

ENGENHARIAS ELÉTRICA E DE COMPUTAÇÃO:

O TERCEIRO PILAR

LILIAN BOELHO DE FREITAS
(ORGANIZADORA)

Atena
Editora

Ano 2021

ENGENHARIAS ELÉTRICA E DE COMPUTAÇÃO:

O TERCEIRO PILAR

LILIAN BOELHO DE FREITAS
(ORGANIZADORA)

Atena
Editora

Ano 2021

Editora chefe

Profª Drª Antonella Carvalho de Oliveira

Editora executiva

Natalia Oliveira

Assistente editorial

Flávia Roberta Barão

Bibliotecária

Janaina Ramos

Projeto gráfico

Camila Alves de Cremo

Luiza Alves Batista

Maria Alice Pinheiro

Natália Sandrini de Azevedo

Imagens da capa

iStock

Edição de arte

Luiza Alves Batista

2021 by Atena Editora

Copyright © Atena Editora

Copyright do texto © 2021 Os autores

Copyright da edição © 2021 Atena Editora

Direitos para esta edição cedidos à Atena Editora pelos autores.

Open access publication by Atena Editora



Todo o conteúdo deste livro está licenciado sob uma Licença de Atribuição Creative Commons. Atribuição-Não-Comercial-NãoDerivativos 4.0 Internacional (CC BY-NC-ND 4.0).

O conteúdo dos artigos e seus dados em sua forma, correção e confiabilidade são de responsabilidade exclusiva dos autores, inclusive não representam necessariamente a posição oficial da Atena Editora. Permitido o *download* da obra e o compartilhamento desde que sejam atribuídos créditos aos autores, mas sem a possibilidade de alterá-la de nenhuma forma ou utilizá-la para fins comerciais.

Todos os manuscritos foram previamente submetidos à avaliação cega pelos pares, membros do Conselho Editorial desta Editora, tendo sido aprovados para a publicação com base em critérios de neutralidade e imparcialidade acadêmica.

A Atena Editora é comprometida em garantir a integridade editorial em todas as etapas do processo de publicação, evitando plágio, dados ou resultados fraudulentos e impedindo que interesses financeiros comprometam os padrões éticos da publicação. Situações suspeitas de má conduta científica serão investigadas sob o mais alto padrão de rigor acadêmico e ético.

Conselho Editorial

Ciências Exatas e da Terra e Engenharias

Prof. Dr. Adélio Alcino Sampaio Castro Machado – Universidade do Porto

Profª Drª Ana Grasielle Dionísio Corrêa – Universidade Presbiteriana Mackenzie

Prof. Dr. Carlos Eduardo Sanches de Andrade – Universidade Federal de Goiás

Profª Drª Carmen Lúcia Voigt – Universidade Norte do Paraná

Prof. Dr. Cleiseano Emanuel da Silva Paniagua – Instituto Federal de Educação, Ciência e Tecnologia de Goiás

Prof. Dr. Douglas Gonçalves da Silva – Universidade Estadual do Sudoeste da Bahia

Prof. Dr. Eloi Rufato Junior – Universidade Tecnológica Federal do Paraná
Profª Drª Érica de Melo Azevedo – Instituto Federal do Rio de Janeiro
Prof. Dr. Fabrício Menezes Ramos – Instituto Federal do Pará
Profª Dra. Jéssica Verger Nardeli – Universidade Estadual Paulista Júlio de Mesquita Filho
Prof. Dr. Juliano Carlo Rufino de Freitas – Universidade Federal de Campina Grande
Profª Drª Luciana do Nascimento Mendes – Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte
Prof. Dr. Marcelo Marques – Universidade Estadual de Maringá
Prof. Dr. Marco Aurélio Kistemann Junior – Universidade Federal de Juiz de Fora
Profª Drª Neiva Maria de Almeida – Universidade Federal da Paraíba
Profª Drª Natiéli Piovesan – Instituto Federal do Rio Grande do Norte
Profª Drª Priscila Tessmer Scaglioni – Universidade Federal de Pelotas
Prof. Dr. Sidney Gonçalo de Lima – Universidade Federal do Piauí
Prof. Dr. Takeshy Tachizawa – Faculdade de Campo Limpo Paulista

Engenharias elétrica e de computação: o terceiro pilar

Diagramação: Camila Alves de Cremo
Correção: Mariane Aparecida Freitas
Indexação: Gabriel Motomu Teshima
Revisão: Os autores
Organizadores: Lilian Coelho de Freitas

Dados Internacionais de Catalogação na Publicação (CIP)

E57 Engenharia elétrica e de computação: o terceiro pilar / Organizadora Lilian Coelho de Freitas. – Ponta Grossa - PR: Atena, 2021.

Formato: PDF
Requisitos de sistema: Adobe Acrobat Reader
Modo de acesso: World Wide Web
Inclui bibliografia
ISBN 978-65-5983-543-0
DOI: <https://doi.org/10.22533/at.ed.430213009>

1. Engenharia elétrica. 2. Computação. I. Freitas, Lilian Coelho de (Organizadora). II. Título.

CDD 621.3

Elaborado por Bibliotecária Janaina Ramos – CRB-8/9166

Atena Editora
Ponta Grossa – Paraná – Brasil
Telefone: +55 (42) 3323-5493
www.atenaeditora.com.br
contato@atenaeditora.com.br

DECLARAÇÃO DOS AUTORES

Os autores desta obra: 1. Atestam não possuir qualquer interesse comercial que constitua um conflito de interesses em relação ao artigo científico publicado; 2. Declaram que participaram ativamente da construção dos respectivos manuscritos, preferencialmente na: a) Concepção do estudo, e/ou aquisição de dados, e/ou análise e interpretação de dados; b) Elaboração do artigo ou revisão com vistas a tornar o material intelectualmente relevante; c) Aprovação final do manuscrito para submissão.; 3. Certificam que os artigos científicos publicados estão completamente isentos de dados e/ou resultados fraudulentos; 4. Confirmam a citação e a referência correta de todos os dados e de interpretações de dados de outras pesquisas; 5. Reconhecem terem informado todas as fontes de financiamento recebidas para a consecução da pesquisa; 6. Autorizam a edição da obra, que incluem os registros de ficha catalográfica, ISBN, DOI e demais indexadores, projeto visual e criação de capa, diagramação de miolo, assim como lançamento e divulgação da mesma conforme critérios da Atena Editora.

DECLARAÇÃO DA EDITORA

A Atena Editora declara, para os devidos fins de direito, que: 1. A presente publicação constitui apenas transferência temporária dos direitos autorais, direito sobre a publicação, inclusive não constitui responsabilidade solidária na criação dos manuscritos publicados, nos termos previstos na Lei sobre direitos autorais (Lei 9610/98), no art. 184 do Código Penal e no art. 927 do Código Civil; 2. Autoriza e incentiva os autores a assinarem contratos com repositórios institucionais, com fins exclusivos de divulgação da obra, desde que com o devido reconhecimento de autoria e edição e sem qualquer finalidade comercial; 3. Todos os e-book são *open access*, desta forma não os comercializa em seu site, sites parceiros, plataformas de *e-commerce*, ou qualquer outro meio virtual ou físico, portanto, está isenta de repasses de direitos autorais aos autores; 4. Todos os membros do conselho editorial são doutores e vinculados a instituições de ensino superior públicas, conforme recomendação da CAPES para obtenção do Qualis livro; 5. Não cede, comercializa ou autoriza a utilização dos nomes e e-mails dos autores, bem como nenhum outro dado dos mesmos, para qualquer finalidade que não o escopo da divulgação desta obra.

APRESENTAÇÃO





Os avanços na pesquisa científica em Engenharias Elétrica e de Computação tem revolucionado nossa vida em sociedade. Conexões cada vez mais rápidas, processadores super velozes e a autonomia dos sistemas decorrentes do progresso em Inteligência Artificial são alguns exemplos de aplicações em nosso dia-a-dia.

Este e-book torna acessível os resultados da pesquisa científica realizada por diversos pesquisadores do país. Ao decorrer dos capítulos, apresenta-se aplicações práticas de inteligência artificial, gerência de redes e técnicas de otimização. Aproveite esse momento para aprimorar seus conhecimentos.

Desejo aos autores, meu mais sincero agradecimento pelas significativas contribuições, e aos nossos leitores, desejo uma proveitosa leitura, repleta de boas reflexões.

Lilian Coelho de Freitas

SUMÁRIO

CAPÍTULO 1	1
PLANNING AS MIXED-HORN FORMULAS SATISFIABILITY	
Razer Anthom Nizer Rojas Montaña	
Bruno César Ribas	
 https://doi.org/10.22533/at.ed.4302130091	
CAPÍTULO 2	15
ASSIMILAÇÃO DE DADOS POR REDES NEURAIS EM AUSÊNCIA PARCIAL DE OBSERVAÇÕES	
Rosangela Saher Corrêa Cintra	
Haroldo Fraga de Campos Velho	
 https://doi.org/10.22533/at.ed.4302130092	
CAPÍTULO 3	30
CONCEPÇÃO E OTIMIZAÇÃO DE UMA CLASSIFICAÇÃO OBJETIVA PARA SALAS DE ÓPERA UTILIZANDO MÉTODOS COMPUTACIONAIS	
Marco Antônio M. Vecci	
Calebe Giaculi Júnior	
Bruno Alberto Soares Oliveira	
 https://doi.org/10.22533/at.ed.4302130093	
CAPÍTULO 4	42
SERVIDOR DE GERÊNCIA DE REDE	
Roger Robson dos Santos	
Jackson Mallmann	
 https://doi.org/10.22533/at.ed.4302130094	
SOBRE A ORGANIZADORA	55
ÍNDICE REMISSIVO	56

CAPÍTULO 1

PLANNING AS MIXED-HORN FORMULAS SATISFIABILITY

Data de aceite: 01/09/2021

Data de submissão: 05/07/2021

Razer Anthom Nizer Rojas Montañó

Setor de Educação Profissional e Tecnológica -
Universidade Federal do Paraná
Curitiba - Paraná
<http://lattes.cnpq.br/8472117901680275>

Bruno César Ribas

Faculdade do Gama - Universidade de Brasília
Brasília - Distrito Federal
<http://lattes.cnpq.br/6871745921667698>

ABSTRACT: Over the course of the last decade, there have been several improvements in the performance of Boolean Satisfiability (SAT) but just a few works focused on Planning as Satisfiability recently. This paper explores a new way to model a planning instance from BLACKBOX and SATPLAN06. This new approach is a mixed-horn formula (MHF) where it can be seen as a conjunction of a Horn formula and a 2-CNF formula. The theoretical complexity to solve an MHF is $O(2^{0.5284n})$. With the theoretical complexity combined with improvements of SAT solvers, it is a promising approach to model Planning as Mixed-Horn Formulas Satisfiability. It is observed that this formulation has some improvements with current state-of-the-art SAT solvers.

KEYWORDS: Planning, Satisfiability, Mixed-horn formula, Satplan.

PLANEJAMENTO COMO SATISFATIBILIDADE DE FÓRMULAS HORN-MISTAS

RESUMO: Ao longo da última década, houveram diversas melhorias no desempenho dos resolvidores de Satisfabilidade Booleana (SAT), mas poucos trabalhos em Planejamento como Satisfabilidade exploraram essas novas estratégias. Este artigo explora uma nova maneira de modelar uma instância de planejamento do BLACKBOX e SATPLAN06. Esta nova abordagem é transformar a codificação padrão em uma fórmula Horn-Mista (MHF), que pode ser separada como uma conjunção de uma fórmula de Horn e uma fórmula 2-CNF. A complexidade teórica para resolver uma MHF é $O(2^{0.5284n})$. A complexidade teórica da MHF combinada com as melhorias dos resolvidores SAT leva a uma abordagem promissora, conforme observamos nos benchmarks, quando comparada com a formulação padrão utilizando os mesmos resolvidores SAT.

PALAVRAS-CHAVE: Planejamento, Satisfabilidade, Fórmula Horn-Mista, Satplan.

1 | INTRODUCTION

Planning in Artificial Intelligence is an important field of study and an interesting application of Satisfiability to solve generic search problems. The planning problem has the objective to find a sequence of actions from an initial state to reach a given goal. These actions sequences and state sequences attained from actions taken corresponds to assignments of a

propositional formula which can be easily generated from the problem description.

In (KAUTZ; SELMAN, 1992) it is proposed a way to solve planning problems using propositional Satisfiability (SAT), but only after 1996, they demonstrated that modern SAT solvers and better translations provided a competitive approach to planning (KAUTZ; SELMAN, 1996).

SATPLAN is the planner based on SAT and one of the most successful approaches to STRIPS planning. This approach translates a planning problem to SAT problems, one formula to each horizon of actions applied. The first satisfiable formula means that a plan was found.

The efficiency of SAT solver is one of the most important characteristics to the success of this logical approach to the planning problem. Consequently, encoding the problem is also important, because the information represented can make the solver work better.

SATPLAN06 constructs a planning graph in a GRAPHPLAN fashion then translates this graph to a set of clauses (CNF) to be solved by a SAT Solver (BLUM; FURST 1995). While the generated formula is UNSAT, the graph is expanded. When the formula is SAT, the valuation found is translated to the solution of the original planning problem.

The classical translation of planning to satisfiability is using axioms generated from the planning graph. The choice of axioms leaves different encodings with different characteristics, as presented by Sideris and Dimopoulos (2010).

While solving time is highly dependent of the SAT solvers, due to better implementation and better solver choices such as: branching heuristics, learning and restart strategies, among others; the way of generating the formula is very important to achieve better solving time, while it is common sense that a minimal formula is better, this is not always the case for current solvers as briefly discussed in (PRESTWICH, 2009). Also, Hooker (2007) suggests that Constraint Programming and Operational Research, in which SAT exchanges ideas and techniques, are Empirical Science and encoding of the problem have to be considered when solving a problem.

In this paper, we show that a planning problem can be translated to a Mixed-Horn Formula (MHF) instance, to be solved by a specific SAT solver. MHFs are formulas composed of 2-CNF positive formulas and Horn formulas. It was demonstrated by Porschen and Speckenmeyer (2007) that SAT in MHF is $O(2^{0.5284n})$, still NP-Complete, but theoretically better than general CNF SAT.

This paper is organized as follows. In section 2 we present theoretical foundations and background information of SAT, planning, and MHF. In section 3 we show the proposed translation of a planning problem in MHF SAT, based on available SATPLAN encodings followed by section 4 where we show preliminary results encoding planning problems in MHF. Finally, in section 5 we present the conclusion and future work.

2.1 THEORETICAL FOUNDATIONS AND BACKGROUND

2.1 Boolean Satisfiability

In Boolean Logic, a formula is said to be in the *Conjunctive Normal Form* (CNF) if it is formed by a sequence of conjunctions, represented by “ \wedge ” consisting of one or more clauses. Each of these clauses should be a disjunction, represented by “ \vee ” of one or more literals. A literal is a symbol or the negation of it, used to represent a Boolean statement in logic that can take the value either *True* or *False*. All conjunctions of literals and all disjunctions of literals are in *CNF*, as they can be seen as conjunctions of one-literal clauses and conjunctions of a single clause, respectively. As mentioned, the only propositional connectives in a *CNF* formula are “ \wedge ” “ \vee ” and “ \neg .”

Given an assignment to the literals in a CNF formula, the formula is satisfied *iff* all of its clauses are satisfied and a clause is satisfied *iff* at least one of its literals is satisfied. A positive literal is satisfied *iff* the corresponding variable is assigned value *True* and a negative literal is satisfied *iff* the corresponding is assigned to *False*.

The problem of deciding whether a CNF (Boolean) formula is satisfiable, i.e. if there exists an assignment to the variables such that the formula is satisfied, is called Boolean *Satisfiability* (SAT). SAT was the first problem that was proved to be NP-complete (COOK, 1971).

A detailed description of modern SAT solvers, techniques, and CNF encodings can be found, respectively in (DARWICHE; PIPATSRISAWAT, 2009), (MARQUES-SILVA; LYNCE; MALIK, 2009) and (PRESTWICH, 2009).

2.2 Planning

A *planning problem*, in the reasoning about actions area, is defined as the problem to discover a sequence of actions that goes from an initial situation to reach a goal state. According to Weld (1999), an algorithm that solves these kinds of problems must receive as input:

- A description of world entities and actual situation;
- A description of goals;
- A description of allowed actions.

These algorithms return a sequence of actions that reach a goal state from a starting state.

Planning problems were computationally modeled by Fikes and Nilsson (1971), where they proposed a formal representation that could solve the problem, in conjunction with a search algorithm. This approach is known as STRIPS. Despite the simplicity of STRIPS, planning was proved PSPACE-Complete in (BYLANDER, 1994).

The first algorithm that solves planning with satisfactory performance was SATPLAN

(KAUTZ; SELMAN, 1992) (KAUTZ; SELMAN,1996), which translates the planning problem to satisfiability (SELMAN; LEVESQUE; MITCHELL,1992) (SAT).

In (BLUM; FURST,1995) a new approach is presented, resulting in a new planner called *GRAPHPLAN*. This approach represents the planning problem as a planning graph and the resolution is using a backward search. In (KAUTZ; SELMAN, 1999) SAT-based and graph-based approaches were united, resulting in a smaller instance compared to the generated from *GRAPHPLAN*, it was called BLACKBOX.

In 2004, Kautz and Selman submitted to the 2004 International Planning Competition (IPC) a new version of SATPLAN (called SATPLAN04) with some improvements. But in 2006, the SATPLAN06 (KAUTZ; SELMAN; HOFFMANN 2006) was implemented with mutex propagation on the planning graph and encoding with boolean variables for actions and fluents. This allowed harder instances to be solved and avoid some memory problems.

2.3 Planning as Satisfiability

A STRIPS planning problem is a triple $P=(I,G,A)$, where I is the set of facts true in the initial state, G is the set of facts true in the goal state, and A a set of allowed actions. Each action $a \in A$ has preconditions ($pre(a)$), added effects ($add(a)$) and deleted effects ($del(a)$).

In SAT, propositions are time-stamped to represent actions and facts in different situations. Let $a \in A$ be an action, so $a(t)$ is a decision if the action a is taken or not at time t . The same notion is applied to facts, eg, $f(t)$.

According to Sideris and Dimopoulos (2010), the direct translation of *GRAPHPLAN* into propositional logic is made using some subset of the following clauses:

1. Unit clauses for initial and goal state;
2. $a(t) \rightarrow f(t)$, for every action a and fact $f \in pre(a)$;
3. $a(t) \rightarrow f(t+1)$, for every action a and fact $f \in add(a)$;
4. $a(t) \rightarrow \neg f(t+1)$, for every action a and fact $f \in del(a)$;
5. $f(t) \rightarrow a_1(t-1) \vee \dots \vee a_m(t-1)$, for every fact $f \in add(a)$ and all actions a_p , $1 \leq i \leq m$ (including the noops);
6. $\neg f(t) \rightarrow a_1(t-1) \vee \dots \vee a_m(t-1) \vee \neg f(t-1)$, for every fact $f \in del(a)$ and all actions a_p , $1 \leq i \leq m$;
- 7.1. $\neg a_1(t) \vee \neg a_2(t)$, for every pair of action a_1, a_2 , such that the set $del(a_1) \cap pre(a_2)$ is non-empty;
- 7.2. $\neg a_1(t) \vee \neg a_2(t)$, for every pair of action a_1, a_2 , such that the set $del(a_1) \cap add(a_2)$ is non-empty;
- 7.3. $\neg a_1(t) \vee \neg a_2(t)$, if there is a pair of facts $f_1 \in pre(a_1), f_2 \in pre(a_2)$, such that f_1, f_2 are mutually exclusive at time t ;
8. $\neg f_1(t) \vee \neg f_2(t)$, for every pair of facts f_1, f_2 that are mutex at time t .

BLACKBOX support different encodings of planning problem, for example:

- BB-7: Clauses 1, 2, 5, 7.1, 7.2, 7.3
- BB-31: Clauses 1, 2, 3, 4, 5, 7.1, 8
- BB-32: Clauses 1, 2, 3, 4, 5, 7.1, 7.2, 7.3, 8

SATPLAN06 also supports different encodings, such as:

- SATPLAN06-3: Clauses 1, 2, 5, 7.1, 7.2, 7.3, 8
- SATPLAN06-4: Clauses 1, 2, 5, 7.1, 7.2, 8

Each encoding has different characteristics and must be chosen carefully to be translated to MHF.

2.4 Mixed-Horn Formulas

Mixed-Horn Formulas (MHF) are formulas composed of a 2-CNF part (clauses containing only two literals) and a Horn part (clauses with at most one positive literal). In our case, the 2-CNF part is composed only of positive literals, called positive monotone 2-CNF. The research in MHF is not new, and many problems can be formulated in terms of MHF, like the level-planarity test (Randerath et al. 2001) and graph colorability. Now we are introducing planning as a problem that can be represented directly as MHFs.

It is well known that 2-CNF satisfiability and Horn satisfiability are solvable in linear time (ASPVALL; PLASS; TARJAN, 1979) (MINOUX, 1988), but SAT for MHF (MHF-SAT) is NP-Complete as showed by Porschen and Speckenmeyer (2007). In (PORSCHEN; SPECKENMEYER, 2005) (PORSCHEN; SPECKENMEYER, 2007), it was studied satisfiability of Mixed-Horn Formulas, proving that the worst case is $O(2^{0.5284n})$, where n is the number of variables in the instance.

Algorithm: MHFSAT

Require: M: MHF Formula
Ensure: r: Model found

```
1  Compute P := P(M)
2  IF P=∅ THEN
3    RETURN r ← HornSat(M)
4  END IF
5  Compute Graph Gp
6  r ← nil; X ← nil;
7  REPEAT
8    Compute by MinVC(Gp) the next minimal vertex cover X of Gp
9    IF X ≠ nil THEN
10     r ← HornSat(M[X])
11   END IF
12 UNTIL r ≠ nil OR X = nil
13 RETURN X ∪ r
```

ALGORITHM 1: MFH-SAT Algorithm using minimal vertex cover.

Porschen and Speckenmeyer also showed there is a polynomial-time transformation

of an unrestricted CNF to MHF. In our case, this transformation is not necessary, because each horizon of planning problem can be generated directly as MHF, with only one simple flip of some literals and it will be explained in the next section.

Algorithm 1 defined by Porschen and Speckenmeyer (2007) solves MHF-SAT using minimal vertex cover. The function $P(M)$ returns all positive monotone 2-clauses in M , $MinVC(G)$ generate all minimal vertex covers of a graph G with polynomial delay. The function $HornSat(H)$ returns a minimal model τ of H if, and only if, H is a satisfiable Horn formula, else returns *nil*.

3 I PROPOSED TRANSLATION

This paper proposes a Mixed-Horn formulation to the formulas generated by SATPLAN06 and BLACKBOX hoping we may gain some improvements in solving time with current state-of-the-art SAT solvers. It is important to note that the transformation is cheap to implement, leading to no extra overheads to the planner, as shown below.

Analyzing the axioms used to generate SAT instances from the planning graph, we can generate Mixed-Horn Formulas from, eg, configuration BB-7. In this set of clauses, clauses 1 are unit, clauses 2, 7.1, 7.2, and 7.3 are 2-CNF, and clause 5 can be Horn if we invert interpretation of action taken. If an action a is taken, we represent it as $\neg a$ and if is not taken we can represent it as a .

The set of axioms (1, 2, 5, 7.1, 7.2 and 7.3), representing actions as described above and converted to clauses, will be:

1. Unit clauses for initial and goal state;
2. $a(t) \vee f(t)$, for every action a and fact $f \in pre(a)$;
5. $\neg f(t) \vee \neg a_1(t-1) \vee \dots \vee \neg a_m(t-1)$, for every fact $f \in add(a_i)$ and all actions a_i , $1 \leq i \leq m$ (including the noops);
- 7.1. $a_1(t) \vee a_2(t)$, for every pair of action a_1, a_2 , such that the set $del(a_1) \cap pre(a_2)$ is non-empty;
- 7.2. $a_1(t) \vee a_2(t)$, for every pair of action a_1, a_2 , such that the set $del(a_1) \cap add(a_2)$ is non-empty;
- 7.3. $a_1(t) \vee a_2(t)$, if there is a pair of facts $f_1 \in pre(a_1), f_2 \in pre(a_2)$, such that f_1, f_2 are mutually exclusive at time t ;

Thus, clauses 1 are unit, clauses 2, 7.1, 7.2, and 7.3 are positive monotone 2-CNF, and clauses 5 are Horn. This set of clauses is MHF (2-CNF and Horn) and can be solved or with general SAT solvers or specific MHF solvers.

To modify SATPLAN06, we used configuration 4 (clauses 1, 2, 5, 7.1, 7.2, and 8). The clauses converted are the same as BLACKBOX, except for the exclusion of clause 7.3 and inclusion of clause 8, which is negated 2-CNF (or negated Horn, depending on

interpretation). Clause 8 just have facts, so its generation to MHF is the same:

8. $\neg f_1(t) \vee \neg f_2(t)$, for every pair of facts f_1, f_2 that are mutex at time .

The translation proposed to change some literals (all action literals) by its negations. This change makes the formula generated by SATPLAN06 be in MHF format. This formula is equi-satisfiable to the original formula and, when applied to a SAT Solver, has the same behavior, but the valuation of action literals must be negated in the subjacent planning problem.

Theorem 1. *Be ϕ a theory in CNF. For all literal α in ϕ , changing α by $\neg\alpha$ gives a theory ϕ' which is equi-satisfiable to ϕ .*

Proof. ϕ and ϕ' are equi-satisfiable if ϕ is satisfiable if, and only if, ϕ' is satisfiable. If ϕ is SAT, then ϕ' is SAT too. Let V be a valuation in ϕ . If α is *true (false)* in V , so, by the transformation made, the same valuation with α as *false (true)* will be a model of ϕ' . If ϕ' is SAT, then ϕ is SAT too. Let V' be a valuation in ϕ' . If α is *true (false)* in V' , so, by the transformation made, the same valuation with α as *false (true)* will be a model of ϕ .

Q.E.D.

As showed by Bylander (1994), planning is PSPACE-Complete. As in CNF, SAT in MHF is NP-Complete and despite the use of MHF representation (where SAT theoretically has better complexity function), solving planning with satisfiability in MHF formulas remains in the PSPACE-Complete class of complexity.

4 | EXPERIMENTS

For evaluation of the efficiency of MHF compared to standard CNF encoding we used SATPLAN06 as our planner and used several domains of classical planning (STRIPS) from the ICAPS competition of various years. Our experiments were executed on an Intel Xeon 2.1GHz with 256GB of memory.

Each problem was executed with SATPLAN06 times for every combination. For all executions SATPLAN06 was set to run with LINGELING and GLUCOSE SAT solvers, for both encodings: default CNF and MHF. Each formula generated by SATPLAN06 was given a time limit of 3600 seconds to solve and the global time limit (including time spent solving formulas and the planner generating them) was set to 4000 seconds. All execution uses SATPLAN06 with configuration 4, explained in section 2.3.

Tables 1 to 14 show the average time in seconds of 10 executions of each problem. All tables are divided into two different results, one with CNF encoding and the other with MHF encoding, in both approaches it is shown average time of SATPLAN06 with LINGELING and GLUCOSE SAT solvers. Inside table, TO means timeout were reached, MEM means memory limit exceeded.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	57.688	12.945	56.925	14.300
2	28.897	10.101	35.936	10.155
3	228.196	35.283	227.794	36.409
4	TO	306.438	TO	304.40

TABLE 1: Domain: Storage.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	3.046	2.581	3.001	2.560
2	5.367	5.336	5.383	5.332
3	43.690	20.257	53.270	17.865
4	19.904	17.761	30.769	17.657

TABLE 2: Domain: Satellite

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	.678	.695	.667	.679
2	1.149	1.171	1.128	1.161
3	1.233	1.211	1.216	1.197
4	4.370	4.088	4.323	4.065
5	24.375	29.546	24.321	29.373
6	106.738	161.330	106.054	160.248
7	333.911	494.478	330.293	491.316

TABLE 3: Domain: Blocks.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	.662	.679	.648	.658
2	1.524	1.453	1.549	1.432
3	3.325	2.461	3.200	2.432
4	10.639	4.833	9.554	4.696
5	36.527	15.645	50.317	16.509
6	239.897	90.239	462.487	90.275
7	2894.689	1236.842	3127.132	2104.082

TABLE 4: Domain: Gripper

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	6.747	6.162	6.816	6.137
2	6.728	6.143	6.799	6.103
3	58.443	82.510	57.055	81.727
4	532.297	494.540	525.645	485.740
5	877.930	820.832	1003.564	813.832

TABLE 5: Domain: Depots.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	2.148	1.824	2.116	1.777
2	5.477	4.203	5.418	4.094
3	9.429	7.053	9.400	6.903
4	150.633	44.106	134.581	44.842
5	1398.500	1691.914	543.222	535.163

TABLE 6: Domain: Logistic.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	3.913	2.831	3.901	2.820
2	30.701	8.340	25.789	8.343
3	21.816	10.215	23.541	10.114
4	49.446	24.298	50.025	23.838
5	85.416	19.327	98.968	19.924
6	1569.046	972.555	1655.302	978.371

TABLE 7: Domain: Drivelog.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	4.130	3.529	4.375	3.544
2	8.101	4.424	7.690	4.384
3	6.967	4.444	8.822	4.428
4	30.288	10.541	30.886	9.653
5	151.592	39.544	158.124	36.591
6	1177.997	949.610	1125.387	1122.207

TABLE 8: Domain: Elevator.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	31.169	27.620	31.260	27.510
2	55.887	35.541	54.953	35.319
3	76.951	75.865	74.593	76.543
4	1310.279	420.913	1363.944	394.910

TABLE 9: Domain: TPP.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	3.046	2.581	3.001	2.560
2	5.367	5.336	5.383	5.332
3	43.690	20.257	53.270	17.865
4	19.904	17.761	30.769	17.657

TABLE 10: Domain: Thoughtful Bootstrap.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	1.774	1.768	1.776	1.734
2	35.109	29.692	36.313	29.765
3	9.844	15.917	9.609	15.916
4	402.043	486.056	421.980	367.658
5	2295.060	TO	2151.280	TO

TABLE 11: Domain: FreeCell.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	97.892	52.732	97.851	54.316
2	284.710	382.130	288.636	384.086
3	154.948	94.122	163.359	86.735
4	369.755	405.247	348.846	397.876
5	517.795	457.101	481.796	452.125

TABLE 12: Domain: PipesWorld.

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	13.692	16.172	13.566	15.994
2	13.691	16.179	13.495	15.991
3	18.711	29.480	18.434	29.238
4	25.335	46.123	24.981	45.756
5	35.531	78.851	35.001	78.547
6	35.483	78.812	34.910	78.360
7	52.280	46.895	51.055	41.637
8	51.976	42.452	50.810	41.675
9	75.915	61.691	74.840	60.234
10	76.071	61.350	74.634	60.196
11	100.243	80.574	98.256	79.517
12	100.081	80.418	98.594	79.988
13	144.287	117.127	142.867	114.805
14	202.641	160.025	193.888	168.339
15	203.591	159.455	192.476	156.243
16	252.118	201.425	251.550	194.885
17	345.853	281.009	345.302	263.331
18	464.029	350.808	457.279	348.500
19	MEM	MEM	MEM	MEM
20	MEM	MEM	MEM	MEM

n	CNF		MHF	
	LING	GLUC	LING	GLUC
1	1.097	1.065	1.127	1.084
2	.734	.732	.703	.747
3	1.239	1.198	1.209	1.232
4	.671	.685	.742	.705
5	2.334	1.753	2.099	1.804
6	16.898	4.362	16.734	4.507
7	1.719	1.486	1.714	1.509
8	31.470	4.730	34.578	4.782
9	5.740	4.402	5.758	4.445
10	TO	21.418	TO	20.151
11	10.002	11.325	10.013	11.356
12	12.683	3.357	13.184	3.417
13	TO	162.792	TO	175.692
14	8.779	7.134	8.615	7.109
15	45.830	26.160	38.673	26.424
16	TO	59.306	TO	53.972
17	TO	182.810	TO	169.837
18	TO	88.624	TO	88.513
19	TO	TO	TO	TO
20	TO	TO	TO	TO
21	TO	TO	TO	TO
22	TO	TO	TO	TO
23	TO	TO	TO	TO

24	TO	TO	TO	TO
25	939.990	808.109	938.461	787.637
26	TO	1199.304	TO	1181.282

TABLE 13: Domain: Childsnack.

TABLE 14: Domain: Rovers.

With these results, we can conclude that solving classical planning problems using SAT with MHF formulas is feasible. As we can see, the results using MHF are close to the results using traditional CNF. State-of-the-art SAT solvers are generic solvers for the Satisfiability problem and might not always take advantage of the specific structural information rooted in the formula, therefore a specific SAT solver to solve MHF is needed to enhance solving time, small changes to the branching heuristics and learning structures might improve the performance.

Despite the closer times, some problems are solved faster in MHF. In the Logistics domain, table 6, problem 5 takes 1398.5s to solve with LINGELING in CNF while it takes 543.222s in MHF. The main reason is the last formula (the SAT formula to be solved) that takes 777s to be solved in CNF and is just 67s in MHF. Other domains have this behavior too, but we are still studying this phenomenon.

Analyzing all executions, we need to know if there is a statistical difference between the $K = 4$ configurations and $N = 93$ databases (taking only bases with execution was successful in all configurations). Friedman test (DEMSAR, 2006) is used to make paired data analysis using performance positions (rank) of all executions. For all N databases, each execution is ranked and an average for each K configuration is calculated.

Although the Friedman test uses a X^2 distribution, (DEMSAR, 2006) recommends a less conservative version proposed by Iman and Davenport (1980), which uses F distribution with $(K-1)$ and $(K-1)(N-1)$ degrees of freedom.

Evidencing the statistic difference, we applied the Nemenyi post-hoc to show it (NEMENYI, 1963). According to it, the effectiveness of the two methods is significantly different if their ranks differ at least for a critical difference value, using a level of significance.

Table 15 shows the average ranking made using all executions.

	CNF		MHF	
	LING	GLUC	LING	GLUC
Average rank	3,21505376344086	2,12903225806452	2,91397849462366	1,74193548387097

TABLE 15: Average ranking Average of execution ranking of formulas. To generate valuable results, we used only formulas solved by both solvers in both encodings. For each database, all executions are ranked (1-4) from best time execution to worst. Then, for each encoding, the average of all rankings is calculated.

Applying Friedman test with $K = 4$ configurations, $N=93$ databases, and significance

level at 5%, Friedman statistic results $F_f = 35.59910199$ and $p\text{-value} = 1.76238E - 19$, pointing to statistical difference.

To evidence this difference, we applied the Nemenyi post-hoc at critical difference $CD = 0.486370723$, $q_{0.05} = 2.569032073$. Figure 1 shows the resulting difference diagram. In this diagram, the x-axis represents the average rank for each configuration. Lines below the x-axis connect configurations without a statistical difference at 95% of confidence level. Critical difference (CD) is showed above the x-axis.

From this diagram, we can see there is no statistical difference because of problem encoding, but the solver used.

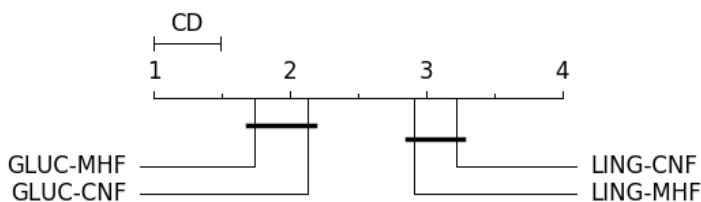


FIGURE 1: Difference graphic for Nemenyi post-test.

5 | CONCLUSION

This paper presented a new way to encode Planning as SAT using Mixed-Horn Formulas. It is important to focus on ways to represent problems as techniques and SAT solver improvements are not the only factor to reduce solving time, as discussed by Hooker (2007).

Analyzing formulas generated by planners that use satisfiability as an engine, we note that clauses are close to MHF. With an equi-satisfiable modification, we can generate MHF instead of CNF and this new encoding is now an option for solvers.

MHFs have received much attention recently (KOTTLER; KAUFMANN; SINZ, 2008) (PORSCHEN; SCHMIDT; SPECKENMEYER, 2009). Many NP-complete problems were proved to have encoding to MHF (PORSCHEN; SCHMIDT; SPECKENMEYER, 2009). In this paper, we show classical planning also can be solved using MHF and have a promising area of study.

In (PORSCHEN; SPECKENMEYER, 2005) and (PORSCHEN; SPECKENMEYER, 2007), they showed that the worst-case satisfiability of an MHF is $O(2^{0.5284n})$ and despite this complexity, it is still NP-complete. They showed too that any SAT instance can be encoded in MHF in polynomial time. In our encoding, the same process that generates CNF is used to generate MHF, without any overhead.

In this paper, we show that MHF can be used successfully to encode a planning problem, modified SATPLAN06 to generate MHFs instead of normal CNF, and executed

many experiments using the same domains applied in IPC. The results show no significant improvements over the traditional CNF encoding, but an implementation of a new SAT solver based on MHF structure can improve our experiments.

We show there is no statistical difference between encodings used, despite some better executions when MHF is applied. This can occur because the solvers used in MHF encoding are not optimized to take advantage of the MHF structure.

As future work, specific SAT solvers for MHF must be used to compare executions. Even a new implementation using specific MHFs for planning can be studied. Besides, as described by Porschen and Speckenmeyer (2007), we can study the effect of reducing the number of essential variables, and if it is applied to our domain (classical planning).

For now, we are studying the phenomena described in problem Logistics-5, trying to identify domains with this behavior and solving UNSAT formulas faster in MHF.

REFERENCES

ASPVALL, Bengt; PLASS, Michael F.; TARJAN, Robert E. **A Linear-Time Algorithm for Testing the Truth of Certain Quantified Boolean Formulas.** *Inf. Process. Lett.*, 121–23. 1979.

BLUM, Avrim; FURST, Merrick. **Fast Planning Through Planning Graph Analysis.** In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI 95)*, 1636–42. 1995. citeseer.ist.psu.edu/blum95fast.html.

BYLANDER, Tom. **The Computational Complexity of Propositional STRIPS Planning.** *Artificial Intelligence* 69 (1-2): 165–204. 1994. citeseer.ist.psu.edu/bylander94computational.html.

COOK, Stephen A. **The complexity of theorem-proving procedures.** In *STOC '71: Proceedings of the Third Annual ACM Symposium on Theory of Computing*, 151–58. New York, NY, USA: ACM Press. 1971.

DARWICHE, Adnan; PIPATSRISAWAT, Knot. **Complete Algorithms.** In *Handbook of Satisfiability*, edited by Armin Biere, Marijn J. H. Heule, Hans van Maaren, and Toby Walsh, 185:99–130. Frontiers in Artificial Intelligence and Applications. IOS Press. 2009.

DEMSAR, Janez. **Statistical Comparisons of Classifiers over Multiple Data Sets.** *Journal of Machine Learning Research* 7: 1–30. 2006.

FIKES, R. O.; NILSSON, N. J. **STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving.** 43r. 333 Ravenswood Ave, Menlo Park, CA 94025: AI Center, SRI International. 1971.

HOOKER, John N. **Good and Bad Futures for Constraint Programming (and Operations Research).** *Constraint Programming Letters* 1: 21. 2007.

IMAN, Ronald L.; DAVENPORT, James M. **Approximations of the Critical Region of the Fbietkan Statistic.** *Communications in Statistics - Theory and Methods* 9 (6): 571–95. 1980. <https://doi.org/10.1080/03610928008827904>.

KAUTZ, Henry A.; SELMAN, Bart. 1992. **Planning as Satisfiability**. In *Proceedings of the Tenth European Conference on Artificial Intelligence (ECAI'92)*, 359–63. 1992. citeseer.ist.psu.edu/kautz92planning.html.

KAUTZ, Henry A.; SELMAN, Bart; HOFFMANN, Jörg. **SatPlan: Planning as Satisfiability**. In *Abstracts of the 5th International Planning Competition*. 2006.

KAUTZ, Henry; SELMAN, Bart. **Pushing the Envelope: Planning, Propositional Logic, and Stochastic Search**. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence and the Eighth Innovative Applications of Artificial Intelligence Conference*, edited by Howard Shrobe and Ted Senator, 1194–1201. Menlo Park, California: AAAI Press. 1996. citeseer.ist.psu.edu/kautz96pushing.html.

— — —. **Unifying SAT-Based and Graph-Based Planning**. In *Workshop on Logic-Based Artificial Intelligence, Washington, DC, June 14–16, 1999*, edited by Jack Minker. College Park, Maryland: Computer Science Department, University of Maryland. citeseer.ist.psu.edu/kautz99unifying.html.

KOTTLER, Stephan; KAUFMANN, Michael; SINZ, Carsten. **A new bound for an NP-hard subclass of 3-SAT using backdoors**. In *Proceedings of the 11th International Conference on Theory and Applications of Satisfiability Testing*, 161–67. SAT'08. Berlin, Heidelberg: Springer-Verlag. 2008. <http://dl.acm.org/citation.cfm?id=1789854.1789870>.

MARQUES-SILVA, J. P.; LYNCE, I.; MALIK, S. **Conflict-Driven Clause Learning SAT Solvers**. In *Handbook of Satisfiability*, edited by Armin Biere, Marijn J. H. Heule, Hans van Maaren, and Toby Walsh, 185:131–53. Frontiers in Artificial Intelligence and Applications. IOS Press. 2009.

MINOUX, Michel. **LTUR: a simplified linear-time unit resolution algorithm for horn formulae and computer implementation**. *Information Processing Letters* 29 (1): 1–12. 1998. [https://doi.org/10.1016/0020-0190\(88\)90124-X](https://doi.org/10.1016/0020-0190(88)90124-X).

NEMENYI, P. **Distribution-Free Multiple Comparisons**. Princeton University. 1963.

PORSCHEN, Stefan; SCHMIDT, Tatjana; SPECKENMEYER, Ewald. **On Some Aspects of Mixed Horn Formulas**. In *Theory and Applications of Satisfiability Testing - SAT 2009*, edited by Oliver Kullmann, 5584:86–100. Lecture Notes in Computer Science. Springer Berlin Heidelberg. 2009. https://doi.org/10.1007/978-3-642-02777-2_11.

PORSCHEN, Stefan; SPECKENMEYER, Ewald. **Worst Case Bounds for Some NP-Complete Modified Horn-SAT Problems**. In *Theory and Applications of Satisfiability Testing*, edited by HolgerH Hoos and DavidG Mitchell, 3542:251–62. Lecture Notes in Computer Science. Springer Berlin Heidelberg. 2005. https://doi.org/10.1007/11527695_20.

— — —. **Satisfiability of mixed Horn formulas**. *Discrete Appl. Math.* 155 (June): 1408–19. 2007. <https://doi.org/10.1016/j.dam.2007.02.010>.

PRESTWICH, Steven. **CNF Encodings**. In *Handbook of Satisfiability*, edited by Armin Biere, Marijn J. H. Heule, Hans van Maaren, and Toby Walsh, 185:75–97. Frontiers in Artificial Intelligence and Applications. IOS Press. 2009.

RANDERATH, Bert; SPECKENMEYER, Ewald; BOROS, Endre; HAMMER, Peter; KOGAN, Alex; MAKINO, Kazuhisa; SIMEONE, Bruno; CEPEK, Ondrej. **A Satisfiability Formulation of Problems on Level Graphs.** *Electronic Notes in Discrete Mathematics* 9 (0): 269–77. 2001. [https://doi.org/http://dx.doi.org/10.1016/S1571-0653\(04\)00327-0](https://doi.org/http://dx.doi.org/10.1016/S1571-0653(04)00327-0).

SELMAN, Bart; LEVESQUE, Hector J.; MITCHELL, D. **A New Method for Solving Hard Satisfiability Problems.** In *Proceedings of the Tenth National Conference on Artificial Intelligence*, edited by Paul Rosenbloom and Peter Szolovits, 440–46. Menlo Park, California: AAAI Press. 1992. citeseer.ist.psu.edu/selman92new.html.

SIDERIS, Andreas; DIMOPOULOS, Yannis. **Constraint Propagation in Propositional Planning.** In *ICAPS*, edited by Ronen I. Brafman, Hector Geffner, Jörg Hoffmann, and Henry A. Kautz, 153–60. AAAI. 2010.

WELD, Daniel S. **Recent Advances in AI Planning.** *AI Magazine* 20 (2): 93–123. 1999. citeseer.ist.psu.edu/article/weld99recent.html.

ASSIMILAÇÃO DE DADOS POR REDES NEURAIS EM AUSÊNCIA PARCIAL DE OBSERVAÇÕES

Data de aceite: 01/09/2021

Data de submissão: 06/07/2021

Rosangela Saher Corrêa Cintra

Instituto Nacional de Pesquisas Espaciais
São José dos Campos – São Paulo
<http://lattes.cnpq.br/8185155301349092>

Haroldo Fraga de Campos Velho

Instituto Nacional de Pesquisas Espaciais
São José dos Campos – São Paulo
<http://lattes.cnpq.br/5142426481528206>

Esta é uma versão em português de um artigo publicado na Conference of Computational Interdisciplinary Science (CCIS-2019): <https://proceedings.science/ccis-2019/papers/data-assimilation-by-neural-network-under-missing-data>.

RESUMO: Previsão numérica do tempo (PNT) usa modelos atmosféricos de circulação geral (MACG) para prever condições futuras da atmosfera. O processo de PNT é realizado inserindo dados de observação ao modelo computacional para calcular as condições iniciais - também chamado de análise. Tal procedimento é chamado de assimilação de dados (AD). Várias técnicas tem sido desenvolvidas para AD. Filtro de Kalman por conjunto (ensemble), métodos variacionais e filtro de partículas são exemplos de técnicas de AD usadas em centros operacionais de PNT. Entretanto, tais metodologias requerem grande esforço computacional. Aqui, um conjunto de redes neurais artificiais (RNA) - perceptron

de multicamadas com aprendizado por retro-propagação - é configurado para emular o Local Ensemble Transform Kalman Filter (LETKF) aplicado ao modelo global 3D SPEEDY (Simplified Parameterizations Primitive-Equation Dynamics). A novidade é calcular a análise por RNA quando algumas observações não estão disponíveis em algum ciclo de AD ou em vários ciclos. Uma comparação entre as análises produzidas por LETKF e RNA é realizada.

PALAVRAS-CHAVE: Assimilação de dados, modelo atmosférico, rede neural artificial, filtro de Kalman por conjunto.

DATA ASSIMILATION USING ARTIFICIAL NEURAL NETWORKS: MISSING OBSERVATION

ABSTRACT: Numerical weather prediction (NWP) uses atmospheric general circulation models (AGCMs) to predict the weather future conditions. The process is done inserting observation data into computer model to compute initial conditions - also named "analysis". Such feature is called data assimilation (DA). Several techniques have been developed for DA. Ensemble Kalman filter, particle filter, variational scheme, are examples of DA methods used in the operational NWP centers. However, such methodologies require a high computational effort. Here, a set of artificial neural networks (ANNs) - multi-layer perceptron with back propagation learning - is configured to emulate the Local Ensemble Transform Kalman Filter (LETKF) applied to the SPEEDY model. The novelty is to perform the analysis by ANN when some observations are missing in a DA cycle, or several cycles. A comparison between

analysis produced by the LETKF and ANN is carried out. The numerical experiment was done at January 1985.

KEYWORDS: Data assimilation, atmospheric model, artificial neural network, ensemble Kalman filter.

1 | INTRODUÇÃO

O procedimento para determinar a condição inicial em centros operacionais de previsão é chamado de assimilação de dados (AD), onde os campos da previsão prévia (background) são combinados com observações para produzir a análise - a nova condição inicial. Há muitos esquemas diferentes de AD. Uma revisão dos métodos mais comuns de AD pode ser encontrada na literatura, em particular nos livros de Daley [7] e Kalnay [16].

Dois esquemas de DA são avaliados aqui: *Local Ensemble Transform Kalman filter* (LETKF) [14] e a abordagem baseada em redes neurais artificiais (RNA) para emular o LETKF [4, 5, 11, 19]. A principal vantagem do uso de RNA é diminuir o esforço computacional.

Experimentos numéricos são realizados usando observações sintéticas de estações meteorológicas de superfície (dados a cada 6 horas/dia) e sondagem da alta atmosfera (dados a cada 12 horas/dia). A pesquisa investiga o comportamento das técnicas de AD com ausência de observações nos ciclos de AD. Cada dia uma sequência de análises para assimilações a cada 4 horas: 00:00, 06:00, 12:00, e 18:00 UTC. A malha de observações reproduz as estações da Organização Meteorológica Mundial (OMM)¹ e das sondagens por radiossondas.

O modelo atmosférico SPEEDY (*Simplified Parameterizations Primitive Equation Dynamics*) [2] simula a dinâmica tri-dimensional (3D) da atmosfera - na verdade, o domínio espacial considerado é até o nível mais alto da troposfera - com parametrizações físicas simplificadas. São executados 44 ciclos de AD (análise - execução do SPPED { previsão) com as técnicas de LETKF e RNA, onde alguns ciclos de AD têm ausência parcial de observações.

2 | ASSIMILAÇÃO DE DADOS

Considerando um sistema não linear com um vetor de estado n -dimensional x^f e um vetor m -dimensional de observações y^o evoluindo de acordo com:

$$x_{k+1}^f = f(x_k^f, t_k) + q_k \quad (1)$$

$$y_k^o = h(x_k^f, t_k) + v_k \quad (2)$$

onde q_k e v_k são termos de ruídos gaussianos, $f(\cdot)$ e $h(\cdot)$ são equações matemáticas para previsão e observação, respectivamente. Previsão é passo para identificar o estado

¹ WMO: World Meteorological Organization.

x_{k+1}^f de um sistema a partir do último estado por um modelo numérico de previsão de tempo. Do ponto de vista matemático, o processo de assimilação pode ser representado por:

$$x_k^a = x_{k+1}^f + W[y_k^o - H(x_{k+1}^f)], \quad (3)$$

$$W = (HP^fH^T + R). \quad (4)$$

A equação (3) é o passo de análise, onde H é o operador de observação, W é a matriz de peso, calculado das matrizes de erros de covariância P^f e R , representando erros do modelo e de observações, respectivamente. A abordagem de assimilação de dados com RNA implementa a função:

$$x^a = F_{NN}(y^o; x^f) \quad (5)$$

onde F_{NN} é o processo de assimilação de dados, y^o representa as observações, x^f é a previsão do modelo, frequentemente chamada de primeira estimativa (*first guess*) e x^a é a análise.

2.1 Local ensemble transform kalman filter - LETKF

O filtro de Kalman por conjunto (EnKF: Ensemble Kalman filter) é uma abordagem bayesiana proposta por Evensen [8]. Vários esquemas de EnKF foram desenvolvidos e o LETKF (local ensemble transform Kalman filter) é um destes esquemas [14].

A ideia básica do LETKF é realizar a análise para cada ponto de grade simultaneamente usando as variáveis de estado e todas as observações na região centrada para um dado ponto. A estratégia local separa grupos de observações vizinhas ao redor do ponto de grade do modelo. Cada ponto de grade tem um subdomínio local; o número de vetores locais é o mesmo número de pontos de grade globais [17].

3 | REDES NEURAIS ARTIFICIAIS (RNAs)

Uma RNA é composta de unidades simples de processamento (neurônios) para calcular certas funções matemáticas e consiste de neurônios artificiais - ou nós - interconectados. Cada neurônio artificial tem uma ou mais entradas e saídas. As conexões entre os neurônios armazenam uma soma ponderada, chamados de pesos sinápticos. Em processamento de RNA, as entradas são multiplicadas pelos pesos alimentando uma função de ativação.

Esta função ativa ou inibe o próximo neurônio. Matematicamente, pode-se descrever o i -ésimo neurônio da seguinte forma:

$$\begin{aligned} \text{Soma da entrada: } u_i &= \sum_{j=1}^p w_{ij}x_j + b_i \\ \text{Saída neurônio } y_i &= \varphi(u_i) \end{aligned} \quad (6)$$

onde x_1, x_2, \dots, x_p são as entradas; w_{i1}, \dots, w_{ip} são os pesos sinápticos; b_i é o viés, u_i é a saída da combinação linear; $\varphi(\cdot)$ é a função de ativação e y_i é a i -ésima saída do

neurônio e p é o número de entradas do i -ésimo neurônio.

A rede de alimentação progressiva (*feed-forward*), que processa na direção da entrada para saída, tem uma estrutura de camadas. A primeira camada de uma RNA é chamada de camada de entrada, as camadas intermediárias são chamadas de camadas escondidas e a última camada é chamada de camada de saída. Alguns parâmetros, como o número de camadas e a quantidade de neurônio em cada camada, definem a topologia da rede neural, mas outros parâmetros são também necessários de serem identificados, tais como a taxas de aprendizado e de momentum. Os parâmetros da RNA são determinados pela natureza do problema.

Há duas fases distintas no uso de redes neurais supervisionadas: a fase de treinamento (processo de aprendizado) e a fase de execução (ativação ou generalização). A fase de treinamento das RNAs consiste de um processo iterativo de ajuste dos pesos de conexão para o melhor desempenho da rede, estabelecido pelo mapeamento dos pares de entrada e saída. O algoritmo de aprendizado é o conjunto de procedimentos para o ajuste dos pesos. Cada passo da interação é denominado época". O processo de treinamento iterativo continua até ser atingido um critério de parada, que pode ser um erro mínimo estabelecido ou um número máximo de épocas. Uma vez treinada, os pesos de uma RNA supervisionada são fixos e a rede estará pronta para receber novas entradas (que não pertencem ao conjunto de treinamento) para o cálculo de saídas correspondentes.

O perceptron de múltiplas camadas (PMC) com o algoritmo de aprendizado retro-propagação de erros é a RNA usada em nosso estudo, com ao menos uma camada intermediária [12].

4 | MODELO SPEEDY

O modelo SPEEDY [18] é um simulador de circulação atmosférico global (SCAG) desenvolvido pelo Centro Internacional de Física Teórica (ICTP: *International Centre for Theoretical Physics*) em Triesty, Itália, com núcleo dinâmico espectral com um conjunto simplificado de parametrizações físicas. As variáveis dinâmicas para as equações primitivas são integradas pelo método numérico espectral na horizontal para cada nível vertical - mais detalhes em [2].

O modelo SPEEDY T30L7 é executado com truncamento de 30 termos da expansão espectral e (T30) com 7 níveis na vertical (L7): 100, 200, 300, 500, 700, 850 e 925 hPa. As coordenadas verticais são definidas no sistema sigma: ($\sigma = p/ps$), onde ps é a pressão da superfície. As coordenadas horizontais são latitude e longitude sobre uma grade gaussiana, com espaçamento regular com 96 pontos zonais (longitude) e 48 pontos meridionais (latitude). As variáveis prognósticas são a temperatura absoluta (T), pressão de superfície (ps), componentes zonal e meridional do vento (u ; v) e a umidade específica (q).

5 I PMC-AD: ASSIMILAÇÃO DE DADOS COM AUSÊNCIA DE OBSERVAÇÕES

A configuração da RNA para este experimento é um conjunto de perceptrons de multicamadas, referenciados como PMC-DA, definido no experimento descrito por Cintra e Campos Velho [6]. A topologia da RNA tem as seguintes características:

1. Dois nós de entrada, um nó para o vetor de observação meteorológica e outro nó para o vetor de previsão de 6-hours do modelo SPEEDY;
2. Um nó de saída para o resultados do vetor de análise;
3. Uma camada escondida com onze neurônios;
4. Função tangente hiperbólica como função de ativação;
5. Taxa de aprendizado h é definida para cada RNA-PMC;
6. Fase de treinamento para quando o erro atinge 10^5 ou após 5000 épocas - com o critério que ocorre primeiro.

Os vectores de entrada representam o ponto de grade para um a simples variável com o corresponde par observação - previsão. No algoritmo de treinamento, PMC-DA calcula a saída e a compara com o vetor de análise do LETKF - o dado alvo. Os vetores de saída representam os valores de análise para o ponto de grande.

O esquema PMC-AD foi desenvolvido com um conjunto de trinta RNAs: o planeta Terra é dividido em 6 regiões com cinco variáveis prognósticas (ps ; u ; v ; T ; q). Deste modo, cada região e para cada variável tem-se uma RNA específica para assimilação de dados. Todas as RNAs tem uma camada escondida com o mesmo número de neurônios para todas as regiões, mas com diferentes pesos de conexão. O esquema de PMC-DA divide o globo inteiro em seis regiões: para o hemisfério Norte, 90° N e três regiões longitudinais de 120° cada; para o hemisfério Sul, 90° S e três regiões longitudinais de 120° cada. todas as regiões tem o mesmo tamanha, mas o número de observações é distinto para cada região - ver Figura 1. Esta divisão em regiões é aplicada somente para PMC-AD e não para o procedimento com LETKF.

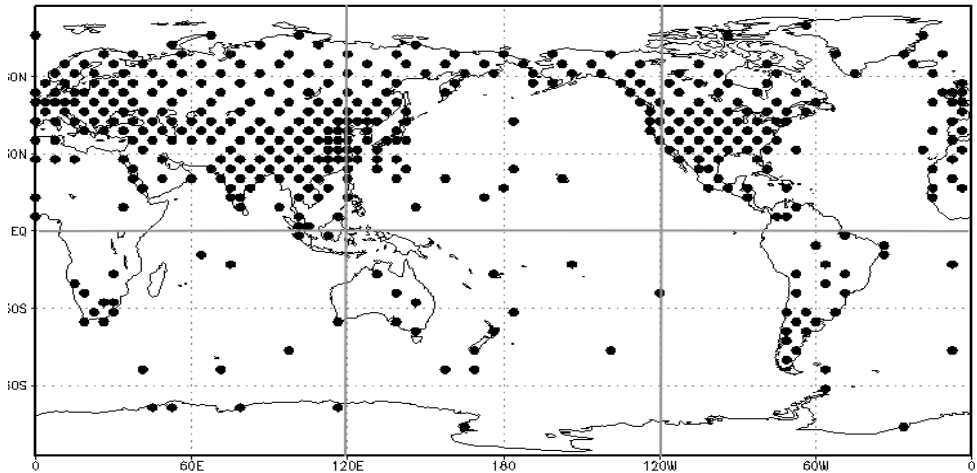


Figura 1: Localizações de observações na área global. Os pontos na figura representam estações de coleta de dados de radiossonda (~ 415).

O esquema PMC-AD é projetado para emular a análise do LETKF do cálculo da condição inicial para o modelo SPEEDY. a análise do LETKF é o campo médio das análises do conjunto (*ensemble*).

As matrizes de covariâncias para os níveis superiores e de superfície para executar o sistema LETKF, bem como as condições de contorno, são as mesmas usadas nos experimentos de Miyoshi [17] e Cintra [6]. A assim chamada “verdade” (ou controle) do modelo é gerada pela simulação do modelo SPEEDY sem ruído nos dados. Os ciclos de assimilação são considerados em 4 períodos em cada dia (00:00, 06:00, 12:00, 18:00 UTC). As observações sintéticas são geradas lendo o campo “verdade” do modelo SPEEDY e adicionando um ruído aleatório sobre as variáveis meteorológicas. A localização das observações estão sobre os pontos de grade do modelo SPEEDY. Uma máscara de observação é projetada, adicionando uma marca positiva para o ponto de grade onde a observação deveria ser considerada. As localizações são similares às estações e de observações de radiossondas da WMO - ver Figura 1. Exceto para observações de *ps*, as outras observações são também descritas para os sete níveis superiores.

O esquema de assimilação de dados PMC-AD não tem matrizes de erro de covariância para distribuir a influência dos dados de observação. Conseqüentemente, é necessário calcular a influência das observações no entorno do ponto de grade. Este cálculo é baseado na distância do ponto de grade relacionado às observações no interior de um determinado raio de influência (inicialmente: $y = 0$)

$$\hat{y}_{i\pm m; j\pm m; k\pm m}^o = \frac{y_{ijk}^o}{(6-\gamma_{r+1})r_{ijk}^2} + \sum_{l=1}^6 \alpha_l \frac{y_{i\pm m; j\pm m; k\pm m}^o}{r_{ijk}^2} \quad (7)$$

$(m = 1; 2; \dots; M)$

$$\alpha_1 = \begin{cases} 0 & (\text{se não há observação}) \\ 1 & (\text{se existe observação e } \gamma_{r+1} = \gamma_r + 1) \end{cases} \quad (8)$$

onde \hat{y}^o é a observação ponderada, M é o número de camadas discretas consideradas no entorno do ponto de grade, $r_{ijk}^2 = (x_p - y_i^o)^2 + (y_p - y_j^o)^2 + (z_p - y_k^o)^2$, onde $(x_p; y_p; z_p)$ é a coordenada do ponto de grade e (y_i^o, y_j^o, y_k^o) é a coordenada da observação, y_r é um contador de pontos de grade das observações no entorno de (y_i^o, y_j^o, y_k^o) . Se $y_{r+1} = 6$, não há influência a ser considerada. Daqui para frente, os valores computados de Eq. (7) são referenciados como *pseudo-observação*.

O processo de treinamento foi feito como no experimento realizado por Cintra e Campos Velho [6]. Com as redes treinadas, o sistema está pronto para ser usado no processo de assimilação de dados. A ativação da PMC-AD tem valores de entrada (previsão e observação) para cada ponto de grade. Os experimentos de assimilação de dados com PMC-AD é realizado para 11 dias (44 ciclos). A data de início foi 10 de janeiro de 1985 00UTC até 20 de janeiro de 1985 18UTC.

Os ciclos de AD iniciam com uma máscara global completa das observações, para as cinco variáveis prognósticas, com quatro variáveis para o ar superior para os sete níveis verticais. Durante três dias (12 ciclos), as análises para o modelo SPEEDY foram calculadas de 415 estações para todas as variáveis (12035 observações) - Fase-1. Nos próximos dois dias (12/Jan e 13/Jan) com oito ciclos, não há observações para U, V e Q, novamente com 415 estações com temperatura e pressão de superfície somente (3320 observações) - Fase-2. Para os próximos cinco dias (20 ciclos), não há observações do conjunto de Estações - Fase-3.

6 | RESULTADOS E CONCLUSÃO

Os valores de entrada e saída das variáveis prognósticas (ps, u, v, T, q) são processados sobre os pontos de grade do modelo para integração no tempo. Levando em conta as pseudo-observações (Eq. 7), com dois espaçamentos de grade ($M = 2$) como raio de influencia no entorno do ponto de grade, como considerado por Cintra e Campos Velho [4]. Os resultados mostram a comparação dos campos de análise gerados pelos esquemas PMC-AD e LETKF com os campos verdade.

As diferenças entre as análises para observação global de umidade (q) para 950 hPa é apresentado na Figura 2, considerado o ciclo de assimilação executado em 10 Janeiro 06 UTC: Fase-1. Figura 3 mostra a umidade com ausência de observações ($u; v; q$) na Fase-2 (13/Janeiro _as 12 UTC).

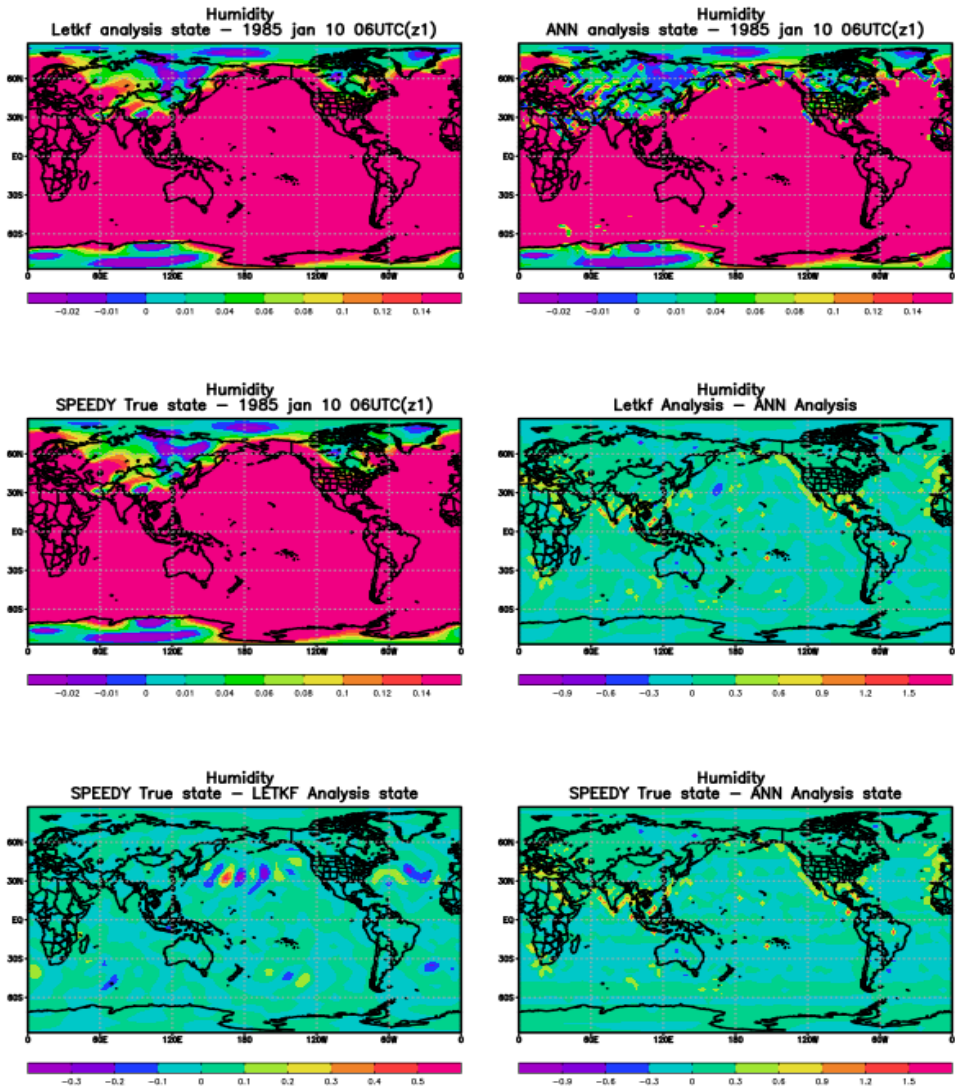


Figure 2: Umidade (kg/kg) em 10/Janeiro _as 06UTC: análises LETKF, PMC-AD e campo Verdade -- e diferenças: (LETKF e PMC-DA), (Verdade e LETKF), (Verdade e PMC-AD).

Para a temperatura absoluta e pressão de superfície no nível de 950 hPa - ver Figuras 4 e 5 para resultados na Fase-2 (13/Janeiro _as 12 UTC), com ausência de observações para (u; v; q), para 415 estações para temperatura (7 camadas verticais) e pressão de superfície. Os campos de temperatura estão unidades de graus Kelvin (valores acima de 273,15) com diferenças até 10 K em alguns pontos. Para o campo de pressão de superfície, Há diferenças em

todo o domínio, porque há pseudo-observações sobre todo o domínio.

Na Fase-3, LETKF e PMC-DA são executados com observações ausentes em alguns pontos. Figura 6 mostra a pressão de superfície em 18/Janeiro às 00UTC, último dia dos ciclos de observação. Para a Fase-3, a máscara de observação está completa. similar à Fase-1. O modelo tem o mesmo comportamento que o experimento prévio de Cintra e Campos Velho [6] - ver Figura 7 para análise de temperatura para 19/Janeiro às 00UTC.

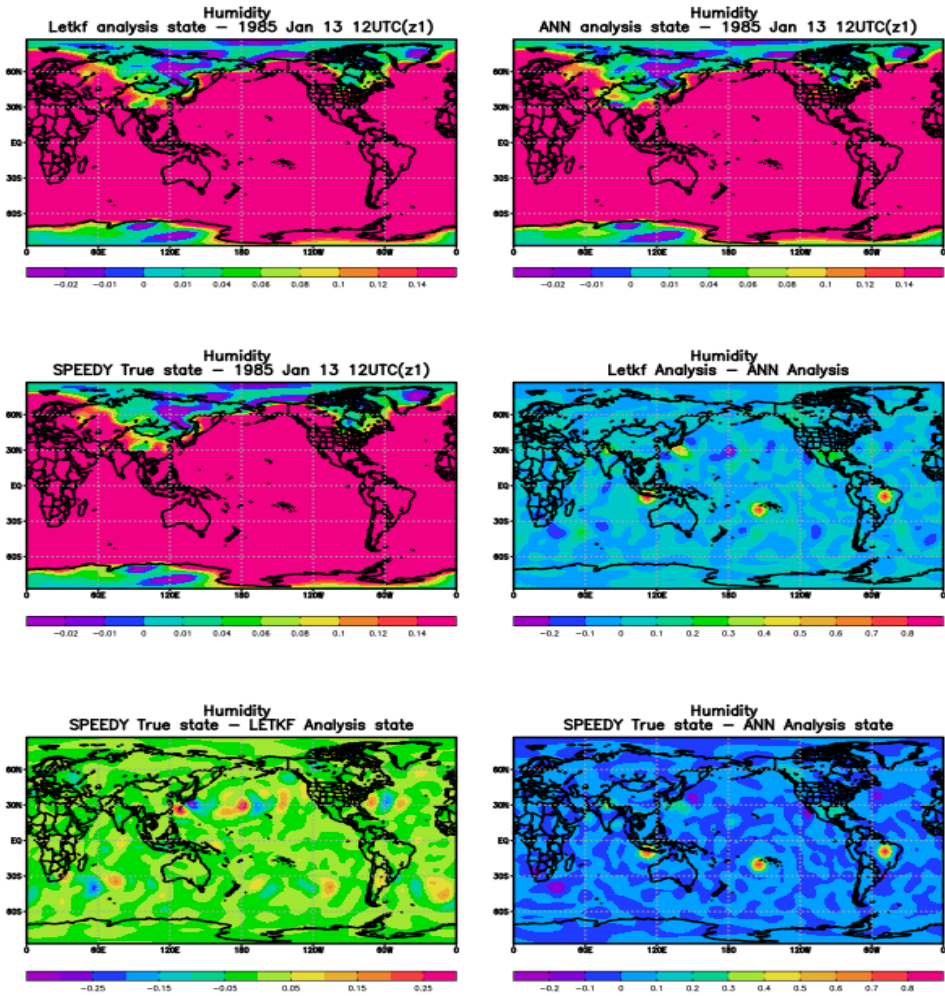


Figure 3: Umidade (kg/kg) em 13/Janeiro _as 12UTC: análises LETKF, PMC-AD e campo Verdade -- e diferenças: (LETKF e PMC-DA), (Verdade e LETKF), (Verdade e PMC-AD).

As análises são tentativas de aproximar as variáveis do modelo à realidade. Menor quantidade de observações significa uma análise pior. Ausências de observações podem ocorrer em diferentes ciclos de assimilação. A técnica de PMC-DA foi treinada com 2

espaçamentos de grade como raio de influência para as observações ($M = 2$ na Eq. (7)). Figuras 3, 5 e 7 mostram uma boa concordância entre as análises do LETKF e PMC-DA. Figuras 2, 4 e 6 mostram algum desacordo entre as análises obtidas com as duas metodologias. Observando a Figura 6, pode-se perceber diferenças significativas sobre a região da zona de interface entre Europa-Ásia e principalmente sobre os oceanos.

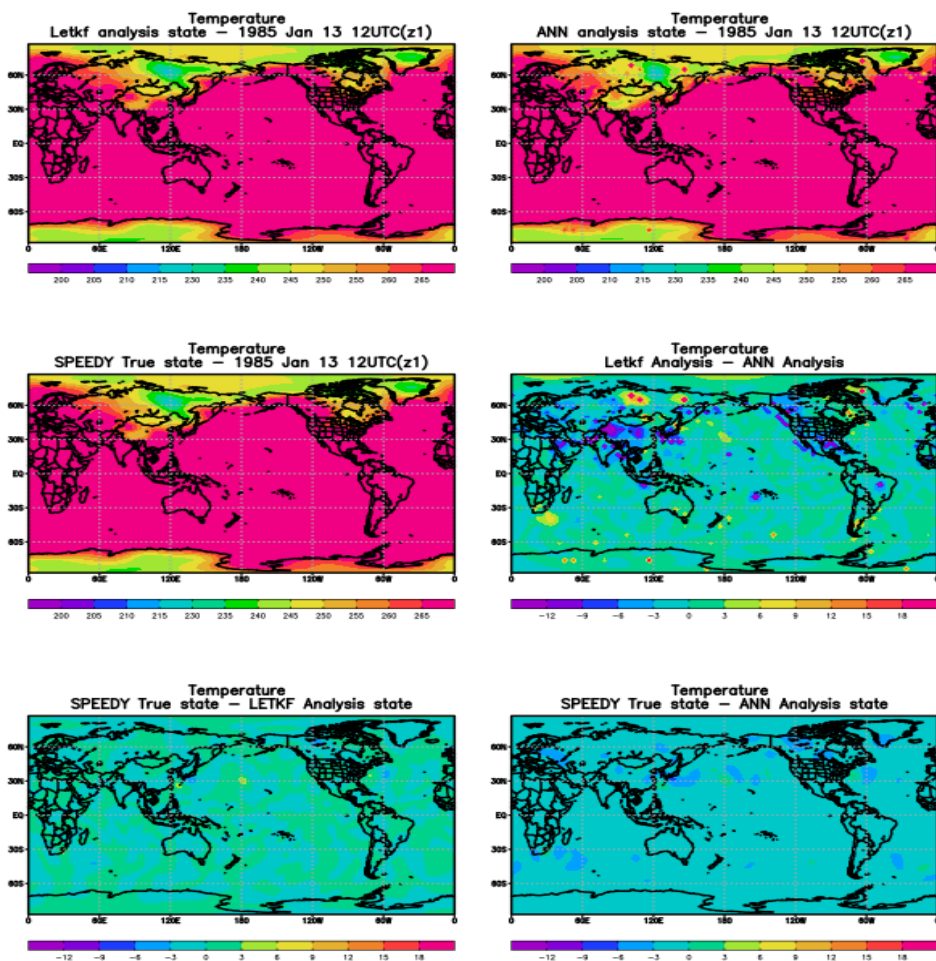


Figure 4: Temperatura absoluta (K) em 13/Janeiro _as 12UTC: análises LETKF, PMC-AD e campo Verdade -- e diferenças: (LETKF e PMC-AD), (Verdade e LETKF), (Verdade e PMC-AD).

Há diferenças em vários pontos do domínio - ver Figura 2 (análise via RNA) sobre o Alaska e Vladivostok (pontos vermelho) e Figura 4 (Norte da Rússia e sobre o mar ao Norte do Canadá e Groenlândia), indicando que o raio de influência da observação não foi efetivo o suficiente para calcular de forma adequada a pseudo-observação com $M = 2$. Uma

maneira de superar as diferenças apontadas é aumentar a influência da observação ($M \geq 3$).

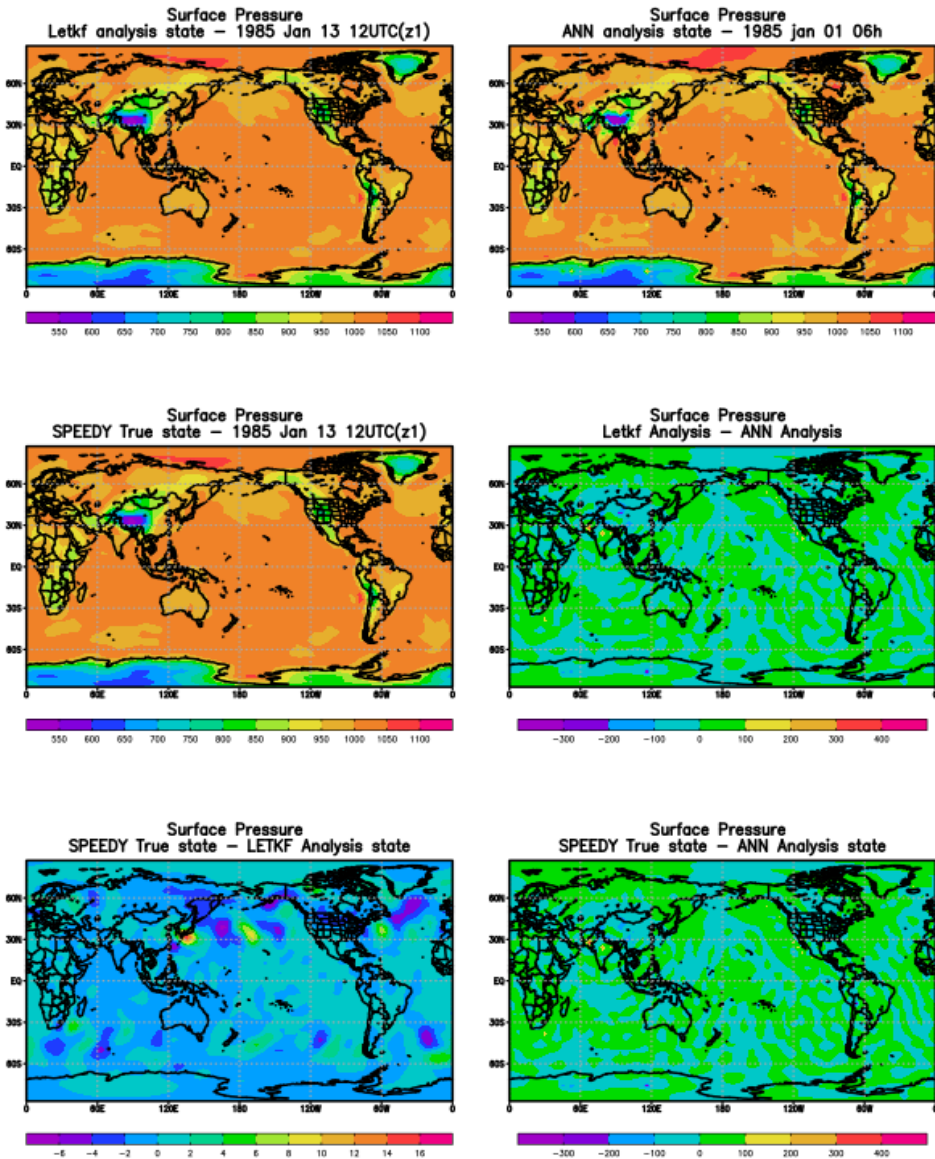


Figure 5: Pressão de superfície (hPa) em 13/Janerio_as 12UTC: análises LETKF, PMC-DA e campo Verdade -- e diferenças: (LETKF e PMC-DA), (Verdade e LETKF), (Verdade e PMC-DA).

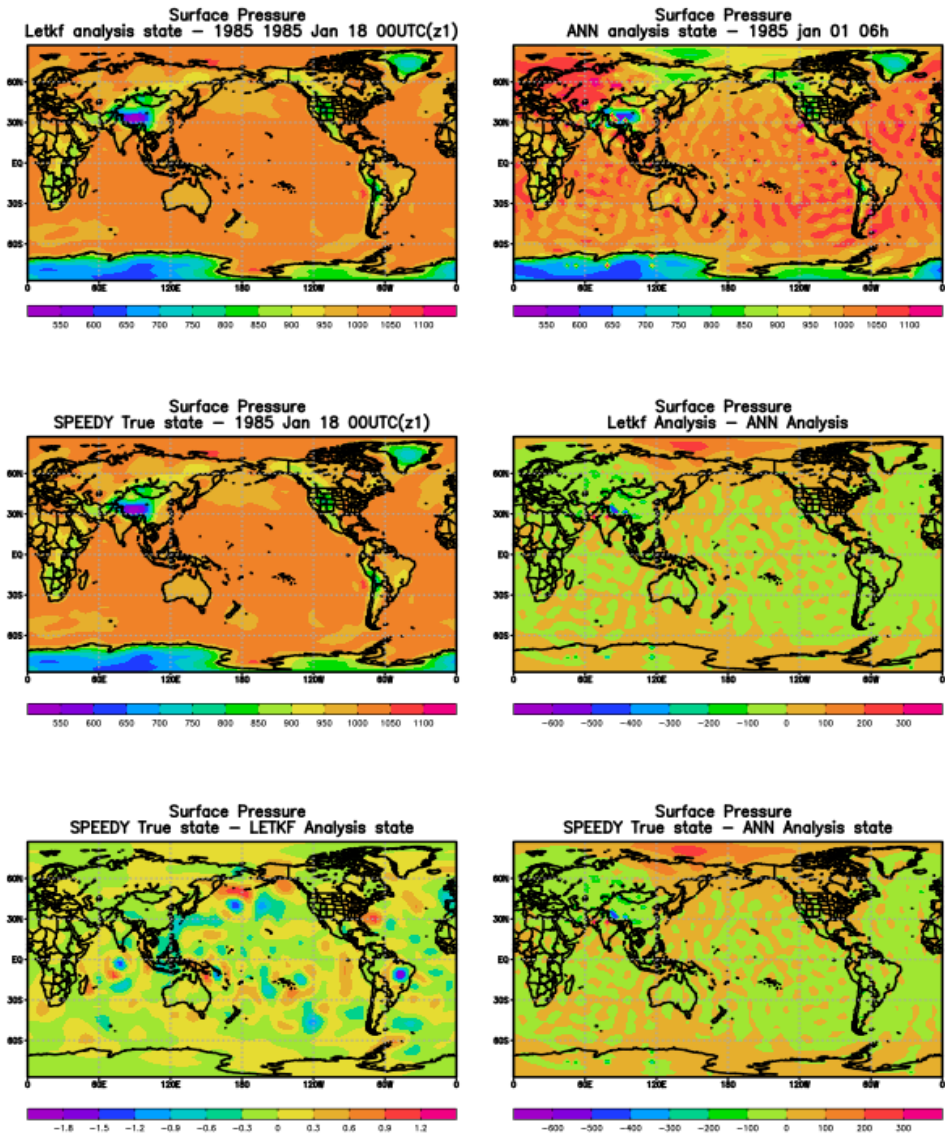


Figure 6: Pressão de superfície em 18/Janeiro _as 00UTC: análises LETKF, PMC-DA e campo Verdade I e diferenças: (LETKF e PMC-DA), (Verdade e LETKF), (Verdade e PMC-DA).

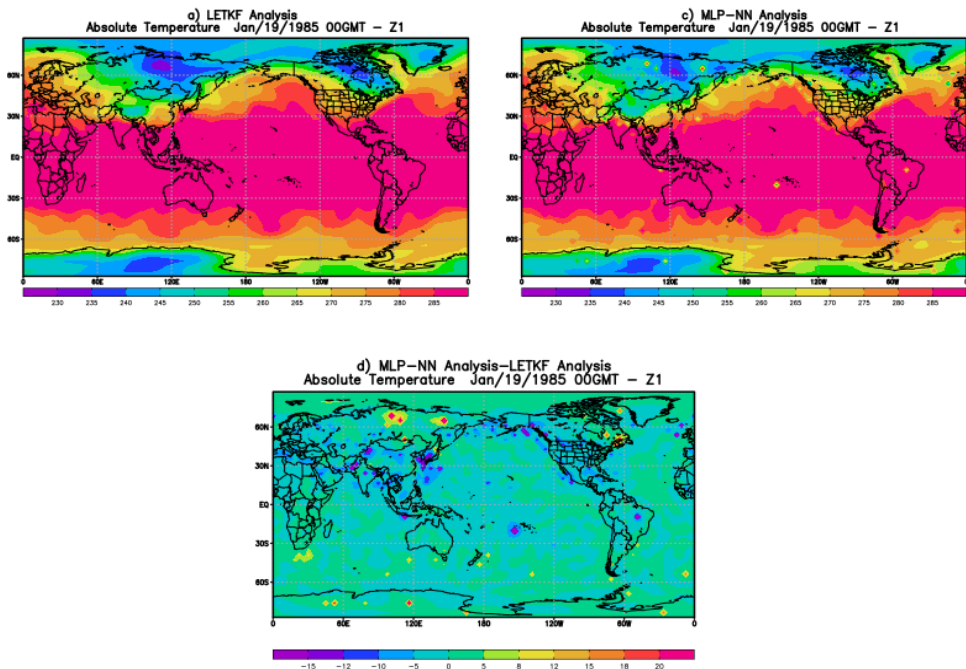


Figure 7: Temperatura absoluta em 19/Janeiro _as 00UTC: análises LETKF e PMC-DA -- e diferenças: (LETKF e PMC-DA).

AGRADECIMENTOS

Os autores agradecem à Fapesp (Proc. 2016/14393-2) e ao CNPq (Proc.: 312924/2017-8), agências brasileiras de suporte à pesquisa.

REFERÊNCIAS

- [1] Bishop, H. C., Etherton, B. J., Majumdar, S. J.: **Adaptive sampling with the ensemble transform Kalman filter**. Part I: Theoretical aspects. *Monthly Weather Review*, 129, 420{436, 2001.
- [2] Bourke, W.: **A multilevel spectral model: I. formulation and hemispheric integrations**. *Monthly Weather Review*, 102, 687{701, 1974.
- [3] Burgers, G., van Leeuwen, P., Evensen, G.: **Analysis scheme in the ensemble Kalman filter**. *Monthly Weather Review*, 126, 1719{1724, 1998.
- [4] Cintra, R. S., Campos Velho, H. F.: **Global Data Assimilation using Artificial Neural Networks in SPEEDY Model**. *Proceedings of the International Symposium on Uncertainty Quantification and Stochastic Modelling*. pp 648-654, Maresias, SP, Brazil, 2012.

- [5] Cintra, R. S. C.; Campos Velho, H. F.; Anochi J.; Cocke S.: **Data Assimilation by Artificial Neural Networks for the global FSU atmospheric model: Surface Pressure**. 2nd Latin-American Congress on Computational Intelligence (LA-CCI), Curitiba, Brazil. CBIC and LA-CCI 2015.
- [6] Cintra, R. S. C.; Campos Velho, H. F.: **Data Assimilation by Artificial Neural Networks for an Atmospheric General Circulation Model**. In: Advanced Applications for Artificial Neural Networks (Org.: Adel El- Shahat), IntechOpen, (2018) { DOI: 10.5772/intechopen.70791.
- [7] Daley, R.: **Atmospheric Data Analysis**. Cambridge University Press, New York, USA, 1991.
- [8] Evensen, G.: **Sequential data assimilation with a nonlinear quasigeostrophic model using monte carlo methods to forecast error statistics**. Journal of Geophysics Research, 99(10), 143{162, 1994.
- [9] Evensen, G.: **The ensemble Kalman filter: theoretical formulation and practical implementation**. Ocean Dynamics, 53, 343{367, 2003.
- [10] Furtado, H. C. M.; Campos Velho, H. F.; Macau, E. E. N.: **Data assimilation: Particle filter and artificial neural networks**. Journal of Physics. Conference Series (Online), v. 135, p. 012073, 2008.
- [11] Harter, F. P., Campos Velho, H. F.: **New approach to applying neural network in nonlinear dynamic model**. Applied Mathematical modeling,32(12), 2621{2633, 2008.
- [12] Haykin, S.: **Neural Networks: A Comprehensive Foundation**. Prentice Hall, 2nd Edition, 1999.
- [13] Houtekamer, P. L. and Mitchell, H. L.: **Data assimilation using an ensemble Kalman filter technique**. Monthly Weather Review, 126, 796--811, 1998.
- [14] Hunt, B., Kostelich, E. J., Szunyogh, I.: **Efficient data assimilation for spatiotemporal chaos: a local ensemble transform Kalman filter**. Physica D, 230, 112-126. 2007.
- [15] Kalman, R. E., Bucy, R. S.: **New results in linear filtering and prediction theory**. Trans. of the ASME Journal of Basic Engineering, 83(Series D), 95{108, 1961.
- [16] Kalnay, E.: **Atmospheric Modeling, Data Assimilation and Predictability**. 2d ed., Cambridge University Press, New York, 2003.
- [17] Miyoshi, T.; Yamane, S: **Local ensemble transform Kalman filtering with an AGCM at a T159/L48 resolution**. Monthly Weather Review, 135, 3841-3861, 2007.
- [18] Molteni, F.: **Atmospheric simulations using a GCM with simplified physical parametrizations: model climatology and variability in multidecadal experiments**. Climate Dynamics, 20, 175{191, 2003.
- [19] Nowosad, A., Rios Neto, A., Campos Velho, H.: **Data assimilation in chaotic dynamics using neural networks**. International Conference on Nonlinear Dynamics, Chaos, Control and Their Applications in Engineering Sciences, 212-221, 2000.

- [20] Ott, E., Hunt, B. R., Szyniogh, I., Zimin, A. V., Kostelich, E. J., Corazza, M., Kalnay, E., Patil, D. J., York, J.: **A local ensemble Kalman filter for atmospheric data assimilation.** Tellus A, 56, 415-428, 2004.
- [21] Widrow, B., Ho, M.: **Adaptive switching circuits.** IRE WESCON Conv. Record, Pt.4.94,95, 96{104, 1960.
- [22] Whitaker, J. S., T. H. Hamill: **Ensemble data assimilation without perturbed observations.** Monthly Weather Review, 130, 1913{1924, 2002.

CONCEPÇÃO E OTIMIZAÇÃO DE UMA CLASSIFICAÇÃO OBJETIVA PARA SALAS DE ÓPERA UTILIZANDO MÉTODOS COMPUTACIONAIS

Data de aceite: 01/09/2021

Data de submissão: 18/06/2021

Marco Antônio M. Vecci

Universidade Federal de Minas Gerais (UFMG)
– Departamento de Engenharia de Estruturas
e DEES)
Belo Horizonte – Minas Gerais
<http://lattes.cnpq.br/9538829669531950>

Calebe Giaculi Júnior

Instituto Federal de Minas Gerais - Campus
Bambuú (IFMG) – Departamento de Engenharia
e Computação
Bambuú – Minas Gerais
<http://lattes.cnpq.br/4325582561181007>

Bruno Alberto Soares Oliveira

Universidade Federal de Minas Gerais (UFMG)
– Programa de Pós-Graduação em Engenharia
Elétrica
Bambuú – Minas Gerais
<http://lattes.cnpq.br/2112685965030446>

RESUMO: Criar a classificação de salas de ópera é uma tarefa não exata, pois depende de diversos fatores subjetivos. Baseado em uma Classificação Subjetiva (Hidaka e Beranek, J. Acoust. Soc. Am. 107, 340–354 (2000)), criou uma Classificação Objetiva utilizando o método de tomada de decisão multicritério *PROMETHEE II*, utilizando como critérios os parâmetros acústicos objetivos para 22 salas de ópera apresentados em Beranek (Springer Verlag, NY, 2004), sendo que 13 também foram incluídas no

estudo realizado por Hidaka e Beranek (2000). Assim, basearam-se em um ranking subjetivo simplificado, o qual considerou apenas as 13 salas que continham dados acústicos disponíveis, o que possibilitou a comparação e a validação da classificação objetiva proposta. São introduzidos os principais conceitos e apresentadas as adequações necessárias para que o trabalho pudesse ser realizado. Como resultado inicial se obteve um ranking objetivo com 4 salas de ópera classificadas com posições iguais e 5 salas com diferença de até duas posições em comparação com a classificação realizada por Hidaka e Beranek (2000), de um total de 13 salas analisadas. Posteriormente, aprimorou-se tal classificação objetiva utilizando o método de otimização Elipsoidal, visando que este chegasse o mais próximo possível da classificação apresentada em Hidaka e Beranek (2000). Nesta etapa foi possível diminuir a discrepância entre 4 salas de ópera. Como resultado final, a Classificação Objetiva Otimizada foi utilizada para criar um ranking contendo todas as 22 salas de ópera com parâmetros acústicos disponíveis, sendo que 9 destas não foram analisadas no estudo realizado por Hidaka e Beranek (2000).

PALAVRAS-CHAVE: Sala de ópera, *PROMETHEE II*, elipsoidal, ranking objetivo.

CONCEPTION AND OPTIMIZATION OF OBJECTIVE RANKING FOR OPERA ROOMS USING COMPUTATIONAL METHODS

ABSTRACT: Creating the classification of opera rooms is not an exact task, as it depends on several

subjective factors. Based on a Subjective Classification (Hidaka and Beranek, J. Acoust. Soc. Am. 107, 340–354 (2000)), he created an Objective Classification using the *PROMETHEE II* multi-criteria decision-making method, using the parameters as criteria objective acoustics for 22 opera houses presented in Beranek (Springer Verlag, NY, 2004), 13 of which were also included in the study by Hidaka and Beranek (2000). Thus, they were based on a simplified subjective ranking, which considered only the 13 rooms that had available acoustic data, which enabled the comparison and validation of the proposed objective classification. The main concepts are introduced and the necessary adjustments for the work to be carried out are presented. As an initial result, an objective ranking was obtained with 4 opera rooms classified with equal positions and 5 rooms with a difference of up to two positions compared to the classification carried out by Hidaka and Beranek (2000), out of a total of 13 rooms analyzed. Subsequently, this objective classification was improved using the Elipsoidal optimization method, aiming to get it as close as possible to the classification presented in Hidaka and Beranek (2000). At this stage it was possible to reduce the discrepancy between 4 opera rooms. As a final result, the Optimized Objective Classification was used to create a ranking containing all 22 opera rooms with available acoustic parameters, 9 of which were not analyzed in the study carried out by Hidaka and Beranek (2000).

KEYWORDS: Opera room, *PROMETHEE II*, ellipsoidal, objective ranking.

1 | INTRODUÇÃO

Classificar a qualidade sonora de um ambiente é uma tarefa não exata, pois depende de diversos parâmetros subjetivos e objetivos, e outro ponto é que análise sonora deste ambiente pode obter um resultado bom para uma pessoa, e ruim para outra. Uma tentativa de determinar quais características acústicas que admiradores de música erudita preferem em uma sala de ópera muitas vezes está intrinsecamente relacionada com a sensação que estes tiveram quando frequentaram determinado ambiente, assim uma gama de elementos é responsável para criar a satisfação: a composição, o maestro, a orquestra, e a sala de ópera devem ter boa qualidade para criar uma audição memorável. Para o profissional da área da música, seja ele um maestro, um músico, ou um engenheiro acústico, é vital distinguir quais destes fatores contribuem para uma boa qualidade sonora do ambiente (Beranek, 2004).

As propriedades acústicas de uma sala de ópera são fortemente influenciadas pelas suas características construtivas, e existem algumas que são importantes, como por exemplo: em salas de concerto têm-se a necessidade da existência de superfícies reflexivas perto do palco (Beranek, 2004). As diferentes formas de construção de tais ambientes alteram diretamente os seus parâmetros acústicos objetivos, os quais são uma representação numérica das características perceptivas do som, e sua análise nos dá uma boa ideia da qualidade acústica de um ambiente para determinada aplicação, como por exemplo: um ambiente projetado para a audição de uma orquestra pode não ser adequado para a voz falada ou para a o show de uma banda de rock.

Hidaka e Beranek (2000) criaram uma classificação subjetiva para 23 salas de ópera, a qual serviu de base e como fonte de dados para a validação do trabalho desenvolvido. Em Beranek (2004) é apresentada uma tabela com parâmetros acústicos objetivos para 23 salas de ópera, sendo que 13 destas também estão incluídas no estudo subjetivo apresentado em Hidaka e Beranek (2000). Dessa forma, o objetivo deste trabalho é criar uma CO utilizando como critérios tais parâmetros acústicos objetivos e comparar seu resultado com o obtido com a CS apresentada em Hidaka e Beranek (2000).

A CO foi criada considerando diversos parâmetros, como o tempo de reverberação (RT) e o BQI, o que a classifica como um problema de tomada de decisão multicritério (MCDM). Tais problemas geralmente lidam com a classificação de várias alternativas do melhor para o pior, considerando múltiplos critérios (Behzadian et al., 2009). Para resolver este problema, implementou-se o método *PROMETHEE II* no *Matlab*. Existem várias versões desta família de métodos, e optou-se pelo tipo II pelo fato deste gerar uma classificação completa sem incomparabilidade (Brans e Vincke, 1986).

São apresentados os resultados da implementação deste método de classificação e as dificuldades e particularidades encontradas na tentativa de otimizar esta CO utilizando o método Elipsoidal. É apresentado o melhor resultado obtido após vários testes, e por último, o procedimento de classificação foi utilizado para criar um ranking com todas as 22 salas de ópera com dados acústicos disponíveis, contendo 13 salas incluídas em Hidaka e Beranek (2000) e mais outras 9 salas que não haviam sido previamente classificadas.

2 | METODOLOGIA

Esta parte do artigo descreve em seus próximos subitens a metodologia utilizada para criar a CO, e posteriormente o procedimento utilizado para otimizá-la.

2.1 Parâmetros acústicos objetivos

Beranek (2004) e Hidaka e Beranek (2000) relacionam diversas medidas físicas de qualidade acústica. A Figura 1 apresenta os descritores utilizados, assim como o seu respectivo nome e o critério que este foi referenciado na implementação do algoritmo de classificação utilizado. Os PO para 23 salas de ópera foram obtidos Beranek (2004) p. 558. A sala Royal Opera House (LO) não foi considerada, pois não há dado para um parâmetro acústico.

Símbolo	Nome	Critério Relacionado
$RT_{occ,M}$	Tempo de Reverberação	g_1
$EDT_{unocc,M}$	Tempo de Decaimento Inicial	g_2
$C_{80,3}$	Fator de Clareza	g_3
G_{125}	Fator de Intensidade para a banda de 125Hz	g_4
G_M	Fator de Intensidade para as médias frequências	g_5
BQI	<i>Binaural Quality Index</i>	g_6
$ITDG$	Tempo de Atraso Inicial	g_7

Figura 1. Parâmetros Acústicos Objetivos (detalhados em Beranek (2004) e Hidaka e Beranek (2000)).

2.2 Classificação subjetiva (CS)

Hidaka e Beranek (2000) apresentam o procedimento utilizado para comparar e classificar 23 salas de ópera. Questionários foram enviados para diversos maestros, e lhes foi solicitado para avaliar as características acústicas das SO's que eles conheciam bem em escalas de 5 subdivisões, que foram marcadas com números de 1 até 5. Neste trabalho foram consideradas apenas as SO's que têm seus parâmetros acústicos objetivos disponíveis em Beranek (2004). Assim, a classificação subjetiva considerada contém 13 salas de ópera, e seu resultado está apresentado na Figura 2.

Rating	Opera House	
1	Teatro Colón, Buenos Aires	BA
2	Semperoper, Dresden	DS
3	Teatro alla Scalla, Milan	MS
4	New National Theater, Tokyo	TT
5	Opéra Garnier, Paris	PG
6	Staatsoper, Prague	PS
7	Staatsoper, Vienna	VS
8	Metropolitan Opera, New York	NM
9	Festspielhaus, Salzburg	SG
10	Staatsoper, Hamburg	HS
11	Deutscheoper, Berlin	BD
12	Civic Opera, Chicago	CC
13	Komischeoper, Berlin	BK

Figura 2. Classificação Subjetiva de 13 salas de ópera (Hidaka e Beranek, 2000). Obs.: Apenas foram consideradas as salas que contêm dados acústicos disponíveis em Beranek (2004).

A classificação apresentada na Figura 2 foi empregada como a referência para a criação e comparação com a CO utilizando o *PROMETHEE II*, cujas principais características de sua implementação e formulação são apresentadas na seção a seguir.

2.3 Classificação objetiva utilizando o *PROMETHEE II*

Para criar a classificação objetiva utilizou os parâmetros acústicos objetivos

apresentados em Beranek (2004). Esta parte do trabalho descreve como o *PROMETHEE II* foi implementado, e quais foram as considerações para a modelagem e resolução do problema proposto.

Para a implementação do *PROMETHEE II* algumas considerações foram adotadas. Primeiro, não temos parâmetros acústicos que são ideais, apenas valores recomendados. Para o caso em questão, adotou-se que os parâmetros acústicos da sala com a melhor classificação (Teatro Cólón em Buenos Aires - *BA*). Desta forma, realizou-se a classificação a partir da segunda SO na CS.

Assim, o problema multicritério foi modelado de tal maneira que quanto mais os parâmetros acústicos de determinada sala se aproximassem dos parâmetros da sala considerada como sendo ideal (*BA*), melhor será sua classificação final.

Cada um dos 7 critérios apresentados na Figura 1 foram obtidos considerando o valor absoluto da diferença entre o valor do PO da sala em questão comparado ao da sala (*BA*). Por exemplo, o valor para o critério do $RT_{occ,M}$ para sala de ópera *DS* é dado por:

$$g_{1(DS)} = |RT_{occ,M(BA)} - RT_{occ,M(DS)}| = |1.56 - 1.6| = 0.04$$

Dessa forma foi possível estabelecer a matriz de critérios como representado na Figura 3.

A seguir são detalhados os passos (*Steps*) descritos no algoritmo representado na Figura 4:

- *Step1* – O cálculo de $d_{j(a,b)}$ é a diferença entre os valores do critério j para as SO a e b . Por exemplo: $d_{1(DS,MS)} = g_{1(DS)} - g_{1(MS)} = -0.28$
- *Step2* – A aplicação da função de preferência foi realizada utilizando um critério gaussiano como descrito em Brans e Vincke (1986). E inicialmente utilizou $\sigma = 0.5$, logo:

$$P_j(a, b) = \begin{cases} 0, & \text{se } d_{j(a,b)} \leq 0 \\ 1 - \exp\left(-\frac{d_{j(a,b)}^2}{0.5}\right), & \text{se } d_{j(a,b)} > 0. \end{cases}$$

- *Step3* – O cálculo de $\pi(a, b)$ foi realizado considerando inicialmente um vetor de pesos igual a: $[w_1, w_2, \dots, w_j] = [1, 1, 1, 1, 1, 3, 2]$
- *Step4* – O parâmetro $\varphi^+(a)$ indica que quanto maior for seu valor, mais a sala de ópera a domina as outras candidatas, e de forma análoga, quanto menor for $\varphi^-(a)$, menos a é dominada (Brans e Vincke, 1986).
- *Step5* – Para o caso implementado, quanto menor o valor de $\varphi(a)$, significa que a SO tem os parâmetros acústicos mais próximos dos ideais.

Hall Name		g_1	g_2	g_3	g_4	g_5	g_6	g_7	SROP
DS	Semperoper, Dresden	0.0400	0.1100	0.3000	0	0.3000	0.0700	2	2
MS	Teatro alla Scalla, Milan	0.3200	0.5800	2.5000	0.6000	2.7000	0.1700	2	3
TT	New National Theater, Tokyo	0.0700	0.0200	0.5000	0.4000	0.7000	0	2	4
PG	Opéra Garnier, Paris	0.3800	0.5600	3.5000	0.6000	1.7000	0.1500	3	5
PS	Staatsoper, Prague	0.3300	0.5500	2	0.3000	0.2000	0.0100	2	6
VS	Staatsoper, Vienna	0.2000	0.2900	1.6000	0.4000	0.4000	0.0500	1	7
NM	Metropolitan Opera, New York	0.0900	0.1000	0.6000	0.4000	1.9000	0.0300	0	8
SG	Festspielhaus, Salzburg	0.0600	0.0800	0.4000	0.4000	1.2000	0.2500	9	9
HS	Staatsoper, Hamburg	0.3300	0.3700	1.1000	0.5000	1.1000	0.1900	16	10
BD	Deutscheoper, Berlin	0.2000	0.1200	0.4000	0.1000	1.2000	0.2600	15	11
CC	Civic Opera, Chicago	0.0500	0.2300	1	0.1000	2.1000	0.1200	23	12
BK	Komischeoper, Berlin	0.3100	0.4900	2	0.3000	3.6000	0.0300	2	13
AM	Music Theater, Amsterdam	0.2600	0.4200	0.8000	0.2000	0.7000	0.1000	14	-
BE	Erkel Theater, Budapeste	0.2600	0.3200	2.7000	0.4000	0.9000	0.2000	1	-
BS	Staatsoper, Budapeste	0.2200	0.3500	0.8000	0.5000	2.0000	0	3	-
EO	Opera House, Essen	0.0500	0.1800	0.2000	0.2000	2.8000	0.1100	2	-
RE	Eastman Theater, Rochester	0.0700	0.1800	0.3000	0.3000	1.2000	0.1100	4	-
SO	Opera House, Seattle	0.4600	0.7800	1.5000	0.9000	0.3000	0.1700	7	-
TB	Bunka Kaikan, Tokyo	0.0500	0.0300	0	0	2.1000	0.0900	4	-
NT	Nissei Theater, Tokyo	0.4600	0.6600	3.3000	0.5000	2.9000	0.0700	1	-
WJ	Opera House, Washington	0.2800	0.4500	3.2000	0.6000	0.7000	0.1200	3	-

Figura 3. Matriz de Critérios.

Step 1: Determination of deviations based on pair-wise comparisons

$$d_j(a,b) = g_j(a) - g_j(b) \quad (1)$$

Where $d_j(a,b)$ denotes the difference between the evaluations of a and b on each criterion.

Step 2: Application of the preference function

$$P_j(a,b) = F_j[d_j(a,b)] \quad j=1, \dots, k \quad (2)$$

Where $P_j(a,b)$ denotes the preference of alternative a with regard to alternative b on each criterion, as a function of $d_j(a,b)$.

Step 3: Calculation of an overall or global preference index

$$\forall a, b \in A, \quad \pi(a,b) = \sum_{j=1}^k P_j(a,b)w_j \quad (3)$$

Where $\pi(a,b)$ of a over b (from 0 to 1) is defined as the weighted sum $p(a,b)$ of for each criterion, and w_j is the weight associated with j th criterion.

Step 4: Calculation of outranking flows/ The PROMETHEE I partial ranking

$$\phi^+(a) = \frac{1}{n-1} \sum_{x \in A} \pi(a,x) \quad (4) \quad \text{and} \quad \phi^-(a) = \frac{1}{n-1} \sum_{x \in A} \pi(x,a) \quad (5)$$

Where $\phi^+(a)$ and $\phi^-(a)$ denote the positive outranking flow and negative outranking flow for each alternative, respectively.

Step 5: Calculation of net outranking flow/ The PROMETHEE II complete ranking

$$\phi(a) = \phi^+(a) - \phi^-(a) \quad (6)$$

Where $\phi(a)$ denotes the net outranking flow for each alternative.

Figura 4. Algoritmo utilizado para a implementação do PROMETHEE II (Behzadian et al., 2009).

3 I RESULTADOS E PROCEDIMENTOS EXPERIMENTAIS

O algoritmo *PROMETHEE II* foi implementado no *Matlab*, e foi possível perceber que existe uma relação entre a CS com a CO criado, assim como ilustrado na Figura 5. Nota-se que três salas de ópera obtiveram a mesma classificação que a CS (*DS*, *TT* e *PS*) e cinco tiveram uma diferença de uma ou duas posições (*VS*, *SG*, *HS*, *BD* e *CC*).

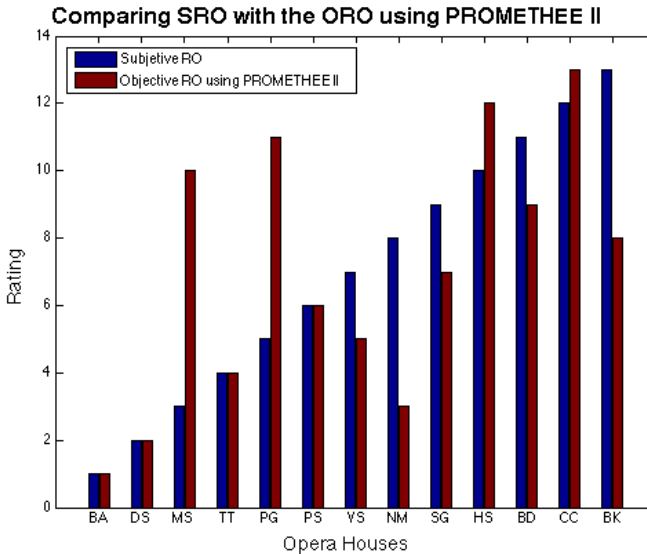


Figura 5. Classificação Objetiva criada utilizando o *PROMETHEE II*.

Na tentativa de obter a melhor classificação possível, um problema de otimização foi modelado. Como parâmetros de entrada, optou-se pela variação dos pesos (w_j , $j = 1, 2, \dots, 7$), e como função objetivo, optou-se por trabalhar com o somatório das diferenças entre a CS e a CO, dessa forma obteve-se:

$$f_{obj}(w_1, w_2, \dots, w_7) = \sum_{i=1}^N (c_{CS}(i) - c_{CO}(i))^2$$

Para avaliar a natureza da função objetivo, os pesos w_1 até w_5 foram fixados em 1, e variou-se os valores de w_6 e w_7 no intervalo de 0 até 1. A Figura 6 mostra este resultado para a função objetivo modelada. Devido a presença de quinais na f_{obj} , a aplicação de métodos determinísticos que utilizem o gradiente não puderam ser utilizados, e uma possibilidade encontrada foi a utilização do método Elipsoidal, o qual utiliza o cálculo do sub-Gradiente para achar a direção de busca.

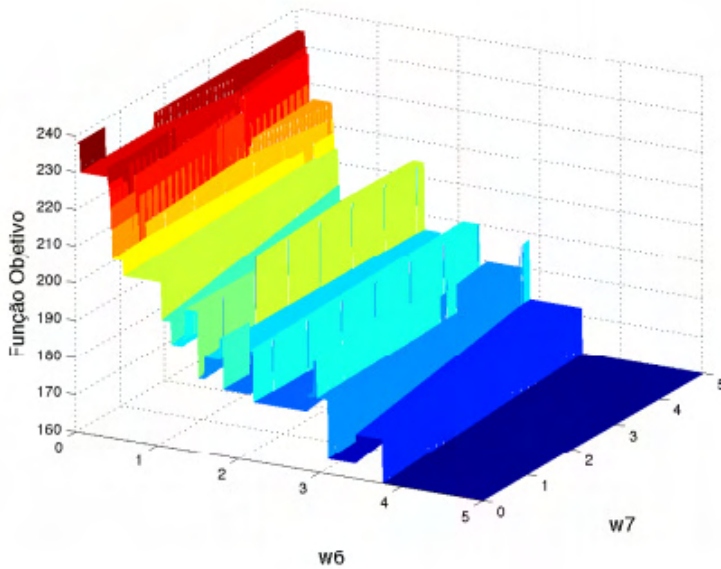


Figura 6. Função Objetivo considerando apenas as variações dos parâmetros w_6 e w_7 .

O algoritmo foi lançado considerando a variação de todos os parâmetros, porém se analisarmos a Figura 6 podemos ver a presença de semiplanos, e quando o algoritmo caía nestes subespaços o cálculo da derivada numérica (obtida a partir de pequenas variações em cada um dos eixos referentes aos critérios) retornava um valor nulo, o que interrompia seu processamento. Isto pode ser verificado na Figura 7, quando o algoritmo executou um número de 11 iterações.

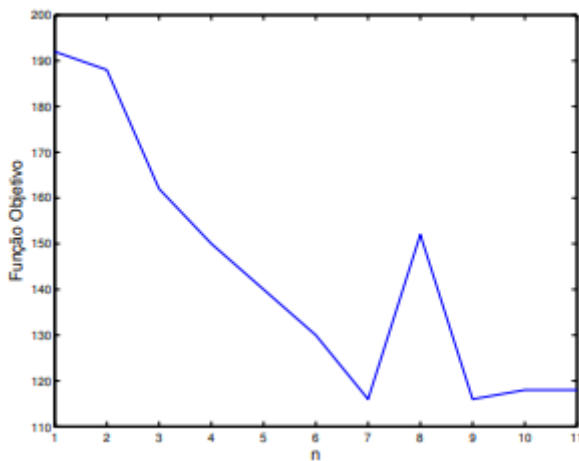


Figura 7. Histórico da Função Objetivo pelo número de iterações.

Outros fatores importantes que foram observados, foi que cada hora que o algoritmo era lançado, encontrava uma solução diferente. Tal ação pode ter sido causada pela característica multimodal da função objetivo, porém devido a alta dimensionalidade do problema, não foi possível concluir a veracidade desta suposição. Outro ponto, é que foi possível verificar que a função objetivo oscila, todavia é possível notar um ganho entre o valor inicial da função objetivo e o valor final, e foi isto que motivou o aprofundamento do método escolhido.

Para buscar um resultado melhor, incrementou-se o algoritmo de forma que diversos lançamentos sucessivos fossem efetivados, e um histórico do valor final da f_{obj} foi sendo armazenado, assim como a posição do vetor solução para os pesos.

O número de lançamentos foi sendo aumentado até que um ganho ocasionado por esta alteração não fosse percebido, e um bom número encontrado foi o de 3000 lançamentos. O resultado de tal análise está representado na figura 8.

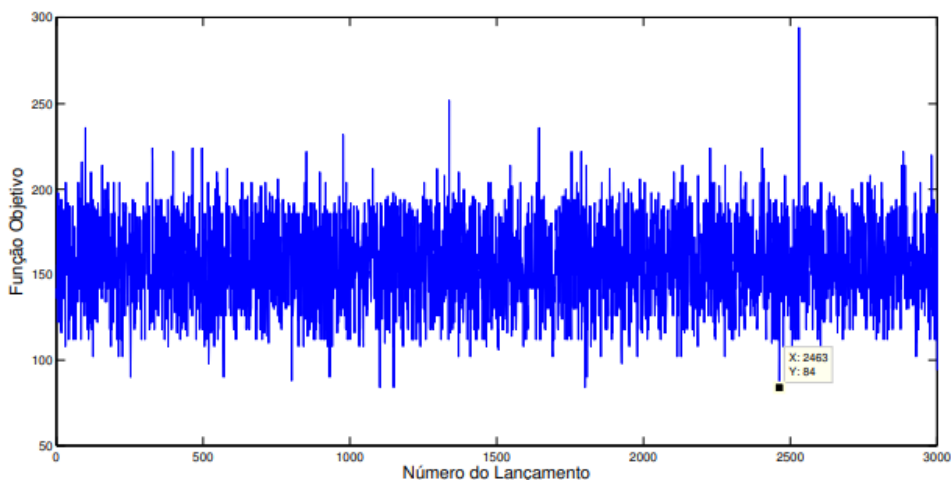


Figura 8. Histórico do valor final para a f_{obj} considerando um número de lançamentos igual a 3000.

Para esta análise, encontrou-se que o valor mínimo encontrado para a função objetivo foi: $f_{obj} = 84$.

Analisando a Figura 8 nota-se a presença de 4 lançamentos em que ocorreu este valor mínimo para a f_{obj} , porém em cada caso encontrou-se um vetor solução, mostrando que foram encontrados mais de um ponto de mínimo.

Desta forma obteve-se a classificação objetiva otimizada (COO), e seu resultado assim como uma comparação com as outras classificações estão representadas na Figura 9.

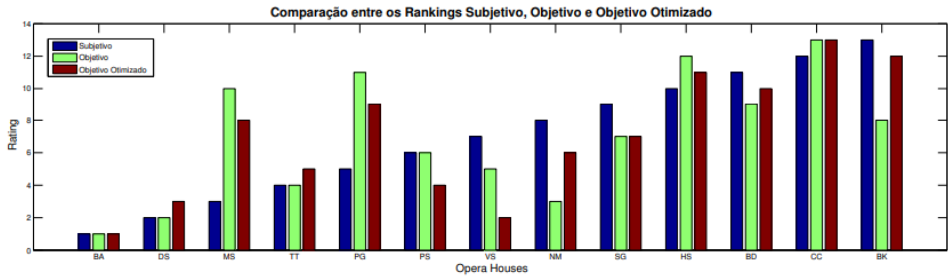


Figura 9. Comparação entre as Classificações Subjetiva (CS), Objetiva (CO) e Objetiva Otimizada (COO).

Outros critérios de implementação foram variados a fim de avaliar o resultado, os quais foram:

- Variação do intervalo de busca – o intervalo inicialmente fixado entre valores de 0 e 5, 0 e 1, 0 e 10 e 0 e 100, porém esta alteração não modificou os resultados.
- Variação do parâmetro σ – Este parâmetro, inicialmente fixado em 0.5 foi variado para valores iguais a 0.01, 0, 1, 0.3, 0.7, 0.9, 0.95 e 0.99, e percebeu que isto altera no melhor valor encontrado para a função objetivo, e os valor que geraram os melhores resultados foram os de 0.95 e 0.99. E para estes, teve-se que: $f_{obj} = 80$ e $w^* = [0.1983 \ 0.1038 \ 0.0082 \ 0.1270 \ 1.0000 \ 0.7559 \ 1.0000]$.

O último resultado obtido foi a inserção dos dados das outras 9 SO's que não foram incluídas no questionário realizado por Hidaka e Beranek (2000). A classificação das 22 salas, cujos parâmetros acústicos estão disponíveis em Beranek (2004), está apresentada na Figura 10.

Rating	Opera House	NOF
1	BA Teatro Colón, Buenos Aires	-
2	DS Semperoper, Dresden	-2.49
3	NM Metropolitan Opera, New York	-2.30
4	TT New National Theater, Tokyo	-2.00
5	VS Staatsoper, Vienna	-1.99
6	BE Erkel Theater, Budapeste	-1.08
7	PS Staatsoper, Prague	-0.92
8	EO Opera House, Essen	-0.86
9	RE Eastman Theater, Rochester	-0.26
10	TB Bunka Kaikan, Tokyo	-0.21
11	BS Staatsoper, Budapeste	0.24
12	SG Festspielhaus, Salzburg	0.47
13	NT Nissei Theater, Tokyo	0.55
14	AM Music Theater, Amsterdam	0.63
15	BK Komischeoper, Berlin	0.71
16	BD Deutscheoper, Berlin	0.78
17	WJ Opera House, Washington	0.79
18	MS Teatro alla Scalla, Milan	1.06
19	SO Opera House, Seattle	1.53
20	PG Opéra Garnier, Paris	1.70
21	HS Staatsoper, Hamburg	1.73
22	CC Civic Opera, Chicago	1.94

Figura 10. Classificação Objetiva Otimizada para as 22 salas de ópera cujos dados acústicos estão disponíveis em Beranek.

4 I CONCLUSÕES E TRABALHOS FUTUROS

Verificou-se que a utilização do algoritmo *PROMETHEE II* para a classificação objetiva de salas de ópera apresentou resultados consideráveis, pois gerou uma classificação com 4 posições iguais e com 5 salas de ópera apresentando diferença de até duas posições, de um total de 13 SO's em questão.

A utilização de um método de otimização para buscar uma classificação mais próxima do ranking subjetivo trouxe melhorias, pois diminuiu a discrepância na classificação de 4 salas de ópera (*MS*, *PG*, *NM*, e *BK*). Porém através da utilização do método de otimização Elipsoidal, não foi possível garantir que o ranking foi completamente otimizado, e se faz necessário o estudo de outros métodos de otimização e outras formas de modelagem da função objetivo, os quais ainda estão em estudo.

A criação de uma classificação incontestável não é possível, devido ao fato desta ser uma tentativa de quantificar parâmetros muitas vezes subjetivos, e que seu resultado é dependente de opiniões pessoais.

E mesmo com essas restrições, foi possível validar e aproveitar o principal resultado obtido, que foi a criação de uma classificação mediante apenas a avaliação de critérios objetivos, a qual é menos trabalhosa e pode ser estendida de forma simples a medida que mais salas de ópera entram em análise.

AGRADECIMENTOS

O presente trabalho foi realizado com o apoio financeiro da CAPES - Brasil.

NOMENCLATURA

<i>BQI</i>	Binaural Quality Index	SO	Sala de Ópera
CS	Classificação Subjetiva	n	Tempo discreto
CO	Classificação Objetiva	$c_{cs}(i)$	Posição da CS para SO(i)
COO	Classificação Objetiva Otimizada	$c_{co}(i)$	Posição da CO para SO(i)
<i>MCDM</i>	<i>Multi-Criteria Decision Making</i>	N	Número de salas de ópera (neste caso $N = 12$)
PO	Parâmetros Objetivos	w_1, w_2, \dots, w_7	Pesos para cada um dos 7 parâmetros acústicos
<i>RT</i>	<i>Reverberation Time</i>	i	Índice da SO

AUTORIZAÇÕES/RECONHECIMENTO

Os autores são os únicos responsáveis pelo material reproduzido nesse artigo.

REFERÊNCIAS

Behzadian, H., Kazemzadeh, R.B., Albadvi, A. e Aghdasi, M., 2009. “**Promethee: A comprehensive literature review on methodologies and applications**”. European Journal of Operational Research, Vol. 200, pp. 198–215.

Beranek, L.L., 2004. **Concert Halls and Opera Houses: Music, Acoustics, and Architecture**. Springer Verlag, New York.

Brans, J.P. e Vincke, P., 1986. “**A preference ranking organization method: (the promethee method for multiple criteria decision-making)**”. Management Science, Vol. 31, No. 6, pp. 647–656.

Hidaka, T. e Beranek, L.L., 2000. “**Objective and subjective evaluations of twenty-three opera houses in europe, japan and the americas**”. The Journal of the Acoustical Society of America, Vol. 107, pp. 368–383.

CAPÍTULO 4

SERVIDOR DE GERÊNCIA DE REDE

Data de aceite: 01/09/2021

Roger Robson dos Santos

Pontifícia Universidade Católica do Paraná
(PUCPR)

Jackson Mallmann

Instituto Federal Catarinense (IFC Brusque)

RESUMO: Este artigo conta com a descrição de ferramentas focadas na gerência de um servidor de rede em um ambiente corporativo, visando assim um melhor controle de toda a estrutura, gerando maior segurança e confiabilidade para a empresa. Com o passar dos anos, o ataque as redes vêm crescendo cada vez mais, e um bom gerenciamento da rede é primordial dentro de um ambiente corporativo. O objetivo é demonstrar a implementação de ferramentas de redes, que se aplicam para um melhor gerenciamento em grande parte de ambientes corporativos e assim poder demonstrar que o uso de softwares livres, podem trazer ótimos resultados para as empresas com um baixo custo.

PALAVRAS-CHAVE: Servidor, segurança, serviços.

ABSTRACT: This article has the description of tools focused on the management of a network server in a corporate environment, aiming at a better control of the entire structure, generating greater security and reliability for the company. Over the years, attacking networks have been growing more and more, and good network management is paramount within a corporate

environment. The goal is to demonstrate the implementation of network tools, which apply to better management in large part of corporate environments and thus to demonstrate that the use of free software, can bring great results for companies with a low cost.

KEYWORDS: Server, security, services.

1 | INTRODUÇÃO

A informática é apresentada como uma ferramenta para auxiliar nos trabalhos diários das empresas e garantir a segurança. Além disso, é uma ferramenta capaz de prover uma conectividade entre as pessoas da própria empresa, tanto em um meio interno, quanto externo. Com isso, temos a segurança de redes para garantir uma melhor segurança de todo o ambiente. Para isso se deve traçar um planejamento de toda estrutura de rede e servidores para se ter uma melhor segurança do ambiente e procurar adequar ao máximo dentro dos orçamentos dispostos pela empresa. Neste planejamento temos como foco todos os equipamentos necessários e os *softwares* para ter o melhor aproveitamento do *hardware* de baixo custo garantindo a segurança da empresa com a utilização de ferramentas (*softwares*) livres. Em nosso meio, utilizamos como estudo de caso uma empresa leiloeira. Problemas na estrutura de rede que causavam problemas durante a transmissão ao vivo do leilão, ocorrendo diversas quedas do sistema. Os

serviços de rede sempre instáveis e sem confiança, paravam de funcionar inesperadamente, e para quem acompanha o leilão *online* isso causa quedas na conexão deixando o serviço do leilão inativo ou com grandes lentidões. Entretanto, para a empresa leiloeira, a maior necessidade é uma transmissão de áudio ao vivo para um melhor aproveitamento do leilão e por ter uma alta demanda de compradores (usuários) *online*. Com estes problemas, buscamos a implantação de uma nova estrutura de rede para solucionar estes problemas, onde possibilita um total gerenciamento do sistema durante os leilões, trazendo a confiança nos serviços hospedados localmente e implantando a transmissão de áudio ao vivo. Como a empresa já possui um parque de *hardware* com bom potencial, foi decidido pela estrutura local, pelo melhor gerenciamento e suporte. Com isso foi desenvolvido servidor de internet, arquivos e *streaming* de áudio ao vivo para o novo ambiente, utilizando máquinas virtuais, onde os servidores utilizam sistema Linux e para utilização, foi aplicado máquinas com ambiente Windows para os testes das funções. Implantamos serviço de Firewall, Squid, DHCP, DNS, Icecast2 e Samba com (AD) *Active Directory*. Ademais, com toda estrutura de gerenciamento da nova rede, esperamos ter como resultado, uma confiabilidade nos serviços oferecidos pela empresa leiloeira, maior agilidade nos processos e um melhor controle da rede. Além de todo gerenciamento, espera-se ter um bom aproveitamento da transmissão ao vivo dos leilões, onde é uma grande necessidade da empresa, devido à alta demanda de compradores *online*. Por fim, a motivação para a realização deste trabalho sobre a administração de rede de computadores é demonstrar com as boas práticas de utilização das ferramentas necessárias, pode-se tornar o ambiente da internet dentro das empresas mais confiáveis através da aplicação de *softwares* (ferramentas) públicas.

2 | ESTADO DA ARTE

2.1 Introdução a História da Internet

A internet é uma ferramenta que se tornou dependência na vida das pessoas, seja para o trabalho, lazer, e outras coisas. A internet surgiu durante a Guerra Fria nos Estados Unidos cobrindo a necessidade da comunicação entre as bases americanas em 1945. Quando se iniciou, se chamava ARPANET, nome derivado da empresa que a desenvolveu (*Advanced Research and Projects Agency*) [STALLING 2007]. Membros da MIT (*Masschusetts Instite of Technology*) tiveram uma ideia: converter toda a linguagem da ARPANET para uma linguagem mais humana. Com isso desencadeou os primeiros conhecedores de informática, conhecidos como *hackers*, que foram capazes de criarem e desenvolverem os primeiros compiladores da época até chegar os compiladores atuais [STALLING 2007]. Com o passar dos anos começaram a nascer problemas de invasões em redes de internet nas organizações, com isso em 1994 a IAB (*Internet Architecture Board*) começou a emitir relatórios de arquitetura de internet. Mostrando assim que havia

a necessidade de melhorar a segurança da internet. Para se ter uma ideia, na Figura 1 é apresentado gráfico com incidentes relatados durante os anos de 1999 até 2015.

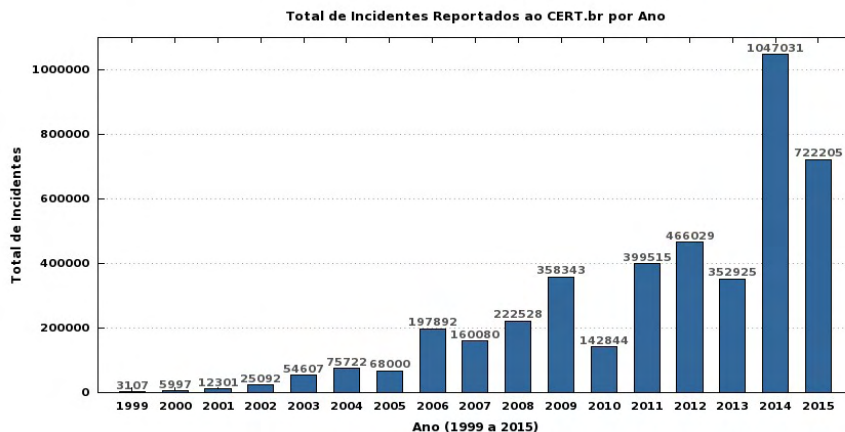


Figura 1 - Gráfico de total de incidentes reportados ao CERT.br por ano.

Retirado em <https://www.cert.br/stats/incidentes/> retirado 27/02/2017.

2.2 Invasores e Serviços de Segurança

Hackers são pessoas que constroem ferramentas que auxiliam e facilitam o trabalho dos usuários. Normalmente os *hackers* costumam ter uma ideia de que o acesso à rede da internet deve ser ilimitado com o objetivo do autoconhecimento. Em meio a isso, *hackers* cometem crimes de invasões a certos lugares, com o intuito de corrigir problemas que os técnicos das organizações desconhecem, ajudando os mesmos a evitar ataques por meio de crackers [FILHO 2010]. Por sua vez, *crackers* são opostos aos *hackers*. Eles têm apenas o entendimento utilizado para o mal, utilizam suas habilidades técnicas e ferramentas com intuito de destruir uma rede, piratear programas, jogos e entre outros, e ainda a construção de vírus capazes de destruir todas as informações de uma rede [FILHO 2010].

Entre alguns serviços de segurança podemos contar com o serviço de autenticação de dados, onde se faz com que a segurança de alguns serviços importantes realize a autenticação dos dados durante uma comunicação, seja ela uma entidade par onde é um serviço utilizado durante uma comunicação ou em fase de transferência de dados. Além desta autenticação utiliza-se também a autenticação de origem de dados ao qual provê a confirmação da origem dos dados. A autenticação nos assegura que a entidade ao qual estamos se comunicando, é realmente aquela que afirma ser [STALLING 2007]. Outro serviço importante é o controle de acesso. Ele impede que o uso não autorizado de algum recurso dentro da organização trabalhe com a instrução de confidencialidade, ou seja, assegura que não haja divulgação dos dados não autorizados e garantindo a proteção de

dados durante a conexão e mesmo em uma conexão de um arquivo local [STALLING 2007]. Além dos serviços citados, existe o serviço de integridade de dados. Ele garante que os dados recebidos durante uma conexão estejam exatamente como foram enviados de uma entidade, sem que haja modificação, inserção, exclusão ou repetição de dados [STALLING 2007]. Ademais, a irretratabilidade é um serviço que oferece uma proteção contra imposição ou negação das entidades envolvidas em uma comunicação. É assegurado que a origem da mensagem prove ao seu destino que sua mensagem foi recebida por quem enviou sem nenhuma ocorrência de desvio durante o caminho até quem vai receber [STALLING 2007]. Por fim, a realização do projeto será baseada na arquitetura apresentada na Figura 2.

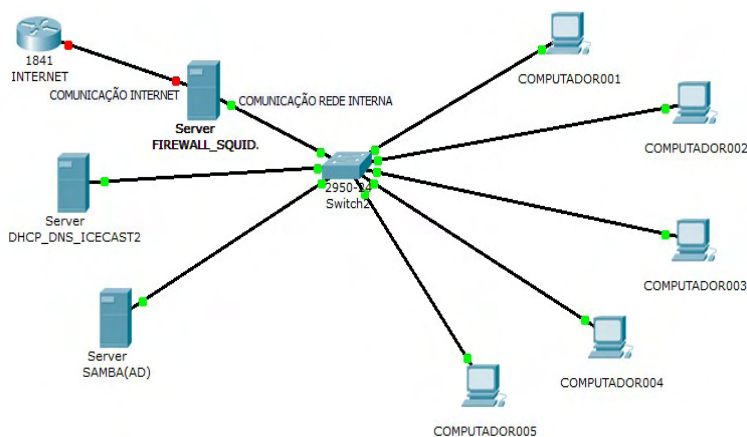


Figura2 - Fluxograma de computadores.

A rede irá possuir três servidores seguindo as seguintes regras: o principal servidor de internet contendo os serviços de Firewall com NetFilter IPTABLES para cuidar da segurança utilizando um conjunto de regras criadas pelo administrador da rede, e Squid com Proxy Transparente, um serviço para restringir acessos indevidos e armazenar em *cache* páginas já navegadas na rede agilizando a requisição destas páginas, um segundo servidor contendo o serviço de DHCP para que seja definido todas as máquinas no mesmo protocolo de comunicação TCP/IP usando Ipv4. Este servidor também contará com serviço de DNS para resolução de nomes dos *sites* de forma localmente. Também teremos o servidor Samba contendo o serviço de AD e *streaming* de áudio com Icecast. E para entendermos o funcionamento dos servidores, faz-se uma breve explicação de cada serviço.

2.3 Firewall

É uma ferramenta de segurança baseada em *hardware* e *software*, que junto de

um conjunto de regras e instruções, analisa o tráfego de rede durante a transmissão e recepção de dados. Neste trabalho, o firewall será construído com o pacote NetFilter Iptables, uma ferramenta que cria e administra através de regras filtrando pacotes na rede [NETO 2004]. O Iptables poderá funcionar baseado em endereços, portas de origem e destino dos pacotes e prioridade, além disso, ele funcionará a partir da comparação de regras para saber se um pacote possui ou não permissão para passar. No firewall será possível restringir e registrar tudo que estiver acontecendo na rede através de diversos *logs* gerados pelo Iptables. Entre os principais objetivos do Iptables: redirecionamentos de portas e serviços, a troca de protocolos (um exemplo ssl3 para tls1), criar regras com intuito de bloquear serviços, usuários da rede, acessos por determinado IP, liberações de navegação, entre muitos outros serviços [NETO 2004].

2.4 Squid

É uma ferramenta utilizada com diversos objetivos, principalmente impor regras de restrição de acesso baseadas no endereço IP da máquina, login, horário e bloqueio de conteúdos indevidos. Além disso, ele possui um *cache* de navegação onde é capaz de armazenar todas as páginas e arquivos já acessados em uma memória e quando alguém acessar uma página que já foi carregada. O Proxy envia os dados que armazenou no *cache*, sem a necessidade de baixar arquivos da mesma página repetidamente. Isso gera uma grande economia de banda e torna o acesso mais rápido sem a necessidade de um investimento com uma conexão rápida de internet [WESSELS 2004]. Hoje em dia, muitos *sites* costumam ter um conteúdo dinâmico em suas páginas onde mudam muitas coisas na página em cada visita do usuário.

Mesmo assim, o Proxy ainda é uma forte ferramenta, vendo que conteúdo de HTML e animações não mudam e acabam sendo reaproveitados pelo Proxy [MORIMOTO 2006]. Outra vantagem na utilização do Proxy é o fato do Squid armazenar todos os acessos dando possibilidade ao administrador da rede ver quem acessou, o que acessou e em quais horários acessou [MORIMOTO 2006]. Na configuração do Squid mais simples é necessário adicionar o Proxy máquina a máquina tornando pouco vantajoso todo esse trabalho, mas existe uma forma simples de resolver tudo isso: Proxy Transparente. Com isto não existe a necessidade de colocar o Proxy no navegador dos usuários. Uma conexão compartilhada via NAT, com a mesma configuração básica nos clientes. Para isso é criado uma regra no firewall que faz com que toda navegação na porta 80 seja redirecionada ao Squid passando automaticamente através do Proxy, sem que se precise fazer nenhuma configuração adicional aos seus clientes [MORIMOTO 2006].

2.5 Dynamic Host Configuration Protocol (DHCP)

É um recurso essencial para distribuir dinamicamente endereços IP e parâmetros de rede. Com este serviço as implementações de redes em Ipv4 e Ipv6 apresentam as

mesmas funcionalidades básicas, mas com significativas diferenças nas opções que podem ser enviadas [ALMEIDA 2000]. O DHCP possibilita uma maior segurança na configuração da rede, pois evita erros causados pela necessidade de digitação manual de valores em cada computador. Além disso, o DHCP ajuda a impedir conflitos de endereço IP na rede causados por um endereço atribuído anteriormente, ou que já esteja sendo utilizado por outro computador na rede, mantendo assim um melhor gerenciamento da rede. Usar servidores DHCP pode diminuir bastante o tempo gasto na configuração e reconfiguração de computadores da rede [TECHNET 2017]. Por exemplo, um cliente, termo utilizado para descrever um computador ligado à rede que obtém as configurações do protocolo TCP/IP a partir de um servidor DHCP, onde qualquer computador ou dispositivo de rede seja capaz de se comunicar com o servidor DHCP e obter as configurações do TCP/IP, é considerado um cliente DHCP. Segundo a RFC 2132, publicado pela Internet *Engineering Task Force* – IETF, cliente DHCP é considerado qualquer equipamento que ofereça suporte e seja compatível com o comportamento de cliente DHCP [BATTISTI 2013].

2.6 Domain Name System (DNS)

É um serviço de resolução de nomes e domínios em um banco de dados distribuído. Permite-se um controle total dos seguimentos que estejam disponíveis em toda a rede através de um sistema de cliente-servidor, ou seja, um computador pessoal que faz uma pergunta a outro computador que por sua vez se encarrega de encontrar a informação que você precisa [CAMPOS 2017].

A essência do DNS é a criação de um esquema hierárquico de atribuição de nomes baseado no domínio e de um sistema de banco de dados distribuídos para implementar esse esquema de nomenclatura (TANEMBAUM 2003, p.617).

Com o serviço DNS é possível efetuar o acesso a *sites* sem a necessidade de saber o seu endereço IP, bastando apenas saber o nome deste *site*. Todo este trabalho de tradução do nome de domínio para o endereço IP é efetuado pelo protocolo DNS. Além da resolução de nomes externos, o DNS pode ser utilizado como ferramenta para poder definir nome a diversos dispositivos dentro da rede, facilitando que sejam encontrados esses aparelhos através dos nomes e não mais apenas pelos seus endereços de IP [CAMPOS 2017].

2.7 Samba

Segundo SILVA (2005), o Samba é um conjunto de ferramentas em um servidor, que permite a comunicação entre máquinas executando Linux e Windows, possibilitam o compartilhamento de arquivos, diretórios e impressão através do protocolo SMB (*Server Message Block*) e pode atuar como um PDC (*Primary Domain Controller*), autenticando assim usuários de redes locais [MORIMOTO 2006]. O Samba foi através deste protocolo para resolver os problemas de comunicação entre um sistema UNIX e um MS DOS, pois

para se ter esta comunicação, se fez necessário o uso da interface NetBIOS (*Network Basic Input Output System*) [MORIMOTO 2006]. Toda a configuração do Samba é centralizada no arquivo `smb.conf`. O `smb.conf` pode ser visto como dividido em três partes: a configuração do servidor Samba (parâmetros na seção [*global*]), a configuração dos diretórios/pastas pessoais dos usuários (parâmetros na seção [*homes*]) e as demais seções que correspondem aos diretórios compartilhados [ZUCARINO 2001].

2.8 AD (*Active directory*)

O AD tem como principal função ser um único diretório, onde ao invés do usuário possuir várias senhas para cada tipo de serviço disponível na rede (e-mail, sistema da empresa, *logar* no computador), o usuário poderá ter apenas uma senha para acessar todos os recursos disponíveis na rede [SANTANA 2014]. Em um ambiente de TI corporativo, o serviço de AD é muito utilizado, sendo o centro para implementação de Serviços de Diretório. O AD gerencia os recursos da rede, armazenando informação sobre os recursos disponíveis. Com isso facilita a pesquisa e a autenticação [STANEK 2009]. O AD utiliza vários protocolos para seu funcionamento, mas segundo STANEK (2009), “o protocolo principal para acesso ao AD é o LDAP (*Lightweight Directory Access Protocol*), um protocolo padrão do setor para acesso a diretórios administrativos através do TCP/IP”. Para o funcionamento do AD é necessário ter o serviço de DNS disponível na rede, pois o AD utiliza do DNS para nomeação dos recursos, e para resolução de nomes na rede [SANTANA 2014].

2.9 Icecast2

É um servidor de *streaming* de áudio e vídeo, que suporta as extensões Mp3, Opus, WebM e Ogg. Pode ser utilizado para criar estações de rádio na internet ou uma transmissão ao vivo de eventos e entre outros. Possibilita novos formatos a serem adicionados com relativa facilidade e suporta padrões abertos para comunicação e interação [PERMALINK 2015].

3 | DESENVOLVIMENTO

Neste trabalho aplica-se o Oracle VM Virtualbox 5.1.2, para implementação do projeto em cinco máquinas virtuais. Tem-se três máquinas virtuais com sistema operacional Debian e duas máquinas com sistema operacional Windows XP. Os arquivos de configurações estão disponíveis¹.

¹ <https://drive.google.com/drive/folders/1tRpo6Z7IDWjYJOW1x1NvaGorA0XmyELt?usp=sharing>

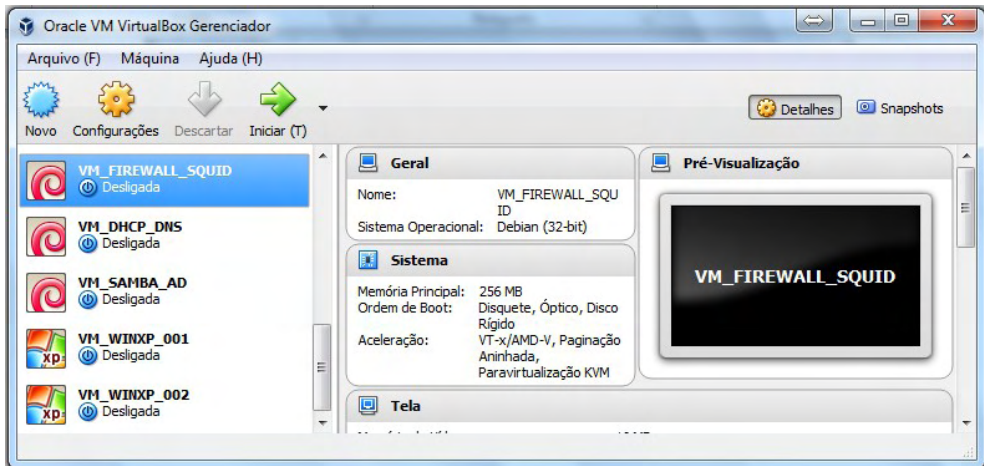


Figura3 – Interface web VirtualBox 5.1.2.

3.1 Configurações do Firewall

A configuração do firewall fica no arquivo `/etc/init.d/firewall.sh`, sendo que é descrito cada regra contida no firewall conforme a representação da rede. Por fim, para iniciar o serviço utilize o comando `/etc/init.d/firewall.sh`.

3.2 Configurações do Squid3

Configurando arquivo de configuração `/etc/squid3/squid.conf`. Para finalizar e iniciar o Proxy, utilize o comando `/etc/init.d/squid3 start`.

3.3 Configurações do DHCP

Configurando o arquivo de configuração `/etc/dhcp/dhcpd.conf`. Após, iniciar o serviço com o comando `/etc/init.d/isc-dhcp-server start`.

3.4 Configurações do DNS

Iniciando configuração dos arquivos de configuração do Bind9. Iniciar o serviço com o comando `/etc/init.d/bind9 start`.

3.5 Configuração do Iccast2

Para finalizar e iniciar o serviço, utilizar o comando `/etc/init.d/icecast2 restart`.

3.6 Configurações Samba - AD

Após, configuração da placa de rede em `/etc/network/interfaces` para iniciar a configuração do serviço.

4 | REALIZAÇÃO DOS TESTES

Para realizações de testes foi verificado se todas as configurações estavam corretas. Para tal, abra o CMD via o iniciar do Windows, e digite “ipconfig /all”. Desta forma, obtêm-se todas as configurações de rede. Endereço do gateway deve ser o IP onde está o firewall 192.168.0.1, servidor DHCP deve ser 192.168.0.2 e os servidores DNS o endereço do servidor com o AD 192.168.0.3.

Sufixo DNS específico de conexão: redevirtual.local .int

Endereço de IP: 192.168.0.100

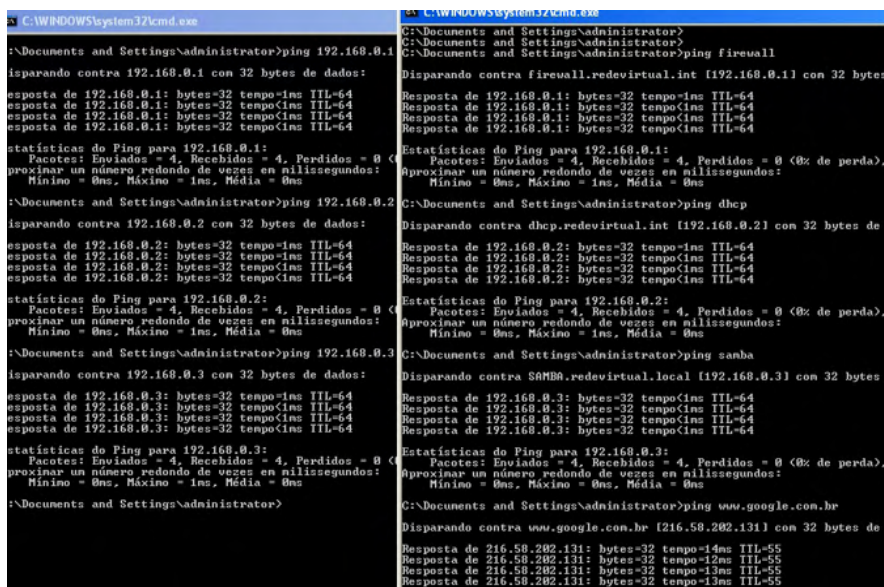
Máscara de Sub-rede: 255:255.255.0

Gateway padrão: 192.168.0.1

Servidores DHCP: 192.168.0.2

Servidor DNS: 192.168.0.3

Realizar testes de Ping (Figura 5) para verificar a comunicação com os servidores da rede, e observar se todos estão respondendo pelo próprio IP e respondendo pelos nomes definidos nas configurações do DNS.



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Administrator>ping 192.168.0.1
Disparando contra 192.168.0.1 com 32 bytes de dados:
Resposta de 192.168.0.1: bytes=32 tempo=1ms TTL=64
Resposta de 192.168.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.1: bytes=32 tempo<1ms TTL=64
Estatísticas do Ping para 192.168.0.1:
    Pacotes: Enviados = 4, Recebidos = 4, Perdidos = 0 (0% de perda),
    Aproximar um número redondo de vezes em milissegundos:
    Mínimo = 0ms, Máximo = 1ms, Média = 0ms
C:\Documents and Settings\Administrator>ping 192.168.0.2
Disparando contra 192.168.0.2 com 32 bytes de dados:
Resposta de 192.168.0.2: bytes=32 tempo=1ms TTL=64
Resposta de 192.168.0.2: bytes=32 tempo=1ms TTL=64
Resposta de 192.168.0.2: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.2: bytes=32 tempo<1ms TTL=64
Estatísticas do Ping para 192.168.0.2:
    Pacotes: Enviados = 4, Recebidos = 4, Perdidos = 0 (0% de perda),
    Aproximar um número redondo de vezes em milissegundos:
    Mínimo = 0ms, Máximo = 1ms, Média = 0ms
C:\Documents and Settings\Administrator>ping 192.168.0.3
Disparando contra 192.168.0.3 com 32 bytes de dados:
Resposta de 192.168.0.3: bytes=32 tempo=1ms TTL=64
Resposta de 192.168.0.3: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.3: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.3: bytes=32 tempo<1ms TTL=64
Estatísticas do Ping para 192.168.0.3:
    Pacotes: Enviados = 4, Recebidos = 4, Perdidos = 0 (0% de perda),
    Aproximar um número redondo de vezes em milissegundos:
    Mínimo = 0ms, Máximo = 1ms, Média = 0ms
C:\Documents and Settings\Administrator>
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Administrator>
C:\Documents and Settings\Administrator>ping firewall
Disparando contra firewall.redevirtual.int [192.168.0.1] com 32 bytes de
Resposta de 192.168.0.1: bytes=32 tempo=1ms TTL=64
Resposta de 192.168.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.1: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.1: bytes=32 tempo<1ms TTL=64
Estatísticas do Ping para 192.168.0.1:
    Pacotes: Enviados = 4, Recebidos = 4, Perdidos = 0 (0% de perda),
    Aproximar um número redondo de vezes em milissegundos:
    Mínimo = 0ms, Máximo = 1ms, Média = 0ms
C:\Documents and Settings\Administrator>ping dhcp
Disparando contra dhcp.redevirtual.int [192.168.0.2] com 32 bytes de
Resposta de 192.168.0.2: bytes=32 tempo=1ms TTL=64
Resposta de 192.168.0.2: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.2: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.2: bytes=32 tempo<1ms TTL=64
Estatísticas do Ping para 192.168.0.2:
    Pacotes: Enviados = 4, Recebidos = 4, Perdidos = 0 (0% de perda),
    Aproximar um número redondo de vezes em milissegundos:
    Mínimo = 0ms, Máximo = 1ms, Média = 0ms
C:\Documents and Settings\Administrator>ping samba
Disparando contra SAMBA.redevirtual.local [192.168.0.3] com 32 bytes de
Resposta de 192.168.0.3: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.3: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.3: bytes=32 tempo<1ms TTL=64
Resposta de 192.168.0.3: bytes=32 tempo<1ms TTL=64
Estatísticas do Ping para 192.168.0.3:
    Pacotes: Enviados = 4, Recebidos = 4, Perdidos = 0 (0% de perda),
    Aproximar um número redondo de vezes em milissegundos:
    Mínimo = 0ms, Máximo = 0ms, Média = 0ms
C:\Documents and Settings\Administrator>ping www.google.com.br
Disparando contra www.google.com.br [216.58.202.131] com 32 bytes de
Resposta de 216.58.202.131: bytes=32 tempo=14ms TTL=55
Resposta de 216.58.202.131: bytes=32 tempo=12ms TTL=55
Resposta de 216.58.202.131: bytes=32 tempo=13ms TTL=55
Resposta de 216.58.202.131: bytes=32 tempo=13ms TTL=55
```

Figura 5 – Testes de Ping.

Realizar alguns testes de navegação para verificar as funcionalidades do Proxy dentro da rede (Figura 6). Verificar se os bloqueios dentro da rede estão funcionando corretamente.

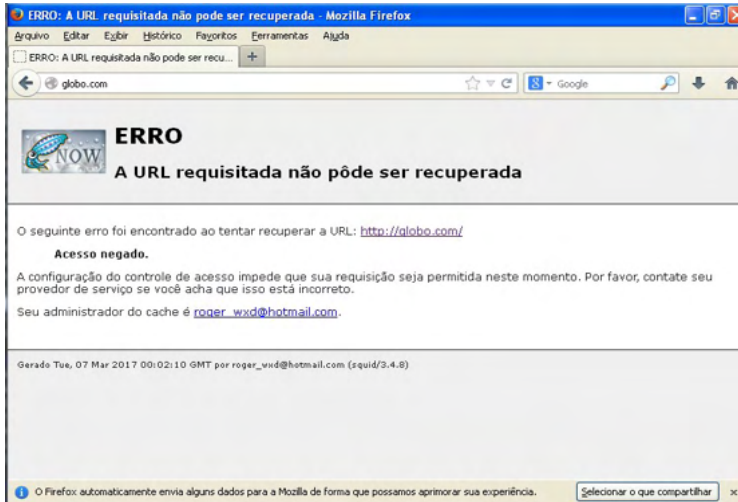


Figura6 – Testes de bloqueio do Proxy.

Testes de verificação da tela web do Icecast2 e verificações de acesso através do nome:porta(icecast:8000) configurados no servidor de DNS. Figura 7.

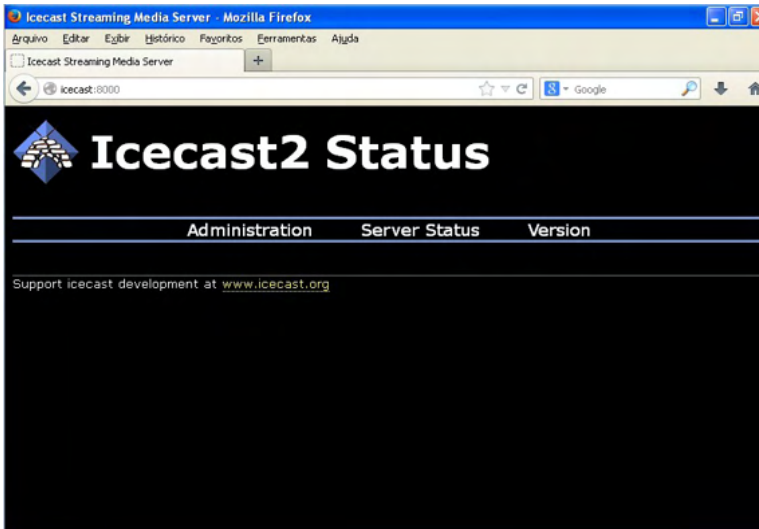


Figura7 – Interface Web do icecast.

Testes de criação de usuário no AD gerenciados pelo Windows (Figura 8).

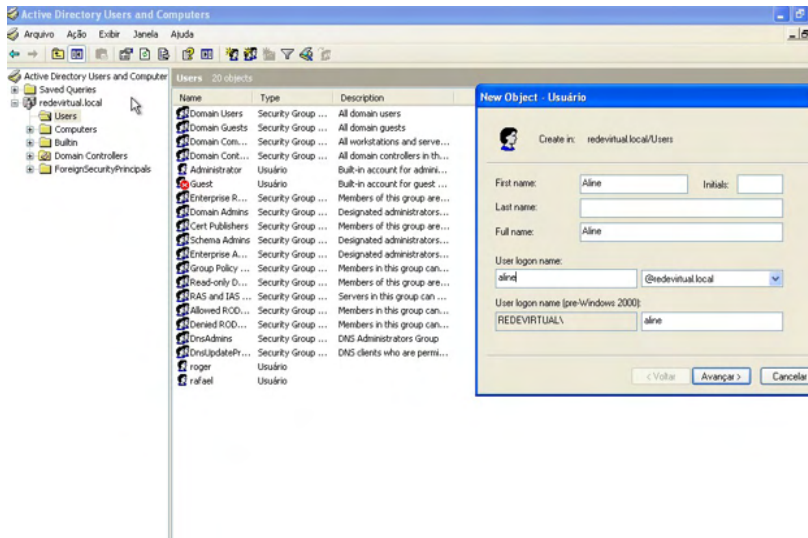


Figura8 – Interface de configuração de usuários do AD.

Realizando testes nas diretivas do AD (Figura 9).

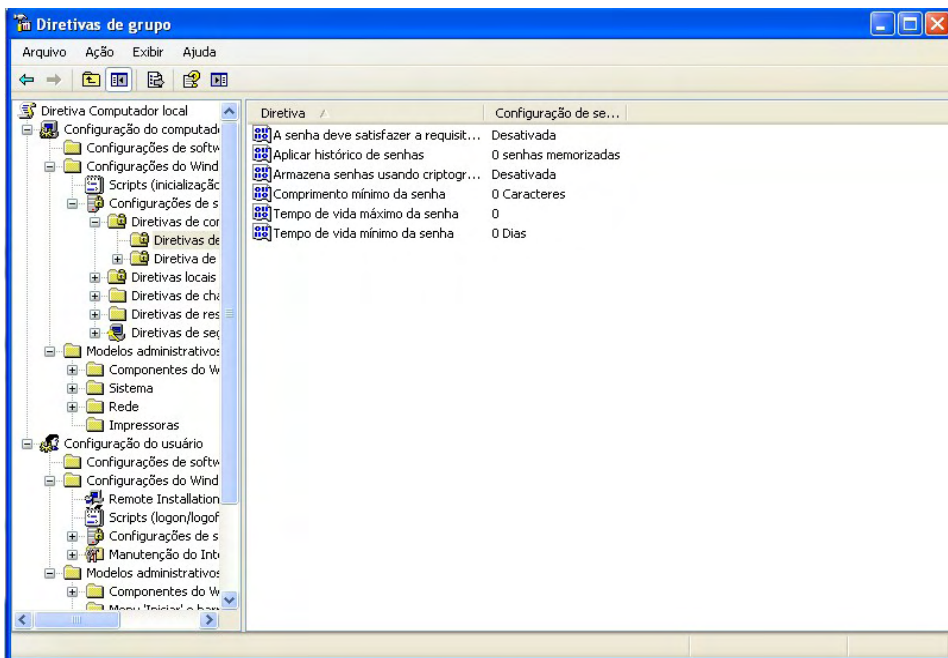


Figura9 – Interface configurações das Diretivas de grupo do AD.

5 | RESULTADOS OBTIDOS

Com a implementação destes servidores será obtido um enorme resultado em tudo que foi planejado e esperado para o ambiente de trabalho. Através das regras do `iptables` será possível compartilhar internet com a rede, bloquear portas e evitar perigos da internet, liberar portas necessárias para o trabalho com segurança, ativar acessos externos para serviços específicos dentro da rede e entre outros. Por sua vez, aplicando o `Squid` será proporcionado uma grande economia de banda de internet na organização, já que os principais *sites* navegados terão suas sintaxes armazenadas no *cache* do Proxy, bloqueios de URL indevidas garantindo a segurança da organização e gerando logs necessários para o aprimoramento da rede. O DHCP proporciona a configuração da rede para que todas as máquinas sejam configuradas de forma automática e fácil, atribuindo o IP, máscaras de rede, gateway e DNS. Além do DHCP, o serviço do DNS irá proporcionar um trabalho de resolução de nomes da internet e da rede interna localmente muito interessante, pois com essa resolução é possível economizar banda de internet sem a busca de resoluções de nomes em algum DNS externo. Agora também poderemos contar com a transmissão ao vivo dos serviços da empresa no próprio *site*, ou em outros domínios com o `Icecast`. Com o serviço do AD, poderemos armazenar e acessar facilmente arquivos da empresa em qualquer máquina dentro da rede utilizando apenas uma conexão de rede com o servidor.

6 | CONCLUSÃO

É importante trabalhar com um risco reduzido e segurança dentro da organização, garantindo integridade de arquivos e dados, pois esta implementação é necessária devido ao crescimento e avanço da tecnologia e técnicas de segurança de rede. O objetivo deste trabalho foi demonstrar a implementação de serviços de segurança para toda a rede garantindo a eficácia em diversos ambientes corporativos com ferramentas livres, garantindo e convencendo as empresas que o *software* gratuito proporciona vantagens devido à alta qualidade e o baixo custo. Assim, este artigo envolveu o estudo da programação de redes de computadores, configurações de servidores Firewall, Proxy, DHCP, DNS, *Streamings* de Áudio, compartilhamento de arquivos de rede. O resultado disto proporciona um enorme aprendizado na área de Redes de Computadores.

7 | REFERÊNCIAS

ALMEIDA, Rubens Queiroz de. **Domain Name Service Configuração e Administração**. Disponível em < http://www.inf.ufes.br/~zegonc/material/Redes_de_Computadores/Apostila%20DNS%20Rubens%20Queiroz.pdf>. Acesso em 23 de julho de 2021.

BATTISTI, Julio. **"Introdução ao DHCP 2013"**. Disponível em: <http://juliobattisti.com.br/artigos/windows/tcpip_p9.asp>. Acessado em 05 de março 2017.

CAMPOS, David Robert Camargo de. **Introdução a DNS & DNSSEC**. 2017. Disponível em <<https://ftp.registro.br/pub/doc/introducao-dns-dnssec.pdf>>. Acesso em 23 de julho de 2021.

FILHO, Glenio Leitão Marques. **“Hackers e Crackers na internet: as duas faces da moeda 2010”**. Disponível em: <www.insite.pro.br>. Acessado em 05 de março 2017.

MORIMOTO, Carlos E. **“Linux Redes e Servidores – Guia prático”**. 2006.

NETO, Urubatan. **Dominando Linux Firewall Iptables**. Rio de Janeiro: Editora Ciência Moderna LTDA. 2004.

PERMALINK. **“Epirat”**. Disponível em: <<http://icecast.org/>>. Acessado 05 março 2017.

SANTANA, Fabiano. **“AD – Active Directory 2014”**. Disponível em: <http://juliobattisti.com.br/fabiano/artigos/activedirectory.asp>. Acessado em 05 de março 2017.

STALLING, Willian. **Criptografia e segurança de redes, Princípios e prática**, 4ª Edição, 2007. Editora O'Reilly.

STANEK, Willian R. (2009). **“Windows Server 2008: guia completo”**, Porto Alegre: Bookman.

TANEMBAU, Andrew S. **“Redes de Computadores”**. Tradução Vandenberg D. de Souza. 4. ed. Campus, 2003.

TECHNET. **“Microsoft. Visão geral do DHCP”**. Disponível em: [https://technet.microsoft.com/pt-br/library/cc731166\(v=ws.11\).aspx](https://technet.microsoft.com/pt-br/library/cc731166(v=ws.11).aspx). Acessado em 05 de março 2017.

ZUCARINO, Victor. **“Utilizando o SAMBA - Parte I”**. Disponível em: <<http://www.ebah.com.br/content/ABAAABeIMAB/aula-08-servidor-samba>>. Acessado em 05 de março 2017.

WESSELS, Duane. **Squid: The Definitive Guide**. 1st Edition. 2004. O'Reilly.

SOBRE A ORGANIZADORA

LILIAN COELHO DE FREITAS - Professora do Instituto Federal de Educação, Ciência e Tecnologia do Pará (IFPA). Possui graduação em Engenharia da Computação pela Universidade Federal do Pará (2007) e mestrado em Computação Aplicada pelo Programa de Pós-Graduação em Engenharia Elétrica (PPGEE) da UFPA, obtido em 2009. Em 2008, realizou estágio de mestrado no Instituto de Engenharia de Sistemas e Computadores (INESC/Porto - Portugal). Atuou como pesquisadora membro do Laboratório de Eletromagnetismo Aplicado (LEA/UFPA) de 2004 a 2014 e do Laboratório de Sensores e Sistemas Embarcados (LASSE/UFPA) de 2008 a 2012. Atuou como Pesquisadora Visitante no *Georgia Institute of Technology* (Atlanta, Georgia, Estados Unidos), no período de Março/2012 a Fev/2013. Tem mais de 50 trabalhos publicados, envolvendo publicações em livros, revistas e eventos científicos. Seus interesses de pesquisa são: telecomunicações (comunicações sem fio, rádio cognitivo) e *machine learning*.

ÍNDICE REMISSIVO

A

Assimilação de dados 15, 16, 17, 19, 20, 21

C

Classificação objetiva otimizada 30, 38, 40, 41

D

Domain Name System (DNS) 47

Dynamic Host Configuration Protocol (DHCP) 46

F

Filtro de Kalman por conjunto 15

Firewall 43, 45, 46, 49, 50, 53, 54

Fórmula Horn-Mista 1

I

Internet 43, 44, 45, 46, 47, 48, 53, 54

M

Modelo atmosférico 15, 16

O

Otimização Elipsoidal 30, 40

P

PROMETHEE II 30, 31, 32, 33, 34, 35, 36, 40

R

Rede neural artificial 15

S

Satplan 1, 2, 3, 4

ENGENHARIAS ELÉTRICA E DE COMPUTAÇÃO:

O TERCEIRO PILAR

-  www.atenaeditora.com.br
-  contato@atenaeditora.com.br
-  [@atenaeditora](https://www.instagram.com/atenaeditora)
-  www.facebook.com/atenaeditora.com.br

 Atena
Editora

Ano 2021

ENGENHARIAS ELÉTRICA E DE COMPUTAÇÃO:

O TERCEIRO PILAR

-  www.atenaeditora.com.br
-  contato@atenaeditora.com.br
-  [@atenaeditora](https://www.instagram.com/atenaeditora)
-  www.facebook.com/atenaeditora.com.br

 Atena
Editora

Ano 2021