. Por exemplo, a crise econômica de 1982 e o confisco de reservas financeiras de órgãos que financiavam a produção de filmes [4]. Efeitos positivos também foram percebidos ao comparar o cinema norueguês com o brasileiro. As comparações mostram que diferenças sociais entre os países não se refletem na indústria cinematográfica, no que se refere à igualdade de gênero. Considerando a quan-

WPCCG '17 Outubro, 2017, Ponta Grossa, Paraná, Brasil

¹www.weforum.org

²www.imdb.com

tidade de pessoas envolvidas, os dados dos dois países são proporcionalmente parecidos.

O artigo está organizado em quatro seções. A Seção 2 descreve como os dados são coletados e como são tratados para a geração dos grafos. Na sequência, explicamos as métricas utilizadas para fazer a comparação entre os gêneros. A Seção 3, apresenta os resultados a partir das métricas utilizadas, assim como a interpretação destes resultados, com destaque à atuação das mulheres. A Seção 4 conclui o trabalho com uma discussão sobre os dados obtidos.

2. ANÁLISE DOS DADOS

Dentre as informações disponibilizadas pelo IMDB, estão uma listagem de todos os atores e atrizes, filmes e suas nacionalidades, diretores, biografias, sinopses e citações importantes de filmes mais conhecidos. O IMDB foi escolhido como fonte de dados por conter uma quantidade considerável de dados abertos sobre filmes a nível internacional.

2.1 Modelagem da Rede Social

Na extração de dados dos arquivos do IMDB foi necessário elaborar um conjunto de restrições para filtrar o conteúdo. Descrevemos a seguir as informações descartadas da base de dados. São considerados apenas os filmes, descartando seriados e programas de auditório. Mantivemos o foco apenas em atrizes e atores, excluindo dubladores, diretores, roteiristas e etc. Foram descartados atores e atrizes que não foram creditados oficialmente ou que possuem apenas um filme. A fim de coletar informações dos últimos cinquenta anos a respeito do cinema brasileiro e norueguês, o tratamento dos dados foi feito considerando períodos de tempo (em anos) e nacionalidade dos filmes. Ferramentas foram desenvolvidas para automatizar o processo de leitura da base de dados.

Definimos um grafo G como um par ordenado $(V(G), E(G))$ que consiste de um conjunto $V(G)$ de vértices e um conjunto $E(G) \subseteq V(G) \times V(G)$ de arestas [3]. Dessa forma, os grafos foram modelados de modo que os vértices são atores e atrizes e as arestas são os filmes, ou seja, se duas pessoas trabalharam juntas em um filme é criada uma aresta entre elas. Por simplicidade, arestas paralelas são ignoradas. Como as redes resultantes relacionam pessoas, temos então grafos de redes sociais. A Figura 1 ilustra os grafos obtidos para a década de 1965 a 1974. Nas Figuras 1(a) e 1(b) estão os grafos de atores e atrizes na Noruega. O grafo de mulheres (b) além de possuir menos vértices que o de homens (a), também possui menos arestas. Nas Figuras 1(c) e 1(d) temos os grafos de atores e atrizes no Brasil, assim como no caso da Noruega, o grafo de homens é maior que o de mulheres. Os grafos resultantes das décadas mais recentes não são ilustrados mas foram considerados no estudo.

2.2 Metodologia de Comparação

Para quantificar a participação das mulheres nos filmes brasileiros nas últimas cinco décadas, analisamos se houve alguma evolução percentual no número de atores/atrizes e relacionamos os resultados ao contexto histórico. Uma vez que o tamanho dos grafos obtidos pode dificultar a descrição dessas redes, definimos algumas medidas quantitativas para descrever propriedades e comparar redes diferentes. As medidas escolhidas são: *grau médio*, comprimento do *caminho médio*, *diâmetro* e *densidade*. Para entendimento dos critérios, seguem algumas definições básicas, baseadas no livro de Murty e Bondy [3].

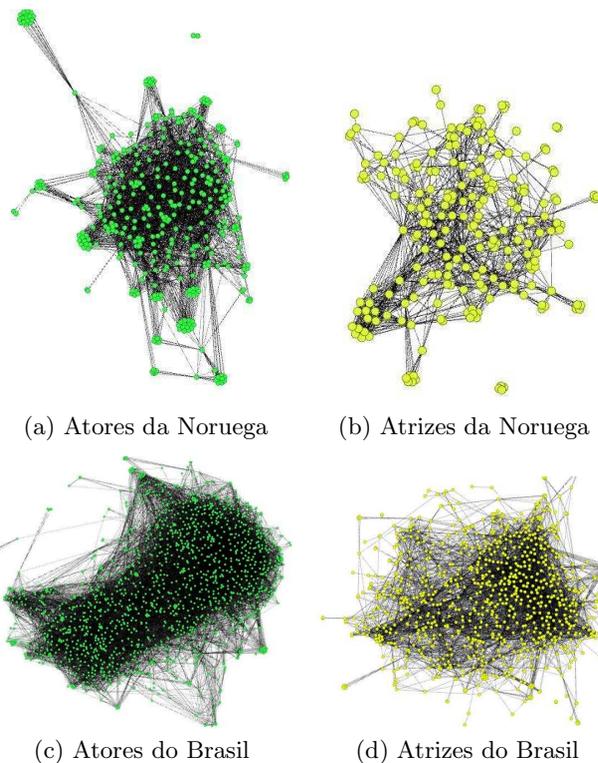


Figura 1: Grafos obtidos a partir dos arquivos do IMDB para a década de 1965-1974.

DEFINIÇÃO 1 (GRAU MÉDIO). Denotamos como $\delta(v)$ o grau de um vértice v em um grafo G , onde $\delta(v)$ é o número de arestas de G que são adjacentes a v . Assim, dizemos que o número $\bar{\delta}(G)$ é o grau médio de G , isto é,

$$\bar{\delta}(G) = \frac{1}{|V(G)|} \sum_{v \in V(G)} \delta(v).$$

DEFINIÇÃO 2 (CAMINHO). Um caminho em um grafo G é uma sequência $P := (v_0, v_1, \dots, v_l)$ de vértices em que os elementos consecutivos são vértices ligados por uma aresta, onde os vértices e arestas não se repetem. O comprimento l de um caminho é o número de arestas.

DEFINIÇÃO 3 (DISTÂNCIA). A distância em um grafo G é o tamanho do menor caminho entre dois vértices u e v de G , denotada por $d(u, v)$. Se não existe caminho entre u e v , então $d(u, v) = \infty$.

DEFINIÇÃO 4 (CAMINHO MÉDIO). O caminho médio de um grafo G é a média das distâncias entre todos os pares de vértices de G .

DEFINIÇÃO 5 (DIÂMETRO). O diâmetro de um grafo G é a maior distância entre dois vértices de G .

DEFINIÇÃO 6 (DENSIDADE). A densidade de um grafo G é a razão entre o número de arestas de G e o número máximo de arestas que G pode ter, isto é, $|E(G)| / \binom{|V(G)|}{2}$.

Como não podemos medir a relevância dos papéis interpretados por mulheres através dos dados coletados, usamos as métricas definidas anteriormente para mensurar a quantidade de interações que as atrizes possuem entre si. Consideramos interação entre indivíduos, neste estudo, quando

Tabela 1: Número de atores e atrizes nas últimas cinco décadas em cada país.

Década	Brasil		Noruega	
	Atrizes	Atores	Atrizes	Atores
65-74	926 (35,4%)	1685	219 (35,3%)	402
75-84	1294 (38,5%)	2065	223 (33%)	451
85-94	792 (39,9%)	1189	452 (33%)	821
95-04	1315 (37,4%)	2196	843 (34,2%)	1617
05-14	5571 (38,5%)	8862	3686 (36,6%)	6360

os membros participaram de algum filme em comum. O grau médio de cada rede é um dos indicadores considerados porque indica quanto os indivíduos interagem uns com os outros. Neste sentido, podemos saber se há semelhança ou não em redes sociais de atores e atrizes. Além disso, dois membros que não possuem filmes em comum, podem estar conectados indiretamente através de um caminho. Dessa forma, consideramos o conceito de caminhos em grafos para capturar interações indiretas entre os membros de uma rede. Assim, a distância determina quantos filmes dois indivíduos estão longes um do outro. Adicionalmente, a densidade é um valor entre 0 e 1 que determina o quão esparsos são os grafos resultantes. Quanto mais interações (arestas), mais próximo de 1 este número tende a estar. As demais métricas reforçam essas informações.

3. RESULTADOS

Os dados que seguem são referentes aos anos de 1965-2014. A Tabela 1 apresenta o total de atrizes e atores que atuaram em filmes no Brasil e na Noruega separados por década. Observe que em todos os intervalos de tempo, o número de mulheres é visivelmente menor do que o de homens em ambos os países. Mas, podemos notar que houve aumento de 3,1% na presença de mulheres nos filmes brasileiros e de 1,3% no cinema norueguês. Além dessa informação, fazemos uma avaliação relativa considerando grau médio, diâmetro, densidade e caminho médio das redes. No total, 20 grafos foram gerados, dois para cada década (um para os atores e outro para as atrizes), separados por país. As próximas subseções detalham as observações feitas para cada década.

3.1 Década de 1965-1974

O grau médio do grafo de atrizes na Noruega para esta década é aproximadamente a metade do grau médio dos atores (Tabela 2). Isso significa que, em média, o número de homens é mais que o dobro em cada filme norueguês neste período. Em dados absolutos, cada atriz trabalhou, em média, com outras 12 mulheres em 10 anos, enquanto os atores trabalharam com outros 26 homens, em média. Os números são proporcionalmente semelhantes aos do Brasil, onde o grau médio de atores também é o dobro do grau médio de atrizes. No que tange o diâmetro do grafo, também existe semelhança, com diferença de apenas um entre os países (Tabela 2). O caminho médio, indicador da quantidade de filmes que estão entre dois indivíduos quaisquer, mostra que entre 1965 e 1974, a situação foi mais favorável para o Brasil em termos de igualdade de gênero no cinema.

3.2 Década de 1975-1984

Tabela 2: Métricas para os grafos de atrizes e atores do Brasil e da Noruega entre 1965 e 1974.

	Brasil		Noruega	
	Atrizes	Atores	Atrizes	Atores
GRAU MÉDIO	16,475	32,275	11,991	25,733
DIÂMETRO	8	7	6	5
DENSIDADE	0,018	0,019	0,055	0,064
CAMINHO MÉDIO	3,023	2,787	2,725	2,395

Nesta década, o Brasil apresentou evolução visível na participação feminina no cinema (veja a Figura 3). O grau médio de atrizes não era mais a metade do valor dos atores, como foi observado na década anterior, passando para mais de 60% do valor para o grafo masculino. Esse fenômeno se deve, principalmente, ao início de um projeto intitulado “*Cinema Novo*” em 1964, que ajudou a impulsionar o desenvolvimento do cinema brasileiro [4]. Após a criação da estatal Embrafilme, em 1969, que tinha o objetivo de financiar a produção de filmes nacionais, ocorreu uma alta considerável no número de atrizes e atores entre os anos 1975 e 1982, comparado a década anterior. Como este fenômeno não aconteceu na Noruega, isso explica o aumento significativo no grau médio dos grafos do Brasil ao passo que a Noruega teve uma leve queda. Os dados da década são resumidos na Tabela 3.

Tabela 3: Métricas para os grafos de atrizes e atores do Brasil e da Noruega entre 1975 e 1984.

	Brasil		Noruega	
	Atrizes	Atores	Atrizes	Atores
GRAU MÉDIO	25,805	42,297	11,368	25,06
DIÂMETRO	8	8	7	6
DENSIDADE	0,02	0,02	0,049	0,055
CAMINHO MÉDIO	2,899	2,763	3,015	2,61

3.3 Década de 1985-1994

Em contraste com os resultados reportados sobre a década de 1975-1984, a partir de 1982, o Brasil entra em recessão econômica e o cinema brasileiro tem uma forte crise [4]. Se na década anterior tivemos evolução na participação feminina, agora os resultados retrocedem a uma proporção similar à de 1965-1974. No grafo, as atrizes brasileiras possuem grau médio menor que a metade do grau médio dos atores, descendo de 60% na década anterior para 49%. Vale notar que mesmo com a crise brasileira, o grafo de atrizes norueguesas ainda apresenta um grau médio relativo pior (45%) frente ao grau médio dos atores. A Noruega quase dobrou o número de atores e atrizes nesse período (vide Tabela 1) e também teve variações significativas nos quatro fatores, mas a relação entre atrizes e atores manteve-se muito parecida com a das últimas duas décadas. Os dados da década são resumidos na Tabela 4.

3.4 Década de 1995-2004

Conforme a Tabela 5, esta década foi um marco no cinema da Noruega. Assim como na década anterior, o número total de atores e atrizes também dobrou (vide Tabela 1). A quantidade de mulheres ainda é 33%, porém os indicativos apresentam uma participação mais efetiva das mulheres no cinema norueguês. No Brasil, a retomada do cinema brasi-

Tabela 4: Métricas para os grafos de atrizes e atores do Brasil e da Noruega entre 1985 e 1994.

	Brasil		Noruega	
	Atrizes	Atores	Atrizes	Atores
GRAU MÉDIO	10,34	21,007	10,521	23,102
DIÂMETRO	11	8	7	8
DENSIDADE	0,013	0,018	0,023	0,028
CAMINHO MÉDIO	3,973	3,054	3,666	3,145

leiro impulsionou novas produções [4] e o crescimento também foi significativo, porém os indicativos de grau médio pioraram. Cada atriz trabalhou com outras 10 atrizes, em média, sendo 43% do grau médio dos atores, isto é, cada ator trabalhou com outros 25 em média. Pela primeira vez, a Noruega apresenta valores melhores que os do Brasil. Os dados da década são resumidos na Tabela 5.

Tabela 5: Métricas para os grafos de atrizes e atores do Brasil e da Noruega entre 1995 e 2004.

	Brasil		Noruega	
	Atrizes	Atores	Atrizes	Atores
GRAU MÉDIO	10,919	25,319	11,031	21,912
DIÂMETRO	11	10	9	10
DENSIDADE	0,008	0,011	0,013	0,013
CAMINHO MÉDIO	4,002	3,473	4,339	3,652

3.5 Década de 2005-2014

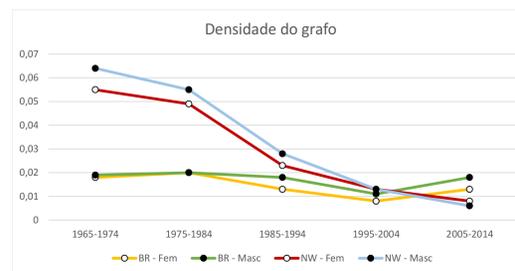
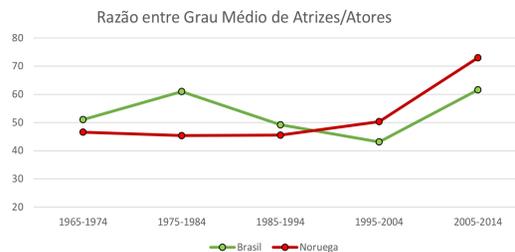
Neste período o número de atores e atrizes de cada país aumenta em grandes proporções. O Brasil quase quadruplicou e a Noruega quase quintuplicou. Revertendo a tendência das últimas duas décadas, a relação entre o grau médio dos grafos de atrizes e atores brasileiros se aproximou bastante e atingiu sua melhor marca (61%). Os diâmetros se igualaram e os caminhos médios se aproximaram bastante, o que indica melhor aproveitamento das atrizes no cinema brasileiro. Os números da Noruega são ainda melhores, a relação entre grau médio das atrizes e dos atores é de 79%. A proporção de atrizes é maior no Brasil do que na Noruega (38,5% e 36,6%, respectivamente) porém a densidade do grafo de atrizes norueguesas é maior do que o de atores, ou seja, as atrizes estão em menor número, mas se mostram mais presentes nos filmes e em maior quantidade em cada filme, de acordo com o aumento do grau médio. Os dados da década são resumidos na Tabela 6.

Tabela 6: Métricas para os grafos de atrizes e atores do Brasil e da Noruega entre 2005 e 2014.

	Brasil		Noruega	
	Atrizes	Atores	Atrizes	Atores
GRAU MÉDIO	20,641	33,503	28,433	38,962
DIÂMETRO	14	14	16	11
DENSIDADE	0,013	0,018	0,008	0,006
CAMINHO MÉDIO	5,531	4,883	5,797	4,587

3.6 Evolução ao longo das décadas

As Figuras 2 e 3 resumem a diferença entre homens e mulheres das cinco décadas, nos aspectos considerados. O valor ideal da Figura 2 é o grafo de atrizes mais denso do que

**Figura 2: Evolução da densidade dos grafos.****Figura 3: Grau médio de atrizes sobre o grau médio de atores (em %).**

o grafo de atores. Isso significa que, mesmo ainda tendo um número de atrizes abaixo do ideal, as mulheres participam de muitos filmes juntas, equilibrando a proporção de atores e atrizes em cada um dos filmes. O valor ideal na Figura 3 é próximo de 100%, significa que o grau médio de atrizes e atores é o mesmo, isto é, mulheres trabalham entre si na mesma quantidade que atores trabalham entre si.

4. CONCLUSÃO

Os resultados deste estudo mostram que a situação tem melhorado nos últimos 20 anos, no que se refere a diminuição da desigualdade de gênero. De 1965 até 1994 as mudanças estiveram bastante relacionadas à economia e pouca atenção tenha se dado a esse assunto, mas a partir de 1995 (na Noruega) e de 2005 (no Brasil) a tendência tornou-se positiva no sentido de encontrar um caminho de igualdade na participação de homens e mulheres no cinema. Os dados ainda apontam para maior participação masculina, que nunca são menores do que 60%. Mas, aparentemente, o cinema brasileiro ruma para a direção correta. Assim sendo, com estes resultados, esperamos trazer mais precisão às discussões de gênero, especialmente no cinema.

5. REFERÊNCIAS

- [1] R. Hausmann. The global gender gap report 2015. World Economic Forum, 2015.
- [2] R. Hausmann. The global gender gap report 2016. World Economic Forum, 2016.
- [3] U. Murty and A. Bondy. Graph theory (graduate texts in mathematics 244), 2008.
- [4] I. Xavier. *O cinema brasileiro moderno*. Editora Paz e Terra, 2001.

Obtenção do mapa de disparidade em imagens capturadas através de software próprio

William Takeshi Omoto
Universidade Tecnológica
Federal do Paraná
Av. Monteiro Lobato, Km 04
Ponta Grossa, Paraná - Brasil
williamomoto@gmail.com

Cauê Felchar
Universidade Tecnológica
Federal do Paraná
Av. Monteiro Lobato, Km 04
Ponta Grossa, Paraná - Brasil
caue.fcr@gmail.com

Erikson F. Morais
Universidade Tecnológica
Federal do Paraná
Av. Monteiro Lobato, Km 04
Ponta Grossa, Paraná - Brasil
emorais@utfpr.edu.br

ABSTRACT

This paper describes the importance of obtaining the depth of images in the area of computer vision. From images captured through a stereo pair, calculations can be performed to estimate the depth of objects in the image. For this, an essential step is the estimation of disparity, which is the change of position of objects, given a pair of images captured from different perspectives. However, for disparity to be estimated, two basic problems must be solved. The first one is called correspondence, which is the process of finding the corresponding pixels of the first image in the second one. The second problem, known as Reconstruction, is the depth perception obtained by computing the position difference of corresponding objects in the two images. A solution to the first problem is presented in this developing study, through the window approach.

RESUMO

Este artigo descreve a importância de obter a profundidade de imagens na área de visão computacional. A partir de imagens capturadas através de um par estéreo, cálculos podem ser realizados para estimar a profundidade de objetos na imagem. Para tal, um passo essencial é a estimação da disparidade, que é a mudança de posição de objetos, dado um par de imagens capturadas de perspectivas diferentes. No entanto, para a disparidade ser estimada, dois problemas básicos devem ser resolvidos. O primeiro é com relação a correspondência, que é o processo de encontrar os pixels correspondentes da primeira imagem, na segunda. O segundo problema, conhecido como Reconstrução, é a percepção de profundidade obtida ao computar a diferença de posição dos objetos correspondentes nas duas imagens. Uma solução para o primeiro problema é apresentada neste estudo em desenvolvimento, através da abordagem por janela.

Palavras-chave

Mapa de disparidade; Par Estéreo; Calibração de Câmeras.

1. INTRODUÇÃO

Sistemas de visão computacional são desenvolvidos para obter dados através de imagens. Um dos dados mais importantes para áreas como a robótica, realidade virtual e realidade aumentada, são dados de profundidade [1, 5].

Apesar de uma imagem conter muita informação, não é possível inferir dados sobre a profundidade de objetos somente através dela [2]. A profundidade de um objeto pode ser estimada

quando se possui duas ou mais imagens de uma mesma cena, porém com perspectivas diferentes. Através da mudança de posição do objeto entre as imagens, é possível estimar a distância do objeto até a câmera, utilizando cálculos de triangulação. Grande parte dos animais possuem ao menos um par de olhos para estimar distâncias, e assim conseguir se deslocar, evitando obstáculos [2, 3].

O cérebro humano, porém, não obtém a profundidade somente de uma maneira, ele estima a distância de objetos através de diversas formas diferentes, e escolhe a que melhor se encaixa naquele momento. Computadores também podem estimar a profundidade de diversas maneiras, como profundidade por foco, defoco, perspectiva, e a mais conhecida e utilizada atualmente, a estereoscopia [4].

A estereoscopia, em termos simples, é como seres humanos obtém a percepção de profundidade. Através de duas imagens obtidas ao mesmo tempo de perspectivas diferentes, é possível inferir dados sobre a distância de objetos [8]. O processo de obtenção da profundidade através da estereoscopia consiste em quatro passos básicos: Obter as imagens, calibrar a câmera e tratar as imagens, obter o mapa de disparidade, e obter o mapa de profundidade [2, 3].

Porém, obter as imagens, e calibrar a câmera, não são processos triviais. As duas câmeras utilizadas devem estar em um *setup* fixo e conhecido para que não ocorra divergências entre as imagens coletadas. Além disso, caso o plano de visão de uma das câmeras sofrer mudanças, a outra câmera deve ser alterada da mesma forma.

A disparidade é qualquer mudança entre a posição de objetos no par de imagens, e é inversamente proporcional à distância do objeto até a lente da câmera [1]. Portanto a distância pode ser calculada obtendo a disparidade de um par estéreo, e a disparidade pode ser estimada através da correspondência dos objetos nas imagens.

Segundo Trucco et al [3], um sistema estéreo deve resolver dois problemas principais. O primeiro é a correspondência, que é o processo de encontrar os pixels da primeira imagem na segunda imagem. O segundo é conhecido como Reconstrução, que é a percepção de profundidade obtida através da diferença de posição dos objetos correspondentes nas duas imagens.

O problema da correspondência, pode ainda ser dividido em duas abordagens. A primeira, é a abordagem por Intensidade, na

qual o objeto encontrado na primeira imagem será procurado na segunda imagem inteira. A segunda, é a abordagem por Janela, na qual é determinada uma janela com tamanho fixo, e a correspondência do objeto na primeira imagem será procurado somente dentro dessa janela, seguindo a linha epipolar [1]. No primeiro caso, é mais provável que se encontre a correspondência de todos os pixels, visto que a procura é realizada na imagem inteira, porém a busca se torna dispendiosa devido ao grande número de cálculos que devem ser realizados. No segundo caso, a janela limita o número de cálculos a serem realizados, porém uma janela muito pequena pode não englobar a correspondência correta.

Este trabalho em desenvolvimento, propõe a construção de um *software* para o controle das câmeras estéreo, e a estimação da disparidade através da correspondência por Janela, visto que o mesmo realiza menos cálculos, e com uma janela de tamanho adequado, o resultado final da disparidade pode ser equivalente à correspondência por Intensidade.

2. DESENVOLVIMENTO DO SOFTWARE

A coleta de imagens e posicionamento de câmeras para fins científicos pode ser feita de várias maneiras, uma delas seria utilizando a própria interface das câmeras, se a mesma permitir que isso seja feito de uma maneira eficiente, ou podemos utilizar um *middleware*, que teria o papel de fazer a configuração e captura de imagens. Criar um *middleware* para tal tarefa seria contra-produtivo caso o objetivo seja utilizar poucas câmeras e obter poucas imagens, mas como se fez necessário garantir que as imagens sejam síncronas, e por se fazer necessário obter uma grande quantidade destas imagens a automação do processo de captura e controle das câmeras começam a fazer sentido.

Este trabalho utiliza um *middleware* para captura, o CtrlCam[6], desenvolvido especialmente para abstrair a interface de controle de câmeras, e obtenção da mesma, além da biblioteca *opencv*[7]. Com essas partes, é possível posicionar as câmeras adequadamente e capturar imagens de duas câmeras simultaneamente. Um resultado deste programa é demonstrado na Figura 3.

As câmeras utilizadas neste projeto foram do modelo IP607W, da marca Smarteye, com as configurações de brilho em 8, contraste em 4, frequência 60Hz e resolução 640x480, escolhidos empiricamente como as que produziam a melhor qualidade de imagens [8].

3. METODOLOGIA

Essa seção apresenta os passos e ferramentas utilizadas para o desenvolvimento deste trabalho. O primeiro passo é a obtenção das imagens através do *middleware* desenvolvido, e calibração das câmeras utilizando a *toolbox* de calibração estéreo do Matlab. O segundo é o cálculo da disparidade utilizando a biblioteca de visão computacional *OpenCV*. O último passo é a análise dos resultados para otimizar o algoritmo desenvolvido.

4. CALIBRAÇÃO DA CÂMERA

Existem métodos da geometria epipolar, que permitem a estimação da disparidade sem a calibração da câmera, como desenvolvido por Kukelova [9]. Porém, para a reconstrução da cena

tridimensionalmente, é necessário conhecer os parâmetros intrínsecos e extrínsecos da câmera no mundo real [2].

A calibração de imagens estéreo calcula a relação geométrica entre as matrizes de rotação e translação nas duas câmeras no espaço tridimensional [10]. Para efetuar a calibração das câmeras, foi utilizado a *toolbox* de calibração estéreo do Matlab.

A utilização da *toolbox* é simples, bastando apenas indicar quais são as imagens referentes as duas câmeras. Como pode ser visualizado na Figura 1, o próprio software calcula a disposição das imagens no plano tridimensional, e os representa em um gráfico tridimensional.

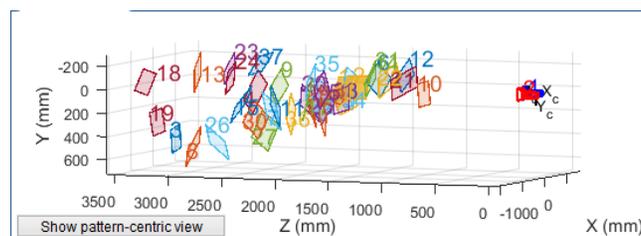


Figura 1: Disposição das imagens de teste no plano

Além da representação das imagens no plano tridimensional, a *toolbox* também calcula os erros de reprojeção em cada imagem, e os representa em um gráfico de barras, como pode ser visualizado na Figura 2. Assim, é possível descobrir os *outliers*, bem como a média do erro da reprojeção.

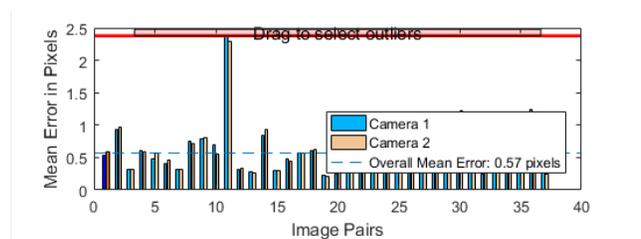


Figura 2: Erros de reprojeção

4.1 Correção das imagens

Com as câmeras calibradas, é possível realizar o processo de retificação e eliminação da distorção radial das imagens. Esses dois processos eliminam erros que ocorrem durante a captura das imagens, como pode ser visualizado na Figura 3 (a), linhas que deveriam ser retas na imagem, aparecem curvas, o que não ocorre na Figura 3 (b), onde foi realizado o processo de retificação e eliminação da distorção radial.



Figura 3: Correção das imagens

5. OBTENÇÃO DO MAPA DE DISPARIDADE

Para o cálculo do mapa de disparidade, foi utilizada a linguagem de programação C++, juntamente com a biblioteca de visão computacional OpenCV na versão 3.1, pois versões anteriores não possuem um suporte satisfatório a operações estéreo, necessárias para a realização deste trabalho. A IDE utilizada é o Netbeans na versão 8.1.

5.1 Conversão de Cores

Após especificadas as imagens à serem analisadas, ambas são convertidas para tons de cinza utilizando uma função disponível na biblioteca OpenCV.

Mapas de disparidade calculados com base nas três matrizes, geralmente obtém resultados melhores, porém exigem muito mais processamento computacional. Ao converter a imagem para tons de cinza, os cálculos são limitados a somente uma matriz, obtendo resultados bons, sem comprometer o desempenho [9].

5.2 Obtendo o Mapa de Disparidade

Para a obtenção do mapa de disparidade, é preciso resolver os dois problemas básicos, citados na seção 1. Para isso o OpenCV disponibiliza algumas funções que ajudam nesses problemas. As principais funções utilizadas serão detalhadas abaixo.

5.2.1 *StereoBM::Create()*

Função utilizada para estabelecer a correspondência entre as imagens analisadas. Recebe como parâmetros dois inteiros. O primeiro define alcance máximo da busca pela disparidade. O algoritmo buscará a melhor disparidade desde 0, que é a disparidade padrão, até o número informado.

O segundo define o tamanho do bloco que será utilizado pelo algoritmo para realizar a comparação das imagens. Deve sempre ser um número ímpar, pois deve existir um pixel central. Quanto maior o bloco, mais suave é o resultado, porém mais impreciso. Quanto menor o bloco, mais detalhado é o mapa, porém a chance de ocorrer erros é maior.

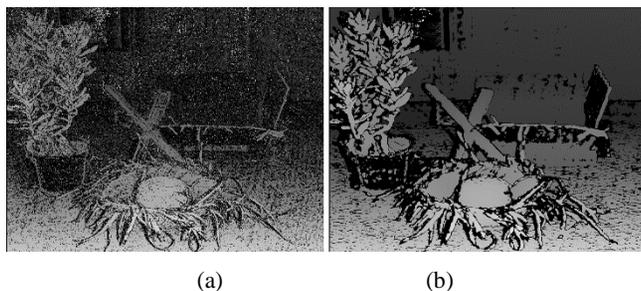


Figura 4: Resultado com janelas de tamanhos diferentes

A Figura 4 (a) demonstra o resultado da estimação da disparidade utilizando uma janela de 7 pixels. A Figura 4 (b) demonstra o resultado da estimação de disparidade utilizando uma janela de 25 pixels. Podemos perceber que a Figura 4 (a) possui apesar de mais ruidosa, possui um detalhamento maior da disparidade. Já Figura 4 (b) apesar de mais suave, apresenta menos detalhes.

Vale ressaltar que no mapa de disparidade quanto maior o valor do pixel, ou seja, quanto mais branco o pixel na imagem, mais próximo ele está da câmera.

5.2.2 *Compute()*

Função utilizada para computar a disparidade do par estéreo analisado. Recebe como parâmetros o par estéreo e a matriz de saída da disparidade. A matriz de saída é computada em 16-bits, e portanto deve ser normalizada para 8-bits antes de ser exibida.

5.2.3 *Normalize()*

Função utilizada para normalizar a matriz de saída. Recebe como parâmetros a matriz de entrada, matriz de saída, os valores mínimos e máximos de saída, e o tipo da saída, nesse caso *unsigned integer 8-bits*.

6. Resultados

Como pode ser visualizado na Figura 5, os resultados obtidos com as imagens capturadas não foram satisfatórios. Um dos fatores considerados para esse resultado, é a qualidade ruim do sensor da câmera utilizada, resultando em imagens com baixa resolução e muito ruído.

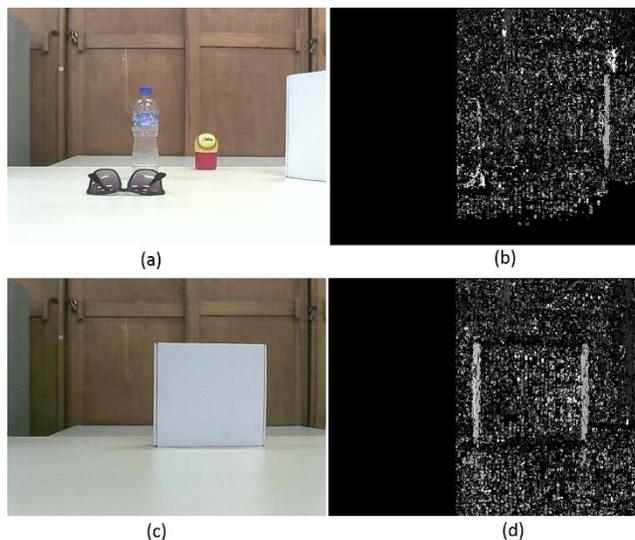


Figura 5: Resultados com as imagens capturadas

Para testar a efetividade do algoritmo desenvolvido, optou-se por utilizar imagens disponibilizadas por Nescic [11], as quais também fazem parte de um Sistema estéreo, e possuem uma qualidade melhor do que as obtidas pelas câmeras próprias.

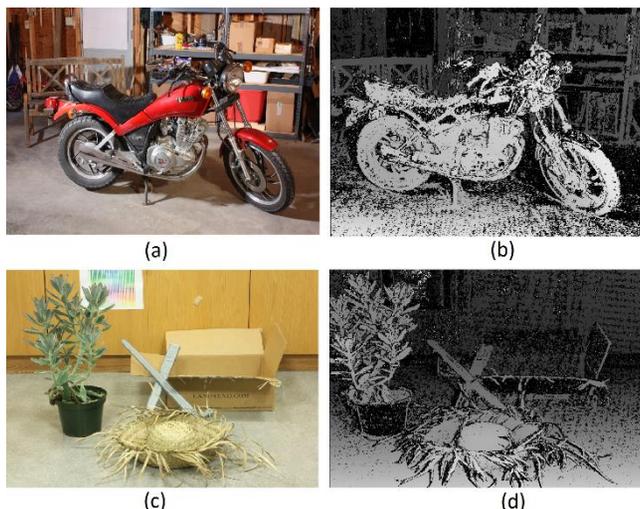


Figura 6: Resultados com imagens de Nesic

Como pode ser visualizado na Figura 6, os resultados obtidos com as imagens de Nesic foram muito melhores, apresentando um mapa de disparidade mais denso.

7. CONCLUSÕES E CONTINUAÇÃO DO TRABALHO

Como pode ser visualizado na Figura 6, apesar dos dois testes utilizarem o mesmo algoritmo com os mesmos parâmetros, os resultados obtidos com as imagens de Nesic foram muito melhores. Uma hipótese para essa diferença dos resultados, é a baixa qualidade das imagens obtidas.

Apesar do resultado obtido utilizando as imagens de Nesic ser melhor do que o obtido com as câmeras próprias, este ainda não é o ideal, portanto o algoritmo ainda deve ser melhorado.

Uma das propostas desse trabalho, é a obtenção do mapa de disparidade utilizando imagens próprias. Portanto um dos problemas a serem resolvidos é a baixa qualidade das imagens, através de câmeras melhores, porém utilizando o mesmo *middleware* aqui desenvolvido

8. REFERÊNCIAS

- [1] Acharyya, A. et al. 2016. Depth Estimation From Focus and Disparity. In *IEEE International Conference on Image Processing (ICIP)* (Phoenix, USA, September 25 - 28, 2016).
- [2] Forsyth, D., Ponce, J., *Computer Vision, A Modern Approach*, Pearson, Nova Jersey, 2011.
- [3] Trucco, E., Verri, A., *Introductory Techniques for 3-D Computer Vision*, Prentice Hall, Nova Jersey, 1998.
- [4] Lakshmanan, R. and Senthilnathan, R. 2016. Depth Map Based Reactive Planning to Aid in Navigation for Visually Challenged. In *IEEE International Conference on Engineering and Technology (ICETECH)* (Coimbatore, India, March 17 - 18, 2016)
- [5] Fernandez, C.S. et al. 2015. Stereoscopy and Haptics Human Eye AR App. In *IEEE Games and Entertainment Media Conference (GEM)* (Toronto, Canada, October 14 - 16, 2015)
- [6] Felchar, C. 2017. CtrlCam, Acesso em 23/08/2017, Disponível em <<https://bitbucket.org/Cauef/ctrlcam>>.
- [7] OpenCV. 2014. Home, Acesso em 23/08/2017, Disponível em <<http://opencv.org/>>.
- [8] Smart Eye Group. 2014. Home, Acesso em 23/08/2017, Disponível em <<http://www.smarteyegroup.com/>>.
- [9] Kukulova, Z. et al. 2015. Efficient Solution to the Epipolar Geometry for Radially Distorted Cameras. In *IEEE International Conference on Computer Vision (ICCV)* (Santiago, Chile, December 13 - 16, 2015)
- [10] Shete, P. Sarode, D. Bose, S. 2014. A Real-Time Stereo Rectification of High Definition Image Stream Using GPU. In *International Conference on Advances in Computing, Communications and Informatics (ICACCI)* (Nova Delhi, India, September 24 - 27, 2014)
- [11] Nesic et al. 2014. Stereo datasets with ground truth, Acesso em 23/08/2017, Disponível em <<http://vision.middlebury.edu/stereo/data/scenes2014/>>.

Adjacent vertex distinguishing edge coloring on complete split graphs and split-indifference graphs

Thamirys Moreira dos Santos Rauch
Federal University of Technology - Paraná
Câmpus Ponta Grossa
Ponta Grossa - PR, Brazil
thamirauch@hotmail.com

Sheila Morais de Almeida
Federal University of Technology - Paraná
Câmpus Ponta Grossa
Ponta Grossa - PR, Brazil
sheilaalmeida@utfpr.edu.br

ABSTRACT

A proper edge coloring of a graph G is an assignment of colors to the edges of G so that the colors of any two adjacent edges are distinct. Given a graph G with a proper edge coloring, the set of colors of a vertex v is the set of colors assigned to the edges incident to v . Two vertices are distinguishable if their sets of colors are distinct. An adjacent vertex distinguishing (AVD) edge coloring of G is a proper edge coloring such that every two adjacent vertices are distinguishable. The minimum number of colors to an AVD edge coloring of a graph G is the AVD chromatic index. There are partial results on the AVD chromatic index for the classes of split-indifference graphs and complete split graphs. In this paper we present the AVD chromatic index for the remaining graphs in these classes.

Keywords

Adjacent distinguishing edge coloring; Split graph; Indifference graph

1. INTRODUCTION

In this paper, we consider simple, finite and undirected graphs. We denote a graph G with vertex set $V(G)$ and edge set $E(G)$ by $G = (V(G), E(G))$, and the maximum degree of G by $\Delta(G)$. A *proper edge coloring* of G is an assignment of colors to the edges of G so that no adjacent edges receive the same color. The minimum number of colors for which a graph G has a proper edge coloring is the *chromatic index* of G , denoted by $\chi'(G)$. Given a proper edge coloring of G , the *set of colors* of a vertex $v \in V(G)$ is the set of colors assigned to the edges incident to v , denoted by $C(v)$. Two vertices u and v are *distinguishable* when $C(u) \neq C(v)$. An *adjacent vertex distinguishing* (AVD) edge coloring of a graph G is a proper edge coloring of G such that every two adjacent vertices are distinguishable. Note that there is no AVD edge coloring for the complete graph K_2 . The *AVD Edge Coloring Problem* was introduced by Zhang et al. in

2002 [8] and consists in determining the least number of colors for an AVD edge coloring of a graph G , called *AVD chromatic index* and denoted $\chi'_a(G)$. In the same paper, Zhang et al. [8] mentioned, without giving more details, that some network problems can be converted to the AVD Edge Coloring Problem. They also presented the first results on the AVD chromatic index.

THEOREM 1. [8] *If G is the disjoint union of n connected components G_1, G_2, \dots, G_n , and $|V(G_i)| \geq 3$, $1 \leq i \leq n$, then $\chi'_a(G) = \max\{\chi'_a(G_i), 1 \leq i \leq n\}$.*

By Theorem 1, it is sufficient to consider connected graphs to solve the AVD Edge Coloring Problem. Hence, from now, we consider all the graphs connected. Zhang et al. [8] also determined the AVD chromatic index for trees, complete graphs, cycles, and complete bipartite graphs. In the same paper, they proposed the following conjecture.

CONJECTURE 2. [8] *If G is a connected graph and $|V(G)| \geq 6$, $\chi'_a(G) \leq \Delta(G) + 2$.*

Balister et al. [1] proved the Conjecture 2 for bipartite graphs and for every graph G with $\Delta(G) = 3$.

Once the AVD chromatic index were determined for complete graphs and some results were known for bipartite graphs, it is interesting to consider this problem on split graphs, since every split graph is an edge disjoint union of a complete graph and a bipartite graph. Precisely, a *split graph* $G = [Q, S]$ is a graph whose vertex set can be partitioned into a clique Q and an independent set S , where a *clique* is a set of pairwise adjacent vertices and an *independent set* is a set of pairwise non-adjacent vertices. A *complete split graph* is a split graph $G = [Q, S]$ where every vertex $u \in Q$ is adjacent to every vertex $v \in S$. In [7], Vilas-Bôas and Mello proved that the Conjecture 2 is true for complete split graphs. Let $G = [Q, S]$ be a complete split graph. They also show that if G has odd $\Delta(G)$ and $|Q| > |S|^2$, then $\chi'_a(G) = \Delta(G) + 2$. If G has $|Q| \geq 2$, they proved that $\chi'_a(G) = \Delta(G) + 1$ when $\Delta(G)$ is even or when $|Q| \leq |S|^2 - |S|$. Note that when $|Q| = 1$ the complete split graph is the complete bipartite graph $K_{1,m}$ for which $\chi'_a(G)$ was determined by Zhang et al. [8]. Therefore, it remains to determine $\chi'_a(G)$ when $|Q| \geq 2$, $\Delta(G)$ is odd, and $|S|^2 - |S| < |Q| \leq |S|^2$. In this case, Vilas-Bôas and Mello [7] conjecture that $\chi'_a(G) = \Delta(G) + 1$.

In the same paper, Vilas-Bôas and Mello [7] considered the split-indifference graphs. An *indifference graph* is a graph

whose vertex set can be linearly ordered such that vertices belonging to the same clique are consecutive. An *split-indifference graph* is a graph that is simultaneously split and indifference. They proved that the Conjecture 2 is true for split-indifference graphs. Furthermore, for a split-indifference graph G , they determined $\chi'_a(G)$ when $|V(G)|$ is even and in some cases when $|V(G)|$ is odd.

In this paper we conclude the work of Vilas-Bôas and Mello [7], presenting the AVD chromatic index for the remaining complete split graphs and split-indifference graphs.

2. THEORETICAL FRAMEWORK

In this section, we present definitions and previous results that are important to the development of this work.

If a graph G satisfies $|E(G)| > \Delta(G) \lfloor \frac{|V(G)|}{2} \rfloor$, then G is *overfull*. If G has a subgraph H with $\Delta(H) = \Delta(G)$ and H is overfull, then G is *subgraph-overfull*. The overfull and subgraph-overfull graphs have $\chi'(G) = \Delta(G) + 1$.

A *universal vertex* of a graph G is a vertex of G with degree $|V(G)| - 1$. If G has a universal vertex and the number of edges in the complement of G is less than $\frac{\Delta(G)}{2}$, then G is overfull. In 1981, Plantholt determined the chromatic index for graphs with universal vertex, as presented in the next theorem.

THEOREM 3. [5] *Let G be a graph with universal vertex. $\chi'(G) = \Delta(G)$ if, and only if, G is not overfull.*

Given a graph G with a set of vertices $X \subseteq V(G)$, the *subgraph of G induced by X* is the subgraph $H = (V(H), E(H))$ with $V(H) = X$ and $E(H) = \{uv : u, v \in X \wedge uv \in E(G)\}$. The *core* of a graph G , denoted $\Lambda(G)$, is the set of maximum degree vertices of G . Fournier [3] determined the chromatic indices of graphs with acyclic $G[\Lambda(G)]$.

THEOREM 4. [3] *If $G[\Lambda(G)]$ is a forest, then $\chi'(G) = \Delta(G)$.*

When a graph G does not have maximum degree adjacent vertices, its core is an independent set and consequently $G[\Lambda(G)]$ is a forest. By Theorem 4, $\chi'(G) = \Delta(G)$. Consider an edge coloring of G with $\Delta(G)$ colors. If any two vertices of G have distinct degrees, the cardinalities of its sets of colors are different. Hence, any two sets of colors are distinguishable. Therefore, $\chi'_a(G) = \Delta(G)$. This result is presented in the following theorem of Zhang et al.

THEOREM 5. [8] *If G is a graph where the degree of every two adjacent vertices are different, then $\chi'_a(G) = \Delta(G)$.*

On the other hand, if G is a graph with two adjacent maximum degree vertices u and v , then $C(u) = C(v)$ for any edge coloring of G with $\Delta(G)$ colors. This observation implies a lower bound for the AVD chromatic index of G , as follows.

THEOREM 6. [8] *If G is a graph with two adjacent maximum degree vertices, then $\chi'_a(G) \geq \Delta(G) + 1$.*

The lower bound given by Theorem 6 is tight as the following two theorems show.

THEOREM 7. [8] *If G is a tree with $|V(G)| \geq 3$, then*

$$\chi'_a(G) = \begin{cases} \Delta(G), & \text{if there are no maximum} \\ & \text{degree adjacent vertices;} \\ \Delta(G) + 1, & \text{otherwise.} \end{cases}$$

THEOREM 8. [8] *If K_n is a complete graph with n vertices, then*

$$\chi'_a(K_n) = \begin{cases} \Delta(K_n) + 1, & \text{if } n \text{ is odd;} \\ \Delta(K_n) + 2, & \text{if } n \text{ is even.} \end{cases}$$

THEOREM 9. [8] *If $K_{m,n}$ is a complete bipartite with $1 \leq m \leq n$, then*

$$\chi'_a(K_{m,n}) = \begin{cases} n, & \text{if } m < n; \\ n + 2, & \text{if } m = n > 1. \end{cases}$$

A *proper total coloring* of a graph $G = (V(G), E(G))$ is an assignment of colors to $V(G) \cup E(G)$ such that no adjacent elements have the same color. The least number of colors that allows a proper total coloring of a graph G is the *total chromatic number*, denoted as $\chi''(G)$. Chen et al. [2] determine the total chromatic number for complete split graphs, presented in the next theorem.

THEOREM 10. [2] *Let G be a split graph. If $\Delta(G)$ is even, then $\chi''(G) = \Delta(G) + 1$.*

Considering the complete split graphs, the known results on the AVD chromatic index are presented below.

THEOREM 11. [7] *If $G = [Q, S]$ is a complete split graph and $\Delta(G)$ is odd, then $\chi'_a(G) \leq \Delta(G) + 2$.*

THEOREM 12. [7] *Let $G = [Q, S]$ be a complete split graph where $|Q| \geq 2$. If $\Delta(G)$ is even or $|Q| \leq |S|^2 - |S|$, then $\chi'_a(G) = \Delta(G) + 1$. If $\Delta(G)$ is odd and $|Q| > |S|^2$, then $\chi'_a(G) = \Delta(G) + 2$.*

Therefore, by Theorem 12, it remains to determine $\chi'_a(G)$ when $\Delta(G)$ is odd and $|S|^2 - |S| < |Q| \leq |S|^2$. For this case, Vilas-Bôas and Mello presented the following conjecture.

CONJECTURE 13. [7] *If $G = [Q, S]$ is a complete split graph, $\Delta(G)$ is odd, and $|S|^2 - |S| < |Q| \leq |S|^2$, then $\chi'_a(G) = \Delta(G) + 1$.*

Theorem 14 shows a characterization for split-indifference graphs, which partition its vertex set into cliques. This partition was used by Vilas-Bôas and Mello [7] on their results.

THEOREM 14. [4] *A graph G is an split-indifference graph if and only if*

1. G is a complete graph, or
2. G is the union of two complete graphs G_1, G_2 , such that $G_1 \setminus G_2 = K_1$, or
3. G is the union of three complete graphs G_1, G_2 , and G_3 , such that $G_1 \setminus G_2 = K_1, G_3 \setminus G_2 = K_1, V(G_1) \cap V(G_2) \neq \emptyset$, and $V(G_1) \cup V(G_3) = V(G)$, or

4. G is the union of three complete graphs G_1, G_2, G_3 , such that $G_1 \setminus G_2 = K_1, G_3 \setminus G_2 = K_1$, and $V(G_1) \cap V(G_3) = \emptyset$.

The Figure 2 shows a schematic representation of the split-indifference graphs according to the characterization given by Theorem 14, where each circle represents a clique with its respective vertices.

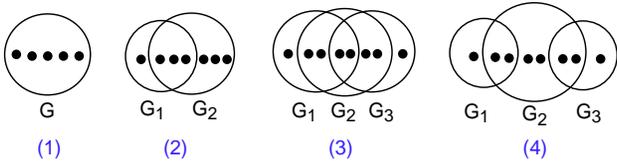


Figure 1: The four cases of the characterization of split-indifference graphs according to Theorem 14.

Let G be a split-indifference graph. If G satisfies the Case (1) of Theorem 14, the AVD chromatic index is determined by Theorem 8. Consider that G satisfies the Case (2) of Theorem 14. If $|\Lambda(G)| = 1$, then G is a tree and the AVD chromatic index is determined by Theorem 7. If $|\Lambda(G)| \geq 2$, then $\chi'_a(G)$ is determined by Vilas-Bôas and Mello, as follows.

THEOREM 15. [6] *Let G be a split-indifference graph such that G is the union of two complete graphs G_1 and G_2 , and $G_1 \setminus G_2 = K_1$. If $|V(G)|$ is odd and $|\Lambda(G)| \geq 2$, then $\chi'_a(G) = \Delta(G) + 1$.*

THEOREM 16. [6] *Let G be a split-indifference graph such that G is the union of two complete graphs G_1 and G_2 , and $G_1 \setminus G_2 = K_1$. If $|V(G)|$ is even and $|\Lambda(G)| \geq 2$, then*

$$\chi'_a(G) = \begin{cases} \Delta(G) + 1, & \text{if } 2 \leq |\Lambda(G)| \leq \frac{3\Delta(G)+1}{4}; \\ \Delta(G) + 2, & \text{if } |\Lambda(G)| > \frac{3\Delta(G)+1}{4}. \end{cases}$$

Consider that G satisfies Case (3) of Theorem 14. Suppose without loss of generality that $|V(G_1) \cap V(G_2)| \geq |V(G_2) \cap V(G_3)|$. There are two cases: when $|V(G)|$ is odd and when $|V(G)|$ is even.

THEOREM 17. [6] *Let G be a split-indifference graph such that G is the union of three complete graphs G_1, G_2 , and $G_3, G_1 \setminus G_2 = K_1, G_3 \setminus G_2 = K_1, V(G_1) \cap V(G_2) \neq \emptyset$, and $V(G_1) \cup V(G_3) = V(G)$. If $|V(G)|$ is odd, then*

$$\chi'_a(G) = \begin{cases} \Delta(G), & \text{if } \Lambda(G) = 1 \text{ and } |V(G_1) \cap V(G_2)| = |V(G_2) \cap V(G_3)|; \\ \Delta(G), & \text{if } \Lambda(G) = 1 \text{ and } |V(G_1) \cap V(G_2)| = |V(G_2) \cap V(G_3)| + 2; \\ \Delta(G) + 1, & \text{otherwise.} \end{cases}$$

THEOREM 18. [6] *Let G be a split-indifference graph such that G is the union of three complete graphs G_1, G_2 , and $G_3, G_1 \setminus G_2 = K_1, G_3 \setminus G_2 = K_1, V(G_1) \cap V(G_2) \neq \emptyset$, and $V(G_1) \cup V(G_3) = V(G)$. If $|V(G)|$ is even, then*

$$\chi'_a(G) = \begin{cases} \Delta(G), & \text{if } \Lambda(G) = 1; \\ \Delta(G) + 1, & \text{if } \Lambda(G) > \frac{3\Delta(G)+1}{4} \text{ and}; \\ \Delta(G) + 2, & \text{otherwise.} \end{cases}$$

Finally, consider that G satisfies Case (4) of Theorem 14. There are two cases: when $|V(G)|$ is odd and when $|V(G)|$ is even.

THEOREM 19. [6] *Let G be a split-indifference graph such that G is the union of three complete graphs G_1, G_2, G_3 , where $G_1 \setminus G_2 = K_1, G_3 \setminus G_2 = K_1$, and $V(G_1) \cap V(G_3) = \emptyset$. If $|V(G)|$ is even, then $\chi'_a(G) = \Delta(G) + 1$, otherwise, $\chi'_a(G) \leq \Delta(G) + 2$.*

When $|V(G)|$ is odd, Vilas-Bôas and Mello [6] presented the following partial results.

THEOREM 20. [6] *Let G be a split-indifference graph such that G is the union of three complete graphs G_1, G_2, G_3 , where $G_1 \setminus G_2 = K_1, G_3 \setminus G_2 = K_1$, and $V(G_1) \cap V(G_3) = \emptyset$. If $|V(G)|$ is odd then*

$$\chi'_a(G) = \begin{cases} \Delta(G) + 1, & \text{if } |V(G_1) \cap V(G_2)| \leq \frac{\Delta(G)+1}{2}; \\ \Delta(G) + 2, & \text{if } |V(G_1) \cap V(G_2)| \geq \frac{3\Delta(G)+1}{4}. \end{cases}$$

Now, it remains to consider the Case (4) when $|V(G)|$ is odd and $\frac{\Delta(G)+1}{2} < |V(G_1) \cap V(G_2)| < \frac{3\Delta(G)+1}{4}$.

THEOREM 21. [6] *Let G be a split-indifference graph such that G is the union of three complete graphs G_1, G_2, G_3 , where $G_1 \setminus G_2 = K_1, G_3 \setminus G_2 = K_1$, and $V(G_1) \cap V(G_3) = \emptyset$. If $|V(G)|$ is odd, $|V(G_1) \cap V(G_2)| = \frac{\Delta(G)+1}{2} + p, |V(G_2) \setminus (V(G_1) \cup V(G_3))| \geq p$, where p is an integer, $0 \leq p < \frac{\Delta(G)-1}{4}$, and $|V(G_1) \cap V(G_2)| \geq |V(G_2) \cap V(G_3)|$, then $\chi'_a(G) = \Delta(G) + 1$.*

For the other cases, Vilas-Bôas and Mello posed Conjecture 22.

CONJECTURE 22. [6] *Let G be a split-indifference graph such that G is the union of three complete graphs G_1, G_2, G_3 , where $G_1 \setminus G_2 = K_1, G_3 \setminus G_2 = K_1$, and $V(G_1) \cap V(G_3) = \emptyset$. If $|V(G)|$ is odd, $|V(G_1) \cap V(G_2)| = \frac{\Delta(G)+1}{2} + p, |V(G_2) \setminus (V(G_1) \cup V(G_3))| < p$, where p is an integer, $0 \leq p < \frac{\Delta(G)-1}{4}$, and $|V(G_1) \cap V(G_2)| \geq |V(G_2) \cap V(G_3)|$, then $\chi'_a(G) = \Delta(G) + 2$.*

3. RESULTS

In this section we present the AVD chromatic index for the split graphs that satisfies the hypothesis of conjectures 13 and 22.

To investigate the Conjecture 13, we consider a complete split graph $G = [Q, S]$ with odd $\Delta(G)$ and $|S^2| - |S| < |Q| \leq |S^2|$. Then we construct a new graph G^* by adding a new vertex v^* to G and connect v^* to every vertex of Q . Since G^* has a universal vertex, $\chi'(G^*) = \Delta(G^*)$ if and only if G^* is not overfull, by Theorem 3. The next lemma proves that when G satisfies the hypothesis of the Conjecture 13, the graph G^* is not overfull.

LEMMA 23. *Let $G = [Q, S]$ be a complete split graph with odd $\Delta(G)$ and $|Q| \leq |S^2|$. If G^* is the graph obtained from G by adding a new vertex v^* adjacent to every vertex of Q , then G^* is not overfull.*

PROOF. Let $G = [Q, S]$ be a complete split graph with odd $\Delta(G)$ and $|Q| \leq |S^2|$. Construct a graph G^* from G ,

adding a new vertex v^* and connecting v^* to every vertex of Q . Thus, $|E(G^*)| = |E(G)| + |Q| = \frac{|Q|(|Q|-1)}{2} + |Q||S| + |Q| = |Q|\frac{|Q+|S|}{2} + |Q|\frac{|S|}{2} + \frac{|Q|}{2}$. By hypothesis $|Q| \leq |S|^2$, then $|Q|\frac{|Q+|S|}{2} + |Q|\frac{|S|}{2} + \frac{|Q|}{2} \leq |Q|\frac{|Q+|S|}{2} + |Q|\frac{|S|}{2} + \frac{|S|^2}{2} = |Q|\frac{|Q+|S|}{2} + |S|\frac{|Q+|S|}{2}$. By hypothesis, $\Delta(G) = |Q| + |S| - 1$ is odd, so $|Q| + |S|$ is even. Then, $|E(G^*)| \leq |Q|\frac{|Q+|S|}{2} + |S|\frac{|Q+|S|}{2} = (|Q| + |S|)\lfloor \frac{|Q+|S|+1}{2} \rfloor = \Delta(G^*)\lfloor \frac{|V(G^*)|}{2} \rfloor$.

Therefore, G^* is not overfull. \square

The next theorem shows Conjecture 13 is true.

THEOREM 24. *If $G = [Q, S]$ is a complete split graph, $\Delta(G)$ is odd, and $|S^2| - |S| < |Q| \leq |S^2|$, then $\chi'_a(G) = \Delta(G) + 1$.*

PROOF. Let $G = [Q, S]$ be a complete split graph, with odd $\Delta(G)$ and $|S^2| - |S| < |Q| \leq |S^2|$. By the hypothesis, $|S| \geq 2$ and $|Q| \geq 4$. So, G has maximum degree adjacent vertices and $\chi'_a(G) \geq \Delta(G) + 1$ by Theorem 6.

We create a new graph G^* by adding a new vertex v^* to G and connect v^* to every vertex of Q . Since G^* has a universal vertex, $\chi'(G^*) = \Delta(G^*)$ if and only if G^* is not overfull, by Theorem 3. By Lemma 23, G^* is not overfull. Thus, consider an edge-coloring of G^* with $\Delta(G^*)$ colors and remove the vertex v^* .

The resulting graph is G with an AVD-edge-coloring. In fact, the degree of each vertex in S is $|Q|$ and the degree of each vertex in Q is $|Q| - 1 + |S|$. Since $|S| \geq 2$, we have $|Q| < |Q| - 1 + |S|$. Then, for any pair of vertices $v \in Q$ and $u \in S$, $|C(v)| > |C(u)|$ and therefore $C(v) \neq C(u)$. Moreover, there is exactly one color missing in each vertex of Q . By construction, the colors missing in any two vertices of Q are pairwise distinct, since this colors were used to color edges incident to v^* . Hence, the sets of colors of any two vertices in Q are different. Therefore, it is an AVD-edge-coloring for G and $\chi'_a(G) = \Delta(G^*) = \Delta(G) + 1$. \square

Now, we prove that Conjecture 22 is true. Let G be a split-indifference graph such that G is the union of three complete graphs G_1, G_2, G_3 , where $G_1 \setminus G_2 = K_1, G_3 \setminus G_2 = K_1$, and $V(G_1) \cap V(G_3) = \emptyset$.

THEOREM 25. *Let G be a split-indifference graph without universal vertex. If $|V(G)|$ is odd, $|V(G_1) \cap V(G_2)| = (\Delta(G) + 1)/2 + p$, $|V(G_2) \setminus (V(G_1) \cup V(G_3))| < p$, $p \in \mathbb{Z}$, $0 \leq p < \frac{\Delta(G)-1}{4}$ and $|V(G_1) \cap V(G_2)| \geq |V(G_2) \cap V(G_3)|$, then $\chi'_a(G) = \Delta(G) + 2$.*

PROOF. Construct a graph G^* by adding a vertex v^* adjacent to every maximum degree vertex of G . Observe that $\Delta(G) = |Q|$, $\Delta(G^*) = |Q| + 1$, and there is no universal vertex in G^* .

Consider the induced subgraph $H = G^*[V(G_1) \cup V(G_2) \cup \{v^*\}]$. Note that $\Delta(H) = \Delta(G^*)$ and H has a universal vertex. Now we will show that H is overfull and therefore, G^* is subgraph-overfull.

Note that the number of edges in the complement of H is $2|V(G_2) \setminus (V(G_1) \cup V(G_3))| + |V(G_2) \cap V(G_3)| + 1$. Since $|V(G_2) \cap V(G_3)| = |Q| - |V(G_2) \setminus (V(G_1) \cup V(G_3))| - |V(G_1) \cap V(G_2)|$, the number of edges in the complement of H is $|V(G_2) \setminus (V(G_1) \cup V(G_3))| + |Q| - |V(G_1) \cap V(G_2)| + 1$. By hypothesis, $|V(G_1) \cap V(G_2)| = (\Delta(G) + 1)/2 + p$. Hence, the number of edges in the complement of H is $|V(G_2) \setminus (V(G_1) \cup V(G_3))| + |Q| - (\Delta(G) + 1)/2 - p + 1 = |V(G_2) \setminus (V(G_1) \cup$

$V(G_3))| + |Q| - (|Q| + 1)/2 - p + 1 = \frac{\Delta(H)}{2} + |V(G_2) \setminus (V(G_1) \cup V(G_3))| - p$. By hypothesis, $|V(G_2) \setminus (V(G_1) \cup V(G_3))| < p$. Thus, the number of edges in the complement of H is less than $\frac{\Delta(H)}{2}$, which implies that H is overfull and consequently G^* is subgraph-overfull. So, $\chi'(G^*) = \Delta(G^*) + 1$.

Therefore, there is no edge coloring for G with $\Delta(G^*) = \Delta(G) + 1$ colors that allows pairwise distinct sets of colors for the maximum degree vertices. So, $\chi'_a(G) \geq \Delta(G^*) + 1 = \Delta(G) + 2$. By Theorem 11, $\chi'_a(G) = \Delta(G) + 2$. \square

4. CONCLUSION

Considering a complete split graph G , we conclude that if $V(G)$ has a partition into a clique Q and an independent set S such that $|Q| > |S|^2$ and $\Delta(G)$ is odd, then $\chi'_a(G) = \Delta(G) + 2$, otherwise, $\chi'_a(G) = \Delta(G) + 1$.

For split indifference graphs, four cases were considered, according to the characterization of these graphs given by Theorem 14. Before considering such cases it is importante to note that if $|\Lambda(G)| = 1$, then G is a tree and the AVD chromatic index is determined by Theorem 7. Then, the following conclusions are about graphs with $|\Lambda(G)| \geq 2$. When G is a complete graph, the AVD Edge Coloring Problem is solved by Theorem 8. For the next cases, consider G_1, G_2 and G_3 as described in Theorem 14. When G is the union of two complete graphs G_1 and G_2 , the AVD chromatic index is determined by Theorems 15 and 16. When G is the union of three complete graphs, G_1, G_2 , and G_3 , such that $V(G_1) \cap V(G_3) \neq \emptyset$, the AVD-Edge Coloring Problem is solved by Theorems 17 and 18. Finally, if G is the union of three complete graphs, G_1, G_2 , and G_3 , such that $V(G_1) \cap V(G_3) = \emptyset$, then the AVD chromatic index is determined by Theorems 19, 20, 21 and 25.

As a future work, we plan to investigate the AVD chromatic indices of other split graphs, such as the split-comparability graphs, a superclass of split-indifference graphs.

5. REFERENCES

- [1] P. N. Balister, E. Gyori, J. Lehel, and R. H. Schelp. Adjacent vertex distinguishing edge-colorings. *SIAM Journal on Discrete Mathematics*, 21(1):237–250, 2007.
- [2] B.-L. Chen, H.-L. Fu, M. Ko, et al. Total chromatic number and chromatic index of split graphs. *Journal of Combinatorial Mathematics and Combinatorial Computing*, 17, 1995.
- [3] J.-C. Fournier. Colorations des arêtes d'un graphe. *Cahiers du CERO*, 15:311–314, 1973.
- [4] C. Ortiz, N. Maculan, and J. L. Szwarcfiter. Characterizing and edge-coloring split-indifference graphs. *Discrete applied mathematics*, 82:209–217, 1998.
- [5] M. Plantholt. The chromatic index of graphs with a spanning star. *Journal of graph theory*, 5(1):45–53, 1981.
- [6] A. d. M. Vilas-Bôas. Coloração de arestas semiforte de grafos split. Master's thesis, Universidade Estadual de Campinas, Instituto de Computação, Campinas, SP, 2015.
- [7] A. d. M. Vilas-Bôas and C. P. de Mello. The avd-edge-coloring conjecture for some split graphs. *Matemática contemporânea*, 44:1–10, 2015.
- [8] Z. Zhang, L. Liu, and J. Wang. Adjacent strong edge coloring of graphs. *Applied mathematics letters*, 15(5):623–626, 2002.

Criticalidade arco-íris dos grafos resultantes de produto cartesiano de ciclos e caminhos

Aleffer Rocha
Universidade Tecnológica Federal do Paraná
Av. Monteiro Lobato, s/n - Km 04 CEP
84016-210 - Ponta Grossa - PR - Brasil
aleffer@alunos.utfpr.edu.br

Sheila Morais de Almeida
Universidade Tecnológica Federal do Paraná
Av. Monteiro Lobato, s/n - Km 04 CEP
84016-210 - Ponta Grossa - PR - Brasil
sheilaalmeida@utfpr.edu.br

RESUMO

Dado um grafo G , uma coloração de arestas de G é uma atribuição de cores para as arestas de G . Uma coloração de arestas é própria se arestas adjacentes têm cores distintas. 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 qualquer aresta de G aumenta o seu número de conexão arco-íris. Neste trabalho mostramos que o produto cartesiano $P_m \times P_n$ é arco-íris crítico se, e somente se, é um caminho P_n , $n > 1$, ou um C_4 . Também mostramos que os produtos cartesianos $C_m \times P_n$ não são arco-íris críticos quando m é par e $n > 1$.

Palavras-chave

Coloração Arco-íris; Produto Cartesiano; Grafos Arco-íris Críticos

ABSTRACT

Given a graph G , an edge-coloring of G is an assignment of colors to the edges of G . A proper edge-coloring is an edge coloring if adjacent edges have different colors. 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 as $rc(G)$, 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 show that the cartesian product $P_m \times P_n$ is rainbow critical if and only if it is a path P_n , $n > 1$, or a C_4 . We also show that the cartesian product $C_m \times P_n$ is not rainbow critical when m is even and $n > 1$.

Keywords

Rainbow Coloring; Cartesian Product; Rainbow Critical Graphs

1. INTRODUÇÃO

Nesse trabalho, os grafos considerados são simples e conexos. Uma coloração de arestas em G é uma atribuição de cores para as arestas de G . As cores em geral são representadas por números inteiros. Uma coloração de arestas é própria se arestas incidentes em um mesmo vértice têm cores distintas. Dado um grafo G com uma coloração de arestas não necessariamente própria, um caminho P em G é arco-íris se as cores das arestas de P são duas a duas 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 [3]. O Problema da Coloração Arco-íris consiste em encontrar o menor número de cores que permite uma coloração arco-íris de um grafo conexo G . Este número é conhecido como número de conexão arco-íris e é denotado por $rc(G)$. Observe que quando G é um grafo desconexo, existe um par de vértices u e v que não está conectado por um caminho. Neste caso, não há cores suficientes para se ter um caminho arco-íris entre u e v , e consequentemente o número de conexão arco-íris é infinito.

Sabe-se que o Problema da Coloração Arco-íris é NP-difícil [2] e decidir se $rc(G) = k$, para $k \geq 3$, é um problema NP-completo [1]. Entretanto, em algumas classes de grafos, o número de conexão arco-íris é trivial, por exemplo, para uma árvore T com n vértices $rc(T) = n - 1$; para um grafo completo K_n , $rc(K_n) = 1$; e para um ciclo C_n , $rc(C_n) = \lceil \frac{n}{2} \rceil$ [3].

Em face da dificuldade de calcular o valor de $rc(G)$ para um grafo qualquer, alguns trabalhos estabelecem limitantes superiores (cada vez mais justos) para o número de conexão arco-íris. Um dos primeiros resultados nesse sentido explora a quantidade de arestas presentes na árvore geradora de G . Uma árvore geradora em um grafo G é um subgrafo de G com o mesmo conjunto de vértices que G e que não contém ciclos. Observe que grafos que contém ciclos possuem mais de uma árvore geradora. Assim, para obter uma coloração arco-íris de qualquer grafo simples e conexo G , é suficiente atribuir cores distintas para todas as arestas de uma das suas árvores geradoras, o que implica $rc(G) \leq |V(G)| - 1$.

Ainda considerando a dificuldade para se determinar o valor de $rc(G)$ em um grafo G qualquer, outros estudos se concentram na solução do problema em classes de grafos mais restritas. Uma dessas classes é a dos grafos resultantes da operação de produto cartesiano. Considere dois grafos

simples $G = (V(G), E(G))$ e $H = (V(H), E(H))$. O produto cartesiano de G e H , denotado por $G \times H$, é um grafo com o conjunto de vértices dado pelo produto cartesiano $V(G) \times V(H)$, onde existe aresta entre dois vértices (v_i, u_j) e (v_k, u_l) se e somente se $v_i v_k \in E(G)$ e $u_j = u_l \in V(H)$ ou se $u_j u_l \in E(H)$ e $v_i = v_k \in V(G)$.

Para os grafos resultantes de produtos cartesianos, o Teorema 1 estabelece uma relação entre o número de conexão arco-íris e o diâmetro dos grafos participantes da operação do produto cartesiano. O conceito de diâmetro de um grafo G depende do conceito de excentricidade de um vértice v , que é a maior distância entre v e qualquer outro vértice do grafo G . O diâmetro de um grafo G , denotado por $diam(G)$, é a maior excentricidade de um vértice de G .

TEOREMA 1. [4] *Seja $G = G_1 \times G_2 \times \dots \times G_n$, $n \geq 2$, tal que G_i é conexo, $1 \leq i \leq n$. Então $rc(G) \leq \sum_{i=1}^n rc(G_i)$. Além disso, se $diam(G_i) = rc(G_i)$ para todo G_i , $1 \leq i \leq n$, então $rc(G) = \sum_{i=1}^n rc(G_i)$.*

Dos resultados para árvores e ciclos [3] e do Teorema 1, pode-se inferir que $rc(P_m \times P_n) = m + n - 2$, para dois caminhos, P_m e P_n , $2 \leq m \leq n$; $rc(C_m \times C_n) = \frac{m+n}{2}$ para dois ciclos, C_m e C_n , m e n pares, $3 < m \leq n$; e $rc(C_m \times P_n) = n - 1 + \frac{m}{2}$ para um caminho P_n e um ciclo C_m , m par. Rocha e Almeida [6] determinaram $rc(C_m \times P_n)$ quando m é ímpar e $rc(C_m \times C_n)$ quando m e n têm paridades distintas. Os mesmos autores provaram que $rc(C_m \times C_n) \leq \frac{m+n}{2}$ se m e n são ímpares, melhorando o limitante apresentado no Teorema 1.

Além do interesse em determinar o número de conexão arco-íris, há estudos sobre a criticalidade dos grafos quanto ao número de conexão arco-íris. Um grafo G é chamado grafo arco-íris crítico se, ao remover uma aresta qualquer de G , o número de conexão arco-íris de G aumenta [5]. Em [5], os autores afirmam que os produtos cartesianos $P_m \times P_n$, $m, n \geq 2$, e $C_m \times P_n$, m par e $n \geq 2$, são grafos arco-íris críticos. Neste trabalho, refutamos tais resultados e apresentamos uma prova de que o grafo $P_m \times P_n$ é arco-íris crítico se, e somente se, é um caminho P_n , $n > 1$, ou um C_4 . Também provamos que os produtos cartesianos $C_m \times P_n$ não são arco-íris críticos quando m é par e $n > 1$.

2. DEFINIÇÕES E RESULTADOS ANTERIORES

Alguns conceitos e resultados anteriores importantes para o entendimento deste trabalho são definidos nesta seção. Pelas definições de diâmetro e de número de conexão arco-íris, pode-se concluir as seguintes observações.

OBSERVAÇÃO 1. *Dados dois grafos, G e H , $diam(G \times H) = diam(G) + diam(H)$.*

OBSERVAÇÃO 2. *Dado um grafo G , $rc(G) \geq diam(G)$.*

OBSERVAÇÃO 3. *Todo caminho P_n tem $diam(P_n) = n - 1$ e todo ciclo C_n tem $diam(C_n) = \lfloor \frac{n}{2} \rfloor$.*

A seguir, são apresentados formalmente os resultados sobre os números de conexão arco-íris de grafos caminhos e ciclos, que são as classes consideradas neste trabalho. Como todo caminho é uma árvore, o Teorema 2 vale para os caminhos.

TEOREMA 2. [3] *Seja G um grafo conexo não trivial com $|E(G)|$ arestas. Então:*

- (a) $rc(G) = 1$ se e somente se G é um grafo completo,
- (b) $rc(G) = |E(G)|$ se e somente se G é uma árvore.

Para os ciclos, vale o Teorema 3.

TEOREMA 3. [3] *Para todo ciclo C_n com $n \geq 4$, $rc(C_n) = \lfloor n/2 \rfloor$.*

Da Observação 3 e dos Teoremas 1, 2 e 3, pode-se inferir que $rc(P_m \times P_n) = m + n - 2$, para dois caminhos, P_m e P_n , com $2 \leq m \leq n$; $rc(C_m \times C_n) = \frac{m+n}{2}$ para dois ciclos, C_m e C_n , com $3 < m \leq n$ e m e n pares; e $rc(C_m \times P_n) = \frac{m}{2} + n - 1$ para um ciclo C_m e um caminho P_n com m par e $m > 3$.

3. RESULTADOS

Esta seção apresenta resultados sobre a criticalidade dos produtos cartesianos entre dois caminhos e entre um ciclo e um caminho. Os resultados estão divididos em dois teoremas, onde o primeiro garante que $P_m \times P_n$ é crítico apenas quando é um caminho P_n , $n > 1$, ou um C_4 ; e o segundo teorema garante que grafos $C_m \times P_n$ com m par e $n > 1$ não são arco-íris críticos. É importante observar que $P_m \times P_n$ e $P_n \times P_m$ são grafos isomorfos, bem como $C_m \times P_n$ e $P_n \times C_m$.

TEOREMA 4. *Sejam m e n inteiros positivos. O grafo $G = P_m \times P_n$ é arco-íris crítico se, e somente se, é um grafo P_n , $n > 1$ ou um C_4 .*

PROVA. Primeiro, observe que o grafo P_1 não possui arestas e em todo caminho P_n tal que $n > 1$, existe uma aresta cuja a remoção desconecta o grafo. Por definição, o número de conexão arco-íris do grafo desconexo é infinito. Logo, quando $n > 1$, P_n é arco-íris crítico.

O grafo $P_2 \times P_2$ é um ciclo com quatro vértices e, pelo Teorema 3, $rc(P_2 \times P_2) = rc(C_4) = 2$. O grafo $P_2 \times P_2$ sem qualquer de suas arestas é um caminho P_4 e pelo Teorema 2 tem $rc(P_4) = 3$. A Figura 1 apresenta um grafo $P_2 \times P_2$ seguido do grafo $P_2 \times P_2 - e$, mostrando que este grafo é arco-íris crítico.

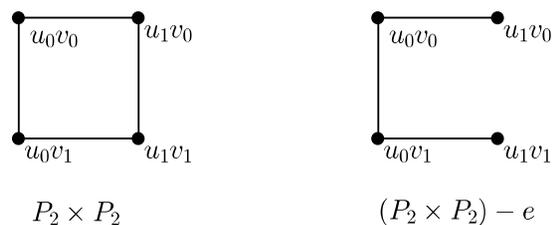


Figura 1: Grafos $P_2 \times P_2$ e $P_2 \times P_2 - e$.

Para todos os outros casos, é suficiente mostrar que existe uma aresta e no grafo $P_m \times P_n$ cuja remoção não aumenta o número de conexão arco-íris do grafo. Sem perda de generalidade, suponha que $m \geq n \geq 2$ e $P_m \times P_n$ não é um C_4 .

Para identificação dos vértices do produto cartesiano, suponha que os vértices de P_m estão rotulados, u_0, u_1, \dots, u_{m-1} , de forma que u_i é adjacente a u_{i+1} no P_m , $0 \leq$

$i < m - 1$. Similarmente, suponha que os vértices de P_n estão rotulados, v_0, v_1, \dots, v_{n-1} , e v_i é adjacente a v_{i+1} , $0 \leq i < n - 1$. Então, cada vértice do $P_m \times P_n$ é um par (u_i, v_j) , $0 \leq i < m$ e $0 \leq j < n$. A Figura 2 apresenta os vértices do grafo $P_4 \times P_3$ rotulados.

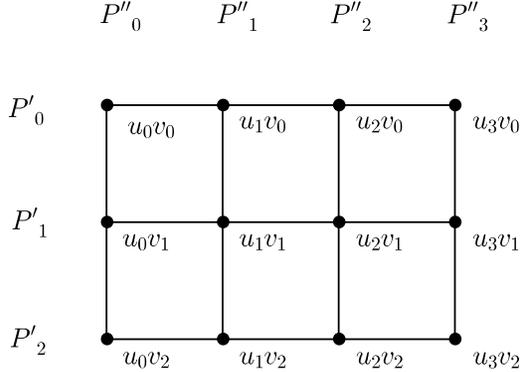


Figura 2: Rotulação dos vértices do grafo $P_4 \times P_3$.

No grafo $P_m \times P_n$, definimos os subgrafos P'_i e P''_i da seguinte forma: $P'_i = (u_0, v_i)(u_1, v_i) \dots (u_{m-1}, v_i)$, $0 \leq i < n$ e $P''_i = (u_i, v_0)(u_i, v_1) \dots (u_i, v_{n-1})$, $0 \leq i < m$.

Vamos mostrar que a remoção de uma aresta $e = (u_k, v_f)$ (u_k, v_{f+1}) em um caminho P''_k , $0 \leq k < m$, $0 \leq f < n - 1$, não aumenta o número de conexão arco-íris do grafo.

Pinte os caminhos P''_i , $0 \leq i < m$, de forma que cada aresta $(u_i, v_j)(u_i, v_{j+1})$ receba cor j , $0 \leq j < n - 1$. Pinte os caminhos P'_i de forma que cada aresta $(u_j, v_i)(u_{j+1}, v_i)$ receba a cor $n - 1 + [(i + j) \bmod (m - 1)]$. Observe que qualquer caminho P'_i ou P''_j é arco-íris, $0 \leq i < n$, $0 \leq j < m$. Além disso, o conjunto das cores usadas em qualquer caminho P'_i é disjuncto do conjunto das cores usadas em qualquer caminho P''_j , $0 \leq i < n$, $0 \leq j < m$. Por fim, observe que as arestas $(u_a, v_i)(u_a, v_{i+1})$ e $(u_b, v_i)(u_b, v_{i+1})$ estão coloridas com a mesma cor, para quaisquer i , a e b , $0 \leq i < n - 1$, $0 \leq a, b < m$.

Resta provar que entre qualquer par de vértices (u_q, v_p) e (u_s, v_r) , $0 \leq q, s < m$ e $0 \leq p, r < n$, existe um caminho arco-íris no grafo $(P_m \times P_n) - e$, onde $e = (u_k, v_f)(u_k, v_{f+1})$.

Quando $p \neq r$ e $q \neq s$, pelo menos um dos seguintes caminhos arco-íris existe:

1. caminho de (u_q, v_p) a (u_s, v_p) em P''_p concatenado com o caminho de (u_s, v_p) a (u_s, v_r) em P''_s ;
2. ou o caminho de (u_q, v_p) a (u_q, v_r) em P''_q concatenado com o caminho de (u_q, v_r) a (u_s, v_r) em P'_r .

A Figura 3 apresenta o grafo $P_4 \times P_3$ com uma aresta removida, onde as arestas pontilhadas indicam o caminho arco-íris. Observe que, neste caso, o caminho arco-íris descrito no item (2) deixou de existir quando a aresta foi removida. Entretanto o caminho descrito no item (1) ainda existe e é apresentado nesse exemplo.

Se $q = s$, suponha sem perda de generalidade que $p < r$. Se $k \neq q$ ou se $f \notin [p, r - 1]$, então o caminho arco-íris é o caminho entre (u_q, v_p) e (u_q, v_r) em P''_q . Por outro lado, se $k = q$ e $f \in [p, r - 1]$, o caminho arco-íris entre (u_q, v_p) e (u_q, v_r) depende do valor de q . Quando $q > 0$, um caminho arco-íris é o caminho de (u_q, v_p) a (u_q, v_f) em P''_q , concatenado com

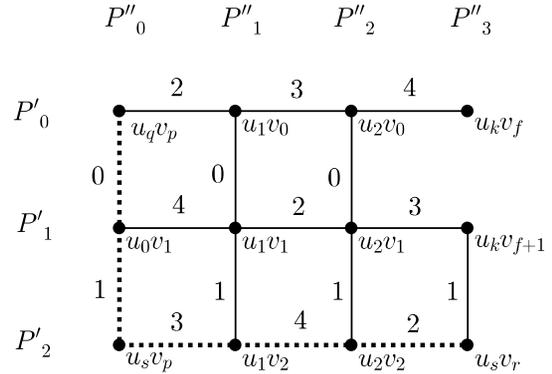


Figura 3: Grafo $(P_4 \times P_3) - e$ com caminho arco-íris.

o caminho $(u_q, v_f)(u_{q-1}, v_f)(u_{q-1}, v_{f+1})(u_q, v_{f+1})$, concatenado com o caminho de (u_q, v_{f+1}) a (u_q, v_r) em P''_q . Quando $q = 0$, um caminho arco-íris é o caminho de (u_q, v_p) a (u_q, v_f) em P''_q , concatenado com o caminho $(u_q, v_f)(u_{q+1}, v_f)(u_{q+1}, v_{f+1})(u_q, v_{f+1})$, concatenado com o caminho de (u_q, v_{f+1}) a (u_q, v_r) em P''_q .

Se $p = r$, suponha sem perda de generalidade que $q < s$. O caminho arco-íris é o caminho de (u_q, v_p) a (u_s, v_p) em P'_p . \square

TEOREMA 5. Sejam m e n inteiros positivos. O grafo $C_m \times P_n$, m par e $n \geq 2$, não é arco-íris crítico.

PROVA. Considere o grafo $C_m \times P_n$, onde C_m é um ciclo com m par e P_n é um caminho $n \geq 2$. Suponha os vértices de C_m rotulados sequencialmente, u_0, u_1, \dots, u_{m-1} , e os vértices de P_n também rotulados sequencialmente, v_0, v_1, \dots, v_{n-1} . A Figura 4 apresenta, como exemplo, o produto cartesiano $C_4 \times P_2$ com seus vértices rotulados.

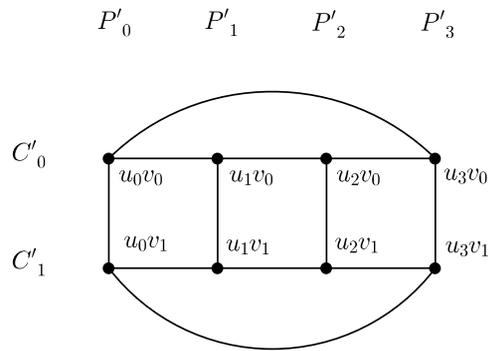


Figura 4: Grafo $C_4 \times P_2$ com vértices rotulados.

Denotamos por C'_i os subgrafos de $C_m \times P_n$ dados por $(u_0, v_i)(u_1, v_i) \dots (u_{m-1}, v_i)(u_0, v_i)$, $0 \leq i < n$. Denotamos por P'_i os subgrafos de $C_m \times P_n$ dados por $(u_i, v_0)(u_i, v_1) \dots (u_i, v_{n-1})$, $0 \leq i < m$.

Primeiro, vamos colorir o grafo $C_m \times P_n$. Pinte C'_i , $0 \leq i < n$, i par, da seguinte forma. Se $0 \leq j < m - 1$, pinte a aresta $(u_j, v_i)(u_{j+1}, v_i)$ com a cor $j \bmod \frac{m}{2}$. Pinte $(u_{m-1}, v_i)(u_0, v_i)$ com a cor $\frac{m}{2} - 1$.

Pinte C'_i , $0 \leq i < n$, i ímpar, da seguinte forma. Se

$0 \leq j < m - 1$, pinte a aresta $(u_j, v_i)(u_{j+1}, v_i)$ com a cor $(j + 1) \bmod \frac{m}{2}$. Pinte $(u_{m-1}, v_i)(u_0, v_i)$ com a cor 0.

Pinte cada aresta $(u_i, v_j)(u_i, v_{j+1}) \in P'_i$, $0 \leq i < m$ e $0 \leq j < n - 1$, com cor $\frac{m}{2} + j$.

Observe que a coloração de cada ciclo C'_i e cada caminho P'_j é arco-íris, $0 \leq i < n$ e $0 \leq j < m$. Para garantir que $C_m \times P_n$ não é arco-íris crítico, basta mostrar que existe uma aresta cuja remoção não aumenta o número de conexão arco-íris do grafo. Então, vamos considerar o grafo $C_m \times P_n - e$, onde $e = (u_k, v_f)(u_k, v_{f+1})$, $0 \leq k < m$ e $0 \leq f < n - 1$. Mostraremos que a remoção da aresta e não aumenta o número de conexão arco-íris do grafo $C_m \times P_n$.

Sejam (u_q, v_p) e (u_s, v_r) dois vértices quaisquer de $C_m \times P_n$. Vamos apresentar um caminho arco-íris entre (u_q, v_p) e (u_s, v_r) no grafo $C_m \times P_n - e$.

Se $p = r$, então o caminho arco-íris entre (u_q, v_p) e (u_s, v_r) é o menor caminho entre esses vértices no subgrafo C'_p .

Se $q = s$, suponha sem perda de generalidade que $p < r$. Se $k \neq q$ ou se $f \notin [p, r - 1]$, então o caminho arco-íris é o caminho entre (u_q, v_p) e (u_q, v_r) em P'_q . Por outro lado, se $k = q$ e $f \in [p, r - 1]$, o caminho arco-íris entre (u_q, v_p) e (u_q, v_r) depende do valor de q . Quando $q > 0$, um caminho arco-íris é o caminho de (u_q, v_p) a (u_q, v_f) em P'_q , concatenado com o caminho $(u_q, v_f)(u_{q-1}, v_f)(u_{q-1}, v_{f+1})(u_q, v_{f+1})$, concatenado com o caminho de (u_q, v_{f+1}) a (u_q, v_r) em P'_q . Quando $q = 0$, um caminho arco-íris é o caminho de (u_q, v_p) a (u_q, v_f) em P'_q , concatenado com o caminho $(u_q, v_f)(u_{q+1}, v_f)(u_{q+1}, v_{f+1})(u_q, v_{f+1})$, concatenado com o caminho de (u_q, v_{f+1}) a (u_q, v_r) em P'_q . A Figura 5 apresenta, como exemplo, o produto cartesiano $C_4 \times P_2$ com uma aresta removida, onde as arestas pontilhadas indicam o caminho arco-íris entre os vértices $u_q v_p$ e $u_s v_r$, onde $q = s = 0$, $k = q$ e $f \in [p, r - 1]$.

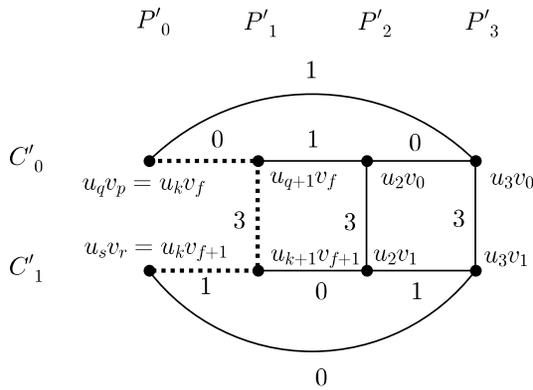


Figura 5: Grafo $(C_4 \times P_2) - e$ com caminho arco-íris.

Se $p \neq r$ e $q \neq s$, então um dos seguintes caminhos arco-íris existe:

1. o menor caminho entre (u_q, v_p) a (u_s, v_p) em C'_p concatenado com o caminho de (u_s, v_p) a (u_s, v_r) em P'_s ;
2. ou o caminho de (u_q, v_p) a (u_q, v_r) em P'_q concatenado com o menor caminho de (u_q, v_r) a (u_s, v_r) em C'_r .

Como existe pelo menos uma aresta em $C_m \times P_n$ que quando removida não altera o número de conexão arco-íris do grafo, conclui-se que o grafo $C_m \times P_n$ não é arco-íris crítico. \square

4. CONCLUSÃO

A definição apresentada anteriormente para grafos arco-íris críticos não era suficientemente precisa, considerando que não estava claro se um grafo é arco-íris crítico quando a remoção de uma aresta o desconecta. Neste trabalho, propomos que o número de conexão arco-íris de um grafo desconexo seja infinito. Dessa forma, se a remoção de uma aresta desconecta o grafo, então ele é arco-íris crítico. Considerando essa definição, todo caminho P_n , $n > 1$ tem uma aresta que quando removida aumenta o número de conexão arco-íris. Portanto, esses grafos são arco-íris críticos. Embora essa seja uma família de grafos arco-íris críticos resultantes do produto cartesiano de dois caminhos, provamos que o produto cartesiano $P_m \times P_n$ não é arco-íris crítico quando $m \geq n \geq 2$ e o grafo não é um C_4 . Observe que este resultado contrapõe o resultado apresentado por Rao e Murali [5]. Dos nossos estudos, concluímos que essa diferença se deve ao fato de que Rao e Murali consideraram uma única possibilidade de coloração. Nesta coloração, a remoção de qualquer aresta uv faz com que não exista mais caminho arco-íris entre u e v . Então, Rao e Murali afirmam que é necessária mais uma cor. Entretanto, apresentamos uma coloração diferente na qual a remoção de uma aresta específica não aumenta o número de conexão arco-íris. Pela definição de grafo arco-íris crítico apresentada por Rao e Murali [5], acreditamos que o nosso resultado está correto. Similarmente, provamos que o produto cartesiano de $C_m \times P_n$ não é arco-íris crítico quando m é par e $n \geq 2$, refutando outra afirmação de Rao e Murali [5].

5. REFERÊNCIAS

- [1] P. Ananth, M. Nasre, e K. K. Sarpatwar. Rainbow Connectivity: Hardness and Tractability. In S. Chakraborty e A. Kumar, 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, 2011. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik.
- [2] S. Chakraborty, E. Fischer, A. Matsliah, e R. Yuster. Hardness and algorithms for rainbow connectivity. *Journal of Combinatorial Optimization*, 21:330–347, 2011.
- [3] G. Chartrand, G. L. Johns, K. A. McKeon, e P. Zhang. Rainbow connection in graphs. *Mathematica Bohemica*, 133(1):85–98, 2008.
- [4] X. Li, Y. Shi, e Y. Sun. Rainbow connections of graphs: A survey. *Graphs and Combinatorics*, 29(1):1–38, 2013.
- [5] K. S. Rao e R. Murali. Rainbow critical graphs. *International Journal of Computer Application*, 4(4):252–259, 2014.
- [6] A. Rocha e S. M. Almeida. Coloração arco-íris em grafos resultantes de produto cartesiano. In *Anais do XXXVII Congresso da Sociedade Brasileira de Computação*, pages 83–86, São Paulo, SP, julho 2017.

Coloração total distinta na vizinhança em potências de caminhos

Pedro Henrique Salgado
 Universidade Tecnológica
 Federal do Paraná
 Av. Monteiro Lobato km 04
 Ponta Grossa, Brasil
 pedrosalgado@alunos.utfpr.edu.br

Mayara Midori Omai*
 Universidade Tecnológica
 Federal do Paraná
 Av. Monteiro Lobato km 04
 Ponta Grossa, Brasil
 omai@alunos.utfpr.edu.br

Sheila Moraes de Almeida
 Universidade Tecnológica
 Federal do Paraná
 Av. Monteiro Lobato km 04
 Ponta Grossa, Brasil
 sheilaalmeida@utfpr.edu.br

RESUMO

Uma coloração total própria de um grafo é uma atribuição de cores para seus vértices e arestas de forma que elementos adjacentes recebam cores distintas. Dada uma coloração total própria em um grafo G , $C(v)$ é o conjunto de cores de um vértice v , composto pelas cores das arestas que incidem em v e pela a cor do próprio v . Uma coloração total distinta na vizinhança é uma coloração total própria em que $C(u) \neq C(v)$ para todo par de vértices adjacentes u e v . O Problema da Coloração Total Distinta na Vizinhança consiste em determinar o menor número de cores necessário para se obter uma coloração total distinta na vizinhança de um grafo. Este trabalho apresenta a solução do Problema da Coloração Total Distinta na Vizinhança para as potências de caminhos P_n^k com $n > 2k + 1$.

Palavras-chave

Coloração total distinta na vizinhança; Grafos indiferença; Potências de caminhos

ABSTRACT

A proper total coloring of a graph is an assignment of colors to the vertices and edges such that adjacent elements receive distinct colors. The set of colors of a vertex v , denoted by $C(v)$, is composed by the colors of the edges incident to v and the color of v . An adjacent vertex distinguishing total coloring is a proper total coloring such that $C(u) \neq C(v)$ for every pair of adjacent vertices u and v . The Adjacent Vertex Distinguishing Total Coloring Problem consists of determining the minimum number of colors to an adjacent vertex distinguishing total coloring of a graph. This work presents the solution of the Adjacent Vertex Distinguishing Total Coloring Problem for the powers of paths P_n^k with $n > 2k + 1$.

*Bolsista da Fundação Araucária

Keywords

Adjacent vertex distinguishing total coloring; Indifference graphs; Powers of paths

1. INTRODUÇÃO

Seja $G = (V(G), E(G))$ um grafo com conjunto de vértices $V(G)$ e conjunto de arestas $E(G)$. Dois elementos do conjunto $V(G) \cup E(G)$ são *adjacentes* se são dois vértices que formam uma aresta, duas arestas que compartilham o mesmo vértice, ou uma aresta e um dos vértices em que ela incide.

Os problemas de coloração consistem originalmente em determinar o menor número de cores necessário para colorir os elementos de um grafo, de forma que elementos adjacentes tenham cores distintas. Dado um grafo G , uma *coloração total própria* consiste na atribuição de cores para os elementos do conjunto $V(G) \cup E(G)$, de forma que elementos adjacentes recebam cores distintas. Dada uma coloração total própria para um grafo G , $C(v)$ é o conjunto de cores do vértice v , composto pelas cores das arestas que incidem em v e pela a cor do próprio v . Uma *coloração total distinta na vizinhança* (ou coloração TDV) é uma coloração total própria em que $C(u) \neq C(v)$ para todo par de vértices adjacentes u e v . O *Problema da Coloração Total Distinta na Vizinhança*, introduzido por Zhang et al. [5] em 2005, consiste em determinar o menor número de cores necessário para se obter uma coloração TDV de um grafo. Tal número é conhecido como *número cromático total distinto na vizinhança* e é denotado por $\chi''_a(G)$. A Figura 1 apresenta um exemplo de coloração TDV para um grafo completo com cinco vértices. Um grafo é *completo* quando existe aresta entre todo par de vértices. Um grafo completo com n vértices é denotado por K_n .

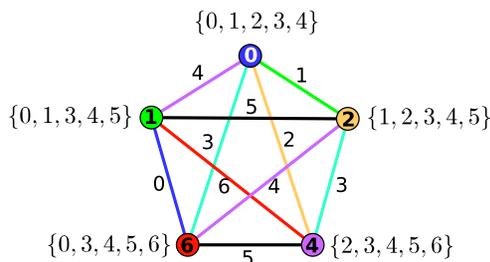


Figura 1: Coloração total distinta na vizinhança de um K_5

Neste trabalho apresentamos uma solução parcial do Problema da Coloração TDV para as potências de caminhos. Uma *potência de caminho*, P_n^k , é um grafo no qual $V(P_n^k) = \{v_0, v_1, \dots, v_{n-1}\}$ e existe aresta $v_i v_j$ se, e somente se, $|j-i| \leq k$. Um exemplo é a potência P_6^3 , apresentada na Figura 2.

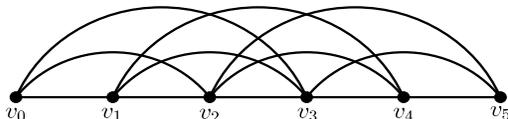


Figura 2: Grafo P_6^3

Seja $\Delta(G)$ o maior grau de um vértice no grafo G . Ao introduzir o Problema da Coloração TDV, Zhang et al. [5] estabeleceram a seguinte conjectura.

CONJECTURA 1. [5] *Seja G um grafo conexo com pelo menos dois vértices, então $\chi_a''(G) \leq \Delta(G) + 3$.*

Uma *clique* é um conjunto de vértices em que todos os vértices são adjacentes. Um grafo é *indiferença* quando seu conjunto de vértices pode ser linearmente ordenado de forma que vértices que pertencem à mesma clique sejam consecutivos nesta ordem. Tal ordem é chamada *ordem indiferença* [4]. As potências de caminhos são uma subclasse dos grafos indiferença.

Pedrotti e Mello [3] provaram que a Conjectura 1 é verdadeira para os grafos indiferença. Além disso, provaram que para um grafo indiferença G com $\Delta(G)$ ímpar e vértices adjacentes de grau máximo, $\chi_a''(G) = \Delta(G) + 2$. Quando $\Delta(G)$ é par e não existem vértices adjacentes de grau máximo, Pedrotti e Mello [3] provaram que $\chi_a''(G) = \Delta(G) + 1$. Consequentemente, o Problema da Coloração TDV em potências de caminhos permanece em aberto para os casos em que $\Delta(P_n^k)$ é ímpar e não existem vértices adjacentes de grau máximo e quando $\Delta(P_n^k)$ é par e existem vértices adjacentes de grau máximo. Esse artigo apresenta uma solução ótima para todo P_n^k com $n > 2k + 1$, um caso em que $\Delta(P_n^k)$ é par e existem vértices adjacentes de grau máximo.

2. RESULTADOS CONHECIDOS

Nesta seção apresentamos os principais resultados conhecidos para o Problema da Coloração TDV.

Quando Zhang et al. [5] definiram a coloração TDV, apresentaram os primeiros resultados sobre o número cromático total distinto na vizinhança, descritos a seguir.

TEOREMA 2. [5] *Se G é um grafo com vértices adjacentes de grau máximo, então $\chi_a''(G) \geq \Delta(G) + 2$.*

O Teorema 2 é bastante intuitivo, já que por definição $\chi_a''(G) \geq \Delta + 1$ e, quando existem vértices adjacentes de grau máximo, $\Delta + 1$ cores não são suficientes para que seus conjuntos de cores sejam distintos.

Para os grafos completos K_n , $\chi_a''(K_n)$ também é conhecido, conforme o teorema a seguir.

TEOREMA 3. [5] *Se K_n é um grafo completo com $n > 1$ vértices, então*

$$\chi_a''(K_n) = \begin{cases} \Delta(K_n) + 2, & \text{se } n \text{ é par,} \\ \Delta(K_n) + 3, & \text{se } n \text{ é ímpar.} \end{cases}$$

Um grafo é um caminho se é uma potência de caminho P_n^1 . Comumente, denota-se a potência de caminho P_n^1 por P_n . Para caminhos, rege o seguinte Teorema 4.

TEOREMA 4. [5] *Se P_n é um caminho com $n \geq 2$ vértices, então,*

$$\chi_a''(P_n) = \begin{cases} 3, & \text{se } n \in \{2, 3\}; \\ 4, & \text{se } n \geq 4. \end{cases}$$

Um *ciclo* é um grafo conexo em que todos os vértices tem grau 2. Um ciclo com n vértices é denotado por C_n . Para os ciclos, $\chi_a''(C_n)$ é dado pelo seguinte teorema.

TEOREMA 5. [5] *Se C_n é um ciclo com $n \geq 4$ vértices, então $\chi_a''(C_n) = 4$.*

Uma *árvore* é um grafo conexo que não possui um ciclo como subgrafo. Para árvores, Zhang et al. [5] apresentam o seguinte resultado.

TEOREMA 6. [5] *Se T_n é uma árvore com $n \geq 2$, então, $\chi_a''(T_n) = \Delta(T) + 2$ quando existem vértices adjacentes de grau máximo e $\chi_a''(T_n) = \Delta(T) + 1$, caso contrário.*

Um grafo é *bipartido* se podemos particionar os seus vértices em dois conjuntos A e B de forma que não existam arestas entre elementos que pertencem ao mesmo conjunto. Um grafo é *bipartido completo* se é um grafo bipartido em que vértices pertencentes a conjuntos diferentes da partição são adjacentes. Um grafo bipartido completo com partição $[A, B]$, $|A| = a$ e $|B| = b$, é denotado por $K_{a,b}$.

TEOREMA 7. [5] *Seja $K_{a,b}$ um grafo bipartido completo, com $a \geq b \geq 1$. Então,*

$$\chi_a''(K_{a,b}) = \begin{cases} 3, & \text{se } a = b = 1, \\ a + 1, & \text{se } a \geq b + 1, \\ a + 2, & \text{se } a = b \geq 2. \end{cases}$$

Em 2008, Chen [1] apresentou um limitante superior para o número cromático total distinto na vizinhança dos grafos com $\Delta(G) = 3$, como mostra o seguinte teorema.

TEOREMA 8. [1] *Se G é um grafo com $\Delta(G) = 3$, então $\chi_a''(G) \leq 6$.*

Pedrotti e Mello [3] apresentaram os seguintes resultados sobre o Problema da Coloração TDV em grafos indiferença.

TEOREMA 9. [3] *Se G é um grafo indiferença com $\Delta(G)$ par sem vértices adjacentes de grau máximo, então, $\chi_a''(G) = \Delta(G) + 1$.*

TEOREMA 10. [3] *Seja G um grafo indiferença com $\Delta(G)$ ímpar. Então, G tem uma coloração TDV com $\Delta(G) + 2$ cores e, se G possui vértices adjacentes de grau máximo, então $\chi_a''(G) = \Delta(G) + 2$.*

Para o desenvolvimento desse trabalho, será necessário o seguinte resultado sobre coloração total em grafos completos.

TEOREMA 11. *Se K_n é um grafo completo com n vértices, então*

$$\chi''(K_n) = \begin{cases} \Delta(K_n) + 2, & \text{se } n \text{ é par,} \\ \Delta(K_n) + 1, & \text{se } n \text{ é ímpar.} \end{cases}$$

3. RESULTADOS

Esta seção apresenta os resultados parciais sobre o Problema da Coloração TDV em potências de caminhos. Para compreender tais resultados algumas definições são necessário. Estas são apresentadas a seguir.

PROPOSIÇÃO 12. *Se P_n^k é uma potência de caminho com $1 < n \neq 2k + 1$, então existem vértices adjacentes de grau máximo.*

PROVA. Se $n > 2k + 1$, então $\Delta(P_n^k) = 2k$ e os vértices $v_k, v_{k+1}, \dots, v_{n-k-1}$ são vértices consecutivos de grau máximo, então, existem $(n - k - 1) - (k) + 1 = n - 2k$ vértices com grau $\Delta(P_n^k)$. Como $n > 2k + 1$, o número de $\Delta(P_n^k)$ -vértices é $n - 2k \geq 2k + 2 - 2k = 2$.

Se $n < 2k + 1$, então $\Delta(P_n^k) < 2k$ e P_n^k possui vértice universal. Logo, $\Delta(P_n^k) = n - 1$. Se $n \leq k + 1$, então P_n^k é um grafo completo e, portanto, possui vértices de grau máximo adjacentes. Se $n > k + 1$, então os graus dos vértices na ordem indiferença são $k, k + 1, \dots, n - 2, n - 1, \dots, n - 1, n - 2, n - 3, \dots, k$, respectivamente. Observe que existem exatamente dois vértices com cada grau menor que $n - 1$. Então, existem $2(n - 2 - k + 1) = 2n - 4 - 2k + 2 = 2n - 2k - 2$ vértices com grau menor que $n - 1$. Logo, o número de vértices com grau máximo é $n - (2n - 2k - 2) = n - 2n + 2k + 2 = 2k - n + 2$. Como $n < 2k + 1$, então existem $2k - (2k) + 2 = 2$ vértices adjacentes de grau máximo. \square

A vizinhança aberta de um vértice v é o conjunto dos vértices que são adjacentes a v e é denotada por $N(v)$. Uma função é injetora quando para todo par de elementos u e v do domínio da função, se $u \neq v$, então $f(u) \neq f(v)$. Ou seja, cada elemento da imagem corresponde a exatamente um elemento do domínio.

Um *pullback* de um grafo H para um grafo G , consiste em uma função $f : V(H) \rightarrow V(G)$, de modo que se $uv \in E(H)$, então $f(u)f(v) \in E(G)$ e f é injetora quando restrita a $N(v)$. Em um artigo de 1999, Figueiredo, Meidanis e Mello [2] introduziram as funções *pullback*, mostrando que, se existe um *pullback* f de um grafo G_1 em um grafo G_2 e se G_2 possui uma coloração total com t cores, então G_1 também possui uma coloração total com t cores. A Figura 3 mostra um grafo G_2 com uma coloração total própria $\tau : V(G_2) \cup E(G_2) \rightarrow \{0, 1, 2, 3, 4\}$. Para obter-se uma coloração total própria $\alpha : V(G_1) \cup E(G_1) \rightarrow \{0, 1, 2, 3, 4\}$, uma função *pullback* $f : V(G_1) \rightarrow V(G_2)$ atribui a cada vértice de G_1 o rótulo de um vértice de G_2 . Então, α é definida por $\alpha(v_i) = \tau(v_i)$ e $\alpha(v_i v_j) = \tau(v_i v_j)$, $0 \leq i, j \leq 3$. Observe que α é uma função composta $f \circ \tau$ que resulta em uma coloração total com $\Delta(G_1) + 2$ cores para G_1 .

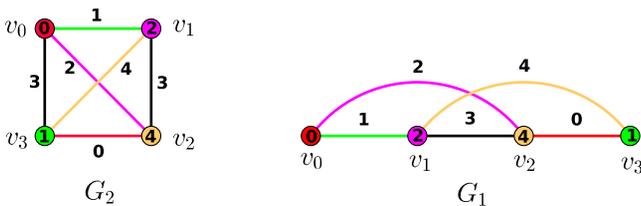


Figura 3: Coloração total G_1 através de um pullback do G_1 para o G_2

Nesse trabalho, consideramos uma potência de caminho P_n^k com $n > 2k + 1$. Neste caso, $\Delta(P_n^k) = 2k$. Vamos

mostrar que esses grafos tem uma coloração TDV com $2k + 2$ cores. Para tanto, realizamos um *pullback* $f : V(P_n^k) \rightarrow K_{2k+2}$. Entretanto, pelo Teorema 11 não existe coloração total própria para o grafo K_{2k+2} com $2k + 2$ cores. Mesmo assim, consideramos uma coloração total não-própria τ de um grafo completo K_{2k+2} e provamos que $f \circ \tau$ é coloração total própria e distinta vizinhança para o grafo P_n^k . Isso prova que o número cromático total distinto na vizinhança do P_n^k , quando $n > 2k + 1$, é exatamente $2k + 2$.

TEOREMA 13. *Seja P_n^k uma potência de caminho com $n > 2k + 1$, então $\chi''_\alpha(P_n^k) = \Delta(P_n^k) + 2$.*

PROVA. Considere uma potência de caminho com $n > 2k + 1$ e vértices rotulados $v_0 v_1 \dots v_{n-1}$ de acordo com uma ordem indiferença. Como $n > 2k + 1$, pela Proposição 12, $\Delta(P_n^k) = 2k$ é par e o grafo possui vértices adjacentes de grau máximo. Portanto, $\chi''_\alpha(P_n^k) \geq \Delta(P_n^k) + 2$ pelo Teorema 2. Rotule os vértices do grafo K_{2k+2} , $u_0, u_1, \dots, u_{2k+1}$. Seja $f : V(P_n^k) \rightarrow V(K_{2k+2})$, tal que $f(v_i) = u_{i \pmod{2k+2}}$. Considere a coloração total não-própria do grafo K_{2k+2} dada pela função $\tau : V(K_{2k+2}) \cup E(K_{2k+2}) \rightarrow \{0, 1, \dots, 2k + 1\}$, apresentada a seguir.

$$\begin{aligned} \tau(v_i) &= 2i \pmod{2k + 2}, \\ \tau(v_i v_j) &= (i + j) \pmod{2k + 2}. \end{aligned}$$

Seja $\alpha : V(P_n^k) \cup E(P_n^k) \rightarrow \{0, 1, \dots, 2k + 1\}$, uma coloração total do P_n^k tal que $\alpha = f \circ \tau$.

Pela definição de α , nenhum vértice v_i do grafo P_n^k é adjacente a dois vértices v_j e v_p tal que $j \equiv p \pmod{2k + 2}$, caso contrário $\Delta(P_n^k) > 2k$. Portanto, não existem arestas com a mesma cor que incidem no mesmo vértice. Também não existem no P_n^k vértices adjacentes v_i e v_j tal que $\alpha(v_i) = \alpha(v_i v_j)$, caso contrário, $2i \equiv (i + j) \pmod{2k + 2}$ e a distância entre v_i e v_j é maior que $2k$. Suponha, por contradição, que existem dois vértices adjacentes, v_i e v_j , tais que $\alpha(v_i) = \alpha(v_j)$. Sem perda de generalidade, considere que $i < j$. Lembre-se que $2i \equiv 2j \pmod{2k + 2}$. Como $i < j$, então $2j - 2i \geq 2k + 2$. Logo, $j - i \geq k + 1$, um absurdo, já que v_i e v_j são adjacentes e estão a uma distância maior que k no P_n^k . Dessa forma, α é uma coloração total própria do P_n^k .

Para garantir que α é uma coloração TDV, vamos provar que em cada vértice v_i do grafo P_n^k , existe uma cor c tal que $c \notin C(v_i)$ e $c \in C(v_j)$, para cada par de vértices adjacentes v_i e v_j , $j > i$. Seja v_i um vértice do P_n^k e $v_j = v_{i+k}$, o vértice adjacente a v_i que está mais a direita na ordem indiferença. Note que, por construção, v_i não é adjacente a $v_{j+1} = v_{i+k+1}$.

Pela função α , a cor $c = (i + j + 1) \pmod{2k + 2}$ aparece somente nas arestas $v_i v_q$, em que $q = j + 1 + x(2k + 2)$, $x \in \mathbb{Z}$. Portanto, todo $v_q \notin N(v_i)$ e $c \notin C(v_i)$. Além disso, vamos mostrar que $\alpha(v_i) \neq c$. Suponha por contradição que $\alpha(v_i) = c = \alpha(v_i v_{j+1})$. Então, $2i \equiv (i + j + 1) \pmod{2k + 2}$. Como $i < j + 1$, $2i + x(2k + 2) = j + 1$ para algum $x \in \mathbb{Z}$, $x \geq 1$, então, $j - i \geq \Delta(P_n^k) + 1 = 2k + 1$, um absurdo já que a distância entre v_i e v_j é no máximo k . Portanto, $\alpha(v_i) \neq c = \alpha(v_i v_{j+1})$.

Agora vamos mostrar que a cor c incide em todos os vértices de v_{i+1} a v_j . Pela função α , $c = \alpha(v_i v_{j+1}) = \alpha(v_{i+m} v_{j+1-m})$, para $1 \leq m < \frac{j-i+1}{2}$. Além disso, se o número de vértices de $v_i + 1$ até v_j é ímpar, então o vértice $v_{\frac{j+i+1}{2}}$ tem cor c . Dessa forma, a cor c incide em todos os vértices entre v_{i+1} e v_j .

Portanto, a coloração α é uma coloração TDV, utilizando $\chi''_a(G) = \Delta(P_n^k) + 2$ cores. \square

Um exemplo de coloração TDV de um P_{11}^3 é apresentado na Figura 4.

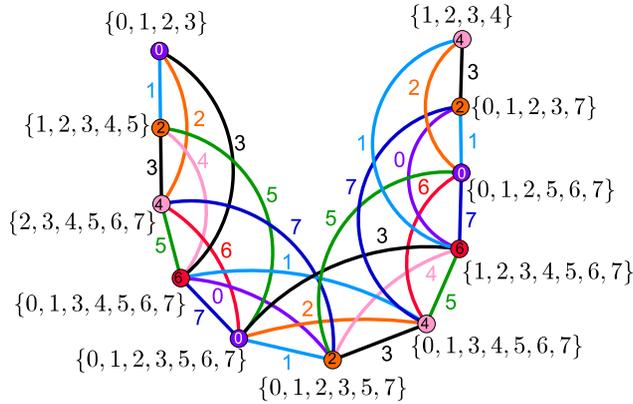


Figura 4: Coloração total distinta na vizinhança de um P_{11}^3

4. CONCLUSÃO

Se a potência de caminho tem $n \leq k + 1$, então o P_n^k é um grafo completo e, pelo Teorema 3, o número cromático total distinto na vizinhança é conhecido.

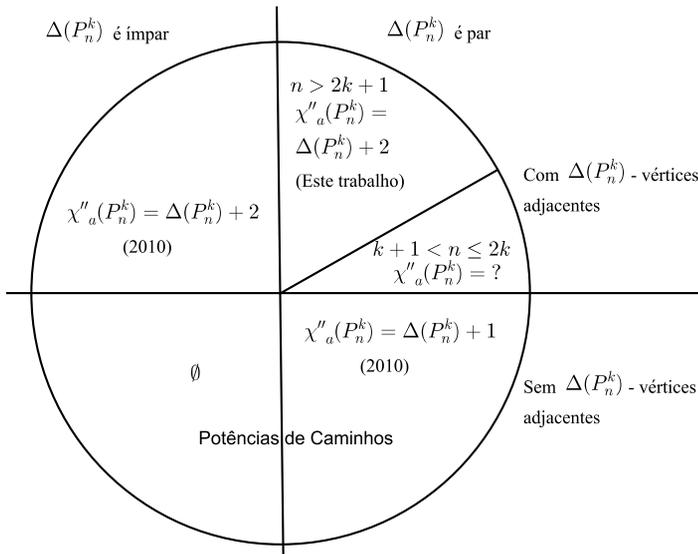


Figura 5: Estado da arte do Problema da Coloração Total Distinta na Vizinhança em potências de caminhos

A Figura 5 apresenta o estado da arte do Problema da Coloração TDV em potências de caminhos que não são grafos completos. O lado esquerdo concentra os grafos com $\Delta(P_n^k)$ ímpar e o lado direito apresenta os grafos com $\Delta(P_n^k)$ par. O lado superior apresenta os grafos com vértices adjacentes de grau máximo e o lado inferior apresenta os grafos sem vértices adjacentes de grau máximo. Os resultados desse trabalho concentram-se no quadrante superior direito. Quando

$n > 2k + 1$, obrigatoriamente $\Delta(P_n^k) = 2k$ é par e determinamos o número cromático total dessas potências de caminhos. O caso em que $k + 1 < n \leq 2k$ e $\Delta(P_n^k)$ é par permanece aberto. Portanto, nesse quadrante ainda há grafos para os quais o problema não está resolvido. Observe que, quando a potência de caminho P_n^k não têm vértices adjacentes de grau máximo, então $n = 2k + 1$ e $\chi''_a(P_n^k) = \Delta(P_n^k) + 1$, pelo Teorema 9. Neste caso, $\Delta(P_n^k)$ é par. Então, não existem potências de caminhos no quadrante inferior esquerdo.

Os grafos do quadrante superior esquerdo são potências de caminhos com $n \neq 2k + 1$. Neste caso, $\chi''_a(P_n^k) = \Delta(P_n^k) + 2$, pelo Teorema 10. Note que este caso só ocorre quando $n < 2k + 1$.

5. REFERÊNCIAS

- [1] X. Chen. On the adjacent vertex distinguishing total coloring numbers of graphs with $\Delta = 3$. *Discrete Mathematics*, 308(17):4003–4007, 2008.
- [2] C. M. d. Figueiredo, J. Meidanis e C. P. de Mello. Total-chromatic number and chromatic index of dually chordal graphs. *Information processing letters*, 70(3):147–152, 1999.
- [3] V. Pedrotti e C. P. de Mello. Adjacent-vertex-distinguishing total coloring of indifference graphs. *Matemática contemporânea*, 39:101–110, 2010.
- [4] F. S. Roberts. On the compatibility between a graph and a simple order. *Journal of Combinatorial Theory, série B*, 11(1):28 – 38, 1971.
- [5] Z. Zhang, X. Chen, J. Li, B. Yao, X. Lu e J. Wang. On adjacent-vertex-distinguishing total coloring of graphs. *Science in China série A: mathematics*, 48(3):289–299, 2005.

Coloração total distinta na vizinhança em grafos 4-partidos completos

Matheus Scaketti
Universidade Tecnológica Federal do Paraná
Av Monteiro Lobato, s/n - Km 04 CEP 84016-210
Ponta Grossa, Brasil
mts.scaketti@gmail.com

Sheila Morais de Almeida
Universidade Tecnológica Federal do Paraná
Av Monteiro Lobato, s/n - Km 04 CEP 84016-210
Ponta Grossa, Brasil
sheilaalmeida@utfpr.edu.br

RESUMO

Uma coloração total de um grafo G consiste na atribuição de cores para os vértices e para as arestas de G de forma que elementos adjacentes recebam cores distintas. O conjunto de cores de um vértice u é composto pela cor do vértice u e pelas cores das arestas que incidem em u . Uma coloração total distinta na vizinhança é em uma coloração total onde vértices adjacentes possuem conjuntos de cores distintos. O Problema da Coloração Total Distinta na Vizinhança consiste em determinar o menor número de cores para se obter uma coloração total distinta na vizinhança para um dado grafo. Neste trabalho, resolvemos o Problema da Coloração Total Distinta na Vizinhança para os grafos 4-partidos completos que possuem vértices adjacentes de grau máximo. Quando o grafo 4-partido completo não possui vértices adjacentes de grau máximo, apresentamos um limitante superior que difere do ótimo em no máximo um.

Palavras-chave

4-partidos completos; Coloração total distinta na vizinhança; Grafos multipartidos completos

ABSTRACT

A total coloring is an assignment of colors to the edges and the vertices of a graph such that adjacent elements receive distinct colors. The color-set of a vertex u is the set with the color of u and the colors of the edges incident to u . An adjacent vertex distinguishing total coloring is a total coloring such that any pair of adjacent vertices have distinct color-sets. The Adjacent Vertex Distinguishing Total Coloring Problem is to determine the least number of colors to obtain an adjacent vertex distinguishing total coloring for a given graph. In this work we solve the Adjacent Vertex Distinguishing Total Coloring Problem of complete 4-partite graphs with adjacent vertices of maximum degree. When the complete 4-partite graph has no adjacent vertices with

maximum degree, we present an upper bound that differs from the optimum by at most one.

Keywords

Complete 4-partite graphs; Adjacent vertex distinguishing total coloring; Complete multipartite graphs

1. INTRODUÇÃO

Neste trabalho os grafos considerados são simples e conexos. Um grafo $G = (V(G), E(G))$ tem conjunto de vértices $V(G)$ e conjunto de arestas $E(G)$. Dois elementos são adjacentes se são dois vértices conectados por uma aresta, duas arestas incidentes no mesmo vértice ou uma aresta e um vértice em que ela incide. Uma coloração total é uma atribuição de cores para os vértices e arestas de um grafo G e é própria se quaisquer dois elementos adjacentes têm cores distintas. O menor número de cores que permite uma coloração total de um grafo G é o número cromático total de G , denotado por $\chi''(G)$. Dado um grafo G com uma coloração total própria, o conjunto de cores de u é o conjunto composto pela cor do vértice u e pelas cores das arestas que incidem em u . Uma coloração total distinta na vizinhança (coloração TDV) é uma coloração total própria, onde os conjuntos de cores de quaisquer dois vértices adjacentes são distintos. O Problema da Coloração TDV consiste em determinar, para um dado grafo G , o menor número de cores possível que permita uma coloração TDV de G . Esse número de cores é chamado de número cromático TDV e é denotado por $\chi_a''(G)$.

Seja $\Delta(G)$ o maior grau de um vértice de G . Em 2005, Zhang et al. [6] introduziram o Problema da Coloração TDV e apresentaram a seguinte conjectura.

CONJECTURA 1. [6] Se G é um grafo simples, então $\chi_a''(G) \leq \Delta(G) + 3$.

No mesmo trabalho, Zhang et al. [6] também apresentaram o seguinte limitante inferior para o número cromático TDV de grafos com vértices adjacentes de grau máximo.

TEOREMA 2. [6] Se G é um grafo simples com vértices adjacentes de grau máximo, então $\chi_a''(G) \geq \Delta(G) + 2$.

Os mesmos autores determinaram o número cromático TDV para os grafos completos, conforme o teorema a seguir.

TEOREMA 3. [6] Se K_n é um grafo completo com n vértices, então $\chi_a''(K_n) = \Delta(K_n) + 2$ quando n é par, e $\chi_a''(K_n) = \Delta(K_n) + 3$ quando n é ímpar.

Um conjunto independente é um conjunto de vértices dois a dois não adjacentes. Os grafos k -partidos são aqueles cujo conjunto de vértices pode ser particionado em k conjuntos independentes. Se existe uma aresta entre quaisquer dois vértices em partes distintas então o grafo k -partido é chamado de k -partido completo. Quando um grafo k -partido completo tem $k = 2$ e partes com tamanhos m e n , o mesmo é chamado de bipartido completo e denotado por $K_{m,n}$. Dado um grafo k -partido completo, se todas as partes têm o mesmo tamanho, então G é chamado k -equipartido completo. O teorema a seguir valida a Conjectura 1 para os k -equipartidos completos.

TEOREMA 4. [1] *Seja G um grafo k -equipartido completo, com $k \geq 2$ e partes de tamanho $r \geq 2$. Se rk é par, então $\chi''_a(G) = \Delta(G) + 2$. Se rk é ímpar, então $\chi''_a(G) \leq \Delta(G) + 3$.*

Zhang [6] provou que o bipartido completo $K_{m,n}$ tem $\chi''_a(K_{m,n}) = \Delta(K_{m,n}) + 1$ quando $1 < n < m$, e $\chi''_a(K_{m,n}) = \Delta(K_{m,n}) + 2$, caso contrário. Para os grafos tripartidos completos, a Conjectura 1 foi provada em 2014 [1]. Mais especificamente, Luiz [1] mostrou que $\chi''_a(G) \leq \Delta(G) + 2$, quando G é um tripartido completo sem vértices adjacentes de grau máximo e que $\chi''_a(G) = \Delta(G) + 2$ quando o grafo tripartido completo tem partição $\{A, B, C\}$ onde $|A| \geq 2|B| = 2|C|$. Em 2016, Tiburcio [2] provou que se G é um tripartido completo não isomorfo ao K_3 , então $\chi''_a(G) \leq \Delta(G) + 2$.

Neste trabalho, determinamos $\chi''_a(G)$ para grafos 4-partidos completos que possuem vértices adjacentes de grau máximo. Quando o grafo 4-partido completo não possui vértices adjacentes de grau máximo, apresentamos um limitante superior que difere do ótimo em no máximo um.

2. DEFINIÇÕES E RESULTADOS ANTERIORES

As definições e teoremas desta seção são importantes para a construção dos resultados que serão apresentados.

Um emparelhamento é um conjunto de arestas duas a duas não adjacentes e é maximal quando não está propriamente contido em outro emparelhamento. Um emparelhamento M é perfeito quando em cada vértice de G incide uma aresta de M .

Uma coloração de arestas é uma atribuição de cores para as arestas de G e é própria se quaisquer duas arestas adjacentes têm cores distintas. O menor número de cores que permite uma coloração de arestas própria é chamado de índice cromático de G e denotado por $\chi'(G)$.

Dado um grafo G e um subconjunto de vértices $X \subseteq V(G)$, o subgrafo de G induzido por X , denotado por $G[X]$, é o subgrafo H com $V(H) = X$ e $E(H) = \{uv : u \in X \wedge v \in X \wedge uv \in E(G)\}$. O núcleo de um grafo G é o conjunto de vértices com grau $\Delta(G)$.

TEOREMA 5. [3] *Seja G um grafo simples. Se o núcleo de G é um conjunto independente, então $\chi'(G) = \Delta(G)$.*

Em 1989, Yap [4] determinou um limitante superior para o número cromático total de todos os grafos k -partidos completos, conforme o teorema a seguir.

TEOREMA 6. [4] *Se G é um grafo k -partido completo, então $\chi''(G) \leq \Delta(G) + 2$.*

Um grafo é k -regular se todos os seus vértices têm grau k .

TEOREMA 7. [5] *Sejam n e p dois inteiros positivos tais que $n > p$. Se pelo menos um entre n e p é par, então é possível construir um grafo $(n - p - 1)$ -regular de ordem n .*

3. RESULTADOS

Nesta seção apresentamos provas de que $\chi''_a(G) \leq \Delta(G) + 2$ para os grafos 4-partidos completos, provando a Conjectura 1 para esta classe. A demonstração foi dividida em casos, de acordo com o tamanho de cada parte. Note que é suficiente considerar os casos em que o grafo tem partes de tamanhos diferentes, pois quando o grafo é 4-equipartido completo, então tem ordem par e a solução é dada pelo Teorema 4.

Se as cardinalidades das partes do grafo k -partido completo G são duas a duas distintas, então qualquer coloração total de G é uma coloração TDV, já que os conjuntos de cores de vértices adjacentes se distinguem por sua cardinalidade. Pelo Teorema 6, conclui-se o seguinte resultado.

TEOREMA 8. [6] *Se G é um grafo k -partido completo tal que as cardinalidades de quaisquer dois conjuntos da partição são distintas, então $\chi''_a(G) \leq \Delta(G) + 2$.*

Resta considerar os casos em que existem duas ou três partes com o mesmo tamanho. A divisão dos casos é feita da seguinte forma: as três partes menores têm a mesma cardinalidade; exatamente duas partes menores têm a mesma cardinalidade; as três partes maiores têm a mesma cardinalidade; as duas partes de mesma cardinalidade não são a menor nem a maior; e apenas as duas partes maiores têm a mesma cardinalidade.

LEMA 9. *Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$. Se $|A| = |B| = |C| = |D| - 1$, então $\chi''_a(G) = \Delta(G) + 2$.*

PROVA. Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$ tal que $|A| = |B| = |C| = |D| - 1$. Considere um emparelhamento perfeito M_{BC} no subgrafo induzido $G[B \cup C]$. Pinte as arestas de M_{BC} com a cor $\Delta(G) + 1$. Observe que A é o núcleo do subgrafo $G - M_{BC}$ e, pelo Teorema 5, é possível colorir as arestas de $G - M_{BC}$ com $\Delta(G)$ cores. Observe que $|V(G)| = 4|A| + 1$ é ímpar. Como o número de vértices é ímpar, em qualquer coloração de arestas de $G - M_{BC}$, cada cor incide em no máximo $|V(G)| - 1$ vértices. Como foram usadas $\Delta(G)$ cores, cada uma delas falta em algum vértice de $G - M_{BC}$. Como em $G - M_{BC}$ existem $\Delta(G)$ vértices com grau igual a $\Delta(G) - 1$, em cada um deles falta uma cor. Então, as cores que faltam em dois vértices quaisquer de $G - M_{BC}$ são duas a duas distintas. Pinte os vértices de B , C e D com as cores que faltam nesses vértices. Faça um emparelhamento maximal no subgrafo induzido $G[C \cup D]$ e troque as cores das arestas desse emparelhamento pela cor $\Delta(G) + 2$. Pinte os vértices de A com a cor $\Delta(G) + 2$.

A Figura 1 apresenta um esquema da coloração aplicada ao grafo 4-partido completo com três partes de tamanho 3 e uma parte de tamanho 4. As arestas que não aparecem coloridas estão pintadas com uma coloração de arestas própria usando cores de 1 a 10. Os vértices de $B \cup C \cup D$ estão coloridos com cores que sobraram após a coloração de arestas própria do grafo $G - M_{BC}$.

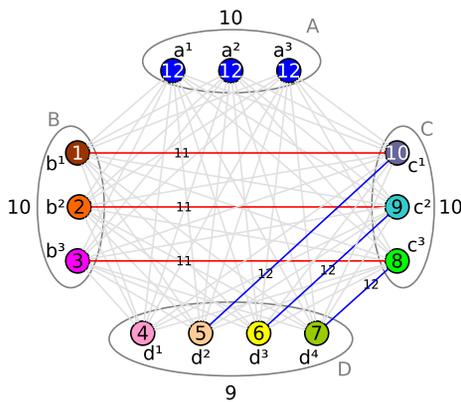


Figura 1: Grafo após passos do Lema 9

Essa é uma coloração TDV para o grafo G : os conjuntos de cores dos vértices de D se distinguem dos demais por sua cardinalidade; os conjuntos de cores dos vértices de C são os únicos que têm tanto a cor $\Delta(G) + 1$ quanto a cor $\Delta(G) + 2$; os conjuntos de cores dos vértices de B são os únicos que têm a cor $\Delta(G) + 1$ e não têm a cor $\Delta(G) + 2$; e os conjuntos de cores dos vértices de A têm a cor $\Delta(G) + 2$ tal como os conjuntos de cores dos vértices em D , mas têm cardinalidades diferentes.

Como existem vértices adjacentes com grau máximo, pelo Teorema 2, $\chi''_a(G) = \Delta(G) + 2$. \square

LEMA 10. *Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$. Se $|A| = |B| = |C| < |D| - 1$ com ordem par, então $\chi''_a(G) = \Delta(G) + 2$.*

PROVA. Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$ tal que $|A| = |B| = |C| < |D| - 1$ e o número de vértices é par. Note que neste caso $|A|$ e $|D|$ têm a mesma paridade. Insira um vértice a' em A e faça-o adjacente aos vértices de $B \cup C \cup D$. Faça dois emparelhamentos perfeitos M_{BC1} e M_{BC2} no subgrafo induzido $G[B \cup C]$. Pelo Lema 7, é possível aumentar o grau dos vértices de D injetando arestas entre seus vértices de forma que $G[D]$ seja $(|D| - |A| - 1)$ -regular. Seja $M = M_{BC1} \cup M_{BC2}$ e $G - M$ o subgrafo de G obtido pela remoção das arestas do conjunto M . Observe que agora A é o núcleo de $G - M$ e pelo Teorema 5, é possível colorir as arestas de $G - M$ com $\Delta(G)$ cores. Note que o número de vértices é ímpar, e em qualquer coloração de arestas de $G - M$, cada cor incide em no máximo $|V(G)| - 1$ vértices. Como foram usadas $\Delta(G)$ cores, cada uma delas falta em algum vértice de $G - M$. Como em $G - M$ existem $\Delta(G)$ vértices com grau igual a $\Delta(G) - 1$, em cada um deles falta uma cor. Então, as cores que faltam em dois vértices quaisquer de $G - M$ são duas a duas distintas. Pinte os vértices de A e as arestas do emparelhamento M_{BC1} com cor $\Delta(G) + 1$. Pinte o emparelhamento M_{BC2} com cor $\Delta(G) + 2$. Pinte os vértices de B, C e D com as cores das respectivas arestas que os conectam ao vértice a' e remova o vértice a' . Remova as arestas que foram injetadas entre os vértices de D . Essa é uma coloração TDV do grafo G : os conjuntos de cores dos vértices de D se distinguem dos conjuntos de cores dos vértices de A, B e C por sua cardinalidade; os conjuntos de cores dos vértices de A se distinguem dos demais pois não tem cor $\Delta(G) + 2$; os conjun-

tos de cores dos vértices de B se distinguem dos vértices de C pois $|V(G) + a'|$ é ímpar, portanto, as cores que sobraram em quaisquer dois desses vértices são distintas. Como existem vértices adjacentes com grau máximo, pelo Teorema 2, $\chi''_a(G) = \Delta(G) + 2$. \square

LEMA 11. *Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$. Se $|A| = |B| = |C| < |D| - 1$ com ordem ímpar, então $\chi''_a(G) = \Delta(G) + 2$.*

PROVA. Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$ tal que $|A| = |B| = |C| < |D| - 1$ e o número de vértices é ímpar. Note que neste caso $|A|$ e $|D|$ têm paridades distintas. Faça um emparelhamento perfeito M_{BC} no subgrafo induzido $G[B \cup C]$. Pelo Lema 7, é possível aumentar o grau dos vértices de D injetando arestas entre seus vértices de forma que $G[D]$ seja $(|D| - |A| - 1)$ -regular. Remova as arestas do emparelhamento M_{BC} . Observe que agora A é o núcleo de $G - M_{BC}$ e pelo Teorema 5, é possível colorir as arestas de $G - M_{BC}$ com $\Delta(G)$ cores. Observe que $|V(G)| = 3|A| + |D|$ é ímpar. Como o número de vértices é ímpar, em qualquer coloração de arestas de $G - M_{BC}$, cada cor incide em no máximo $|V(G)| - 1$ vértices. Como foram usadas $\Delta(G)$ cores, cada uma delas falta em algum vértice de $G - M_{BC}$. Como em $G - M_{BC}$ existem $\Delta(G)$ vértices com grau igual a $\Delta(G) - 1$, em cada um deles falta uma cor distinta. Então, as cores que faltam em dois vértices quaisquer de $G - M_{BC}$ são duas a duas distintas. Pinte os vértices de B, C e D com as cores que faltam nesses vértices. Pinte os vértices de A com cor $\Delta(G) + 2$ e o emparelhamento M_{BC} com cor $\Delta(G) + 1$. Faça um emparelhamento perfeito M_{CD} no subgrafo induzido $G[C \cup D]$ e pinte o emparelhamento M_{CD} com cor $\Delta(G) + 2$. Remova as arestas que foram injetadas entre os vértices de D . Essa é uma coloração TDV do grafo G : os conjuntos de cores dos vértices de D se distinguem dos conjuntos de cores dos vértices de A, B e C por sua cardinalidade; os conjuntos de cores dos vértices de A se distinguem dos conjuntos de cores dos vértices de B e C pois são os únicos que possuem a cor $\Delta(G) + 2$ e não possuem a cor $\Delta(G) + 1$; os conjuntos de cores dos vértices de C se distinguem dos vértices de B pois são os únicos que possuem tanto a cor $\Delta(G) + 1$ quanto a cor $\Delta(G) + 2$. Como existem vértices adjacentes com grau máximo, pelo Teorema 1, $\chi''_a(G) = \Delta(G) + 2$. \square

LEMA 12. *Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$. Se $|A| = |B| < |C| \leq |D|$, então $\chi''_a(G) = \Delta(G) + 2$.*

PROVA. Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$ tal que $|A| = |B| < |C| \leq |D|$. Faça um emparelhamento maximal M_{BD} no subgrafo induzido $G[B \cup D]$ e pinte as arestas de M_{BD} com a cor $\Delta(G) + 2$. Como $|B| < |D|$, existe um conjunto T de vértices de D onde não incidem arestas de M_{BD} . Pinte os vértices de T com a cor $\Delta(G) + 2$. Insira um vértice a' em A e faça-o adjacente aos vértices de $B \cup C \cup (D \setminus T)$. Faça um emparelhamento maximal M_{BC} no subgrafo induzido $G[B \cup C]$ e pinte as arestas desse emparelhamento com a cor $\Delta(G) + 1$. Como $|B| < |C|$ existe um conjunto S de vértices de C onde não incidem arestas do emparelhamento M_{BC} . Faça um emparelhamento maximal M_{SD} no subgrafo induzido $G[S \cup D]$ e pinte as arestas de M_{SD} com a cor $\Delta(G) + 1$. Seja $M = M_{BD} \cup M_{BC} \cup M_{SD}$ e $G - M$ o subgrafo de G obtido pela remoção das arestas do conjunto M . Observe

que A é o núcleo de $G - M$ e pelo Teorema 5, é possível colorir as arestas de $G - M$ com $\Delta(G)$ cores. Pinte os vértices de B , C e $D \setminus T$ com as cores das respectivas arestas que os conectam ao vértice a' e remova o vértice a' . Pinte os vértices de A com a cor $\Delta(G) + 1$.

Essa é uma coloração TDV do grafo G : como os conjuntos de cores dos vértices em B e D possuem a cor $\Delta(G) + 2$ eles se distinguem dos conjuntos de cores dos vértices de A e C ; os conjuntos de cores dos vértices de D se distinguem dos conjuntos de cores dos vértices de B por sua cardinalidade; os conjuntos de cores dos vértices de A têm cor $\Delta(G) + 1$ e não têm a cor $\Delta(G) + 2$ tal como o conjunto de cores dos vértices em C , mas têm cardinalidades diferentes.

Como existem vértices adjacentes com grau máximo, pelo Teorema 2, $\chi''_a(G) = \Delta(G) + 2$. \square

LEMA 13. *Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$. Se $|A| < |B| = |C| = |D|$, então $\chi''_a(G) \leq \Delta(G) + 2$.*

PROVA. Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$ tal que $|A| < |B| = |C| = |D|$. Insira um vértice a' em A e faça-o adjacente a todos os vértices das outras partes. Faça um emparelhamento perfeito M_{BC} no subgrafo induzido $G[B \cup C]$ e um emparelhamento perfeito M_{CD} no subgrafo induzido $G[C \cup D]$. Pinte as arestas dos emparelhamentos M_{BC} e M_{CD} com cores $\Delta(G) + 1$ e $\Delta(G) + 2$, respectivamente. Sejam $M = M_{BC} \cup M_{CD}$ e $G - M$ o subgrafo de G obtido pela remoção das arestas de M . Observe que A é o núcleo de $G - M$ e pelo Teorema 5, é possível colorir as arestas de $G - M$ com $\Delta(G)$ cores. Pinte os vértices de B , C e D com as cores das respectivas arestas que os conectam ao vértice a' e remova a' . Pinte os vértices de A com a cor $\Delta(G) + 1$.

Essa é uma coloração TDV do grafo G : os conjuntos de cores dos vértices de C são os únicos que têm tanto a cor $\Delta(G) + 1$ quanto a cor $\Delta(G) + 2$; os conjuntos de cores dos vértices de B se distinguem dos conjuntos de cores dos vértices de D pois os primeiros têm a cor $\Delta(G) + 1$ e os últimos têm a cor $\Delta(G) + 2$. Portanto, $\chi''_a(G) \leq \Delta(G) + 2$. \square

LEMA 14. *Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$. Se $|A| < |B| = |C| < |D|$, então $\chi''_a(G) \leq \Delta(G) + 2$.*

PROVA. Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$ tal que $|A| < |B| = |C| < |D|$. Insira um vértice a' em A e faça-o adjacente a todos os vértices das outras partes. Faça um emparelhamento perfeito M_{BC} no subgrafo induzido $G[B \cup C]$, e um emparelhamento maximal M_{CD} no subgrafo induzido $G[C \cup D]$. Pinte as arestas dos emparelhamentos M_{BC} e M_{CD} com cores $\Delta(G) + 1$ e $\Delta(G) + 2$, respectivamente. Como $|C| < |D|$, existe um conjunto S de vértices em D onde não incidem arestas do emparelhamento M_{CD} . Então, pinte os vértices de S com a cor $\Delta(G) + 2$. Sejam $M = M_{BC} \cup M_{CD}$ e $G - M$ o subgrafo de G obtido pela remoção das arestas de M . Observe que A é o núcleo de $G - M$ e pelo Teorema 5, é possível colorir as arestas de $G - M$ com $\Delta(G)$ cores. Pinte os vértices de B , C e D com as cores das respectivas arestas que os conectam ao vértice a' e remova a' . Pinte os vértices de A com a cor $\Delta(G) + 1$.

Essa é uma coloração TDV do grafo G : o conjunto de cores dos vértices de D e B não têm, respectivamente, as

cores $\Delta(G) + 1$ e $\Delta(G) + 2$; os conjuntos de cores dos vértices de C são os únicos que têm tanto a cor $\Delta(G) + 1$ quanto a cor $\Delta(G) + 2$; os conjuntos de cores dos vértices de A têm cor $\Delta(G) + 1$ tal como os conjuntos de cores dos vértices de B , mas têm cardinalidades diferentes. Portanto, $\chi''_a(G) \leq \Delta(G) + 2$. \square

LEMA 15. *Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$. Se $|A| < |B| < |C| = |D|$, então $\chi''_a(G) \leq \Delta(G) + 2$.*

PROVA. Seja G um grafo 4-partido completo com partição $\{A, B, C, D\}$ tal que $|A| < |B| < |C| = |D|$. Insira um vértice a' em A e faça-o adjacente a todos os vértices das outras partes. Faça um emparelhamento maximal M_{BD} no subgrafo induzido $G[B \cup D]$. Como $|B| < |D|$, existe um conjunto S de vértices em D onde não incidem arestas de M_{BD} . Pinte os vértices de S e as arestas do emparelhamento M_{BD} com a cor $\Delta(G) + 1$. Pinte os vértices de A com a cor $\Delta(G) + 2$. Observe que A é o núcleo de $G - M_{BD}$ e pelo Teorema 5, é possível colorir as arestas de $G - M_{BD}$ com $\Delta(G)$ cores. Pinte os vértices das partes B , C e D com as cores das respectivas arestas que os conectam ao vértice a' e remova a' .

Essa é uma coloração TDV do grafo G : os conjuntos de cores dos vértices de D se distinguem dos conjuntos de cores dos vértices de C pois possuem a cor $\Delta(G) + 1$; os conjuntos de cores dos vértices de A , B e C se distinguem entre si por suas cardinalidades. Portanto, $\chi''_a(G) \leq \Delta(G) + 2$. \square

Pelos lemas desta seção, vale o seguinte corolário.

COROLÁRIO 16. *Se G é um grafo 4-partido completo, então $\chi''_a(G) \leq \Delta(G) + 2$.*

4. CONCLUSÃO

Dos resultados apresentados, conclui-se que a Conjectura 1 é válida para os grafos 4-partidos completos. Pelos resultados do Teorema 4 e dos Lemas 9, 10, 11 e 12 conclui-se o seguinte teorema.

TEOREMA 17. *Se G é um grafo 4-partido completo com vértices adjacentes de grau máximo, então $\chi''_a(G) = \Delta(G) + 2$.*

Pelo Teorema 3 e pelos resultados conhecidos para k -partidos completos quando $k \in \{2, 3, 4\}$, propomos a seguinte conjectura.

CONJECTURA 18. *Se G é um grafo k -partido completo não isomorfo ao grafo completo K_k , então $\chi''_a(G) \leq \Delta(G) + 2$.*

5. REFERÊNCIAS

- [1] A. G. Luiz. Sobre a coloração total semiforte. Master's thesis, Unicamp, 2014.
- [2] I. R. Tiburcio and S. M. Almeida. Coloração total semiforte de grafos tripartidos completos, 2016. Trabalho de Conclusão de Curso.
- [3] V. G. Vizing. Critical graph with a given chromatic class. *Diskret. Analiz*, (5):6–17, 1965.
- [4] H. P. Yap. Total colourings of graphs. *Bulletin of the London Mathematical Society*, 21(2):159–163, 1989.
- [5] H. P. Yap. *Total colourings of graphs*. Springer, 2006.
- [6] Z. Zhang. On adjacent-vertex-distinguishing total coloring of graphs. *Sci China Ser A*, 48(3):289, 2005.

Assinatura Comportamental e Detecção de Anomalias Utilizando *K-means*

Wagner Senger
Universidade Tecnológica Federal do Paraná
Ponta Grossa-PR, Brasil
wagnersenger@gmail.com

Lourival Aparecido de Góis
Universidade Tecnológica Federal do Paraná
Ponta Grossa-PR, Brasil
gois@utfpr.edu.br

RESUMO

Neste artigo é descrito um método de caracterização do padrão de comportamento de um recurso computacional utilizando o algoritmo de clusterização *k-means*. Através da coleta dos dados de utilização dos componentes internos de um recurso por uma janela de tempo, um elemento denominado Assinatura Comportamental é criado, o qual neste artigo é gerada através da análise do *cluster* executada pelo algoritmo *k-means*. O referido modelo que ainda é um trabalho em andamento apresenta uma abordagem diferenciada de outros modelos onde outros algoritmos são aplicados no seu lugar ou além do próprio algoritmo são utilizadas técnicas complementares em conjunto. A expectativa é criar um formato mais simples e leve, que passe por uma única etapa de processamento, diferente de outros métodos que necessitam de duas ou mais para se determinar a Assinatura Comportamental.

Palavras-chave

k-means, clusterização, detecção de anomalias, padrões de comportamento, assinatura comportamental

ABSTRACT

In this paper we describe a method to characterize a computational resource behavior pattern using the clusterizing algorithm k-means. By collecting usage data from the resource's internal components in a time window, a element denominated Behavior Signature is created, that in this paper is generated through the cluster's analysis generated by the algorithm k-means. The current model is still a work in progress, and shows a different approach of others models where another algorithms are applied in his place or some other technics are used togheter it. The main focus is create a more simple and fast format, in a single step, differing from others who needs two or more steps to determine the Behavior Signature.

WPCCG '17 Outubro, 2017, Ponta Grossa, Paraná, Brasil

keywords

k-means, clusterizing, anomaly detection, behavior's pattern, bahavior's signature

1. INTRODUÇÃO

Áreas da computação que utilizam vários recursos computacionais com características diferentes como Grades Computacionais, Computação em Nuvem e Redes de Computadores entre outras, têm mudado a forma como estes equipamentos são utilizados. Seus funcionamentos proporcionam usos flexíveis de recursos e armazenamentos, permitindo que usuários evitem gastos excessivos com investimentos e manutenções [6].

Não somente nestas áreas, mas também sistemas que funcionam através de *Internet of Things* (IoT), a qual tem ganhado muito espaço pelo crescimento dos dispositivos móveis, precisam de abordagens que promovam o crescimento da segurança através de mecanismos de detecção de uso indevido de equipamentos, já que frequentemente são expostos a ambientes hostis e inseguros [9].

Dentro desta perspectiva se faz necessário conhecer como cada recurso se comporta durante um período normal de utilização. Para esta finalidade é necessário inferir um padrão de comportamento para que através dele então se possa detectar uma possível anomalia no seu uso, sendo ela causada simplesmente por uma requisição de uso acima do comum ou então por uma falha parcial ou total no equipamento.

Alguns mecanismos de estabelecimento de padrões e detecção de anomalias já têm sido avaliados e, neste artigo é proposta a criação de um novo mecanismo utilizando um algoritmo de clusterização que já é amplamente difundido principalmente na área de *Data Mining*, o *k-means* [10].

Este artigo está dividido em 3 seções, sendo a primeira abordando métodos correlatos de criação de padrões. Na segunda seção é demonstrada a metodologia proposta e a forma como o algoritmo é aplicado, bem como os resultados esperados, e na terceira as conclusões sobre o método.

2. TRABALHOS RELACIONADOS

Diferentes formas de identificação do padrão de comportamento de recursos já foram propostas, e em várias áreas elas têm suas aplicações testadas e validadas. É perceptível que este tema ainda instiga muitas outras pesquisas e aplicações, pois várias são as formas de se obter o resultado.

Em [5], os autores abordam três algoritmos para análise do fluxo de dados de rede dos recursos e criar o que foi chamado de "assinatura digital", a qual demonstra em um gráfico o

comportamento de um recurso ao longo do tempo. Os algoritmos utilizados foram o *Ant Colony Optimization* (ACO), *Holt-Winters*(HW) e *Principal Component Analysis*(PCA) e, após aplicados ao processo, avaliados os resultados. Os três algoritmos tiveram resultados muito próximos, e em parte do trabalho a complexidade de cada algoritmo e o volume de dados foi um diferencial entre eles, porém todos se mostraram suficientemente eficientes para serem aplicados em um ambiente de *cloud computing*.

Em [3] os mesmos algoritmos citados anteriormente foram testados na área de saúde e então houve uma diferença acentuada nos resultados, onde o PCA demonstrou uma maior efetividade tanto para caracterização do tráfego quanto para a detecção de anomalias.

Um trabalho que também utiliza clusterização como parte do método de identificação de anomalias foi descrito em [8], onde os autores utilizaram o algoritmo *k-means* para identificação de um padrão nos dados, porém, este foi aplicado em conjunto com um modelo denominado *Hidden Markov Model*(HMM), onde dados clusterizados pelo *k-means* eram fornecidos como entrada para este modelo, o qual calculava uma probabilidade e precisão para os dados fornecidos. Se a precisão calculada fosse maior que a dos dados já conhecidos então é decidido que os dados recebidos e o padrão conhecido combinam, sendo assim adicionados ao modelo de treinamento do algoritmo.

Em [1] a abordagem para determinar o padrão utiliza *Recurrent Neural Networks*(RNN), a qual é amplamente utilizada para detecção de padrões e, através de refinamentos sucessivos um padrão é estabelecido. Seu conceito considera que se sucessivos treinamentos geram um padrão refinado, o treinamento em dados já classificados pode reduzir o tempo necessário de treinamento, encaixando desta forma o RNN.

Na área de sensores sem fio, é proposto em [11] a criação de um módulo específico para detecção de anomalias, pois segundo o autor, em equipamentos que possuem poucos recursos de processamento, uma arquitetura de detecção de anomalias em tempo real pode necessitar de mais recursos do que o equipamento pode prover, prejudicando por sua vez o funcionamento normal do dispositivo.

A descoberta de padrões pode ser aplicada de várias maneiras em ambientes diferentes e ainda assim obter um resultado de alta qualidade, por isso, considerando estes estudos, neste artigo a abordagem proposta com o algoritmo *k-means* será aplicada de uma maneira diferenciada que será detalhada mais a frente, a fim de futuramente ser possível uma aplicação e testes de seu método.

3. METODOLOGIA

Durante o ciclo de vida de um recurso computacional, é comum que alterações na intensidade de uso ocorram, e mais que isso, ocorram de forma cíclica durante o tempo.

Em sistemas que funcionam em horário comercial por exemplo, desconsiderando processos externos como backup e movimentações de dados, entre outros, a maior carga de requisições recebidas se concentra dentro deste horário, havendo ainda uma possível diminuição durante o almoço. Com o tempo se torna possível prever que os recursos estarão com uma maior disponibilidade no almoço e fora do horário comercial, então, quando um recurso estiver com sua utilização fora desse padrão, existe a possibilidade de algo não estar em conformidade.

Ao se controlar o fluxo de dados de um equipamento, o

termo "anomalia do fluxo" dado a um período em que um ele se encontra fora do seu uso normal é bem difundida e cobre qualquer desvio das características do fluxo normal de um recurso [7], incluindo tráfego malicioso como ataques de *Distributed Denial-Of-Service* (DDoS), ou que utilize uma quantidade incomum da banda disponível ou ainda, o aumento natural da demanda sobre um determinado serviço.

Esta métrica de previsão de comportamento é nomeada normalmente como "tráfego normal" ou "uso comum" de um recurso, sendo também referida como Assinatura Digital em alguns estudos [5] [3]. Devido a esta última nomenclatura possuir uma referência muito próxima às assinaturas de documentos eletrônicos, neste documento ela será definida como Assinatura Comportamental.

A referida Assinatura Comportamental permite que através de uma representação gráfica um operador possa visualizar o comportamento de um recurso computacional ao longo do tempo e, além disso, permite que um sistema através dela possa determinar o que é um comportamento comum e o que é uma anomalia. Este comportamento se refere ao uso de cada componente do recurso, podendo ser ele de processamento, rede, memória, armazenamento, etc. A definição de uma Assinatura Comportamental para um recurso é importante para que se determine um padrão, e com o tempo seja possível prever como ele irá se comportar em um determinado horário de funcionamento.

Para determinar a Assinatura Comportamental do recurso computacional é necessário que os dados de utilização de seus componentes seja monitorada constantemente. A estratégia de monitoramento pode variar de acordo com a aplicação do método, podendo inserir os registros diretamente em um banco de dados ou simplesmente armazenando em um arquivo texto, sendo o intervalo de medição dos recursos variável de acordo com necessidade. Estes dados coletados são fornecidos como entrada para o algoritmo de clusterização *k-means*, o qual agrupa os valores e determina qual o padrão de utilização.

Como citado anteriormente, os recursos possuem uma utilização padrão para um determinado horário do dia, e além disso também possuem uma utilização diferenciada para cada dia da semana, podendo possuir um nível inferior em finais de semana ou feriados. Por isso é necessário que os dados avaliados sejam sempre comparados de acordo com o dia e horário em que ocorreram, por exemplo, o uso de memória de uma segunda-feira das 6h25m40s às 6h25m46s somente será comparada com este mesmo intervalo de tempo, evitando a comparação de horário com um final de semana. Nesta janela de tempo os dados coletados durante um período serão juntos analisados para se determinar o padrão.

Para o algoritmo *k-means* é necessário informar o número de *clusters* que se deseja gerar, isso é referenciado pela letra *k* do seu nome, que é uma variação inferida pela necessidade de cada aplicação [2]. Os dados enviados para análise são aqueles agrupados na janela de tempo citada acima, e o número de *clusters* ideais para a definição do padrão pode variar para cada situação, por isso testes de refinamento são necessários afim de aperfeiçoar a precisão.

Para exemplificar o processo, na Fig. 1 é demonstrado uma coleta fictícia de um recurso computacional, demonstrando a leitura do percentual de uso de um dos componentes, como o processador, em relação ao tempo. Separando uma janela de tempo dessa amostra para análise como demonstrado na Fig. 2 pela área demarcada em verde, é possí-

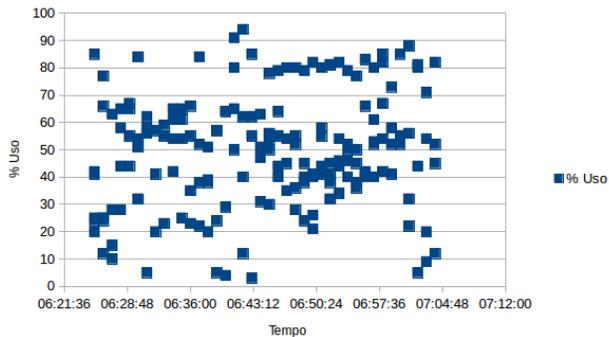


Figure 1: Coleta fictícia de dados do uso de um recurso.

vel perceber que a grande maioria dos valores encontram-se entre 30% e 65%.

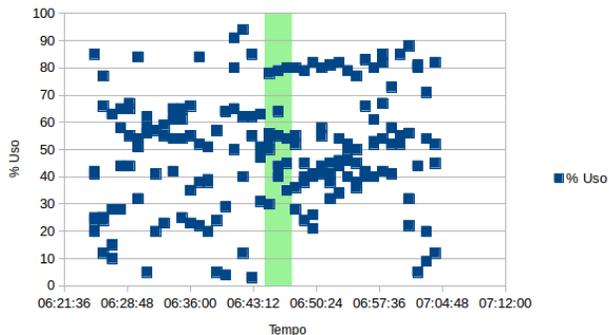


Figure 2: Janela de tempo da amostra e *cluster* com mais registros.

O algoritmo *k-means* recebendo os dados desta janela calcula um centroide para o número de *clusters* solicitados, e a partir dele os registros que possuem uma similar proximidade a ele passam a ser agrupados. A partir disto, sendo solicitado dois *clusters* ao algoritmo, o resultado de um procedimento de clusterização executado deve gerar os *clusters* como pode-se perceber na Fig. 3. Neste processo o que será levado em consideração como funcionamento padrão do recurso será o *cluster* com maior número de leituras registradas, no caso do exemplo abaixo o *cluster* vermelho possui grande parte dos registros, portanto será considerado como o padrão do recurso monitorado.

Tendo detectado o *cluster* com o funcionamento padrão do recurso pode-se passar para uma segunda etapa que é a de eliminação dos registros intermediários desta janela selecionada. Neste *cluster* haverá um registro em cada extremo, acima e abaixo, ou seja, permanecerão para análise o registro que estiver o percentual de uso mais alto e também o registro com percentual mais baixo dentro deste *cluster* selecionado, e estes limites irão determinar qual é a margem normal de funcionamento do recurso, sendo para o exemplo anterior de aproximadamente entre 30% e 65%.

Existe a possibilidade de algumas das leituras não serem incorporadas aos *clusters* gerados, dentro de um processo de clusterização estes elementos recebem o nome de *outliers*, ou ruídos, que são elementos que possuem uma natureza

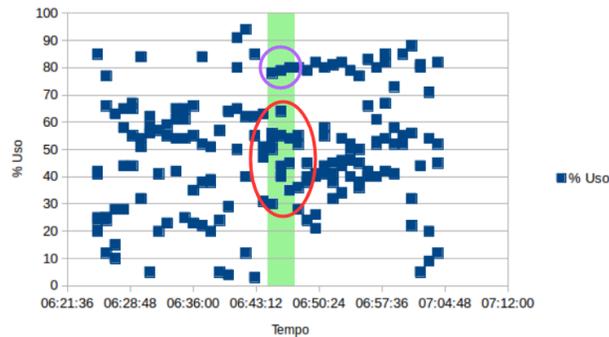


Figure 3: Clusters gerados na janela de tempo selecionada.

incerta, apresentam um comportamento inesperado ou então propriedades anormais [4], por isso não se enquadram em nenhum dos agrupamentos. Estas leituras são comuns, pois dentro do ciclo de operação de um recurso existem picos ou quedas de processamento momentâneas que podem ocorrer.

Após passar por todas as janelas do intervalo de tempo e processar os dados, serão identificados todos os registros de limite acima e abaixo dos *clusters*, que traçarão a margem de funcionamento normal do recurso computacional, ou no caso como definido anteriormente, a Assinatura Comportamental. Este resultado pode ser visto na Fig. 4.

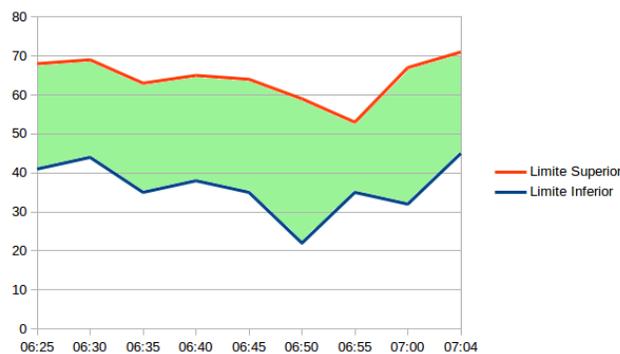


Figure 4: Assinatura Comportamental do recurso computacional.

Dentro da área de detecção de anomalias, graficamente esta Assinatura Comportamental pode servir como base para análise de um comportamento por um operador, porém para um sistema é necessário que seja levado em consideração um pouco mais do que simplesmente observar se uma leitura está dentro ou fora da margem de operação.

Como citado anteriormente, é possível que picos de execução apareçam no decorrer do ciclo de operação de um recurso, por isso não é correto avaliar uma leitura isoladamente, visto que até mesmo a Assinatura Comportamental é composta de vários registros de leituras.

Na Fig. 5 pode-se perceber que algumas das leituras não se enquadram na Assinatura Comportamental do recursos, porém isso não é motivo suficiente para determinar que um recurso está com seu funcionamento fora do padrão. Nesta figura existem dois pontos isolados fora da Assinatura, um laranja e outro preto, os quais representam picos naturais do

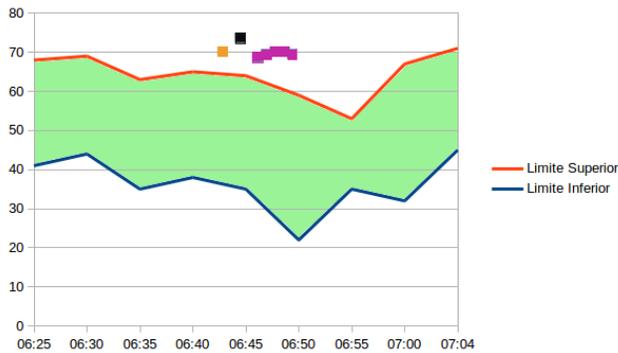


Figure 5: Assinatura Comportamental comparada a outliers.

processamento do recurso e neste sentido devem ser compreendidos apenas como ocorrências extraordinárias ao comportamento. Porém existe uma sequência de leituras em rosa que podem ser um indício de uma anormalidade no funcionamento do recurso, visto que estas além de estarem a mais tempo no mesmo nível de processamento, também atravessam mais de uma janela de leitura processada anteriormente.

Dentro desta abordagem é possível que um sistema analise o comportamento de um recurso, identifique e gere um padrão para o seu funcionamento natural e também faça análise em tempo real de alguma anomalia, o que permite que não somente problemas de aumento anormal de uso de recursos sejam analisados e reportados, como também quedas de equipamentos.

Não são somente anomalias acima da Assinatura Comportamental que são nocivas ao funcionamento de um sistema, a diminuição anormal do uso dos recursos pode denotar uma queda em algum serviço ou componente, fato que em determinados casos pode ser ainda mais impactante para os usuários.

4. CONCLUSÃO

Vários são os métodos dentro da detecção de anomalias e definição do comportamento padrão de recursos, e neste artigo se apresenta uma abordagem diferenciada, a qual oferece uma métrica para determinar uma área possível de utilização de um recurso e estabelecê-la como comum, através da Assinatura Comportamental, assim como a sua criação de um método de etapa única através da aplicação do algoritmo *k-means*.

Com esta Assinatura, se torna possível perceber mais claramente qual é a faixa em que o equipamento trabalha e também prever com base em um histórico anterior qual será a faixa em que ele estará em um determinado momento, possibilitando ações de melhoria ou ainda de incentivo de utilização de momentos em que o serviço se encontra com o uso em baixa.

Os resultados do modelo poderão ser avaliados através da averiguação da Assinatura Comportamental criada e a sua moldagem referente ao fluxo de uso normal do sistema, assim permitindo detectar momentos de utilização fora do padrão, e ainda averiguar se esta apresenta alertas que não condizem com uma utilização regular, ou seja, falsos positivos para uma anomalia. Poderá ser considerada como uma boa alternativa se os níveis de falso positivos forem inferiores

aos apresentados em outros estudos aplicados nesta área.

Apesar de testes ainda não terem sido executados com a metodologia, o algoritmo aplicado já é de ampla utilização em áreas como *Data Mining* e Grades Computacionais, portanto se espera um resultado satisfatório tanto de desempenho quanto de qualidade de resultados.

5. REFERÊNCIAS

- [1] A. Bhattacharyya, S. A. J. Jandaghi, S. Sotiriadis, and C. Amza. Semantic aware online detection of resource anomalies on the cloud. In *2016 IEEE 8th International Conference on Cloud Computing Technology and Science*. IEEE, January 2017.
- [2] P. S. Bradley and U. M. Fayyad. Refining initial points for k-means clustering. pages 91–99. Morgan kaufmann, 1998.
- [3] L. F. Carvalho, G. F. Jr., M. V. O. de Assis, J. J. P. C. Rodrigues, and M. L. P. Jr. Digital signature of network segment for healthcare environments support. In *15th International Conference On E-Health Networking, Application & Services*, pages 299–309. IEEE, October 2014.
- [4] F. Jiang, Y. Sui, and C. Cao. An information entropy-based approach to outlier detection in rough sets. *Expert Systems with Applications*, 37:6338–6344, September 2010.
- [5] M. L. P. Jr., G. F. Jr., and L. F. Carvalho. Digital signature to help network management using flow analysis. *International Journal of Network Management*, 26:76–94, May 2015.
- [6] C.-Y. Lin, Y.-A. Chen, Y.-C. Tseng, and L.-C. Wang. A flexible analysis and prediction framework on resource usage in public clouds. *4th IEEE International Conference on Cloud Computing Technology and Science Proceedings*, pages 309–316, February 2013.
- [7] J. Moraney and D. Raz. Efficient detection of flow anomalies with limited monitoring resources. In *2016 12th International Conference on Network and Service Management (CNSM)*. IEEE, January 2017.
- [8] Y. Ohno, M. Sugaya, A. van der Zee, and T. Nakajima. Anomaly detection system using resource pattern learning. In *2009 Software Technologies for Future Dependable Distributed Systems*. IEEE, January 2009.
- [9] H. Sedjelmaci, S. M. Senouci, and M. Al-Bahri. A lightweight anomaly detection technique for low-resource iot devices: A game-theoretic methodology. In *2016 IEEE International Conference on Communications (ICC)*. IEEE, May 2016.
- [10] P.-N. Tan, M. Steinbach, and V. Kumar. *Introduction to Data Mining*. Pearson Addison-Wesley, 2006.
- [11] M. Usman. Agent-enabled anomaly detection in resource constrained wireless sensor networks. In *Proceeding of IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks 2014*, Jun 2015.

Análise de Métricas para Recuperação de Casos visando à aplicação no Problema de Condução de Trens de Carga

Matheus Streisky
Universidade Tecnológica
Federal do Paraná

Av. Monteiro Lobato, s/n
Ponta Grossa – PR – Brasil
streisky@alunos.utfpr.edu.br

Richardson Ribeiro
Universidade Federal do Paraná
Departamento de Ciências Florestais,
Campus Jd. Botânico – Curitiba – PR
– Brasil

richardsonr@utfpr.edu.br

André Pinz Borges
Universidade Tecnológica
Federal do Paraná
Av. Monteiro Lobato, s/n
Ponta Grossa – PR – Brasil
apborges@utfpr.edu.br

RESUMO

Este artigo propõe uma análise de métricas de similaridade para o algoritmo k -NN, a ser usado na etapa de recuperação de casos no problema de condução de trens de carga. A condução de trens é uma tarefa complexa que exige experiência e conhecimento das ações a serem executadas para uma viagem econômica. O conhecimento pode ser obtido de históricos de viagens armazenadas e reutilizadas no formato de experiências passadas (casos). Para tal, pode-se utilizar a técnica de Raciocínio Baseado em Casos (RBC), onde uma das etapas é a recuperação destas experiências. Na recuperação, métricas são usadas para avaliar qual dos casos passados mais se assemelham ao problema atual. Este artigo efetua uma análise de diferentes métricas de recuperação k -NN que podem ser usadas na etapa de recuperação. Dentre elas, a Distância de *Minkowski* mostrou-se eficiente para o problema estudado.

Palavras-chave

Raciocínio Baseado em Casos; k -NN; Condução de trens.

ABSTRACT

This article proposes an analysis of similarity metrics for k -NN algorithm, to be used in the case retrieval stage in the problem of conducting freight trains. Driving trains is a complex task that requires experience and knowledge of the actions to be performed for an economic trip. Knowledge can be obtained from stored travel histories and reused in the form of past experiences (cases). For this, the technique of Case-Based Reasoning (CBR) can be used, where one of the steps is the recovery of these experiences. In recovery, metrics are used to assess which of the past cases most resemble the current problem. This article examines the different k -NN recovery metrics that can be used in the recovery step. Among them, Minkowski's distance proved to be efficient for the problem studied.

Keywords

Case-based Reasoning; k -NN; Train driving.

1. INTRODUÇÃO

O Raciocínio Baseado em Casos (RBC) é um método de Inteligência Artificial que utiliza um conhecimento prévio adquirido em problemas passados para reaproveitar soluções bem sucedidas e evitar que falhas se repitam [1].

Em RBC descreve-se o conhecimento armazenado em memória como um caso. O caso consiste em representar um episódio em que um problema foi total ou parcialmente resolvido [6]. Em

outras palavras, caso é um acontecimento problemático da vida de um ator ou objeto que pode ter sido solucionado ou não.

O ciclo do RBC é composto por 4 etapas principais (cf. Figura 1). Dado um novo problema, a partir da percepção de sensores de um ambiente, o objetivo é encontrar uma nova solução com uma consulta na base de casos (composta de vários casos) onde buscase um ou mais casos semelhantes ao novo problema (geralmente os casos mais semelhantes são os melhores para o mesmo problema). É recuperada então, da base de casos, uma ou mais soluções para serem reutilizadas. Quando for necessário, é realizada uma adaptação (reuso) no caso escolhido para tentar solucionar o problema, visto que cada caso recuperado pode não ser completamente igual para todas as etapas do problema. Após a adaptação, o novo caso deve ser revisado para verificar se pode ser aplicado e, se resolver o problema de forma adequada, o caso é aplicado e retido na base de casos. A retenção dá-se para ajudar na resolução de problemas futuros. Se a solução adaptada não for aplicável, o caso deve ser novamente adaptado, modificando o caso [6].

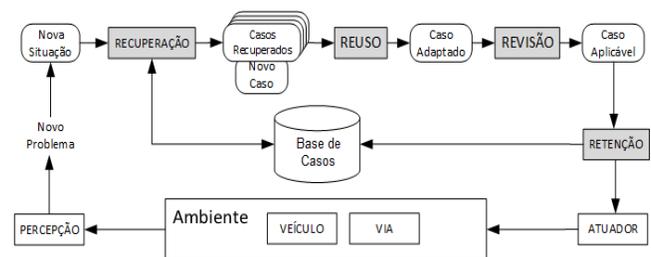


Figura 1 - Ciclo CBR [5].

Descobrir os casos com maior potencialidade é essencial para o sucesso de um sistema que usa RBC [6]. Logo, na etapa de recuperação de casos são usadas heurísticas e avaliações na seleção do caso mais semelhante, com o objetivo de diminuir o processo de busca. Esta etapa é importante no método RBC, pois filtra os casos mais semelhantes a um determinado problema em uma base de casos com n elementos [1].

É fundamental ter uma quantidade razoável de elementos registrados em uma base de casos, pois quanto mais casos estiverem disponíveis para recuperação, mais preciso será o resultado obtido nesta etapa. Bases de casos pequenas podem irão gerar resultados precisos, já que haverá uma gama de comparações menores a serem feitas ocorrendo, no pior caso, a recuperação de um resultado não satisfatório para o problema.

Entretanto, devido às heurísticas e avaliações necessárias na etapa

de recuperação de casos, uma base de casos grande traz consigo maior demanda de tempo e processamento, o que em algumas situações é inaplicável. Um dos desafios de usar algoritmos de recuperação eficientes é ter um bom desempenho quando aplicado a centenas ou milhares de casos sem diminuir sua precisão [8].

A recuperação também deve ser precisa e eficiente porque a mesma afeta as demais etapas presentes no método RBC. Para aplicar o reuso de casos, o caso recuperado passa por um processo de adaptação, modificando-o a fim de satisfazer o problema atual. Se o caso encontrado diferir do problema a ser solucionado, o processo de adaptação se torna demorado, pois é necessário “corrigir” as diferenças entre o caso e o problema.

Neste artigo serão analisadas métricas que podem ser usadas em conjunto com o algoritmo *k-NN* (*k-Nearest Neighbor*) utilizado na etapa de Recuperação. Tais métricas são usadas para calcular a distância, ou semelhança, entre o problema atual e os elementos de uma base de casos, onde *k* é o número de casos comparados ao problema atual. A distância de similaridade é uma medida numérica de quão parecidos dois casos/objetos são, sendo maior quando os casos/objetos são parecidos. O inverso é a medida de dissimilaridade, onde quanto maior seu valor, maior é a diferença entre dois casos/objetos [9].

Assim como mencionado por Zia *et al* (2014), *k-NN* é um dos algoritmos de recuperação mais difundidos pela literatura, pois possui uma fácil implementação e é menos susceptível a dados com ruídos. Dentre diversas técnicas de similaridade, a mais utilizada é a Distância Euclidiana para realizar a seleção dos vizinhos mais próximos. Porém, técnicas como a Distância do Cosseno e a Distância de *Mahalanobis* também podem ser usadas.

O presente artigo está organizado da seguinte forma. Na seção 2 há menções de trabalhos relacionados com métricas de avaliação e utilização do algoritmo *k-NN*. A seção 3 mostra a metodologia utilizada na realização deste artigo, analisando a importância da recuperação de casos e as vantagens e desvantagens de cada uma das métricas selecionadas. Logo em seguida, temos na seção 4 os resultados obtidos pela implementação destas métricas. Por fim na seção 5, as considerações finais e os trabalhos futuros.

2. TRABALHOS RELACIONADOS

RBC tem sido empregado em diversas pesquisas recentes. Vasconcelos (2016) comparou algoritmos de junção por similaridade em memória, utilizando o algoritmo *k-NN* para realizar consultas de similaridade entre dados multimídia (áudio, vídeo, imagem, textos longos, etc). Para calcular as similaridades foram utilizadas distâncias da família de *Minkowski* (Distância Euclidiana e de *Manhattan*). As consultas serviram para executar um filtro nas diversas informações contidas no banco de dados e recuperar os elementos mais similares entre si, para posteriormente executar um algoritmo de junção.

Na área de jogos, Bricce *et al* (2016) aplicou o algoritmo *k-NN* (aplicando Distância Euclidiana) para controlar os movimentos de *non-players characters* (NPCs) em um ambiente dinâmico, armazenando as jogadas do usuário para realizar o aprendizado de máquina. Isto permitiu que o NPC se torne mais inteligente de acordo com o tempo jogado pelo usuário, e essas informações armazenadas são recuperadas posteriormente para aplicar na jogabilidade do NPC.

Na condução de trens de carga, RBC é usado como uma técnica capaz de recuperar ações que maquinistas devem executar para

percorrer determinados trechos de via, conforme descrito em Borges (2015). A técnica mostrou-se eficaz pois o compartilhamento das experiências dos maquinistas para conduzir um trem não é algo trivial de ser compartilhado e explicado. Logo, a partir de um conjunto de dados obtidos por sensores de locomotivas, é criada uma base de casos. A base é submetida ao ciclo de RBC, que retorna ações aplicáveis durante a condução.

Contudo, a eficácia de RBC passa pela recuperação de um conjunto de casos próximos aos dados lidos dos sensores. Se os casos recuperados forem distantes do problema atual, o esforço da etapa de adaptação aumenta. Sendo assim, a principal contribuição deste trabalho é analisar algumas métricas de similaridade a serem usadas na etapa de recuperação de casos, para verificar qual métrica melhor se encaixa no problema da condução de trens de carga.

3. METODOLOGIA

O que pode influenciar se a recuperação de casos será boa ou ruim é a escolha da métrica de similaridade aplicada nesta etapa. Esta escolha impacta em todo o sistema RBC. Na literatura não há um consenso sobre uma medida de similaridade que seja aplicável a todos os tipos de variáveis que podem existir numa base de casos. Os pesquisados empregam medidas variadas, como: Distância Euclidiana, Distância Euclidiana Ponderada, Distância de *Manhattan* e Distância de *Minkowski*. Logo, tais medidas serão analisadas neste trabalho para uso futuro, sendo importante ressaltar que algumas medidas necessitam que os dados sejam normalizados antes da aplicação. A normalização consiste em colocar todas as variáveis que compõem um caso/objeto em uma mesma escala [9].

3.1 Distância Euclidiana

É uma medida de distância frequentemente empregada quando as variáveis são quantitativas [11]. Sua aplicação é recomendada para dados não padronizados, porém há uma desvantagem se houverem dados com diferença de escala (cm e km, por exemplo). No momento da transformação os resultados euclidianos sofrem uma influência das dimensões que possuem valores maiores. Apesar de ser uma distância de fácil entendimento e útil para aplicações de recuperação, ela não varia em um intervalo de 0 a 1 e, então, o valor obtido não é comparável aos valores alcançados por outras medidas de similaridade que variam entre 0 e 1. O cálculo desta distância é mostrado na Equação 1, onde *x* e *y* representam os atributos dos casos a serem comparados.

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

Um exemplo de utilização desta métrica é para calcular a rota de GPS de um helicóptero. Por se tratar de um veículo aéreo apenas um segmento de reta entre 2 pontos indica a distância a ser percorrida. Supondo que os pontos (4, 1) e (3, 7) representem *x* e *y* (origem e destino) respectivamente, realizando os cálculos tem-se a distância igual a 6,082.

3.2 Distância Euclidiana Ponderada

A Distância Euclidiana Ponderada é semelhante ao da Distância Euclidiana comum, porém com adição de pesos nas características dos dados relativos à sua importância. O cálculo é mostrado na Equação 2, onde *x* e *y* são os atributos dos casos a serem comparados e *w* é o peso atribuído à essas características.

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

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

$$d1 = \sqrt{5 \cdot (5 - 2)^2 + 5 \cdot (5 - 7)^2} = \sqrt{65} = 8,062$$

$$d2 = \sqrt{1 \cdot (5 - 2)^2 + 1 \cdot (5 - 7)^2} = \sqrt{13} = 3,605$$

3.3 Distância de Manhattan

A Distância de *Manhattan*, possui funcionamento similar à Distância Euclidiana, mas não é influenciado pela diferença de escala sobre o resultado já que não há elevação ao quadrado das características a serem medidas.

Mesmo sendo mais fácil de calcular que as distâncias citadas anteriormente, esta métrica pode não ser adequada se os atributos estão correlacionados, porque não existe a garantia da qualidade dos resultados obtidos [4]. Para calcular a Distância de *Manhattan* utiliza-se da Equação 3, onde x e y representam os atributos dos casos a serem comparados.

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

Para exemplificar, uma pessoa pega um taxi no centro de uma cidade no ponto $x(1, 1)$ e deseja ir até o ponto $y(3, 4)$. Calculando, obtemos a seguinte distância percorrida igual a 5.

3.4 Distância de Minkowski

Segundo Linden (2009), outra medida de distância utilizada é a Distância de *Minkowski*, que é uma generalização das distâncias Euclidiana e de *Manhattan*. A Distância de *Minkowski* pode ser calculada através da Equação 4, onde x e y representam os atributos dos casos a serem comparados e q é um inteiro positivo que, caso assuma valor igual a 1 resulta na Distância de *Manhattan*, caso assuma valor igual a 2 resulta na Distância Euclidiana e caso assuma valor igual a infinito resulta na Distância de *Chebyshev*, sendo esta última dada pela Equação 5.

$$d = \left(\sum_{i=1}^n |x_i - y_i|^q \right)^{\frac{1}{q}} \quad (4)$$

$$d = \max_k |x_k - y_k| \quad (5)$$

A escolha do valor de q depende exclusivamente da ênfase que se deseja dar a distâncias maiores. Quanto maior for o valor de q , maior será a sensibilidade da métrica. Um exemplo simples é quando se calcula a distância entre os pontos $x(0, 2, 1)$ e $y(1, 18, 2)$. Usando diferentes valores de q , temos os seguintes valores para as distâncias, conforme Tabela 1.

Para aplicação do algoritmo k -NN, primeiro é necessário definir o valor de k . Recomenda-se que k possua um valor ímpar, assim a ocorrência de empates se tornam mais difícil de ocorrer. Será assumido os valores de $k=3$ e $k=5$ para um problema fictício a ser solucionado com valores iguais a [5.6, 3.7, 5.2, 0.4].

Tabela 1 – Cálculos das distâncias de Minkowski.

Valor de q	Cálculo
$q=1$	$d = [1 - 0 + 18 - 2 + 2 - 1] = 18$
$q=2$	$d = \sqrt{(1 - 0)^2 + (18 - 2)^2 + (2 - 1)^2} = 16,06$
$q=3$	$d = \sqrt[3]{(1 - 0)^3 + (18 - 2)^3 + (2 - 1)^3} = 16,002$
$q=\infty$	$d = \sqrt[\infty]{(1 - 0)^\infty + (18 - 2)^\infty + (2 - 1)^\infty} = 16$

Estes valores contêm informações correspondentes a: velocidade, pressão dos freios, peso total dos vagões (em toneladas) e inclinação (subida, descida ou reta). Para aplicação das métricas no algoritmo k -NN, será utilizado uma base de casos contendo 150 elementos com valores simuladas de condução de trens de carga. Na utilização da Distância Euclidiana e da Distância de *Manhattan*, não há outros fatores a serem declarados, apenas a aplicação normal de suas fórmulas. Já a Distância Euclidiana Ponderada, será assumido os seguintes valores de w (peso) em cada uma das características dos casos: 4 para velocidade, 5 para a pressão dos freios, 3 para peso dos vagões e 1 para inclinação. Os valores foram atribuídos apenas para teste de forma empírica. Na Distância de *Minkowski*, q (grau de sensibilidade) assumirá valor igual ao infinito, pois se aplicado valores iguais a 1 ou 2, não serão obtidos resultados diferentes da Distância Euclidiana ou da Distância de *Manhattan*.

4. RESULTADOS

As tabelas 2 e 3 mostram os resultados do algoritmo k -NN para cada uma das métricas analisadas, onde d é o valor de dissimilaridade entre o problema a ser solucionado e os casos recuperados da base de casos.

Conforme possível analisar, os 3 primeiros casos recuperados não diferem quando o k assume valor igual a 3 ou igual a 5 em nenhuma das métricas aplicadas no algoritmo. Entretanto isso não significa que um valor maior que 3 não deve ser aplicado à k , pois se na pior das hipóteses nenhum dos 3 primeiros casos satisfizerem o problema, há mais opções de casos para reuso.

Também é possível analisar que mesmo métricas diferentes recuperaram casos iguais, porém em diferentes prioridades. A Distância Euclidiana Ponderada recupera o caso [6.0, 3.4, 4.5, 1.6] como o mais similar ao problema proposto, assim como a Distância de *Manhattan* recupera o mesmo caso como o terceiro mais similar. O mesmo ocorre no caso [6.1, 2.8, 4.7, 1.2], onde a Distância Euclidiana o indica como mais similar e a Distância de *Minkowski* o indica como o segundo mais similar. Para uma visão mais geral do problema, na Tabela 4 são representadas as médias de similaridade de cada métrica aplicada no problema.

A Distância de *Minkowski* obteve as menores médias de dissimilaridade entre todas as métricas analisadas, seguidas das distâncias Euclidiana, Euclidiana Ponderada e de *Manhattan*. Sendo assim, *Minkowski* se mostra como a métrica mais precisa e *Manhattan* como a menos precisa, para resolução deste problema dentre as quatro métricas investigadas.

5. CONSIDERAÇÕES FINAIS

O método RBC é uma ferramenta versátil e pode ser aplicada para diversas funcionalidades, sendo utilizada também em outras áreas fora a da computação, podendo ser moldada de acordo com as necessidades de quem a desenvolve.

Tabela 2 - Resultados obtidos com k=3.

Métrica	k = 3	
	Casos	d
Distância Euclidiana	[6,1; 2,8; 4,7; 1,2]	1,396
	[6,1; 3,0; 4,6; 1,4]	1,449
	[5,7; 3,0; 4,2; 1,2]	1,449
Distância Euclidiana Ponderada	[6,0; 3,4; 4,5; 1,6]	2,000
	[5,9; 3,0; 5,1; 1,8]	2,190
	[5,4; 3,0; 4,5; 1,5]	2,300
Distância de Manhattan	[5,6; 3,0; 4,5; 1,5]	2,500
	[5,9; 3,0; 5,1; 1,8]	2,500
	[6,0; 3,4; 4,5; 1,6]	2,600
Distância de Minkowski	[5,7; 2,8; 4,5; 1,3]	0,900
	[6,1; 2,8; 4,7; 1,2]	0,900
	[6,4; 2,9; 4,3; 1,3]	0,900

Tabela 3 - Resultados obtidos com k=5.

Métrica	k = 5	
	Casos	d
Distância Euclidiana	[6,1; 2,8; 4,7; 1,2]	1,396
	[6,1; 3,0; 4,6; 1,4]	1,449
	[5,7; 3,0; 4,2; 1,2]	1,449
	[5,4; 3,0; 4,5; 1,5]	1,493
	[5,7; 2,9; 4,2; 1,3]	1,568
Distância Euclidiana Ponderada	[6,0; 3,4; 4,5; 1,6]	2,000
	[5,9; 3,0; 5,1; 1,8]	2,190
	[5,4; 3,0; 4,5; 1,5]	2,300
	[6,1; 3,0; 4,6; 1,4]	2,351
	[6,0; 3,0; 4,8; 1,8]	2,351
Distância de Manhattan	[5,6; 3,0; 4,5; 1,5]	2,500
	[5,9; 3,0; 5,1; 1,8]	2,500
	[6,0; 3,4; 4,5; 1,6]	2,600
	[5,6; 3,0; 4,1; 1,3]	2,700
	[5,5; 2,6; 4,4; 1,2]	2,799
Distância de Minkowski	[5,7; 2,8; 4,5; 1,3]	0,900
	[6,1; 2,8; 4,7; 1,2]	0,900
	[6,4; 2,9; 4,3; 1,3]	0,900
	[6,2; 2,9; 4,3; 1,3]	0,900
	[5,6; 2,7; 4,2; 1,3]	1,000

Tabela 4 - Médias de dissimilaridade.

Métrica	k = 3	k = 5	Média (k=3 e k=5)
Distância Euclidiana	1,431	1,471	1,451
Distância Euclidiana Ponderada	2,163	2,238	2,201
Distância de Manhattan	2,533	2,620	2,577
Distância de Minkowski	0,900	0,920	0,910

A recuperação de casos é uma etapa importante em todo método RBC, pois esta etapa pode determinar se o problema enfrentado terá uma boa solução ou não. Implementando uma boa recuperação, a etapa de adaptação, se ocorrer, necessitará de menor esforço computacional e isso resultará em um método rápido e otimizado.

As métricas de dissimilaridade aplicadas no problema também impactam no desempenho do método, por mais que as diferenças entre as métricas analisadas neste trabalho sejam pequenas (uma diferença de 1,667 pontos de dissimilaridade entre a melhor e a pior métrica), em bancos de casos com milhares ou milhões de casos, tais métricas podem apresentar uma diferença significativa de dissimilaridade. É tarefa do desenvolver escolher a métrica a ser empregada em seu algoritmo, adotando a qual se encaixar melhor para solucionar o problema enfrentado.

Como este artigo propõe uma análise de diferentes métricas voltadas ao algoritmo *k-NN*, é interessante também examinar, em trabalhos futuros, outras métricas de dissimilaridade não citadas neste artigo no mesmo algoritmo. Outro trabalho futuro é analisar tais métricas utilizando outros algoritmos de recuperação existentes, os aplicando ao problema de condução de trens.

6. REFERÊNCIAS

- [1] Aamodt, A.; Plaza, E. Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. In: AICOM – Artificial Intelligence Communications. Vol. 7. Ed. IOS Press, EUA, 1994.
- [2] Borges, A., P. Uma contribuição para geração de políticas de ações para condução de trens de carga usando Raciocínio Baseado em Casos. 2015. 220 f. Dissertação (Doutorado) – PUC-PR. Curitiba, 2015.
- [3] Bricce, B. R., Silva, P. P., Martins, H. P., Silva, E. G. Aplicação do algoritmo knn para controle de movimentos de npcs em um ambiente dinâmico (jogo). REGRAD - Revista Eletrônica de Graduação do UNIVEM - ISSN 1984-7866, [S.l.], v. 9, n. 1, p. 18-32, Aug. 2016. ISSN 1984-7866.
- [4] Cole, R., M. Clustering with Genetic Algorithms, M. Sc., Department of Computer Science, University of Western Australia, Australia, 1998.
- [5] Corchado, J. M., Lees, B., Aiken, J. Hybrid instance-based system for predicting ocean temperature. International Journal of Computational Intelligence and Applications, v. 1, n. 1, p. 35-52, 2001.
- [6] Kolodner, J. Reconstructive memory: a computer model. Cognitive Science 7(4), 1983 apud KOLODNER 1993.
- [7] Linden, R. Técnicas de Agrupamento. Revista de Sistemas de Informação de FSMA. n. 4, pp. 18-36, 2009.
- [8] Abel, M. Um estudo sobre Raciocínio Baseado em Casos. 1996. 41 f. Dissertação (Pós-Graduação em Ciência da Computação) – UFRGS. Porto Alegre, 1996.
- [9] Mitchell, T. Machine Learning. McGraw Hill, 1997.
- [10] Vasconcelos, G. Q. Estudo comparativo de algoritmo de junção por similaridade em memória. 2016. 57 f. Dissertação (Graduação) – Curso de Bacharelado em Ciência da Computação, UEL, Londrina, 2016.
- [11] Zia, S. S., Akhtar, P., Mughal, T. J., & Mala, I. Case Retrieval Phase of Case-Based Reasoning Technique for Medical Diagnosis. World Applied Sciences Journal, 32(3), 451-458. 2014.

On a conjecture on edge-colouring join graphs

Leandro M. Zatesko^{1 2 * †}Renato Carmo^{2 *}André L. P. Guedes^{2 *}

¹ Federal University of Fronteira Sul
Chapec , Brazil
leandro.zatesko@uffs.edu.br

² Department of Informatics
Federal University of Paran 
Curitiba, Brazil
{renato, andre}@inf.ufpr.br

ABSTRACT

We bring new insights on a recent conjecture on edge-colouring. The conjecture states that if G is the disjoint join of two graphs with same order and same maximum degree such that the vertices of maximum degree of one of them induce an acyclic graph, then G is $\Delta(G)$ -edge-colourable. We present a polynomial-time heuristic for $\Delta(G)$ -edge-colouring such graphs. Our algorithm may fail in a specific case, and we conjecture that it is always possible to handle this case.

Keywords

Edge-colouring; Join graphs; Recolouring procedures

1. INTRODUCTION

Computing the chromatic index of an n -order graph G is an NP-hard [3] problem, but a conjecture proposed in 1984 [1] would imply, if proved, that the problem can be solved in linear time if $\Delta(G) \geq n/2$ [7], as all join graphs have. Linear-time algorithms for the chromatic index of join graphs are known only for some cases and the problem remains open even restricted to disjoint joins of graphs with same order and same maximum degree [5, 6, 8, 9, 10, 11, 13].

Let G be the join of two disjoint graphs G_1 and G_2 with same order and same maximum degree. In [5] the authors conjectured that $\chi'(G) = \Delta(G)$ whenever G_1 and G_2 are cographs and $\Lambda[G_1]$ is acyclic (see definitions in the sequel). The authors showed that their conjecture holds when $|V(G_2) \setminus V(\Lambda[G_2])| \geq |V(\Lambda[G_1])|$. This conjecture was extended for the case wherein G_1 and G_2 are not required to

be cographs [13], and in the same paper proved under

$$|V(G_2) \setminus V(\Lambda[G_2])| \geq |\{u \in V(\Lambda[G_1]) : d_{\Lambda[G_1]}(u) > 1\}| + |\{C \text{ connected component of } \Lambda[G_1] : |V(C)| = 2\}|.$$

We provide a partial proof for the latter conjecture (and consequently for the former), not imposing $V(G_2) \setminus V(\Lambda[G_2]) \neq \emptyset$. Our proof can be viewed as a polynomial-time heuristic to obtain a $\Delta(G)$ -edge-colouring. We use the term *heuristic* since there is a very specific case in which we do not know yet how to proceed. We identify this case and we conjecture that it can always be handled in polynomial time.

This paper is organised as follows. The remaining of this section provides some of the definitions used. Section 2 presents the above-mentioned proof and other accessory results. Section 3 concludes with remarks for future works.

Preliminary definitions

In this work, a *graph* is a simple graph, that is, an undirected loopless graph with no multiple edges. The *degree* of a vertex u in a graph G is denoted by $d_G(u) := |N_G(u)| = |\partial_G(u)|$, wherein $N_G(u)$ denotes the set of the neighbours of u in G and $\partial_G(u)$ denotes the set of the edges incident to u in G . Also, for any $X \subseteq V(G)$, we denote by $\partial_G(X)$ the *cut* defined by X in G , i.e. the set of the edges of G with exactly one endpoint in X . At last, we denote by $\Lambda[G]$ the *core* of G , i.e. the subgraph of G induced by all its vertices of maximum degree. Other graph-theoretical definitions follow their usual meanings and notation found in the literature.

A *t-edge-colouring* of G is a function $\varphi: E(G) \rightarrow \mathcal{C}$ such that $|\mathcal{C}| = t$ and adjacent edges have different images (or *colours*) assigned by φ . We say that a vertex u *miss* some colour $\alpha \in \mathcal{C}$, and that α is *missing* at u , if no edge incident to u is coloured with α . The *chromatic index* $\chi'(G)$ of G is the least t for which G is t -edge-colourable. In 1964, Vizing showed that either $\chi'(G) = \Delta(G)$ or $\chi'(G) = \Delta(G) + 1$ for every graph G [12], in which case G is said to be respectively *Class 1* or *Class 2*. Vizing's proof is based on a polynomial-time recolouring procedure under which a $(\Delta(G) + 1)$ -edge-colouring of G can be constructed edge by edge. The same procedure can be used to show that every graph with acyclic core is *Class 1*, colouring the edges in a convenient order [2].

The *join* of two graphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, denoted by $G_1 * G_2$, is given by $V(G_1 * G_2) := V_1 \cup V_2$ and $E(G_1 * G_2) := E_1 \cup E_2 \cup \{v_1 v_2 : v_1 \in V_1 \text{ and } v_2 \in V_2\}$. A *join graph* is a K_1 graph or the result of the join of two disjoint

*Partially supported by CNPq, Proc. 428941/2016-8.

†Partially supported by UFFS, Proc. 23205.001243/2016-30.

graphs. The K_1 is considered a join graph because, in a join operation $G_1 * G_2$, the graphs are not required to be disjoint. Nevertheless, as pointed by [13], if $G_1 * G_2 \neq K_1$, then there are disjoint graphs G'_1 and G'_2 such that $G_1 * G_2 = G'_1 * G'_2$.

2. THE HEURISTIC

We start with a proposition on general graphs. This proposition is used in the proof of Theorem 1, the main theorem which completes the presentation of our heuristic. Precede the theorem two lemmas and a conjecture, which discuss a recolouring procedure on which our heuristic is based.

PROPOSITION 1. *If a graph G has maximum degree $\Delta > 1$ and an acyclic core with s vertices, then G has at least*

$$\max\left\{\Delta - 1, s - \left\lfloor \frac{s-2}{\Delta-1} \right\rfloor\right\}$$

vertices of degree less than Δ .

PROOF. Consider the set

$$X := \{(u, e) : u \in V(\Lambda[G]) \text{ and } e \in \partial_G(u)\}.$$

As each $u \in V(\Lambda[G])$ appears in X exactly $d_G(u) = \Delta$ times, we have $|X| = s\Delta$. On the other hand, in view of $\bigcup_{u \in V(\Lambda[G])} \partial_G(u) = E(\Lambda[G]) \cup \partial_G(V(\Lambda[G]))$, each edge of $E(\Lambda[G])$ appears in X exactly twice, whereas each edge of $\partial_G(V(\Lambda[G]))$ appears in X exactly once, which brings $|X| = 2|E(\Lambda[G])| + |\partial_G(V(\Lambda[G]))|$. Since $\Lambda[G]$ is acyclic, $|E(\Lambda[G])| \leq s - 1$. Therefore, $|\partial_G(V(\Lambda[G]))| \geq s(\Delta - 2) + 2$.

Because $|\partial_G(u) \cap \partial_G(V(\Lambda[G]))|$ is at most s and at most $\Delta - 1$ for all $u \in V(G) \setminus V(\Lambda[G])$, and since $|\partial_G(V(\Lambda[G]))| = \sum_{u \in V(G) \setminus V(\Lambda[G])} |\partial_G(u) \cap \partial_G(V(\Lambda[G]))|$, we have $(n - s)s \geq s(\Delta - 2) + 2$ and $(n - s)(\Delta - 1) \geq s(\Delta - 2) + 2$, implying

$$n - s \geq \left\lceil (\Delta - 2) + \frac{2}{s} \right\rceil \geq \Delta - 1 \quad \text{and}$$

$$n - s \geq \left\lceil s - \frac{s-2}{\Delta-1} \right\rceil = s - \left\lfloor \frac{s-2}{\Delta-1} \right\rfloor,$$

since $n - s$ is an integer and $s > 0$. \square

The lower bound of Proposition 1 is tight, being the diamond $(K_2 * \overline{K_2})$ an example of a graph with $\Delta - 1 = s - \lfloor (s - 2) / (\Delta - 1) \rfloor = 2$ vertices not in its acyclic core.

From now on, let G be the join of two disjoint k -order graphs G_1 and G_2 with same maximum degree d . Recall that $\Delta(G) = k + d$. Let $B_G := G - (E(G_1) \cup E(G_2))$, $G_M := (G_1 \cup G_2) + M$ for any perfect matching M on B_G , and $M(x)$ be the vertex matched to x by M for any $x \in V(G_M)$. As observed by [8], G is *Class 1* whenever G_M is *Class 1* for some perfect matching M on B_G , since $G = G_M \cup (B_G - M)$, $\Delta(G_M) = d + 1$, and $B_G - M$ is bipartite, hence *Class 1* [4].

Our recolouring procedure is similar to Vizing's [12], for which we extend and adapt the concept of a recolouring fan. In Definition 1, in Lemmas 1 and 2, and in Conjecture 1, M is a perfect matching on B_G , $uv \in E(\Lambda[G_1])$, \mathcal{C} is a set with $d + 1$ colours, and $\varphi: E(G_M - uv) \rightarrow \mathcal{C}$ is an edge-colouring.

Definition 1. A sequence v_0, \dots, v_k of distinct neighbours of u in G_M is a *recolouring fan* for u if $v_0 = v$ and, for all $i \in \{0, \dots, k - 1\}$, either v_i *actually* misses the colour $\alpha_i := \varphi(uv_{i+1})$, or v_i misses α_i *virtually*, that is, $i > 0$, $v_i = M(u)$, and $\varphi(wM(w)) = \alpha_i$ for some $w \in V(G_1) \setminus \{v_{i-1}\}$ which actually misses α_{i-1} . If v_k misses, actually or virtually, a colour α_k missing at u , the fan is said to be *complete*; otherwise, it is said to be *incomplete*.

The figure below illustrates a complete recolouring fan. The dashed line indicates the edge to be coloured and the dotted lines indicate the colours missing at the vertices.

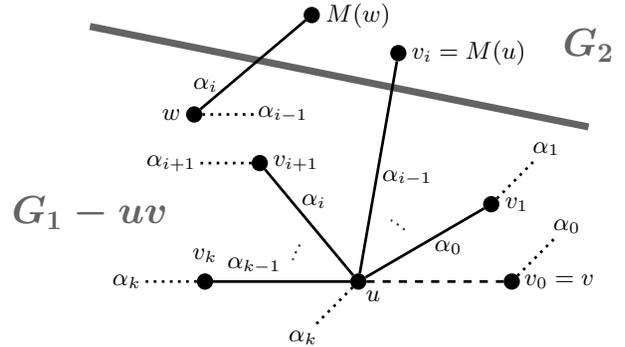


Figure 1: A complete recolouring fan

LEMMA 1. *If there is a complete recolouring fan for u , then $G_{M'}$ is Class 1 for some perfect matching M' on B_G .*

PROOF. We shall perform a procedure called *decay of the colours*, for i from k down to 0 . At the beginning of each iteration, it shall be invariant that both u and v_i miss α_i (the latter possibly virtually). Note that this is true for $i = k$.

If v_i actually misses α_i , we simply assign α_i to uv_i . If $i = 0$, we are done. If $i > 0$, u now misses α_{i-1} , which is still missing (possibly virtually) at v_{i-1} , so we can continue.

If v_i misses α_i virtually, recall that $i > 0$ and $v_i = M(u)$. Then, we take $M' := (M \setminus \{uv_i, wM(w)\}) \cup \{uM(w), wv_i\}$, colouring $uM(w)$ and wv_i with respectively α_i and α_{i-1} (see Figure 2). Now, both u and v_{i-1} actually miss α_{i-1} . \square

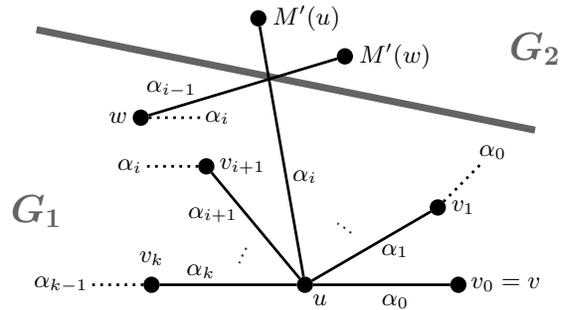


Figure 2: The result of the decay of the colours on the complete recolouring fan of Figure 1

CONJECTURE 1. *If v_0, \dots, v_k is a maximal, but not complete, recolouring fan for u such that $v_k = M(u)$, then there are some perfect matching \overline{M} on B_G and some $(d + 1)$ -edge-colouring of $G_{\overline{M}}$, both obtainable in polynomial time, under which there is a complete recolouring fan for u or a maximal recolouring fan for u starting in v_0 but not ending in $\overline{M}(u)$.*

LEMMA 2. *If Conjecture 1 holds for all non-complete maximal recolouring fan for u ending in $M(u)$, and if, for all $y \in N_{G_M}(u)$, either y misses a colour of \mathcal{C} , or $y = M(u)$ and $\varphi(uy)$ is missed by at least two vertices in $V(G_1)$, then $G_{M'}$ is Class 1 for some perfect matching M' on B_G .*

PROOF. Let $F = v_0, \dots, v_k$ be a maximal recolouring fan for u . If F is complete, the proof follows immediately from Lemma 1. Otherwise, as v_0 is itself a (not necessarily complete) recolouring fan for u , remark that $k \geq 0$. Moreover, if $M(u) = v_i$ for some $i \in \{1, \dots, k\}$, the conditions of the statement imply that there must be some $w \in V(G_1) \setminus \{v_{i-1}\}$ which miss $\varphi(uv_i) = \alpha_{i-1}$. Ergo, the only reason why F is not complete is because every colour α missing (actually or virtually) at v_k is equal to α_j for some $j < k$.

Let $\alpha = \alpha_j$ for some $j < k$ be a colour missing (actually or virtually) at v_k , β be any colour missing at u , and e be the edge incident to v_k coloured with β . Observe that $j < k - 1$, as $\alpha_{k-1} = \varphi(uv_k)$, and also that every component of the subgraph H of G_M induced by the edges coloured with α or β is a path or an even cycle. We have the following cases:

Case 1. The vertex v_k actually misses α .

Case 2. The vertex v_k misses α virtually.

In Case 1, the component of H to which e belongs is a path P , wherein v_k is one of its outer vertices. Exchanging the colours along P , we have the following sub-cases:

1. If the other outer vertex of P is u (which implies that $uv_{j+1} \in E(P)$), $v_j \notin V(P)$ and, thus, after the colour exchanging operation, both u and v_j miss α (the latter possibly virtually). Now, $F' := v_0, \dots, v_j$ is a complete recolouring fan for u , so we are done by Lemma 1.
2. If the other outer vertex of P is v_j , then $u \notin V(P)$ and, thus, after exchanging the colours along P , both u and v_j miss β (the latter possibly virtually). As in the previous sub-case, $F' := v_0, \dots, v_j$ is now a complete recolouring fan for u and Lemma 1 applies.
3. If the other outer vertex of P is neither u nor v_j , then, after the exchanging operation, u still misses β , v_j still misses α_j , and F is thus still a recolouring fan. But now F is complete, since now v_k misses β , so we apply Lemma 1, but in this sub-case in F instead of in F' .

In Case 2, $v_k = M(u)$ and there is some $w \in V(G_1) \setminus \{v_{k-1}\}$ which misses α_{k-1} such that $\varphi(wM(w)) = \alpha$ (see Figure 3). This is the case wherein our heuristic fails, but, if Conjecture 1 is true, we can handle this, ending up with a complete recolouring fan or going back to Case 1. \square

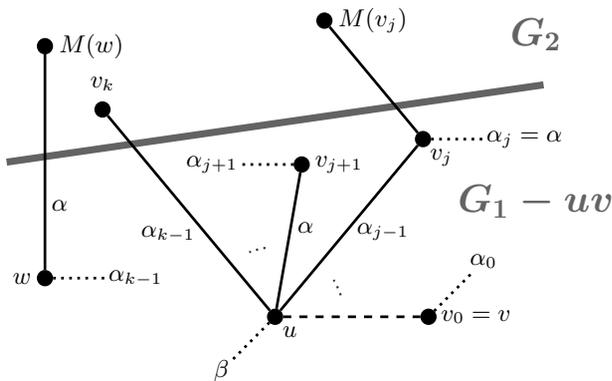


Figure 3: Case 2 in the proof of Lemma 2

THEOREM 1. Let G be the join of two disjoint k -order graphs G_1 and G_2 with same maximum degree d . If $\Lambda[G_1]$ is acyclic and Conjecture 1 is true with respect to any $uv \in E(\Lambda[G_1])$, any perfect matching M on B_G , and any non-complete maximal recolouring fan for u ending in $M(u)$, then G is Class 1.

PROOF. We assume that $d > 1$, since otherwise G_1 and G_2 are disjoint unions of cliques and we already know that G is Class 1 by [8]. For each of the components of $\Lambda[G_1]$, which are trees, choose a vertex to be the root of the tree. For each $u \in V(\Lambda[G_1])$, let $h(u)$ be the depth of u in its tree, i.e. the number of edges in the unique path between u and the root of its tree. If $h(u) > 0$, let also $p(u)$ denote the parent of u , i.e. the unique neighbour of u in $\Lambda[G_1]$ with depth equal to $h(u) - 1$. Consider the non-root vertices in $V(\Lambda[G_1])$ in a non-decreasing order of depth $\sigma = u_1, \dots, u_s$. If G_2 is not regular, take a perfect matching M on B_G such that $M(u_s) \notin V(\Lambda[G_2])$. Otherwise, take any perfect matching M on B_G . As proved in [5], G_M is Class 1 if $\Lambda[G_1]$ is edgeless, so $G_M - E(\Lambda[G_1])$ has an edge-colouring φ using a colour set \mathcal{C} with $|\mathcal{C}| = d + 1$.

Now, take the non-root vertices in $V(\Lambda[G_1])$, one at each time, following the order σ . For each u taken, we shall colour the edge $up(u)$. This shall complete the $(d + 1)$ -edge-colouring of G_M , possibly replacing, at each step, the current matching in the role of M with another perfect matching on B_G . However, if G_2 is not regular, the edge $u_sM(u_s)$ shall never be replaced.

In each step of our algorithm, let u be the non-root vertex of $V(\Lambda[G_1])$ taken. The only neighbour of u which may not miss a colour of \mathcal{C} is $M(u)$, because for every $x \in N_{\Lambda[G_1]}(u)$, either $x = p(u)$ or $h(x) > h(u)$, so the edge ux has not been coloured yet. If $u \neq u_s$, we have the following cases to investigate, with $\alpha := \varphi(uM(u))$:

Case 1. No vertex in G_1 misses α .

Case 2. At least two vertices in $V(G_1) \setminus \{u_s\}$ miss α .

Case 3. At most one vertex in $V(G_1) \setminus \{u_s\}$ misses α .

In Case 1, no recolouring fan for u starting in $v_0 = p(u)$ will contain $M(u)$, which means that every vertex in the fan will miss a colour. So, we will be able to apply Vizing's usual recolouring procedure and thence colour $up(u)$.

In Case 2, since we have assumed Conjecture 1, we can apply Lemma 2 in order to colour $up(u)$, and do so preserving the edge $u_sM(u_s)$ in M .

In Case 3, we must recall that $\sum_{v \in V(G_1)} (d - d_{G_1}(v)) \geq d - 1$, from Proposition 1. As $u \neq u_s$, at least two edges of $E(\Lambda[G_1])$ have not been coloured yet, one of them being $u_s p(u_s)$. Ergo, if H is the subgraph of G_M induced only by the coloured edges, $\sum_{v \in V(G_1) \setminus \{u_s\}} ((d + 1) - d_H(v)) = \sum_{v \in V(G_1)} ((d + 1) - d_H(v)) - 1$. Furthermore,

$$\sum_{v \in V(G_1)} ((d + 1) - d_H(v)) \geq \sum_{v \in V(G_1)} (d - d_{G_1}(v)) + 4.$$

Consequently,

$$\sum_{v \in V(G_1) \setminus \{u_s\}} ((d + 1) - d_H(v)) \geq d + 2.$$

By the Pigeonhole Principle, this means that there must be a colour γ missed by at least two vertices in $V(G_1) \setminus \{u_s\}$.

Looking at the subgraph H' of G_M induced by the edges coloured with γ or α , it is not hard to verify that there is some maximal path in H' along whose edges the exchanging of the colours brings us back to one of the previous cases.

Finally, let us consider the last step, when $u = u_s$. Defining again H as the subgraph of G_M induced by the coloured edges, remark that

$$\sum_{v \in V(G_1)} ((d + 1) - d_H(v)) \geq d + 1. \tag{1}$$

If G_2 is not d -regular, $M(u) \notin V(\Lambda[G_2])$ and we can apply the usual Vizing's recolouring procedure in order to colour $up(u)$. Assume then that G_2 is regular, which implies that no vertex in G_2 misses a colour of \mathcal{C} . We claim that we must have at least one colour missed by at least two vertices in G_1 , so the proof can follow analogously as in the previous steps. If we assume, by the sake of contradiction, that no colour of \mathcal{C} is missed by more than one vertex in G_1 , we have by (1) that every colour γ of \mathcal{C} is missed by exactly one vertex v_γ . If this is true, then also $v_{\gamma_1} \neq v_{\gamma_2}$ whenever $\gamma_1 \neq \gamma_2$ for all $\gamma_1, \gamma_2 \in \mathcal{C}$, because $U \cup \{u, p(u)\}$ is a set with $d + 1$ vertices of degree less than $d + 1$ in H for any set $U \subset V(G_1)$ with $d - 1$ vertices of degree less than d in G_1 . The existence of such U is guaranteed by Proposition 1. But then, creating a new vertex b in H and, for all $\gamma \in \mathcal{C}$, creating the edge bv_γ in H and colouring it with γ , H would be an odd-order Class 1 regular graph, something impossible to happen, as one can easily verify. Therefore, the claim holds. \square

Figure 4 illustrates the proof of Theorem 1 for the join of a diamond in the role of G_1 with a K_4 in the role of G_2 , which have both maximum degree $d = 3$. The figure depicts a perfect matching M on B_G , an edge-colouring of $G_M - E(\Lambda[G_1]) = G_M - up(u)$ with a colour set $\mathcal{C} = \{1, 2, 3, 4\}$, and a complete recolouring fan v_0, v_1, v_2 for u .

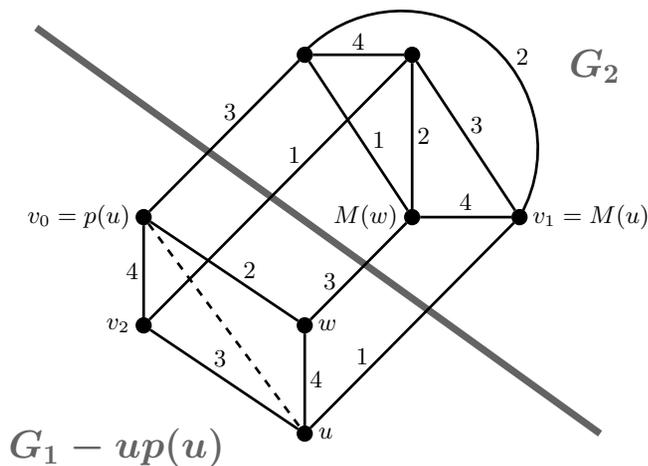


Figure 4: A perfect matching M on B_G when G_1 is a diamond and $G_2 = K_4$, a $(d + 1)$ -edge-colouring of $G_M - E(\Lambda[G_1])$, and a complete recolouring fan

3. FINAL REMARKS

We conclude with further discussion on Case 2 in the proof of Lemma 2. This is the case illustrated in Figure 3, which we have not been able to solve yet and for which we proposed

Conjecture 1. We remark that there are some sub-cases with which we know how to deal, as what follows enlightens.

For example, let H be the subgraph of G_M induced by the edges coloured with α or β , and let X be the component of H to which $wM(w)$ belongs. We know that X can be a path or an even cycle. If u is not in X , exchanging the colours of the edges of X brings that:

1. either v_0, \dots, v_j is a complete recolouring fan, because v_j is in X and now actually misses β ,
2. or v_0, \dots, v_k is a complete recolouring fan, because v_j is not in X and now v_k misses β virtually.

Either way, we know how to proceed by Lemma 1. This is sufficient to prove Conjecture 1 for the sub-case wherein u is not in X . Hence, the only remaining sub-case to prove is when u is in X . We encourage future works to investigate this sub-case and complete the proof.

4. ACKNOWLEDGEMENTS

We most heartily express our gratitude to the anonymous referee for their careful reading and valuable suggestions.

5. REFERENCES

- [1] A. G. Chetwynd and A. J. W. Hilton. The chromatic index of graphs of even order with many edges. *J. Graph Theory*, 8:463–470, 1984.
- [2] J.-C. Fournier. Méthode et théorème générale de coloration des arêtes. *J. Math. Pures Appl.*, 56:437–453, 1977.
- [3] I. Holyer. The NP-completeness of edge-colouring. *SIAM J. Comput.*, 10(4):718–720, 1981.
- [4] D. König. Graphok és alkalmazásuk a determinánsok és a halmazok elméletére. *Math. Természettudományi Értesítő*, 34:104–119, 1916.
- [5] A. R. C. Lima, G. Garcia, L. M. Zatesko, and S. M. de Almeida. On the chromatic index of cographs and join graphs. *Electron. Notes Discrete Math.*, 50:433–438, 2015.
- [6] R. C. S. Machado and C. M. H. de Figueiredo. Decompositions for edge-coloring join graphs and cobipartite graphs. *Discrete Appl. Math.*, 158:1336–1342, 2010.
- [7] T. Nielsen. How to find overfull subgraphs in graphs with large maximum degree, II. *Electron. J. Combin.*, 8, 2001.
- [8] C. Simone and C. P. de Mello. Edge-colouring of join graphs. *Theor. Comput. Sci.*, 355:364–370, 2006.
- [9] C. D. Simone and A. Galluccio. Edge-colouring of regular graphs of large degree. *Theor. Comput. Sci.*, 389:91–99, 2007.
- [10] C. D. Simone and A. Galluccio. Edge-colouring of join of regular graphs. *J. Comb. Optim.*, 18:417–428, 2009.
- [11] C. D. Simone and A. Galluccio. Edge-colouring of joins of regular graphs II. *J. Comb. Optim.*, 25:78–90, 2013.
- [12] V. G. Vizing. On an estimate of the chromatic class of a p -graph. *Diskret. Analiz.*, 3:25–30, 1964. In Russian.
- [13] A. Zorzi and L. M. Zatesko. On the chromatic index of join graphs and triangle-free graphs with large maximum degree. *Discrete Appl. Math.*, 2016. Article in press: <http://dx.doi.org/10.1016/j.dam.2016.10.022>.

ISSN 2526-1371



9 772526 137102

Departamento Acadêmico de Informática - Câmpus Ponta Grossa
Universidade Tecnológica Federal do Paraná