EAMC2021

XIV Encontro Acadêmico de Modelagem Computacional

Laboratório Nacional de Computação Científica - LNCC/MCTIC

08 a 11 de Fevereiro de 2021.

Livro de resumos

XIV Encontro Acadêmico de Modelagem Computacional

Laboratório Nacional de Computação Científica - LNCC/MCTIC

08 a 11 de Fevereiro de 2021. Petrópolis - RJ.

Comitê Organizador

Ana Luiza Martins Karl - LNCC

Daniel Raom Santiago - UnB

Guilherme Guilhermino Neto - UFJF

Gustavo Alves Bezerra - LNCC

João Vitor de Oliveira - UFRJ

Luis Alonso Mansilla Alvarez - LNCC

Luis Fernando Mendes Cury - LNCC

Matheus Muller Pereira da Silva - LNCC

Comitê Científico

Caio César Graciani Rodrigues - LNCC

Camila de Oliveira Vieira - UnB

Felipe Figueredo Rocha - EPFL

Gregório Kappaun Rocha - IFF

Guilherme Guilhermino Neto - UFJF

Heber Lima da Rocha - IUB

Juliana Vianna Valério - UFRJ

Karina Baptista dos Santos - LNCC

Lucas dos Santos Fernandez - LNCC

Marcel Duarte da Silva Xavier - UFF

Weslley da Silva Pereira - UFJF

XIV Encontro Acadêmico de Modelagem Computacional

Laboratório Nacional de Computação Científica - LNCC/MCTIC

08 a 11 de Fevereiro de 2021. Petrópolis - RJ.

Apresentação

O Encontro Acadêmico de Modelagem Computacional do LNCC (EAMC/LNCC) é um evento dedicado à Modelagem Computacional, que busca promover maior integração entre os alunos, docentes, pesquisadores e demais profissionais da área de Ciência e Tecnologia. O EAMC é inteiramente organizado por discentes do LNCC e de outras instituições colaboradoras, sendo composto por minicursos, mesas redondas, palestras, além de apresentações de trabalhos em formato pôster e apresentações orais.

Criado em 2007, o EAMC tem como principal objetivo proporcionar um ambiente científico que promova a divulgação científica, estimule a cooperação entre os profissionais da área, além de fomentar a multidisciplinaridade na formação de novos profissionais. O EAMC abrange diferentes áreas do conhecimento, tais como Computação Científica, Controle e Filtragem de Sistemas Dinâmicos, Modelagem de Biossistemas e Bioinformática, Modelagem de Circulação e Transporte, Modelagem de Equilíbrio e Otimização e, mais recentemente, Ciência de Dados e áreas correlatas, como Inteligência Artificial.

Em 2021, o XIV Encontro Acadêmico de Modelagem Computacional (EAMC) foi realizado no Laboratório Nacional de Computação Científica, em Petrópolis/RJ, nos dias 08 a 11 de fevereiro. Em razão da pandemia do SARS-CoV-2, os minicursos e apresentações orais foram transmitidas de forma virtual visando a segurança de todos os participantes.

Agradecimentos

O Comitê Organizador do XIV EAMC gostaria de agradecer à comunidade do LNCC, entre pesquisadores, professores, corpo discente e funcionários pela participação e dedicação ao evento.

Em particular, gostaríamos de expressar profunda gratidão à professora Sandra Malta, maior incentivadora deste evento, pela ajuda durante a organização, bem como à secretaria do Programa de Pós Graduação do LNCC, pelo apoio concedido – em particular à secretária Roberta Machado pela ajuda ativa e constante no decorrer do encontro.

Agradecemos de forma especial a disponibilidade e contribuição dos membros do Comitê Científico, dos palestrantes convidados e professores dos minicursos.

Finalizamos os agradecimentos esperando que o XIV EAMC tenha sido de grande valia à todos os participantes, sem os quais o sucesso dessa edição não seria possível.

Comitê Organizador
XIV Encontro Acadêmico de Modelagem Computacional - XIV EAMC
Laboratório Nacional de Computação Científica - LNCC

Trabalhos completos

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

How the spread of an infectious disease is affected by the contagion's probabilistic model

Beatriz Borges[1], Roberta Lima[1] e Rubens Sampaio[1]

[1] *Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro/RJ, Brasil*

Abstract

This work analyzes the spread of an epidemiological disease with a stochastic approach. In the analysis, the number of individuals that each infected member of the population can infect is modeled as a random variable and the number of individuals infected over time is modeled as a stochastic branching process. The focus of the work is to characterize the influence of the probabilistic model of the random variable that models the contagion between individuals in the spread of the disease and in the probability of extinction. The comparison is made based on histograms and sample statistics of the number of infected individuals over time, such as, mean and variance. Statistical models are computed using Monte Carlo simulations to 3 different families of random variables: binomial, geometric-1 and geometric-0. For each family, 21 different distributions were selected and, for each distribution, 2000 simulations of the branching process were computed. In total, 126 thousand simulations were performed, characterizing a big data problem.

Keywords: Stochastic Modeling, Uncertainties Propagation, Infectious Disease, Big Data

1 INTRODUCTION

The spread of infectious diseases is a subject of interest in many different research areas. The knowledge of the spread behavior over time can help governments to plan and take best control strategies. Accurate predictions of the evolution of the number of infected individuals over time can help, for example, in the organization of hospital supplies.

In the literature, there are several different models for the propagation of an infectious disease. Some examples are the compartmental models, as the susceptible-infectious (SI model) and the susceptible-infectious-removed (SIR model). In each of these models, the population is divided into groups (called compartments) and an initial value problem characterizes the evolution of the number of individuals in each of the groups [9], [3], [6].

Contact: Beatriz Borges, beatrizrbborges@gmail.com

Although these models are widely used, they approach the problem of propagation from a deterministic point of view. They do not consider the inherent random behavior of the contagion. For more accurate predictions, stochasticity should be taken into account.

This paper analyzes the spread of an epidemiological disease in a population with a stochastic approach, as found in [2] and [5]. In the analysis, the number of individuals that each infected member of the population can infect is modeled as a random variable and the number of individuals infected over time is modeled as a stochastic branching process. The focus of the work is to characterize the influence of the probabilistic model of the random variable that models the contagion between individuals in the spread of the disease and in the probability of extinction. The comparison is made based on histograms and sample statistics of the number of infected individuals over time, such as, mean and variance. The evolution of the histograms characterizes the uncertainty propagation in the spread of the disease [7], [8]. Statistical models are computed using Monte Carlo simulations to 3 different families of random variables: binomial, geometric-1 and geometric-0. For each family, 21 different distributions were selected and, for each distribution, 2000 simulations of the branching process were computed. In total, 126 thousand simulations were performed and over 2 million realizations of random variables were generated, characterizing a big data problem. The computations were developed in *Matlab* software and consumed more than 2 days of CPU time, in a MacOs system, 1.3 GHz Dual-Core Intel Core i5 processor with 4 GB 1600 MHz DDR3 memory.

2 CONSTRUCTION OF PROBABILISTIC MODELS TO THE CONTAGION AND SPREAD OF THE DISEASE

In this section, the probabilistic models of the contagion between individuals and the spread of the disease are constructed. The number of individuals that each infected member j of the population can infect is modeled as a random variable, C_j. It is assumed that all C_j are independent and identically distributed (IID). We call C the discrete random variable that models the contagion.

The number of individuals infected over time is modeled as a stochastic branching process I with discrete parameter. Given a set of discrete parameters $N = \{0, 1, 2, 3, \cdots\}$, representing the generations over time, for each $n \in N$, $I(n) = I_n$ is a discrete random variable that represents the number of infected individuals in the n^{th} generation. It is given by the sum of the number of individuals who were contaminated by each infected individual of the previous generation (generation $n-1$), characterizing it as a Markov process, that is:

$$I_n = \sum_{k=0}^{I_{n-1}} C_k, \qquad n > 0. \tag{1}$$

Figure 1 shows the first five generations of one realization of the branching process I. The process of spread starts with a single infected person $i_0 = 1$. In the next generation, $n = 1$, 3 new individuals are now infected by the initial one. This amount of new infected individuals is obtained by a sample of the discrete random variable C. In the following generation, $n = 2$, the total of infected individuals is the sum of the number of individuals who were contaminate by each one of the 3 infected individuals of the generation before. To achieve this number, 3 independent samples of C were generated, and the drawn

quantities were added. Given that the first individual infected only one person and, the two others have not infected anyone, $i_2 = 1$. In the next generation, $n = 3$, the single person infected in generation 2 infects 4 new individuals, thus $i_3 = 4$. In the generation $n = 4$, the total number of infected individuals, which is the sum of the number of individuals who were contaminated by each of the four infected in $n = 3$, is obtained generating four independent realizations of C and adding the amounts drawn. Given that, the first infected person infected three individuals, the second infected person did not make any contamination, and the third and fourth persons made two infections each, we have $i_4 = 7$. If none of these seven contaminated individuals make new contaminations, we will have $i_n = 0, \forall n \geq 5$ and, in this case, the disease will be extincted. The probability of extinction of a disease is one of the variables of interest in this work.

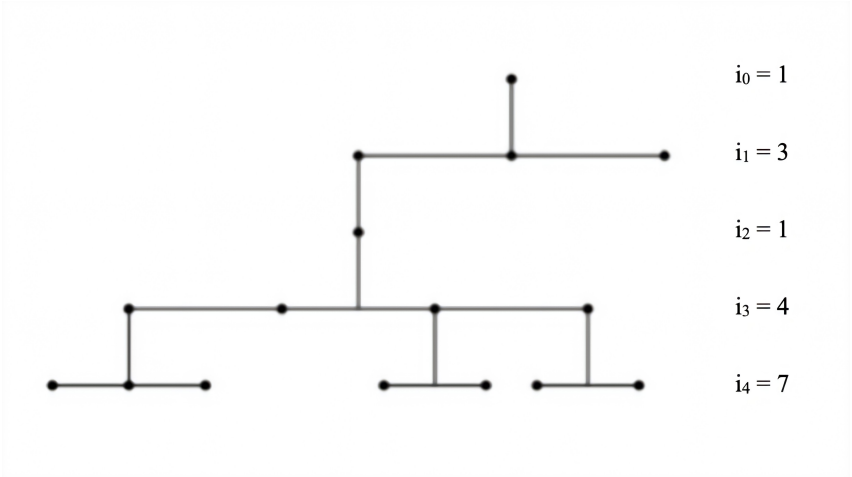


Fig. 1: One realization of the branching process that models the number of infected individuals over generations. It is considered $i_0 = 1$.

The probability generating function (pgf) of C is:

$$G_C(x) = \sum_{k=0}^{\infty} p_k \, x^k, \tag{2}$$

where each p_k is the probability of C taking the value k. According to theorem 4.36 of [4], the probability generating function (pgf) of I_n is given by:

$$G_{I_n}(x) = G_C(G_{I_{n-1}}(x)). \tag{3}$$

3 DISEASE SPREAD STATISTICS

This section shows some statistics of the stochastic process that models the spread of the disease,[1], [2] and [5]. According to theorem 9.8 and the theory of probability generating function from [4], given that $E(I_n) = E(I_{n-1})\mu$, the average of the number of infected individuals in each generation is $E(I_n) = \mu^n$, where $\mu = \sum_k k \; p_k$ is the mean of the random variable C. Long-term we have:

$$\lim_{n\to\infty} E(I_n) = \begin{cases} 0 \;\text{ if }\; \mu < 1, \\ 1 \;\text{ if } \mu = 1, \\ \infty \;\text{ if } \mu > 1. \end{cases} \tag{4}$$

Adapting theorem 9.8 from [4], it is possible to write the variance of the number of infected individuals in each generation as:

$$var(I_n) = \begin{cases} n\sigma^2 & \text{if } \mu = 1, \\ \sigma^2 \mu^{n-1}\left(\frac{\mu^n - 1}{\mu - 1}\right) & \text{if } \mu \neq 1, \end{cases} \tag{5}$$

where σ^2 the variance of C. Long-term:

$$\lim_{n\to\infty} var(I_n) = \begin{cases} 0 \text{ if } \mu < 1, \\ \infty \text{ if } \mu \geq 1. \end{cases} \tag{6}$$

Analyzing Eqs. 4 and 6, we observe that for $\mu \leq 1$ the epidemic will be extinct someday. For $\mu > 1$, on the other hand, the epidemic may or may not end. To determine the probability of extinction, we define two events. First event: A, extinction of the disease in the n^{th} generation, i.e., $I_n = 0$. Evaluating Eq. 3 at $x = 0$, we get:

$$\begin{aligned} G_{I_n}(x) &= G_C(G_{I_{n-1}}(x)) \\ G_{I_n}(0) &= G_C(G_{I_{n-1}}(0)) = e_n, \end{aligned} \tag{7}$$

where e_n is the probability of A. Second event: B, the disease extinction. By theorem 1.54 from [4], the probability of this event is given by $\lim_{n\to\infty} e_n$, represented by e. Therefore:

$$\begin{aligned} e_n = G_{I_n}(0) = G_C(G_{I_{n-1}}(0)) \quad &\rightarrow \quad e_n = G_C(e_{n-1}) \\ \lim_{n\to\infty} e_n = e \quad &\rightarrow \quad e = G_C(e), \end{aligned} \tag{8}$$

where the smallest root of the equation $e = G_C(e)$ in $[0, 1]$ is the probability of the epidemic extinction.

4 ESTIMATED STATISTICS, HISTOGRAMS, AND PROBABILITY OF EXTINCTION

To analyze the influence of the probabilistic model of the random variable that models the contagion between individuals in the spread of the disease and and probability of extinction, statistical models were computed for the disease propagation using Monte Carlo simulations to 3 different families of random variables C: binomial, geometric-1 and geometric-0. For each family, 21 different distributions were selected and, for each distribution, 2000 simulations of the branching process from generation $n = 0$ to $n = 20$ were computed. In total, 126 thousand simulations were performed. A convergence study was developed to determine the number of simulations [11]. A chart of the simulations is shown in Fig.2. In order to analyze the stochastic behavior of the epidemic over the generations, sample statistics, as mean, variance and probability of extinction, and normalized histograms of the number of infected individuals were computed for different generations of the branching process. With those histograms is possible to observe the propagation of uncertainties [10] in the spread of the disease. The most expressive results obtained with the Monte Carlo simulations are shown in the following sections.

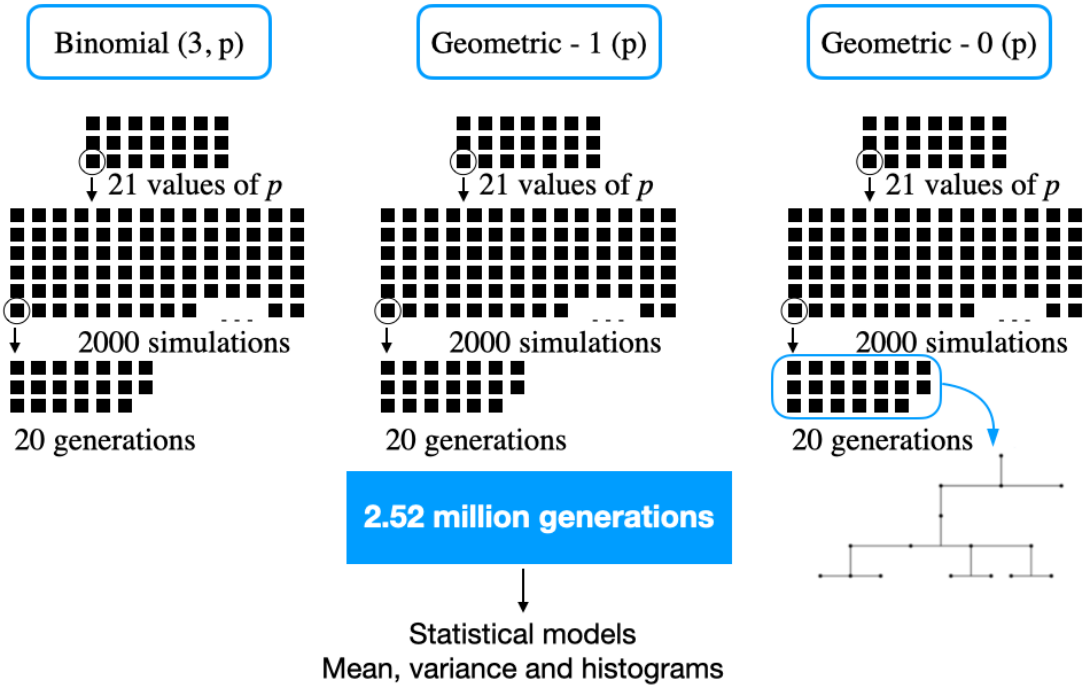


Fig. 2: Chart of the performed simulations.

4.1 Probabilistic model of C - binomial(m, p)

The first family chosen to C was the binomial(m, p) distribution. This distribution can be interpreted as an experiment in which an individual meets m others individuals and have probability p of infecting each one of them. After those meetings, we evaluate whether each one of the m individuals was infected, and count the number of new infected. A binomial(m, p) distribution has mean $\mu = mp$, variance $\sigma^2 = mp(1-p)$ and pgf $G_C(x) = (q + px)^m$. Figure 3 shows the normalized histograms of the number of infected individuals constructed for different generations of the branching process with contagion modeled by the binomial$(3, 0.5)$. Please observe the evolution of the function mass of the number of infected throughout the generations. Figure 4 shows the sample mean and variance of the process I over the generations for three different values of p ($p = 0.16$, 0.33 and 0.50). The results agree with the analytical results given by Eqs. (4) and (6). For $p = 0.16$, with $\mu = 0.5$, we can see from Fig. 4(a) that both average and variance of the stochastic process tends to zero, indicating that for $\mu < 1$ we have the extinction of the epidemic. For $p = 0.33$, with $\mu = 1.0$, we can see from Fig. 4(b) that the process average is around 1, while the variance grows linearly along of the generations. This result is in accordance with Eq. (5). For $p = 0.50$, with $\mu = 1.5$, we can see from Fig.4(c) that both mean and process' variance grow over the generations. This behavior indicates that $\lim_{n\to\infty} E(I_n) = \infty$ and $\lim_{n\to\infty} var(I_n) = \infty$, so that there is no possibility of epidemic extinction. According to Eq.(8), the probability of extinction of the disease, e, is given by $G_C(e) = (q+pe)^m = e$. Figure 5 shows the probabilities of extinctions over the generations for the three values of p set above. Note that, with probability $e = 1$, the epidemic tends to extinguish for values of $\mu \leq 1$, and that for values of $\mu < 1$, it extinguishes with values of $e \in [0, 1]$.

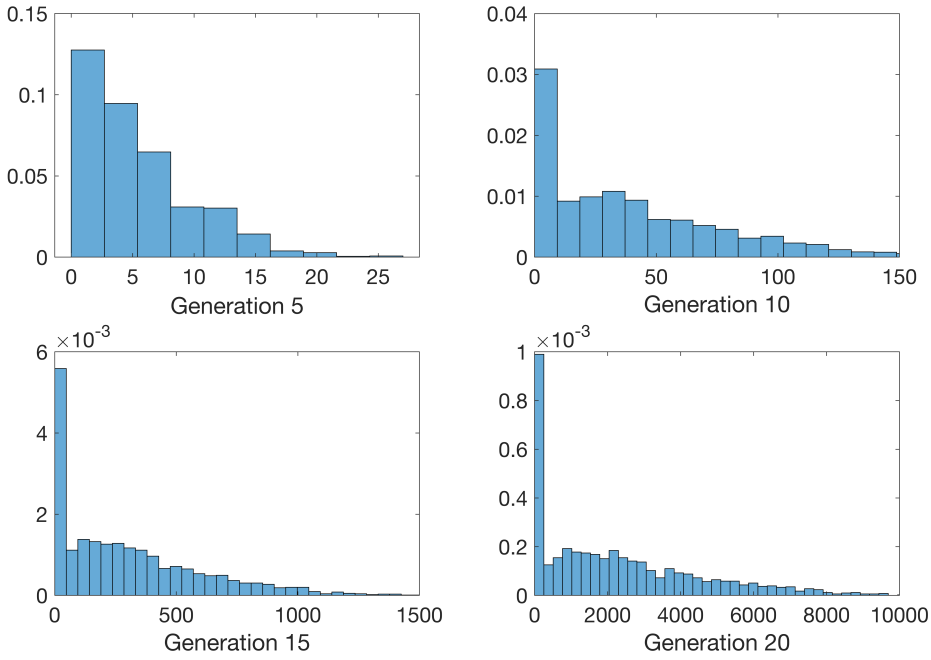
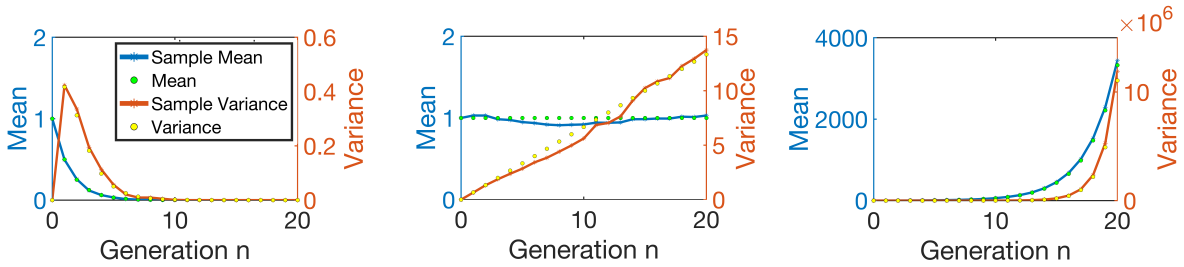
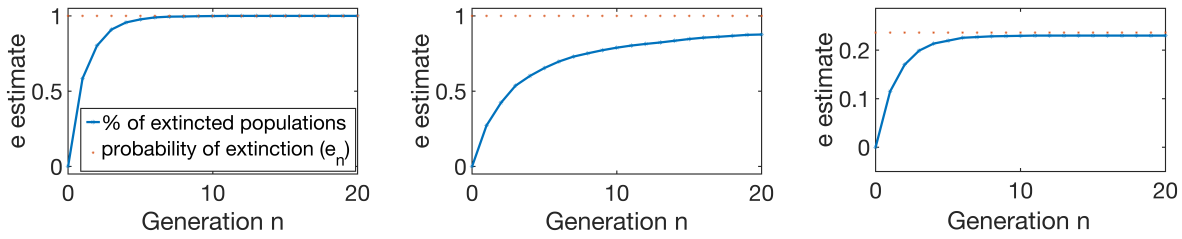


Fig. 3: Normalized histograms of the number infected individuals in the 5^{th}, 10^{th}, 15^{th} and 20^{th} generations constructed with 2000 realizations of the branching process I,with $C =$ binomial(3, 0.50).



(a) $C =$ binomial(3, 0.16) (b) $C =$ binomial(3, 0.33) (c) $C =$ binomial(3, 0.50)

Fig. 4: Sample's and analytic mean and variance for the different values of p.



(a) $C =$ binomial(3, 0.16) (b) $C =$ binomial(3, 0.33) (c) $C =$ binomial(3, 0.50)

Fig. 5: Analytic and sample's probability of extinction for the different values of p.

4.2 Probabilistic model of C - geometric-1(p)

The second family chosen to C was the geometric-1(p) distribution. On this less conservative analysis, an infected individual has no limit of individuals that he can infect, and, always, infects at least one person. What is singular in this analysis is that: if every individual infects at least one other individual, this epidemic will never be over. This model depicts a society with low, or not at all, restrictions concerning social distancing. The geometric-1 distribution has its pgf given by $G_C(x) = \dfrac{px}{1-(1-p)x}$, its mean $\mu = \dfrac{1}{p}$, and variance $\sigma^2 = \dfrac{1-p}{p}$. For every value of $p \neq 0$ we have $\mu \geq 1$, in agreement with the fact that the epidemic will never get extinguished. This implies that the probability of extinction is $e = 0$.

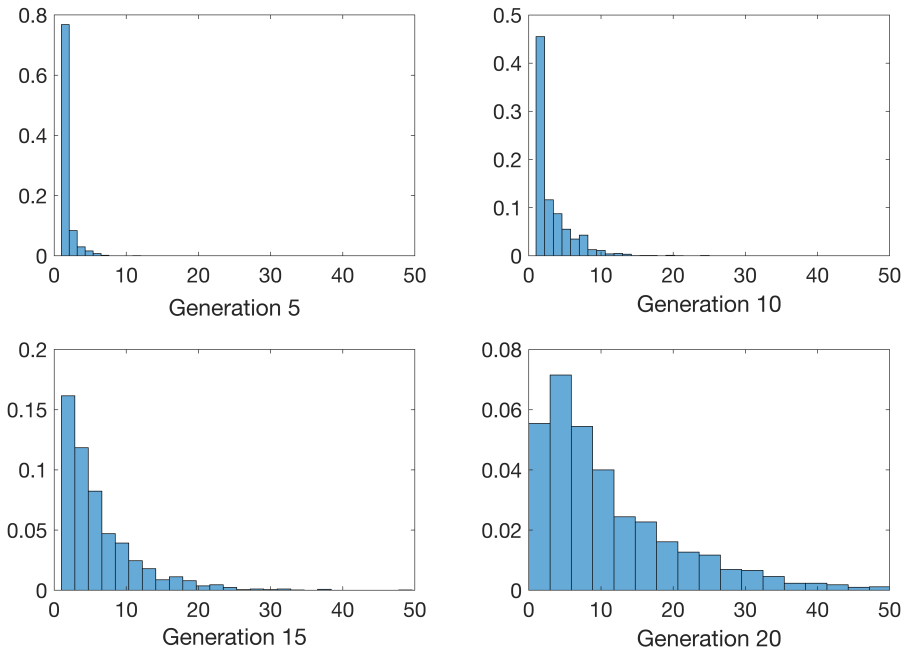
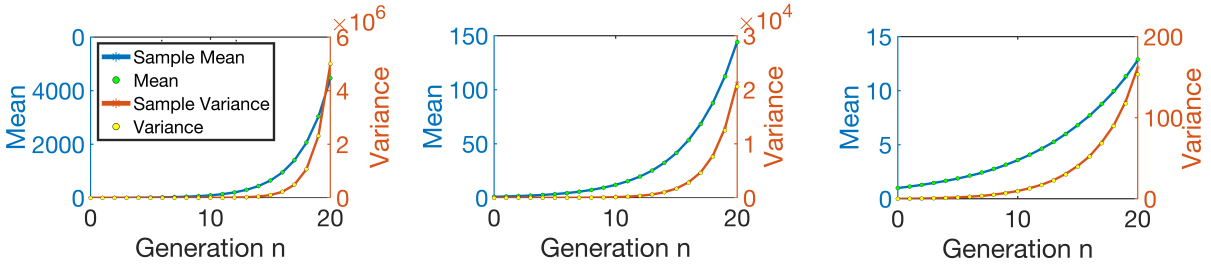
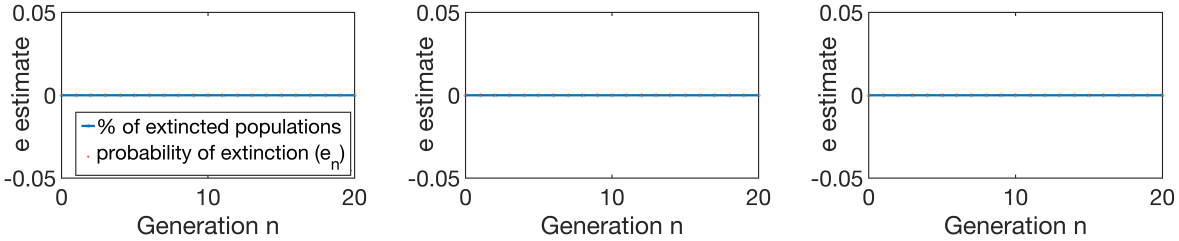


Fig. 6: Normalized histograms of the number infected individuals in the 5^{th}, 10^{th}, 15^{th} and 20^{th} generations constructed with 2000 realizations of the branching process I, with C = geometric-1(0.88).



(a) C = geometric-1(0.68) (b) C = geometric-1(0.78) (c) C = geometric-1(0.88)

Fig. 7: Sample's and analytic mean and variance for the different values of p.



(a) C = geometric-1(0.68) (b) C = geometric-1(0.78) (c) C = geometric-1(0.88)

Fig. 8: Analytic and sample's probability of extinction for the different values of p.

Figure 6 shows the normalized histograms of the number of infected individuals constructed for different generations of the branching process with contagion modeled by the geometric-1(p), with $p = 0.88$. Figure 7 shows the agreement between the sample statistics, mean and variance, and the analytic ones. Figure 8 shows the probability of extinction $e_n = 0$ over the generations.

4.3 Probabilistic model of C - geometric-0(p)

The third family chosen to C was as the geometric-0(p). In this scenario, an infected individual has no limit of individuals that he can infect, but he has the possibility of not infecting anyone. Geometric-0 portrays a society with low restrictions of social distancing, but with some isolated individuals. The pgf of the geometric-0 distribution is $G_C(x) = \dfrac{p}{1-(1-p)x}$, and its mean is $\mu = \dfrac{1-p}{p}$. Now, as the individuals can go out and not infect someone, there is a non null probability of extinction of the epidemic. By Eq. 8, with C being geometric-0, the probability of extinction for $\mu > 1$ is given by $e = \dfrac{p}{1-p}$. Figure 10 shows the agreement between the sample statistics, mean and variance, and the analytic ones. Figure 11 shows the probability of extinction $e_n = 0$ over the generations. Figure 9 shows the normalized histograms of the number of infected individuals constructed for different generations of the branching process with contagion modeled by the geometric-0(p), with $p = 0.4$.

5 CONCLUSIONS

This work analyzes with a stochastic approach the spread of an epidemiological disease in a population. The number of individuals infected by each infected member of the population was modeled by a discrete random variable and the number of infected individuals over the time was modeled by a stochastic branching process. To characterize the propagation of uncertainties in the spread of the disease, histograms of the number of infected individuals were constructed. Sample statistics were also calculated. Analysis of the influence of the probabilistic model of the random variable that models the contagions between individuals in the branching process were made and the probability of extinction was calculated for three different families of random variables.

The results of the work show that the parameters of the probabilistic model of the random variable that models the contagion, C, strongly influence the propagation's behavior of the disease. C being a binomial(m, p), the number of individuals infected by

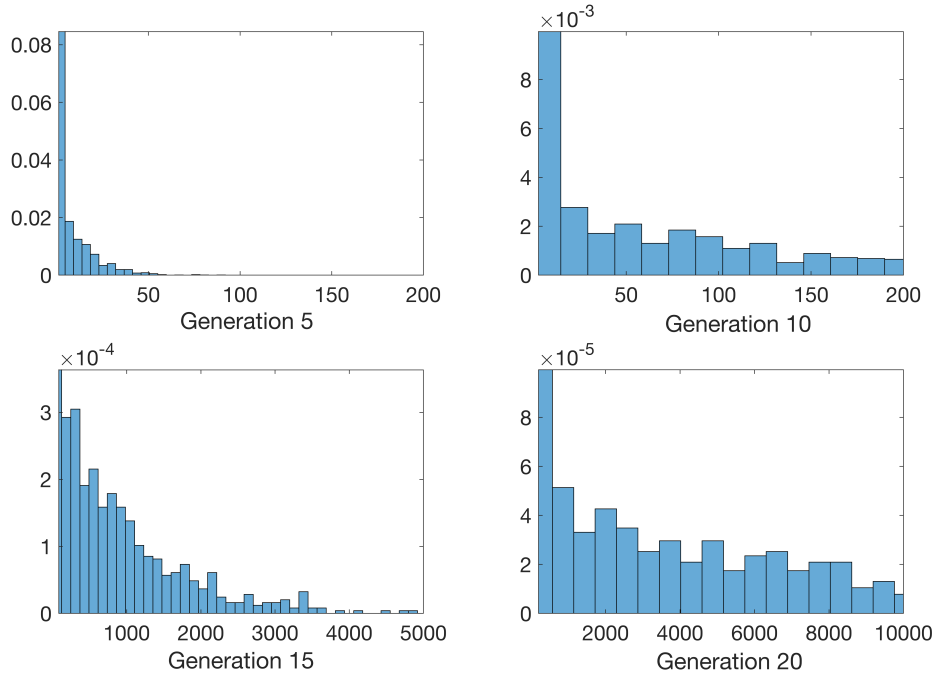
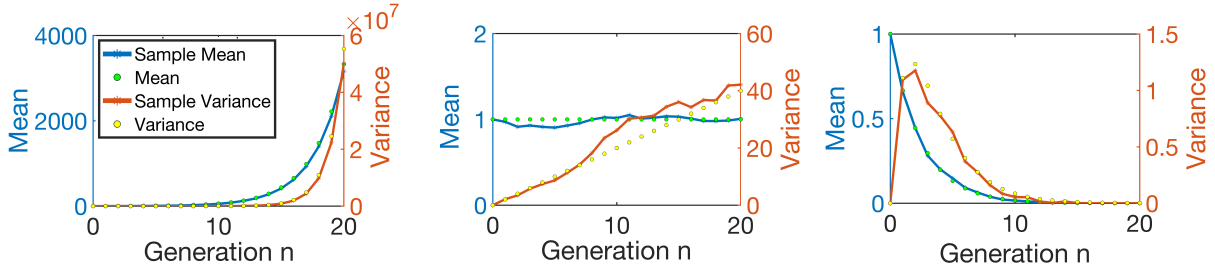
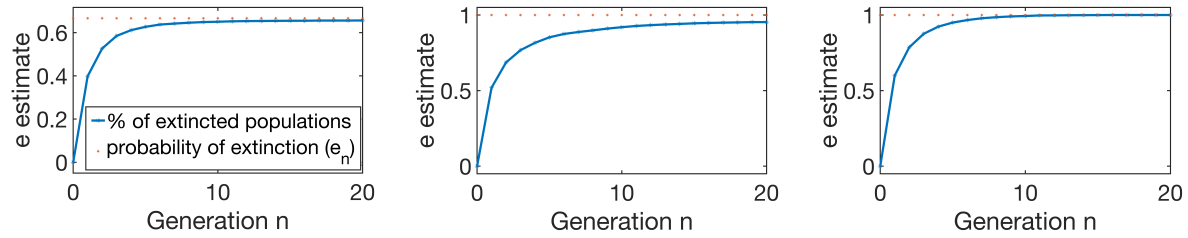


Fig. 9: Normalized histograms of the number infected individuals in the 5^{th}, 10^{th}, 15^{th} and 20^{th} generations constructed with 2000 realizations of the branching process I, with $C =$ geometric-0(0.4).



(a) $C =$ geometric-0(0.4) (b) $C =$ geometric-0(0.5) (c) $C =$ geometric-0(0.6)

Fig. 10: Sample's and analytic mean and variance for the different values of p.



(a) $C =$ geometric-0(0.4) (b) $C =$ geometric-0(0.5) (c) $C =$ geometric-0(0.6)

Fig. 11: Analytic and sample's probability of extinction for the different values of p.

each infectious member is limited to m individuals. C being either geometric-1(p) or

geometric-0(p) there is no limit on how many new individual can be infected. As has been seen, for the binomial(m, p) and geometric-0(p), for values of $\mu \leq 1$ the epidemic is extinguished with probability $e = 1$. Whereas, for $\mu > 1$, it can be extinguished with probability $e \in [0.1]$. For the geometric-1 distribution, since for every $p \neq 0$ the mean is $\mu > 1$, the probability of extinction is $e = 0$.

6 *Acknowledgments*

The authors acknowledge the support given by FAPERJ, CNPq, and CAPES.

References

[1] L. Allen. *An Introduction To Stochastic Processes With Applications To Biology*. CRC Press, New York, USA, 2010.

[2] L. Allen. A primer on stochastic epidemic models: Formulation, numerical simulation, and analysis. *Infectious Disease Modelling*, 2:128–142, 2017.

[3] F. Brauer, C. Castillo-Chavez, and Z. Feng. *Mathematical Models In Epidemiology*. Springer, New York, USA, 2019.

[4] G. Grimmett and D. Welsh. *Probability An Introduction*. Oxford University Press, Oxford, GBR, 2014.

[5] P. Haccou, P. Jagers, and V. Vatutin. *Branching Processes: Variation, Growth, and Extinction of Populations*. Cambridge University Press, Cambridge, GBR, 2017.

[6] M. Li. *An Introduction To Mathematical Modeling of Infectious Diseases*. Springer, Switzerland, 2018.

[7] R. Lima and R. Sampaio. How to deal with uncertainty quantification and propagation. *Asociación Argentina de Mecánica Computacional*, XXXVI:723–739, 2018.

[8] R. Lima and R. Sampaio. What is uncertainty quantification? *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 40:155, 2018.

[9] M. Martcheva. *An Introduction To Mathematical Epidemiology*. Springer, New York, USA, 2015.

[10] R. Sampaio and R. Lima. *Modelagem Estocástica e Geração de Amostras de Variáveis e Vetores Aleatórios*, volume 70. Notas de Matemática Aplicada, SBMAC, São Carlos, SP, Brazil, 2012.

[11] J. E. Souza de Cursi and R. Sampaio. *Modelagem Estocástica e Quantificação de Incertezas*, volume 66. Notas de Matemática Aplicada, SBMAC, São Carlos, SP, Brazil, 2012.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Estimation of Stellar Parameters for J-PLUS Survey with Machine Learning

C. A. Galarza[1], S. Daflon[1], V. M. Placco[2,3] and C. Allende-Prieto[4,5]

[1] *Observatório Nacional - MCTI (ON), Rua General José Cristino 77, São Cristóvão, 20921-400, Rio de Janeiro, Brazil*
[2] *Department of Physics, University of Notre Dame, Notre Dame, IN. 46556, USA*
[3] *JINA Center for the Evolution of the Elements (JINA-CEE), USA*
[4] *Instituto de Astrofísica de Canarias, Vía Láctea S/N, 38205 La Laguna, Tenerife, Spain*
[5] *Universidad de La Laguna, Departamento de Astrofísica, 38206 La Laguna, Tenerife, Spain*

Abstract

The amount of data produced by the current and future astronomic surveys will require to develop more efficient ways of processing it in order to carry out scientific researches that lead us to more interesting discoveries. One particular example is the identification of extremely low metallicity stars ($[Fe/H] < -3.0$) that can allow us to imposed constraints improving our current understanding of the formation and chemical evolution of our galaxy. To accomplish this goal we present the J-PLUS Stellar Parameters Estimation based on Ensemble Methods (J-PLUS SPEEM) pipeline that is capable of performing a variety of tasks suchs as morphological and spectral classification along with the estimation of the three main stellar parameters (T_{eff}, $[Fe/H]$ and $\log g$) with absolute mean errors of 139.2 K, 0.16 dex and 0.32 respectively.

Keywords: Machine learning, Astroinformatics, Stellar parameters, Low-metallicity stars

1 INTRODUCTION

The main goal of any astronomical observation is the determination of the physical properties of observed objects in order to study its chemical composition which provide reasonable arguments about how they were formed and how they will evolve.
Particularly, in the case of stars one can estimate its surface parameters such as effective temperature T_{eff}; metallicity which is usually expressed in terms of the iron-to-hydrogen ratio $[Fe/H]$; and surface gravity $\log(g)$ based on the light received from them and collected by a telescope or satellite. To accomplish this goal there are at least two different

Contato: Andres Galarza, carlosgalarza@on.br

approaches that we can use in order to obtain data through observations: these are Photometry and Spectrometry. The first offers the advantage of better signal-to-noise ratios and the collection of data from multiple sources at the same time in exchange of losing the ability to track individual spectral features. On the other hand, spectrometry allows you to make a detailed study of chemical abundances (high resolution observations); with the disadvantange of being extremely difficult to obtain spectra from many sources at the same time (even using multi-slit or multi-fiber spectrographs).
Different photometric calibrations have been tested in order to make a reliable estimations of stellar parameters. For instance, Ivezić et al. [10] found that $(g-r)$ color from SDSS provides an accurate estimation for T_{eff} and $[Fe/H]$ can be obtained through a polynomial fitting using $(u-g)$[1] with some restrictions on $(g-r)$ while Casagrande et al. [5] used the infrared flux method (Casagrande et al. [4]) to obtain T_{eff} of a sample of F, G and K type stars using $BVJHK_s$ photometry. Regarding spectroscopic observations, Lee et al. [11] presented the SEGUE Stellar Parameter Pipeline (SSPP) which made extimations of stellar parameters using both theoretical and empiric calibrations of medium resolution stellar spectra ($R \approx 10000$) along with the implementation of neural networks based on spectral libraries (ELODIE and MILES; Prugniel and Soubiran [14], Falcón-Barroso J. et al. [7]) and high-resolution spectra for the validation process.
Nevertheless, in the next years, the data sets produced by different large surveys will be so huge (the order of many Petabytes) that we will need to come up with new effective ways to deal with it. In this sense, machine learning algorithms and any other statistical tools could be extremely helpful to analyze data and make predictions in a reasonable short amount of time. Some machine learning methods like artificial neural networks (ANN) have been applied successfully for somme time ago in a wide variety of astronomical applications such as Gulati and Gupta [8] that propose a model to estimate $E(B-V)$[2] for O and B stars; and Whitten et al. [16] that implemented ANNs to estimate T_{eff} and $[Fe/H]$ for J-PLUS data in order to search for low-metallicity stars. On the other hand, we can find that other models based on algorithms such as Random Forest (Breiman [3]) have produced interesting results in terms of morphological classifications and estimation of physical parameters. For instance, Miller [13] presented a Random Forest (hereafter RF) model capable of inferring T_{eff}, $\log g$ and $[Fe/H]$ based on SDSS de-reddened colors; Bai et al. [1] built a RF model that perform a Star-Galaxy-QSO[3] classification and calculate the T_{eff} for stars using data from SDSS and LAMOST.
In this work, we present the J-PLUS Stellar Parameters Estimation based on Ensemble Methods pipeline (J-PLUS SPEEM) which consist on a series of machine learning models that are capable of performing different tasks suchs as separating stars from QSOs, estimating three main stellar parameters (T_{eff}, $[Fe/H]$, $\log g$), and make spectral classification (A, F, G, K and WDs[4]) which can allow us to build a catalog of stars for a wide variety of interests. We also tested the model for looking for new very metal poor stars observed with the Javalambre Photo-metric Local Universe Survey (J-PLUS, Cenarro et

[1] u, g and r are photometric magnitudes measured by the optical filter system of Sloan survey.
[2] A measure of the interstellar reddening that is related to the quality of the light received by a telescope from the observed stars
[3] quasi-stellar object: A type of galaxy with extremely active nucleus that appear as a bright star
[4] White Dwarf: A kind of star in its final evolutionary state

al. [6]) that have not been identified before thanks to the comparisson between machine learning predictions and analysis of medium resolution spectra obtained with the William Hershel Telescope (WHT) at Canary Islands.

2 METHODOLOGY

There are two different strategies in order to properly implement machine learning algorithms which allow one to obtain reliable estimations and predictions on astronomic data; these are: the *supervised learning* and the *unsupervised learning*. In the supervised learning one usually start with a fraction of data for which the target variables of interest are well known, being T_{eff}, $[Fe/H]$, and $\log g$ in this work. Then an optimized algorithm is applied to search for statistical relations between these target variables an a set measured parameters, being photometric magnitudes, colors[5] and in our case. The data used to applied the algorithm is known as the training sample, the variables to be estimated are refered as labels (for classification problems) or targets (for regression), while the input parameters considered to deploy a model are simply called features. In contrast to supervised methods are very useful to make regressions and classifications unsupervised learning is based on unknown labels in order to find posible patterns or associations from the features that can be interpreted as classes. [2][9]

For the purpose of this work we restrained the focus to the application of the random forest algorithm which is a supervised learning method.[3]

2.1 Training Datasets

The sample used in this work was obtained applying a query to retrieve magnitudes in the dual AB system from the second data release of JPLUS Survey (hearafter JPLUS DR2[6]). Additionally, selections of 6 arcsec aperture photometry and proper configuration in MASKS and FLAGS values in the query were made to ensure high quality measurements in each one of the 12 filters, and the stellar wide dwarf loci photometric calibration proposed by López-Sanjuan et. al. [12] was also taken into account.
We ended up with a sample of 575,593 (hereafter refered as gold sample) objects which in principle should be considered as unknown/unlabelled data. Then the gold sample was crossed with other surveys such as Sloan DR12[7], SEGUE(Yanny et al [18]), LAMOST and WISE (Wright et. al. [17]) in order to gain new information regarding the target variables. From this strategy 7393 objects (referred as JPLUSxSLOANxWISE) were found in common with both Sloan DR12 and WISE that provides information on morphological classification. Another 9436 objects (JPLUSxSSPP) were found in commmon with SEGUE which is a spectroscopic survey of Sloan that allows us to get information regarding stellar parameters (T_{eff}, $[Fe/H]$, and $\log g$). Finally we also found and 106769 (JPLUSxLAMOST) objects in common with LAMOST DR6[8] for testing the predictions of the model.

[5]A photometric color is defined as the subtraction between two magnitudes.
[6]http://www.j-plus.es/datareleases/data_release_dr2
[7]https://www.sdss.org/dr12/
[8]http://dr6.lamost.org

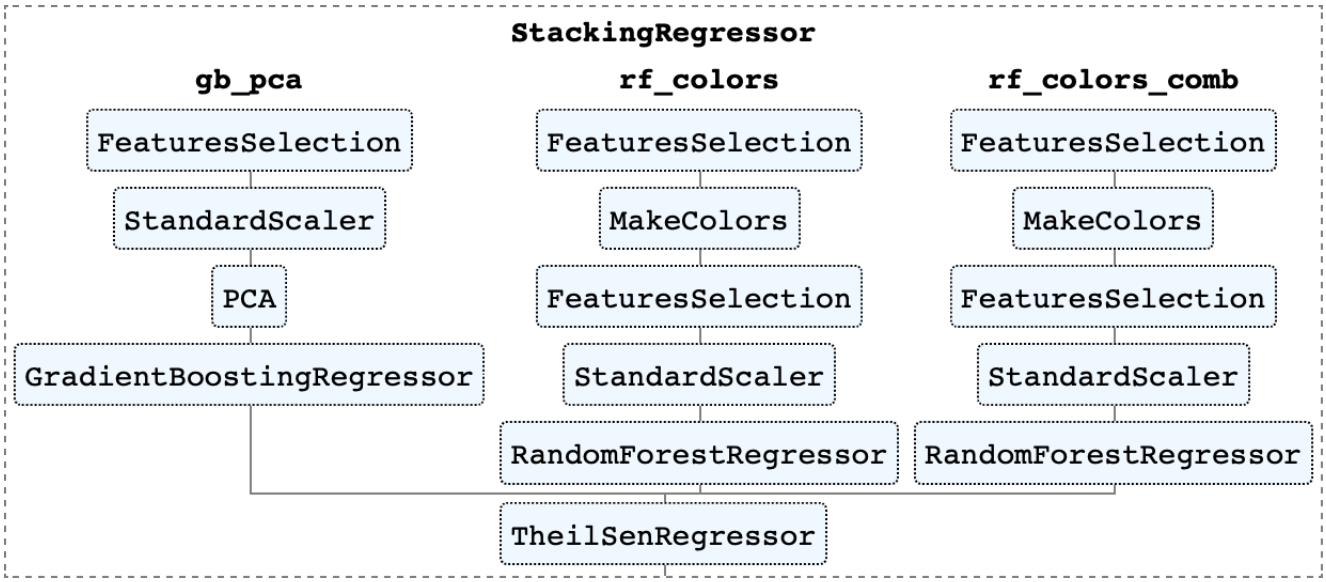


Fig. 1: SPEEM Architecture.

2.2 Architecture of SPEEM

SPEEM is in charge of two main tasks. The first one is two make a morphological classification two distinguish between different types of stars and contaminants such as QSOs. Then the second task that SPEEM accomplishes is to estimate the stellar parameters through a stacking regressor that combines three different emsemble regressors passing into a final Theil-Sen estimator as shown in Figure 1. Each one of the emsemble regressors is based on different set of features applying a PCA analysis and different combinations of photometric colors.

3 RESULTS AND DISCUSSION

3.1 Morphological and Spectral Classification

In oder to estimate stellar parameters from a sample we need to make sure that no contaminants are present in the data; like QSOs or WDs in this case. So, the first step consisted on attempting a morphological classification of the data. For this purpose we used the JPLUSxSLOANxWISE dataset as the training data in order to build a model that allows us to make a preliminar classification of DR1 which can helps us to clean the sample as best as possible (keeping a pure sample of stars). In figure 2 we can see that without applying any model we can use W1, W2, and W3 magnitudes from WISE (that fortunately has a lot of common fields with JPLUS) in order to classify at least Stars from QSOs. This color-color diagram is widely discussed by Wright et al. [17]; and Scaringi et al. [15]. Recall that WISE magnitudes are located in the infrared region of the electromagnetic spectrum while JPLUS magnitudes belong to optical part; and in general, QSOs tend to be redder than Stars in respect to WISE colors.

On the other hand the purpose of identifing possible White Dwarfs is to exclude them from the sample, otherwise they will be mixed with stars with very low metallicity the moment a photometric estimation is applied, resulting in many false positives. This misinterpretation had been previosly detected in the analysis of SEGUE and BOSS data

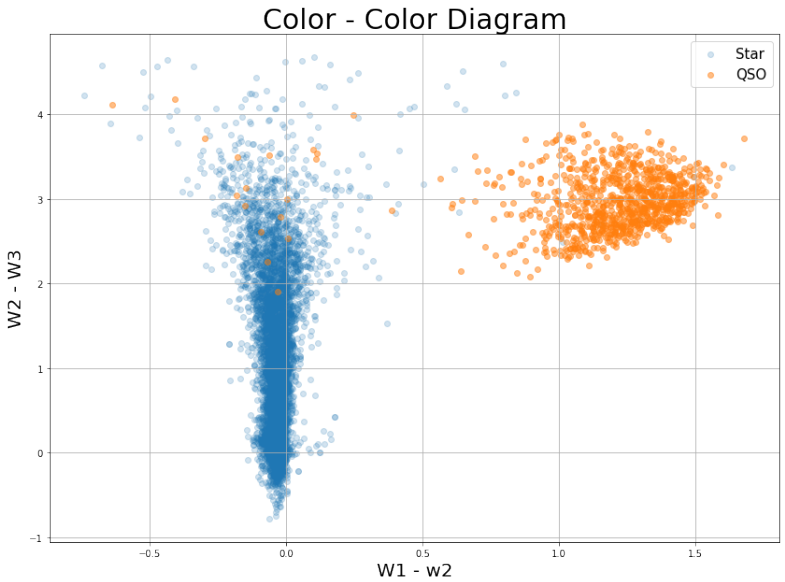


Fig. 2: Color-color diagram using W1, W2, W3 magnitudes from WISE.

Confussion Matrix		
Actual / Predicted Morph. Type	QSO	STAR
QSO	224	5
STAR	6	1614

Table 1: Validation of Stars/QSOs separation model with accuracy score of 0.99.

from Sloan. It is very important to mention that in this work we are not interested in making a precise spectral classification of stars that composed the sample, we just want to identify and separate WDs froms our sample of interest. A classification report containing metrics regarding the precision of the model is presented in Table 2. The term Galaxy refers to broadband sources identified by SEGUE and Recall means how well the objects were classified in the test sample. Lower values on G-type stars are expected since they can be easily mistaken as F or K type stars due to its limited range of temperature.

3.2 Stellar Parameters Estimations - T_{eff}, $[Fe/H]$ and $\log g$

3.2.1 Effective Temperature T_{eff}

This parameter is usually the easiest to calculate using photometry. Specifically in the case of the JPLUS survey one can define color indexes either based on broad or narrow band filters in order to fit some polynomial function to adjust the data in the same way that was done by Bond and Izevic [10] with the formula.

$$\log(T_{eff}) = F(g - r) \tag{1}$$

Nevertheless in this work a machine learning model was deployed based on different color indexes taking advantage of the 12 filter system. This model was trained on a sample of 8523 stars with temperatures between 4200 K and 9200 K Figure 3 shows the validation results for a test sample of more than 100000 stars that can be used to test the performance

Classification Report				
Spectral Class	Precision	Recall	f1-score	Support
A	0.88	0.86	0.87	268
F	0.86	0.93	0.89	1434
G	0.58	0.37	0.45	222
Galaxy	1.00	0.88	0.93	8
K	0.91	0.82	0.87	205
M	0.71	0.83	0.77	6
WD	1.00	0.62	0.77	8
accuracy			0.85	2151
macro avg	0.85	0.76	0.79	2151
weighted avg	0.84	0.85	0.84	2151

Table 2: Validation of Spectral Class model based on photometric magnitudes with accuracy score of 0.99.

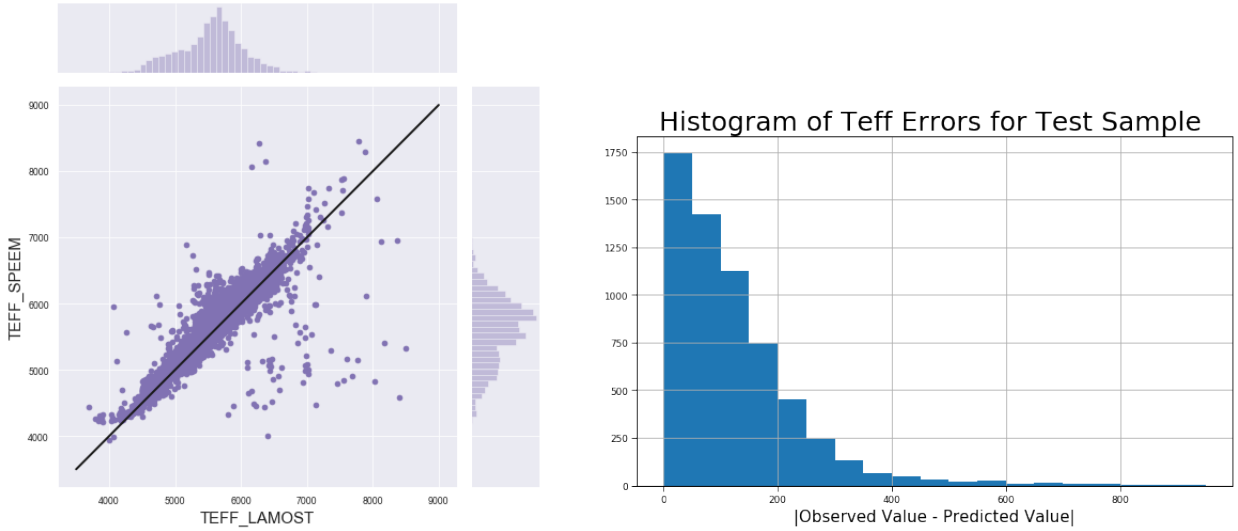


Fig. 3: Validation of effective temperature estimations for a test sample of nearly 106769 stars from JPLUSxLAMOST.

of SPEEM. The left side presents a good agreement between the predicted and adopted values while the right side shows the distribution of error for the predicted values by SPEEM. A mean absolute error of 139.5 K was calculated for the validation process.

3.2.2 Metallicity $[Fe/H]$

This is another physical parameter suitable of photometric estimation. Its determination is usually more challenging than T_{eff} due to high sensitivities to signal-to-noise ratio observation which decreases the precision of the measurements of some filters specially in the blue region of the spectrum.
Since one of the goals of this work is to identify possible candidates to very metal poor stars (VMPs) we need to train the model on a sample tha contains a wide range of $[Fe/H]$ values. The sample used contains stars of $[Fe/H]$ between -3.0 dex and 0.5 dex

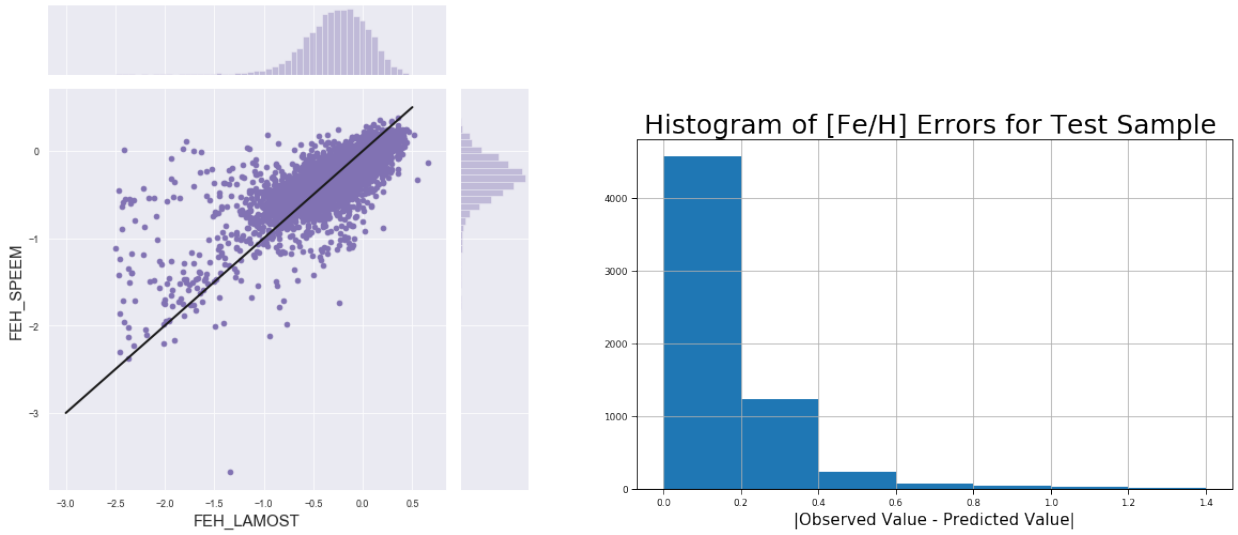


Fig. 4: Validation of metallicity estimations for a test sample of nearly 106769 stars from JPLUSxLAMOST.

and figure 4 shows the validation test results for the same validation sample used in the estimation of T_{eff}. The left side presents a reasonable agreement between the predicted and the adopted values with some outliers that require further analysis. On the right side the histogram of errors suggest that the majority of estimations presents an error of less of equal than 0.2 dex. The mean absolute error calculated was 0.16 dex.

3.2.3 Surface Gravity $\log g$

The same process applied to calculate T_{eff} and $[Fe/H]$ was repeated to estimate $\log g$ and figure 5 shows the results of the validation test. In this case a wider dispersion can be appreciated compared to the previous cases. The range of values used in the training data were between $[1.0, 5.0]$ dex. The results obtained for this particular parameter suggests that either additional restrictions are needed or the input features used by the machine larning algorithms are not enough to develop a higher precision model. Despite of the bigger dispersion, and the outliers presented the histogram of errors suggest that the majority of estimations presented an error between 0 and 0.5 dex with a calculated mean absolute error of 0.32 dex.

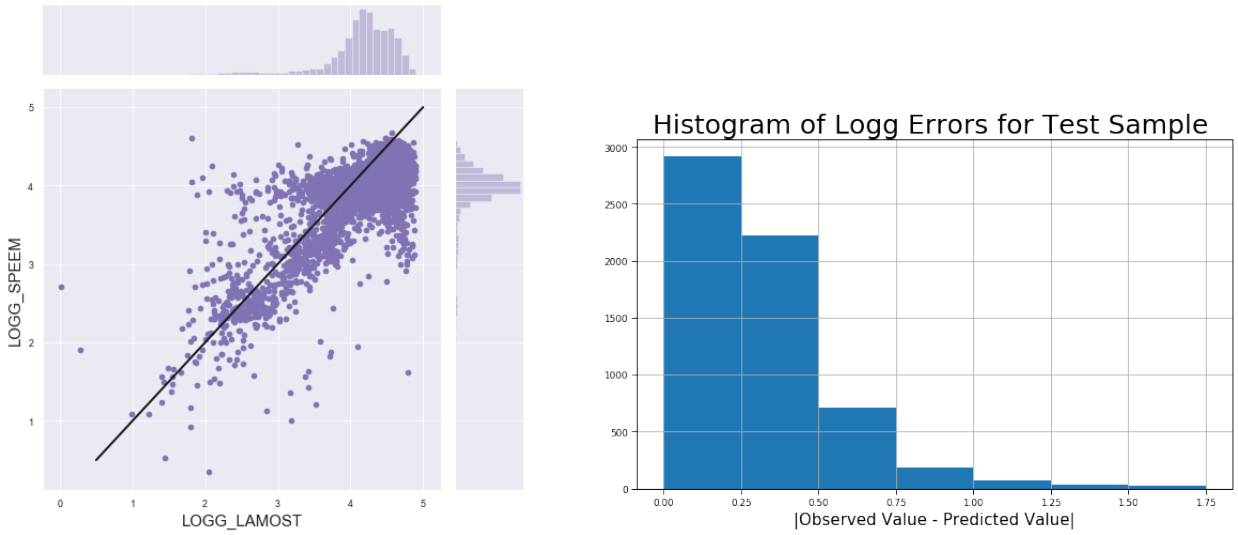


Fig. 5: Validation of surface gravity estimations for a test sample of nearly 106769 stars from JPLUSxLAMOST.

3.3 Spectroscopical confirmation of VMPs candidates

Once the model was tested it was applied to a sample of one million stars in order to search for new very metal poor stars candidates to validate with spectroscospic analysis. A total of 11 interesting stars were selected and the observations were carried out by Carlos Allende Prieto using the William Herschel Telescope (WHT). To obtain the spectroscopic values for $T_{eff}, [Fe/H]$ and $\log g$ the pipeline n-SSPP (Beers et. al 2014) was used. Comparisson between the parameters estimated by the model and those obtained by spectroscopic analysis are shown in figure 6. Systematic median error of 178 K, 0.76 dex and 0.5 dex were obtained for T_{eff}, $\log g$ and $[Fe/H]$ respectively.

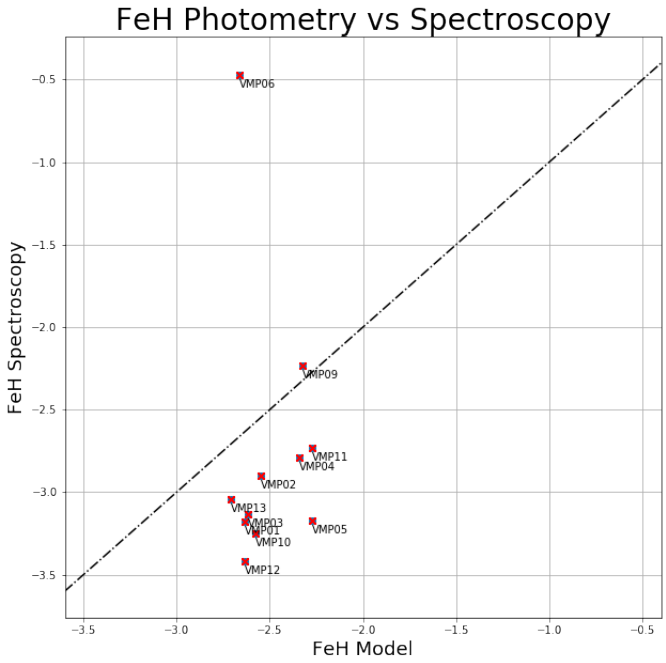


Fig. 6: Validation of metallicity estimations for a sample of nearly 1945 stars from JPLUSxSSPP.

4 CONCLUSION

The primary goal of J-PLUS SPEEM is to identify good very metal poor stars candidates that can be used to detailed study its chemical abundances with high-resolution spectroscopy that is one of the most powerful observational methods up to date in the field of astrophysics. As it was shown specially in figures 4 and 6 the pipeline was able to identify new low metallicity stars which were further confirmed by independent spectroscopic analysis. There is a good agreement between the values predicted by J-PLUS SPEEM and those found by an independent spectroscopic analysis. We expect to find and validate more interesting candidates applying the pipeline to future data releases of the J-PLUS survey.

5 *Acknowledgements*

C. A. Galarza acknowledges the full finantial support from CAPES for the development of this Phd. project. All the authors thank to Observatorio Astrofísico de Javalambre in Teruel, managed and operated by the Centro de Estudios de Física del Cosmos de Aragón (CEFCA) in charge of the JAST/T80 telescope responsible of carrying out the observations for the J-PLUS survey. There is also a special acknowledgment to the Roque de los Muchachos Observatory in charge of the Isaac Newton group of telescopes where the William Herschel Telescope (WHT) belongs. A final mention to the Scikit-learn team for the great contributions to the machine learning libraries for the python language.

References

[1] Y. Bai, J. Liu, S. Wang, and F. Yang. Machine learning applied to star–galaxy–qso classification and stellar effective temperature regression. *The Astronomical Journal*, 157(1):9, 2018.

[2] C. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer, 2006.

[3] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[4] L. Casagrande. Infrared flux method and colour calibrations. *Physica Scripta*, 2008(T133):014020, 2008.

[5] L. Casagrande, I. Ramírez, J. Melendez, M. Bessell, and M. Asplund. An absolutely calibrated teff scale from the infrared flux method-dwarfs and subgiants. *Astronomy & Astrophysics*, 512:A54, 2010.

[6] A. J. Cenarro, M. Moles, D. Cristóbal-Hornillos, A. Marín-Franch, A. Ederoclite, J. Varela, C. López-Sanjuan, C. Hernández-Monteagudo, R. Angulo, H. V. Ramió, et al. J-plus: The javalambre photometric local universe survey. *Astronomy & Astrophysics*, 622:A176, 2019.

[7] J. Falcón-Barroso, P. Sánchez-Blázquez, A. Vazdekis, E. Ricciardelli, N. Cardiel, A. Cenarro, J. Gorgas, and R. Peletier. An updated miles stellar library and stellar population models. *Astronomy & Astrophysics*, 532:A95, 2011.

[8] R. Gulati, R. Gupta, and H. Singh. E (bv) determinations of o and b stars using artificial neural networks. *Publications of the Astronomical Society of the Pacific*, 109(737):843, 1997.

[9] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer series in statistics. Springer, 2001.

[10] M. Jurić, Ž. Ivezić, A. Brooks, R. H. Lupton, D. Schlegel, D. Finkbeiner, N. Padmanabhan, N. Bond, B. Sesar, C. M. Rockosi, et al. The milky way tomography with sdss. i. stellar number density distribution. *The Astrophysical Journal*, 673(2):864, 2008.

[11] Y. S. Lee, T. C. Beers, T. Sivarani, C. A. Prieto, L. Koesterke, R. Wilhelm, P. R. Fiorentin, C. A. Bailer-Jones, J. E. Norris, C. M. Rockosi, et al. The segue stellar parameter pipeline. i. description and comparison of individual methods. *The Astronomical Journal*, 136(5):2022, 2008.

[12] C. López-Sanjuan, J. Varela, D. Cristóbal-Hornillos, H. V. Ramió, J. Carrasco, P.-E. Tremblay, D. Whitten, V. Placco, A. Marín-Franch, A. Cenarro, et al. J-plus: photometric calibration of large-area multi-filter surveys with stellar and white dwarf loci. *Astronomy & Astrophysics*, 631:A119, 2019.

[13] A. Miller, J. Bloom, J. Richards, Y. Lee, D. Starr, N. Butler, S. Tokarz, N. Smith, and J. A. Eisner. A machine-learning method to infer fundamental stellar parameters from photometric light curves. *The Astrophysical Journal*, 798(2):122, 2015.

[14] J. Moultaka, S. Ilovaisky, P. Prugniel, and C. Soubiran. The elodie archive. *Publications of the Astronomical Society of the Pacific*, 116(821):693, 2004.

[15] S. Scaringi, P. Groot, K. Verbeek, S. Greiss, C. Knigge, and E. Körding. Spectroscopic identifications of blue-h α-excess sources in the kepler field of view. *Monthly Notices of the Royal Astronomical Society*, 428(3):2207–2215, 2013.

[16] D. Whitten, V. Placco, T. Beers, A. Chies-Santos, C. Bonatto, J. Varela, D. Cristóbal-Hornillos, A. Ederoclite, T. Masseron, Y. Lee, et al. J-plus: Identification of low-metallicity stars with artificial neural networks using sphinx. *Astronomy & Astrophysics*, 622:A182, 2019.

[17] E. L. Wright, P. R. Eisenhardt, A. K. Mainzer, M. E. Ressler, R. M. Cutri, T. Jarrett, J. D. Kirkpatrick, D. Padgett, R. S. McMillan, M. Skrutskie, et al. The wide-field infrared survey explorer (wise): mission description and initial on-orbit performance. *The Astronomical Journal*, 140(6):1868, 2010.

[18] B. Yanny, C. Rockosi, H. J. Newberg, G. R. Knapp, J. K. Adelman-McCarthy, B. Alcorn, S. Allam, C. A. Prieto, D. An, K. S. Anderson, et al. Segue: A spectroscopic survey of 240,000 stars with g= 14-20. *The Astronomical Journal*, 137(5):4377, 2009.

Predição do número de casos de SARS-CoV-2 através de análises no Esgoto da Cidade de Niterói-RJ utilizando Árvore de Regressão

Carmen Lúcia Corrêa Bonifácio[1], Mariza Ferro[1] e Fábio Machado Porto[1]

[1] *LNCC, Petrópolis/RJ, Brazil*

Resumo

O monitoramento do esgoto é uma forma de medição indireta da saúde da população. Neste trabalho o método de Árvore de Regressão foi empregado com intuito de prever o número de casos confirmados da doença através de medições de fragmentos de cópias de genomas (CG) no esgoto. As análises virais detectadas nas fezes humanas estão presentes antes mesmo dos sintomas físicos aparecerem e também permanecem de 22 a 26 dias após desaparecerem os sintomas ou *swab* de faringe negativados. A compreensão da relação entre a CG no esgoto e o número de contaminados pode trazer informação prévia sobre a disseminação de SARS-CoV-2. Esta é uma forma de medição e avaliação que capacita autoridades na tomada de decisões em relação a investimentos específicos voltados a melhorias ou isolamento de áreas de alto risco.

Palavras Chaves:SARS-CoV-2, Esgoto, Árvore de Regressão, Aprendizado de Máquina

1 INTRODUÇÃO

Desde o final de Fevereiro de 2020 o Brasil vem enfrentando o desafio de contornar o espalhamento de uma Síndrome Respiratória Aguda Grave conhecida como COVID-19 causada pelo SARS-CoV-2 [6]. Essa cepa de coronavírus se alastrou por todo o território mostrando capacidade de evolução rápida para pneumonia grave e levando a óbitos. Devido as poucas informações sobre a comportamento do vírus, medidas preventivas, tratamento, além das condições de saúde e saneamento do Brasil as notificações da doença não refletiam a realidade.

Contato: Carmen Bonifácio, bonicarm@lncc.br

O primeiro tipo de coronavirus detectado em humanos SARS-CoV, foi conhecido na China em 2002. Desde então novas cepas de vírus são identificadas. Em 2012 o MERS-CoV no Oriente Médio apresentou alta letalidade. Em Dezembro de 2019 o SARS-CoV-2 surgiu na cidade de Wuhan na China apresentando um maior poder de contágio porém com menor letalidade [8].

Estudos indicam que uma porcentagem significativa de infectados, cerca de 40%, não desenvolvem sintomas [3], uma condição que contribui para propagação silenciosa da doença. A principal rota de transmissão conhecida até o momento é através de membranas mucosas contendo gotículas respiratórias de pessoas contaminadas. Embora a dinâmica viral no trato intestinal e urinário não tenha sido elucidada a presença de vírus foi confirmada para fezes e urina de pacientes sintomáticos e assintomáticos [2]. Pesquisas sobre derramamento viral fecal indicam que cerca de 53,9% dos pacientes testaram positivo para RNA fecal [4].

A detecção de fragmentos de cópias genômicas (CG) em esgoto é uma forma de monitorar indiretamente a saúde da população. Esse estudo de epidemiologia chamado WBE *(wastewater-based epidemiology)* já vem sendo praticado em muitos países para auxílio a tomada de decisões [1].

A excreção do vírus nas fezes pode ser identificada até uma semana antes dos sintomas físicos aparecerem e de 22 a 26 dias após desaparecerem os sintomas ou *swab* de faringe negativados [12], [5], deste modo, sendo possível rastrear e quantificar pode auxiliar autoridades para investimentos direcionados antecipadamente.

Este trabalho está organizado da seguinte forma: a Seção 2 apresenta os conceitos necessários para compreensão do tema abordado e também são citados os trabalhos relacionados. A Seção 3 apresenta os resultados e discussões, encerrando na Seção 4 com as conclusões. Assim este trabalho tem por objetivo identificar a eficácia do Método de Árvore de Regressão e obter maiores informações sobre o conjunto de dados.

2 CONCEITUAÇÃO

O Aprendizado de Máquina (AM) é uma sub-área da Inteligência Artificial (IA). O AM pode ser dividido em supervisionado, não supervisionado ou por reforço. No caso supervisionado as tarefas podem ser divididas em classificação quando o conjunto de dados recebe rótulos nominais ou regressão quando a rotulagem é numérica [10]. O algoritmo de Árvore de Decisão (AD) é um dos mais utilizados por apresentar de forma transparente as regras e conceitos aplicados na obtenção dos resultados. É um método de fácil compreensão que pode fornecer maiores informações sobre o conjunto de dados.

O algoritmo de AD representa uma função que através de um processo indutivo toma como entrada vetores com valores de atributos e retorna uma saída única (atributo alvo), no caso da regressão um valor médio que compõe os ramos da árvore. A decisão é alcançada executando uma sequência de testes. Cada nó interno na árvore corresponde a um teste do valor de um dos atributos de entrada e as ramificações dos nós são classificadas com os valores possíveis do atributo. Os nós de folha na árvore especificam os valores retornados pela função como resultados da decisão. Com isso, cada ramificação da árvore representa uma conjunção de testes necessários para se obter uma resposta final. Quanto maior o caminho, mais complexas são as regras de decisão e menor a generalização (necessária para atingir modelos abrangentes) [10]. A divisão dos nós é feita

através de métricas de distribuição sendo que a utilizada neste trabalho foi o erro médio quadrático (MSE). Além disso, os modelos gerados foram avaliados segundo o coeficiente de determinação (r2_score), MSE e raiz quadrada do erro médio quadrático (RMSE).

Para a construção dos modelos foi utilizado Jupyter Notebook V 6.0.2, linguagem Python em ambiente Anaconda V1.9.2, biblioteca scikit learn com modelo CART para Árvore de Decisão [7].

A realização da regressão tem por objetivo encontrar a melhor partição dos dados que leve a uma aproximação do número de pessoas contaminadas pelo vírus SARS-CoV-2 utilizando como entrada os resultados analíticos de fragmentos de RNA viral detectados em amostras de esgoto e variáveis relacionadas.

Os dados relativos a carga viral foram captados no site de Gestão da Informação da prefeitura de Niterói [1]. As datas do monitoramento compreendem do dia 12 de Abril até 15 de Agosto de 2020 (Semana Epidemiológica 16 a 33). As análises de detecção foram realizadas pelo laboratório de virologia da FIOCRUZ [9]. O projeto tem parceria com a prefeitura de Niterói e a companhia Águas de Niterói responsável pelo abastecimento de água e saneamento da cidade. Os pontos exatos das coletas não foram especificados, portanto, foram estimados na entrada das Estações de Tratamento de Esgoto.

A cidade de Niterói é dividida geograficamente em 5 regiões: Oceânica, Praias, Norte, Pendotiba e Leste.

Para a atributo alvo foi utilizado o número de casos passíveis de detecção (CD), ou seja, os casos comprovados da doença que podem contribuir com RNA viral no esgoto na semana avaliada. Esta hipótese foi construída levando em consideração que no Brasil, nos primeiros meses da pandemia somente os casos graves e críticos eram reportados e confirmados. Estes casos são os que excretam cargas virais com maior intensidade. A métrica foi obtida através de relações matemáticas utilizando casos totais (CT) e casos novos (CN) relatados por dia por bairros. A Equação 1, apresenta o cálculo aplicado para construção deste atributo.

$$CD_{[semana_x]} = CN[semana_x] + CT_{[semana_{(x-1)}]} - CN[semana_{(x-3)}] \qquad (1)$$

O primeiro caso registrado da doença na cidade de Niterói ocorreu no dia 09 de Março. O dia 12 de Abril, data de início das amostragens, compreende aproximadamente 33 dias após a comprovação do vírus na cidade. Esta situação reflete os resultados do conjunto de dados onde poucas são as amostras que apresentaram resultados de CG inferiores ao limite de detecção do método de 10 CG por mililitro de amostra de esgoto (10 CG/ml). Isso não significa que não havia presença do vírus circulando na região nestas datas.

O conjunto de dados é composto por 83 instâncias e originalmente concebido contendo 6 atributos sendo 5 numéricos e um categórico. A Tabela 1 identifica os atributos, respectivos tipos, int = inteiro (dado numérico) e str = string (dado categórico) e descreve sua representação.

[1] http://sigeo.niteroi.rj.gov.br/

Tabela 1: Descrição do Conjunto de Dados

Atributo	Tipo	Descrição
Semana Epidemiológica	int	linha temporal iniciada na semana 16
Região	str	região geográfica onde a amostra foi coletada
CG RNA/Litro	int	resultado analítico de cópias Genômicas no esgoto
Vazão Litro/hora	int	vazão na Estação de Tratamento de Efluentes
População	int	nº pessoas de abrangência da Estação de Tratamento.
Casos Detecção (CD)	int	nº de casos relatados passíveis de detecção no esgoto.

2.1 Trabalhos Relacionados

A Austrália propôs uma estimativa analítica para o cálculo do número de pessoas contaminadas utilizando os resultados de carga viral no esgoto, além de valores mínimos e máximos da carga viral excretado em fezes humanas, para estimar uma faixa provável de casos [1]. Um estudo Espanhol estimou o número de contaminados em uma região através de regressão linear e usou esta estimativa para desenvolver modelo estatístico aliado aos resultados de carga viral de uma estação de tratamento de efluentes para prever o número de casos ativos usando regressão linear e quadrática [11]. O aprendizado proposto pela Árvore de Regressão é baseado nos dados de casos comprovados da doença, este trabalho inicial visa capturar as tendências de correlações sobre o conjunto de dados para posteriormente avançar em uma pesquisa mais detalhada na tentativa de incluir a estimação do número de casos assintomáticos que usualmente não são reportados.

3 RESULTADOS E DISCUSSÃO

Nesta sessão serão analisadas as características estatísticas sobre o conjunto de dados e apresentados os resultados obtidos com a aplicação dos modelos.

3.1 Exploração dos Dados

A variável meta CD não apresenta distribuição normal, segundo teste de normalidade de Kolmogorov-Smirnov com 99,9% de significância estatística. Segundo a Figura 1, o histograma com as frequências e a curva de densidade indicam que os dados parecem seguir uma distribuição lognormal com média igual a 438 e desvio padrão de 453. Este desvio padrão alto, já indica uma forte dispersão dos dados.

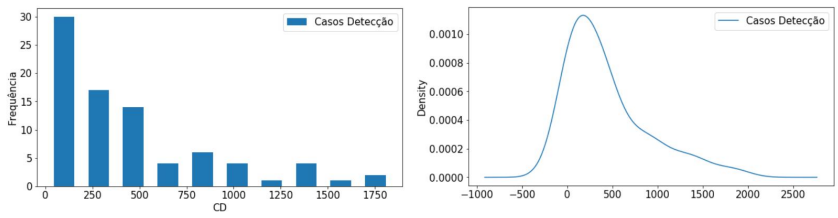


Fig. 1: Distribuição dos dados CD

Os boxplots na Figura 2 dispostos por semana no gráfico à esquerda, identificam

presença de *outliers* (valores discrepantes) na semana 18 e das semanas 29 a 32. Observa-se uma tendência de crescimento nas caixas da semana 22 até a 25 e neste grupo, como não se verifica *outlier*, significa que há mais de um elemento nestes conjuntos com valores elevados que contribuem para elevar os valores do terceiro quartil. O maior valor observado pertence a Semana 25. Já na disposição por região, gráfico à direita, não se verifica *outlier*. Os maiores valores são referentes a Região das Praias, a mais populosa e com maior infra estrutura social.

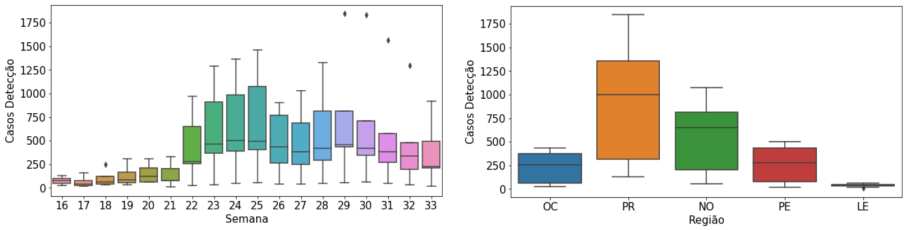


Fig. 2: Distribuição dos dados CD por semana e por região

Analisando os dados de CG no esgoto (Figura 3) verifica-se diversos pontos considerados *outliers*. Esses *outliers* podem ser vistos pelo ponto de vista temporal, no gráfico expresso por semana epidemiológica ou espacial, quando expressos por região. No gráfico semanal destaca-se a Semana 23 identificando uma alta carga antecipando com 7 e 14 dias o que foi visto para o atributo CD onde o pico de maior contribuição foi na Semana 25. Nesta caixa a inclusão dos *outliers* provoca o alongamento do terceiro quartil elevando, principalmente, os resultados de média e provocando assimetria no conjunto. Entretanto, os *outliers* parecem fornecer informações relevantes sobre o comportamento da doença naquele período da amostragem e por isso, não foram removidos da base de dados. É esperado que a árvore possa fornecer uma boa generalização com a presença destes elementos.

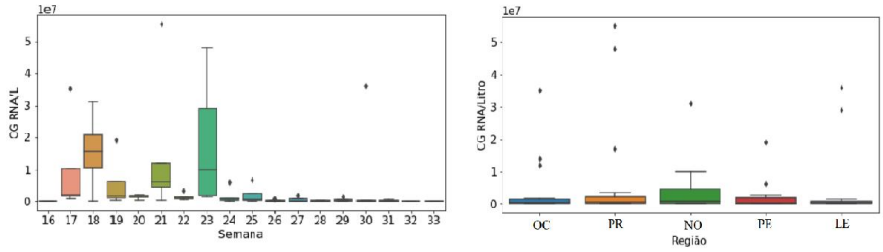


Fig. 3: Distribuição da Carga Genômica RNA por semana e por região

A Figura 4 apresenta a distribuição dos dados ao longo das semanas epidemiológicas. A esquerda o gráfico de CD e à direita CG RNA. Nas semanas epidemiológicas de início do monitoramento onde o número de casos comprovados CD é crescente, porém baixo, é possível constatar que o esgoto já apresentava picos para diversas regiões revelando que o vírus já estava em circulação. Mais indícios da precocidade da avaliação do esgoto são obtidos verificando no gráfico da direita um pico de CG RNA na semana 21 para a maioria

das regiões, porém, mais visível para região das Praias que só foi refletir nos casos CD a partir da Semana 22. Ou seja, os casos confirmados são diagnosticados com pelo menos uma semana de atraso em relação ao esgoto.

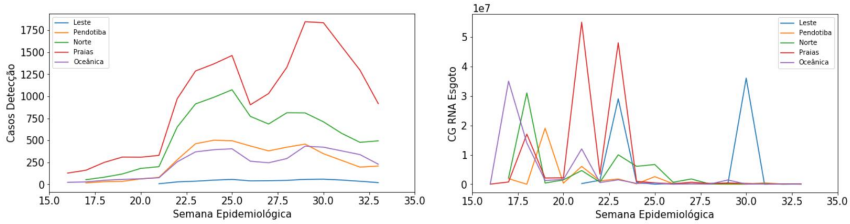


Fig. 4: Casos Detecção e CG RNA Esgoto

As variações das curvas também seguem a mesma estrutura. Há um primeiro acréscimo de registros de CD na Semana 22 (24-30/Maio) que perdura até a Semana 25 (14-20/Jun). Embora não perceptível, o mesmo efeito é visto na região Leste. A segunda onda, aproximadamente nas semanas 28 a 30 (05-24/Jul), é bem mais acentuada para região das Praias, inclusive, nesta única região a magnitude se mostra maior que na primeira onda. A primeira onda é compatível com os picos do esgoto e seria natural pensar em observar picos semelhantes referentes a segunda onda. No entanto, isso não é perceptível. Para o gráfico de CG RNA à direita, referente as semanas 28 a 30, não se verifica um acréscimo com tal magnitude, principalmente para a região das Praias. Uma hipótese para o ocorrido seria que o acréscimo do número de registros se deu pela imputação de casos confirmados, possivelmente oriundos de campanhas de testes em massa. Neste sentido duas vertentes podem ser seguidas: 1) Os casos foram detectados de forma postergada e foram lançados como casos novos. Essa condição seria não desejada porque imputa em erro na relação do aprendizado. Neste caso os resultados do esgoto não seriam condizentes com os registros CD. 2) Os casos detectados são realmente ativos. Nessa situação tem-se uma imputação em CD referente aos casos leves ou assintomáticos que não seriam confirmados por outro modo. Esta pode ser uma informação de grande relevância, visto que, estudos até o momento indicam uma menor excreção de CG RNA nas fezes para estes pacientes confirmando os resultados obtidos através do esgoto. A Região que efetivamente apresentou um crescimento de carga viral no esgoto na Semana 30 foi a região Leste. Neste caso, isso também se refletiu no número CD na mesma semana. Fenômeno este atípico pelo que se viu anteriormente, já que usualmente os casos CD se refletem posteriormente aos resultados do esgoto.

É fato que existem variantes associadas aos resultados analíticos de CG RNA, contudo, esta avaliação ressalta que o comportamento dos fragmentos de SARS-CoV-2 detectados no esgoto permaneceram estáveis e refletiram as mesmas características de acréscimo e decaimento dos casos comprovados da doença, principalmente vinculados aos primeiros meses de avaliação.

3.2 Construção dos Modelos

Após realização de vários testes, a modelagem ficou estabelecida com partição da base de dados em 60% das amostras usadas para o treinamento e 40% para a validação, utilizando

a função *split* com iniciação randômica da semente em 10. Os atributos, CG RNA, Vazão e População foram os dados de entrada . O Modelo 0, foi utilizado como base para quantificação das métricas de avaliação. A função regressora não recebeu nenhum ajuste de parâmetros e por consequência o Modelo 0 obteve altura de 13 e 49 nós folhas com alta característica de sobre ajuste (*overfitting*). Para o Modelo 1, contendo os mesmos atributos de entrada, o crescimento da árvore foi limitado em 7 sub-ramos. Com intuito de evitar *overfitting* foi determinado o valor máximo de 6 amostras para efetivamente fazer a divisão do nó e estipulado número mínimo de amostras no nó folha em 3. Estes parâmetros auxiliaram na obtenção de uma melhor generalização da árvore. As métricas são descritas na Tabela 2.

Com base na exploração dos dados, a Semana Epidemiológica traz informações temporais que possuem uma certa correlação com o número de casos. Isso é o determina a curva epidêmica da doença que costuma ter início, meio e fim. Com intuito de testar o grau de impacto de inclusão da Semana Epidemiológica aos dados de entrada este atributo foi incorporado ao modelo, chamado de Modelo 2, mantendo os mesmos parâmetros estipulados para o Modelo 1. A métrica R2_score avaliada isoladamente não comprova relação estatística de significância entre os atributos de entrada e a saída estimada, para auxiliar também foi apresentado o MSE e RMSE.

Tabela 2: Métricas da Árvore de Regressão

Modelo	R2_score	MSE	RMSE
Modelo 0	0.59	103036	321
Modelo 1	0.71	71984	268
Modelo 2	0.84	39818	199

Considerando a esparcialidade dos dados e fazendo uso dos parâmetros da função regressora, o Modelo 1 obteve um ganho no coeficiente de determinação de 12% e uma redução de erro da ordem de 29,9%. Já o Modelo 2 apresentou um ganho ainda maior. Para o coeficiente de determinação um acréscimo de 25% e redução do erro da ordem de 61,3%. Avaliando e comparando os modelos, na Figura 5 é apresentada a curva de predição para o Modelo 1 e o Modelo 2. O Modelo 1 generaliza o caminho encontrando um traçado intermediário entre os pontos. Por exemplo, analisando o intervalo do eixo Dados CD [18-22]. Para o Modelo 2, constata-se que o traçado atinge exatamente os pontos o que pode levar a um sobre ajuste do modelo.

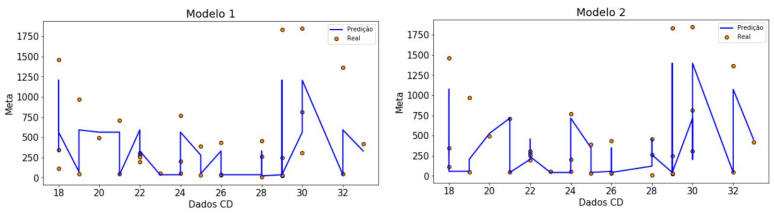


Fig. 5: Predição dos Modelos

É importante analisar o equilíbrio entre viés e variância. Para isso foi investigado o

grau de significância dos atributos. A Figura 6 demostra que o atributo População no Modelo 1 apresenta baixa significância contribuindo apenas com 3,9% de participação para construção da árvore. Realmente quando este atributo é retirado da base de dados não afeta em nada os resultados. O coeficiente de determinação é mantido em 71%. Na verdade a inclusão do atributo População utilizado juntamente com o atributo Vazão é o que não agrega valor para o modelo, pois se for utilizado somente População junto a CG RNA também se verifica a manutenção do resultado para coeficiente de determinação, MSE e RMSE. Ou seja, ambos os atributos Vazão e População fornecem ao modelo as mesmas características para serem utilizadas na distribuição dos ramos.

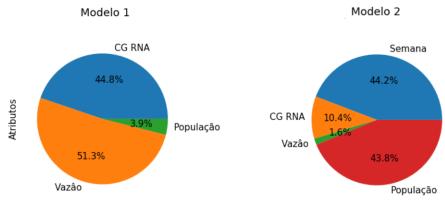


Fig. 6: Grau de significância dos atributos de entrada

Já para o Modelo 2 percebe-se que o algoritmo alterou a configuração dos sub-ramos usando primordialmente como base estrutural da divisão dos nós a Semana Epidemiológica. Isso implicou em uma distribuição do grau de importância dos atributos muito distinta do Modelo 1, considerando menor significância para a variável CG RNA, a qual representa o maior interesse de estudo. Com base nesta avaliação e no gráfico da curva de predição é possível dizer que a inclusão do atributo Semana, embora tenha reduzido a variância, acrescentou um viés não desejado ao modelo.

Usando *k-fold Cross Validation* (K=5) a Figura 7 à esquerda, apresenta as curvas de acuracidade para fase de treinamento e validação do Modelo 1. Como é possível avaliar, com número de amostras próximo a 50, a curva de treinamento atinge até 75% de acurácia enquanto a validação apresenta 26%. O perfil de crescimento de ambas as curvas e a grande separação existente entre elas ao final do processo indicam que aumentar o conjunto de dados poderia trazer benefícios para a acuracidade do modelo. O sombreado em *background* indica a dispersão dos dados, portanto, representa a variância do modelo. Quanto maior a largura maior a variância. No conjunto com 2 amostras de treinamento a variância na validação foi mais alta. Além disso a variância é bastante afetada pela presença dos *outliers*. No entanto, verifica-se que com o aumento do número de elementos no conjunto de treinamento ocorre a redução da variância.

Para a curva do MSE, Figura 7 à direita, verifica-se que o erro de treinamento tem um acréscimo conforme o aumento do número de elementos no conjunto de treinamento. Já para a validação o erro inicial é alto e vai decaindo com o aumento do número de elementos no conjunto de treinamento, mas o *gap* ainda é grande entre as curvas.

A Árvore gerada pelo Modelo 1 estabeleceu o atributo Vazão como nó raiz com o menor MSE. A configuração estabeleceu o ramo direito contendo apenas amostras referentes a região das Praias. Esta característica tornou a árvore assimétrica com crescimento maior no ramo esquerdo. A segunda separação dos nós utiliza o atributo CG RNA com *th-*

resholds distintos para cada sub-ramo. A configuração obteve 12 nós folhas onde somente 2 continham apenas uma amostra, porém, ainda é observado um aprendizado específico para os sub-ramos da esquerda. A tentativa de redução da altura da árvore implica em redução no *score* e aumento do MSE, de modo que maiores estudos serão realizados para melhorias neste modelo.

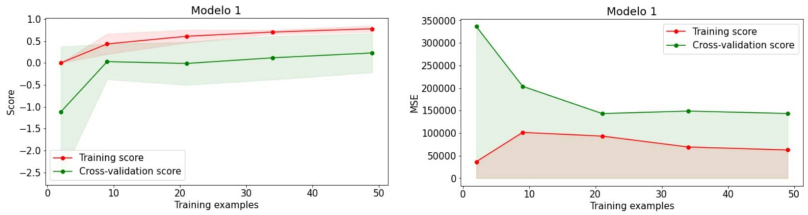


Fig. 7: Curva de Aprendizado Modelo 1

Na Figura 8 pode-se verificar o gráfico de barras do conjunto de amostras de validação em comparação com os valores preditos pelo Modelo 1. São apresentados duas regiões, a mais populosa, Região das Praias e a menos populosa Região Leste. Os valores preditos são diferentes para cada uma das regiões, pois percorrem diferentes caminhos até os nós folhas. A região Leste apresentou o maior número de predições mais aproximadas do valor real: seis amostras do total de nove com erro absoluto abaixo da média dos erros da região de 237.

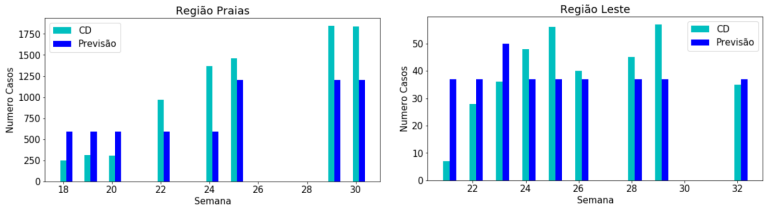


Fig. 8: Comparação dos valores reais e preditos

Para as duas comparações, se houvessem mais amostras no conjunto de dados, a variância seria reduzida, com isso as médias definidas por cada nó folha ficariam mais próximas da realidade. A remoção de três dos maiores *outliers* da base de dados não implicou em aumento do coeficiente de determinação. Talvez uma análise específica para cada região ajudasse na melhor quantificação dos casos.

4 CONCLUSÕES

Este estudo inicial é muito vinculado as características das regiões da cidade de Niterói. Contudo, foi possível identificar comportamento muito similar entre a quantidade de fragmentos virais no esgoto e do número de casos confirmados. Quando o número de fragmentos cresce a quantidade de casos também cresce. Além disso, a precocidade com que esta relação se manifesta confere ao estudo do esgoto uma importância única do ponto de vista temporal para tomada de decisões. Através da identificação de regiões com alta

incidência é possível alocar recursos para confirmação dos casos naquele local e proceder com medidas específicas para prevenção da disseminação.

A Árvore de Regressão forneceu resultados aceitáveis para o Modelo 1 que podem ser melhorados na medida em que mais dados forem adicionados ao conjunto. Além disso, novos atributos podem ser incorporados a entrada do modelo visando aproximar ainda mais as previsões e até mesmo possibilitar a detecção dos casos assintomáticos que na maioria das vezes não são reportados. O Modelo 2 provou não ser adequado. A Semana Epidemiológica é um atributo temporal fortemente vinculado a disseminação da doença. No entanto, provocou distorção na generalização introduzindo um viés não desejado ao modelo.

Identificar a presença e a disseminação do risco biológico ainda é fator de estudo, principalmente em localidades com poucos recursos. A utilização de métodos de Aprendizado de Máquina para extração de informações pode contribuir como ferramenta auxiliar na gestão da pandemia. Maiores informações sobre as amostragens são esperadas para que seja possível um estudo mais amplo. O método de Árvore de Regressão ajudou na compreensão do comportamento dos atributos contudo a aplicação de um método com maior complexidade como rede neural será avaliado em trabalhos futuros como alternativa para alcançar maior acuracidade.

5 *AGRADECIMENTOS*

Agradeço a equipe ***DexlLab***, a ***FIOCRUZ*** **e a** ***CAPES***.

REFERÊNCIAS

[1] W. Ahmed, N. Angel, J. Edson, K. Bibby, A. Bivins, J. O'Brien, P. Choi, M. Kitajima, S. Simpson, J. Li, B. Tscharke, R. Verhagen, W. Smithg, J. Zaugg, L. Dierens, P. Hugenholtz, K. Thomas, and J. Mueller. First confirmed detection of SARS-CoV-2 in untreated wastewater in Australia: A proof of concept for the wastewater surveillance of COVID19 in the communityh. *Science of the Total Environment*, 728(138764):doi=10.1016/j.scitotenv.2020.138764, 2020.

[2] Y. Chen, L. Chen, Q. Deng, G. Zhang, K. Wu, L. Ni, Y. Yang, B. Liu, W. Wang, C. Wei, J. Y. G. Ye, and Z. Cheng. The presence of SARS-CoV-2 RNA in the feces of COVID-19 patients. *Medical Virology*, 92:833–840. doi=10.1002/jmv.25825, 2020.

[3] M. Day. Covid-19: four fifths of cases are asymptomatic, china figures indicate. *BMJ (Clinical research ed.)*, 369:doi=10.1136/bmj.m1375, 2020.

[4] S. Gupta, J. Parker, S. Smits, J. Underwood, and S. Dolwani. Persistent viral shedding of sars-cov-2 in faeces - a rapid review. *Colorectal disease : The Official Journal of the Association of Coloproctology of Great Britain and Ireland*, 22:611–620. doi=10.1111/codi.15138, 2020.

[5] X. Huang, Y. Wu, C. Guo, L. Tang, Z. Hong, J. Zhou, X. Dong, H. Yin, Q. Xiao, Y. Tang, X. Qu, L. Kuang, X. Fang, N. Mishra, J. Lu, H. Shan, and G. Jiang. Prolonged presence of SARS-CoV-2 viral RNA in faecal samples. *Gastroheap*, 5:434–435. doi=10.1016/S2468–1253(20)30083–2, 2020.

[6] L. Michelin, R. S. Lins, and A. Falavigna. COVID-19: perguntas e respostas. *Centro de Telemedicina da UCS* , 2020.

[7] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[8] N. Petrosillo, G. Viceconte, O. Ergonul, G. Ippolito, and E. Petersen. COVID-19, SARS and MERS: are they closely related? *Clinical Microbiology and Infection: Elsevier*, 26:729–734, doi=/10.1016/j.cmi.2020.03.026, 2020.

[9] T. Prado, T. Fumian, C. Mannarino, A. Maranhão, M. Siqueira, and M. Miagostovich. Preliminary results of SARS-Cov-2 detection in sewerage system in Niterói municipality , RIO DE JANEIRO, BRAZIL. *Mem Inst Oswaldo Cruz*, doi=10.1590/0074-02760200196, 2020.

[10] S. Russell and P. Norvig. Inteligência Artificial. *tradução Regina Célia Simille. – Rio de Janeiro: Elsevier*, pages 809–832, 2013.

[11] J. Vallejo, S. Rumbo-Feal, L.-O. K. Conde-Pérez, J. Tarrío, R. Reif, S. Ladra, B. Rodiño-Janeiro, M. Nasser, Cid, M. Veiga, A. Acevedo, C. Lamora, G. Bou, R. Cao, and M. Poza. Highly predictive regression model of active cases of COVID-19 in a population by screening wastewater viral load. doi=10.1101/2020.07.02.20144865.t, 2020.

[12] Y. Zhang, C. Chen, S. Zhu, C. Shu, D. Wang, J. Song, S. Y., W. Zhen, Z. Feng, G. Wu, J. Xu, and W. Xu. Isolation of 2019-nCoV from a Stool Specimen of a Laboratory Confirmed Case of the Coronavirus Disease 2019 (COVID-19). *Chinese Center for Disease Control and Prevention*, 2:123–124, 2020.

Desempenho de Métodos de Tratamento de Superfície Livre em Propagação de Ondas

Carolina Maria Nunes Bezerra[1], Raquel Jahara Lobosco[2], José Antônio Fontes Santiago[1] e Edmundo Guimarães de Araújo Costa[1]

[1] *Programa de pós-graduação em Engenharia Civil, UFRJ, Rio de Janeiro/RJ, Brasil*
[2] *Departamento de Engenharia Mecânica, UFRJ, Macaé/RJ, Brasil*

Abstract

Um tanque de ondas numérico (numerical wave tank - NWT) é uma ferramenta útil para analisar problemas de engenharia costeira e offshore. Uma variedade de metodologias numéricas, aplicadas à Fluidodinâmica Computacional (CFD), são fornecidas na literatura para a geração e absorção de ondas de superfície livre dentro de um NWT. Entretanto, quando o usuário configura esse problema em CFD, é necessário selecionar o modelo numérico mais apropriado para representar o fenômeno físico do movimento da onda. O presente artigo fornece uma avaliação quantitativa da propagação de uma onda em um modelo bi-dimensional, considerando uma avaliação dos diferentes métodos de captura de interface no software livre OpenFOAM. Este trabalho mostra que a abordagem dos métodos de geração e absorção de ondas podem causar diferenças significativas na representação da interface entre dois fluidos e, consequentemente, na definição da altura da superfície livre do escoamento e no campo de velocidade.

Keywords: CFD, captura da interface, ondas, escoamento multifásico, OpenFOAM

1 INTRODUÇÃO

No campo da ciência e da engenharia, escoamentos multifásicos instáveis ou periódicos com superfície livre estão presentes em muitas das aplicações práticas e industriais. A correta representação da interface entre dois fluidos permite avaliar o movimento de ondas oceânicas, caracterizar a expansão de jatos externos e através da transferência de massa na região interfacial, é possível também quantificar a ruptura e coalescência de gotas [2].

No dimensionamento de estruturas oceânicas, seja nos projetos portuários ou de encostas, é necessário conhecer, definir e representar as características das ondas: geração, propagação, e transformação ao longo do movimento. Nesse contexto, o uso de modelos numéricos tem uma grande contribuição para os estudos hidrodinâmicos. A dinâmica

Contato: Carolina Bezerra, carolmnb@coc.ufrj.br

dos fluidos computacional, associada aos estudos do tanque de ondas numérico (NWT), contribui amplamente para as aplicações de representação do modelo físico no campo dos estudos da engenharia marinha e costeira [3].

2 MODELAGEM MATEMÁTICA

Essa pesquisa científica avalia os métodos numéricos de representação de onda em um meio com viscosidade, para um canal bi-dimensional com uso do software livre OpenFOAM. Foram utilizados os solvers interFoam, interIsoFoam e olaFlow para investigação da superfície livre, do campo de velocidade e da absorção da onda. Os resultados são apresentados em função da solução numérica do solver interFoam que utiliza o método padrão de Volume de Fluido (VOF) para captura da interface [7].

2.1 Equações Governantes

As Equações de conservação da massa e da quantidade de movimento são utilizadas para a representação numérica do escoamento com superfície livre e propagação de ondas em um canal. Foi utilizada a abordagem Euleriana para solução das Equações de Navier-Stokes, e o escoamento é considerado incompressível e laminar, conforme demonstram as Equações 1 e 2.

$$\frac{\partial U_i}{\partial x_i} = 0 \tag{1}$$

$$\frac{\partial(\rho U_i)}{\partial t} + \rho U_j \frac{\partial U_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \rho g_i + \frac{\partial[\mu(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i})]}{\partial x_j} \tag{2}$$

Em que U é a velocidade, ρ é a densidade, g é a aceleração da gravidade, p é a pressão e μ é a viscosidade dinâmica.

2.2 Método de Volume de Fluido (VOF)

O método VOF, proposto originalmente por Hirt e Nichols [5], é um método de rastreamento em que uma função é utilizada para marcar a localização de dois ou mais fluidos imiscíveis. O método resolve um único conjunto de equações de conservação de massa e quantidade de movimento linear para ambas as fases. Uma variável auxiliar, denominada fração volumétrica α, é utilizada para identificar a região de cálculo para cada uma das fases. No presente trabalho, como os fluidos considerados são a água e o ar, para $\alpha = 1$, o volume é inteiramente ocupado pela água e quando $\alpha = 0$ o volume correspondente é ocupado pelo ar. A interface entre as fases ocorre nas células em que $0 < \alpha < 1$ e, dessa forma, as propriedades e campos de variáveis do escoamento, representadas por Φ conforme mostra a Equação 3, são comuns à ambas as fases. No presente trabalho, Φ representa o cálculo para ρ, μ e U.

$$\Phi = \alpha(x)\Phi(x,t)_{agua} + (1 - \alpha(x))\Phi(x,t)_{ar} \tag{3}$$

A Equação governante para o transporte da fração volumétrica α em cada volume de controle, no método VOF, é representada pela Equação da advecção-difusão conforme a Equação 4.

$$\frac{\partial \alpha}{\partial t} + \nabla . U(x,t)\alpha(x) = 0 \tag{4}$$

Portanto, para o cálculo das variáveis na região da interface, são resolvidas as equações 1, 2 e 4.

2.3 Representação da Superfície Livre

Os solvers interFoam e olaFlow utilizam o método denominado MULES (*Multidimensional Universal Limiter for Explicit Solution*) para manter os limites de fração volumétrica independente do esquema numérico e da estrutura da malha. Por sua vez, o solver interIsoFoam utiliza o método isoAdvector, o qual baseia-se na construção de isosuperfícies para rastreamento da interface. Nas seções a seguir serão apresentados como funciona cada método.

2.3.1 Método MULES

O método MULES é um método numérico em que o termo da advecção da Equação 4 é modificado para comprimir a interface [11]. Reescrevendo a Equação 4 na forma integral temos:

$$\int_{\Omega_i} \frac{\partial \alpha}{\partial t} dV + \int_{\partial \Omega_i} \alpha U.ndS = 0 \tag{5}$$

Ao utilizar um esquema de discretização temporal para o primeiro termo da Equação 5 e aplicando uma soma no volume de todas as faces do elemento, no segundo termo, é possível escrever a Equação 6.

$$\frac{\alpha_i^{n+1} - \alpha_i^n}{\Delta t} = -\frac{1}{|\Omega_i|} \sum_{f \in \partial \Omega_i} (F_u + \lambda_M F_c)^n \tag{6}$$

em que Ω_i, $\partial\Omega_i$ e λ_M são respectivamente o volume, a face do volume, e um delimitador que indica superfície quando o valor é 1 e 0 quando a região é fora da interface. Os termos F_u e F_c são os fluxos advectivos, descritos conforme as Equações 7 e 8.

$$F_u = \Phi_f \alpha_{f,upwind} \tag{7}$$

$$F_c = \Phi_f \alpha_f + \Phi_{rf} \alpha_{rf} (1-\alpha)_{rf} - F_u \tag{8}$$

O subscrito f indica a quantidade avaliada na face e *upwind* é o esquema numérico utilizado. O fluxo advectivo na face é dado por $\Phi_f \alpha_f$ e $\Phi_{rf}\alpha_{rf}(1-\alpha)_{rf}$ e representa o fluxo compressivo, aonde Φ_{rf} é α_{rf} são fornecidos pelas expressões das Equações 9 e 10.

$$\Phi_{rf} = min\left(C_\alpha \frac{|\Phi_f|}{|S_f|}, max\left[\frac{|\Phi_f|}{|S_f|}\right]\right)(n_f.S_f) \tag{9}$$

$$\alpha_{rf} = \alpha_P + \frac{\alpha_N - \alpha_P}{2}[1 - \chi(\Phi_f)(1-\lambda_{rf})] \tag{10}$$

Em que C_α, S_f e n_f são, respectivamente, um parâmetro de redução da mancha na interface, o vetor área da face e o vetor normal da interface centrada na face. Os termos

N e P denotam os vizinhos à montante e à jusante, enquanto χ é uma função marcadora que indica valor 1 para o fluxo positivo na face e -1 para o fluxo negativo. Na Equação 6, a expressão no somatório representa a combinação do esquema de alta ordem para a advecção e um termo de fluxo compressivo, que permite uma maior precisão e diminuição da difusão numérica na interface, assim como o parâmetro do aspecto de mancha [9].

2.3.2 Método isoAdvector

O método isoAdvector é um método geométrico VOF para advecção de uma interface mais nítida para dois fluidos incompressíveis. Este método pode ser utilizado para malhas estruturadas e não-estruturadas, sem requisitos para o formato dos volumes. A teoria que descreve o isoAdvector pode ser encontrada no trabalho desenvolvido por Johan Roenby [6].

A fração da fase é calculada no tempo t a partir de uma função $H(x,t)$, conforme mostra a Equação 11.

$$\alpha_i(t) = \frac{1}{V_i}\int_{\Omega_i} H(x,t)dV \tag{11}$$

A fração de fase α_i pode ser calculada no intervalo de tempo seguinte, através da Equação 12, porque α_i é conhecida em cada volume no tempo t.

$$\alpha_i(t+\Delta t) = \alpha_i(t) - \frac{1}{V_i}\sum_{j\in B_i} s_{ij}\Delta V_j(t,\Delta t) \tag{12}$$

$$\Delta V_j(t,\Delta t) = \int_t^{t+\Delta t}\int_{F_j} H(x,\tau)U(x,\tau)dSd\tau \tag{13}$$

Em que, a Equação 13 representa o volume total do fluido A transportado através da face j durante um intervalo de tempo. Os termos V_i, Ω_i, B_i, F_j, s_{ij} e τ são, respectivamente, o volume da célula i, o volume de cada célula i, lista de todas as faces, face j que pertence à célula i, termo que orienta o fluxo para fora do volume e a variável de integração usada em cada intervalo de tempo. Assim, fica definida a quantidade que é estimada no método isoAdvector, por meio da fração da fase α_i, velocidade U_i e fluxo Φ_j que atravessa a face j no tempo t, Equação 14.

$$\Phi_j(t) = \int_{F_j} U(x,t)dS \tag{14}$$

2.3.3 interFoam, olaFlow e interIsoFoam

Apesar do olaFlow ter origem no solver interFoam, compartilhando o método MULES para construção da interface, sua principal diferença está na sua utilidade. Enquanto o interFoam é um solver que resolve escoamentos entre dois fluidos incompressíveis, o olaFlow é específico apenas para solução de problemas que envolvem dinâmica de ondas em um tanque numérico com ou sem a presença de modelos para quebra-mar. Por sua vez, o interIsoFoam é uma modificação do VOF que utiliza o método isoAdvector para construção da interface e também pode ser aplicado para escoamentos de dois fluidos incompressíveis.

2.4 Geração e Absorção da Onda

O software OpenFOAM possui uma biblioteca de geração de ondas lineares e não-lineares acoplada com um esquema de absorção ativo de ondas projetadas para a condição de águas rasas [10].

A geração das ondas é modelada pela função cnoidal representada pela Equação 15.

$$\eta = H\left[\frac{1}{m}\left(1 - \frac{E(m)}{K(m)}\right) - 1 + cn^2\left(2K(m)\frac{x - ct}{\lambda}\right)\right] \tag{15}$$

em que η é a elevação da superfície livre, λ é o comprimento da onda, m é o parâmetro elíptico que depende da característica da onda, $K(m)$ é o primeiro tipo da integral completa elíptica, $E(m)$ é o segundo tipo da integral completa elíptica, cn é a função elíptica de Jacobi, c é a celeridade, H é a altura da onda (diferença entre a elevação da crista e do vale) e t é o tempo.

Os esquemas de absorção podem ser utilizados para minimizar a reflexão das ondas na saída e no contorno em que as ondas são introduzidas. O método de absorção ativo de ondas para águas rasas é o mais conveniente para ser utilizado, pois a velocidade ao longo da coluna de água é considerada constante, ou seja, a velocidade horizontal da partícula de fluido é constante ao longo do eixo vertical [4]. Logo, a absorção da velocidade $u(t)$ no contorno é expressa como

$$u(t) = -\sqrt{\frac{g}{t}}\zeta(t) \tag{16}$$

sendo

$$\zeta(t) = \eta(t) - d \tag{17}$$

onde ζ é a altura da onda refletida, η é a altura da superfície livre ao longo da saída e d é altura da lâmina d'água antes da propagação da onda.

3 MODELO NUMÉRICO

No presente trabalho, apenas as fases água e ar são consideradas. A seguir, são apresentadas nas tabelas 1 e 2 as propriedades de ambos os fluidos e da onda, respectivamente. A aceleração da gravidade é de $g = 9,81 m.s^{-2}$.

Tabela 1: Propriedades do fluido

Fluido	$\rho[kg.m^{-3}]$	$\nu[m^2.s^{-1}]$
Água	1000	10^{-6}
Ar	1,2	$1{,}48 \times 10^{-5}$

Tabela 2: Propriedades da onda

Altura H(m)	0,1
Comprimento $\lambda(m)$	6,0
Período T(s)	3,0

3.1 Configuração geométricas do Canal

O canal possui comprimento, altura e nível da água de 10, 0,7 e 0,4 metros, respectivamente, conforme mostra a Figura 1. As linhas de 1 a 4 indicam as posições em que foram medidas a velocidade ao longo da coluna vertical, quando tem-se um comprimento de onda no domínio, ou seja, a crista da onda posiciona-se na linha 1. As posições são 5,0 (linha 1), 9,4 (linha 2), 9,6 (linha 3) e 9,95 (linha 4) metros.

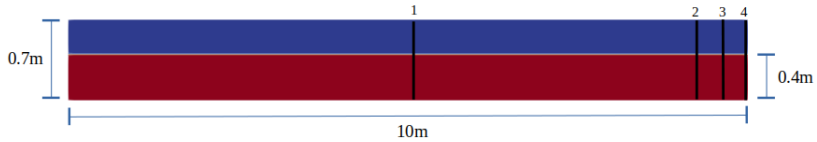


Fig. 1: Posições transversais de captura da velocidade

3.2 Condições de Contorno

Para a correta representação do comportamento físico da propagação da onda no canal, o modelo numérico necessita de condições de contorno apropriadas. Desta forma, fez-se uso de cinco condições de contorno específicas: *inlet* (entrada), *outlet* (saída), *bottom* (fundo), *atmosphere* (atmosfera) e $frontAndBack$ (laterais), conforme demonstra a Tabela 3. As laterais foram definidas com a condição vazia (*empty*) para todas as variáveis físicas do problema, o que permite uma aproximação bi-dimensional para a solução computacional [8].

Tabela 3: Condições de Contorno

Contorno	alpha	p-rgh	U
inlet	waveAlpha	fixedFluxPressure	waveVelocity
outlet	zeroGradient	fixedFluxPressure	waveVelocity
bottom	zeroGradient	fixedFluxPressure	fixedValue
atmosphere	inletOutlet	totalPressure	pressureInletOutletVelocity

As condições de contorno encontram-se descritas por sub-item conforme definição do tutorial do software OpenFOAM [1].

4 TESTE DE CONVERGÊNCIA DA MALHA

As malhas computacionais são testadas para obter a ordem formal de convergência da solução. O teste de malha foi realizado em um canal com comprimento longitudinal de 6 metros, utilizando o interFoam. A variável considerada para avaliação da malha foi o perfil de velocidade transversal no meio do domínio, para o instante de 97,5 segundos, posicionado na passagem da crista da onda. Na Tabela 4 são apresentadas as malhas utilizadas, passo de tempo para cada simulação e o número total de elementos para cada malha.

Tabela 4: CONVERGÊNCIA DA MALHA

Malha	1	2	3	4
$\delta x(m)$	$1x10^{-2}$	$1x10^{-2}$	$7{,}5x10^{-3}$	$5x10^{-3}$
$\delta y(m)$	$1x10^{-2}$	$5x10^{-3}$	$7x10^{-3}$	$2{,}5x10^{-3}$
$\Delta t(s)$	$1x10^{-4}$	$1x10^{-4}$	$1x10^{-4}$	$1x10^{-5}$
Nelem.	42000	84000	80000	336000

De acordo com a Figura 2 abaixo, as malhas 3 e 4 apresentaram resultados muito próximos, indicando que o número de elementos do refinamento atingiu a convergência.

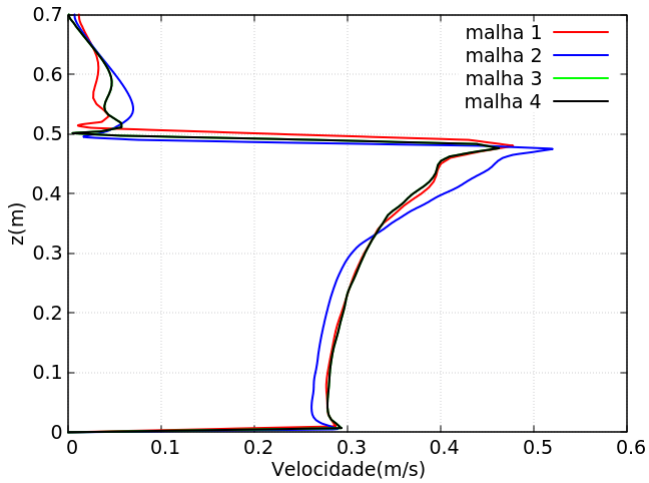


Fig. 2: Convergência da malha.

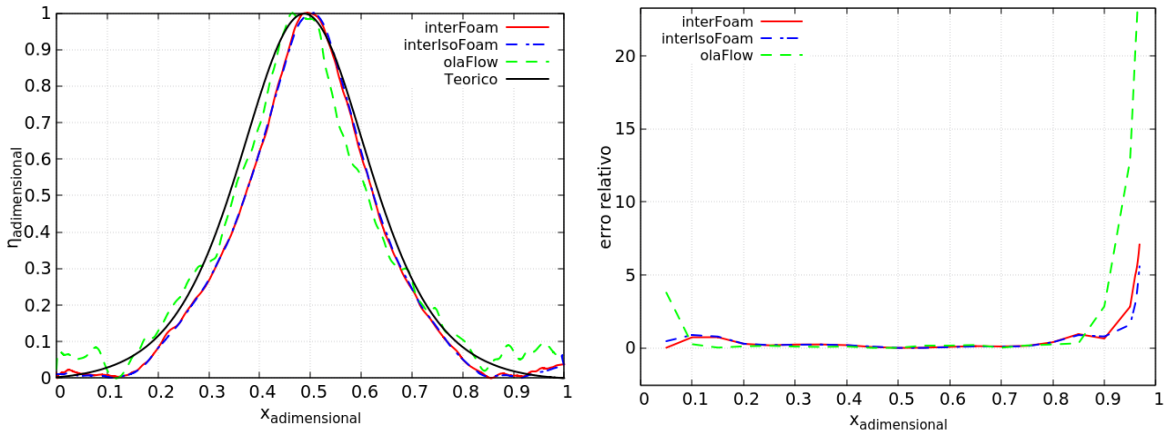
5 RESULTADOS E DISCUSSÕES

Com o objetivo de propagar uma onda próxima da condição de onda ideal, a simulação utiliza a condição de fluido sem viscosidade para os resultados de altura da superfície livre. Para os resultados do campo de velocidade (Seção 5.3), os efeitos viscosos foram considerados. Os resultados obtidos são comparados com o solver interFoam, adotado como parâmetro de referência.

5.1 Comparação numérica e teórica da altura da interface

Uma comparação dos resultados numéricos obtidos para superfície livre com o resultado teórico do perfil cnoidal da onda (Equação 15) e o respectivo erro relativo, são apresentados na Figura 3. A extensão do canal considerada para essa comparação foi de 6 metros, e os resultados do perfil da onda foram adimensionalizados.

O erro médio para o resultado obtido pelo interFoam, interIsoFoam e olaFlow são respectivamente, 1%, 2% e 1%. Logo, diante dos resultados de superfície livre obtidos, o solver escolhido como parâmetro para comparação dos resultados é o interFoam. Nota-se que o olaFlow apresenta o mesmo erro médio que o interFoam, porém, mostra erros elevados próximo à extremidade da saída devido ao efeito da reflexão da onda, o qual será discutido mais adiante.

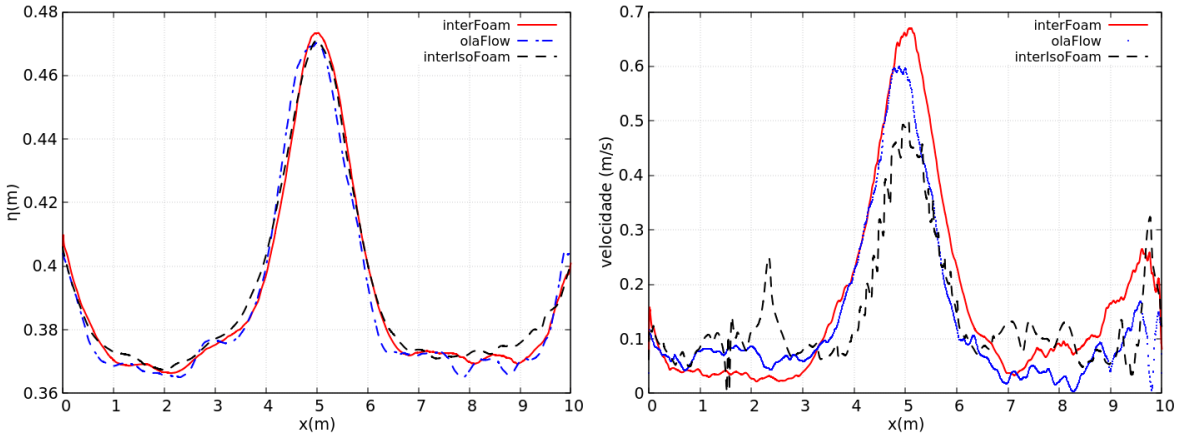


(a) Teórico *versus* numérico. **(b)** Erro relativo.

Fig. 3: Comparação entre o perfil teórico da onda cnoidal e o perfil numérico.

5.2 Analise da altura da superfície livre

A Figura 4 representa uma comparação da captura da superfície livre no instante de tempo $t = 119,5$ segundos e da velocidade nessa interface, quando tem-se a crista da onda passando na metade do comprimento do domínio.



(a) Representação da superfície livre. **(b)** Magnitude da velocidade na interface.

Fig. 4: Representação da superfície livre e velocidade

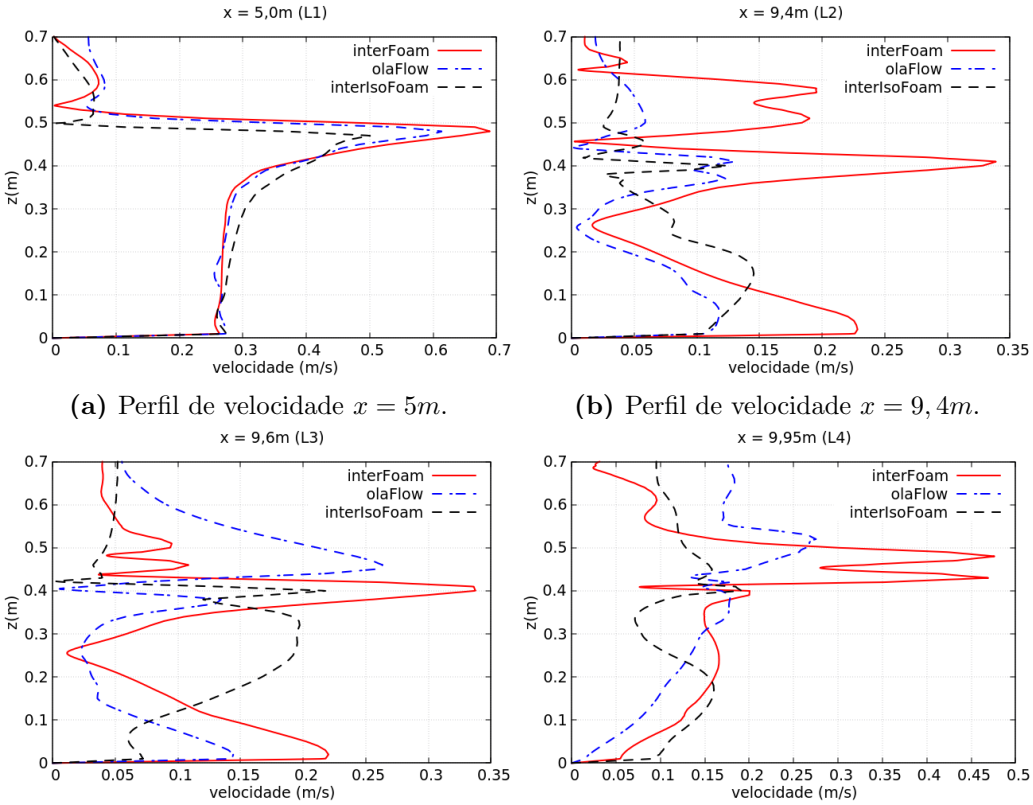
Na Figura 4a, próximo das coordenadas referentes ao início e final do domínio computacional, é possível verificar pequenas variações entre os métodos de captura da interface. Como o comprimento longitudinal do canal é de 10m e o comprimento de onda é de 6m, observa-se parte de uma crista em x = 0 e em x = 10m, com uma variação mais acentuada próximo a saída para os solvers olaFlow e interIsoFoam. Essas regiões são mais sensíveis aos efeitos de reflexão da onda e aos erros numéricos que estão associados aos algoritmos de construção da interface.

O gráfico da Figura 4b representa a variação da magnitude da velocidade ao longo da interface. De acordo com os resultados, existe uma diferença significativa tanto nos valores das velocidades como nas flutuações próximas as posições extremas do canal. É esperado um campo de velocidade maior do que a região de vale, pois há ainda a formação

da onda nesses locais. Nas posições de $1 \leq x \leq 3$ e $6 \leq x \leq 10$, aparecem velocidades espúrias devido ao efeito da reflexão da onda sobre o balanço de massa na interface, o que ocasiona uma resposta ruim tanto para o campo de velocidade como para representação da superfície livre.

5.3 Velocidade

De acordo com o que foi apresentado no item anterior, existe um forte efeito da reflexão da onda sobre o campo de velocidade na superfície livre. Nesta seção, será avaliado a reflexão da onda na velocidade transversal em 4 seções próximas a saída do canal, conforme mostra a Figura 1. Os resultados foram avaliados no instante $t = 119,5s$ quando o escoamento já pode ser considerado desenvolvido. Os perfis de velocidade ao longo das seções transversais podem ser avaliados através da Figura 5.



(a) Perfil de velocidade $x = 5m$. **(b)** Perfil de velocidade $x = 9,4m$.

(c) Perfil de velocidade $x = 9,6m$. **(d)** Perfil de velocidade $x = 9,95m$.

Fig. 5: Velocidades das seções transversais 1 a 4.

Próximo ao fundo, a velocidade do fluido tende à zero. A velocidade na fase líquida aumenta à medida que se aproxima da superfície livre, atingindo a velocidade máxima onde ocorre a mudança de fase. Acompanhando o valor através da componente z, é possível perceber que um pouco acima da altura da superfície livre (região do ar), a velocidade decresce para valores próximos à zero. Este é o efeito de cisalhamento nulo da resistência do ar ao escoamento.

Na Figura 5a, para a posição $x = 5,0m$, a velocidade ao longo da seção transversal é bastante similar para todos os solvers. Pode-se observar nas Figuras 5b, 5c e 5d, maior

é o efeito da reflexão, conforme pode ser verificado na altura $z \approx 0,4m$. Essa resposta numérica é mais acentuada nos solvers interFoam e olaFlow. No interIsoFoam, a absorção da onda é mais efetiva e pode-se notar variações menores no perfil de velocidade próximo da saída.

As imagens apresentadas na Figura 6, mostram o campo de velocidade na saída do canal para os instantes de tempo de 109, 65, 110, 05 e 110, 45 segundos, respectivamente. Pode-se notar uma significativa diferença na resposta de absorção da onda, dada pela método apresentado na Equação 16. No interIsoFoam tem-se um gradiente de velocidade bastante elevado que aumenta no fundo do canal conforme se aproxima da fronteira, porém apresenta boa absorção acima da superfície. Já no interFoam e no olaFlow, nota-se campos de velocidade que apresentam valores menores conforme se aproxima do fundo. No entanto, mostra uma reflexão da onda a partir da fronteira para o interior do domínio, acima da superfície.

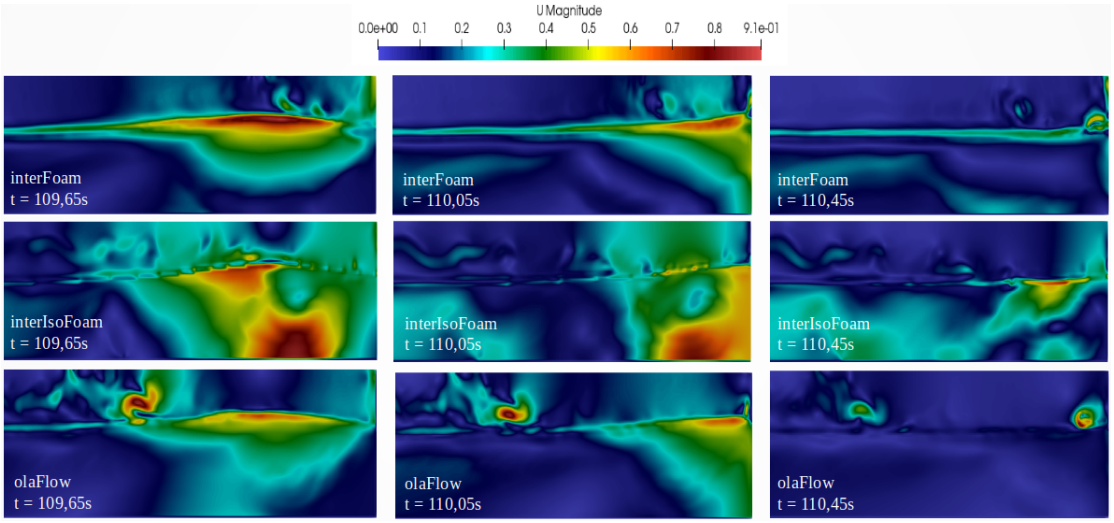


Fig. 6: Efeitos da absorção da onda no campo de velocidade.

5.4 Conclusões

A partir da função cnoidal, foi possível avaliar a propagação de uma onda ao longo do canal hidráulico bidimensional. Os resultados foram calculados através de três diferentes solvers: interFoam, interIsoFoam e olaFlow. Através da análise da captura da superfície livre e da representação do campo de velocidades do escoamento foi possível avaliar comparativamente os métodos numéricos entre si.

Uma avaliação da função de absorção da onda na saída do canal permitiu avaliar as características representativas da condição de contorno na saída.

Com relação ao perfil da interface, os resultados foram próximos ao solver interFoam, apresentando uma diferença maior nas fronteiras devido ao efeito de reflexão da onda. No interIsoFoam, o elevado gradiente de velocidade abaixo da superfície próximo à fronteira, mostrou um efeito da reflexão mais intenso nessa região, enquanto que a velocidade capturada na região da interface pelo lado do ar apresentou pouca variação da energia cinética da crista da onda, tendendo à valores próximos de zero conforme esperado. Já no

olaFlow, como também no interFoam, a baixa absorção foi observada acima da superfície, ocasionando uma reflexão da onda para o interior do domínio.

Podemos concluir que, como olaFlow e interFoam possuem o mesmo método para advecção da interface (MULES), apresentaram resultados próximos tanto de superfície livre, velocidade na interface e efeitos da absorção da onda acima da interface.

Conclui-se que, no geral, os solvers são sensíveis aos efeitos de reflexão da onda próximo as fronteiras e que os métodos de construção da interface alteram a representação da superfície livre e do campo de velocidade.

6 *Agradecimentos*

Os autores agradecem o apoio das seguintes agências de fomento: CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior) e FAPERJ (Fundação Carlos Chagas de Amparo à Pesquisa do Estado do Rio de Janeiro) .

Referências

[1] J. N. C. G. Bahram Haddadi, Christian Jordan. OpenFOAM Basic Training. Technical report, Vienna University of Technology, Viena, Austria, 2015.

[2] J. M. P. Conde. Comparison of Different Methods for Generation and Absorption of Water Waves. *Thermal Engineering*, 18(1):71–77, 2019.

[3] A. M. M. et al. Analysis of Different Methods for Wave Generation and Absorption in a CFD-Based Numerical Wave Tank. *Marine Science and Engineering*, 6(73):1–21, 2018.

[4] P. Higuera, J. L. Lara, and I. J. Losada. Realistic wave generation and active wave absorption for Navier–Stokes models Application to OpenFOAM®. *Coastal Engineering*, 71:102 – 118, 2013.

[5] C. Hirt and B. Nichols. Volume of fluid (vof) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201 – 225, 1981.

[6] H. B. Johan Roenby and H. Jasak. A Computational Method for Sharp and Interface Advection. *Royal Society Open Science*, 2016.

[7] H. B. H. J. Johan Roenby, Bjarke Eltard. A New Volume-Of-Fluid Method in Openfoam. In *VII International Conference on Computational Methods in Marine Engineering*, Barcelona, Spain, 2017. International Center for Numerical Methods in Engineering.

[8] R. J. Lambert. *Development of a Numerical Wave Tank Using OpenFOAM*. PhD thesis, Universidade de Coimbra, Coimbra, Portugal, 2012.

[9] E. Olsson. A description of isoAdvector - a numerical method for improved surface sharpness in two-phase flows. Technical report, Chalmers University of Technology, Gothenburg, Sweden, 2018.

[10] OpenCFD Ltd. OpenFOAM® - User Guide. 2020.

[11] OpenFOAM. Description and utilization of interFoam multiphase solver: General description of the OpenFOAM suite. 2009.

Predição por árvores de regressão da estatura de pacientes após tratamento de hipopituitarismo

Caroline de Oliveira Costa Souza Rosa[1], Alex Borges Vieira[2], Artur Ziviani[1] e Mariza Ferro[1]

[1] *Laboratório Nacional de Computação Científica (LNCC), Petrópolis/RJ, Brasil*
[2] *Depto. de Ciência da Computação, Universidade Federal de Juiz de Fora (UFJF), Juiz de Fora/MG, Brasil*

Resumo

A utilização de dados de saúde para descoberta de trajetórias reais de tratamentos médicos e suas consequências é um tópico de crescente interesse. Este trabalho buscou predizer a estatura final de pacientes com hipopituitarismo considerando características pessoais, os tratamentos, e sequências de tratamentos adotados. Foram utilizados os métodos de regressão linear múltipla, árvores de regressão, árvores de regressão com *bagging* e florestas aleatórias. O erro relativo médio dos modelos variou entre 3% e 5% e as características pessoais foram os atributos com maior influência sobre os modelos.

Palavras-chave: Árvores de regressão, Trajetórias de pacientes, Hipopituitarismo

1 INTRODUÇÃO

Protocolos clínicos são diretrizes de acompanhamento médico para pacientes com alguma condição de saúde específica. Eles beneficiam pacientes e provedores de saúde e, portanto, avaliar se as trajetórias de cuidado dos pacientes, ou seja, as sequências de tratamentos e atendimentos recebidos, estão de acordo com as preconizadas é importante para otimizar os serviços oferecidos. Nesse contexto, a crescente adoção de prontuários eletrônicos e de outros registros de dados na área tem viabilizado a adoção de técnicas, como a mineração de processos, que auxiliam na tarefa de descoberta das trajetórias reais de pacientes através do sistema de saúde a partir da exploração desses dados (Cho et al., 2020). Neste âmbito de análise de trajetória de pacientes, Silva et al. (2020) avaliaram recentemente se as gestantes atendidas pelo Sistema Único de Saúde (SUS) no município de São Paulo entre 2014 e 2015 receberam os acompanhamentos pré-natais recomendados. Os autores

Contato: Caroline O. C. S. Rosa, carolrosa@lncc.br

concluíram que houve grande variabilidade de trajetórias, sendo que a maior parte dessas trajetórias apresentou menos atendimentos do que o preconizado oficialmente pelo SUS para consultas e exames no período pré-natal. Considerando essa abordagem, um tópico de interesse é avaliar como as diferentes trajetórias observadas influenciam no resultado do tratamento de uma condição de saúde (Kempa-Liehr et al., 2020). Dependendo da modelagem feita, esse resultado pode assumir valores categóricos, como identificador de cura ou óbito, ou ainda valores numéricos, como o percentual de melhoria de um problema de saúde.

Para avaliar trajetórias é importante que se tenha uma fonte de dados temporalmente vasta sobre os pacientes. O sistema DATASUS (Brasil, 2020) possui dados abertos de diversos sistemas do Ministério da Saúde (MS), como o Sistema de Informação Ambulatorial (SIA) que inclui o controle de fornecimento de medicamentos. Os registros do banco de dados incluem um identificador do paciente e o código da Classificação Internacional de Doença (CID-10) associado. Em particular, o CID E23.0 compreende casos de hipopituitarismo, uma deficiência de algum dos hormônios da adenohipófise, cuja forma mais comum é a deficiência do hormônio do crescimento (DGH), o que leva à baixa estatura na infância e pode causar outros problemas de saúde posteriores (Portes et al., 2006). Uma trajetória clínica interessante de ser acompanhada é a de pacientes com DGH na infância ou adolescência.

Neste contexto, o objetivo deste trabalho é utilizar informações sobre os diferentes tratamentos adotados e características pessoais dos pacientes com DGH para estimar a estatura do paciente ao término do tratamento. Para isso, deve-se considerar não apenas de quais medicamentos ou procedimentos um paciente fez uso, mas também as sequências de tratamentos consecutivos presentes. Esses dados podem então ser usados como atributos de entrada de algoritmos de regressão. Optou-se por utilizar modelos baseados em árvores de regressão e comparar seu desempenho com o modelo de regressão múltipla. Os dados utilizados são relacionados ao estado do Rio de Janeiro, com registros entre 2008 e 2019, e correspondem a 449 pacientes e 1471 registros.

Este trabalho está organizado da seguinte forma: na Seção 2 são apresentados alguns trabalhos relacionados; na Seção 3 descreve-se tanto o processo de coleta e limpeza dos dados, quanto os métodos utilizados; na Seção 4 são apresentados os resultados obtidos; por fim, na Seção 5 apresentam-se as conclusões.

2 TRABALHOS RELACIONADOS

Em uma revisão sistemática, Erdogan and Tarhan (2018) identificaram um número crescente de publicações na área de mineração de processos com aplicação em protocolos clínicos, especialmente após 2010. A maior parte desses trabalhos buscou descobrir um modelo que descrevesse os processos realizados na prática a partir de dados de prontuários eletrônicos e de outros registros. Prodel et al. (2015), por exemplo, propuseram o uso de programação linear inteira para elaborar um modelo que descreve as trajetórias clínicas de pacientes. Foi feito um estudo de caso para pessoas que receberam um desfibrilador cardíaco implantável. Os autores citam que, tendo mais informações sobre os pacientes, o modelo pode ser usado como base para análises preditivas.

Nesse contexto, Kempa-Liehr et al. (2020) buscaram estimar o tempo para alta hospitalar de pacientes com apendicite a partir dos tratamentos e medicamentos recebidos

durante a internação, além de outros atributos como idade e tempo de cirurgia. Foi utilizado um modelo de regressão generalizado para a predição do tempo e a trajetória de atividades seguida foi tratada como um todo, i.e. não se avaliou a presença de tratamentos isoladamente, apenas a sequência completa deles.

O objetivo deste artigo é semelhante ao de Kempa-Liehr et al. (2020), no sentido de que se busca estimar o resultado de um tratamento considerando as trajetórias seguidas pelos pacientes. No entanto, ao invés de estimar o ganho de altura de pacientes com hipopituitarismo considerando a trajetória completa como um atributo, tem-se o interesse de avaliar como a presença de determinados tratamentos ou de sequência de pares deles interfere no resultado.

3 METODOLOGIA

Nesta seção será descrito o processo de coleta e preparação da base de dados do estudo e, em seguida, serão apresentados os métodos utilizados para o desenvolvimento do trabalho que foi implementado em R (R Core Team, 2020).

3.1 Coleta e Preparação dos Dados

O processo de coleta e preparação dos dados usados neste trabalho é sumarizado na Figura 1. Os passos são descritos com mais detalhes em sequência.

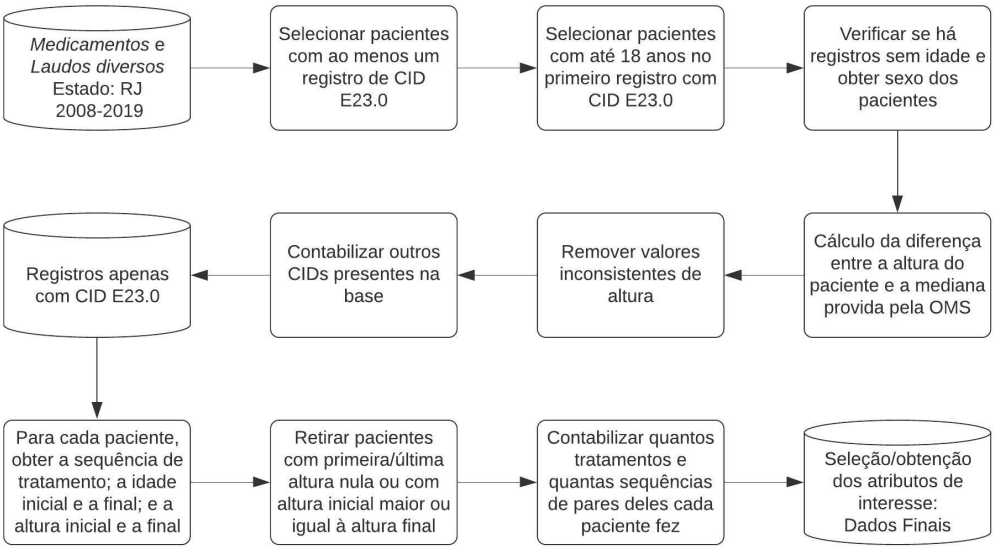


Fig. 1: Principais passos seguidos durante a etapa de coleta e preparação dos dados.

Inicialmente, coletou-se os dados das tabelas *Medicamentos* e *Laudos Diversos* do sistema SIA-SUS referentes ao estado do Rio de Janeiro e compreendendo o período entre jan/2008 e dez/2019, totalizando 12 anos de dados. As tabelas possuem 46 atributos em comum que incluem o código da APAC (Autorização de Procedimento Ambulatorial), sua data de ocorrência, o código do estabelecimento de saúde, o identificador do paciente (codificado) e sua idade, o procedimento principal, o CID primário, dentre outros. A tabela de *Medicamentos* possui ainda mais 5 campos, incluindo peso e altura do paciente.

Filtrou-se as observações para que apenas pacientes que registraram o CID E23.0 em algum momento fossem mantidos na base de dados. Dentre estes, selecionou-se apenas pacientes cujo primeiro registro com CID E23.0 ocorreu até os 18 anos de idade.

Para o acompanhamento de pacientes com hipopituitarismo, além da altura absoluta, deve-se conhecer a distância dela à mediana de altura para a idade e o sexo do paciente. A Organização Mundial de Saúde (OMS)[1,2] provê esses valores de referência, incluindo mediana e desvio-padrão.[3] Todos os registros apresentavam o campo de idade preenchido. Seis pacientes apresentaram o campo relativo ao sexo preenchido de forma inconsistente entre os seus diversos registros. Estes pacientes foram removidos do conjunto de dados pois a comparação com a altura de referência é dependente do sexo. Em seguida, para a altura em cada registro, calculou-se a quantidade correspondente de desvios-padrões abaixo (ou acima) da mediana. Esse valor também foi utilizado para validar os dados de altura, ou seja, a fim de minimizar o risco de se obter resultados inconsistentes por erros de documentação, dados de altura acima/abaixo de 10 desvios-padrões da mediana por idade e sexo foram substituídos por nulo. Esse valor de corte foi considerado conservador, pois alturas inferiores a 2 desvios-padrões são consideradas baixa estatura e valores inferiores a 3 desvios-padrões são considerados anormais (Silva, 2010). Ao todo, 430 valores de altura foram considerados inconsistentes.

Após essas etapas, contabilizou-se os outros CIDs presentes na base para verificar se havia outras condições de saúde que apareciam com frequência nos pacientes com hipopituitarismo. Constatou-se que 96% das observações eram do próprio CID E23.0 e os outros CIDs observados com mais frequência foram E23.2 (Diabetes insípido) com 1.4% e H90.3 (Perda de audição bilateral neuro-sensorial) com 1.2%. Os demais CIDs apresentaram frequência relativa inferior a 0.5% cada.

Selecionou-se apenas os registros do CID E23.0 para a realização das etapas seguintes. Para cada paciente, obteve-se sua sequência de procedimentos principais (atributo "AP_PRIPAL"); a altura absoluta e a altura relativa à mediana no primeiro e no último registro; e a idade no primeiro e no último registro. Utilizou-se a biblioteca *bupaR* do R para realização dessas etapas (Janssenswillen, 2020).

A análise da trajetória clínica de pacientes com hipopituitarismo proposta neste trabalho depende de se conhecer ao menos a altura do paciente no início e no fim do tratamento. Portanto, pacientes com a primeira ou última altura nula foram removidos. Removeu-se também casos em que a altura final era menor do que a inicial. Além disso, os pacientes considerados estão em fase de crescimento e espera-se que haja naturalmente aumento estatural, ainda que abaixo da média. Deste modo, removeu-se casos em que não foi registrada alteração de altura ao longo do tratamento, considerando que essas observações podem estar relacionadas à falta de atualização dos registros dos pacientes. Por fim, obteve-se uma base com 1471 registros de tratamentos de 449 pacientes distintos.

[1]Disponível em `www.who.int/childgrowth/standards/height_for_age/en/`

[2]Disponível em `www.who.int/growthref/who2007_height_for_age/en/`

[3]As referências da OMS são providas de acordo com o número de meses de vida da criança/adolescente, porém a base de dados do SUS apresenta apenas a idade em anos. Então, considerou-se o valor correspondente ao sexto mês de uma idade (em anos) como sua referência. Além disso, as referências incluem idades de 0 a 18 anos. Logo, para registros de pacientes com mais de 18 anos utilizou-se como referência o valor correspondente a 18 anos.

O passo seguinte envolveu identificar quais eram os procedimentos presentes na base e contabilizar quantos cada paciente fez, bem como as sequências de pares de procedimentos realizadas. A Figura 2 apresenta uma ilustração desta etapa. Na Configuração I tem-se as trajetórias dos pacientes obtidas a partir dos dados originais. Com base nessas informações, elabora-se a Configuração II, que contém a contagem de observações de cada tratamento e de seus pares consecutivos de cada paciente.

Configuração I

Paciente	Seq. de tratamentos
Paciente1	A,B,A,A,B
Paciente2	A,A,B,B,B

Configuração II

Paciente	A	B	A→ A	A→ B	B→ A	B→ B
Paciente1	3	2	1	2	1	0
Paciente2	2	3	1	1	0	2

Fig. 2: Configuração antes(I) e após(II) a etapa de contabilização de procedimentos e de pares de procedimentos em sequência realizados por cada paciente.

Dois tratamentos estavam presentes na base de dados deste trabalho. Seus códigos do Sistema de Gerenciamento da Tabela de Procedimentos, Medicamentos e OPM do SUS (SIGTAP)[4] são 604610017 e 604610025. Ambos correspondem a somatropina injetável, com 4UI e 12UI, respectivamente. Além dos medicamentos e suas sequências, os demais atributos são:

- *num_atividades*: número total de instâncias de tratamentos feitos
- *duracao*: duração do tratamento(dias)
- *idade_inicial*: idade inicial (anos)
- *idade_final*: idade final (anos)
- *sexo*: sexo do paciente
- *dp_inicial*: diferença entre a estatura inicial do paciente e a mediana por idade e sexo em termos de desvios-padrões
- *altura_inicial*: estatura inicial (cm)
- *altura_final*: estatura final (cm) - atributo que será utilizado como variável resposta

3.2 Métodos de regressão

Como o objetivo deste trabalho é obter um modelo que faça predição do valor da altura final dos pacientes, que é uma variável contínua, tem-se um problema que pode ser resolvido por métodos de regressão.

Inicialmente, utilizou-se o método de regressão linear múltipla, que é uma generalização do método de regressão linear simples. Na regressão linear múltipla, a variável resposta é aproximada por uma combinação linear de n variáveis previsoras mais um termo constante. Os coeficientes dessa equação são escolhidos de forma que a soma dos

[4]Disponível em `http://sigtap.datasus.gov.br/tabela-unificada/app/sec/inicio.jsp`

quadrados dos resíduos, i.e. a diferença entre o valor real e o predito pela equação, seja mínima. Utilizou-se a função *lm* da biblioteca *stats* do R nesta etapa (R Core Team, 2020).

Utilizou-se também o método de árvore de regressão, que é uma técnica de aprendizado de máquina supervisionado. Neste algoritmo, o conjunto de valores das variáveis previsoras é divido em duas regiões que não se interceptam. Esse particionamento é feito escolhendo-se um atributo e um valor assumido por ele (limiar) tal que observações com valor desse atributo maior do que o limiar são atribuídas a uma região, e observações cujo valor é inferior ao limiar são atribuídas à outra região. O atributo e o limiar são definidos de forma que que a soma dos quadrados dos resíduos entre o valor real das observações em cada região e o valor médio das observações naquela região seja mínimo. Esses passos se repetem para criação de sub-regiões até que algum critério de parada seja atingido, como número mínimo de observações em uma região (James et al., 2013).

Neste trabalho, utilizou-se a função *rpart* do pacote homônimo do R (Therneau and Atkinson, 2019), que se baseia no algoritmo CART (Breiman et al., 1984) para a criação de árvores de regressão. Optou-se por construir árvores sem limite mínimo de observações por folha e realizar a pós-poda da árvore com o critério do parâmetro de custo-complexidade (CP) ótimo provido pelo algoritmo, a fim de se evitar o superajuste aos dados.

Alguns métodos podem ser utilizados a fim de aprimorar a acurácia de árvores de regressão. A técnica de *bagging* consiste em extrair m amostras de tamanho p do conjunto de dados de treino e construir uma árvore de regressão a partir de cada amostra. A predição ($\hat{y}_i$) do valor da variável resposta de uma observação do conjunto teste é obtida da seguinte forma: para cada árvore j, calcula-se o valor predito por ela ($\hat{y}_{ij}$); o valor final predito pelo modelo completo será a média das predições feitas pelas árvores, ou seja, $\hat{y}_i = \frac{1}{m}\sum_{j=1}^{m}\hat{y}_{ij}$ (James et al., 2013).

Uma dificuldade que pode surgir para modelos gerados através de *bagging* é que eventualmente as árvores treinadas podem ser muito semelhantes, de forma que o resultado da predição não apresente ganho significativo comparado ao obtido pela utilização de árvores individuais. Neste contexto, foram propostas florestas aleatórias que funcionam de forma semelhante ao *bagging*, mas com o diferencial de que a cada árvore gerada, o particionamento de um nó não considera todos os atributos como candidatos à divisão, mas apenas um subconjunto aleatório de tamanho *n_atr* deles. Com isso, as árvores tendem a ser menos correlacionadas, aprimorando a acurácia do modelo final (James et al., 2013).

Destaca-se que tanto o *bagging* como as florestas aleatórias aumentam a acurácia da predição, mas diminuem a interpretabilidade do modelo. Utilizou-se a função *randomForest* da biblioteca homônima do R, cujo algoritmo baseia-se em Breiman (2001), para realizar o *bagging* e para criar as florestas aleatórias.

Para a utilização da função para *bagging*, definiu-se *n_atr* (número de atributos a serem considerados para os particionamentos) igual ao número de atributos previsores e optou-se por gerar 50 árvores, após uma avaliação empírica. Para a floresta, foram geradas 1000 árvores e todos os valores possíveis de *n_atr* (de 1 até o total de atributos previsores) foram testados e escolheu-se aquele cujo resultado apresentou menor erro de predição. Em ambos os casos, o tamanho p das amostras tomadas pelo algoritmo para construção das árvores corresponde a 80% do conjunto de treino.

A fim de validar os modelos, os dados foram divididos aleatoriamente em um conjunto

de treino com 80% dos dados e um conjunto de teste com os demais 20%. Para cada método, o modelo correspondente foi gerado a partir do conjunto de treino e a Raiz do Erro Quadrático Médio (RMSE) e o Erro Percentual Absoluto Médio (MAPE) foram calculados com base na predição do valor da altura final das observações contidas no conjunto de teste. A fim de mitigar os efeitos da aleatoriedade na escolha dos conjuntos de treino e teste, repetiu-se o processo (randômico) de divisão dos dados, treinamento dos modelos, predição e cálculo dos erros 50 vezes e, então, calculou-se a média dos 50 valores obtidos do RMSE e do MAPE.

4 RESULTADOS E DISCUSSÃO

Apresenta-se inicialmente algumas análises feitas sobre os dados obtidos após a etapa de preparação da base. Os gráficos de dispersão das variáveis previsoras versus a variável resposta são apresentados na Figura 3. O medicamento cujo código é 604610017 é representado por "X604610017"; o medicamento de código 604610025 é representado por "X604610025"; duas utilizações do medicamento 604610017 em sequência são representadas por "X604610017..X604610017"; duas utilizações do medicamento 604610025 em sequência são representadas por "X604610025..X604610025"; a sequência entre os medicamentos 604610017 e 604610025 é representada por "X604610017..604610025"; e, de forma análoga, a sequência entre os medicamentos 604610025 e 604610017 é representada por "X604610025..604610017". As observações foram coloridas de acordo com o sexo do paciente.

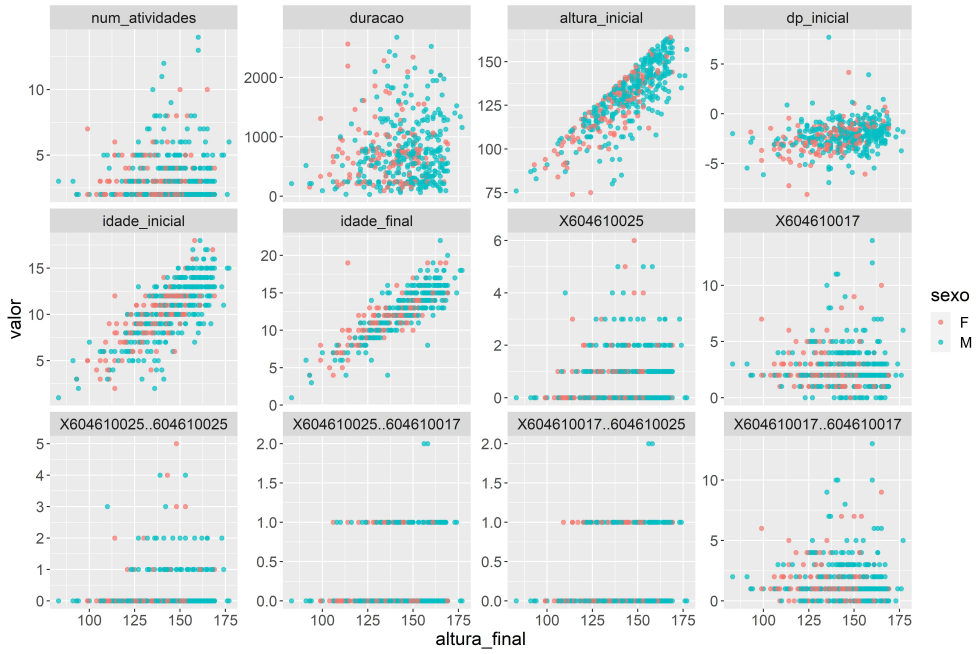


Fig. 3: Distribuição das variáveis previsoras com relação à variável resposta.

Observa-se que os gráficos de *idade_inicial*, *idade_final* e *altura_inicial* sugerem uma possível relação linear com a variável resposta. Os gráficos dos demais atributos não são tão claros quanto a uma possível correlação com a altura final.

Observando a Figura 3, pode-se observar que o medicamento 604610017 possui maior

densidade de pontos. De fato, ele é o tratamento mais frequente, com 1167 ocorrências. Por outro lado, há 304 ocorrências do tratamento 604610025. Além disso, foram observadas 70 trajetórias completas distintas, sendo que seis delas correspondem a 70.8% dos pacientes, como mostrado na Figura 4.

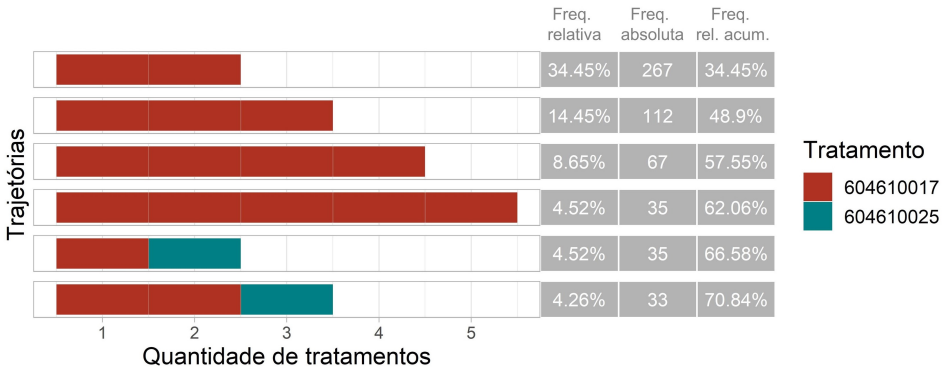


Fig. 4: Trajetórias mais frequentes observadas.

Verifica-se que pouco mais de um terço dos pacientes recebe dois tratamentos 604610017. Além disso, somando esses pacientes aos que receberam três tratamentos 604610017, cobre-se aproximadamente 50% dos casos. Dentre essas seis trajetórias mais frequentes, as que contêm o tratamento 604610025, o têm como seu último registro.

Procedeu-se então à geração dos modelos de regressão. Após as 50 repetições do processo de divisão dos dados em conjuntos de treino e teste, geração de modelos e cálculo de predições, obteve-se os valores médios do RMSE e do MAPE. Os resultados são apresentados na Tabela 1. Destaca-se que o valor de *n_atr* utilizado nas florestas aleatórias foi 9.

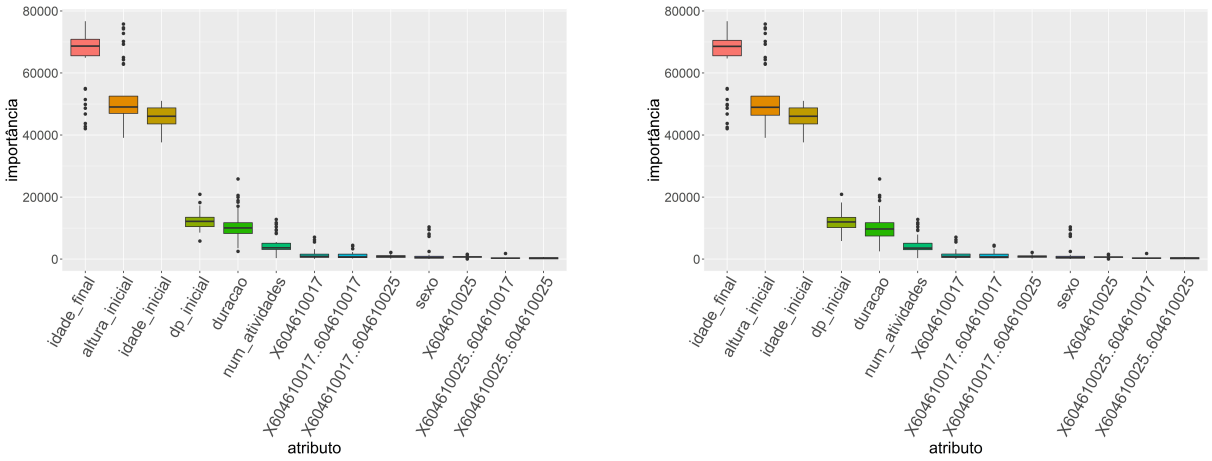
Tabela 1: Valores médios dos erros de predição da variável "altura final" por cada método ao fim de 50 execuções.

Método	**RMSE**	**MAPE**
Regressão linear múltipla	5.839	0.029
Árvore de regressão	8.177	0.045
Árvore de regressão com poda	8.221	0.045
Árvore de regressão com *bagging*	5.619	0.030
Floresta aleatória	5.768	0.031

Observa-se que o ranqueamento dos métodos com base nos valores do erro produziria resultados distintos dependendo da medida de erro adotada. Esta diferença é em parte explicada pelo fato de que o RMSE tende a penalizar observações com erros grandes, enquanto que para o MAPE, se o erro for grande, mas o valor real também for elevado, o impacto é reduzido. No entanto, pode-se inferir que o modelo de regressão linear teve desempenho similar aos modelos de árvore de regressão com *bagging* e floresta aleatória. Além disso, estes três tiveram erro menor do que os modelos de árvore de regressão individual e árvore podada. Os erros relativos médios variaram entre 3% e 5%. Apesar de

serem aparentemente baixos, para uma altura final de 1.5m, esses valores corresponderiam a uma variação entre ±4.5 e ±7.5cm na altura do paciente, o que pode ser considerado elevado, dependendo do caso tratado.

Averiguou-se também a importância de cada atributo para os modelos envolvendo árvores, de acordo com os valores retornados pelas respectivas funções. Os resultados para árvores de regressão e árvores de regressão podadas são apresentados na Figura 5.

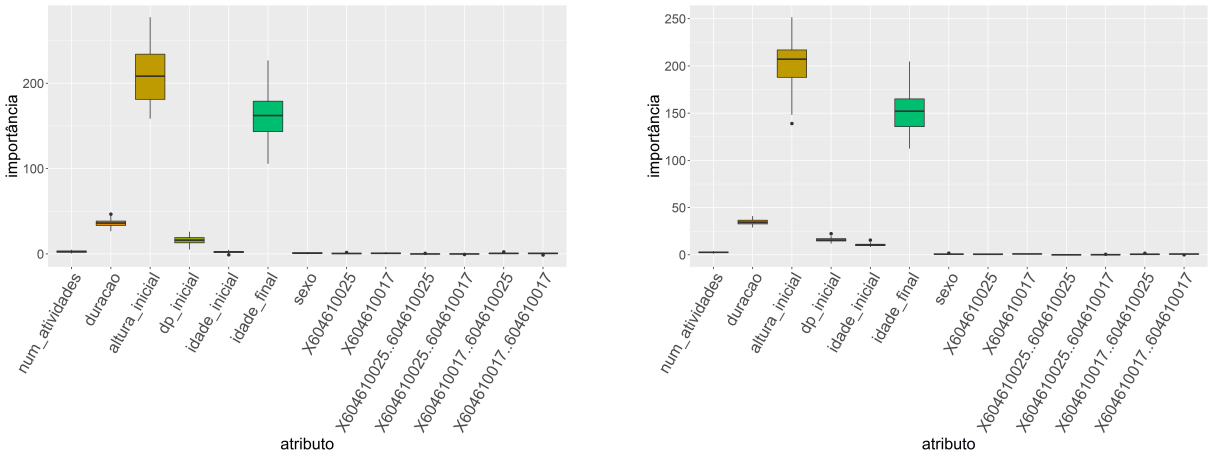


(a) Árvores de regressão. **(b)** Árvores de regressão podadas.

Fig. 5: Importância dos atributos para árvores e árvores podadas.

Observa-se que os resultados de importância das variáveis são bastante semelhantes entre árvores simples e podadas. As três variáveis que aparentavam ter relação linear com a altura final de acordo com a Figura 3 foram as que apresentaram maior importância. Os atributos "dp_inicial", "duracao" e "num_atividades" aparecem em sequência. Os tratamentos e as sequências de tratamentos em si não foram essenciais para os modelos, bem como o sexo dos pacientes.

Os valores de importância das variáveis para os modelos de árvores com *bagging* e florestas randômicas são apresentados na Figura 6.



(a) Importância dos atributos para as árvores com *bagging*. **(b)** Importância dos atributos para as florestas randômicas.

Fig. 6: Importância dos atributos para árvores com *bagging* e florestas randômicas.

Nestes casos, ao invés de três atributos se destacando, tem-se apenas dois: altura inicial e idade final. A terceira variável com maior importância é a duração do tratamento, mas esta possui valor de importância médio cerca de três vezes menor do que o das duas primeiras. Novamente, os tratamentos e suas sequências não apresentaram contribuição significativa para os modelos.

5 CONSIDERAÇÕES FINAIS

Constatou-se que utilizar árvores de regressão com *bagging* ou florestas randômicas gera erros de predição menores do que modelos de árvores de regressão individuais, podadas ou não. No entanto, para o caso estudado, a eficiência dos modelos não superou à obtida com regressão linear múltipla. As variáveis que mais contribuíram para os modelos foram aquelas relacionadas à estatura inicial e idade dos pacientes. Entretanto, alcançou-se o objetivo de utilizar as trajetórias percorridas como atributos dos modelos de regressão e recomenda-se testar esse método no contexto de outras condições de saúde. A fim de aumentar a precisão da predição da altura final de pacientes com hipopituitarismo, sugere-se utilizar outros algoritmos de aprendizado de máquina, como SVR e redes neurais, em trabalhos futuros.

Agradecimentos

Este trabalho recebe apoio parcialmente do CNPq, CAPES e FAPERJ.

REFERÊNCIAS

[1] Brasil (2020). *Arquivos de Dados*. Disponível em `http://www2.datasus.gov.br/DATASUS/index.php?area=0901`. Acesso em 18 ago. 2020.

[2] Breiman, L. (2001). "Random forests". *Machine learning*, 45(1):5–32.

[3] Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

[4] Cho, M., Kim, K., Lim, J., Baek, H., Kim, S., Hwang, H., Song, M., and Yoo, S. (2020). "Developing data-driven clinical pathways using electronic health records: The cases of total laparoscopic hysterectomy and rotator cuff tears". *International Journal of Medical Informatics*, 133(May 2019):104015.

[5] Erdogan, T. G. and Tarhan, A. (2018). "Systematic Mapping of Process Mining Studies in Healthcare". *IEEE Access*, 6:24543–25567.

[6] James, G., Witten, D., Hastie, T., and Tibshirani, R. (2013). *An introduction to statistical learning*, volume 112. Springer.

[7] Janssenswillen, G. (2020). *bupaR: Business Process Analysis in R*. R package version 0.4.4.

[8] Kempa-Liehr, A. W., Lin, C. Y. C., Britten, R., Armstrong, D., Wallace, J., Mordaunt, D., and O'Sullivan, M. (2020). "Healthcare pathway discovery and probabilistic machine learning". *International Journal of Medical Informatics*, 137.

[9] Portes, E., Maccagnan, P., Vieira, T., and Ribeiro, S. (2006). *Projeto Diretrizes. Hipopituitarismo: Diagnóstico*. Disponível em `https://diretrizes.amb.org.br/_BibliotecaAntiga/hipopituitarismo-diagnostico.pdf`. Acesso em 16 ago. 2020.

[10] Prodel, M., Augusto, V., Xie, X., Jouaneton, B., and Lamarsalle, L. (2015). "Discovery of patient pathways from a national hospital database using process mining and integer linear programming". In: *IEEE International Conference on Automation Science and Engineering*, IEEE, pages 1409–1414.

[11] R Core Team (2020). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria.

[12] Silva, A. C. C. S. d. (2010). *Estudo clínico e molecular em uma coorte de portadores de deficiência de hormônio de crescimento na Bahia*. Fiocruz.

[13] Silva, M. A., Santos, N. B., Rosa, C. O. C. S., Santos, M. R., Ito, M., Vieira, A. B., Ziviani, A., and Oliveira, R. M. (2020). "Análise dos Atendimentos de Gestantes na Rede de Atenção Básica de Saúde no Município de São Paulo". In: *XX Simpósio Brasileiro de Computação Aplicada à Saúde*, SBC, pages 250–261.

[14] Therneau, T. and Atkinson, B. (2019). *rpart: Recursive Partitioning and Regression Trees*. R package version 4.1-15.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Bioinformática aplicada ao estudo dos coronavírus: análise de mutações nas proteínas Spike e Main Protease

Emily dos Santos Silva[1], Pablo Nunes Cortez[1] e Gregório Kappaun Rocha[1]

[1] *Instituto Federal Fluminense (IFF), Macaé/RJ, Brasil*

Abstract

Os coronavírus (CoVs) são patógenos de origem zoonótica, sendo associados a problemas respiratórios leves a moderados em seres humanos. Três espécies podem causar quadros graves de insuficiência respiratória em humanos (SARS-CoV, MERS-CoV e SARS-CoV-2 [1]. As informações sobre os genomas de CoVs sequenciados estão disponíveis para estudo em diversos bancos de dados internacionais, tais como o *China National GeneBank* (CNGB) e o *GenBank*. O projeto no qual este trabalho enquadra-se busca, através da aplicação de ferramentas de bioinformática, contribuir para o entendimento dos CoVs. De maneira específica, busca-se, aqui, identificar e analisar mutações entre sequências já determinadas de duas proteínas importantes para o mecanismo de ação do SARS-CoV-2 (*Spike* e a *Main Protease*). Coletou-se 72 sequências da proteína *Spike* e 186 da enzima *Main Protease* depositadas no PDB. As sequências foram comparadas através do alinhamento múltiplo realizado pelo programa *T-Coffee*. O genoma depositado no *GenBank* com o código MN975262 foi usado como referência para os estudos de mutação. Como resultado parcial para a *Spike*, identificou-se nas sequências 6ZOZ, 6ZOY e 6X29 mutações S383C; na sequência 6ZOX mutação G413C; nas sequências 6X2A, 6X2C e 6X2B as mesmas mudanças foram encontradas, mutações A570L e T573I. Análises envolvendo a enzima *Main Protease* estão em elaboração. Estudos com base na estrutura tridimensional das proteínas identificadas com mutações em sítios importantes também serão realizados futuramente.

Keywords: Coronavírus, Spike, Main Protease, Covid-19.

Referências

[1] C. Wang, P. W. Horby, F. G. Hayden, and G. F. Gao. A novel coronavirus outbreak of global health concern. *The Lancet*, 395(10223):470–473, 2020.

Contato: Gregório Kappaun Rocha, D.Sc., gregorio.rocha@ifff.edu.br

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Nova abordagem numérica para representar ruptura de obras subterrâneas profundas via FEM

Erick Rógenes[1], Leandro Lima Rasmussen[2] e Márcio Muniz de Farias[3]

[1,2,3] *Universidade de Brasília, Brasilia/DF, Brasil*

Resumo

Neste trabalho é apresentada uma nova abordagem numérica, chamada Continuum Voronoi Block Model. Essa ferramenta visa representar o processo de ruptura em obras subterrâneas e consiste em simular o meio rochoso por meio de um conjunto de blocos, formados por um mosaico Voronoi, unidos em suas interfaces por elementos de junta. Para validação da ferramenta, representou-se numericamente o estudo de caso do túnel Mine-By. O modelo foi capaz de capturar a geometria de ruptura da escavação e também apresentou de forma explícita o processo de deterioração do maciço. Tais resultados evidenciam a potencialidade do Continuum Voronoi Block Model para previsão do comportamento de escavações com elevadas tensões de campo.

Palavras-chave: Rochas, RocSciense, Mosaico Voronoi, Túnel Mine-By, Junta de Goodman

1 INTRODUÇÃO

Obras subterrâneas podem envolver escavações em meios rochosos com elevadas tensões de campo. Nessas obras, rupturas frágeis podem se manifestar durante as etapas de escavação na forma de desplacamentos rochosos ou explosões rochosas, acarretando perdas econômicas e riscos à segurança dos trabalhadores [9]. Assim, a realização de análises computacionais que indiquem os fenômenos de ruptura frágil em etapa anterior às escavações é desejável. No entanto, modelos constitutivos usualmente adotados para solos (por exemplo Mohr-Coulomb, Von-Mises, Cam-Clay, etc) não são adequados para rochas, pois não conseguem representar os processos físicos relacionados com o dano e a perda de integridade física desses materiais. Outro problema na simulação de escavações nessas condições é que a resistência da rocha registrada em laboratório é incompatível com aquela registrada em campo [4].

Contato: Erick Rógenes, rogenessimao@hotmail.com

Em análises de escavações em rocha por elementos finitos, três modelos constitutivos são normalmente utilizados: Cohesion Weakening and Frictional Strengthening [8]; Damage Initiation and Spalling Limit [4]; e Cohesion Softening and Friction Hardening [5]. Todos esses modelos são fundamentados na hipótese de que durante a ruptura do maciço ocorre uma mobilização não simultânea da coesão e do atrito. Eles são efetivos para retroanalisar a profundidade de ruptura, entretanto são limitados quando aplicados para interpretação do processo físico real, pois essas ferramentas representam o fraturamento implicitamente.

Diante disso propõe-se o Continuum Voronoi Block Model (CVBM), um modelo desenvolvido no programa RS2 [14] e baseado na representação do maciço rochoso por meio de um conjunto de blocos, formados por um mosaico Voronoi, unidos por elementos de junta de Goodman [7]. Essa nova abordagem numérica visa representar a ruptura frágil de escavações e também o processo de fraturamento de maneira explícita e assim possibilitar um maior entendimento sobre o processo de ruptura em escavações sujeitas a elevadas tensões de campo.

2 METODOLOGIA

O CVBM é um modelo baseado no método dos elementos finitos e implementado no programa Rs2 [14]. Nesse modelo o maciço rochoso é representado como um conjunto de partículas unidas em suas interfaces. As partículas são estabelecidas a partir de um mosaico Voronoi e cada bloco da tesselação Voronoi é discretizado internamente por uma malha de elementos finitos. Nas bordas das partículas são inseridos elementos de junta de Goodman [7] que permitem simular a interação entre blocos.

2.1 Mosaico Voronoi

O processo de formação do mosaico Voronoi inicia a partir de uma distribuição de pontos, denominados sementes ou geradores. Ao redor dos geradores são criadas retas que separam a região de domínio de uma semente em relação às suas vizinhas e é por meio da intersecção dessas retas que os polígonos Voronoi são formados, em outras palavras, isso significa que qualquer ponto dentro da área de uma célula Voronoi estará mais próximo da sua semente mãe que de qualquer outra semente [6].

O Rs2 possui um gerador de mosaico Voronoi no qual a formação das partículas é feita de forma randômica e o tamanho dos blocos Voronoi é controlado pelo comprimento médio das juntas.

2.2 Modelos constitutivos

As juntas e os blocos Voronoi podem ser tratados como elásticos ou elastoplásticos. No CVBM ambos serão admitidos como elementos elastoplásticos. Adotou-se essa hipótese para que as rupturas inter-granulares (entre grãos), intra-granulares (no interior dos grãos) e trans-granulares (atravessando vários grãos) pudessem ser representadas [15]. Além do mais, a possibilidade de ruptura dos elementos sólidos e das juntas diminui a existência de caminhos preferenciais para a ruptura.

O programa Rs2 permite que diferentes critérios de ruptura possam ser adotados para os grãos Voronoi e para os elementos de junta. Empregou-se o critério de Mohr-Coulomb

acoplado com o critério de Rankine para ambos os elementos. Uma vantagem em adotar o critério de Mohr-Coulomb é que seus parâmetros possuem significado físico simples, o que contribui para uma melhor interpretação fenomenológica do modelo. A Tabela 1 apresenta um resumo dos parâmetros do CVBM.

Tabela 1: Parâmetros de entrada do modelo CVBM

Tipo	Bloco Voronoi	Juntas
Parâmetros elásticos	Módulo de Young (E_v) Coeficiente de Poisson (ν_v)	Rigidez normal (K_n) Rigidez cisalhante (K_s)
Parâmetros de Pico	Ângulo de atrito de pico (ϕ_{pv}) Coesão de pico (C_{pv}) Tração de pico (T_{pv})	Ângulo de atrito de pico (ϕ_{pj}) Coesão de pico (C_{pj}) Tração de pico (T_{pj})
Parâmetros residuais	Ângulo de atrito residual (ϕ_{rv}) Coesão residual (C_{rv}) Tração residual (T_{rv}) Ângulo de dilatância (Ψ)	Ângulo de atrito residual (ϕ_{rj}) Coesão residual (C_{rj}) Tração residual (T_{rj}) -

Como forma de simplificar o modelo, a coesão e a resistência à tração residuais (C_{rv}, T_{rv}, C_{rj} e T_{rj}) foram adotadas como nulas tanto para os blocos Voronoi como para as juntas. Para os parâmetros de ângulo de atrito residual desses elementos (ϕ_{rv} e ϕ_{rj}) foram atribuídos valores inferiores aos de pico, para se obter um comportamento frágil no modelo. Tais simplificações foram feitas baseada na hipótese da mobilização de uma resistência puramente friccional após a ruptura, assim como foi considerado em [6].

2.3 Estudo de caso

Para validar a aplicação do modelo CVBM na realização de simulações numéricas de obras subterrâneas buscou-se simular o caso do túnel Mine-By. Esta escavação possui 3,50 m de diâmetro, 46,00 m de comprimento e foi executada a 420,00 m de profundidade na província de Manitoba, Canadá. Este túnel faz parte dos trabalhos desenvolvidos no Underground Research Laboratory (URL) da Atomic Energy of Canada Limited (AECL) e foi executado entre os anos de 1989 a 1995 com o objetivo de estudar o processo de ruptura que ocorre em escavações com elevadas tensões de campo, por isso este caso foi bem registrado em diferentes trabalhos ([10], [11] e [13]).

A rocha do laboratório URL localizada a 420,00 m de profundidade, trata-se de um granito. Esse material foi estudado por [10] e suas propriedades estão resumidas na Tabela 2. Além das propriedades mecânicas da rocha circundante do túnel, as tensões *in situ* também foram investigadas e possuem os seguintes valores: σ_1=60 $\pm$ 3 MPa, σ_2=45 $\pm$ 4 MPa e σ_3=11 $\pm$ 2 MPa. O túnel Mine-By foi escavado na direção da tensão principal intermediária e o ângulo da tensão principal maior com a horizontal variou entre 11° a 14° [11].

Tabela 2: Propriedades mecânicas do Granito Lac du Bonnet a 420,0 m de profundidade

Parâmetro	Valor	Referência
Módulo de Young (GPa)	60,00	[8]
Coeficiente de Poisson	0,20	[8]
UCS (MPa)	224,00	[11]
Resistência à Tração (MPa)	10,00	[8]

Tendo em vista a caracterização precisa das propriedades mecânicas do granito, dos níveis de tensões *in situ* e do processo evolutivo de ruptura, o caso do túnel Mine-by frequentemente é utilizado em simulações numéricas ([8], [4], [16] e [12]). Outro fato que colabora para o uso desse estudo de caso como base para a calibração de modelos numéricos é que o maciço circundante ao túnel possui as mesmas características da rocha intacta e assim pode ser tratado como um maciço homogêneo e isotrópico.

2.3.1 Geometria do modelo, condições de contorno e tensões in situ

Com o objetivo de representar a ruptura do túnel Mine-By usando o CVBM, construiu-se um modelo composto por três regiões. A primeira região, denominada de zona externa, possui dimensões de 50 x 50 m e foi estabelecida para diminuir o efeito das condições de contorno. A segunda região, denominada de zona interna, é delimitada por um círculo com diâmetro de 10,5 m, três vezes o diâmetro do túnel. A terceira região, conhecida como zona escavada, consiste numa região circular com diâmetro de 3,5 m, a qual representa a seção transversal do túnel Mine-by. O modelo CVBM foi implementado na zona interna, enquanto que nas demais regiões adotou-se materiais elásticos. Essa divisão foi estabelecida para diminuir os custos computacionais das simulações. Tal técnica já foi utilizada por outros pesquisadores como, por exemplo, [3], [12], [19]. A Fig. 1 mostra os detalhes da geometria do modelo.

Os polígonos Voronoi precisam ser pequenos o suficiente para que não sejam formadas regiões preferenciais de ruptura [2]. Com isso em mente e levando-se em conta o custo computacional da simulação, definiu-se um mosaico Voronoi com comprimento médio de juntas igual a 7,5 cm.

A discretização do modelo foi feita com uma malha de elementos triangulares de seis nós. Na região central delimitada por uma área circular de 7,5 m foi realizado um refinamento da malha como pode ser observado na Fig. 1. Tal configuração gerou uma malha com mais de 96000 elementos.

Por se tratar de um túnel profundo, restrições laterais e horizontais foram aplicadas em todas as bordas do modelo. As tensões principais, σ_1=60 MPa, σ_2=45 MPa e σ_3=11 MPa, estabelecidas a partir de medidas em campo foram incorporadas na simulação. Adotou-se um ângulo de 11° entre a direção da tensão principal maior com a horizontal, assim como foi feito em [4].

Para simular o efeito de suporte imposto pela face do túnel empregou-se o método da relaxação das tensões [17]. A simulação foi dividida em 11 estágios [14], dos quais o primeiro é utilizado apenas para imposição das tensões *in situ*. No segundo estágio é realizada a escavação do túnel e aplica-se um conjunto de tensões na parede do túnel,

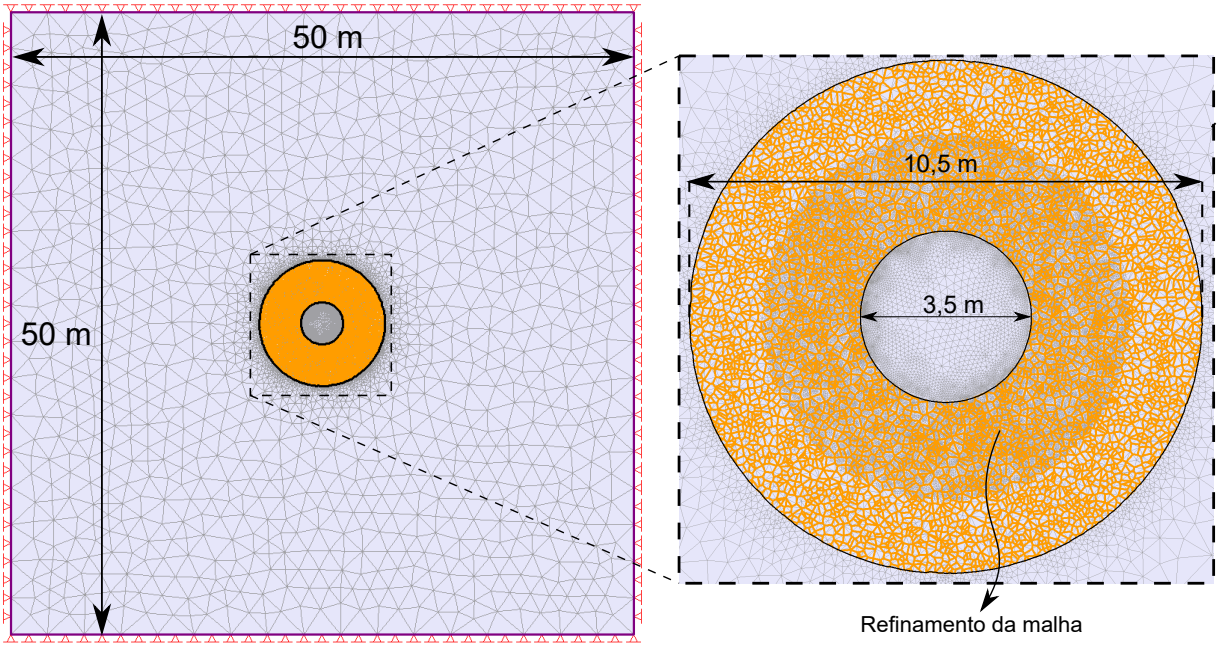


Fig. 1: Configuração do modelo usado para representar o túnel Mine-By

tais tensões se equalizam às tensões induzidas pela escavação. As tensões adicionadas no segundo estágio foram incrementalmente reduzidas nos estágios subsequentes até que no último estágio nenhuma tensão foi aplicada na parede do túnel. Destaca-se que a zona escavada torna-se uma região vazada desde o segundo estágio, momento no qual a região é excluída do modelo. As propriedades elásticas dessa região não interferem nos resultados posteriores à escavação, visto que essa zona só faz parte do modelo no momento de imposição das tensões *in situ*.

2.3.2 Calibração do modelo

As propriedades elásticas do material aplicado na zona externa e na zona escavada foram obtidas diretamente dos resultados de laboratório (Tabela 2). Como as micropropriedades do modelo CVBM não são medidas diretamente do laboratório, o processo de calibração foi dividido em duas etapas: calibração de laboratório e calibração de campo.

A primeira etapa destinou-se para calibração das propriedades de laboratório do granito: módulo de Young, coeficiente de Poisson, resistência à compressão simples e à tração. Assim simulou-se ensaios de compressão simples e de tração brasileiro. Adotou-se no modelo de laboratório o mesmo comprimento médio das juntas estabelecido para o modelo de campo. Por conta disso foi simulado um corpo de prova com 4,00 m de comprimento e 2,00 m de diâmetro (Fig. 2). Assim como no modelo de campo também foi utilizada uma malha de elementos triangulares de seis nós na simulação dos ensaios de laboratório.

Para simular o ensaio de compressão simples com controle de deformação, foram definidas restrições de deslocamento verticais na base do corpo de prova e na parte superior foram impostos deslocamentos verticais. O processo de carregamento foi dividido em 20 estágios e utilizou-se 20 pontos de monitoramento no centro do corpo de prova para análise da tensão axial média e outros 40 pontos de monitoramento nas laterais (20 em

cada lado) para quantificação das deformações laterais. Os dados coletados nesses pontos foram utilizados para o cálculo do módulo de Young, coeficiente de Poisson e da resistência à compressão.

Com o intuito de simular o ensaio de tração brasileiro, aplicou-se restrições verticais na parte inferior do corpo de prova e deslocamento verticais na parte superior. Os deslocamentos verticais foram aplicados de forma incremental ao longo de 20 estágios. A força máxima registrada no processo de carregamento (F_{max}) foi usada na Eq. (1) para o cálculo da resistência à tração (T):

$$T = \frac{F_{max}}{2\pi Rt} \tag{1}$$

onde R é o raio e t é a espessura da amostra, adotada como 1 m por se tratar de uma análise de deformação plana.

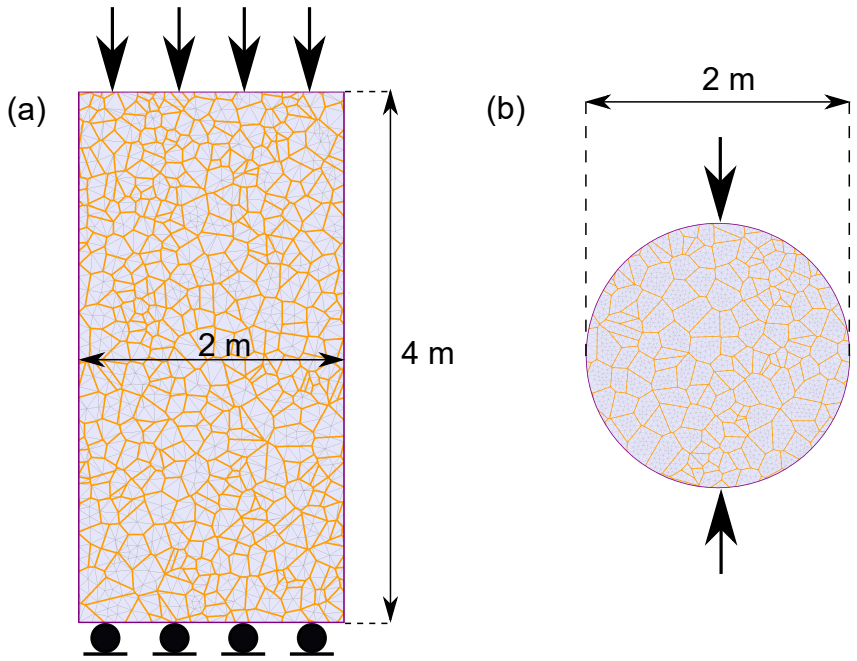


Fig. 2: Configuração do modelo de laboratório: (a) ensaio de compressão simples e (b) ensaio de tração brasileiro

Após a calibração dos modelos de laboratório, procedeu-se com a calibração do modelo de campo. Essa etapa se faz necessária tendo em vista a incompatibilidade entre a resistência de campo e de laboratório ([4] e [9]). As micropropriedades estabelecidas na primeira etapa da calibração foram inseridas no modelo de campo e em seguida reduziu-se a resistência do material até que a profundidade de ruptura do modelo fosse compatível com a registrada em campo.

3 RESULTADOS E DISCUSSÕES

A partir de um processo iterativo de modificação dos microparâmetros do modelo verificou-se que o conjunto apresentado na Tabela 3 é capaz de representar o comportamento do granito circundante do túnel Mine-By em escala de laboratório. Os resultados numéricos

encontram-se na Tabela 4, tais valores foram calculados a partir dos dados registrados nos pontos de monitoramento inseridos nos corpos de prova numérico, conforme discutido na Seção 2.3.2. Na Tabela 4 é feito ainda uma comparação dos resultado do CVBM com os valores registrados em laboratório, onde verifica-se um erro máximo de 7,5%.

Tabela 3: Microparâmetros do CVBM para representar o comportamento do granito Lac du Bonnet em laboratório

Tipo	Blocos Voronoi		Juntas	
Parâmetros elásticos	E_v(GPa)	90,0	K_n (GPa/m)	1200
	ν_v	0,19	K_n/K_s	1,5
Parâmetros de pico	C_{pv}(MPa)	100	C_{pj}(MPa)	65,0
	$\phi_{pv}(^\circ)$	55,0	$\phi_{pj}(^\circ)$	42,0
	T_{pv}(MPa)	14,0	T_{pj}(MPa)	6,0
Parâmetros residuais	C_{rv}(MPa)	0,0	C_{rj}(MPa)	0,0
	$\phi_{rv}(^\circ)$	30,0	$\phi_{rj}(^\circ)$	10,0
	T_{rv}(MPa)	0,0	T_{rj}(MPa)	0,0
	$\psi(^\circ)$	30,0	-	-

Tabela 4: Comparação entre resultados numéricos e laboratoriais no granito Lac du Bonnet

Parâmetro	Numérico	Laboratório	Erro (%)
Módulo de Young (GPa)	59,00	60,00	-1,7
Coeficiente de Poisson	0,20	0,20	0,0
UCS (MPa)	212,00	224,00	-5,4
Resistência à Tração (MPa)	10,75	10,00	7,5

Os parâmetros apresentados na Tabela 3 foram inseridos no modelo de campo, e o resultado da simulação é apresentado na Fig. 3a. A extensão da zona deteriorada foi definida a partir da plastificação das juntas e dos blocos. Observando o resultado apresentado, verifica-se que o maciço continua praticamente intacto, com apenas alguns elementos plastificados ao redor da escavação, evidenciando a divergência entre a resistência de campo e de laboratório.

Novas simulações foram conduzidas, as quais fizeram parte da etapa de calibração de campo. Nessas simulações, reduziu-se a coesão das juntas e dos blocos Voronoi, enquanto os demais parâmetros foram mantidos constantes. A redução da coesão foi feita de forma gradual até que se definiu o par C_{pv}=36,0 MPa e C_{pj}=19,5 MPa como os melhores parâmetros para representação da zona de ruptura do túnel Mine-By. O resultado da simulação com esses parâmetros é apresentado na Fig. 3b.

Verifica-se que a geometria da ruptura representada no modelo ocorre em formato de "V" de maneira similar com a ruptura registrada em campo (Fig. 3c). A formação do

entalhe em forma de "V" é típica de escavações com elevadas tensões de campo e recebe o nome de *Spalling* [4]. A profundidade da zona deteriorada também foi representada com precisão e, apesar do CVBM ser baseado no método dos elementos finitos, é possível observar de forma explícita a formação das macrofraturas por meio da plastificação dos elementos de junta.

Na Fig. 3b é possível ver também a ruptura por tração ocorrida nas laterais do túnel. Tal ruptura não pode ser verificada visualmente por meio das imagens do perfil final da escavação, entretanto [1] relatou sua ocorrência a partir de medições realizadas com emissões acústicas.

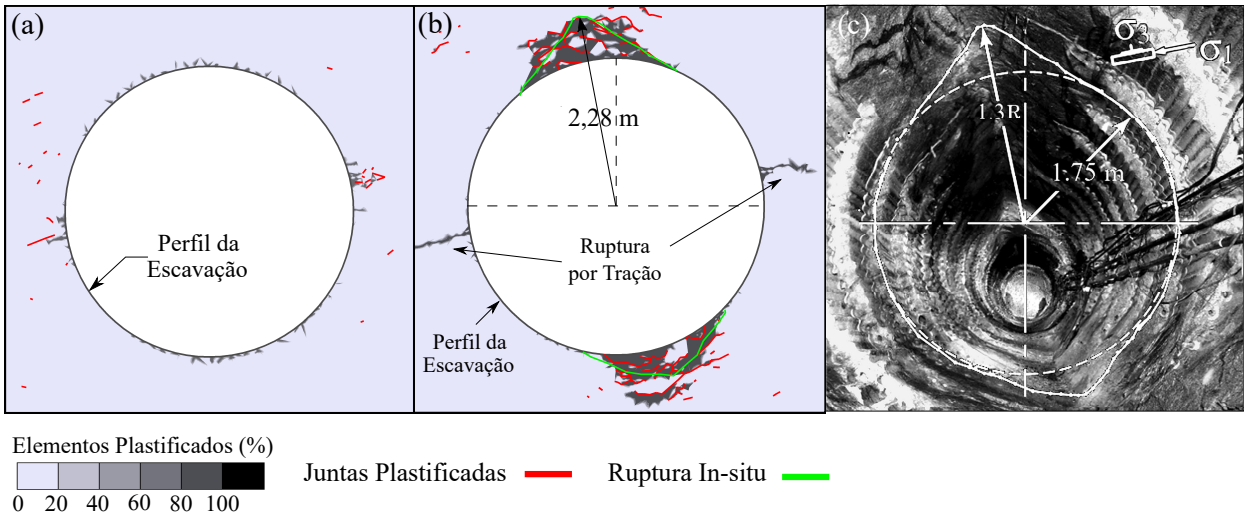


Fig. 3: (a) Modelo CVBM com resistência *in situ* equivalente a resistência de laboratório, (b) modelo CVBM calibrado para o túnel Mine-By e (c) ruptura registrada em campo [4]

Os parâmetros ajustados para o modelo de campo foram reempregados no modelo de laboratório do ensaio de compressão simples. A redução da coesão implicou numa redução da resistência da amostra para 64,3 MPa. Tal resultado equivale a uma resistência *in situ* aproximadamente 30% da resistência laboratorial, valor compatível com o intervalo de redução proposto por [4].

4 CONCLUSÕES

A partir desse estudo verificou-se que o modelo CVBM é capaz de representar o processo de ruptura de escavações em rochas intactas sujeitas a elevadas tensões de campo. O modelo se mostrou robusto e conseguiu capturar diferentes aspectos da ruptura frágil como a divergência ocorrida entre a resistência registrada em campo e em laboratório, a profundidade e o formato da ruptura por *spalling* e também a ruptura por tração nas paredes da escavação.

Embora o modelo seja baseado em uma formulação FEM, a adoção do mosaico Voronoi permite observar de forma explícita a formação das macrofraturas a partir da plastificação dos elementos de junta, comportamento que não pode ser capturado com outros modelos baseados no FEM como o Damage Iniciation Spalling Limit, por exemplo.

A representação do processo de fraturamento de forma explícita possibilita uma simulação mais verosímil da rocha deteriorada ao redor da escavação, o que evidencia a potencialidade do CVBM. Por fim, destaca-se que a retroanálise da ruptura do túnel Mine-By apresentada neste trabalho confirma a aplicabilidade do CVBM como uma nova ferramenta para previsão de ruptura frágil de escavação com elevadas tensões de campo.

5 *Agradecimento*

Este trabalho foi financiado pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico - CNPq e por Furnas Centrais Elétricas. Os autores agradecem esse incentivo.

Referências

[1] M. Cai, P. Kaiser, and C. Martin. Quantification of rock mass damage in underground excavations from microseismic event monitoring. *International Journal of Rock Mechanics and Mining Sciences*, 38(8):1135 – 1145, 2001.

[2] B. Damjanac, M. Board, M. Lin, D. Kicker, and J. Leem. Mechanical degradation of emplacement drifts at yucca mountain—a modeling case study: Part ii: Lithophysal rock. *International Journal of Rock Mechanics and Mining Sciences*, 44(3):368 – 399, 2007.

[3] J. Day, M. Diederichs, and D. Hutchinson. Composite geological strength index approach with application to hydrothermal vein networks and other intrablock structures in complex rockmasses. *Geotechnical and Geological Engineering*, 06 2019.

[4] M. S. Diederichs. The 2003 canadian geotechnical colloquium: Mechanistic interpretation and practical application of damage and spalling prediction criteria for deep tunnelling. *Canadian Geotechnical Journal*, 44(9):1082–1116, 2007.

[5] C. Edelbro. Numerical modelling of observed fallouts in hard rock masses using an instantaneous cohesion-softening friction-hardening model. *Tunnelling and Underground Space Technology*, 24(4):398 – 409, 2009.

[6] E. Ghazvinian, M. Diederichs, and R. Quey. 3d random voronoi grain-based models for simulation of brittle rock damage and fabric-guided micro-fracturing. *Journal of Rock Mechanics and Geotechnical Engineering*, 6(6):506 – 521, 2014.

[7] R. E. Goodman, R. Taylor, and Brekke. A model for the mechanics of jointed rock. *Journal of the Soil Mechanics and Foundations Division*, page 23, 1968.

[8] V. Hajiabdolmajid, P. K. Kaiser, and C. D. Martin. Modelling brittle failure of rock. *International Journal of Rock Mechanics and Mining Sciences*, 39(6):731–741, 2002.

[9] P. Kaiser. Ground support for constructability of deep underground excavationsechallenges of managing highly stressed ground in civil and mining projects. pages 1–36, 2016.

[10] C. Martin. The strength of massive lac du bonnet granite around underground openings. 1993.

[11] C. Martini, R. Read, and J. Martino. Observations of brittle failure around a circular test tunnel. *International Journal of Rock Mechanics and Mining Sciences*, 34(7):1065 – 1073, 1997.

[12] L. L. Rasmussen and M. M. de Farias. Lattice modelling of gravity and stress-driven failures of rock tunnels. *Computers and Geotechnics*, 116:103183, 2019.

[13] R. Read. 20 years of excavation response studies at aecl's underground research laboratory. *International Journal of Rock Mechanics and Mining Sciences*, 41(8):1251 – 1275, 2004. Rock Mechanics Results from the Underground Research Laboratory, Canada.

[14] Rocscience. Rs2 version 10.0. `https://www.rocscience.com/software/rs2`, 2019.

[15] S. Sinha and G. Walton. A study on bonded block model (bbm) complexity for simulation of laboratory-scale stress-strain behavior in granitic rocks. *Computers and Geotechnics*, 118:103363, 2020.

[16] I. Vazaios, N. Vlachopoulos, and M. Diederichs. Assessing fracturing mechanisms and evolution of excavation damaged zone of tunnels in interlocked rock masses at high stresses using a finite-discrete element approach. *Journal of Rock Mechanics and Geotechnical Engineering*, 11(4):701 – 722, 2019.

[17] N. Vlachopoulos and M. Diederichs. Appropriate uses and practical limitations of 2d numerical analysis of tunnels and tunnel support response. *Geotechnical and Geological Engineering*, 32, 04 2014.

[18] N. Vlachopoulos and I. Vazaios. The numerical simulation of hard rocks for tunnelling purposes at great depths: A comparison between the hybrid fdem method and continuous techniques. pages 2–45, 07 2018.

[19] N. Vlachopoulos and I. Vazaios. The numerical simulation of hard rocks for tunnelling purposes at great depths: A comparison between the hybrid fdem method and continuous techniques. pages 2–45, 07 2019.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Analysis of informative priors' effects on epidemic curve fitting

Felipe Fontinele Nunes[1], João Pedro Valeriano Miranda[2], Pedro Henrique Pinheiro Cintra[2], Igor Reis[2], Lorena Reis de Lima[2] and Tábata Luiza de Sousa Alves[2]

[1] *Department of Physics, University of Alberta, Edmonton, Canada*
[2] *Instituto de Física, Universidade de Brasília, DF, Brazil*

Abstract
In order to verify the effects of using empirical priors to fit epidemiological models through the ABC-SMC method, we considered the SEIRD model with different prior distributions combinations, admitting experimental measurements or not. Comparing the results of these combinations, we observe that the fit with the minimal RMSD final value was the one using all available empirical distributions as priors. However, the values of the parameters obtained do not belong, in general, to the 95% CI of the empirical distributions. We further discuss possible reasons for this deviation.

Keywords:COVID-19, SEIRD, Empirical Priors, ABC-SMC

1 INTRODUCTION

The year 2020 was marked by the tragedy of the pandemic caused by COVID-19, mobilizing scientists, healthcare workers and authorities in the fight against the spread of its etiological agent, the SARS-CoV-2 virus.

One of the tools scientists have been using are the so called epidemiological models. These models are used for simulation and forecasting of pandemic scenarios [5, 8, 11], data analysis [4] and also to investigate the economic impact of the pandemic [17], among others. Based on the analysis provided by the models, decisions can be made by the authorities aiming to avoid the worst scenarios, also allowing scientists to quantitatively investigate the effect of specific non-pharmaceutical policies.

In the midst of these studies, the National Laboratory for Scientific Computing (LNCC) provides scientific and computational apparatus (SDumont supercomputer) to several

Contatcs: feradofogo@hotmail.com (Felipe Fontinele) pedrohpc96@hotmail.com (Pedro Cintra), joaopedrovm98@gmail.com (João Valeriano)

studies – such as ours – for comparing models to describe the pandemic[1]. Among the methods used to study infectious diseases, compartmental models represent a valuable contribution. In such cases, the population is assigned to compartments, each of them ruled by their respective differential equations, representing the possible states that an individual can be. These equations may involve as many parameters as needed for the model being considered. For example, one of the simplest and most common is the SIR model, where S stands for Susceptible, I for Infected and R for Recovered. It requires some parameters, such as the infection rate, recovery rate and the total population. If more compartments are required, the interaction between the populations in each of the groups has to be taken into account in the respective differential equations.

Considering the dynamics of the COVID-19 disease, our goal is to use the SEIRD (Susceptible - Exposed - Infected - Recovered - Dead) model, taking into account seven parameters, namely: incubation period, infection fatality rate, time from symptoms onset until death or recovery of infected individuals, infection fatality ratio, infection rate by symptomatic patients, and infection rate by presymptomatic patients (those who have been exposed to the virus, but do not show symptoms yet, that is, those in the incubation period). For a complete description, see Table 1.

When predicting the development of a pandemic, it is crucial to compare the results of the model with the real world data in order to validate their use. With this purpose, we analyze the root-mean-square deviation (RMSD), from the fitted curve to the data, aiming to obtain an estimate of how close to the real world the parameters of our model can be adjusted.

There is a large quantity of methods to fit data to a model, ranging from deterministic minimization algorithms, such as non-linear least squares [3, 15], to stochastic processes [7, 2]. In this work, we choose to use the ABC-SMC, (Approximate Bayesian Computing with Sequential Monte Carlo Sampling) method [10].

Just as common deterministic minimization algorithms depend on the initial guess for the fitted parameters [14, 12], in the ABC-SMC method, the prior distribution for a given parameter should result in better or worse outcomes. Usually, in the absence of prior knowledge about the probability distribution of the parameters, an uniform distribution bounded by a certain interval is used as prior. However, the choice of the intervals can affect significantly the fitting algorithm. If the interval is too restrictive, the sampled values could never be close to the ones that best fit our model, or it could be that the interval is excessively large so that the algorithm has problems to reduce the parameter space to the true interval. Several studies have been dedicated to addressing this problem, from extending the confidence interval [6] to methods for estimating better priors that represent the ignorance on the correct prior [13]. In this fashion, we also aim to compare adjustments using priors obtained from experimental studies from hospitals and tracking of infections from China.

2 METHODOLOGY

To perform the benchmark analysis for the effectiveness of using the experimental priors in comparison to the uniform distributions, we need to choose a reliable dataset, that

[1] *Click-Covid: Uma ferramenta de informação* at SDumont COVID-19 Projects.

can represent the time-series of the real epidemiological data. This requirement arises because, if the dataset comes from suitable collected and cataloged data, one can expect the model's parameters to correspond to data measured in other studies, which is needed if we want to use experimental distributions as priors. For example, if the dataset was created with problems in the identification of new cases of infection, the incubation period value that governs the differential equations of the model can be very different of the real value. The dataset we choose to analyze – which seemingly contains all these requirements – is the epidemiological COVID-19 data from China.

We performed a search on google scholar by keywords regarding the key parameters, such as *incubation period COVID-19*, *epidemiological parameters COVID-19*. Some other papers were previously known by the researches, and therefore they were not necessarily found by the keywords search. After that, the distribution for each parameter was taken according to the distribution function described by the corresponding article. The experimental distribution parameters were described by one of the following two equations (Lognormal, Gamma)

$$\text{Lognormal}(x; p_1, p_2) = \frac{1}{x p_2 \sqrt{2\pi}} \exp\left(- \frac{(\ln(x) - p_1)^2}{2 p_2^2} \right), \tag{1}$$

$$\text{Gamma}(x; p_1, p_2) = \frac{p_1^{p_2}}{\Gamma(p_2)} x^{p_2 - 1} \exp\left(-p_1 x\right). \tag{2}$$

Distributions were obtained for more countries than China, as a point of reference for confidence since some of these epidemiological parameters, such as the incubation period, are not expected to drastically change from country to country. Although parameters associated with cultural characteristics, such as time from symptoms onset to hospital admission, are expected to present variations, we use them as qualitative metrics for comparison in the following way: if some country presents an average time taken from symptoms onset to hospitalization in 5 days, and another country presents an average for the same time interval of 30 days, however both countries present similar means for the time between symptoms onset to death of, say 15 days, we conclude something must not be correct in the measurements for the later and discard this measurement.

2.1 Mathematical Model

Figure 1 shows a flowchart of the SEIRD compartmental model, where we have equations assigned to each arrow, describing the rate in which people are moved from one compartment to another, in the direction given by the arrows. Beside the flowchart, we have the SEIRD differential equations system written explicitly. Notice that $\mu = P_{IFR}/\tau_d$ and $\gamma = (1 - P_{IRF})/\tau_r$, where P_{IFR}, τ_d and τ_r denote respectively the Infection Fatality Rate, the time period from symptoms onset to death and the time period from symptoms onset to recovery.

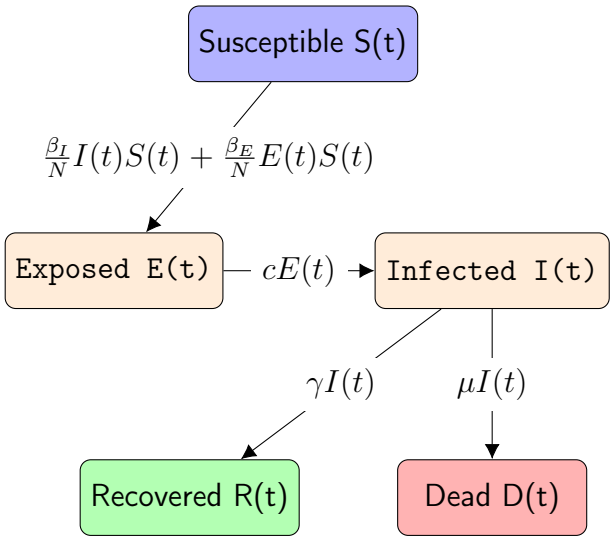


Figure 1: Flowchart of the SEIRD model. Each individual is moved from one compartment to another by a rate given by the equation assigned to the arrow between compartments.

$$\frac{dS}{dt} = -\frac{\beta_I}{N}IS - \frac{\beta_E}{N}ES \tag{3}$$

$$\frac{dE}{dt} = \frac{\beta_I}{N}IS + \frac{\beta_E}{N}ES - cE \tag{4}$$

$$\frac{dI}{dt} = cE - \gamma I - \mu I \tag{5}$$

$$\frac{dR}{dt} = \gamma I \tag{6}$$

$$\frac{dD}{dt} = \mu I \tag{7}$$

All differential equations (3)-(7) are subject to initial conditions at time $t = 0$. We assume that $R(t = 0) = E(t = 0) = 0$, $S(t = 0) = N - I(t = 0) - D(t = 0)$, with $I(t = 0) = 548$ and $D(t = 0) = 17$[2].

Physical meaning of all parameters is listed in Table 1. Some of these parameters appear as frequency distributions, while others appear as fixed values. Here, we are interested in parameters associated with distributions.

2.2 ABC-SMC

With the definition of the model, using all the required parameters, we proceed to the task of finding the optimal parameters for a given epidemiological curve. We rely on the fitting method ABC-SMC (Approximate Bayesian Computation Sequential Monte Carlo) to fulfill this task. The algorithm behind the method receives as input the prior distribution for each parameter to be estimated, the data points, a tolerance value ϵ, the number of samples to be generated from the prior distribution and the number of repetitions. By the end of the process we shall have a new distribution, called posterior, of the sampled values that were accepted given a certain tolerance, or in other words, the RMSD calculated with these parameters was smaller or equal than the tolerance (Figure 2).

The posterior distribution is expected to be more precise regarding its most probable value, then the second part of the algorithm comes in, where new sorting sections begin. It starts by using the posterior generated in the first part as its new prior. In each

[2]Numbers based on Chinese data on infections on January 22.

iteration, the algorithm calculates a new tolerance value, making it smaller each time a new posterior is created. With this process, the posteriors generated by the algorithm becomes narrower, and thus the most probable value that fits the model becomes more apparent. The effect of the decrease in the value of the tolerance can be viewed as a decrease in the range of the parameter space for each of the parameters: as the tolerance gets smaller, only values in a more restrict range can be accepted within that tolerance. After calculating the last posterior, we choose from it the parameter set that minimizes the RMSD.

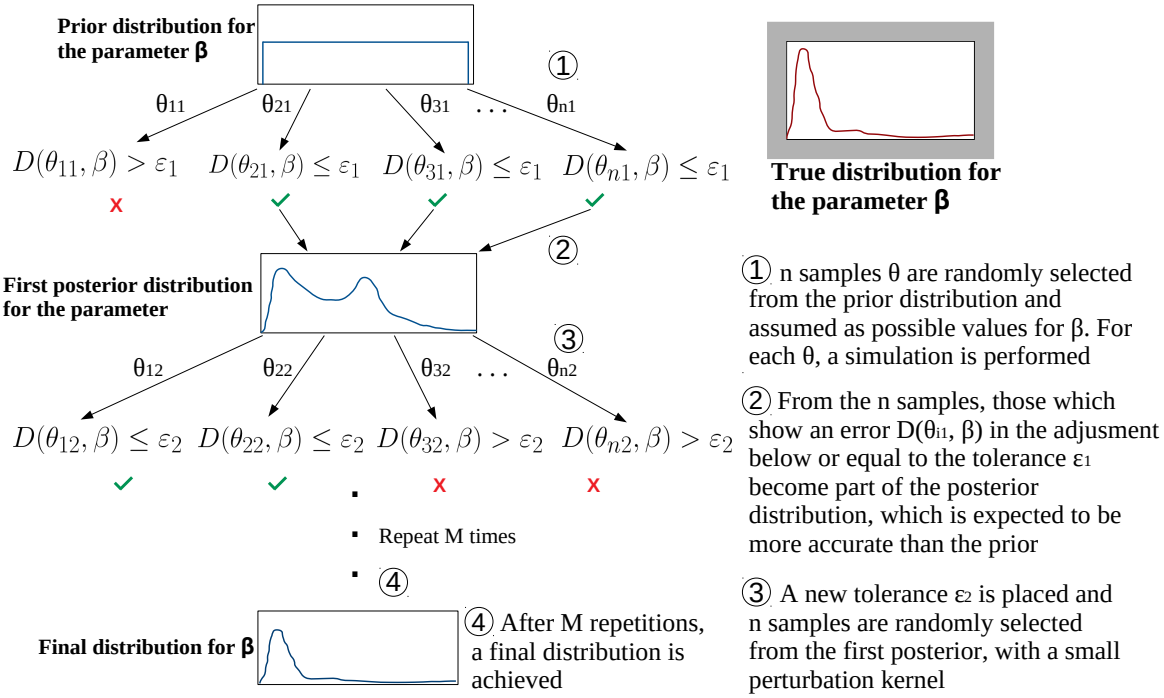


Figure 2: Illustration on the algorithm behind ABC-SMC method. Based on the original illustration in [16].

The epidemiological data was taken from a Kaggle dataset named COVID-19 Dataset[3]. It contains worldwide information about COVID-19 data of confirmed, death and recovered cases grouped by levels that goes from country data to province data. It is a complete dataset, with information that can be used in a great range of studies and researches. There is an open GitHub repository[4] with all the code needed to reproduce the results presented here.

[3] imdevskp/corona-virus-report
[4] LNCC-COVID-19-prediction/EAMC_2021_paper

Parameter	Interpretation	Experimental Prior
β_I	Infection rate: Probability of disease transmission times the average number of contacts per person per time.	–
β_E	Infection rate by Exposed people.	–
N	Effective population that could be infected by the disease.	–
$c^{-1} = \tau_c$	Incubation period.	Lognormal$(x; 1.57, 0.65)$[1].
τ_r	Time since the onset of the symptoms until recovery.	Gamma$(x; 6.68, 0.33)$[18].
τ_d	Time since the onset of the symptoms until death.	Lognormal$(x; 2.84, 0.58)$[9].
P_{IFR}	Infection Fatality Ratio, probability of dying after becoming infected.	–

Table 1: Parameter definitions for the SEIRD model. The parameters without a distribution description in the last column do not have empirical priors, or if they do, the priors were not used.

3 RESULTS AND DISCUSSION

We are interested in finding out the effect of using empirical priors in the fit of the SEIRD model to epidemiological data. However, we need a first result to compare with the fits we get by using these priors. We begin by fitting our dataset using uniform distributions as priors with 20 ABC-SMC iterations. The results are satisfying, with a final RMSD ≈ 155 as one can see in Figure 3, indicating that our model and fitting procedure indeed represents, to a certain degree, the data behaviour.

Proceeding with the investigation of the effects of considering information from the empirical data to fit our model, we tested four different combinations of prior distributions: **(I)** using the empirical prior for τ_r, while keeping the other priors uniform; **(II)** using the τ_r distribution and fixing τ_c and τ_d on the mean value on their empirical distributions; **(III)** using τ_c and τ_d fixed on the mean, now with an uniform prior for τ_r; and **(IV)** using empirical priors for all three key parameters τ_c, τ_r and τ_d.

Figure 4 shows RMSD values through each iteration[5] of the ABC-SMC fitting process, where the shaded area represents the uncertainty, given by the standard deviation of the RMSD calculated in 20 different runs of the ABC-SMC algorithm. Since this is a stochastic method, the averaging procedure is mandatory to obtain consistent and robust results. The optimal parameters were selected from the posterior distribution as being those which minimize the RMSD .

The most successful fit, regarding the minimization of the RMSD, is the one using the combination **(IV)**, a result we can use to advocate in favour of using empirical priors, specially noticing that the second best result is given by the combination involving only the empirical prior of τ_r. However, it is important to note that the observed difference

[5]Here, iteration denotes the consecutive posterior calculated through ABC-SMC.

between the obtained final RMSDs can be considered rather negligible if we take into account the scale of the fitted data.

An interesting result from Figure 4 is that setting parameters on fixed values may limit the quality of the obtained fit, which may seem counterintuitive as doing it not only reduces the number of parameters to be fitted, but also sets the unknown parameter value to its actual empirical measurement. From this, we realize that the parameters that best fit the available data do not correspond to their measured values. This can be due to a variety of factors. Firstly, we are considering a very simplified model, which does not take into account hospitalizations or countermeasures like the use of masks and social isolation. Secondly, there can be problems in the data being fitted, most notably possible underreporting, mainly of confirmed cases.

Despite this, considering the best set of prior distributions (by RMSD-based criterion), the use of empirical priors seems to push the parameters in the right way. Table 2 shows the comparison between the empirical parameter values (with 95% CI, obtained from references mentioned in Table 1) and average with standard deviation results from the 20 runs of ABC-SMC considering the experimental priors.

	τ_c	τ_r	τ_d
Empirical average parameter	$5.95(4.94 - 7.11)$	$24.7(22.9 - 28.1)$	$20.2(15.1 - 29.5)$
ABC-SMC's best parameter	8.47 ± 0.17	18 ± 2	31 ± 3

Table 2: Comparison between the empirical data and the fit results, using combination **(IV)**, for the three period parameters of our model.

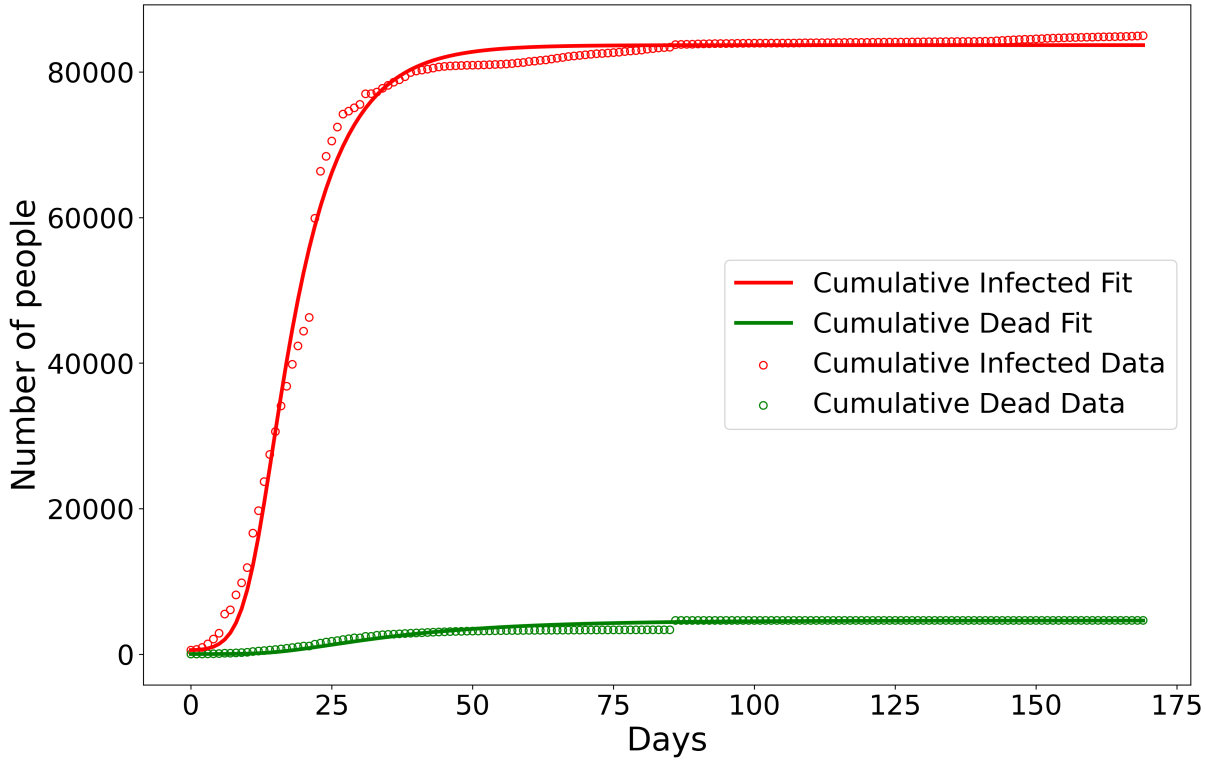


Figure 3: SEIRD model fit for China's cumulative infected and dead data using the empirical priors for τ_r, τ_d and τ_c.

Since the different prior combinations do not converge to a same value of RMSD, it is expected that the parameters set found by each model is not the same. This is shown in Figure 5, which presents the evolution of all parameters values through the iterations of ABC-SMC, again with the uncertainty given by the standard deviation of the results of 20 runs of the algorithm.

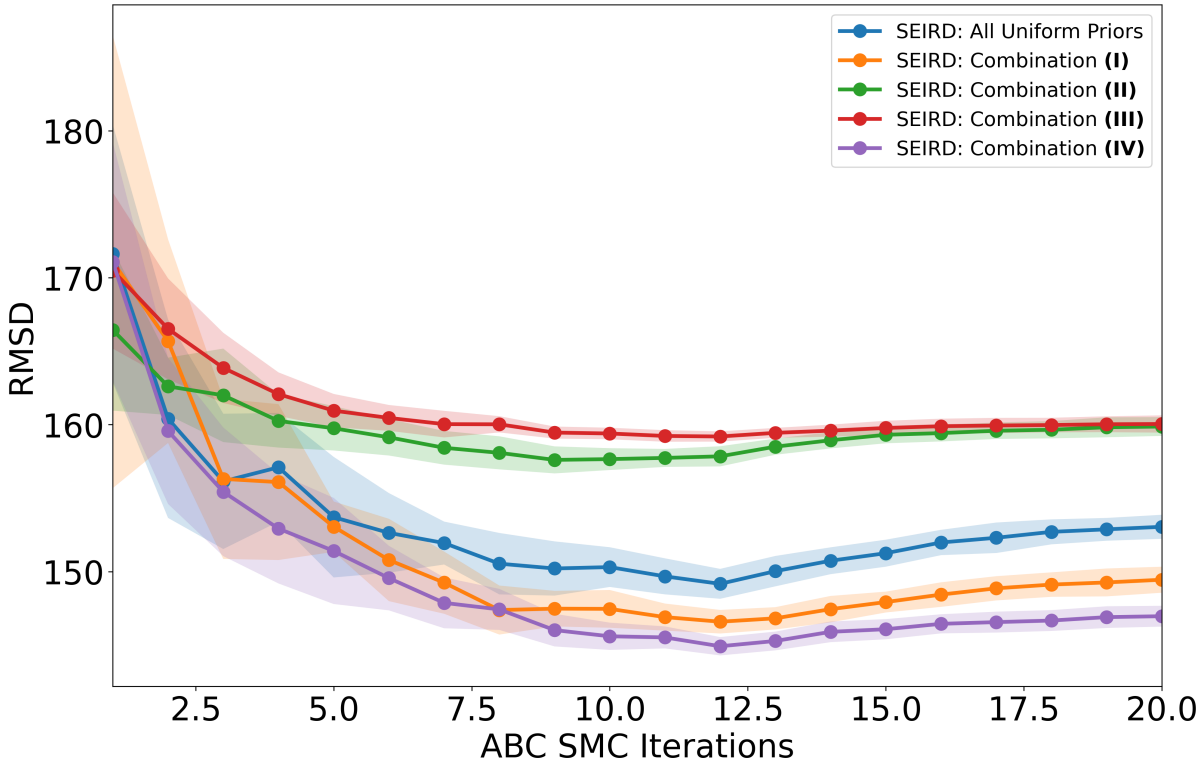


Figure 4: Evolution of RMSD, calculated for each posterior generated by the ABC-SMC algorithm, with each color assigned to a different combination of prior distributions for the model parameters.

The incubation period τ_c found in the fitting procedure is clearly different from the one given by the empirical data, as it can be seen in Figure 5 and Table 2. This can be thought of as a compensation made by the fitting algorithm to take into account the fact that some of the parameters of our model, as β_I and β_E, should in actually change in time, as intervention and social measures are adopted by the population to try to prevent the spread of coronavirus, and this change in the parameters would modify the rate at which the infection curve increases. Also, the delay on the infection and death numbers notifications can play a role on the deviation of the parameters found and the ones empirically measured, since this delay could change the real shape of the epidemiological data curve.

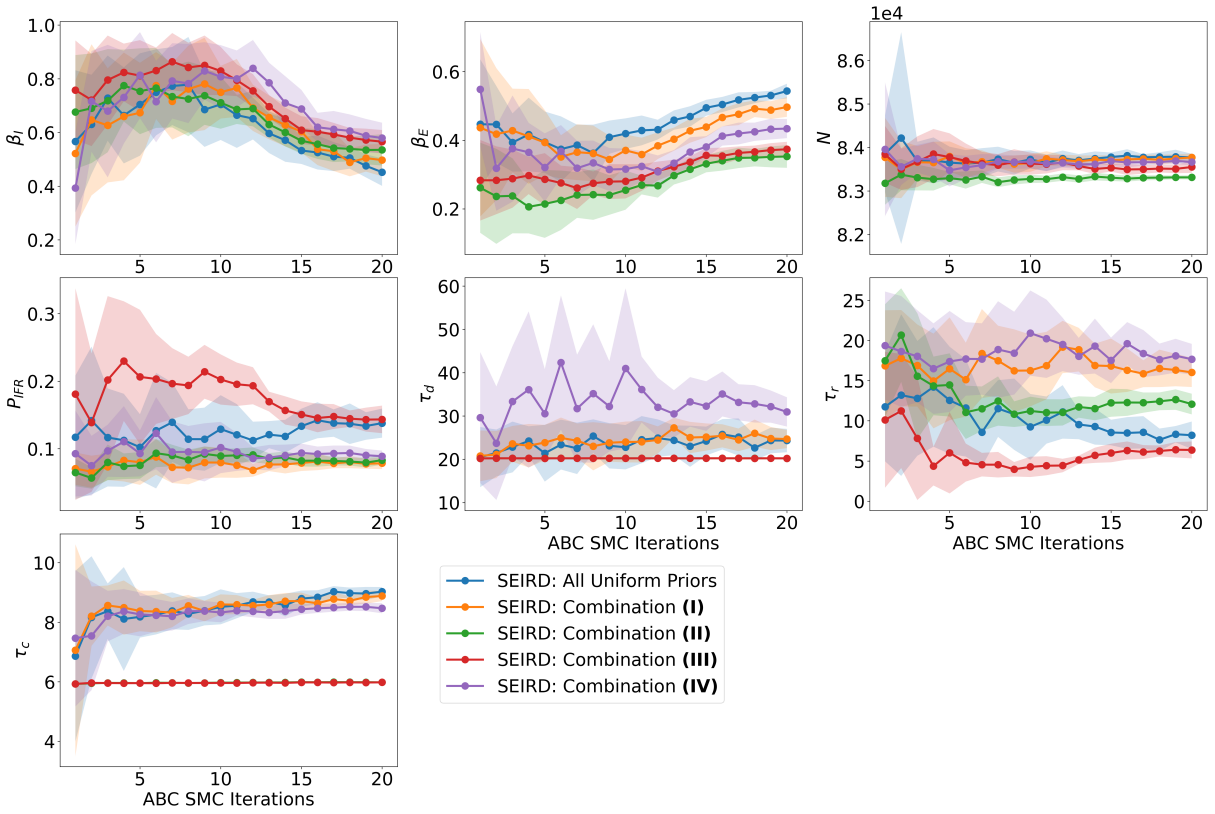


Figure 5: Evolution of the model parameters, calculated for each posterior generated by the ABC-SMC algorithm, with each color assigned to a different combination of prior distributions.

Notice that there is an interplay between the values of β_I and β_E, with the first one decreasing while the latter increases throughout the ABC-SMC iterations, but we cannot see a really appreciable result of this exchange on the RMSD values in Figure 4, which means that there probably exists some correlation between these two parameters. Calculating the Pearson Correlation Coefficient between each pair of β_I and β_E curves in Figure 5, we obtain a correlation coefficient of -0.89 ± 0.07[6].

4 CONCLUSIONS

We conclude that the model and the fitting method have shown satisfactory results, since the data were well represented by the fitted curve. However, there was no significant difference between the fits calculated using the different combinations of prior distributions, which means that different parameter sets can represent the epidemiological data. This can be viewed as an example of the influence from the correlation found between β_I and β_E and fact that $\beta_{I,E}$ are in fact time dependent, and thus influence the shape of the curve in a different way of the one described in our model, and also to the delay in the notification of new cases.

This implies that the real parameters that describe the data are very difficult to find, but this does not represent a drawback regarding fitting our data, and possibly making

[6]See GitHub repository for additional information on correlation between the epidemiological parameters.

predictions and other analysis.

5 *Acknowledgements*

The authors would like to thank Gustavo Barbosa Libotte, Renato Simões Silva and Sandra Mara Cardoso Malta for all the discussions and guidance on the development of this work. The authors would also like to thank the LNCC for granting the computational resources at the SDumont supercomputer that were crucial to perform the simulations for this project.

References

[1] Q. Bi, Y. Wu, S. Mei, C. Ye, X. Zou, Z. Zhang, X. Liu, L. Wei, S. A. Truelove, T. Zhang, et al. Epidemiology and transmission of covid-19 in 391 cases and 1286 of their close contacts in shenzhen, china: a retrospective cohort study. *The Lancet Infectious Diseases*, 2020.

[2] K. Chatterjee, K. Chatterjee, A. Kumar, and S. Shankar. Healthcare impact of covid-19 epidemic in india: A stochastic mathematical model. *Medical Journal Armed Forces India*, 2020.

[3] Y.-C. Chen, P.-E. Lu, and C.-S. Chang. A time-dependent sir model for covid-19. *arXiv preprint arXiv:2003.00122*, 2020.

[4] J. Dehning, J. Zierenberg, F. P. Spitzner, M. Wibral, J. P. Neto, M. Wilczek, and V. Priesemann. Inferring change points in the spread of covid-19 reveals the effectiveness of interventions. *Science*, 2020.

[5] J. Fernández-Villaverde and C. I. Jones. Estimating and simulating a sird model of covid-19 for many countries, states, and cities. Technical report, National Bureau of Economic Research, 2020.

[6] S. Greenland. Interval estimation by simulation as an alternative to and extension of confidence intervals. *International Journal of Epidemiology*, 33(6):1389–1397, 2004.

[7] S. He, S. Tang, and L. Rong. A discrete stochastic model of the covid-19 outbreak: Forecast and control. *Math. Biosci. Eng*, 17:2792–2804, 2020.

[8] K. Iwata and C. Miyakoshi. A simulation on potential secondary spread of novel coronavirus in an exported country using a stochastic epidemic seir model. *Journal of clinical medicine*, 9(4):944, 2020.

[9] N. M. Linton, T. Kobayashi, Y. Yang, K. Hayashi, A. R. Akhmetzhanov, S.-m. Jung, B. Yuan, R. Kinoshita, and H. Nishiura. Incubation period and other epidemiological characteristics of 2019 novel coronavirus infections with right truncation: a statistical analysis of publicly available case data. *Journal of clinical medicine*, 9(2):538, 2020.

[10] A. Minter and R. Retkute. Approximate bayesian computation for infectious disease modelling. *Epidemics*, 29:100368, 2019.

[11] S. M. Moghadas, A. Shoukat, M. C. Fitzpatrick, C. R. Wells, P. Sah, A. Pandey, J. D. Sachs, Z. Wang, L. A. Meyers, B. H. Singer, et al. Projecting hospital utilization during the covid-19 outbreaks in the united states. *Proceedings of the National Academy of Sciences*, 117(16):9122–9126, 2020.

[12] T. Ojika and Y. Kasue. Initial-value adjusting method for the solution of nonlinear multipoint boundary-value problems. *Journal of Mathematical Analysis and Applications*, 69(2):359–371, 1979.

[13] L. R. Pericchi and P. Walley. Robust bayesian credible intervals and prior ignorance. *International Statistical Review/Revue Internationale de Statistique*, pages 1–23, 1991.

[14] F. Rabier, E. Klinker, P. Courtier, and A. Hollingsworth. Sensitivity of forecast errors to initial conditions. *Quarterly Journal of the Royal Meteorological Society*, 122(529):121–150, 1996.

[15] D. Rafiq, S. A. Suhail, and M. A. Bazaz. Evaluation and prediction of covid-19 in india: A case study of worst hit states. *Chaos, Solitons & Fractals*, 139:110014, 2020.

[16] M. Sunnåker, A. G. Busetto, E. Numminen, J. Corander, M. Foll, and C. Dessimoz. Approximate bayesian computation. *PLoS Comput Biol*, 9(1):e1002803, 2013.

[17] A. A. Toda. Susceptible-infected-recovered (sir) dynamics of covid-19 and economic impact. *arXiv preprint arXiv:2003.11221*, 2020.

[18] R. Verity, L. C. Okell, I. Dorigatti, P. Winskill, C. Whittaker, N. Imai, G. Cuomo-Dannenburg, H. Thompson, P. G. Walker, H. Fu, et al. Estimates of the severity of coronavirus disease 2019: a model-based analysis. *The Lancet infectious diseases*, 2020.

Detecção e Classificação de Bots Utilizando Redes Neurais Artificiais e Análise de Sentimentos

Gabrieli Silva[1], Eliaquim Ramos[1], Eric Araújo[2], Fábio Borges[1] e Mariza Ferro[1]

[1] *Laboratório Nacional de Computação Científica, Petrópolis/RJ, Brasil*
[2] *Universidade Federal de Lavras, Lavras/MG, Brasil*

Abstract

Autonomous entities known as social bots generate an increasing amount of social media content. Not all of those bots are harmful. Lots of them are used for good purposes indeed, like helping users resolve their questions without the need of a human being. However, there is an increasing number of malicious bots that aim to persuade, slander, or even fool humans. Nowadays there are more than 330 million monthly active Twitter accounts throughout the world and the exact amount of bot accounts is unknown. Different techniques in Machine Learning have been explored for bots detection in social networks. Therefore, the aims of this work are: *i)* detect bot accounts on Twitter utilizing the supervised ML by means of an Artificial Neural Network (phase 1); *ii)* classify the dangerousness of the bot based on what sentiments are involved in the contents published by those accounts, through the use of Sentiment Analysis technique (phase 2). In phase 1, of the 594 users evaluated, 242 were considered bot with an accuracy of 82%. In phase 2 of the 22,855 tweets analyzed, 14.8% had a negative feeling.

Keywords:Social Bot, Artificial Neural Network, Sentimental Analysis

1 Introdução

As mídias sociais têm se tornado cada vez mais populares e acessíveis por facilitar a vida de muitas pessoas e instituições. Uma das grandes vantagens que elas oferecem é a comunicação instantânea, possibilitando compartilhar informações e acontecimentos do mundo em tempo real. Segundo [29], no Brasil, 96,2% dos usuários de internet estão em alguma mídia social, e esse crescimento não dá sinais de que vai desacelerar.

Uma quantidade crescente de conteúdo advindo das mídias sociais (também popularmente chamadas de redes sociais, usado como termo intercambiável ao longo deste

Contato: Gabrieli Silva, gabrieli@lncc.br

trabalho) é gerada por entidades autônomas conhecidas como bots ou robôs sociais [36]. Com o uso da Inteligência Artificial (IA) para gerar conteúdo automaticamente, essas entidades podem imitar o comportamento de um usuário real e se passar por seres humanos [10]. Muitos robôs sociais desempenham funções úteis como, por exemplo, facilitam o atendimento aos usuários de várias empresas na resolução de problemas ou compra de produtos, sem a necessidade de um atendente real. Porém, há um registro crescente de contas bot com intenções maliciosas, as quais possuem o objetivo de persuadir, difamar ou enganar os humanos [13]. Segundo [27], contas bot são agentes que possuem alta influência na propagação de notícias falsas *(fake news)*, sobretudo, na rede social do Twitter.

Atualmente, no Twitter, são mais de 330 milhões de usuários ativos mensalmente no mundo todo, e a quantidade exata de bots é desconhecida [34]. O Twitter afirma que contas falsas ou *spammers* [4] representam apenas 5% dos seus usuários ativos. No entanto, pesquisadores como [20] e [31] acreditam que o número real alcança percentuais mais elevados. Portanto, a necessidade de estratégias para a detecção de contas bot no Twitter é essencial para garantir a credibilidade e a segurança daqueles que usam essa plataforma de mídia social como fonte de informação.

O Aprendizado de Máquina (AM) é uma subárea da IA dedicada ao desenvolvimento de programas de computador que melhoram automaticamente com a experiência [22]. Diferentes técnicas de AM vêm sendo exploradas na detecção de bots em mídias sociais.

O objetivo deste trabalho é detectar contas bot no Twitter e verificar a periculosidade dos bots encontrados, analisando qual o sentimento envolvido nos *tweets* publicados por essas contas. Para isso, foram utilizadas duas diferentes técnicas de AM: para a fase de detecção de contas bot (primeira fase) é utilizado o aprendizado supervisionado com a implementação de uma RNA. Na segunda fase, para avaliar a periculosidade, é utilizado a técnica de Análise de Sentimentos, a qual permite analisar conteúdos textuais combinando o Processamento de Linguagem Natural (NLP) com técnicas de AM.

2 Trabalhos Relacionados

Muitos robôs sociais desempenham funções úteis, como aqueles que automatizam o compartilhamento de notícias da imprensa e os que ajudam consumidores em atendimentos virtuais. Por outro lado, existem os bots maliciosos que espalham spam ou conteúdo malicioso a fim de persuadir, difamar e enganar os humanos, principalmente com a propagação de notícias falsas. Diante deste cenário, trabalhos, tais como [12], [6], [31], [1], [7] e [33], foram desenvolvidos para a detecção de contas bot nas redes sociais. Alguns destes trabalhos são discutidos a seguir, juntamente com suas vantagens, limitações e técnicas de AM utilizadas para o seu desenvolvimento. Ainda, são apresentados trabalhos que fazem uso da técnica de Análise de Sentimentos.

No trabalho de [6], para ajudar usuários humanos a identificar com quem estão interagindo, as contas da mídia social Twitter são classificadas em humanas, bots ou ciborgues (robôs híbridos, operados parte por humanos, parte por computadores [28]). Para isso os autores propõem um sistema classificador usando as técnicas de *Random Forest* [8] e *Naïve Bayes* [18]. Porém, após a detecção de quais contas pertencem a cada grupo, os autores não avaliam o conteúdo publicado por contas bot, como neste trabalho, a fim de verificar qual o sentimento atrelado a cada publicação e o tipo de impacto que o bot poderá trazer para a conta humano.

Em [1] são apresentadas as estratégias para a detecção de *botnets*, as quais se referem à contas bot gerenciadas por um indivíduo ou grupo [2], através da análise passiva do tráfego de rede. As estratégias são separadas em função dos protocolos de comunicação, comumente utilizados para implementação dos canais de comando e controle de *botnets*. Os autores fazem uso de um conjunto básico de ferramentas para o apoio ao processo de detecção. Porém, vale ressaltar, que nem sempre a alteração do fluxo de rede está relacionada a uma atividade maliciosa. Existem outros fatores que podem levar a este fim, como por exemplo, horários de pico que comumente alteram a estabilidade da rede.

Trabalhos como o de [12], [31] e [10] avaliam a influência de contas bot em problemas específicos. Em [12] é estudado as características e a atividade de bots em torno de grandes eventos políticos, incluindo eleições em diferentes países, tais como as eleições presidenciais e intermediárias de 2016 e 2018 nos EUA, e as eleições presidenciais francesas de 2017. [10] apresenta um estudo sobre as *fake news* na eleição presidencial de 2018, no Brasil, e o envolvimento de bots no compartilhamento dessas informações falsas.

Diversos trabalhos exploram a técnica de Análise de Sentimentos, tais como [7], [23], [11], [30] e [14]. Porém, nestes trabalhos as análises são realizadas em publicações relacionadas a um assunto (ou evento) especifico, como por exemplo, no trabalho de [7], o qual analisa o sentimento envolvido nos tweets relacionados aos filmes indicados à categoria de Oscar 2017, ou ainda em [9], o qual faz a análise dos *tweets* relacionados aos protestos que ocorreram no Brasil entre junho e agosto de 2013.

Portanto, ainda que diversos trabalhos realizem a detecção de contas bot do Twitter e a técnica de Análise de Sentimentos tenha sido utilizada, até o momento não foram encontrados trabalhos que, além de realizarem a detecção do tipo de conta, também avaliem a periculosidade dos bots encontrados, analisando qual o sentimento envolvido nos *tweets* publicados por essas contas. Ainda, neste trabalho, para a Análise de Sentimentos, os *tweets* são avaliados de uma maneira geral, sem levar em consideração qualquer evento ou assunto específico, o que caracteriza o principal diferencial deste trabalho.

3 Metodologia de experimentos e Resultados

Nesta seção são apresentados a metodologia de experimentos e os resultados obtidos nas duas diferentes fases deste trabalho. Lembrando que a primeira se refere à detecção das contas, classificando-as como bot ou não, com a implementação de uma RNA; enquanto a segunda, à análise da periculosidade, utilizando a técnica de Análise de Sentimentos.

3.1 Metodologia

Ao longo desta seção é apresentado o trabalho realizado nas etapas de coleta de dados, seleção de atributos e pré-processamento de dados. Também é detalhada a metodologia adotada durante as implementações computacionais para detectar contas bot e realizar a análise da periculosidade.

3.1.1 Coleta de dados

Por permitir o acesso a coleta de informações dos usuários através de sua API[1], *Twitter API Reference* [35], os dados analisados neste trabalho são da rede social Twitter. Esta

[1]API: Do termo em inglês Application Programming Interface

API está disponível para usuários desenvolvedores e pode ser acessada, após a análise do Twitter, com a autenticação básica da conta (usuário e senha) através de um *endpoint*.

Para as etapas de treinamento e teste da RNA foram utilizados os conjuntos de dados *Kaggle* [15] e *Bot Repository* [37], no formato atributo-valor. As contas disponíveis no conjunto *Kaggle* são do período de criação entre 2004 e 2017, enquanto que as do *Bot Repository*, foram criadas entre 2007 e 2018. Devido ao crescente número de contas bot, o Twitter vem desenvolvendo medidas para desativar os perfis considerados suspeitos, e por isso, para a análise da periculosidade, algumas contas poderiam não existir mais, inviabilizando a coleta. A fim de abranger contas mais recentes e garantir que os usuários classificados como bot na primeira fase ainda estejam disponíveis para a realização da segunda fase, neste trabalho a base de dados utilizada corresponde à combinação desses dois conjuntos de dados, os quais somados, possuem as rotulações de 2968 contas, indicando quais são bot (classe 1 com 1254 exemplos) ou não (classe 0 com 1714 exemplos). Na Tabela 1 são apresentados os atributos envolvidos na coleta de dados, o tipo, se categórico (C), numérico (N) ou data/hora (D/H), e o que cada um deles representam.

	Atributo	**Descrição**	**Tipo**
1*	Default_profile	Indica se o usuário alterou ou não o tema de seu perfil	C
2*	Followers_count	Número de seguidores que a conta possui	N
3*	Friends_count	Número de usuários que a conta está seguindo	N
4*	Listed_count	Número de listas públicas das quais este usuário é membro	N
5*	Favourites_count	Número de tweets que o usuário curtiu	N
6	Url	Uma URL fornecida pelo usuário em associação com seu perfil	C
7	Location	Local definido pelo usuário para o perfil da conta	C
8	Description	Breve resumo sobre a conta do usuário	C
9	Created_at	Data e hora em que a conta foi criada	D/H
10*	Verifield	Indica se o usuário possui uma conta verificada ou não	C
11	Screen_name	Nome pelo com o qual o usuário se identifica	C
12*	Default_profile_image	Indica se o usuário carregou uma imagem de perfil	C
13	Language	Idioma da conta	C
14*	Statuses_count	O número de tweets (incluindo retweets) emitidos pelo usuário	N
15	Name	Nome do usuário definido por ele mesmo	C

Tabela 1: Conjunto de atributos envolvidos na coleta de dados.

3.1.2 Seleção de atributos e pré-processamento de dados

Na detecção de bots em redes sociais, o atributo data de criação da conta é de suma importância na análise. Normalmente, contas bot possuem elevada movimentação de atividades em um curto intervalo de tempo desde a sua data de criação. Diversas implementações de algoritmos utilizados em AM não são capazes de analisar certos tipos de dado como, por exemplo, atributos do tipo data e hora. Desta forma, frequentemente, atributos com esses tipos de dados são transformados em um outro atributo com a mesma informação, mas com um tipo de dado que o algoritmo seja capaz de analisar [3]. A fim de contornar essa limitação, no conjunto de dados utilizado neste trabalho, o atributo data de criação da conta foi transformado para um atributo do tipo inteiro, o qual representa o número de dias corridos a partir de uma data fixa para informar a idade da conta.

Os atributos dos dados de entrada, normalmente, estão em intervalos de variação bastante distintos. Isto faz com que o treinamento e teste da RNA fiquem prejudicados, pois a rede neural pode interpretar valores mais altos como de maior importância e valores menores como menos importantes [19]. A fim de evitar este problema os dados foram

normalizados através da abordagem *MinMaxScaler* da biblioteca *scikit-learn* [24], o qual redimensiona os valores dos atributos no intervalo entre 0 e 1. Além disso, alguns atributos, tais como *verified*, *default_profile* e *default_profile_image*, encontram-se com valores nominais, o que torna seu processamento pela rede impossível de ser executado. Por isso, também foi utilizada a técnica de binarização, com a qual é possível atribuir números binários aos atributos categóricos. Desta forma, os campos com valor $True$ receberam valor 1 e os de valor $False$ receberam 0.

Os atributos apresentados na Tabela 1 foram analisados quanto ao seu tipo de dado. Aqueles do tipo categórico em que não era possível aplicar a binarização por apresentarem textos longos (frases), tais como *url, description, screen_name, language* e *name*, foram excluídos da análise. Além disso, foi eliminado o atributo *location* devido a alta porcentagem de dados faltantes (80%). Na Tabela 1, os atributos com um (*) foram os selecionados durante a fase de seleção de atributos e usados para a tarefa de classificação.

3.1.3 Implementações Computacionais

Para a fase de detecção de contas bot foi implementada uma RNA do tipo *Multilayer Perceptron (MLP)*, na qual os neurônios de uma camada anterior se conectam a todos os neurônios da camada seguinte, formando uma rede neural totalmente conectada [32].

Para a implementação da RNA foi utilizada a biblioteca Keras[5], por ser amplamente difundida na área e facilitar o desenvolvimento de novas aplicações de AM. A RNA desenvolvida neste trabalho por meio de testes experimentais possui quatro camadas: a de entrada com 10 neurônios; a primeira camada oculta com 8 neurônios; a segunda camada oculta com 3 neurônios; e a camada de saída com 1 neurônio. Segundo [25] a função de ativação Unidade Linear Retificada (ReLU) tem sido amplamente utilizada por facilitar o processo de treinamento. Portanto, nas três primeiras camadas foi utilizada a função de ativação ReLU. Além disso, para restringir a saída do neurônio ao valor de interesse, 0 (não bot) e 1 (bot), na camada de saída foi utilizada a função de ativação Sigmoide.

Os parâmetros da RNA foram ajustados de forma a minimizar a função de perda durante o treinamento através algoritmo de otimização *Adaptative Moment Estimation (ADAM)* [17]. De acordo com [26], em RNA, o processo de treinamento é repetido até que um dos critérios de parada especificado sejam atingidos. Neste trabalho, a condição de parada adotada corresponde ao número máximo de épocas igual a 200.

O conjunto de dados de entrada foi dividido em dois subconjuntos, treinamento (80%) e teste (20%), esse processo foi realizado utilizando a função *train_test_split* da biblioteca *sklearn*. A RNA foi submetida ao conjunto de treinamento, gerando um modelo preditivo para posteriormente aplicar o conjunto de teste. Para avaliar a capacidade da rede na tarefa de detecção de contas bot, foram utilizadas as métricas de desempenho acurácia[2] e erro médio quadrático (MSE[3]). Além disso, para evitar o problema de sobreajuste foi aplicada a técnica de regulação *dropout*, a qual influencia na performance da rede. Com essa técnica, na fase de treinamento e durante o passo de propagação dos dados pela rede, os neurônios são desativados, aleatoriamente, com probabilidade p, em particular neurônios de camadas ocultas. Enquanto que na fase de teste, as ativações são

[2]Indica a performance geral do modelo, ou seja, de todas as classificações, quantas estão corretas.
[3]Mean Square Error: média das diferenças quadradas entre os valores previstos e reais.

re-escaladas, com fator p, para compensar as ativações que foram desligadas durante a fase de treinamento [25].

Para a análise da periculosidade, foram coletados os *tweets* publicados pelos usuários classificados como bot pela RNA. Utilizando a *Twitter API Reference*, foram obtidos os últimos 200 *tweets* de cada usuário. Neste trabalho a periculosidade de um bot se refere ao tipo de sentimento que o usuário expõe, o qual pode ser classificado como positivo, negativo ou neutro, ao publicar um *tweet*. Para essa classificação foi utilizada a técnica de Análise de Sentimentos, área dedicada ao estudo computacional das opiniões e sentimentos expressos em textos [16]. Para implementar a Análise de Sentimentos, foi utilizada a biblioteca *textblob* [21], a qual é amplamente utilizada no processamento computacional de linguagem natural. Com a *textblob* é possível obter o valor da polaridade[4] do texto, que varia entre -1 e 1. Para a classificação adotada neste trabalho, se a polaridade é maior que 0, o sentimento é positivo, se menor que 0, o sentimento é negativo e, se igual a 0, o sentimento é neutro.

Como mencionado, devido ao crescente número de contas bot, o Twitter vem desenvolvendo medidas para desativar os usuários considerados suspeitos. Por isso, das 242 contas classificadas como bot pela RNA, 46 já não existiam mais, inviabilizando a coleta dos *tweets*. Ainda, por limitação da biblioteca *textblob*, com a qual é possível avaliar textos na língua inglesa, 77 contas foram excluídas da análise por apresentarem publicações em outros idiomas (italiano, português, japonês, espanhol e indonésio). Portanto, a análise da periculosidade foi realizada em 119 contas. Além disso, como mencionado, foram coletados 200 *tweets* de cada usuário. Porém, como alguns usuários possuíam uma quantidade menor de publicações em sua linha do tempo, foram avaliados, no total, 22855 *tweets*.

3.2 Resultados

Nesta seção são apresentados os resultados para a detecção de contas bot e análise da periculosidade.

3.2.1 Detecção de contas bot

Como mencionado, a técnica de regularização *dropout* influencia a performance da rede. Na Tabela 2 é apresentado o resultado das métricas de desempenho da rede neural com e sem a aplicação do *dropout*. Podemos verificar que os valores das métricas de desempenho para essas diferentes configurações da RNA foram bem próximos. Assim, nas Figuras 1 e 2, são apresentados somente os resultados com a aplicação do *dropout*, os quais correspondem, respectivamente, ao comportamento do erro médio quadrático (Figura 1) e da acurácia (Figura 2) durante o processo (número de épocas) de treinamento e teste.

É possível observar que o erro diminui (Figura 1) e a acurácia aumenta (Figura 2) a medida que as épocas evoluem, o que é um bom resultado para a RNA implementada. Além disso, para os dois resultados, a curva de teste (cor laranja) tem um comportamento semelhante a curva de treinamento (cor azul), não apresentando sobreajuste, ou seja, o modelo implementado está adequado para receber um conjunto de dados nunca visto.

[4]Representa o grau de positividade e negatividade de um texto.

	Sem dropout	Com dropout
Acurácia	0,78	0,82
Erro	0,17	0,15

Tabela 2: Comparação das métricas de desempenho da RNA com e sem dropout.

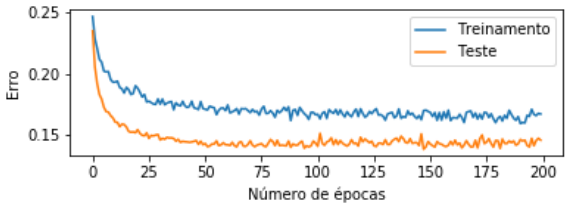


Fig. 1: Erro médio quadrático

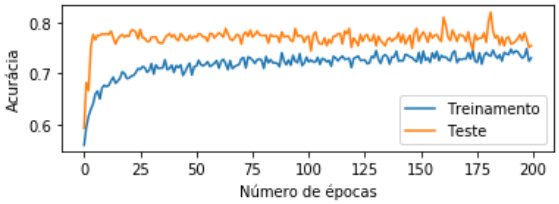


Fig. 2: Acurácia

3.2.2 Periculosidade do bot

Dos 594 usuários do conjunto de teste, 242 foram classificados como bot pela RNA. Devido aos fatores detalhados na Seção 3.1, foi possível coletar as publicações de 119 usuários, totalizando 22855 *tweets* para análise. Esses usuários foram os investigados quanto à periculosidade, para os quais o resultado é apresentado a seguir.

Os gráficos das Figuras 3 e 4 representam a porcentagem de usuários bot com periculosidade positiva (fatia azul), negativa (fatia laranja) ou neutra (fatia verde). Como cada usuário possui mais de um *tweet* publicado, a periculosidade considerada se refere ao sentimento que ocorreu em maior frequência dentre todos os *tweets* avaliados deste usuário. Também foram avaliados todos os *tweets* da base de uma maneira geral, sem levar em consideração a quais usuários pertencem. O resultado é apresentado no gráfico da Figura 4. Em todos os resultados é possível observar que a quantidade de contas (Figura 3) e *tweets* (Figura 4) classificados como neutros ou positivos foi consideravelmente maior que aqueles classificados como negativos.

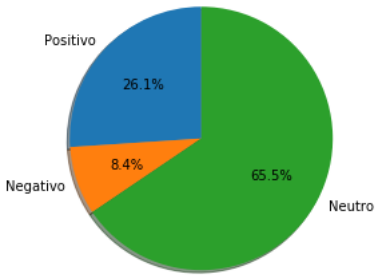


Fig. 3: Sentimentos dos 119 usuários bot

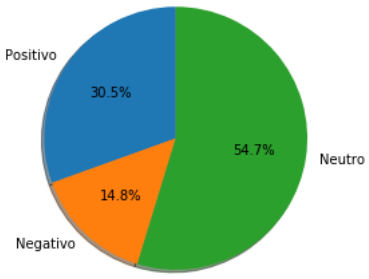


Fig. 4: Sentimentos nos 22855 tweets

4 Considerações Finais e Trabalhos Futuros

Neste trabalho foi implementada uma RNA do tipo MLP para detectar contas bot no Twitter. Além disso, foi utilizada a técnica de Análise de Sentimentos para avaliar qual a periculosidade das contas classificadas como bot.

Com os resultados obtidos na detecção de contas bot foi observado que o maior tempo é investido na fase de pré-processamento de dados do que com o ajuste dos parâmetros da rede, para que o treinamento e teste da RNA não fiquem prejudicados e as métricas

de desempenho (erro e acurácia) tenham um resultado satisfatório. Na segunda fase, mesmo que a técnica de Análise de Sentimentos tenha sido realizada sobre os *tweets* de uma maneira geral, sem levar em consideração qualquer evento ou assunto específico, ainda assim a análise é relevante. Normalmente, quando se avalia publicações de um evento, pessoa ou assunto específico, contas bot que não possuem publicações envolvidas no contexto de interesse deixam de ser investigadas, podendo prejudicar os usuários reais caso possuam intenções maliciosas.

Assim, este trabalho contribui para a compreensão de um fenômeno recente, mas de grande impacto em nossa sociedade. Entender a periculosidade de robôs em redes sociais é tarefa inextinguível e com vários desafios, tendo em consideração a dificuldade da coleta de dados em grande quantidade para fins de pesquisa. Além disso, os resultados obtidos certamente enriquecerão técnicas de mensuração dos danos causados pela presença de robôs e suas multifacetadas atuações em vários contextos. A combinação de análise de sentimentos e periculosidade ainda carece de trabalhos que possam propor modelos para avaliar, de acordo com o contexto, como se comportam os robôs e como eles podem ser neutralizados caso visem causar danos à imagem de indivíduos ou promover comportamentos que violem leis, mais especificamente, os direitos humanos.

Como trabalhos futuros são propostos: avaliar usuários de outras mídias sociais; identificar e classificar qual o tipo de dano que um bot com periculosidade negativa poderá trazer para os usuários reais, caso seu objetivo malicioso seja alcançado; e implementar um analisador de sentimentos que seja capaz de avaliar textos em diferentes idiomas.

Para fins de reprodutibilidade, os algoritmos implementados estão disponíveis no repositório `https://github.com/dsilva0101/Bot-Detection-and-Classification`.

5 *Agradecimentos*

Ao LNCC, ao grupo ComCiDis e à Faperj pelo suporte no desenvolvimento deste trabalho.

Referências

[1] K. R. d. Barbosa, G. Martins, E. Souto, and E. Feitosa. *Botnets: Características e Métodos de Detecção Através do Tráfego de Rede*, pages 99–144. 11 2014.

[2] D. Barojan. Noções básicas sobre bots, botnets e trolls. 2019. `https : / / ijnet . org / pt-br / story / no % C3 % A7 % C3 % B5es-b % C3 % A1sicas-sobre-bots-botnets-e-trolls`.

[3] G. E. d. A. P. Batista et al. *Pré-processamento de dados em aprendizado de máquina supervisionado*. PhD thesis, Universidade de São Paulo, 2003.

[4] F. Benevenuto, G. Magno, T. Rodrigues, and V. Almeida. Detecting spammers on twitter. In *Collaboration, electronic messaging, anti-abuse and spam conference (CEAS)*, volume 6, page 12, 2010.

[5] F. Chollet et al. Keras. `https://keras.io`, 2015.

[6] Z. Chu, S. Gianvecchio, H. Wang, and S. Jajodia. Detecting automation of twitter accounts: Are you a human, bot, or cyborg? *IEEE Transactions on Dependable and Secure Computing*, 9(6):811–824, 2012.

[7] I. T. Corrêa et al. Análise dos sentimentos expressos na rede social twitter em relação aos filmes indicados ao oscar 2017. 2017.

[8] A. Cutler, D. R. Cutler, and J. R. Stevens. *Random Forests*, pages 157–175. Springer US, Boston, MA, 2012.

[9] T. C. de França and J. Oliveira. Análise de sentimento de tweets relacionados aos protestos que ocorreram no brasil entre junho e agosto de 2013. In *Anais do III Brazilian Workshop on Social Network Analysis and Mining*, pages 128–139. SBC, 2014.

[10] T. Dourado. *Fake news na eleição presidencial de 2018 no Brasil*. PhD thesis, 06 2020.

[11] D. A. M. Dutra and E. de Rezende Francisco. Text mining: Análise de sentimentos nas eleições 2018. In *Congresso Transformação Digital 2018*, 2018.

[12] E. Ferrara. Bots, elections, and social media: a brief overview. *arXiv e-prints*, page arXiv:1910.01720, Oct 2019.

[13] E. Ferrara, O. Varol, C. Davis, F. Menczer, and A. Flammini. The rise of social bots. *Communications of the ACM*, 59(7):96–104, 2016.

[14] C. A. Iglesias and A. Moreno. Sentiment analysis for social media, 2019.

[15] C. Jain. Detecting twitter bot data, 2019. `https://www.kaggle.com/charvijain27/detecting-twitter-bot-data`.

[16] A. U. Kauer. Análise de sentimentos baseada em aspectos e atribuições de polaridade. 2016. `https://www.lume.ufrgs.br/handle/10183/140910`.

[17] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[18] P. Langley and S. Sage. Induction of selective bayesian classifiers. In *Uncertainty Proceedings 1994*, pages 399–406. Elsevier, 1994.

[19] Á. Lima, W. Lacerda, and H. Neto. Seleção de características de dados utilizando redes neurais artificiais. In *Anais do XIII Simpósio Brasileiro de Sistemas de Informação*, pages 135–142. SBC, 2017.

[20] T. Lokot and N. Diakopoulos. News bots: Automating news and information dissemination on twitter. volume 4, pages 682–699. Taylor & Francis, 2016.

[21] S. Loria. Textblob: Simplified text processing, 2021. `https://textblob.readthedocs.io/en/dev/`.

[22] T. M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[23] B. Neto and N. Arruda. Análise de sentimentos do twitter como suporte aditivo para a previsão da volatilidade do bitcoin. 2018.

[24] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[25] M. A. Ponti and G. B. P. da Costa. Como funciona o deep learning. *arXiv preprint arXiv:1806.07908*, 2018.

[26] E. M. Ramos. Análise de fluidos via técnicas de decomposição em modos dinâmicos e aprendizado profundo. Master's thesis, Laboratório Nacional de Computação Científica, Petrópolis, 2019.

[27] R. Recuero and A. Gruzd. Cascatas de fake news políticas: um estudo de caso no twitter. *Galáxia (São Paulo)*, (41):31–47, 2019.

[28] L. L. Regattieri. Perfis ciborgues: humanos-robôs e robôs-humanos nos ecossistemas de informação online. *Anais da ReACT-Reunião de Antropologia da Ciência e Tecnologia*, 4(4), 2019.

[29] C. Rock. Social media trends 2018. Technical report, 2018. `https://cdn2.hubspot.net/hubfs/355484/Ebooks%20MKTC/Social%20Media.pdf`.

[30] A. Rogers, A. Romanov, A. Rumshisky, S. Volkova, M. Gronas, and A. Gribov. Rusentiment: An enriched sentiment analysis dataset for social media in russian. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 755–763, 2018.

[31] A. S. d. Silva et al. Detectando comportamento automatizado nos tópicos de tendência do twitter no brasil. Universidade Federal do Amazonas, 2015.

[32] C. Q. Silva. *Redes neurais aplicadas no reconhecimento de símbolos matemáticos manuscritos online*. PhD thesis, Universidade de São Paulo, 2019.

[33] D. Silva, S. Silva, and R. M. Salles. Metodologia de detecção de botnets utilizando aprendizado de máquina. 2017. `http://www.sbrt.org.br/sbrt2017/anais/1570362118.pdf`.

[34] J. Toth. Bot or not: Detectando robôs no twitter. 2018. `http://data7.blog/bot-or-not-detectando-robos-no-twitter-botrnot/`.

[35] Twitter. Get started with the twitter developer platform, 2021. `https://developer.twitter.com/`.

[36] O. Varol, E. Ferrara, C. A. Davis, F. Menczer, and A. Flammini. Online human-bot interactions: Detection, estimation, and characterization. In *Eleventh international AAAI conference on web and social media*, 2017.

[37] K. Yang. Bot repository, 2019. `https://botometer.osome.iu.edu/bot-repository/datasets.html`.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Simulação de grandes escalas do escoamento gás-líquido em um misturador estático

Guilherme Santos Souza[1], Guilherme Barbosa[2], Vinícius Lobosco[3] e Raquel J. Lobosco[1]

[1] *Universidade Federal do Rio de Janeiro, Brasil*
[2] *Universidade Federal de Pernambuco, Brasil*

Abstract

Misturadores industriais são equipamentos essenciais em diversos processos da industria química, devido principalmente à aceleração de reações químicas e à transferência de massa associada. Este trabalho de pesquisa avalia a mistura gás-liquido em um misturador estático do tipo Venture. A turbulência tem um papel importante no contexto da mistura e dissipação viscosa. O fenômeno turbulento da mistura gás-líquido foi avaliado pelo método de média de Reynolds(RANS) e de vortices de grandes escalas(LES). Foram simuladas 5 condições de contorno, variando a velocidade de entrada de gás para cada um dos modelos de turbulência. Apesar da velocidade e da pressão apresentarem a mesma ordem de grandeza, houveram diferenças relevantes entre o comportamento dos modelos numéricos na representação do problema físico.

Keywords: Modelos de Turbulência, Escoamento Multifásico, OpenFOAM

1 INTRODUÇÃO

Nos últimos anos, diversos cientistas conduziram pesquisas em misturadores para melhorar a eficiência das aplicações em processos de misturadas industriais [1], [7], [12]. Otimizar as configurações dos equipamentos melhora a taxa de mistura em função do consumo energético, o que reduz os custos operacionais.

Os misturadores estáticos podem ser utilizados em processos em que a mistura dos componentes ocorre dentro de uma tubulação. Estes dispositivos aumentam o grau de turbulência e dissipação e, por consequência, aumentam a taxa de transferência de massa entre as fases. Dessa forma, é possível reduzir o comprimento de mistura de 100 diâmetros para uma faixa entre 4 e 6 diâmetros [6]. Na caso de misturas bifásicas entre um gás e um

Contato: Guilherme, gui.eng.santos@gmail.com

líquido, especificamente, o principal objetivo é maximizar a superfície de contato, entre as fases, o mais rápido possível.

Alguns modelos de misturadores estáticos usam tubos de Venturi para melhorar o processo de mistura. A constrição mecânica produz uma queda de pressão que pode estimular o processo de cavitação. Esse fenômeno contribui para a dissolução dos gases. A grande vantagem em adotar esse formato geométrico pode percebida nos níveis de eficiência da mistura, no baixo consumo energético e baixo comprimento de mistura [8];

A fluidodinâmica computacional é uma ferramenta muito útil no contexto, para propor possíveis configurações dimensionais e avaliar a respectiva eficiência energética. A turbulência e a cavitação são fenômenos cuja complexidade torna, a solução analítica das equações, muitas vezes, uma alternativa inviável. Da forma análoga, em alguns escoamentos, não é possível mensurar propriedades experimentais de forma direta. E para esses casos, a simulação numérica torna-se uma boa alternativa.

Shi et al [8] conduziram um estudo avaliando a mistura em tubos Venturi em 2 e 3 dimensões comparando o model LES WALE e o modelo RANS $k-\epsilon$. *Dittakavi et al* [3] aplicaram método LES com o esquema $AUSM + up$ para avaliar os cavitação em bocal de tipo Venturi.

O presente trabalho busca avaliar numericamente a mistura em um dispositivo industrial utilizado no processo de extração de lignina (Fig.1). O equipamento consiste na geometria de uma tubulação que sofre um estrangulamento da seção transversal para injeção de gás [2]. Diferentes modelos de turbulência foram avaliados no tratamento numérico de representação desse processo de mistura.

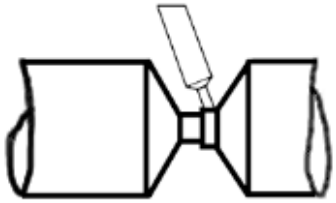


Fig. 1: Misturador estático usado na extração de lignina [2]

2 MODELO COMPUTACIONAL

Foi utilizado o método de volumes finitos para resolver as Equações de Navier-Stokes. Elas representam a quantidade de movimento dos fluidos através das equações de conservação. Foi adotado um escoamento adiabático e incompressível, de forma que a conservação da massa (1) e do momento(2) são resolvidas quando aplicadas essas simplificações.

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0 \tag{1}$$

$$\frac{\partial}{\partial t}(\rho \vec{v}) + \nabla \cdot (\rho \vec{v} \otimes \vec{v}) = \rho \frac{D\vec{v}}{Dt} \tag{2}$$

Em que, $\vec{v}$ é o vetor velocidade, ρ é a densidade e t é o tempo.

Existem duas formas de avaliar o movimento dos fluidos: através da abordagem Euleriana e Lagrangiana. A abordagem de Euler adota um referencial estático, enquanto no método Lagrangiano o referencial acompanha o escoamento. Essa diferenciação é especialmente importante em escoamentos multifásicos que requerem a captura da interface entre dois fluidos. No presente trabalho optou-se pela abordagem euleriana, em ambas as fases e a interface é capturada através da fração volumétrica.

Os escoamentos turbulentos são caóticos e randômicos. Resolver esses escoamento através da resolução direta nas Equações (1 e 2) demanda uma capacidade computacional extremamente avançada. Para contornar esse problema, duas classes de modelos de turbulência são comumente utilizados, a média de Reynolds e os vórtices de grandes escalas.

2.1 Média de Reynolds

Os modelos de média de Reynolds partem da hipótese de que o campo de velocidade e pressão do escoamento podem ser divididos em uma propriedade média e uma propriedade flutuante, em que a pressão e velocidade podem ser escritas conforme as Equações 3.

$$\vec{u} = \bar{\vec{u}} + \vec{u}'; \vec{p} = \bar{\vec{p}} + \vec{p}' \tag{3}$$

2.1.1 $k - \omega SST$

A substituição das variáveis nas Equações de Navier - Stokes e da conservação da massa fazem com que 6 novas variáveis derivem dos termos das médias e das flutuações. Para resolver este sistema de equações, faz-se necessário acrescentar um modelo de fechamento para o problema. O modelo $k - \omega SST$ se apropria de duas abordagens muito utilizadas. Ele combina os modelos $k - \epsilon$ e $k - \omega$.

$$\frac{\partial \rho\omega}{\partial t} + \frac{\partial \rho U_i \omega}{\partial x_i} = \alpha\rho S^2 + \beta\rho\omega^2 + \frac{\partial}{\partial x_i}\left[(\mu + \sigma_\omega \mu_i)\frac{\partial \omega}{\partial x_i}\right] + 2(1-F)\rho\sigma_{\omega^2}\frac{1}{\omega}\frac{\partial k}{\partial x_i}\frac{\partial \omega}{\partial x_i} \tag{4}$$

O modelo $k - \epsilon$ apresenta grande ineficiência nas proximidades das paredes. Para solucionar este problema Menter propôs o modelo $k-\omega$ como alternativa para resolver essa região de instabilidade. Em 2003, ele publicou uma série de modificações para aprimorar o modelo proposto. Revisou as constantes do modelos e propôs uma função para suavizar a transição entre os diferentes métodos, conforme mostra a Equação 5. [5]

$$C = F_c C_1 + (1 - F_c)C_2 \tag{5}$$

Em que C é a constante de proporcionalidade no calculo da viscosidade turbulenta, C_1 é a constante referente ao modelo $k - \omega$, C_2 é a constante referente ao modelo $k - \epsilon$ e F_c é a função de mistura. Esta, é retorna um valor de 0 a 1 é calculada em função da energia cinética trubulenta k, a frequência da turbulência ω, a distância da parade y e o número de Reynolds Re.[10]

2.2 Vórtices de grandes escalas

Os modelos de grandes escalas, diferentemente do método de médias de Reynolds, rastreiam o surgimento de grandes vorticidades. Filtram os vórtices e incluem apenas os

maiores na resolução das Equações de Navier-Stokes, removendo as vorticidades de menores escalas. Por fim, onde as equações não foram resolvidas, calcula-se as propriedades em função da média da vizinhança. Esses métodos vêm sendo cada vez mais aplicados em problemas de maior complexidade do escoamento.

2.2.1 Smagorinsky

Smagorinsky supôs que, desde que os menores turbilhões fossem isotrópicos, era esperado que a hipótese de Boussinesq desse uma boa aproximação para os vórtices não resolvidos em um escoamento de grandes escalas [9]. Isso implica que a tensões locais, τ_{ij} nessas regiões seriam proporcionais às taxas de deformação da vizinhança já resolvida, conforme mostra a Equação 6.

$$\tau_{ij} = -\mu_{SGS}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right) + \frac{1}{3}\tau_{ii}\delta_{ij} \tag{6}$$

Em que, μ_{SGS} é a viscosidade dinâmica SGS em $[Pa.s]$. O termo $\frac{1}{3}\tau_{ii}\delta_{ij}$ garante que a soma das tensões locais modeladas seja igual à energia cinética da turbulência.[10]

A viscosidade cinemática pode ser escrita em termos da escala de velocidade e de comprimento. Como os vórtices são rastreados pela função de filtragem pelo seu comprimento, esse valor é adotado como uma escala de comprimento. A escala de velocidade é definida como o produto da escala de comprimento, Δ e a taxa de deformação $\Delta \times |\overline{S}|$, em que $|\overline{S}| = \sqrt{2\overline{S}_{ij}\overline{S}_{ij}}$ [10]. Dessa forma, temos que,

$$\nu_{SGS} = (C_{SGS}\Delta)^2|\overline{S}| \cong \rho(C_{SGS}\Delta)^2\sqrt{2\overline{S}_{ij}\overline{S}_{ij}} \tag{7}$$

C_{SGS} é uma constante, em que, para escoamentos internos o valor $C_{SGS} = 0,1$ e ν_{SGS} é viscosidade cinemática na região filtrada.[10]

2.2.2 k Equation

O modelo estático de grandes escalas de energia cinética ao invés de utilizar a hipótese de Boussinesq calcula a energia cinética na região não resolvida e representa a viscosidade conforme descrito na Equação 8.[4]

$$\nu_{SGS} = C_k k_{SGS}^{\frac{1}{2}}\Delta \tag{8}$$

Em que δ é o comprimento de filtragem de vórtices, é C_k é uma constante e k_{SGS} é a energia cinética na região filtrada é calculada através da Equação 9.

$$\frac{\partial \overline{k}_{SGS}}{\partial t} + \frac{\partial \overline{u}_j \overline{k}_{SGS}}{\partial x_j} = -\tau_{ij}\frac{\partial \overline{u}_i}{\partial x_j} - C_\epsilon \frac{k_{SGS}^{\frac{3}{2}}}{\Delta} + \frac{\partial(\frac{\nu_{SGS}}{\sigma_k}\frac{\partial k_{SGS}}{\partial x_j})}{\partial x_i} \tag{9}$$

Em que τ_{ij} é a tensão cisalhante, C_ϵ é uma constante e os três termos à direita representam, respectivamente, a taxa de produção, dissipação e transporte da energia cinética.[4]

A constante C_k para o caso estático foi adotada com o valor $1,048$ e C_ϵ, $0,094$ [11]. Para o modelo dinâmico, as constantes são calculadas em conforme as equações a seguir:

$$C_k = \frac{L_{ij}M_{ij}}{2M_{ij}M_{ij}} \tag{10}$$

Onde L_{ij} é tensor das tensões resovildas, e M_{ij} é calculado em função da energia cinética, taxa de deformação e comprimento de corte para as escala resolvidas, [4]

$$C_\epsilon = \frac{F}{G} \tag{11}$$

Onde, F é calculado é função das taxas de deformação das escalas resolvidas e G é calculado em função o comprimento de corte e energia cinética das escalas resolvidas.[4]

3 METODOLOGIA

Esse trabalho realiza uma análise comparativa entre os modelos de turbulência apresentados. Foram realizada 20 simulações ao todo com variação entre 5 condições de contorno com alterações entre 4 hipóteses do tratamento da turbulência. A velocidade de entrada do fluido principal foi definida em $1m/s$ e a saída estimada em $100kPa$. A velocidade de entrada do gás variou entre 6, 7, 8, 9 e $10m/s$.

O software utilizado para calcular os campos de propriedades do escoamento foi o software livre OpenFOAM. Os quatro modelos de turbulência adotados foram: 1− O modelo de equações de média de Reynolds $k-\omega SST$ com equações de transporte de tensão de cisalhamento; 2− Os modelos de vórtices de grandes escalas pelas hipóteses de Smagorinsky; 3− O modelo de equações estáticas de energia cinética e 4− O Método de equações dinâmicas de energia cinética.

O líquido adotado foi a água e o gás o CO_2. Para a água, adotou-se a viscosidade cinemática de $1\mu m^2/s^2$ e densidade $1000kg/m^3$. Para o gás CO_2 adotou-se a viscosidade cinemática de $14,8\mu m^2/s^2$ e densidade $1kg/m^3$.

A Figura 2 mostra a geometria adotada no presente trabalho e é uma simplificação da geometria do dispositivo original[2].O diâmetro principal (Dm) mede $80mm$, o diâmetro no estrangulamento (Dt) mede $50mm$ e o comprimento total da tubulação mede $892mm$. Há duas entradas de gás tangenciais no estrangulamento, com $10mm$ de diâmetro cada. O comprimento do pescoço (Lt) é de $40mm$. O comprimento do estrangulamento (Ls) é de $25,981mm$.

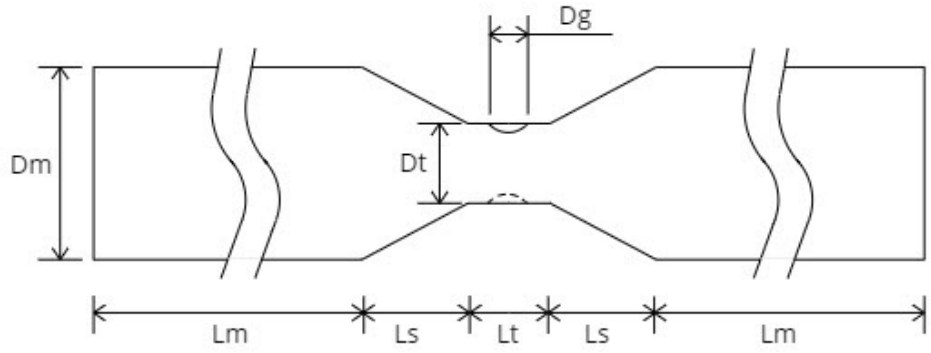


Fig. 2: Imagem representativa da geometria utilisada durante as simulações

Foi realizado uma análise da convergência do refinamento da malha. Cinco modelos de malha tetraédrica foram gerados através do software livre SALOME e avaliadas quando aplicadas a condição de velocidade de entrada do gás de $10m/s$ com o modelo de

turbulência RANS. Os resultados aqui apresentados fazem uso da malha otimizada que composta por 67731 elementos.

4 RESULTADOS

Em uma comparação dos modelos de turbulência, o gráfico da Figura 3 apresenta a flutuação de velocidade para cada modelo no centro da seção de saída. Com as variáveis adimensionalizadas e a velocidade de entrada do gás de $10m/s$, tem-se $t0$ como o tempo total de simulação e $v0$ como a velocidade de entrada de água. Pelo gráfico é possível perceber que o modelo de média de Reynolds (em azul) apresentou menor flutuação em comparação com os demais modelos. Conforme esperado para um modelo de média de Reynolds.

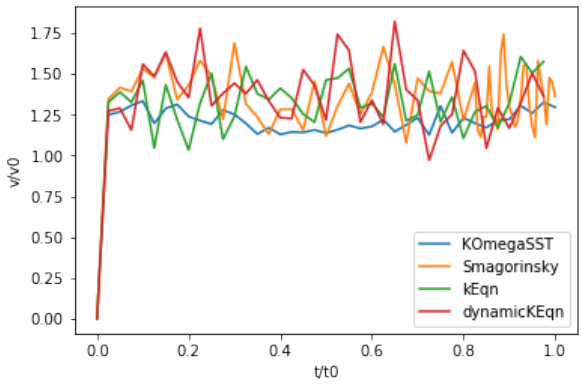
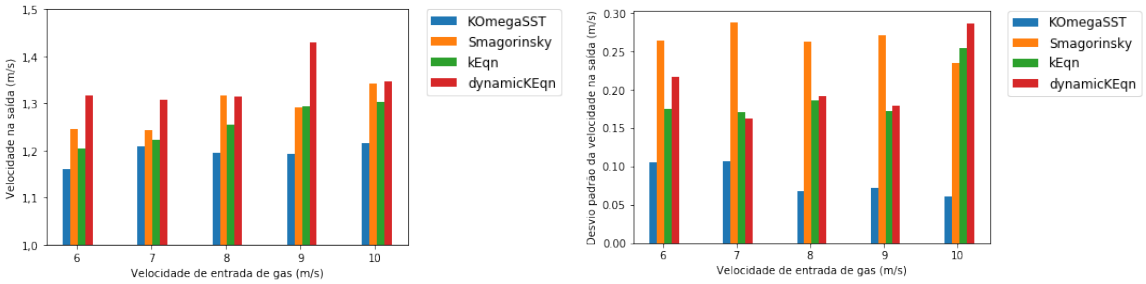


Fig. 3: Gráfico da flutuação de velocidade em cada modelo de turbulência para a condição de contorno com entrada de gás de $10m/s$

A velocidade média e o desvio padrão para a mesma posição sobre as demais condições de contorno podem ser visualizado na Figura 4. Neste caso é possível perceber que o modelo dinâmico foi o que apresentou os maiores ajustes para compensação das flutuações de velocidade. No entanto, o modelo de Smagorinsky mostrou uma similarida ao método de média de Reynolds no comportamento do desvio padrão, apesar da diferença de escala.

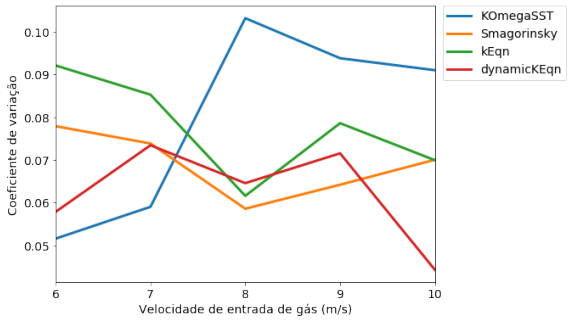


(a) Velocidade média **(b)** Desvio padrão

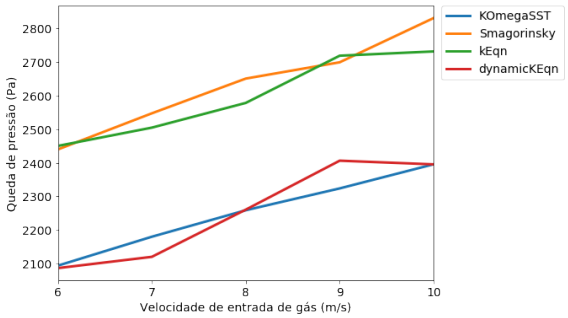
Fig. 4: Gráficos referentes à velocidade no raio zero da seção de saída.

Para o desempenho do dispositivo na aplicação industrial, é essencial avaliar a qualidade da mistura e a quantidade de energia empregada durante o processo operacional. Esse parâmetro pode ser medido através do coeficiente de variação e da queda de pressão.

A Figura 5 apresenta o resultado de ambas as métricas nas diferentes condições de contorno.



(a) Coeficiente de variação na saida do misturador



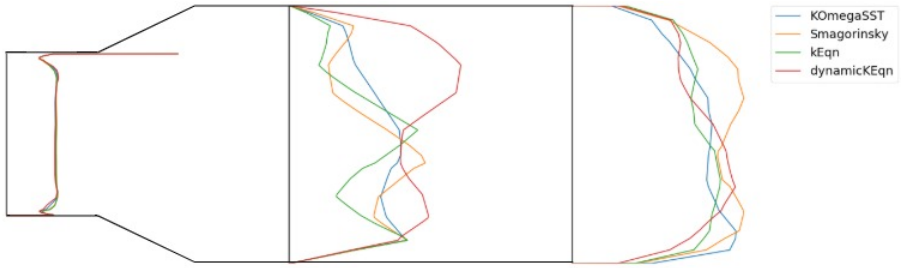
(b) Queda de pressão

Fig. 5: Graficos referentes ao **(a)** coeficiente de variação na saída do misturador e a **(b)** a diferença de pressão entre a entrada de água e a saída da mistura.

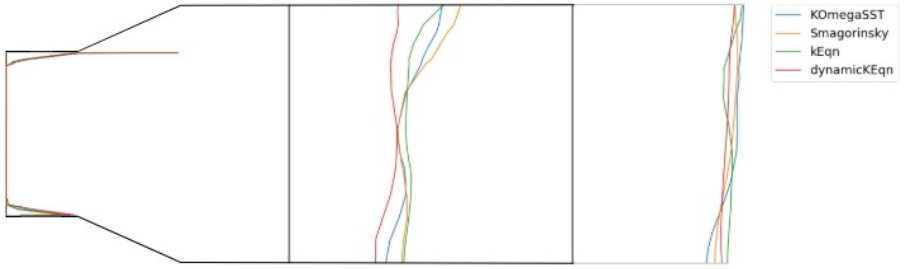
Na Figura 5a é possível perceber que o modelo de média de Reynolds apresenta uma tendência de crescimento do coeficiente de variação de acordo com o aumento da velocidade de entrada do gás. Nesse caso é possível afirmar que o modelo de média de Reynolds sofre uma influência maior da velocidade de entrada do gás do que os modelos de grandes escalas que devem sofrer uma influência maior com o aumento da velocidade da fase dispersa. É interessante notar que para um valor intermediário, da velocidade de entrada, em torno de $8m/s$, o comportamento do coeficiente de variação sofre uma alteração de comportamento entre o modelo de média de Reynolds e das grandes escalas. Acredita-se que esse comportamento se deve justamente ao somatório das médias das flutuações.

A Figura 5b mostra a queda de pressão com relação a velocidade de entrada do gás para cada um dos modelos numéricos. O modelo de média de Reynolds foi o em único que apresentou uma relação linear entre a queda de pressão e a velocidade de entrada de gás. Os modelos de equação cinética tiveram uma resposta similar entre si. As hipóteses de grandes escalas estáticas apresentaram um deslocamento em relação aos demais de aproximadamente $200Pa$. Uma diferença significativa dos modelos numéricos que se agrupam em suas características de representação numérica do problema físico.

A Figura 6 é uma ilustração do perfil de velocidade e fração volumétrica. Observa-se as distribuições de velocidade e fração volumétrica e percebe-se que o tratamento da turbulência por média de Reynolds levou a uma distribuição mais uniforme da velocidade com uma distribuição menos uniforme das fases, o que caracteriza um menor grau de agitação, conforme ilustrado também pela Figura 3. Há uma grande divergência entre dos perfis de velocidade (na seção intermediária) entre o modelo de grandes escalas dinâmico e os demais métodos. O perfil de velocidade converge nas parede e no centro (na velocidade mínima e máxima) entretanto, enquanto os demais perfis apresentam similaridade na declividade da curva do perfil de velocidade, o método dinâmico apresenta uma característica contrária.



(a) Velocidade



(b) Fração volumétrica

Fig. 6: Representação gráfica dos perfis de **(a)** velocidade e **(b)** fração volumétrica das seções transversais para a condição de entrada de gás a $10m/s$ no instante $t/t0 = 1$.

5 CONCLUSÃO

Conclui-se então que a escolha do modelo de turbulência pode causar divergência nos resultados das simulações. É preciso avaliar, de acordo com as características do problema físico, o método numérico mais apropriado correta representação. Os métodos de grandes escalas, apesar de exigirem um maior custo computacional, são capazes de capturar oscilações e flutuações de fenômenos de transição que seriam amortizados através dos métodos de média de Reynolds. Os modelos estáticos demonstraram características bastante similares entre si na representação do problema físico proposto. É de grande importância validar os resultados numéricos com dados experimentais ou da literatura.

Referências

[1] A. Basso, F. A. Hamad, and P. Ganesan. Effects of the geometrical configuration of air–water mixer on the size and distribution of microbubbles in aeration systems. *Asia-Pacific Journal of Chemical Engineering*, 13, 2018.

[2] Borges, R. S. . Misturador Estático. *Suzano Papel e Celulose*, BR 102014013032-2 A2.

[3] N. Dittakavi, A. Chunekar, and S. Frankel. Large Eddy Simulation of Turbulent-Cavitation Interactions in a Venturi Nozzle. *Journal of Fluids Engineering*, 132(121301), 2010.

[4] W.-W. Kim and S. Menon. *A New Dynamic One-Equation SubgridScale Model for Large Eddy Simulations*, volume 4. AiAA, 1995.

[5] F. Menter, M. Kuntz, and R. Langtry. Ten years of industrial experience with the sst turbulence model. *Heat and Mass Transfer*, 4, 01 2003.

[6] A. Paglianti. Recent Innovations in Turbulent Mixing with Static Elements. *Recent Patents on Chemical Engineering*, 1:80–87, 2008.

[7] M. Romańczyk and W. Elsner. Effect of Cylindrical Turbulators on the Mixing Process in Basic Venturi Gas Mixer Using OpenFOAM. *MATEC Web Conf.*, 252(04004):6, 2019.

[8] H. Shi, Q. Liu, and P. Nikrityuk. Numerical study of mixing of cavitating flows in a venturi tube. *The Canadian Journal of Chemical Engineering*, n/a(n/a), 2020.

[9] J. Smagorinsky. General circulation experiments with the primitive equations. *Monthly Weather Review*, 91(3), 1963.

[10] H. K. Versteeg and W. Malalasekera. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Pearson Education Limited, 1995,2007.

[11] A. Yoshizawa. Statistical theory for compressible turbulent shear flows, with the application to subgrid modeling. *The Physics of Fluids*, 29(7):2152–2164, 1986.

[12] Z. Zhuang, J. Yan, C. Sun, H. Wang, Y. Wang, and Z. Wu. The numerical simulation of a new double swirl static mixer for gas reactants mixingEffect of Cylindrical Turbulators on the Mixing Process in Basic Venturi Gas Mixer Using OpenFOAM. *Chinese Journal of Chemical Engineering*, 28:2438, 2019.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Aprendizagem Estrutural de Redes Bayesianas utilizando Algoritmo Genético Multi-Agente

Itallo Guilherme Machado[1] and Michel Bessani[1]

[1] *Programa de Pós-Graduação em Engenharia Elétrica - Universidade Federal de Minas Gerais, Belo Horizonte/MG, Brasil*

Abstract

As redes Bayesianas são um importante modelo gráfico probabilístico na área da inteligência artificial. A aprendizagem estrutural da rede Bayesiana que melhor representa os dados disponíveis é um problema NP-difícil. Esse problema apresenta, mais frequentemente, uma maior pesquisa em métodos heurísticos. Atualmente, o campo de pesquisa de problemas com grande volume de dados se tornou interessante na área de pesquisa de inteligência artificial. Nesse trabalho, será apresentado um método heurístico chamado Algoritmo Genético Multi-Agente (MAGA), esse algoritmo utiliza um sistema multi-agente para representar os indivíduos do Algoritmo Genético como agentes que interagem entre si e com o ambiente. Além dos operadores genéticos de cruzamento e mutação, nesse algoritmo também existe um operador de auto-aprendizagem que refina o melhor indivíduo. Os experimentos compararam a performance do Algoritmo Genético Multi-Agente com um Algoritmo Genético em instâncias de diferente complexidade e números de amostras. Os resultados desta primeira avaliação indicam que o MAGA é promissor para o problema de aprendizado estrutural de redes Bayesianas, inclusive sendo mais interessante para instâncias com maior volume de dados, em pesquisas futuras serão realizados experimentos em outras instâncias e com outros algoritmos.

Keywords: Redes Bayesianas, aprendizado estrutural, algoritmo genético, algoritmo genético multi-agente, sistema multi-agente

1 Introdução

As redes Bayesianas (*Bayesian Networks* - BNs) [15] são um tipo de modelo gráfico probabilístico que descreve a relação entre variáveis aleatórias, as quais podem ser discretas ou contínuas. Uma BN é composta por um grafo acíclico dirigido (*Directed Acyclic Graph*

Contato: Itallo Guilherme Machado, itallomac@ufmg.br

- DAG) e um conjunto de probabilidades condicionais. No DAG, onde os nós do grafo representam as variáveis e as arestas representam as relações de dependência entre os nós, a direção da aresta indica que um nó, chamado de pai, possui influência nos possíveis estados do nó apontado, sendo esse chamado de filho. As probabilidades condicionais são representadas através de uma tabela de probabilidade condicional (*Conditional Probability Table* - CPT) no caso de variáveis discretas, ou por distribuições de probabilidade condicional no caso de variáveis contínuas.

As BNs podem ser utilizadas em diferentes aplicações e contextos onde a incerteza nas variáveis envolvidas é relevante. Podemos citar o uso como ferramenta auxiliar no entendimento e na tomada de decisão sobre problemas de saúde, como câncer de pulmão [28] e Alzheimer [10]. Outra aplicação é em problemas de manutenção, como no diagnóstico de falhas em motores de aviões [22]. Outra utilização é no contexto de predição, como da capacidade de manutenção dos sistemas de um *software* [20], do consumo de energia elétrica de múltiplos domicílios [1], e dos padrões de movimento de múltiplos veículos em redes tolerantes ao atraso [32].

Dentre o campo de pesquisa relacionado as BNs, temos o desafio de encontrar o DAG que melhor representa os dados amostrados. Esse desafio é definido como aprendizagem estrutural das BNs e é um problema NP-difícil [4] e de natureza combinatorial, resultando em intensa pesquisa nessa área [23]. Na literatura, temos três principais abordagens para esse problema [5]: *Constraint-based*, onde a estrutura é determinada através de testes de independência condicional para determinar a estrutura não direcionada, e através de regras de orientação se direciona essa estrutura não direcionada; *Search and Score* (SS), a qual é baseada em heurísticas de busca pela melhor estrutura no espaço de estruturas possíveis, sendo orientada por alguma métrica de qualidade para cada estrutura verificada durante a busca; e uma abordagem híbrida dessas duas, onde os testes de independência são realizados para restringir o espaço de busca que será explorado pelo SS.

A abordagem mais utilizada para lidar com o problema de aprendizado estrutural de BNs é a SS [23], sendo esta a mais rápida entre as três [26], tanto para amostras pequenas como grandes, o que resultou no desenvolvimento de diversos algoritmos. Existem os que primeiro definem os conjuntos dos possíveis pais de cada uma das variáveis para depois definir a estrutura com melhor ajuste, caso do algoritmo K2 [7]. Também existem as abordagens que utilizam os algoritmos meta-heurísticos para realizar a busca pela estrutura que resulte no melhor ajuste, como o *Hill-Climbing* (HC) e Busca Tabu [26], o Algoritmo Genético (*Genetic Algorithm* - GA) [16], a Otimização por Enxame de Partículas [22] e outras abordagens utilizando versões modificadas do GA como o elitGA [6] e o PSAGA [18].

O objetivo deste trabalho é aplicar um algoritmo híbrido chamado Algoritmo Genético Multi-Agente (*Multi-Agent Genetic Algorithm* - MAGA) [35] para o aprendizado estrutural de BNs com grande volume de dados, o qual é um problema computacional atual e desafiador [27]. O MAGA utiliza um sistema multi-agente, em que os indivíduos são representados pelos agentes que interagem entre si e com o ambiente, e utilizam os operadores do algoritmo genético. A principal diferença entre o MAGA e o GA é que no MAGA os indivíduos irão interagir, de forma competitiva, apenas com os seus vizinhos [35], enquanto no GA a interação pode ocorrer entre quaisquer indivíduos. Essa característica do MAGA resulta em um processo mais similar ao mecanismo evolutivo real [31]. Essa forma de interação torna o MAGA interessante para problemas de otimização grandes,

tanto em dimensão como em volume de dados [35, 34]. Também já é sabido que esse algoritmo possui um bom desempenho em problemas de natureza combinatorial [31, 21]

O restante deste trabalho está organizado da seguinte maneira. A próxima seção apresentará a metodologia utilizada, descrevendo como o MAGA foi implementado e como os experimentos computacionais foram realizados, já a Seção 3 apresentará os resultados juntamente com a discussão dos mesmos, a Seção 4 traz as conclusões desse estudo e a Seção 5 os agradecimentos.

2 Metodologia

No MAGA, os agentes representam os indivíduos que estão distribuídos em um ambiente em formato de grade, de tamanho L, em que cada agente possui 4 vizinhos, como mostrado na Figura 1. O tamanho da grade, L, é definido com $L_{size} \times L_{size}$, onde $L_{size} \in N^*$. Supondo que um agente se localiza na posição (i, j) sendo $i, j \in \{1, 2, \dots\ L_{size}\}$, o conjunto de vizinhos é definido como mostrado na equação (1).

$$
\begin{aligned}
Vizinhos_{i,j} &= \{L_{i',j}, L_{i,j'}, L_{i'',j}, L_{i,j''}\} \qquad (1)\\
i' &= \begin{cases} i-1 & i \neq 1 \\ L_{size} & i = 1 \end{cases}, \qquad j' = \begin{cases} j-1 & j \neq 1 \\ L_{size} & j = 1 \end{cases},\\
i'' &= \begin{cases} i+1 & i \neq L_{size} \\ 1 & i = L_{size} \end{cases}, \; j'' = \begin{cases} j+1 & j \neq L_{size} \\ 1 & j = L_{size} \end{cases}
\end{aligned}
$$

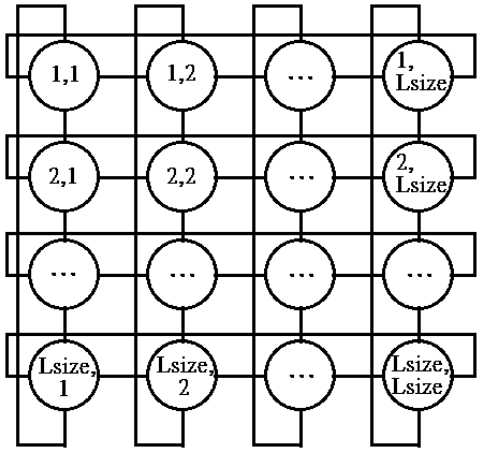


Fig. 1: Modelo da grade de agentes.

O MAGA possui os operadores convencionais do GA, cruzamento e mutação, além de um operador específico chamado de auto-aprendizagem [34] em que alguma heurística é utilizada para refinar a qualidade do agente que representa a melhor solução encontrada em cada iteração. O MAGA inicia gerando os agentes iniciais de forma aleatória. Cada geração consiste em: para cada agente, dado uma probabilidade de cruzamento (Pc), se encontra o vizinho com melhor pontuação. Se essa pontuação do melhor vizinho for melhor que o agente avaliado, os dois agentes passam pela operação de cruzamento que gera um novo agente que substitui o agente avaliado.

Após a avaliação de cada agente é determinado, através da probabilidade de mutação (Pm), o número de quantas arestas serão mutadas. Então é escolhido um agente de forma aleatória, caso ele não seja o agente que possui melhor pontuação na geração, ele sofre mutação. Se a pontuação do agente mutado for melhor que quando ele não tinha sido mutado, então o agente mantém a mutação. Se a mutação gerar uma agente com pontuação menor, então há uma probabilidade Po de manter a mutação. E por fim, é feito a operação de auto-aprendizado no agente com melhor pontuação dentre os agentes da geração corrente. Esse processo é repetido em cada geração até alcançar o critério de parada definido. Esses passos do MAGA são mostrados no pseudocódigo abaixo.

Algorithm 1: MAGA

1 Construção da grade multiagente e inicialização da população dos agentes;

2 Calcular o valor da qualidade da solução representada para cada agente;

3 Cruzamento com probabilidade Pc;

4 Mutação com probabilidade Pm, caso o agente gerado possuir pontuação pior que o agente multado, há uma probabilidade Po de aceite do agente gerado;

5 Aplicar o método da auto-aprendizado no melhor agente da geração;

6 Se o critério de parada for satisfeito retorna a melhor solução, caso contrário vá para o passo 2.

O operador de cruzamento utilizado no algoritmo funciona da seguinte forma: Dados dois agentes, são mantidas as arestas que são comuns e que possuam a mesma direção nos dois agentes para o agente filho, e para as arestas diferentes é escolhido ou do primeiro ou do segundo agente com uma probabilidade de 50% para o novo agente gerado. Já o operador de mutação verifica para todos os pares de nós, há uma probabilidade de Pm, se será ou não alterada a aresta entre os nós. Se sim, caso não exista uma aresta entre os nós, uma aresta é adicionada e sua direção é escolhida há uma probabilidade de 50%. Caso exista uma aresta entre os nós, é escolhido há uma probabilidade de 50% se essa será removida ou se será invertida a sua direção.

Tanto no operador de cruzamento, quanto no operador de mutação se a adição ou inversão de uma aresta tornar a rede cíclica, busca-se de forma aleatória uma das arestas desse ciclo, que não seja aquela alterada pelos operadores, e inverte sua direção ou a remove. Essa operação se repete até que não haja mais ciclos.

No operador de auto-aprendizado, que representa a interação do melhor agente com o ambiente, utilizou-se a heurística de busca tabu [21] que já é amplamente utilizada no aprendizado de BNs [3] [13] e possui bons resultados com o MAGA como mostrado no trabalho [21].

Na busca tabu, a principal ideia é iterativamente avaliar a vizinhança da solução. Para não ficar preso em mínimos locais, é utilizado a lista tabu para permitir, de forma limitada, as movimentações quando se explora soluções vizinhas com pior pontuação no espaço de busca. O algoritmo procura o melhor vizinho do agente, que não esteja na lista tabu. O operador de mutação gera esses vizinhos. Então é verificado se o melhor vizinho é melhor que o melhor agente, caso positivo o melhor agente é substituído pelo melhor vizinho, e por fim o atual agente é substituído pelo melhor vizinho. Então se atualiza a lista tabu de tamanho definido igual a 100. E o agente segue até que não tenha um

vizinho melhor que o melhor agente em t_max gerações. Na literatura [21] foram relatados melhores resultados em um problema de otimização combinatória ao utilizar a busca tabu comparada com a abordagem originalmente proposta de auto-aprendizagem do MAGA. Adicionalmente, a busca tabu é amplamente utilizada para o aprendizado estrutural de Redes Baysianas [3, 13]. Esses dois fatores motivaram a escolha da busca tabu como heurística de auto-aprendizagem no algoritmo proposto.

A métrica utilizada nesse trabalho será a BIC [24] apresentada na equação (2). Essa métrica calcula a verossimilhança da estrutura da rede em relação aos dados ($BIC\,(R \mid D)$), a qual é subtraída da complexidade estrutural da rede, resultando em uma medida de verossimilhança penalizada pela complexidade estrutural.

$$BIC\,(R \mid D) = \sum_{i=1}^{n}\sum_{j=1}^{q_i}\sum_{k=1}^{r_i} N_{ijk} log\left(\frac{N_{ijk}}{N_{ij}}\right) - \frac{1}{2} log(N) \sum_{i=1}^{n} (r_i - 1)\, q_i \tag{2}$$

Em que n é o número de nós; r_i é o número de estados do i-ésimo nó; q_i é o número de pais do i-ésimo nó, sendo que o valor de q_i é igual a 1 se o i-ésimo nó não possuir pais; N é o número de observações em D; N_{ijk} é o número total de observações na amostra em que o i-ésimo nó assume seu k-ésimo estado e os seus pais assumem o j-ésimo estado; N_{ij} é a quantidade de observações do i-ésimo nó quando seus pais assumem o j-ésimo estado.

2.1 Experimentos computacionais

Os *benchmarks* utilizados nos experimentos foram extraídos do *Bayesian Network Repository*[1] e são o *ASIA* [17] que representa um problema para diagnóstico de câncer de pulmão, o *CHILD* [29] que é referente a um problema de diagnóstico de problemas cardíacos em recém-nascidos, e o *INSURANCE* [2] que é relacionado com a estimativa dos riscos em seguro de automóveis. A Tabela 1 apresenta a quantidade de nós, arestas e de parâmetros para cada um dos problemas.

Tabela 1: Os valores dos problemas *benchmark*

Problema	N° de Nós	N° de Arestas	N° de Parâmetros
ASIA	8	8	18
CHILD	20	25	230
INSURANCE	27	52	984

O MAGA descrito anteriormente foi comparado com um GA, sendo que os dois algoritmos utilizaram os mesmos operadores de mutação e cruzamento. O GA utilizado é baseado no trabalho de Larranaga [16], amplamente utilizado para comparação como nos trabalhos [5, 9, 30]. O MAGA foi executado com parâmetros Pc de 0.9, Pm de 0.01, Po de 0.05, número de vizinhos da busca tabu em 25, o número máximo de gerações para a busca tabu em 10 e o L_{size} em 10, dessa forma o MAGA possui 100 agentes. O GA foi utilizado com uma Pc de 0.7, Pm de 0.01 e tamanho da população igual a 100. Os parâmetros tanto do MAGA quanto do GA, foram estabelecidas através de testes preliminares utilizando os mesmos valores avaliados no trabalho [16]. Foram testados os valores combinados de Pc de 0.5 até 0.9 com intervalos de 0.1 e Pm igual aos valores 0.01, 0.03,

[1] `https://www.bnlearn.com/bnrepository/`

0.05, 0.07 e 0.1. No caso do GA, para o problema ASIA foi obtido o melhor resultado com Pc igual à 0.9, similar aos trabalhos [14, 8, 19], mas para problemas com maior número de variáveis ou volume de dados, como o INSURANCE e CHILD, o GA com Pc igual à 0.7 resultou em um desempenho melhor. Portanto, foi escolhido o valor 0.7, devido ao objetivo do trabalho. Os parâmetros Pm de 0.01 para o GA e Pc 0.9, Pm 0.01 e Po 0.05 para o MAGA resultaram nos melhores resultados em todos os testes.

O critério de parada para ambos os algoritmos foi o número de avaliações da função BIC ou pela convergência para a rede ótima conhecida para cada instância. Para o problema *ASIA* foram utilizadas quatro instâncias, sendo o tamanho de amostra de cada instância igual a 500, 1000, 5000,10000 e 20000 com o número máximo de avaliações da função BIC de 50000, 50000, 50000, 75000 e 100000, respectivamente. Para o *CHILD* foram utilizadas três instâncias, sendo o tamanho de amostra igual a 500, 1000, 5000 e 10000 para cada instância e o número máximo de avaliações da função BIC de 50000, 75000, 100000 e 100000, respectivamente. E para o *INSURANCE* foram utilizadas cinco instâncias de 1000, 3000, 5000, 10000 e 20000 números de amostras para cada instância e e o número máximo de avaliações da função BIC de 15000, 50000, 80000 e 140000 para cada instância respectivamente. Os tamanhos das amostras em cada instância foram definidos de forma similar aos trabalhos [33, 30], que utilizaram tamanhos amostrais entre 500 até 10000. Como o trabalho trata de trabalhos com grande volume de dados, foi escolhido o problema com maior e com menor número de parâmetros para verificar o comportamento dos algoritmos com instâncias de até 20000 amostras. Para definir o número de avaliações da função BIC como critério de parada, foi feito um teste para posteriormente definir esses valores como a maior quantidade necessária de avaliações para encontra o ótimo entre os dois algoritmos.

O ambiente computacional utilizado para desenvolvimento e execução dos testes foi uma máquina com SO Windows 10, uma *CPU* i5-3570k 3.4GHz e com uma memória *RAM* de 8GB. Os algoritmos foram implementados na linguagem de programação *python*, e utilizaram as bibliotecas *networkx* [11] para manipulação dos grafos, *numpy* [12] para manipulação de vetores, a *rpy2* para utilizar a função de cálculo do BIC da biblioteca *bnlearn* [25] do R. Os dados utilizados foram gerados utilizando o R e a biblioteca *bnlearn*.

3 Resultados e Discussão

Nesta seção, serão apresentadas as informações de tempo de execução e porcentagem de convergência para cada uma das instâncias avaliadas. Na Figura 2 (a) apresentamos a porcentagem de convergência para o ótimo em cada uma das instâncias avaliadas do problema *ASIA*. É possível observar que, para instâncias com menos que 20000 amostras, o MAGA possui uma convergência para o ótimo de 100% em todas as instâncias testadas. Enquanto o GA apresentou convergência de 100% para o ótimo na instância com com 500 amostras, o GA possui uma média de 60% de convergência para as instâncias de 1000, 5000 e 10000 amostras e não convergiu na instância de 20000. Na Figura 2 (b) apresentamos o tempo computacional, na instância de 500 amostras o tempo do GA e do MAGA estão bem próximos, entretanto com o aumento dos dados, o tempo computacional do GA cresce de forma mais acentuada que o MAGA, sendo que para a instância de 20000 amostras, o tempo do GA é em torno de 430 segundos sem convergência, sendo muito maior que o do MAGA que é em torno de 60 segundos e resultou em 96% de convergência.

A Figura 2 indica uma melhor convergência para o ótimo e menor tempo computacional do MAGA, em todas as instâncias analisadas do *ASIA*. O número médio de avaliações da função BIC apresentou um comportamento similar ao tempo computacional, portanto nesse trabalho iremos apenas analisar o tempo computacional em todos os problemas.

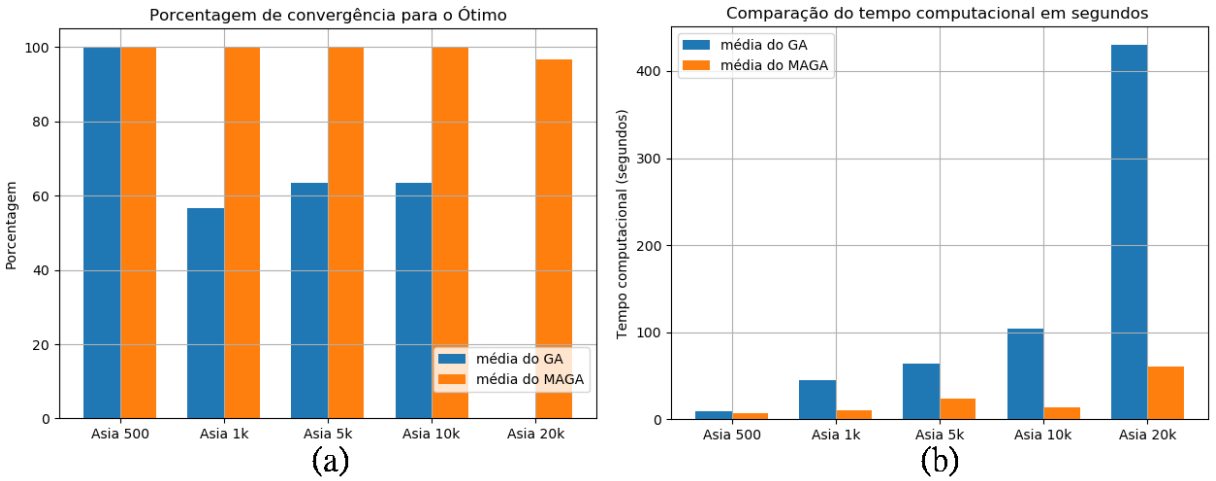


Fig. 2: Comparação entre MAGA e o GA para a porcentagem de convergência para o ótimo e para o tempo computacional. Para as quatro instâncias do *ASIA*.

A Figura 3 apresenta os resultados para as três instâncias do problema *CHILD*. Na Figura 3 (a) observamos que o MAGA mantém a convergência para o ótimo na primeira instância em 90% e as outras três instâncias em torno de 86%. Enquanto o GA possui uma convergência para a instância de 500 em 60%, uma convergência de 30% na instância de 1000 e as outras duas instâncias em aproximadamente 20%. Em relação ao tempo computacional, mostrado na Figura 3 (b), observamos que para a primeira instância o GA possui um tempo computacional próximo ao MAGA, mas quanto maior o número de amostras maior fica a diferença entre eles, sendo que na instância de 10000 temos uma diferença do dobro do tempo. Podemos ver que para o problema *CHILD*, o MAGA possui melhor desempenho que o GA, sendo que quanto maior o número de amostras disponíveis maior é a diferença de desempenho.

Na Figura 4, observamos as cinco instâncias do problema *INSURANCE*. Em relação à porcentagem de convergência para o ótimo, mostrados na Figura 4 (a), observamos que para as instâncias 1000, 3000 e 5000 o MAGA possui uma convergência em 100%. Enquanto o GA possui uma convergência de mais que 90% em todas as três primeiras instâncias. No entanto, nas instâncias maiores, sendo elas de 10000 e 20000 amostras, observamos uma redução da porcentagem nos dois algoritmos. O MAGA possui uma redução de aproximadamente 25% entre a instância 5000 e 20000, enquanto o GA possui uma redução mais significativa, para a instância 10000, o GA possui uma redução de aproximadamente 50% em relação à instância 5000 e na instância 20000 não consegue convergir para o ótimo. Na Figura 4 (b), temos que os tempos computacionais nas quatro primeiras instâncias são próximos, nas três primeiras instâncias de 1000, 3000 e 5000 o GA possui um tempo ligeiramente menor que o MAGA, sendo que na instância de 5000 a diferença da média dos tempos computacionais é de aproximadamente 6 segundos. Na instância 10000 o MAGA já possui um menor tempo que o GA e na instância de 20000 o

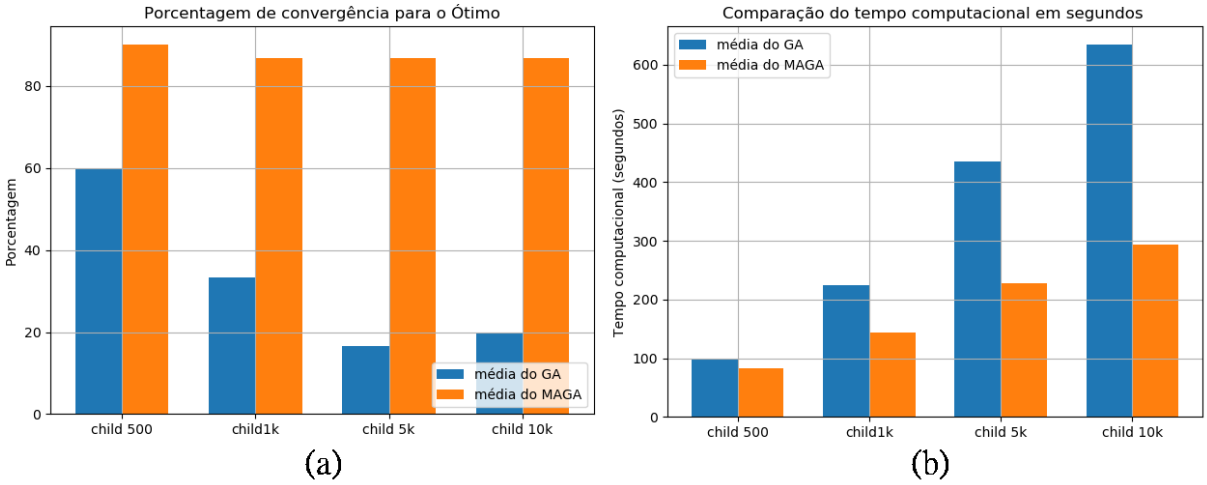


Fig. 3: Comparação entre MAGA e o GA para a porcentagem de convergência para o ótimo e para o tempo computacional. Para as três instâncias do *CHILD*.

MAGA possui uma diferença ainda maior em relação ao GA.

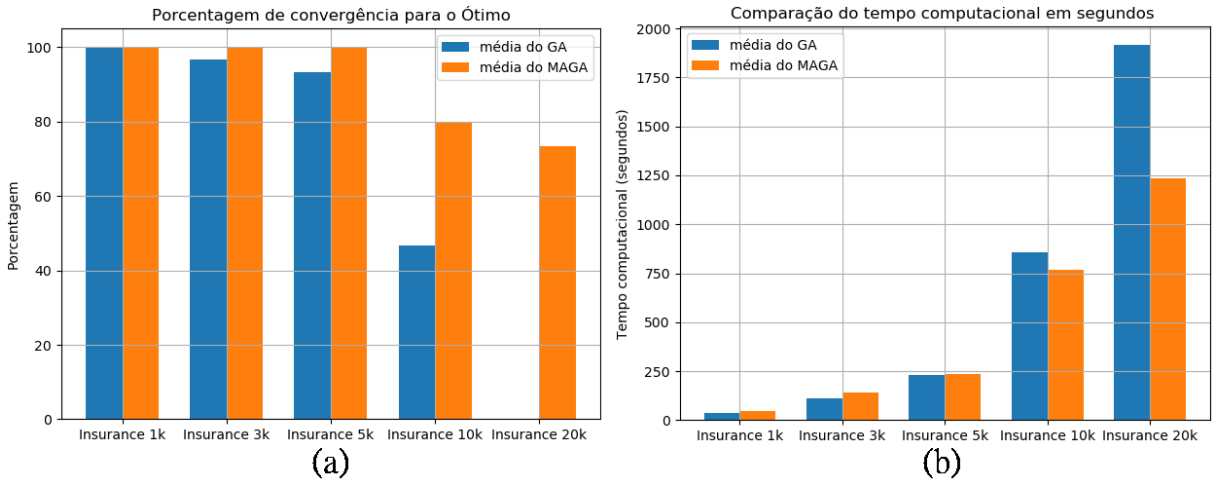


Fig. 4: Comparação entre MAGA e o GA para a porcentagem de convergência para o ótimo e para o tempo computacional. Para as cinco instâncias do *INSURANCE*.

Considerando esses resultados, podemos observar que o MAGA possui um melhor desempenho que o GA, principalmente para instâncias com maiores números de amostras. Para instâncias com menores números de amostras os algoritmos possuem um desempenho próximo, sendo que o MAGA converge mais para o ótimo que o GA, porém possui um desempenho um pouco pior no tempo computacional nas instâncias do *INSURANCE*. Nos demais problemas o MAGA possuiu o desempenho melhor no tempo computacional em todas as instâncias.

4 Conclusões

Esse trabalho teve como objetivo a aplicação do MAGA para o aprendizado estrutural de BNs com grande volume de dados. O MAGA implementado foi comparado com o GA. Para a comparação dos algoritmos, foram testados em três problemas diferentes, sendo

eles o *ASIA*, *CHILD* e o *INSURANCE*, com variações no número de amostras em cada problema, e utilizou-se o número de convergência ao ótimo e o tempo computacional como critérios de avaliação.

Com os experimentos realizados, o MAGA se mostrou vantajoso quando comparado com o GA nas situações avaliadas com maior número de observações. Com os resultados obtidos nesta primeira investigação, é pertinente afirmar que o MAGA, com as configurações apresentadas nesse artigo, parecer ser um algoritmo promissor para aprendizado estrutural de BNs com maiores volumes de dados, cenário esse atual e relevante. Tanto a interação entre os agentes, que se da apenas com os seus vizinhos, quanto o operador de busca tabu, são importantes para os resultados do MAGA. Para trabalhos futuros, podemos verificar as características do MAGA aplicado nesse trabalho através de testes, como no trabalho [21], comparando o desempenho da busca tabu, o MAGA com o autoaprendizado originalmente proposto e o MAGA com busca tabu. Também acreditamos, para trabalhos futuros, ser interessante avaliar o MAGA em outras instâncias de BNs disponíveis e comparar com outros algoritmos propostos na literatura, e aplicar alguma abordagem formal tanto para comparar estatisticamente os resultados como para realizar o ajuste dos parâmetros dos algoritmos avaliados.

5 *Agradecimentos*

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

Referências

[1] M. Bessani, J. A. Massignan, T. M. Santos, J. B. London Jr, and C. D. Maciel. Multiple households very short-term load forecasting using bayesian networks. *Electric Power Systems Research*, 189:106733, 2020.

[2] J. Binder, D. Koller, S. Russell, and K. Kanazawa. Adaptive probabilistic networks with hidden variables. *Machine Learning*, 29(2-3):213–244, 1997.

[3] R. R. Bouckaert. *Bayesian belief networks: from construction to inference*. PhD thesis, 1995.

[4] D. Chickering, D. Geiger, and D. Heckerman. Learning bayesian networks: Search methods and experimental results. In *proceedings of fifth conference on artificial intelligence and statistics*, pages 112–128, 1995.

[5] C. CONTALDI. *Bayesian Network Hybrid Learning Using a Parent Reducing Site-specific Mutation Rate Genetic Algorithm*. PhD thesis, Politecnico di Torino, 2016.

[6] C. Contaldi, F. Vafaee, and P. C. Nelson. Bayesian network hybrid learning using an elite-guided genetic algorithm. *Artificial Intelligence Review*, 52(1):245–272, 2019.

[7] G. F. Cooper and E. Herskovits. A bayesian method for the induction of probabilistic networks from data. *Machine learning*, 9(4):309–347, 1992.

[8] F. A. Fournier, Y. Wu, J. McCall, A. Petrovski, and P. J. Barclay. Application of evolutionary algorithms to learning evolved bayesian network models of rig operations

in the gulf of mexico. In *2010 UK Workshop on Computational Intelligence (UKCI)*, pages 1–6. IEEE, 2010.

[9] S. Gheisari and M. R. Meybodi. Bnc-pso: structure learning of bayesian networks by particle swarm optimization. *Information Sciences*, 348:272–289, 2016.

[10] T. J. Gross, R. B. Araujo, F. A. C. Vale, M. Bessani, and C. D. Maciel. Dependence between cognitive impairment and metabolic syndrome applied to a brazilian elderly dataset. *Artificial intelligence in medicine*, 90:53–60, 2018.

[11] A. A. Hagberg, D. A. Schult, and P. J. Swart. Exploring network structure, dynamics, and function using networkx. In G. Varoquaux, T. Vaught, and J. Millman, editors, *Proceedings of the 7th Python in Science Conference*, pages 11 – 15, Pasadena, CA USA, 2008.

[12] C. R. Harris, K. J. Millman, and S. J. van der Walt et al. Array programming with NumPy. *Nature*, 585(7825):357–362, Sept. 2020.

[13] J.-Z. Ji, H.-X. Zhang, R.-B. Hu, and C.-N. Liu. A tabu-search based bayesian network structure learning algorithm. *Beijing Gongye Daxue Xuebao(Journal of Beijing University of Technology)*, 37(8):1274–1280, 2011.

[14] R. Kabli, F. Herrmann, and J. McCall. A chain-model genetic algorithm for bayesian network structure learning. In *Proceedings of the 9th annual conference on Genetic and evolutionary computation*, pages 1264–1271, 2007.

[15] D. Koller and N. Friedman. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

[16] P. Larrañaga, M. Poza, Y. Yurramendi, R. H. Murga, and C. M. H. Kuijpers. Structure learning of bayesian networks by genetic algorithms: A performance analysis of control parameters. *IEEE transactions on pattern analysis and machine intelligence*, 18(9):912–926, 1996.

[17] S. L. Lauritzen and D. J. Spiegelhalter. Local computations with probabilities on graphical structures and their application to expert systems. *Journal of the Royal Statistical Society: Series B (Methodological)*, 50(2):157–194, 1988.

[18] S. Lee and S. B. Kim. Parallel simulated annealing with a greedy algorithm for bayesian network structure learning. *IEEE Transactions on Knowledge and Data Engineering*, 2019.

[19] H. Li, F. Wang, and H. Li. Integrating expert knowledge for bayesian network structure learning based on intuitionistic fuzzy set and genetic algorithm. *Intelligent Data Analysis*, 23(1):41–56, 2019.

[20] A. Okutan and O. T. Yıldız. Software defect prediction using bayesian networks. *Empirical Software Engineering*, 19(1):154–181, 2014.

[21] C. Peng, G. Wu, T. W. Liao, and H. Wang. Research on multi-agent genetic algorithm based on tabu search for the job shop scheduling problem. *PloS one*, 14(9), 2019.

[22] F. Sahin, M. Ç. Yavuz, Z. Arnavut, and Ö. Uluyol. Fault diagnosis for airplane engines using bayesian networks and distributed particle swarm optimization. *Parallel Computing*, 33(2):124–143, 2007.

[23] M. Scanagatta, A. Salmerón, and F. Stella. A survey on bayesian network structure learning from data. *Progress in Artificial Intelligence*, pages 1–15, 2019.

[24] G. Schwarz et al. Estimating the dimension of a model. *The annals of statistics*, 6(2):461–464, 1978.

[25] M. Scutari. Learning bayesian networks with the bnlearn R package. *Journal of Statistical Software*, 35(3):1–22, 2010.

[26] M. Scutari, C. E. Graafland, and J. M. Gutiérrez. Who learns better bayesian network structures: Accuracy and speed of structure learning algorithms. *International Journal of Approximate Reasoning*, 115:235–253, 2019.

[27] M. Scutari, C. Vitolo, and A. Tucker. Learning bayesian networks from big data with greedy search: computational complexity and efficient implementation. *Statistics and Computing*, 29(5):1095–1108, 2019.

[28] M. B. Sesen, A. E. Nicholson, R. Banares-Alcantara, T. Kadir, and M. Brady. Bayesian networks for clinical decision support in lung cancer care. *PloS one*, 8(12), 2013.

[29] D. J. Spiegelhalter, A. P. Dawid, S. L. Lauritzen, and R. G. Cowell. Bayesian analysis in expert systems. *Statistical science*, pages 219–247, 1993.

[30] X. Sun, C. Chen, L. Wang, H. Kang, Y. Shen, and Q. Chen. Hybrid optimization algorithm for bayesian network structure learning. *Information*, 10(10):294, 2019.

[31] S. Wang and J. Liu. A multi-agent genetic algorithm for improving the robustness of communities in complex networks against attacks. In *2017 IEEE Congress on Evolutionary Computation (CEC)*, pages 17–22. IEEE, 2017.

[32] J. Wu, Y. Guo, H. Zhou, L. Shen, and L. Liu. Vehicular delay tolerant network routing algorithm based on bayesian network. *IEEE Access*, 8:18727–18740, 2020.

[33] Y. Wu. Problem dependent metaheuristic performance in bayesian network structure learning. 2012.

[34] Y. Zhang, M. Zhou, Z. Jiang, and J. Liu. A multi-agent genetic algorithm for big optimization problems. In *2015 IEEE Congress on Evolutionary Computation (CEC)*, pages 703–707. IEEE, 2015.

[35] W. Zhong, J. Liu, M. Xue, and L. Jiao. A multiagent genetic algorithm for global numerical optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 34(2):1128–1141, 2004.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Estudo do comportamento dos efluentes lançados pelo emissário urbano na baía de Santos, SP

Júlia Konflanz Freitas[1,2] and Wiliam Correa Marques[1,2]

[1] *Universidade Federal do Rio Grande, Rio Grande/RS, Brasil*
[2] *Laboratório de Análise Numérica e Sistemas Dinâmicos, Rio Grande/RS, Brazil*

Abstract

Os ecossistemas costeiros têm sido alvo de uma grande pressão causada pelo amplo e acelerado crescimento populacional e econômico das zonas litorâneas. Os efeitos desse crescimento destacam problemas relacionados à pressão demográfica sobre a degradação do meio ambiente, como a disposição final dos esgotos domésticos. Como uma solução para esta problemática, emissários submarinos têm sido considerados uma alternativa eficaz para o descarte dos efluentes provenientes dessas regiões. Entretanto, essa alternativa pode acabar agravando o problema caso não seja bem operada. O presente estudo avaliou, através do uso da modelagem numérica, as taxas de dispersão e contaminação dos efluentes oriundos do emissário submarino de Santos, em São Paulo. Foi utilizado um módulo de poluentes acoplado diretamente ao módulo hidrodinâmico TELEMAC-3D. As concentrações dos contaminantes avaliados se apresentaram, em sua maioria, como elevadas, comprovando que o corpo d'água receptor pode sofrer com alterações, como a contaminação microbiológica, acréscimo de matéria orgânica e o enriquecimento por nutrientes, podendo levar à eutrofização.

Keywords: Modelagem numérica, TELEMAC-3D, emissários submarinos, esgoto, efluentes.

1 INTRODUÇÃO

As zonas litorâneas são áreas que se encontram em constante transformação, resultado da influência de processos naturais, associados com a escala temporal e sobre formas distintas de ocupação e desenvolvimento de atividades antrópicas [15][18]. No Brasil, a elevada e crescente parcela da população que habita municípios de zonas costeiras, somada às pressões socioeconômicas dessas regiões, vem sendo responsável pela grande pressão sobre os ecossistemas litorâneos [1].

Contato: Júlia Konflanz Freitas, juliakonflanz@outlook.com

Em regiões costeiras, a qualidade das águas tem sido prejudicada com o passar dos anos. Práticas como o escoamento de esgoto doméstico diretamente nas praias, na maioria das vezes, sem qualquer tipo de tratamento, acabam gerando um intenso processo de degradação da qualidade das águas da região litorânea. Isto porque o vasto crescimento populacional que atinge estas áreas não tem sido acompanhado por instalações de esgotamento sanitários adequadas [10].

O lançamento inadequado de esgoto doméstico pode representar uma ameaça à sustentabilidade econômica e à qualidade ambiental e de vida das populações, visto que a composição desse efluente costuma apresentar elevadas taxas de sólidos totais e nutrientes [11], além de teores variáveis de contaminantes e outras substâncias potencialmente tóxicas [16].

A elevada competência do ambiente marinho em dispersar e depurar matéria orgânica naturalmente, faz com que emissários submarinos sejam considerados como um recurso efetivo na disposição final de resíduos sanitários. Essa competência se dá devido à alta disponibilidade de oxigênio dissolvido e de energia de correntes para a dispersão de efluentes, além de apresentar a região marinha como um local hostil à sobrevivência de microrganismos [10]

Um emissário submarino tem como função lançar os esgotos sanitários provenientes de uma dada região no meio marinho, a uma distância segura da costa. Essa distância costuma depender do tipo de tratamento dado ao efluente e das condições da área que o receberá, uma vez que regiões sensíveis podem ser incapazes de depurar os resíduos, deixando-os retornar à costa [5].

Apesar de serem apresentados como uma solução, é importante destacar que, caso não sejam bem dimensionados e operados, os emissários submarinos podem provocar danos ambientais. Esses danos podem ser causados através da contaminação microbiológica, do acréscimo por matéria orgânica no meio marinho, aumento da turbidez e enriquecimento por nutrientes, podendo levar à eutrofização da água [14].

A maioria dos 17 emissário submarinos de esgotos domésticos do Brasil se encontra no Estado de São Paulo [14]. Dentre esses, cinco estão localizados na Baixada Santista, sendo três em Praia Grande, um em Santos e um no Guarujá.

Segundo relatórios da Companhia Ambiental do Estado de São Paulo [5][6][7], em Santos, o ponto de lançamento do emissário responsável pela disposição dos efluentes domésticos do município se encontra em uma região ambientalmente sensível, a aproximadamente 5 km da costa. A vulnerabilidade dessa zona se dá devido à presença de um complexo sistema estuarino, composto por diversas atividades antrópicas. Portanto, torna-se necessário que haja um monitoramento mais frequente da qualidade de água na região, e que essa seja reportada aos órgãos ambientais responsáveis.

Neste sentido, o objetivo do presente estudo é avaliar numericamente a pluma de efluentes urbanos lançada pelo emissário submarino de Santos, através dos aspectos hidrodinâmicos da região da Baía de Santos.

2 METODOLOGIA

Neste trabalho, a representação do cenário foi feita através da utilização de modelos numéricos. O cenário empregado é composto pela reprodução da dispersão de efluentes na região da Baía de Santos a partir de um emissário submarino. As simulações utilizadas

foram conduzidas por um período de 366 dias, entre 01 de janeiro e 31 de dezembro de 2012, devido à presença de medições e amostragens de qualidade de água nos relatórios da CETESB [8]. Logo, todas as condições iniciais e de contorno inseridas no sistema de modelagem serão referentes ao ano em questão.

Para a análise dos processos hidrodinâmicos foi utilizado o modelo TELEMAC-3D, que compõe a suíte de modelagem open TELEMAC-MASCARET (www.opentelemac.org). Este modelo tridimensional é utilizado para o estudo de características associadas à hidrodinâmica, transporte de sedimentos, ondas e qualidade de água das regiões costeiras e oceânicas. Além disso, o modelo resolve as equações de Navier-Stokes assumindo ou não as condições de pressão hidrostática, e utiliza o Método de Elementos Finitos para a discretização espacial e vertical em coordenadas sigma, de forma a acompanhar os limites superficiais e de fundo [13].

O TELEMAC-3D leva em consideração a evolução da superfície livre como função do tempo, e utiliza equações de advecção e difusão para a simulação de traçadores, como a salinidade e a temperatura. Os principais resultados obtidos através da aplicação do modelo são a elevação do nível do mar, gerada pela camada superficial da malha computacional, e as componentes das velocidades de corrente e concentrações dos traçadores em cada ponto do domínio.

Para descrever os processos dos efluentes no meio marinho foi utilizado um modelo de poluentes desenvolvido no Laboratório de Análise Numérica e Sistemas Dinâmicos (LANSD), da Universidade Federal do Rio Grande (FURG). Esse módulo numérico tridimensional considera o cálculo de processos de advecção, difusão e decaimento, acoplado diretamente ao modelo hidrodinâmico TELEMAC-3D.

Cada uma das propriedades do esgoto doméstico é considerada como uma equação diferencial de um traçador. Em cada uma dessas propriedades é adicionado um termo fonte na forma de um decaimento ou criação, e o poluente modelado é considerado inerte e completamente miscível na água do mar [12].

O modelo de poluentes representa a concentração de propriedades específicas e, neste contexto, foram considerados como propriedades indicadoras do grau de contaminação da água os coliformes fecais e a Demanda Bioquímica de Oxigênio (DBO) [2]. A concentração de nutrientes como o fósforo e a amônia também foram avaliados. Além disso, os valores típicos de características químicas de esgotos domésticos brutos estão de acordo com Sperling [19], relatórios da CETESB [5][6][7] e com a Resolução nº 357 de março de 2005 do Conselho Nacional do Meio Ambiente (CONAMA) [3].

Os dados impostos no modelo foram inseridos de forma constante no tempo e os valores típicos de características químicas de esgotos domésticos brutos (não tratados) utilizados estão representados na Tabela 1, com suas respectivas referências. A vazão inserida no emissário submarino foi de 7 m^3/s, conforme o relatório da CETESB de 2012 [8].

Uma malha computacional foi construída no software BlueKenue para a realização das simulações (Figura 1). A utilização desse tipo de malha batimétrica possibilita uma boa representação das características batimétricas e morfológicas na grade do modelo numérico. Além disso, otimiza a simulação, permitindo representar, com melhor resolução, regiões de morfologia complexa.

O domínio computacional é composto por polígonos triangulares, totalizando 97.571 nós, sendo que a distância entre os vértices destes elementos varia de 10 km na zona

Tabela 1: Concentrações de contaminantes inseridas no modelo.

Contaminantes	Unidade	Concentração	Fonte
DBO	mg/L	300	Sperling
Nitrogênio Orgânico	mg/L	20	Sperling
Nitrito	mg/L	0,07	CONAMA
Nitrato	mg/L	0,40	CONAMA
Amônia	mg/L	68	CETESB
Fósforo	mg/L	8,3	CETESB
Oxigênio	mg/L	0,40	CONAMA
Coliformes	NMP/100mL	5.10^7	Sperling

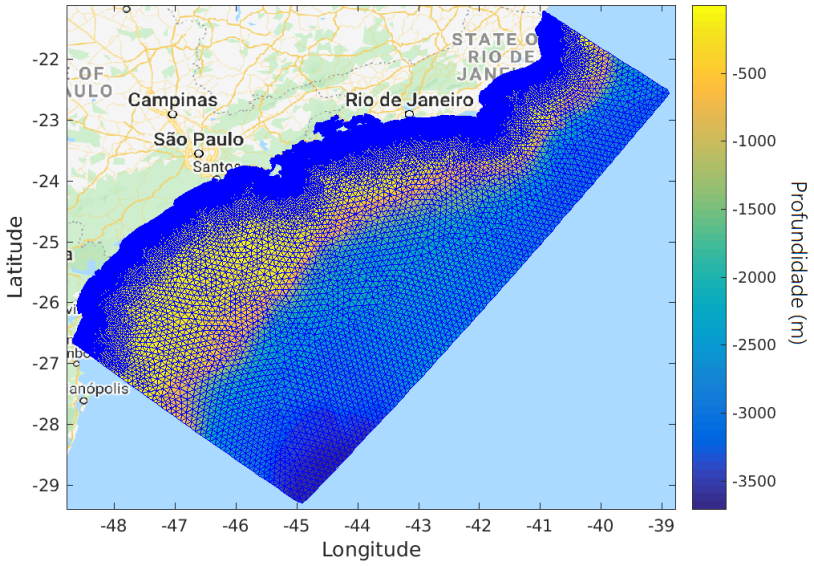


Fig. 1: Malha numérica de elementos finitos.

oceânica e 400 m na região costeira, com 10 níveis sigma. Esta região se estende de Balneário Barra Velha, em Santa Catarina, até Itabapoana, no Rio de Janeiro.

Os dados de batimetria da região de estudo foram extraídos e digitalizados a partir das cartas náuticas fornecidas pela Diretoria de Hidrografia e Navegação (DNH), além do banco de dados proveniente do General Bathymetric Chart of the Oceans (GEBCO). A linha de costa da malha foi obtida através da National Oceanic and Atmospheric Administration (NOAA).

Os dados utilizados como condições iniciais e de contorno para o domínio do modelo hidrodinâmico foram extraídos de diferentes fontes e, posteriormente, interpolados e prescritos para cada ponto da malha de elementos finitos.

Informações globais de circulação oceânica, representadas pelas componentes da velocidade da corrente, salinidade e temperatura da água do mar, utilizadas para definir as condições iniciais do modelo, foram obtidas através do Hybrid Coordinate Ocean Model (HYCOM). A resolução espacial do HYCOM é de 0,083° (cerca de 7 km) de latitude e longitude, com escala temporal de 24 horas.

Dados globais de circulação atmosférica são provenientes do modelo de reanálise ERA-Interim, que foi desenvolvido pelo European Centre for Medium-Range Weather Forecast (ECMWF). A resolução espacial é de 0,125° (aproximadamente 14 km), com escala temporal de 3 horas.

3 VALIDAÇÃO DO TELEMAC-3D

Como forma de avaliar o desempenho do modelo hidrodinâmico TELEMAC-3D, uma validação entre os dados observados em campo e os dados simulados pelo modelo foi realizada. A simulação utilizada para esta etapa foi conduzida por um ano, durante o ano de 2012.

Os dados observados foram obtidos no site do Programa Nacional de Boias (PNBOIA), coordenado pelo Centro de Hidrografia Marinha (CHM) e é uma contribuição brasileira para o GOOS-BRASIL (Global Ocean Observing System). A boia utilizada como parâmetro foi a boia de Santos (São Paulo), fundeada em 200 metros e localizada em 25° 26' 22.2"S de latitude e 45° 2' 9.96"W de longitude.

A validação foi realizada comparando os resultados das componentes longitudinais e transversais das correntes, obtidas no PNBOIA e modeladas pelo TELEMAC-3D. Esses dados foram submetidos a um tratamento preliminar, para remoção de valores muito extremos.

As séries temporais para comparação dos dados obtidos com o PNBOIA e simulados com o TELEMAC-3D estão apresentados nas Figuras 2 e 3. Nestas análises, foram consideradas as componentes longitudinais e transversais da velocidade das correntes, respectivamente.

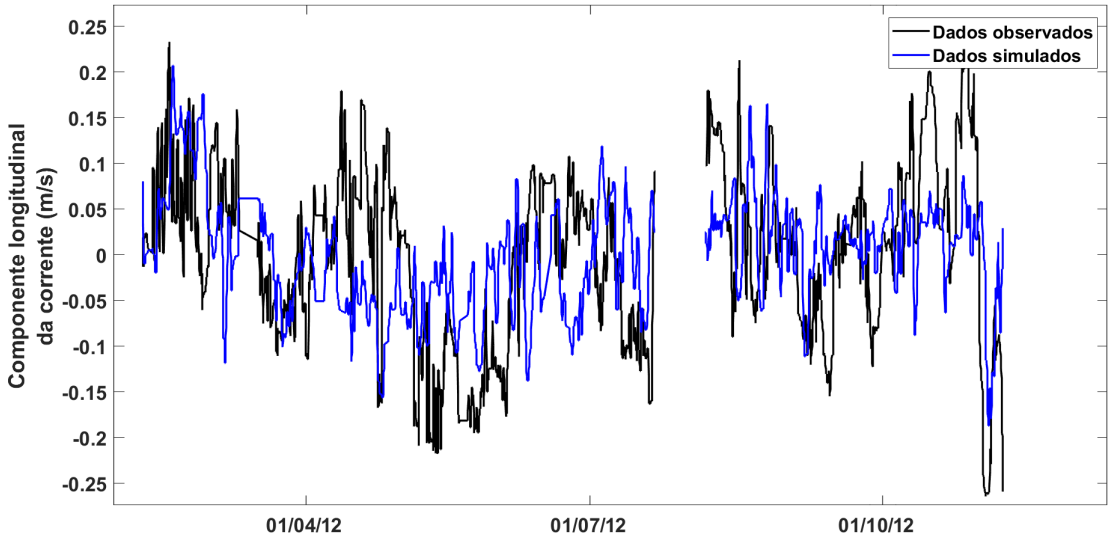


Fig. 2: Série temporal para validação do modelo hidrodinâmico, considerando a componente longitudinal da velocidade das correntes, ao longo do ano de 2012. A linha azul representa os dados simulados, e em preto, os dados observados.

Para a validação da componente longitudinal da velocidade das correntes, foi observado que o modelo TELEMAC-3D apresenta resultados subestimados em relação aos dados ob-

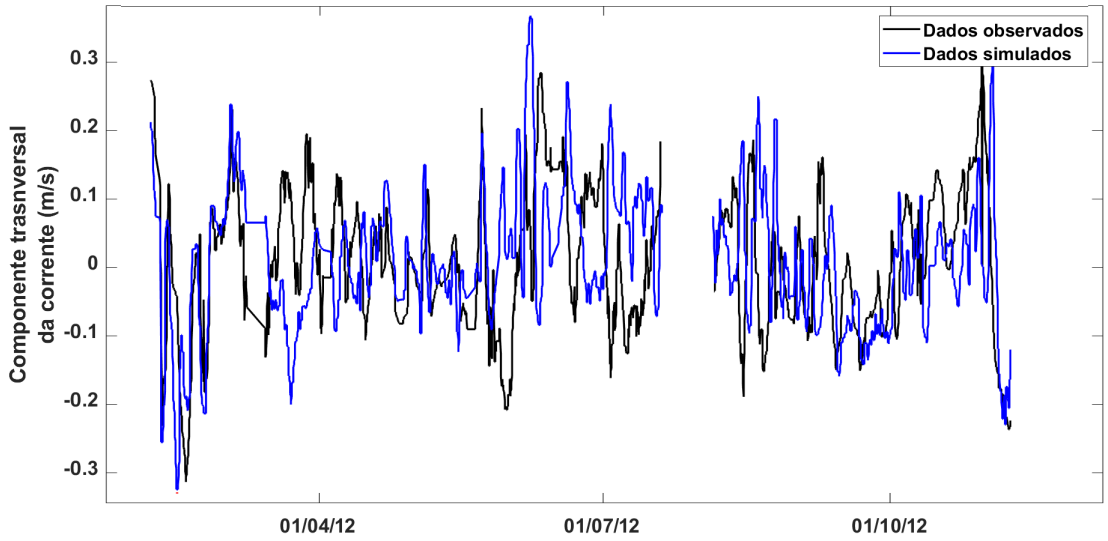


Fig. 3: Série temporal para validação do modelo hidrodinâmico, considerando a componente transversal da velocidade das correntes, ao longo do ano de 2012. A linha azul representa os dados simulados, e em preto, os dados observados.

servacionais, principalmente no mês de abril e a partir de outubro. Entretanto, o modelo representou adequadamente as tendências de aumento e decréscimo dos parâmetros analisados, demonstrando os padrões sazonais.

Em relação à componente transversal da velocidade das correntes, os resultados obtidos com o modelo e coletados em campo se mostraram muito próximos, ao longo de todo o ano de 2012. Em alguns períodos, o modelo apresentou valores mais altos ou mais baixos que os observados, mas de forma pouco significativa.

De modo geral, as séries temporais demonstraram uma boa concordância entre os dados modelados e simulados, ao longo de 2012, na posição da Boia de Santos. Algumas discrepâncias entre os resultados foram verificadas, mas que não interferem significativamente nas análises propostas no trabalho.

Para complementar os resultados, foi realizada uma análise quantitativa do desempenho do modelo numérico, através de índices estatísticos de performance. Os resultados destes índices estão apresentados na Tabela 2 e consideram o Erro Absoluto Médio (MAE), o Erro Quadrático Médio (MSE), a Raiz do Erro Quadrático Médio (RMSE), o Viés e a Relação de Variância (RVAR).

Para todos os indicadores de performance, com exceção do RVAR, o resultado desejado é um valor próximo a zero, indicando uma concordância significativa entre os dados simulados pelo TELEMAC-3D e os dados observacionais do PNBOIA. Em relação ao RVAR, o ideal é que os valores resultem próximos a 1, indicando que as variâncias entre os dados modelados e observados são iguais.

Os resultados apresentados na Tabela 2, demonstram que todos os índices atingiram os valores desejados, se aproximando de zero ou de 1, quando necessário. Os índices de erro (MAE, MSE e RMSE) atingiram valores muito baixos, praticamente nulos, nas duas componentes de velocidade analisadas.

Tabela 2: Indicadores de performance utilizados para validação do modelo hidrodinâmico, considerando as componentes longitudinais e transversais da velocidade das correntes.

Indicadores de performance	Componente longitudinal	Componente transversal
$\text{MAE} = \sqrt{\frac{\sum \lvert obs-mod \rvert}{n}}$	0,3183	0,4112
$\text{MSE} = \frac{1}{n} * \sum (obs - mod)^2$	0,1209	0,2163
$\text{RMSE} = \sqrt{\frac{\sum (obs-mod)^2}{n}}$	0,3478	0,4651
$\text{Viés} = \frac{\sum obs-mod}{n}$	-0,3161	-0,3976
$\text{RVAR} = \frac{mod_{var}}{obs_{var}}$	0,4270	0,5194

Em relação ao Viés, os valores positivos representam que os resultados do modelo estão superestimados em relação aos dados de campo, e os valores negativos, demonstram uma subestimação. Nas análises da validação, foi concluído que o modelo está subestimado em relação aos dados da boia, conforme verificado anteriormente nas séries temporais. Entretanto, os valores são muito próximos a zero, não demonstrando uma diferença considerável para as duas componentes.

A variância é uma medida de dispersão que mostra o quão distante os resultados estão da sua média, logo, o RVAR deve ser próximo a 1, indicando que as variâncias entre dados observados e modelados é igual. Para valores abaixo de 1, os resultados indicam que os dados modelados estão subestimados em relação aos dados de campo, e para valores acima de 1, estão superestimados.

Conforme discutido anteriormente nas séries temporais e no resultado do Viés, segundo o índice de performance RVAR, os dados obtidos com o modelo TELEMAC-3D estão subestimados em relação aos dados da Boia de Santos. Entretanto, as diferenças não são significativas, e os valores se aproximam de 1 nas duas componentes da velocidade das correntes analisadas.

De modo geral, a componente longitudinal da velocidade obteve melhores resultados em comparação à componente transversal, com diferenças apenas decimais. Portanto, a verificação destes índices indica que o modelo apresentou uma boa reprodutibilidade para as componentes longitudinais e transversais da velocidade das correntes, quando comparado aos dados observacionais da Boia de Santos.

4 RESULTADOS E DISCUSSÃO

Os resultados apresentam uma média para todo o ano de 2012. Sendo assim, a Figura 4.A apresenta um histograma de frequência da intensidade média do vento e a Figura 4.B, a média do padrão de correntes (vetores) e elevação média do nível do mar. A Figura 5.A representa a concentração média da Demanda Bioquímica de Oxigênio (DBO) e a Figura 5.B, a concentração média de coliformes para o ano de 2012 na Baía de Santos. Já a Figura 6.A representa a concentração média de fósforo e a Figura 6.B, a concentração média de amônia para o mesmo período e região.

O nível do mar apresenta padrão médio de elevação em direção à costa, comportamento característico relacionado com a direção predominante da corrente. Na costa, a direção preferencial da corrente obtida se deve à incidência de ventos do quadrante norte

(totalizando mais de 90% da frequência obtida), com intensidades mais fortes nas direções de noroeste e nordeste. Este padrão gera correntes residuais direcionadas, em maioria, para o oeste e sudoeste da baía de Santos, que influenciam diretamente na elevação do nível do mar e, consequentemente, na dispersão dos efluentes lançados.

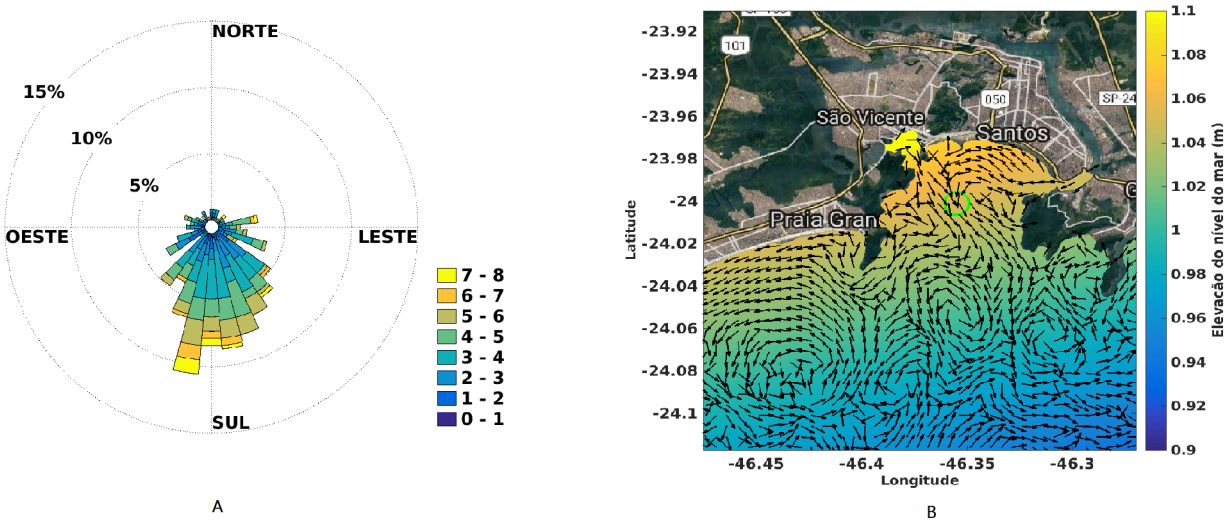


Fig. 4: A) Histograma de frequência da intensidade do vento e B) representação da média do padrão de correntes (vetores) e elevação média do nível do mar

Na Figura 5.A é possível observar que a concentração de DBO varia de 0 a 4 mg/L, possuindo seu valor mais elevado no local de desembocadura do emissário submarino e atingindo, com teores menores, a região de Praia Grande.

A DBO representa a quantidade necessária de oxigênio utilizada na oxidação bioquímica da matéria orgânica na água. Portanto, quanto maior a quantidade de matéria orgânica presente na água, maior será a DBO, e consequentemente, maior a poluição, visto que ocorrerá uma diminuição de oxigênio dissolvido na água [4].

De acordo com a legislação, a DBO máxima no esgoto deve ser de 60 mg/L. Em ambientes naturais não poluídos, a concentração de DBO é baixa (1 mg/L a 10 mg/L). Sendo assim, a concentração da DBO aponta para um baixo teor de matéria orgânica presente na Baía de Santos.

Entretanto, apesar do baixo teor de DBO na região de estudo, a concentração de coliformes varia de 0 a 1000 NMP/100 mL na mesma área (Figura 5.B). Todavia, não apresenta uma zona de dispersão tão ampla quanto a DBO.

A OMS recomenda que as bactérias do grupo coliformes fecais sejam utilizadas como parâmetro microbiológico de qualidade da água quando se deseja mensurar a presença de organismos patogênicos [17].

O contato com águas contaminadas por esgoto doméstico sujeita as pessoas a contraírem doenças devido à presença de enterobactérias contidas nestes dejetos. Exemplos comuns são as infecções de olhos, ouvidos e gargantas. Tais bactérias são de difícil detecção, por isso comumente se adota como indicador de contaminação fecal, as bactérias do grupo coliformes fecais [19].

A concentração de coliformes não deve exceder um limite de 4000 NMP/100 mL. Contudo, o valor de 1000 NMP/100 mL já representa uma possível presença de organismos patogênicos no corpo d'água.

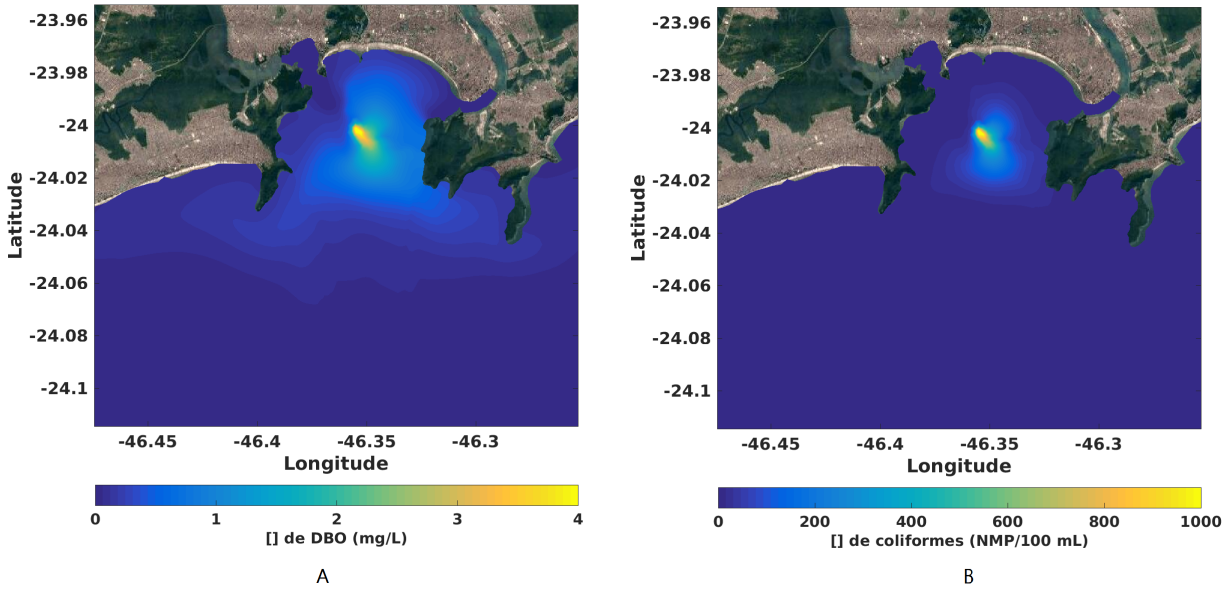


Fig. 5: A) Concentração da Demanda Bioquímica de Oxigênio e B) concentração de coliformes fecais.

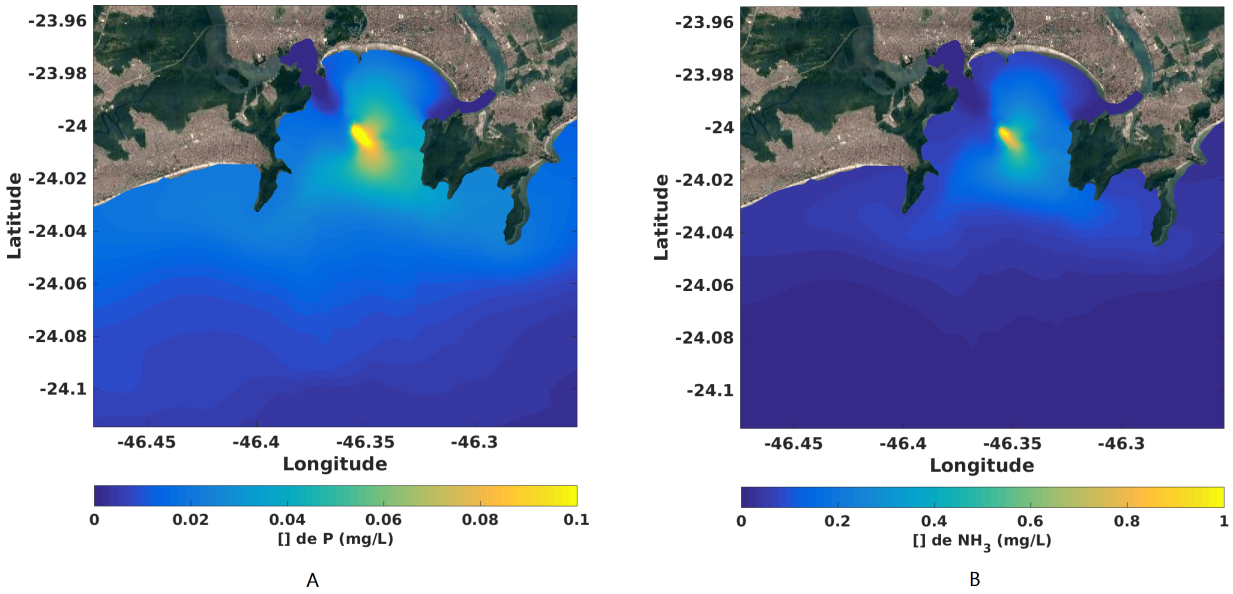


Fig. 6: A) Concentração de fósforo e B) concentração de amônia

Na Figura 6.A está representada a concentração de fósforo na região de lançamento

de efluentes, variando de 0 a 0,1 mg/L. Sua dispersão é bastante ampla e atinge tanto a região de Praia Grande quanto do Guarujá.

A importância do fósforo no meio aquático associa-se principalmente ao fato dele ser um nutriente essencial para o crescimento de algas, podendo, por isso, em certas condições, conduzir a fenômenos de eutrofização da água [19].

No Brasil, a legislação do CONAMA [3] estabelece que o nível crítico de fósforo total na água é de 0,075 mg/L na classe mais delicada. Portanto, o nível de 0,1 mg/L é considerado muito elevado.

A concentração de amônia está representada na Figura 6.B, com uma variação de 0 a 1 mg/L. Sua dispersão alcança as regiões adjacentes, porém, com uma concentração menor.

Altas concentrações de amônia podem ter implicações ecológicas nos sistemas aquáticos, pois no processo de nitrificação, a quantidade de oxigênio dissolvido diminui, sendo prejudicial principalmente para os peixes.

Concentrações de 0,25 mg/L ou maiores afetam o crescimento dos peixes, porém para que ocorra sua morte, o nível de amônia deve ser superior a 0,5 mg/L [9].

5 CONCLUSÕES

O presente estudo teve como objetivo avaliar numericamente os processos hidrodinâmicos relacionados à emissão de efluentes urbanos, através do emissário submarino de Santos. Além disso, foram identificadas as condições de contaminação ambiental na região, assim como, a disseminação de sua influência sobre a região da Baía de Santos e zonas adjacentes.

Em relação aos resultados hidrodinâmicos, foi constatado que ação dos eventos, das correntes e a elevação do nível do mar, influenciam diretamente e indiretamente na dispersão dos efluentes lançados pelo emissário submarino de Santos. Portanto, a incidência dos ventos na região, permite a caracterização de uma maior ou menor poluição na zona praial.

As características hidrodinâmicas da região direcionam os efluentes lançados pelo emissário submarino, causando um comportamento semelhante entre a dispersão dos mesmos. Isto já era esperado, visto que todos se encontram sob a influência dos mesmos padrões de direção de correntes.

Por fim, o presente estudo poderá contribuir para o estado da arte da região, fornecendo subsídios para o monitoramento da qualidade da água e impactos gerados por emissários submarinos. Em análises futuras, outros parâmetros de qualidade de água serão analisados, considerando um período mais longo de tempo.

6 *Agradecimentos*

Os autores agradecem à Universidade Federal do Rio Grande (FURG) e ao Laboratório de Análise Numérica e Sistemas Dinâmicos (LANSD) pelo fornecimento de recursos computacionais para o desenvolvimento deste trabalho. Ao consórcio Open TELEMAC-MASCARET por disponibilizar gratuitamente o sistema TELEMAC, ao Laboratório Nacional de Computação Científica (LNCC) pela disponibilidade para uso do Supercomputador Santos Dumont, e ao Centro Nacional de Supercomputação (CESUP) da Universidade Federal do Rio Grande do Sul (UFRGS) pelo apoio ao desenvolvimento desta pesquisa.

Este estudo foi apoiado pela Fundação de Amparo à Pesquisa do Estado do Rio Grande do Sul (FAPERGS), e pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq).

Referências

[1] F. B. L. Abreu, F. P. Vasconcelos, and M. F. Albuquerque. A Diversidade no Uso e Ocupação da Zona Costeira do Brasil: A Sustentabilidade como Necessidade. *Conexões - Ciência e Tecnologia*, 11(5):8–16, 2011.

[2] S. C. Chapra. *Surface Water-Quality Modeling*, volume 1. Waveland Press, Inc., 1997.

[3] CONAMA. *Resolução nº 357 de 17 de março de 2005*. Conselho Nacional do Meio Ambiente, 2005.

[4] C. D. L. D. N. Cunha and A. P. Ferreira. Modelagem matemática para avaliação dos efeitos de despejos orgânicos nas condições sanitárias de águas ambientais. *Cad. Saúde Pública*, 22(8):1715–1725, 2006.

[5] C. A. do Estado de São Paulo (CETESB). *Relatório de monitoramento de emissários submarinos*, volume 1. CETESB, 2007.

[6] C. A. do Estado de São Paulo (CETESB). *Relatório de qualidade das águas litorâneas no estado de São Paulo: balneabilidade das praias em 2007*, volume 1. CETESB, 2008.

[7] C. A. do Estado de São Paulo (CETESB). *Relatório de qualidade das águas litorâneas no estado de São Paulo: balneabilidade das praias em 2008*, volume 1. CETESB, 2009.

[8] C. A. do Estado de São Paulo (CETESB). *Zona costeira paulista: relatório de qualidade ambiental 2012*, volume 1. CETESB, 2012.

[9] F. D. A. Esteves. *Fundamentos de Limnologia*. Interciência, 1998.

[10] R. C. Feitosa. Emissários submarinos de esgotos como alternativa à minimização de riscos à saúde humana e ambiental. *Ciênc. saúde coletiva*, 22(6):2037–2048, 2017.

[11] F. B. Gonçalves and A. P. Souza. *Disposição Oceânica de Esgotos Sanitários. História, Teoria e Prática*, volume volume. Associação Brasileira de Engenharia Sanitária, 1997.

[12] J. Harrari and M. Gordon. Simulações Numéricas da Dispersão de Substâncias no Porto e Baía de Santos, sob a Ação de Marés e Ventos. *Revista Brasileira de Recursos Hídricos*, 6(4):115–131, 2001.

[13] J. M. Hervouet. *Hydrodynamics of Free Surface Flows: Modelling with the Finite Element Method*, volume 1. John Wiley Sons Ltd, 2007.

[14] C. C. Lamparelli. *Desafios para o licenciamento e monitoramento ambiental de emissários: a experiência de São Paulo. In: Lamparelli, C. and Ortiz, J. Emissários submarinos: projeto, avaliação de impacto ambiental e monitoramento*, volume 1. CETESB, 2006.

[15] K. Mello, R. H. Toppa, D. M. de Souza Abessa, and M. Castro. Dinâmica da expansão urbana na zona costeira brasileira: o caso do município de São Vicente, São Paulo, Brasil. *Revista da Gestão Costeira Integrada*, 13(4):539–551, 2013.

[16] B. R. Rachid, E. C. Souza, C. J. David, and D. M. Abessa. Ensaios de toxicidade utilizando efluentes domésticos lançados através de emissários submarinos na Baixada Santista, SP. *Anais do 4º Simpósio de Ecossistemas Brasileiros*, 1(104):378–85, 1998.

[17] T. Saita, P. Natti, E. Cirilio, N. Romeiro, M. Candezano, R. Acuna, and L. Moreno. Simulação Numérica da Dinâmica de Coliformes Fecais no Lago Luruaco, Colômbia. *Tendências em Matemática Aplicada e Computacional*, 3:435–447, 2017.

[18] J. S. Silva and M. S. F. Filho. Instrumentos legais de prevenção de impactos ambientais na zona costeira: estratégias integradas de gestão territorial. *Rev. Eletrônica Mestr. Educ. Ambient.*, 32(2):7–26, 2013.

[19] M. V. Sperling. *Introdução à qualidade das águas e ao tratamento de esgotos*, volume 1. UFMG, 1996.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Forecasting Dengue Fever in Brazil: An Assessment of Climate Conditions

Lucas M. Stolerman[1], Pedro D. Maia[2] e J. Nathan Kutz[3]

[1] *Department of Mechanical and Aerospace Engineering, University of California San Diego, California, USA.*
[2] *Department of Mathematics, University of Texas at Arlington, Texas, USA.*
[3] *Department of Applied Mathematics, University of Washington, Seattle, USA*

Abstract

Local climate conditions play a major role in the biology of the *Aedes aegypti* mosquito, the main vector responsible for transmitting dengue, zika, chikungunya and yellow fever in urban centers. For this reason, a detailed assessment of periods in which changes in climate conditions affect the number of human cases may improve the timing of vector-control efforts. In this work [1], we develop new machine-learning algorithms to analyze climate time series and their connection to the occurrence of dengue epidemic years for seven Brazilian state capitals. Our method explores the impact of two key variables – frequency of precipitation and average temperature – during a wide range of time windows in the annual cycle. Our results indicate that each Brazilian state capital considered has its own climate signatures that correlate with the overall number of human dengue-cases. However, for most of the studied cities, the winter preceding an epidemic year shows a strong predictive power. Understanding such climate contributions to the vector's biology could lead to more accurate prediction models and early warning systems.

Keywords: dengue forecasting, early-warning systems, support vector machines

Referências

[1] L. M. Stolerman, P. D. Maia, and J. N. Kutz. Forecasting dengue fever in brazil: An assessment of climate conditions. *PLOS ONE*, 14(8):1–16, 08 2019.

Contato: Lucas M. Stolerman, lstolerman@eng.ucsd.edu

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Modelagem de Energia de Ondas em Zonas Costeiras

Luciano Garim Garcia[1,2], Vinícius Lôndero[1], Márcio Cardoso Junior[1] e Ariane Santos da Silveira[1]

[1] *Universidade do Vale do Rio dos Sinos, São Leopoldo/RS, Brasil*
[2] *Universidade Federal do Rio Grande do Sul, Porto Alegre/RS, Brasil*

Abstract

Neste trabalho apresentamos um modelo simplificado de ondas oceânicas baseado em teoria linear, com a finalidade de estimar a energia dessas ondas em ambientes costeiros. Para isto, utilizamos a equação de dispersão de Airy com aproximações para águas rasas e profundas. Como fatores de dissipação de energia, levamos em consideração fatores como quebra de ondas e velocidade orbital máxima de fundo. As equações abordadas são resolvidas numericamente em um grid obtido a partir de modelo digital de elevação (DEM) da Grande Barreira de Corais Australiana. A partir de uma simulação, constatamos que as atenuações de energia acompanharam a morfologia da área em estudo. Para evidenciar este fato, exibimos um corte na área simulada para analisar variação de altura ondas.

Keywords: Energia, Ondas, Dissipação, Airy.

1 INTRODUÇÃO

O comportamento de ondas em zonas costeiras têm sido amplamente discutido e modelado matematicamente pela comunidade científica. Tais zonas constituem regiões, entre os continentes e os oceanos, caracterizadas pela natureza geológica dos continentes (litologias e arcabouços tectônicos) e principalmente pela energia imposta pela variação do nível médio do mar. Por meio dos fenômenos de refração, difração e reflexão, as ondas tendem a amplificar e a transformar a dinâmica na zona costeira por meio da distribuição e dissipação de sua energia ao longo da costa, uma vez que estas respondem, sobretudo, pelo transporte de sedimentos litorâneos e pelos processos erosivos e deposicionais [6].

A forma mais comum de ondas são as geradas pela ação dos ventos sobre a superfície do oceano. A energia é transferida do vento para a água quando este atua sobre a superfície do oceano resultando na formação de ondas. Os modelos matemáticos utilizados

Contato: Luciano Garim Garcia, lucianogarim@gmail.com

para a simulação de ondas geradas pelos ventos são divididos de acordo com a interação entre os componentes do espectro de energia. Os modelos de primeira ordem calculam os parâmetros de onda (altura, direção e período) de forma independente. Os de segunda ordem calculam a interação de forma parametrizada, impedindo o crescimento independente dos componentes do espectro [10]. Os modelos de terceira ordem não possuem nenhuma restrição da interação dos componentes do espectro e são baseados na equação do balanço de energia. Porém, modelos de segunda e terceira ordem demandam um processamento de dados muito maior em relação aos modelos de primeira ordem. Sendo assim, o uso de modelos simplificados e de bom desempenho se faz necessário.

A região em que as ondas são geradas pelo vento são chamadas de pistas de vento (fetch). Os primeiros cálculos de ondas geradas pelo vento foram realizados no projeto JONSWAP (Joint North Sea Wave Project)[8], em que, baseado em dados observacionais, foram desenvolvidas relações empíricas. O aumento da intensidade do vento e da pista produz maiores alturas de onda, mas existe um limite para o crescimento; este limite ocorre quando a velocidade de fase da onda atinge a velocidade do vento em superfície. Quando ambos propagam-se com a mesma velocidade, o vento não transfere mais energia para o oceano, atingindo o estágio de maturação (ou desenvolvimento total) [7]. Há diversas formulações para a altura significativa das ondas em estágio de maturação ou desenvolvimento total. Observações feita por [3] comprovam a existência do limite, ou seja, dado um valor de velocidade de vento, mesmo que a pista aumente, a altura significativa da onda (e o período) não ultrapassa o valor limite.

Baseado nas limitações de altura significativa de onda, apresentamos um modelo simplificado de propagação de ondas de primeira ordem para estimar a variação de energia superficial em zonas costeiras. A partir dessa altura limitante, a onda tende a se propagar em direção a regiões costeiras por meio de frentes de ondas, atenuando seu tamanho em função da morfologia da área simulada. Sendo assim, podemos estimar a variação de energia ao longo destas frentes.

O modelo será apresentado na Seção 2 e sua abordagem numérica será exibida na Seção 3. Finalmente, na Seção 4 apresentaremos uma simulação gerada a partir do modelo descrito.

2 MODELO DE ONDAS

Uma descrição matemática simples de ondas é atribuída aos trabalhos de Airy em 1845 [1]. Sua teoria é aplicável a condições em que a altura da onda é pequena se comparada ao seu comprimento e a profundidade da água. É comumente referida como teoria linear ou de primeira ordem das ondas, devido às suposições simplificadoras feitas em sua derivação.

A Figura (1) mostra uma onda sinusoidal, de comprimento L e altura H. A frequência angular ω e o número de onda k são descritos pelo período T e comprimento de onda L, respectivamente. A medida de profundidade d é tomada em função do nível de água calma SWL e o fundo oceânico.

A principal equação a ser utilizada para a modelagem de ondas será dada pela relação de dispersão para ondas lineares:

$$\omega^2 = gk \tanh(kd). \tag{1}$$

Esta equação trata da separação das ondas devido as diferenças de velocidade e de

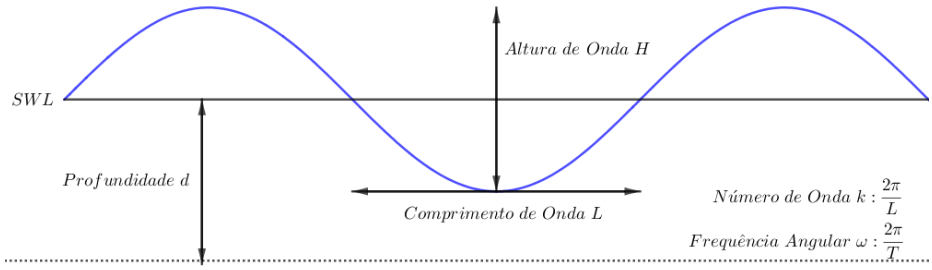


Fig. 1: Modelo de onda linear.

direção, que particularmente decorrem de diferenças na frequência da onda. Além disso, ela expressa uma única relação entre ω, k e d (ou T, L e d). Se duas dessas variáveis são conhecidas, a terceira estará unicamente definida. Reescrevendo a Equação (1) em termos do comprimento e período de onda, temos:

$$\left(\frac{2\pi}{T}\right)^2 = g\frac{2\pi}{L}\tanh\left(\frac{2\pi d}{L}\right), \tag{2}$$

$$L = \frac{gT^2}{2\pi}\tanh\left(\frac{2\pi d}{L}\right). \tag{3}$$

A velocidade de fase ou celeridade de ondas C é definida como a razão entre o comprimento de onda e seu período, então dividindo a Equação (3) por T, obtemos:

$$C = \frac{gT}{2\pi}\tanh\left(\frac{2\pi d}{L}\right). \tag{4}$$

Para valores dados de ω e k, a onda se propagará mais rapidamente em águas profundas do que em águas rasas. Este fato explica porque ondas normalmente chegam com suas cristas paralelas à praia: uma onda em mar aberto com direção oblíqua à praia tenderá a dobrar-se, já que sua parte mais distante da praia, e portanto sujeita a profundidades maiores, será mais rápida, emparelhando com a parte da onda que estará em águas mais rasas e portanto, mais lenta. Este processo de mudança de direção de ondas chama-se refração [5].

Em teoria de ondas aquáticas é comum separar níveis de profundidade em função do comprimento de ondas, onde tais discretizações ocorrem para dois ou três níveis em geral. Essas restrições trazem simplificações no cálculo de funções hiperbólicas envolvidas no modelo físico e são utilizadas aqui.

Tabela 1: Discretização da profundidade em função do comprimento de ondas.

Profundidade	Restrição	$\tanh\left(\frac{2\pi d}{L}\right)$	C
Águas Rasas	$\frac{d}{L} \leq \frac{1}{20}$	$\frac{2\pi d}{L}$	$\sqrt{gd}$
Águas Profundas	$\frac{d}{L} \geq \frac{1}{2}$	1	$\frac{gT}{2\pi}$

Na Tabela 1 são apresentadas duas restrições, uma para águas rasas e outra para águas profundas. Em ambas restrições a função tangente hiperbólica possui comportamento assintótico, fazendo que seja possível estabelecer uma simplificação da Equação (4) para esses níveis de profundidade. Observa-se que em situações de águas rasas, o comprimento de onda é uma função da profundidade e do período, decrescendo à medida que ambos diminuem. No entanto, a celeridade da onda C depende apenas da profundidade e independe do período. Por isso, verifica-se que as ondas diminuem e abrandam à medida que se aproximam do litoral. O comprimento de onda em águas profundas e a correspondente celeridade são funções apenas do período, ambas as grandezas crescendo à medida que o período aumenta. Por isso, verifica-se que as ondas aumentam e são mais rápidas à medida que estão mais afastadas do litoral [5]. A fim de formalizar essa dinâmica das ondas, recorremos a definição de frentes de ondas.

Frente de onda é a região do espaço que reúne todos os pontos fonte da onda que estão em fase e a um mesmo número de comprimentos de onda da fonte. Segundo [4], cada ponto de uma frente de onda possui a funcionalidade de uma nova fonte pontual. O conhecimento da velocidade e do período das ondas permite calcular a trajetória da onda, para um determinado conjunto de fontes de ondas e distribuição de profundidade [14]. A direção de propagação é perpendicular a frente de ondas, implicando que uma deformação na frente de ondas gere uma variação de direção de propagação. Considerando a direção e o tempo de viagem das ondas é possível estimar a altura de onda, conhecendo a fonte inicial [9].

A energia carregada por uma onda no oceano é a soma das energias potencial e cinética, e é cotada como a energia total por unidade de área da superfície do mar. Pela teoria de Airy, a energia potencial E_p e cinética E_k são iguais. Portanto, a energia E por unidade de área do oceano é:

$$E = \frac{1}{8}\rho g H^2, \tag{5}$$

onde g é a constante gravitacional e ρ é a densidade da água. Deduzimos da Equação (5), que a variação de energia está diretamente relacionada com a altura de onda, sendo assim é necessário analisar quando acontecem mudanças nesse parâmetro. Estudos como de [12] apontam que em um mar plenamente desenvolvido, as ondas atingem sua altura máxima, e conforme aproximam-se da costa diminuem de tamanho devido a quebra de ondas, fator que está relacionado com a profundidade. O índice de quebra γ, pode ser expresso por:

$$\gamma = \frac{H}{d} = 0.78, \tag{6}$$

onde na prática $0.4 < \gamma < 1.2$ [11]. Assim, quando $\gamma > 0.78$ acontece a quebra de onda, causando diminuição no seu tamanho e, consequentemente, dissipando energia. Além desse fator de dissipação, existem outros fenômenos físicos que influenciam na atenuação de energia, principalmente fatores relacionados com o fundo oceânico.

Uma onda que se propaga em águas profundas tem suas partículas percorrendo uma trajetória circular quase fechada. Nesse caso, o diâmetro orbital da trajetória em superfície corresponde à altura da onda, decrescendo exponencialmente até uma profundidade equivalente à metade do comprimento da onda. A partir daí, desconsidera-se o diâmetro

orbital, e considera-se que o deslocamento das partículas da água deixa de existir. Em águas com profundidades menores que a metade do comprimento da onda, águas rasas, a onda interage com o fundo oceânico e as órbitas se tornam cada vez mais achatadas, tomando forma de elipse.

Ondas que fluem em águas rasas produzem uma velocidade oscilatória no fundo do mar denominada velocidade orbital de fundo. Usando a aproximação para águas rasas e a teoria de Airy, temos a velocidade orbital máxima de fundo dada por:

$$U = \frac{H}{2}\sqrt{\frac{g}{d}} \tag{7}$$

Para o nosso modelo de ondas usaremos U como um parâmetro de atenuação de energia levando em consideração seu comportamento em relação a altura de onda. Na Figura (2), podemos ver o comportamento assintótico de U para diferentes alturas H e profundidades d. Note que quanto maior a profundidade da lâmina d'água menor será o valor de U, tendo o comportamento de uma função monótona decrescente. Vemos este mesmo decaimento para outros valores de H variando de 1 m a 5 m. Para um valor de H de 2 m, U depende apenas de d, tendo um comportamento assintótico ao se aproximar de zero. A fim de analisar com mais detalhe essas relações, consideramos uma normalização de U de modo que $U \in [0, 1]$.

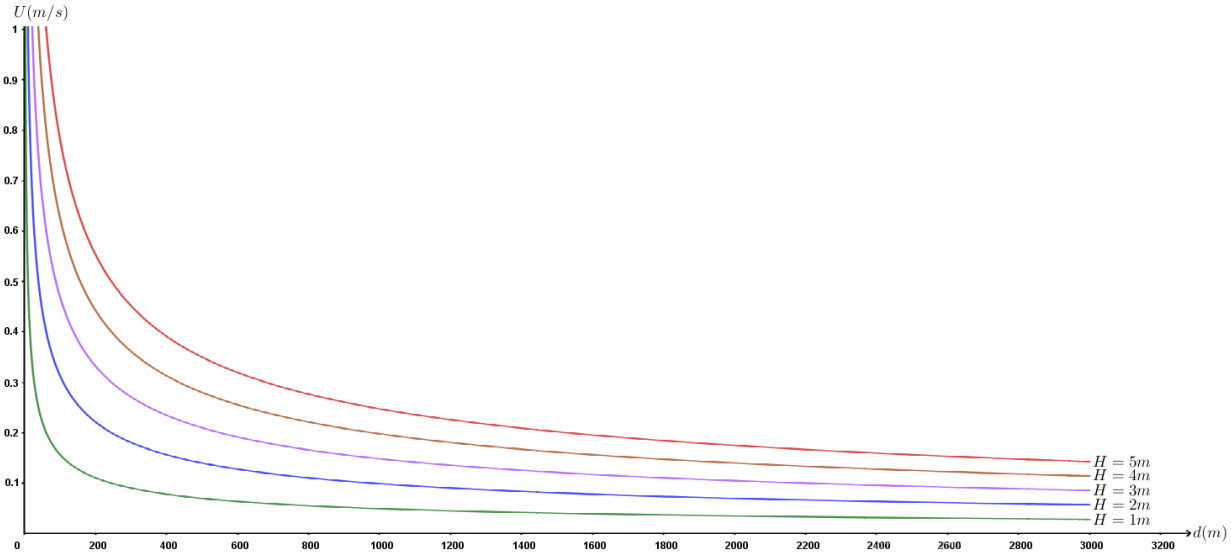


Fig. 2: Velocidade orbital máxima de fundo para 5 alturas de ondas.

Com o auxílio de limites de funções de duas variáveis escrevemos a Tabela 2, a qual retrata que a altura de onda é máxima quando a profundidade é grande (usando ∞ como abuso de notação), assim como a altura é mínima quando a onda se aproxima da costa. Essa relação nos permite entender a altura da onda, não somente em função da profundidade mas também da velocidade orbital máxima. Suponha que tenhamos uma altura significativa inicial onda H_0 que é máxima para uma dada região do oceano. A fim de atenuar sua magnitude consideramos a Equação:

$$H = H_0(1 - U), \tag{8}$$

Tabela 2: Relações assintóticas entre U, d e H

U	d	H
≈ 0	$\approx \infty$	$H_{\max}$
≈ 1	≈ 0	$H_{\min}$

onde H é a nova altura de onda obtida em função de U. Note que quando $U \to 1 \Rightarrow H \to 0$ e quando $U \to 0 \Rightarrow H \to H_0$. Definindo a variação de altura como $\Delta H = H_0 - H$, temos:

$$\Delta H \approx \begin{cases} 0 & , \text{ quando } U \to 0 \\ H_0 & , \text{ quando } U \to 1 \end{cases} \tag{9}$$

A partir da definição de ΔH vemos que as variações na altura das onda acontecem quando a velocidade orbital máxima atinge seus maiores valores, ocorrendo em regiões rasas. Já em regiões de maior profundidade a altura de onda se mantém quase constante, variando muito pouco. Como a energia de ondas é proporcional a altura de onda ao quadrado, então essa variação dada por ΔH implica em uma atenuação de energia. Sendo assim, para o nosso modelo de ondas os fatores de dissipação de energia serão dados a partir da quebra de ondas e da velocidade orbital máxima no fundo. Na próxima seção será abordado o método numérico a ser utilizado no modelo.

3 MODELO NUMÉRICO

Para a caracterização do modelo numérico considere um grid de tamanho $n \times m$ com n linhas e m colunas. Considere cada nó do grid dado por uma entrada (i, j), onde $i = 1, ..., n$ e $j = 1, ..., m$. A distância a cada dois nós adjacentes possui valor 1, e em relação a nós não-adjacentes pode ser obtida via teorema de Pitágoras. A distância física entre dois nós do grid será dada em função do tamanho célula $cell_{size}$. Por exemplo, se o nó (i, j) é a adjacente ao nó $(i + 1, j)$ sua distância no grid será 1 e sua distância física será $1 \times cell_{size}$.

Para a inicialização do método é fornecido uma altura significativa de onda H_0 e uma direção inicial α de propagação. Desse modo, inicializamos os seguintes parâmetros:

$$T_0 = \max\left\{0.47H_0 + 6.76, \pi^2\sqrt{\frac{H_0}{g}}\right\}, \tag{10}$$

$$L_0 = \frac{gT_0^2}{\pi},\ f_0 = \frac{2\pi}{T_0},\ k_0 = \frac{2\pi}{L_0},\ c_0 = \frac{gT_0}{2\pi}. \tag{11}$$

Com base nesta direção de propagação os pontos fonte das ondas são inicializados.

Na Figura 3 os pontos em vermelho representam as fontes inicias de ondas, formando uma frente de ondas linear. Os pontos em azul não possuem atribuição, e conforme as ondas se propagam estes pontos vão recebendo os tempos de viagem das ondas. Cada ponto azul que é alcançado por uma frente de ondas torna-se uma nova fonte de ondas de acordo com o princípio de Huygens.

Para calcular o tempo de viagem das ondas parte-se do conhecimento da altura significativa inicial e de aproximações para o cálculo de seu período e comprimento. Com

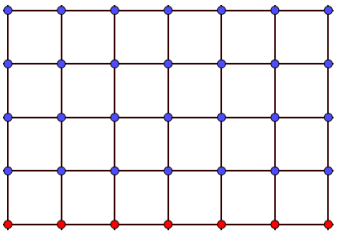


Fig. 3: Inicialização dos pontos fontes com direção de propagação de baixo para cima

isso, é possível obter a celeridade de ondas e atualizar os tempos de viagem em função deste parâmetro. Essa atualização dos tempos é feita de maneira local, considerando o ponto fonte e uma vizinhança de ordem $\sqrt{5}$, ou seja, tomando-se 8 direções e 20 vizinhos. Se o tempo do nó fonte mais o tempo de viagem entre os nós é menor do que o tempo já atribuído a um nó vizinho, então o nó vizinho recebe um tempo menor. As atualizações terminam quando não há mais modificações nos tempos. A altura de onda é então calculada ao longo das frentes de propagação.

Considerando o ajuste na altura da onda dada pela sua quebra, podemos então calcular a velocidade orbital máxima no fundo. Normalizando U e calculando ΔH, atualizamos o valor de altura e enfim obtemos a energia final do modelo de ondas.

4 SIMULAÇÃO

A área de estudo para aplicação do modelo de energia é a Grande Barreira de Coral Australiana, composta por cerca de 2900 recifes, 600 ilhas continentais e 300 atóis de coral, situada entre as praias do nordeste da Austrália e Papua-Nova Guiné. Para proceder com a simulação utilizamos um Modelo Digital de Elevação (DEM) disponibilizado pelo projeto 3DGBR [2]. O grid utilizado possui formato retangular com 272×282 células, onde cada par de células vizinhas estão a uma distância real de 2 km, compreendendo uma área total de 306816 km^2. A profundidade máxima na área é de 2865 m, com morfologia de rampa entre 150 m e 2865 m de profundidade, e plataforma com extensão até a costa. Para fins de implementação usamos o software MATLAB (2019b) e plugin TopoToolbox [13] em um computador com processador Intel(R) Core(TM) i9-9900K CPU @ 3.60 GHz.

A representação da simulação é apresentada na Figura 4, com altura significativa de onda inicial de $H_0 = 2.5$ m e direção de propagação de ondas da direita para esquerda, conforme orientação da Figura 4. Os valores na barra lateral representam densidade de energia por unidade de área da superfície do mar E com unidade em (J/m^2). Note que na borda da área de simulação temos a energia máxima do modelo devido ao valor de H_0 informado; e conforme as ondas propagam-se em direção ao continente dissipam energia em profundidades menores, como é possível ver nos dois montes submarinos e quando as ondas alcançam a barreira de corais a qual situa-se entre 1 m e 50 m de profundidade. Quando as ondas passam pela barreira de corais já transportam uma energia menos acentuada, que é dissipada novamente pelos fatores de quebra de ondas, e atenuação devido a velocidade máxima orbital de fundo.

Para analisar a variação de altura de ondas, foi feito um corte vertical na posição $j = 129$ do grid, apresentado na Figura 5. Note que a altura diminui devagar até encontrar a barreira, tendo uma diminuição expressiva na posição $i = 150$. Logo após, recupera-se

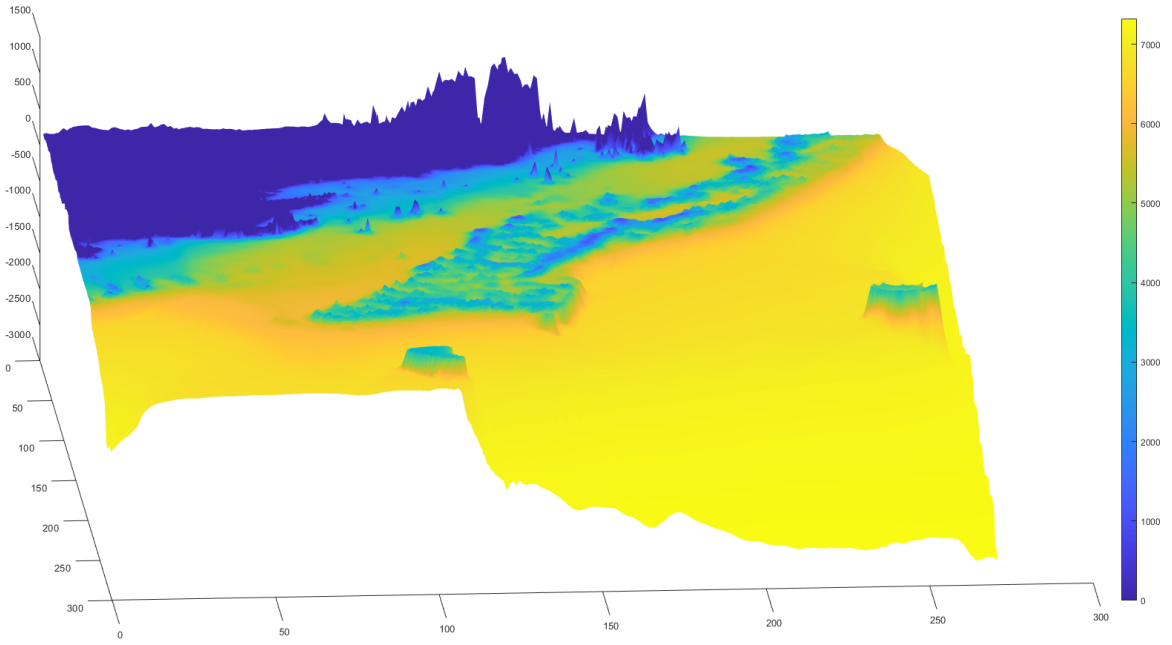


Fig. 4: Modelo de energia de ondas simulado na Grande Barreira de Corais Australiana. A região em azul forte representa a costa Australiana (área de terra); as demais cores representam a intensidade de energia de ondas simulada. Cores em escala de amarelo representam as energias mais altas, e as cores em escala de verde e azul claro representam as energias intermediárias e baixas, respectivamente.

parte da energia devido ao tempo que a onda viaja, e novamente, próximo à costa ocorre um decréscimo acentuado após os 40 m de profundidade, devido aos fatores de dissipação.

A partir dessa simulação, verificamos que a energia diminui ao se aproximar da zona costeira, ou quando a onda passa por rampas ou picos submarinos. Destacamos também a eficiência do modelo computacional que obtém o mapa de energia em aproximadamente 5 segundos para as dimensões de grid descrito anteriormente. Ressaltamos que é interessante fazer simulações para maiores dimensões e estabelecer uma correlação entre os valores de altura de ondas simulados com alturas medidas, se disponíveis.

5 CONCLUSÃO

Apresentamos um modelo de ondas baseado em teoria linear para estimar a energia superficial de ondas. Consideramos fatores físicos como quebra de ondas e velocidade orbital máxima de fundo para atenuação de energia. A partir da implementação de um método numérico, usamos a Grande Barreira de Corais da Austrália como área de simulação. Os resultados obtidos foram apresentados graficamente em uma superfície gerada em MATLAB com os níveis de energia apresentados em escala de cores. Com base em um corte vertical na superfície em estudo, verificamos a variação de altura de ondas em relação a zona de propagação.

Para trabalhos futuros esperamos que o presente estudo possa nos guiar a obter modelos genéricos para caracterização de ambientes costeiros em função de parâmetros oceanográficos. A teoria de ondas de segunda e terceira ordem é uma alternativa de modelagem

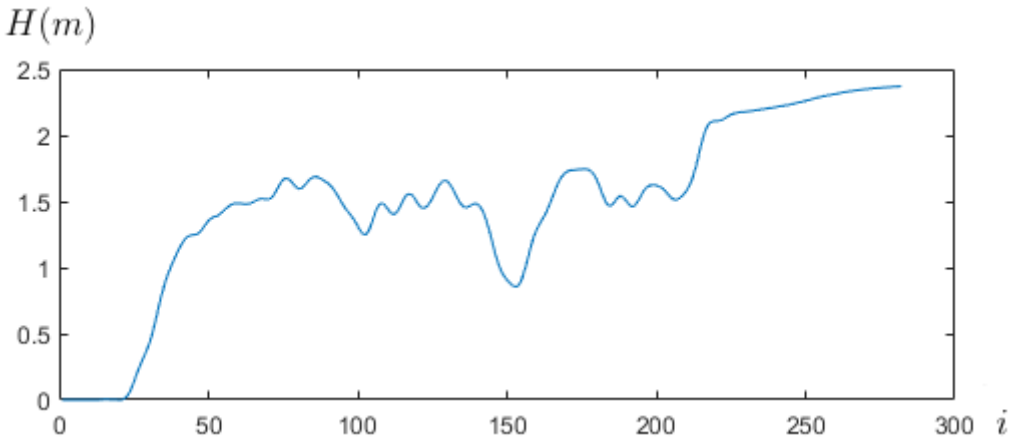


Fig. 5: Corte vertical do grid para exibir a variação da altura de ondas.

a ser considerada, pois descreve mais precisamente as assimetrias das velocidades orbitais, propiciando um modelo de energia mais conciso. Neste sentido, o uso de técnicas de modelagem numérica será essencial para obter soluções que demandem um custo computacional reduzido do que se conhece para estes modelos atualmente.

6 *Agradecimentos*

Esta pesquisa foi realizada dentro do termos do Acordo de Cooperação e Assistência Técnica assinada pela Universidade do Vale do Rio dos Sinos (UNISINOS) e pelo Petróleo Brasileiro S.A (PETROBRAS).

Referências

[1] G. B. Airy. *Tides and waves*. B. Fellowes, 1845.

[2] R. Beaman. 3dgbr: A high-resolution depth model for the great barrier reef and coral sea. *Marine and Tropical Sciences Facility (MTSRF) Project*, 2:13, 2010.

[3] D. Carter. Prediction of wave height and period for a constant wind velocity using the jonswap results. *Ocean Engineering*, 9(1):17–33, 1982.

[4] F. J. Dijksterhuis. *Lenses and waves: Christiaan Huygens and the mathematical science of optics in the seventeenth century*, volume 9. Springer Science & Business Media, 2004.

[5] L. Farina. Ondas oceânicas de superficie. *Notas em Matemática Aplicada, SBMAC*, 25, 2006.

[6] G. Griggs and A. Trenhaile. *Coastal cliffs and platforms*. Cambridge University Press, Cambridge, UK, 1994.

[7] J. Harari and R. de Camargo. Numerical simulation of the tidal propagation in the coastal region of santos (brazil, 24 s 46 w). *Continental Shelf Research*, 23(16):1597–1613, 2003.

[8] K. Hasselmann, T. P. Barnett, E. Bouws, H. Carlson, D. E. Cartwright, K. Enke, J. Ewing, H. Gienapp, D. Hasselmann, P. Kruseman, et al. Measurements of wind-wave growth and swell decay during the joint north sea wave project (jonswap). *Ergänzungsheft 8-12*, 1973.

[9] J. Hill, D. Tetzlaff, A. Curtis, and R. Wood. Modeling shallow marine carbonate depositional systems. *Computers & Geosciences*, 35(9):1862–1874, 2009.

[10] M. d. F. A. d. Matos, C. J. E. Fortes, V. E. Amaro, and A. C. Scudelari. Análise comparativa da agitação obtida com o modelo numérico (swan) na modelagem de ondas do litoral setentrional do rio grande do norte, brasil e dados de campo. *Revista de Gestão Costeira Integrada*, 13(3):283–299, 2013.

[11] C. B. Rachman et al. *Two Dimensional (2D) Experimental of Piling Up Behind Submerged Breakwater*. PhD thesis, [Yogyakarta]: Universitas Gadjah Mada, 2012.

[12] O. Sato. Ondas e marés. *Universidade de São Paulo*, 2010.

[13] W. Schwanghart and D. Scherler. Topotoolbox 2–matlab-based software for topographic analysis and modeling in earth surface sciences. *Earth Surface Dynamics*, 2(1):1–7, 2014.

[14] D. M. Tetzlaff and M.-T. Schafmeister. Interaction among sedimentation, compaction, and groundwater flow in coastal settings. *Coastline Changes: Interrelation of Climate and Geological Processes*, 426:65, 2007.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Electromagnetic loudspeaker: an energetic approach

Natasha Hirschfeldt[1], Roberta Lima[1] e Rubens Sampaio[1]

[1] *PUC-Rio, Rio de Janeiro/RJ, Brazil*

Abstract

Electromechanical systems are composed by two subsystems with distinct origins: one of a mechanical nature and another of electromagnetic nature. The energies in the system have also different origins. Some of them are mechanical, as kinetic and potential, and others are electromagnetic, as magnetic and electrical. For a proper description of the dynamics of an electromechanical system, it is not sufficient to describe each subsystem separately. Coupling terms must be considered in the system dynamics. These terms characterize the mutual influence between the two subsystems and the interplay of the energies of the system. The objective of this paper is to analyze from an energetic point of view an electromechanical system. This paper shows how the dynamics of an electromechanical system can be constructed by the definition of the energies that are present in the system and their interplay using the Lagrangian method. To exemplify, an electromagnetic loudspeaker will be analyzed. Its dynamics will be constructed and numerical integrated in order to make an energetic analysis.

Keywords: Lagrangian; Energy; Co-energy; Electromechanical; Transducer.

1 INTRODUCTION

Electromechanical systems are composed by two coupled subsystems, a mechanical and an electromagnetic. They can be found in several applications used in daily life. However, even though they are so common, it is still a challenge to find references correctly describing their dynamics. Several published papers, books and thesis present serious mistakes in the description of the dynamics of electromechanical systems. A common error found in the literature is to neglect the dynamics of the electromagnetic subsystem and its interactions with the mechanical subsystem (see reference *Cveticanin, L., Zukovic, M., Balthazar, J. M.* from [6]). Without the the dynamics of the electromagnetic subsystem,

Contact: Natasha Hirschfeldt, natashaboh97@gmail.com

the electromechanical system becomes a purely mechanical system described by mechanical variables. The recent published paper [4] discusses about some of the references with mistakes and shows how to neglect the electromagnetic subsystem changes the dynamics.

The objective of this paper is to make a step by step of how to describe properly the dynamics of an electromechanical system using the Lagrangian method, also seen in [3, 8]. To accomplish this goal, the dynamic equations of an electromagnetic loudspeaker are going to be deducted and analyzed.

The electromagnetic loudspeaker analyzed in this paper is presented in Sec. 2. The variables, parameters and conditions required to use this system are also given in Sec. 2. The coupling term that produces the interaction between the mechanical subsystem and the electromagnetic subsystem in this loudspeaker is a transducer. This element is presented in Sec. 2.1. In Sec. 2.2, the energies that are presented in the system are defined and used in the construction of the system dynamics by the Lagrangian method.

After all the calculations, an energy analysis will be made in Sec. 3 to compare the different types of energy (kinetic, potential, electric and magnetic) and show their interplay by the results of numerical integrations of the system dynamics.

2 Dynamics of the electromagnetic loudspeaker

To exemplify the interaction of the two subsystems that compose an electromechanical system, an electromagnetic loudspeaker will be simulated and discussed. This loudspeaker is illustrated in Fig. 1.

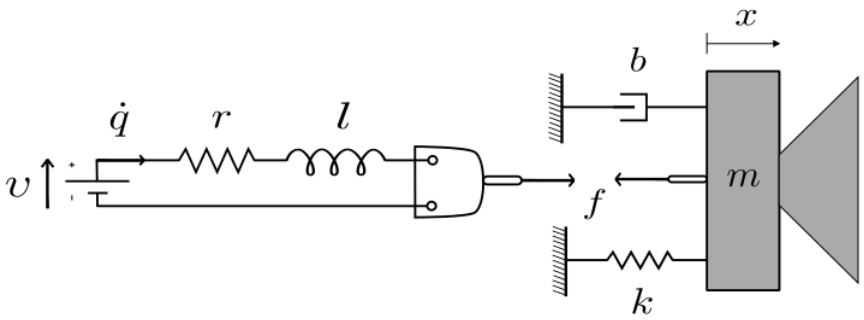


Fig. 1: Electromagnetic loudspeaker. [7]

The system is composed by a mechanical subsystem (a mass m, a spring of constant k and a damper of constant b, where the last two are simulating a membrane that dislocates the air), an electromagnetic subsystem (a voltage source υ in series with an RL circuit, which means an inductor of inductance l and a resistor of resistance r) and an element called moving-coil transducer (with transducer constant ϱ, explored topic in Sec. 2.1) that couples the two subsystems. Two variables are used as the parameterization the system dynamics. One of them is mechanical, it is called x and represents the displacement of the mass m from the mechanical subsystem's equilibrium point, and the other is electromagnetic, the charge q passing through the circuit. It is important to stand out the fact that the displacement x does not represent the sound waves produced by the loudspeaker, it is merely the displacement of the mass m from the chosen equilibrium point. To produce the sound waves, the dynamics of speakers should be considered.

2.1 Moving-coil transducer

A moving-coil transducer is an energy transformer element of a system that converts electrical power into mechanical power or vice versa and can not store energy. In the loudspeaker case, the current $\dot{q}$ originated by the potential difference e passing through the ends of the circuit is being converted into a displacement x. The transducer's elements and its symbolic representation are illustrated in Fig. 2.

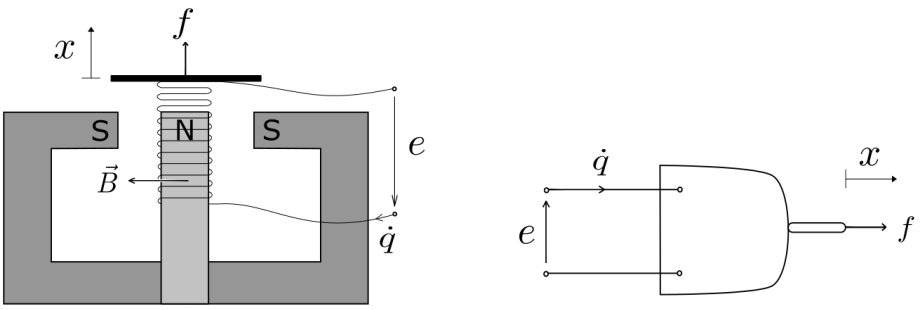


Fig. 2: A moving-coil transducer and its symbolic representation, respectively. [7]

A coil is passing around one of the poles from a magnet that generates a magnetic flux density B. Being f the force to keep the coil's equilibrium (it means, opposite to the electromagnetic forces) and knowing that the coil is free to move in the direction of f, it is possible to obtain the magnetic co-energy [1]:

$$U_m^*(x, \dot{q}) = \varrho \dot{q} x, \tag{1}$$

where ϱ is called the transducer constant and is given by:

$$\varrho = 2\pi n r B. \tag{2}$$

Here, n is the number of turns of the coil, r is the radius of the coil and, therefore, $2\pi nr$ is the coil's length that is immersed in the magnetic flux B.

2.2 Lagrangian formulation for an electromagnetic loudspeaker

The Lagrangian for an electromechanical system [7] is written as:

$$\Gamma = T^* - V + E_m^* - E_e \pm U^*, \tag{3}$$

where T^* is the kinetic co-energy, V the potential energy, E_m^* the magnetic co-energy and E_e the electric energy. More about co-energy can be seen in [1, 6, 7, 9]. The coupling term U^* can have an electric or magnetic origin and it's signal depends on this fact. If it is transmitted as a magnetic coupling (U_m^*), the signal is positive and if it is transmitted as an electric one (U_e^*), the signal is negative. It is shown in the next equations:

$$\Gamma = T^* - V + (E_m^* + U_m^*) - E_e, \tag{4}$$

$$\Gamma = T^* - V + E_m^* - (E_e + U_e^*). \tag{5}$$

Being z_i a generalized coordinate of the system, each differential equation of the system dynamics can be found by:

$$\frac{d}{dt}\left(\frac{\partial \Gamma}{\partial \dot{z}_i}\right) - \frac{\partial \Gamma}{\partial z_i} = \frac{d\delta W}{d\delta z_i}. \tag{6}$$

where δW is the virtual work given through the non-conservative elements and δz_i is a generic coordinate of the system.

The virtual work for an electromechanical system can be found by using the following relation:

$$\delta W = \delta_f - \delta_d. \tag{7}$$

where δ_f refers to elements that supply or absorbs energy for the system (such as a source υ for the electromagnetic subsystem and an external force f for the mechanical subsystem) and δ_d to elements that dissipate energy (such as a resistance r for the electromagnetic subsystem and a damper b for the mechanical subsystem). They can be written as:

$$\delta_f = \upsilon_i \; \delta q_i + f \delta x_i, \quad \delta_d = r_i \; \dot{q}_i \; \delta q_i + b \; \dot{x}_i \; \delta x_i. \tag{8}$$

To determinate the signal of the terms of Eq. (8) that refers to the elements supplying energy, we will attribute that it is positive when the element supplies energy to the system and negative when the element absorbs energy from it.

In the case of an electromagnetic loudspeaker, the coupling term in the Lagrangian formulation is given by a moving-coil transducer (explained in Sec. 2.1), an element that contributes with an energy of magnetic origin. Another example of how the interaction between the two subsystems appears is given in [2, 6, 5], where the coupling term is now given by a DC motor, also an element that contributes with an energy of magnetic origin.

Next, the equations that describe the system dynamics for the loudspeaker will be constructed using the Lagrangian formulation [7, 9].

For the mechanical subsystem:

$$T^* = \frac{m\dot{x}^2}{2}, \quad V = \frac{kx^2}{2}. \tag{9}$$

For the electromagnetic subsystem:

$$E_m^* = \frac{l\dot{q}^2}{2}, \qquad E_e = 0. \tag{10}$$

The coupling term is given by the energy transmitted by the transducer with Eq. (1):

$$U_m^* = \varrho \dot{q} x.$$

So, using Eq. (3), the Lagrangian function is given by:

$$\Gamma = \frac{m\dot{x}^2}{2} - \frac{kx^2}{2} + \frac{l\dot{q}^2}{2} + \varrho \dot{q} x. \tag{11}$$

Obtaining the virtual work for the non-conservative elements of the system:

$$\delta_f = \upsilon \; \delta q, \quad \delta_d = r \; \dot{q} \; \delta q + b \; \dot{x} \; \delta x. \tag{12}$$

Therefore, the virtual work is:

$$\delta W = \delta_f - \delta_d \;=\; \upsilon\,\delta q - r\,\dot{q}\,\delta q - b\,\dot{x}\,\delta x. \tag{13}$$

For the generalized coordinate x:

$$\frac{\partial \Gamma}{\partial \dot{x}} = m\dot{x} \;\rightarrow\; \frac{d}{dt}\left(\frac{\partial \Gamma}{\partial \dot{x}}\right) = m\ddot{x}, \quad \frac{\partial \Gamma}{\partial x} = -kx + \varrho\dot{q}, \quad \frac{d\delta W}{d\delta x} = -b\dot{x}. \tag{14}$$

For the generalized coordinate q:

$$\frac{\partial \Gamma}{\partial \dot{q}} = l\dot{q} + \varrho x \;\rightarrow\; \frac{d}{dt}\left(\frac{\partial \Gamma}{\partial \dot{q}}\right) = l\ddot{q} + \varrho\dot{x}, \quad \frac{\partial \Gamma}{\partial q} = 0, \quad \frac{d\delta W}{d\delta q} = \upsilon - r\dot{q}. \tag{15}$$

Substituting Eq. (14) and (15) into Eq. (6) for each coordinates used in the system, the dynamic equations for the electromagnetic loudspeaker can be found. It is given by the following initial value problem. Given a source voltage $\upsilon(t)$, find (x, q) such that, for all $t > 0$ with initial conditions $x(0) = x_0,\ q(0) = q_0,\ \dot{x}(0) = v_0$ and $\dot{q}(0) = i_0$:

$$\begin{cases} m\ddot{x}(t) + b\dot{x}(t) + kx(t) - \varrho\dot{q}(t) = 0, \\ \\ \quad l\ddot{q}(t) + r\dot{q}(t) + \varrho\dot{x}(t) = \upsilon(t). \end{cases} \tag{16}$$

3 Energy analysis

To analyze the interplay between the different types of energy in this system, a routine was developed using the software *MATLAB* to simulate how the electromagnetic loudspeaker responds during 15 seconds to a situation where the initial conditions are $x(0) = 1,\ q(0) = 0,\ \dot{x}(0) = 0$ and $\dot{q}(0) = 0$. To simulate, the initial value problem that gives the system dynamics was integrated by the $4^{th} - 5^{th}$ order Runge-Kutta method with the *ode45* *MATLAB* function. The time-step used was 0.002 seconds and the parameters were chosen for a better interpretation of the results and they are given by: $m = 0.15\ kg,\ b = 0\ Ns/m,\ k = 0.10\ N/m,\ \varrho = 0.30\ mT,\ l = 1.00\ H, r = 0\ \Omega$ and $\upsilon = 0\ V$.

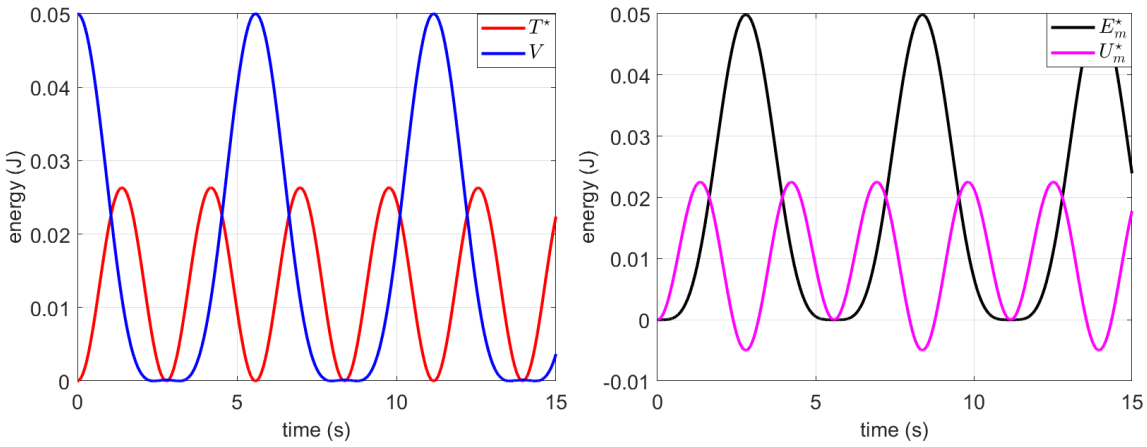


Fig. 3: Graphics showing the different types of energies in the system.

Figure 3 compares the kinetic co-energy T^* with the potential energy V and the magnetic co-energy E_m^* with the electric energy E_e. It is possible to notice that the potential energy reaches its maximum and minimum values when the kinetic energy is at its minimum. Something similar, but not equal, occurs this the magnetic co-energy and electric energy: when E_m^* reaches its maximum, E_e is at its minimum (a negative value), but when the magnetic co-energy reaches its minimum, the electric energy is null, reaching a local minimum. It is also possible to do an energy balance of the system. Using Eq. (16) and multiplying the first equation by $\dot{x}(t)$ and the second one by $\dot{q}(t)$:

$$\begin{cases} m\ddot{x}(t)\dot{x}(t) + b\dot{x}(t)\dot{x}(t) + kx(t)\dot{x}(t) - \varrho\dot{q}(t)\dot{x}(t) = 0, \\ \\ l\ddot{q}(t)\dot{q}(t) + r\dot{q}(t)\dot{q}(t) + \varrho\dot{x}(t)\dot{q}(t) = \upsilon\dot{q}(t). \end{cases} \tag{17}$$

Adding the two equations found in Eq. (17) and making $b = 0,\ r = 0,\ \upsilon = 0$:

$$\frac{d}{dt}\left(\frac{m\dot{x}(t)^2}{2} + \frac{kx(t)^2}{2} + \frac{l\dot{q}(t)^2}{2}\right) = 0. \tag{18}$$

Here, the parameters b, r and υ were chosen as null to make the system became conservative. The objective was to simplify the analysis and highlight the interplay of energies between the electromagnetic and mechanical subsystems.

It is possible to see in Eq. (18) that the coupling term of the system dynamics no longer appears. This happens because the moving-coil transducer is an element that does not store energy and, therefore, does not contribute to the energy balance. Figure 4 shows the graphic representation of Eq. (18), with the sum of the different types of energy: mechanical $(T^* + V)$ and electromagnetic (E_m^*). The green line represents this sum and is a constant with a value that depends, in this case, on the parameter k of the system.

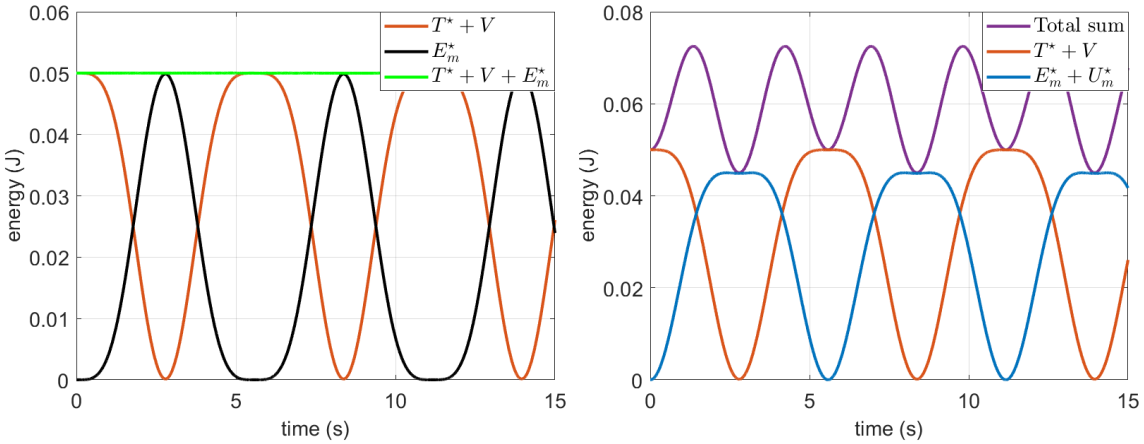


Fig. 4: Energy balance and different energies sums, respectively.

Figure 4 also shows two types of sums: one adding the two energies of mechanical origin and another one adding the two energies of electromagnetic origin. It is also shown the total sum of the energies, that is, the sum of all energies regardless of its origin. This last one does not equal a constant, showing once more that U_m^*, energy passing through

the transducer, is not stored in this element, it is only transmitted from one subsystem to another.

After this simple example, it is possible to change one of the parameters so a more accurate analysis can be made. Giving a $v = sin(t)$, the same graphics can be analyzed. The patterns in Fig. 5 and 6 are repeated every 50 seconds. Figure 5 shows a different pattern compared to the previous one: now, the potential energy V reaches its maximums when the kinetic energy T^* is at its minimums and vice versa. The relation between E_m^* and U_m^* also changes: their minimums are always close and the same occurs to the maximums. The fact that U_m^* is, in the most part, negative, shows that this energy is flowing contrary to the one before most of the time.

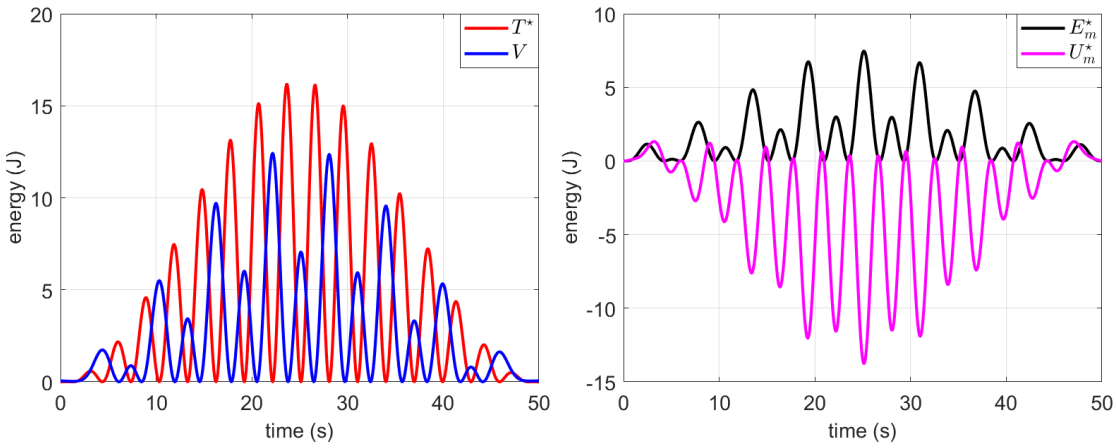


Fig. 5: Graphics showing the different types of energies in the system.

The different kinds of energy sums are given in Fig. 6.

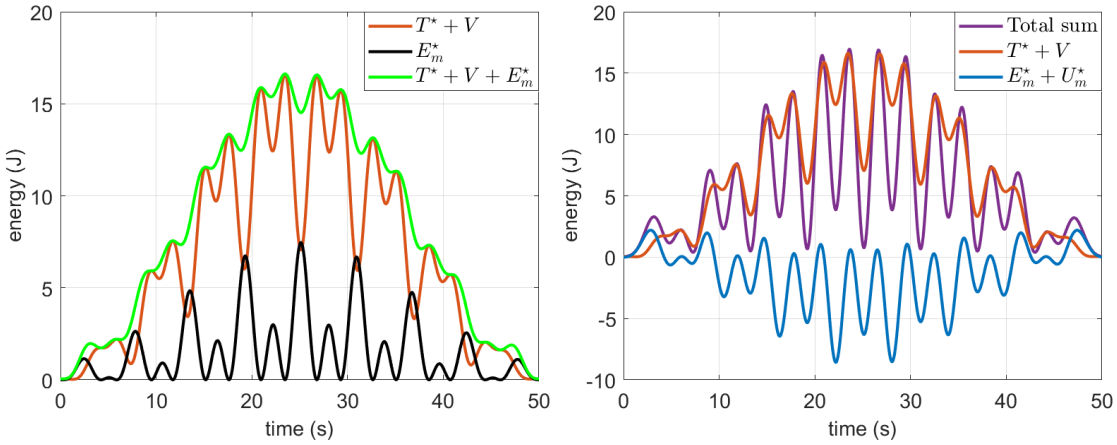


Fig. 6: Energy balance and different energies sums, respectively.

4 CONCLUSIONS

For a right description of an electromechanical system dynamics, it is important to have in mind the parameterization and the coupling element. This paper showed the correct way

of using the Lagrangian method to do that wile using an example of an electromagnetic loudspeaker. Please remark that it is not sufficient to describe each subsystem (mechanical and electromagnetic) separately, there must be a coupling term between them. This term can have a magnetic origin (as shown with the transducer in the loudspeaker and in [6], [5] and [7]) or an electric origin [7]. It is also explored the fact that a coupling element does not have to store energy, it can only transform it and, therefore, the energy flux in this case is a little bit different from a pure mechanical system.

5 *Acknowledgements*

The authors acknowledge the support given by FAPERJ, CNPq and CAPES.

References

[1] D. JELTSEMA and J. M. A. SCHERPEN. Multidomain modeling of nonlinear networks and systems. *IEEE Control Systems*, 29:28–59, 2009.

[2] R. LIMA and R. SAMPAIO. Two parametric excited nonlinear systems due to electromechanical coupling. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 38:931–943, 2016.

[3] R. LIMA and R. SAMPAIO. Pitfalls in the dynamics of coupled electromechanical systems. In *CNMAC 2018, Proceeding Series of the Brazilian Society of Computational and Applied Mathematics*, Campinas, SP, Brazil, 2019.

[4] R. LIMA, R. SAMPAIO, P. HAGEDORN, and J.-F. DEÜ. Comments on the paper "on nonlinear dynamics behavior of an electro-mechanical pendulum excited by a nonideal motor and a chaos control taking into account parametric errors" published in this journal. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 41:552, 2019.

[5] W. MANHÃES, R. SAMPAIO, R. LIMA, and P. HAGEDORN. Two coupling mechanisms compared by their lagrangians. 2019.

[6] W. MANHÃES, R. LIMA, R. SAMPAIO, P. HAGEDORN, and J.-F. DEÜ. Lagrangians for electromechanical systems. *Mecánica Computacional*, XXXVI:1911–1934, 2018.

[7] A. PREUMONT. *Mechatronics: Dynamics of Electromechanical and Piezoelectric Systems*, volume 136. Springer, Brussels, Belgium, 2006.

[8] R. SAMPAIO, R. LIMA, and P. HAGEDORN. One alone makes no coupling. *Mecánica Computacional*, XXXVI:931–944, 2018.

[9] D. A. WELLS. *Schaum's outline of theory and problems of Lagrangian Dynamics with a treatment of Euler's Equations of Motion, Hamilton's Equations and Hamilton's Principle*. McGraw-Hill, Inc, USA, 1967.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Detecção de Possíveis Irregularidades nas Inspeções de Equipamentos que Transportam Produtos Perigosos usando Aprendizagem Profunda

Pablo Holzmeister Ortiz[1], Rosembergue Pereira De Souza[1] e Luiz Fernando Rust Da Costa Carmo[1]

[1] *Instituto Nacional de Metrologia, Qualidade e Tecnologia, Duque de Caxias/RJ, Brasil*

Resumo

No Brasil, é do Inmetro o papel de acreditar e supervisionar serviços acreditados. Dentre esses serviços, está a inspeção de equipamentos para transporte de produtos perigosos, cujos registros visuais da inspeção devem ser disponibilizados ao Inmetro. Entretanto, uma dificuldade enfrentada pela equipe avaliadora é o grande volume de informações que deve ser processado. Denúncias indicam que existem fraudes no setor, incluindo clonagens e adulterações das evidências fotográficas das inspeções. Este trabalho tem o objetivo de propor um método para detecção de fotos clonadas e adulteradas. As adulterações foram identificadas pela distância semântica entre imagens através das camadas intermediárias do modelo de classificação RESNET50, pois imagens similares têm camadas intermediárias semelhantes. O trabalho também determina significado para os intervalos de distância entre as imagens. Trata-se de pesquisa científica com aplicação prática e resposta a um problema real. O método é validado a partir da aplicação e detecção em amostra contendo imagens clonadas e adulteradas.

Palavras-chave: Classificação de imagens, Comparação de imagens, Inspeção, Transporte de produtos perigosos, Acreditação.

1 INTRODUÇÃO

Segundo a resolução ANTT Nº 5.848/19, equipamentos utilizados para transporte de produtos perigosos devem ser inspecionados periodicamente por organismos acreditados pelo Inmetro [2]. Acreditação é o reconhecimento formal, concedido por um organismo autorizado, de que a entidade foi avaliada, segundo guias e normas nacionais e internacionais e

Contato: Pablo Ortiz, phortiz@inmetro.gov.br

tem competência técnica e gerencial para realizar tarefas específicas. Uma vez que a organização solicitante é acreditada, anualmente o Inmetro realiza avaliações de supervisão para verificar se essa empresa continua mantendo as condições técnicas e regulamentares para prestar o serviço acreditado. Nessas supervisões, os avaliadores verificam as condições das instalações e equipamentos do organismo, bem como a competência técnica do pessoal e capacidade de gestão da empresa. Além disso, a equipe do Inmetro verifica os registros dos serviços de inspeção prestados pelo organismo acreditado. Esta análise investiga se o organismo prestou serviços conforme os regulamentos técnicos pertinentes. Dos registros analisados pela equipe avaliadora, destacam-se os registros fotográficos dos objetos inspecionados, filmagens da execução do serviço, listas com constatações da inspeção e certificados de inspeção emitidos.

O Inmetro procura estabelecer ações de melhoria no acompanhamento dos serviços acreditados pelo órgão, pois já recebeu denúncias e existe preocupação com fraudes cometidas pelos organismos. Uma dessas ações é o uso intensivo de ferramentas de Tecnologia da Informação (TI), destacando-se a criação do banco de imagens do Inmetro contendo registros fotográficos das inspeções. A NIT-DIOIS-019 [6] define critérios específicos para a acreditação de organismos de inspeção e dispõe que os organismos devem enviar registros fotográficos das inspeções de equipamentos de transporte de produtos perigosos para o Inmetro. De posse de subconjunto de dados suspeitos, uma das formas de verificar se o organismo de inspeção está realizando suas tarefas corretamente é através da análise das fotos das inspeções referentes a esse subconjunto de dados. Confirmando-se a existência de irregularidades nessas imagens, pode-se retroalimentar o sistema na procura de irregularidades similares em outros serviços prestados pelo organismo de inspeção acreditado. Entretanto, uma dificuldade enfrentada pela equipe avaliadora é o grande volume de informações que deve ser processado.

Algumas das fraudes reportadas referem-se à clonagem e adulteração de fotografias e as contribuições deste trabalho são: (i) um método para detectar imagens clonadas e (ii) um método para detectar imagens adulteradas. O método para detectar imagens clonadas é baseado no HASH de cada imagem e o método para detectar imagens adulteradas é baseado na distância semântica entre imagens, medida através da aquitetura de aprendizagem profunda para classificação de imagens RESNET50 [5]. É objetivo secundário a definição de intervalos de distância que possibilitem indicar: (i) fotografias clonadas com leves alterações e (ii) fotografias similares semanticamente a uma imagem de referência.

O diferencial do método proposto para detectar imagens adulteradas está na comparação de fotos pela extração de informação diretamente das camadas intermediárias de aprendizado, no momento de avaliação de uma nova imagem, através da distância semântica entre as imagens. Imagens semelhantes teriam camadas intermediárias de aprendizado similares. Comprovando-se a hipótese, seria possível detectar casos similares à imagem de referência contendo irregularidade perceptíveis visualmente.

Este trabalho está organizado da seguinte maneira: a seção 2 descreve estudos relacionados à comparação e classificação de imagens, a seção 3 apresenta a metodologia utilizada, a seção 4 retrata resultados e discussão e a seção 5 mostra as conclusões.

2 TRABALHOS RELACIONADOS

A comparação de imagens correspondentes à mesma cena não é uma tarefa simples, pois pode sofrer influência de muitos fatores, como: ponto de vista, variação de iluminação, sombras e diferença de configuração da câmera. Na literatura, Zagoruyko e Komodakis apresentaram um estudo sobre a comparação de imagens através de seus fragmentos, utilizando rede neural treinada para criar uma função de similaridade [14]. Snow e Lent, baseados na premissa que em imagens semelhantes a distribuição dos pixels seria parecida, propuseram medida de similaridade através da distância Wasserstein, que tem o objetivo de encontrar a maneira ideal de transportar uma imagem para outra [9]. Já Liu *et al.* propuseram método para identificar veículos registrados com mais de uma placa baseado em marcas similares, através da distância euclidiana entre espaços das características normalizadas das imagens, gerados por rede de aprendizagem profunda [8].
O estado da arte de algoritmos de classificação de imagem utiliza redes neurais e aprendizagem profunda. Segundo Lecun, Bengio e Hinton (2015), aprendizagem profunda corresponde a modelos computacionais compostos por múltiplas camadas de processamento para aprender representações de dados com múltiplas camadas de abstração. A classificação de imagens é uma tarefa que possui a propriedade de a semântica ser disposta hierarquicamente, sendo as características de camadas mais altas compostas pelas de inferiores e isso pode ser compreendido pelas máquinas através de aprendizagem profunda [7]. Tipicamente, redes de aprendizagem profunda para classificação de imagens são compostas por múltiplas camadas intermediárias, responsáveis pelo aprendizado hierárquico, e são finalizadas com uma ou mais camadas responsáveis pela classificação [3]. Alguns exemplos de arquitetura são AlexNet, GoogleNet e RESNET, que venceram o desafio de classificação de imagens ILSVRC (*ImageNet Large Scale Visual Recognition Challenge*) em 2012, 2014 e 2015, respectivamente [3]. Este trabalho utiliza a rede RESNET, proposta inicialmente por He, Zhang, Ren e Sun [5], e que foi escolhida por ser uma arquitetura tradicional e porque para o problema em questão uma abordagem genérica seria suficiente, pois os objetos não possuem particularidades que justifiquem a adoção de arquiteturas específicas. A rede é composta por múltiplas camadas, sendo que as camadas intermediárias têm o propósito do aprendizado propriamente dito das características do objeto e as 2 últimas camadas efetivamente classificam a imagem.
O treinamento da rede fez parte da dissertação de mestrado de Ortiz (2020) [10], foi supervisionado [4] e foi utilizada amostra de 51.396 imagens selecionada a partir do banco de imagens do Inmetro. Treinamentos de redes profundas requerem quantidade enorme de dados, mas a transferência de aprendizagem é uma forma de diminuir a dependência dos dados, ou seja, diminuir a quantidade necessária de imagens. Essa técnica já foi utilizada com sucesso em várias aplicações [13], reduz o tempo de treinamento e acelera a convergência, por isso raramente redes neurais convolucionais são treinadas do zero, sem começar a partir de outra rede já treinada [1]. Neste treinamento foi utilizada a transferência de aprendizagem baseada em rede, que preserva as camadas de aprendizagem e os parâmetros da rede pré-treinada [13], uma RESNET50 treinada no banco de imagens ImageNet para distinguir 1.000 classes, e que a camada final de classificação foi modificada para a quantidade de classes do problema.

3 METODOLOGIA

Como dito anteriormente, a comparação de imagens neste trabalho foi realizada através da medição da distância semântica entre camadas intermediárias do modelo de classificação RESNET50. A arquitetura desse modelo é composta por 50 camadas, sendo algumas agrupadas em estágios, como pode ser visto na Fig. 1. Os dados resultantes do quarto estágio foram utilizados para comparação semântica entre imagens através da distância euclidiana. Nesta etapa, a semântica da imagem foi disposta em uma matriz de 4 dimensões (1 x 2048 x 7 x 7), utilizada para avaliação da distância. Os passos posteriores ao quarto estágio são responsáveis apenas pela classificação, no caso em 6 classes, por isso não foram utilizados para avaliação da distância.

Fonte: Produção do Autor baseado em [12].

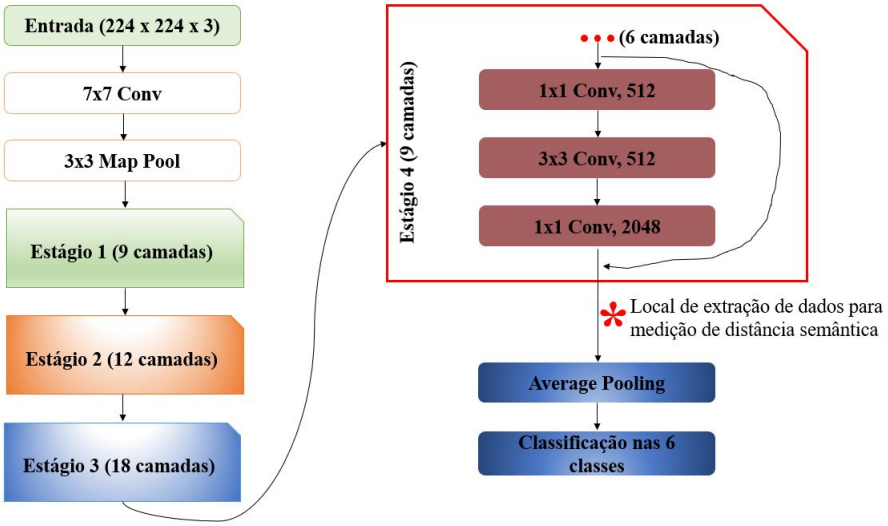


Fig. 1: Arquitetura da rede neural RESNET50 indicando camada utilizada para comparação semântica.

As etapas do experimento são: (i) treinamento da RESNET50, (ii) classificação das imagens do banco do Inmetro e identificação das que correspondem a equipamentos de transporte de produtos perigosos, (iii) cálculo do HASH de cada imagem, (iv) extração da matriz semântica da imagem a partir do resultado do quarto estágio do modelo RESNET50, (v) cálculo da distância semântica, por organismo de inspeção, entre imagens de equipamento inseridas no banco de imagens do Inmetro, (vi) categorização dos intervalos de distância para extrair indicação de significado para a medida, (vii) validação dos métodos de clonagem idêntica e de clonagem com adulterações, (viii) validação do intervalo para identificar fotos similares e (ix) detecção de clonagens e adulterações suspeitas no banco de imagens do Inmetro.

O modelo de redes neurais foi treinado para classificação 6 categorias, sendo que para identificação de clonagens e adulterações foram utilizadas somente as do equipamento, classificadas como “tanque” ou “não tanque”. Equipamentos “tanque” são utilizados para transportar líquidos ou gases, e os “não tanque” são utilizados para transporte de substâncias sólidas em caçamba, caminhão de carga aberto, caminhão de carga fechado

ou até veículos menores, como pick-ups.
O HASH de cada imagem foi utilizado para detectar fotos completamente idênticas. Uma boa função HASH gera resultados idênticos com a mesma entrada e diferentes para entradas distintas. O valor numérico correspondente ao HASH de cada imagem foi armazenado e valores iguais identificaram imagens idênticas [11]. Entretanto, a variação de um bit na imagem resulta em um HASH diferente. Para identificar fotos de equipamento adulteradas com leves alterações, foi calculada a distância semântica entre imagens, acessando camada intermediária do modelo de classificação.
Com o intuito de compreender o significado das distâncias medidas, foi prevista a avaliação gráfica de intervalos de distâncias para identificar os que correspondem à mesma cena ou a equipamentos similares. Foi coletada uma amostra de 100 comparações para cada intervalo. Era esperado que imagens correspondentes à mesma cena estivessem concentradas nas menores distâncias e que a frequência de fotos diminuísse com o aumento das distâncias. Analogamente, fotos completamente distintas estariam concentradas nas maiores distâncias. Também era esperado que o gráfico apresentasse intervalos intermediários que tivessem maior probabilidade de indicar o mesmo equipamento e equipamentos similares. Para auxiliar na definição dos intervalos foi construído um histograma com a frequência de comparações por intervalo.
Os métodos foram validados a partir de amostra contendo 100 inspeções e alterada para simular clonagens, sendo 5 clonagens idênticas e 10 com leves adulterações. A amostra foi avaliada conforme HASH de cada imagem para detectar clonagens idênticas e pelo algoritmo construído para detectar similaridades de fotos adulteradas.

4 RESULTADOS E DISCUSSÕES

4.1 CLASSIFICAÇÃO DAS IMAGENS

A partir do modelo RESNET50 treinado, foram avaliadas 989.388 imagens do banco do Inmetro de inspeções de produtos perigosos, conforme Tabela 1. O código do treinamento foi disponibilizado em https://github.com/phortiz/TreinamentoPP/tree/main/RESNET. Em 122.471 inspeções, cujas imagens foram disponibilizadas para o Inmetro entre agosto/2018 e agosto/2020, 125.094 foram classificadas como “tanque” ou “não tanque”, que foram objetos da pesquisa.

Tabela 1: Quantidade de imagens por classe

Rótulo/ Classe	Quantidade
Tanque	111.985
Não Tanque	13.109
CIPP Frente	122.799
CIPP Verso	123.396
Manômetro	215.976
Placas	402.123

4.2 MEDIÇÕES

Para cada imagem foi gerado um número identificador através de uma função HASH, sendo que números HASH iguais correspondem a imagens idênticas.
As imagens de tanque e não tanque de um mesmo organismo foram comparadas e calculadas as distâncias semânticas. O procedimento iniciou-se com nova avaliação das imagens, acessando a camada de saída do quarto estágio da rede neural. Foram realizadas 122.078.414 comparações, sendo que 730 apresentaram distância inferior a 100 e 12 superior a 310. O restante foi agrupado em intervalos de tamanho 10 e apresentado no histograma da Fig. 2. Foi utilizada escala logarítmica de base 10 para melhor visualização da distribuição das comparações. Mais de 93% concentraram-se no intervalo entre 260 e 290 e pela variedade de equipamentos seria razoável imaginar que a maior parte das comparações estivessem reunidas em intervalos que não apresentassem similaridade entre as fotos.

Fonte: Produção do Autor.

Intervalo	Quantidade de comparações	Percentual
100-110	26	0,00%
110-120	27	0,00%
120-130	43	0,00%
130-140	60	0,00%
140-150	68	0,00%
150-160	81	0,00%
160-170	118	0,00%
170-180	140	0,00%
180-190	140	0,00%
190-200	197	0,00%
200-210	411	0,00%
210-220	1.917	0,00%
220-230	11.413	0,01%
230-240	85.600	0,07%
240-250	743.192	0,61%
250-260	5.882.383	4,82%
260-270	28.580.624	23,41%
270-280	56.986.088	46,68%
280-290	28.354.952	23,23%
290-300	1.420.187	1,16%
300-310	10.005	0,01%
Total	122.077.672	100,00%

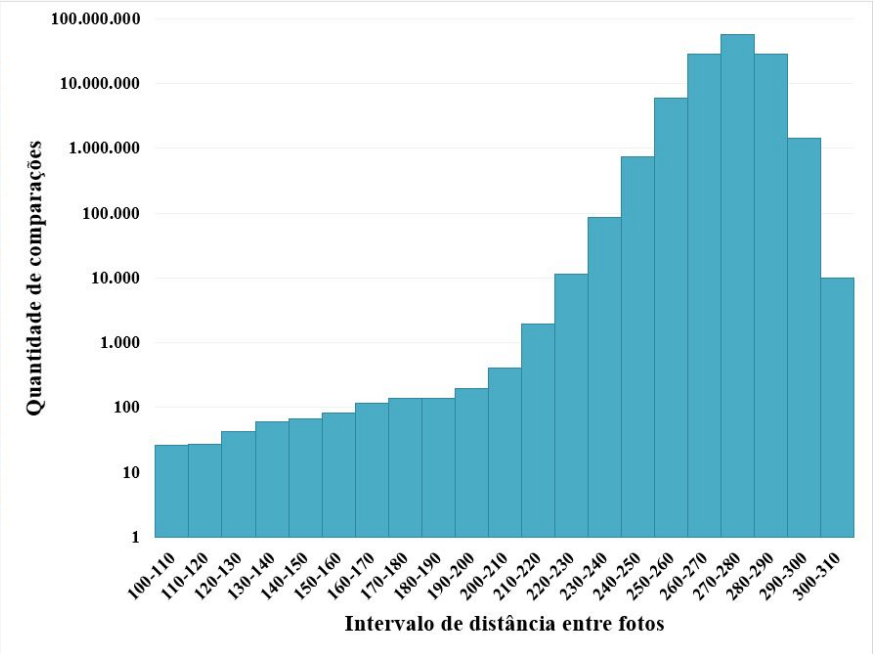


Fig. 2: Histograma de quantidade de comparações por intervalo de distância.

Para cada um dos intervalos foi selecionada uma amostra de tamanho 100 com o intuito de avaliar o significado da distância medida. Foram definidas 3 categorias, sendo: (i) sem diferença perceptível, (ii) fotos similares e (iii) fotos distintas. Imagens "sem nenhuma diferença perceptível" poderiam representar a mesma foto, que passou por algum tipo de tratamento, ou fotos tiradas em sequência, com pouca variação. As 2 interpretações sugerem que foram registradas para a mesma inspeção, não podendo constar como artefatos de inspeções distintas. Fotos foram consideradas similares por corresponderem ao mesmo equipamento, a equipamentos praticamente idênticos, mas com placas distintas, ou a equipamentos com formato ligeiramente diferente ou com cores diferentes.
As amostras foram classificadas nessas categorias, conforme Fig. 3, e como esperado as

menores distâncias corresponderam a imagens sem diferença perceptível e as maiores a fotos distintas. É possível perceber que assim que a curva de imagens sem diferença decresce, há aumento de equipamentos similares. Seria esperado que distâncias maiores indicassem diferença de tipo de equipamento, entretanto é razoável que equipamentos de mesmo tipo também possuam grande distância, pois um tipo de equipamento pode compreender grande variedade de modelos de equipamento, como é o caso de tanque de carga, que pode apresentar grande diferença visual entre equipamentos, apesar de manter a característica comum de possuir tanque.
Possíveis adulterações de fotos correspondem às comparações classificadas como "sem diferença perceptível". Comparações entre imagens com distância inferior a 180 indicaram a mesma imagem em 100% da amostra e com distância inferior a 200 quase 80% ainda indicaram a mesma imagem, por isso o intervalo "menor que 200" foi escolhido para definir adulterações.

Fonte: Produção do Autor.

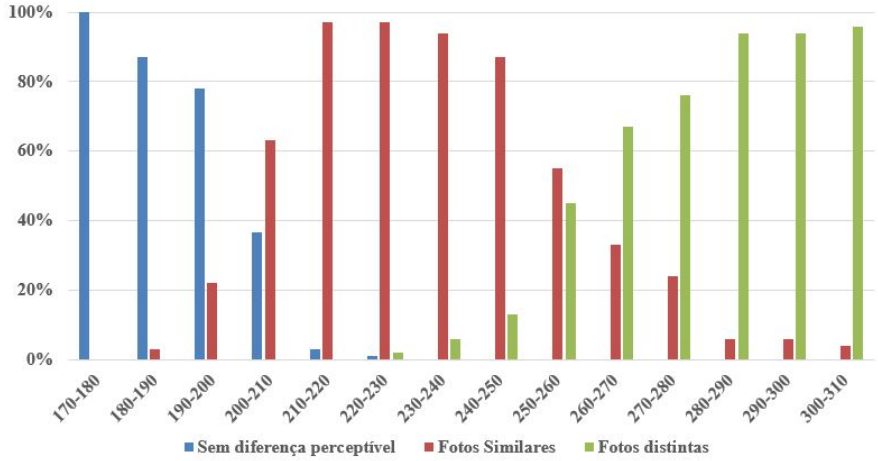


Fig. 3: Distribuição da categorização agrupada por intervalo.

4.3 VALIDAÇÃO

A validação dos métodos foi realizada a partir de uma amostra de 100 inspeções contendo 5 imagens clonadas e 10 adulteradas. Todas foram identificadas conforme previsto, sendo as 5 totalmente clonadas através do HASH e as 10 adulteradas através da similaridade com distâncias inferiores a 200.
Com a intenção de exemplificar visualmente o que seria uma adulteração e a comparação entre imagens, uma foto foi coletada e a placa do veículo alterada, conforme Fig. 4. Por se tratar de imagens diferentes, os HASH foram diferentes, excluindo a possibilidade de clonagem idêntica. Mas a distância semântica calculada foi 44,26 e sendo menor que 200 a ferramenta indicou como possível adulteração, o que corresponde à realidade. O código do exemplo foi disponibilizado em https://github.com/phortiz/ClonagemAdulteracao.
O intervalo definido para indicação de fotos similares foi de 210 a 250, pois teve mais de 80% de compatibilidade para cada faixa do intervalo. Nas fotos do banco do Inmetro foram encontradas 84.283 fotos distintas com similaridade entre 210 e 250, num total de 842.150 comparações. Foram selecionadas aleatoriamente 20 imagens para servirem de

modelo para comparação. Para cada uma delas foram selecionadas outras imagens cuja distância era maior ou igual a 210 e menor que 250, num total de 405 imagens. Tais imagens foram avaliadas visualmente e 367 (90,6%) apresentaram grande similaridade com a foto modelo.

Fonte: Extraído de chemicalrisk.com.br.

a – Foto original b – Foto com a placa adulterada

Fig. 4: Exemplo fictício de adulteração de placa.

4.4 DETECÇÃO DE CLONAGENS E ADULTERAÇÕES SUSPEITAS

A última etapa prevista foi a detecção de clonagens e adulterações suspeitas no banco de imagens do Inmetro. Os HASH de todas as imagens foram comparados e considerando que o envio de imagens iguais dentro de uma mesma inspeção não configuraria irregularidade, foram identificadas 542 imagens do equipamento completamente idênticas em mais de uma inspeção. Entretanto, é possível ter havido falha na impressão de alguns certificados e como o número do certificado vem impresso no documento, outro seria utilizado e o primeiro inutilizado. Para eliminar a possibilidade de envio em duplicidade por erro na impressão, foi incluída na análise a data da inspeção, que é enviada pelo organismo em um arquivo texto juntamente com as imagens. Considerando que o organismo poderia enviar as imagens de uma inspeção para o Inmetro e posteriormente enviar as mesmas fotos em outra inspeção, mas referenciando a data da primeira, a data de *download* da foto também foi avaliada. Foram encontradas 132 imagens duplicadas em inspeções realizadas em datas distintas ou enviadas em datas distintas, que são possíveis irregularidades de imagens clonadas completamente idênticas.

O intervalo definido entre 0 e 200 foi utilizado para encontrar no banco de imagens do Inmetro possíveis adulterações. Imagens com distância igual a 0 correspondem a clonagens idênticas e foram detectadas pelo HASH. Das 122.078.414 comparações, 1.630 tiveram distância inferior a 200, sendo 939 com HASH diferente, 91 para inspeções distintas e

38 em dias diferentes. Como comentado anteriormente, imagens iguais em uma mesma inspeção não configurariam irregularidade e existe a possibilidade de um mesmo ato de inspeção resultar em mais de um identificador de certificado de inspeção, mas somente um seria válido. Portanto, foram encontradas 38 possíveis clonagens utilizando a estratégia de identificar comparações com distância menor que 200, para inspeções distintas e em dias distintos.

5 CONCLUSÕES

O método proposto não apenas contribui com a produção científica, mas favorece a solução de problemas reais do Inmetro. Uma importante vantagem da utilização de ferramenta de TI para detecção de clonagens e adulterações é o melhor aproveitamento da equipe avaliadora do Inmetro. A ferramenta detecta suspeitas das irregularidades, supre o avaliador humano com a informação necessária para melhor aproveitamento da equipe e torna o processo de supervisão dos serviços de acreditação mais assertivo na identificação de falhas. A ferramenta detectou 132 imagens clonadas em inspeções distintas e 38 suspeitas de adulteração. Além disso, o métodos proposto aumenta a qualidade do banco de imagens do Inmetro, pois identifica dados sujos. Dados limpos são fundamentais para tomada de decisão exata, primordialmente em decisões apoiadas em ferramentas de TI que aprendem e decidem através de dados.
Foi evidenciado que imagens semelhantes têm camadas intermediárias de aprendizado similares. O método proposto obteve sucesso na detecção de clonagens e adulterações de fotos a partir da distância semântica, calculada a partir de camada intermediária do modelo de classificação com arquitetura RESNET50. Na validação proposta a partir de conjunto fictício de imagens, foram detectadas todas imagens clonadas e adulteradas sem nenhum falso positivo ou falso negativo, demostrando a exatidão do método. Fica como possível trabalho futuro a aplicação do método utilizando outras arquiteturas de aprendizagem profunda e com imagens de outros contextos.
Por fim, a validação do intervalo definido para detecção de imagens similares apresentou correspondência superior a 90%. O fato de existir intervalo de distância que defina similaridade de imagens com boa exatidão abre oportunidade para outras pesquisas, como por exemplo para identificar irregularidades semelhantes a partir de uma imagem de referência que contenha irregularidade.

Referências

[1] N. Aloysius and M. Geetha. A review on deep convolutional neural networks. In *2017 International Conference on Communication and Signal Processing (ICCSP)*, volume 5, pages 0588–0592. IEEE, apr 2017.

[2] ANTT. Resolução Nº 5.848/19 Regulamento para o Transporte Rodoviário de Produtos Perigosos, 2019. Disponível em: https://anttlegis.antt.gov.br/. Acesso em: 17 de out. de 2020.

[3] X. Feng, Y. Jiang, X. Yang, M. Du, and X. Li. Computer vision algorithms and hardware implementations: A survey. *Integration*, 69:309—-320, 2019.

[4] X. Hao, G. Zhang, and S. Ma. Deep Learning. *International Journal of Semantic Computing*, 10(03):417–439, 2016.

[5] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[6] INMETRO. NIT-DIOIS-019 Critérios específicos para a acreditação de organismos de inspeção, 2020. Disponível em: http://www.inmetro.gov.br/credenciamento/sobre_org_insp.asp?iacao=imprimir. Acesso em: 12 de set. de 2020.

[7] Y. Lecun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[8] H. Liu, Y. Tian, Y. Yang, L. Pang, and T. Huang. Deep relative distance learning: Tell the difference between similar vehicles. pages 2167–2175, 2016.

[9] M. Miller and J. V. Lent. Monge's optimal transport distance with applications for nearest neighbour image classification. *CoRR*, abs/1612.00181, 2016.

[10] P. H. ORTIZ, L. F. R. d. C. Carmo, and R. P. de Souza. Identificação de irregularidades e fraudes através de imagens para monitoramento de inspeções de transporte de produtos perigosos, 2020.

[11] A. Rosebrock. Detect and remove duplicate images from a dataset for deep learning, 2020. Disponível em: https://www.pyimagesearch.com/2020/04/20/detect-and-remove-duplicate-images-from-a-dataset-for-deep-learning/. Acessado em: 12 de set. de 2020.

[12] A. SACHAN. Detailed Guide to Understand and Implement ResNets. Disponível em: https://cv-tricks.com/keras/understand-implement-resnets/. Acessado em: 27 de jan. de 2021.

[13] C. Tan, F. Sun, T. Kong, W. Zhang, C. Yang, and C. Liu. A survey on deep transfer learning. *The 27th International Conference on Artificial Neural Networks (ICANN 2018)*, 11141 LNCS:270–279, 2018.

[14] S. Zagoruyko and N. Komodakis. Deep compare: A study on using convolutional neural networks to compare image patches. *Computer Vision and Image Understanding*, 164:38 – 55, 2017. Deep Learning for Computer Vision.

Anais do XIII Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 17 a 21 de Fevereiro de 2020.

Coarse-mesh method applied to 2D S_N neutron transport problems considering anisotropic scattering and multigroup theory

Rafael Barbosa Libotte[1], Hermes Alves Filho[1] e Ricardo Carvalho de Barros[1]

[1] *Instituto Politécnico/UERJ, Nova Friburgo/RJ, Brazil*

Abstract

We propose an extension of a coarse-mesh method used in the solution of X,Y-cartesian geometry fixed-source neutron transport problems, considering linearly anisotropic neutron scattering in the discrete ordinates formulation, using the energy multigroup theory. This method, named Modified Spectral Deterministic-Constant Nodal (*MSD-CN*), is a modification of the Spectral Deterministic Method (*SDM*), which uses an iterative process based on the *Source Iteration* algorithm and the analytical solution of the discretized neutron transport equation. Two model-problems are solved in this paper, and the numerical results obtained with the *MSD-CN* are compared to methods found in the literature.

Keywords:Neutron Transport Theory, Discrete Ordinates, Energy Multigroup, Fixed-source Problems

1 INTRODUCTION

In the pursuit of solving realistic neutron transport problems, the development of faster and more reliable algorithms are necessary to achieve results in a viable time. Although being able to solve a big portion of these problems with good precision, fine-mesh methods, e.g., *Diamond Difference* (*DD*) (Lewis and Miller, 1993), consumes a lot of CPU time, even with modern computers. Thus, the development of coarse-mesh methods have been the study object of several groups, e.g. *Spectral Deterministic Method* (*SDM*) (Oliva, 2018), *Spectral Green's Function* (*SGF*) (Barros and Larsen, 1992) and *Response Matrix* (da Silva, 2018), once it delivers results with good precision without the need of as much mesh refinement.

Contato: Rafael Barbosa Libotte, rafaellibotte@hotmail.com

Neutron transport problems are modelled with the linearized Boltzmann equation, which was originally developed to model gas kinetics (Bell and Glasstone, 1970). It is a balance equation, which weighs the production and loss of neutrons inside a unitary volume, considering that the neutron interactions with the medias nuclei does not affect its structure and there are no neutron-neutron interactions. It has 7 independent variables: 3 spatial, 2 angular, an energetic one and a temporal one. In this paper, we consider two-dimensional problems in stationary form. The angular variable is discretized according to the Discrete Ordinates technique (Lewis and Miller, 1993), and for the energetic variable, we consider the multigroup theory (Duderstadt et al., 1976).

In this paper, we present an extended study of the *Modified Spectral Deterministic-Constant Nodal* (*MSD-CN*), applied to two-dimensional fixed-source problems, considering linearly anisotropic neutron scattering. This is a hybrid method which uses the discretized neutron transport equations analytical solution alongside with an iterative process based on the *Source Iteration* scheme to calculate the neutron angular fluxes (dependent variable) in the domains nodal interfaces. Two model problems are solved in this work and the results are compared with the literature.

The paper is organized as follows. Section 2 shows the mathematical model of the neutron transport equation. In Section 3 we present the so called *MSD-CN*. In Section 4, it is shown the solution of the model-problems and the results obtained with different methods of solution. In Section 5, the conclusions and discussions of this work are presented.

2 MATHEMATICAL MODEL

The neutron transport in a non-multiplying media, considering a two-dimensional cartesian geometry, with linearly anisotropic scattering, stationary form, considering the discrete ordinates formulation and using the multigroup theory is given by

$$\mu_m \frac{\partial}{\partial x}\psi_{m,g}(x,y) + \eta_m \frac{\partial}{\partial y}\psi_{m,g}(x,y) + \sigma_{T,g}(x,y)\psi_{m,g}(x,y) = \frac{1}{4}\sum_{g'=1}^{G}\sum_{n=1}^{M}\left[\sigma_{S0}^{g'\to g}(x,y) + 3\sigma_{S1}^{g'\to g}(x,y)(\mu_m\mu_n + \eta_m\eta_n)\right]\psi_{n,g'}(x,y)\omega_n + Q_g(x,y), \quad m = 1 \le M,\ g = 1 \le G, \quad (1)$$

where ψ represents the neutron angular flux in the direction of index m, σ_T the macroscopic total cross section, σ_{S0} and σ_{S1} are respectively the macroscopic scattering cross sections of zero-th and first degree from the energy group g' to g, Q is an external neutrons fixed-source, and G is the number of energy groups modelled. The constants μ and η represents the roots (that describes the polar angle of the neutrons incidence), and ω the weights of the Level-Symmetric Quadrature (LQ_N), used in the angular variable discretization of order N. The number of discrete directions is calculated as $M = N(N+2)/2$, with $M/4$ discrete directions in each quadrant. This scheme can be seen in Figure 1. Here, we use the LQ_6 set as an example. The different kinds of lines represents the different weights used for each direction set.

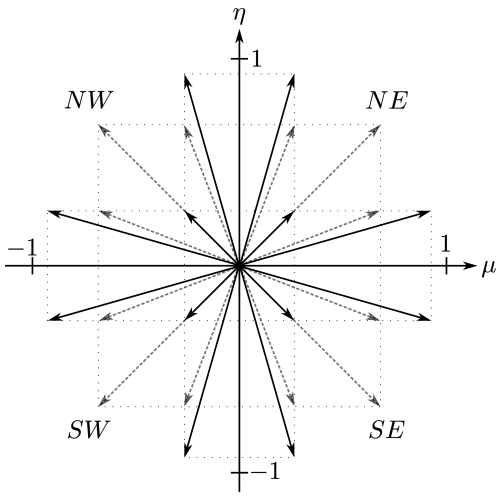


Fig. 1: Directions and axial coordination representation.

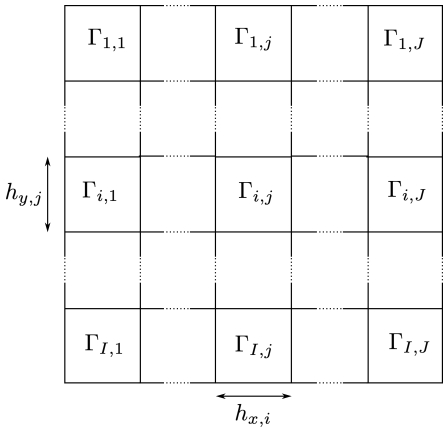


Fig. 2: Spatial grid $\Gamma_{x,i} \times \Gamma_{y,j}$

The analytical solution of Eq. (1) can be achieved only in a very restricted number of problems. Thus, we can divide the domain into a grid of $I \times J$ nodes ($\Gamma_{i,j}$), and apply Eq. (1) in each one of these nodes, resulting in the intranodal neutron transport equation, which has constant physical-material parameters, as seen in Eq. (2)

$$\mu_m \frac{\partial}{\partial x}\psi_{m,g}(x,y) + \eta_m \frac{\partial}{\partial y}\psi_{m,g}(x,y) + \sigma_{T,g,i,j}\psi_{m,g}(x,y) = \sum_{g'=1}^{G}\sum_{n=1}^{M} S_{1,m}\psi_{n,g'}(x,y) + Q_{g,i,j}, \;\; m = 1 \leq M, \;\; g = 1 \leq G. \quad (2)$$

where the scattering source is defined as

$$S_{1,m} = \frac{\omega_n}{4}\left[\sigma_{S0,i,j}^{g' \to g} + 3\sigma_{S1,i,j}^{g' \to g}(\mu_m\mu_n + \eta_m\eta_n)\right].$$

With this equation, we can obtain the intranodal analytical solution, and proceed to solve the neutron shielding problems presented in this work.

3 MODIFIED SPECTRAL DETERMINISTIC-CONSTANT NODAL (MSD-CN)

The *MSD-CN* can be described as a hybrid method, which uses the analytical solution of the intra-nodal neutron transport equation and an iterative process that is based on the *Source Iteration* scheme, used in the traditional fine-mesh method *Diamond Difference* (Lewis and Miller, 1993). This method relies on only one approximation (node-edge incoming and outgoing neutron angular fluxes), which allows it to solve problems within a good precision in coarser meshes when compared to fine-mesh methods.

The first step to derive the solution via *MSD-CN* is to integrate the nodes in the coordinated x, and divide the result by the size of the node $h_{x,i}$, resulting in

$$\mu_m \frac{\mathrm{d}}{\mathrm{d}x}\widetilde{\psi}_{m,g,j}(x) + \frac{\eta_m}{h_{y,j}}\left(\psi_{m,g,j+\frac{1}{2}}(x) - \psi_{m,g,j-\frac{1}{2}}(x)\right) + \sigma_{T,g,i,j}\widetilde{\psi}_{m,g,j}(x) = \sum_{g'=1}^{G}\sum_{n=1}^{M} S_{1,m}\widetilde{\psi}_{n,g'}(x,y) + Q_{g,i,j},\ m = 1 \leq M,\ g = 1 \leq G \quad (3)$$

analogously, to the coordinated axis y, we have

$$\frac{\mu_m}{h_{x,i}}\left(\psi_{m,g,i+\frac{1}{2}}(y) - \psi_{m,g,i-\frac{1}{2}}(y)\right) + \eta_m \frac{\mathrm{d}}{\mathrm{d}y}\widehat{\psi}_{m,g,i}(y) + \sigma_{T,g,i,j}\widehat{\psi}_{m,g,i}(y) = \sum_{g'=1}^{G}\sum_{n=1}^{M} S_{1,m}\widehat{\psi}_{n,g'}(x,y) + Q_{g,i,j},\ m = 1 \leq M,\ g = 1 \leq G,. \quad (4)$$

where the transverse integrated angular fluxes, in coordinated directions x and y are respectively

$$\widetilde{\psi}_{m,g,j}(x) = \frac{1}{h_{y,j}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}} \psi_{m,g}(x,y)\mathrm{d}y$$

$$\widehat{\psi}_{m,g,i}(y) = \frac{1}{h_{x,i}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \psi_{m,g}(x,y)\mathrm{d}x.$$

Knowing that the system of Eqs. (3–4) has GM equations and $2GM$ unknowns, being GM unknowns represented by the neutron angular fluxes and GM by the transverse neutron leakage terms, described by

$$\frac{\eta_m}{h_{y,j}}\left(\psi_{m,g,j+\frac{1}{2}}(x) - \psi_{m,g,j-\frac{1}{2}}(x)\right) \text{ and } \frac{\mu_m}{h_{x,i}}\left(\psi_{m,g,i+\frac{1}{2}}(y) - \psi_{m,g,i-\frac{1}{2}}(y)\right),$$

we must perform an approximation in order to solve this system of equations. For this, we consider the neutron angular flux as constant in the node-edge, in a mathematical point of view, we have

$$\psi_{m,g,i,j\pm\frac{1}{2}}(x) \approx \widehat{\psi}_{m,g,i,j\pm\frac{1}{2}} \text{ and } \psi_{m,g,i\pm\frac{1}{2},j}(y) \approx \widetilde{\psi}_{m,g,i\pm\frac{1}{2},j}.$$

Thus, we can write the transverse neutron leakage as

$$\frac{\eta_m}{h_{y,j}}\left(\widehat{\psi}_{m,g,i,j+\frac{1}{2}} - \widehat{\psi}_{m,g,i,j-\frac{1}{2}}\right) = \widehat{L}_{m,g,i,j} \text{ and } \frac{\mu_m}{h_{x,i}}\left(\widetilde{\psi}_{m,g,i+\frac{1}{2},j} - \widetilde{\psi}_{m,g,i-\frac{1}{2},j}\right) = \widetilde{L}_{m,g,i,j}.$$

Now, with the unicity of the transverse integrated intranodal equation solution in the node $\Gamma_{i,j}$ guaranteed, we can rewrite Eqs. (3–4) respectively as

$$\mu_m \frac{\mathrm{d}}{\mathrm{d}x}\widetilde{\psi}_{m,g,j}(x) + \widehat{L}_{m,g,i,j} + \sigma_{T,g,i,j}\widetilde{\psi}_{m,g,j}(x) = \frac{1}{4}\sum_{g'=1}^{G}\sum_{n=1}^{M} S_{1,m}\widetilde{\psi}_{n,g'}(x,y) + Q_{g.i,j},\ m = 1 \leq M,\ g = 1 \leq G, \quad (5)$$

$$\widetilde{L}_{m,g,i,j} + \eta_m \frac{\mathrm{d}}{\mathrm{d}y}\widehat{\psi}_{m,g,i}(y) + \sigma_{T,g,i,j}\widehat{\psi}_{m,g,i}(y) = \frac{1}{4}\sum_{g'=1}^{G}\sum_{n=1}^{M} S_{1,m}\widehat{\psi}_{n,g'}(x,y)\omega_n + Q_{g,i,j},\ m = 1 \leq M,\ g = 1 \leq G. \quad (6)$$

With this system of equations, we can work in the analytical solution.

3.1 Spectral Analysis

The solution of Eqs. (5–6) is divided into an homogeneous and a particular component, in the form

$$\widetilde{\psi}_{m,g}(x) = \widetilde{\psi}^h_{m,g}(x) + \widetilde{\psi}^p_{m,g}$$

The particular solution is given by (Oliva, 2018)

$$\sum_{g'=1}^{G}\sum_{n=1}^{M}\left(\sigma_{T,g,i,j}\delta_{mn}\delta_{g'g} - \frac{1}{4}\sigma^{g'\to g}_{S0,i,j}\omega_n + \frac{3}{4}\sigma^{g'\to g}_{S1,i,j}(\mu_m\mu_n + \eta_m\eta_n)\right)\widetilde{\psi}^p_{n,g'} = Q_{g,i,j} - \widehat{L}_{m,g,i,j}.$$

where δ represents Kronecker's delta. For the homogeneous part of the solution, consider the expression

$$\widetilde{\psi}^h_{m,g,j}(x) = a^x_{m,g}(\vartheta^x)\exp\left(\frac{-\left(x - x_{j-\frac{1}{2}}\right)}{\vartheta^x}\right). \quad (7)$$

Substituting Eq. (7) in the homogeneous part of Eq. (5), we obtain the following eigenvalue problem

$$\frac{1}{\mu_m}\sum_{g'=1}^{G}\sum_{n=1}^{M}\left(\sigma_{T,g,i,j}\delta_{mn}\delta_{g'g} - \frac{1}{4}\sigma^{g'\to g}_{S,i,j}\omega_n + \frac{3}{4}\sigma^{g'\to g}_{S1,i,j}(\mu_m\mu_n + \eta_m\eta_n)\right)a^x_{n,g'}(\vartheta^x) = \frac{1}{\vartheta^x}a^x_{m,g}(\vartheta^x)$$

Solving this problem, we obtain a set of MG symmetric eigenvalues (ϑ) and MG eigenvectors ($\boldsymbol{a}(\vartheta)$). Thus, we can write the solution of the intranodal neutron transport equation for the coordinated axis x as

$$\widetilde{\psi}_{m,g,j}(x) = \sum_{l=1}^{GM} \alpha_l^x a_{m,g}^x(\vartheta_l^x) \exp\left(\frac{-\left(x - x_{i-\frac{1}{2}}\right)}{\vartheta_l^x}\right) + \widetilde{\psi}_g^p \tag{8}$$

Analogously, for the y coordinated axis we have

$$\widehat{\psi}_{m,g,i}(y) = \sum_{l=1}^{GM} \alpha_l^y a_{m,g}^y(\vartheta_l^y) \exp\left(\frac{-\left(y - y_{j-\frac{1}{2}}\right)}{\vartheta_l^y}\right) + \widehat{\psi}_g^p. \tag{9}$$

3.2 Iterative Process

In order to obtain the equations used to iterate the neutron angular fluxes in the node-edges, we have to calculate the average neutron angular fluxes for each coordinated axis, according to

$$\overline{\psi}_{m,g,i,j}^x = \frac{1}{h_{x,i}} \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \widetilde{\psi}_{m,g,i}(x)\mathrm{d}x = \frac{1}{h_{x,i}} \sum_{l=1}^{GM} \alpha_l^x a_{m,g}^x(\vartheta_l^x) \left(e^{\frac{-h_{x,i}}{\vartheta_l^x}} - 1\right) + \widetilde{\psi}_g^p. \tag{10}$$

With this system of equations, we proceed to apply the two-dimensional average operator in the intranodal neutron transport equation, resulting in the equations used to calculate the neutron angular fluxes respectively in the right and left node-edges

$$\begin{aligned}\widetilde{\psi}_{m,g,i+\frac{1}{2},j} = \widetilde{\psi}_{m,g,i-\frac{1}{2},j} + \frac{h_{x,i,j}}{\mu_m}\Bigg(&- \widehat{L}_{m,g,i,j} - \sigma_{T,g,i,j}\overline{\psi}_{m,g,i,j}^{\,x} \\ &+ \frac{1}{4}\sum_{g'=1}^{G}\sum_{n=1}^{M}\left[\sigma_{S0,i,j}^{g'\to g} + 3\sigma_{S1,i,j}^{g'\to g}(\mu_m\mu_n + \eta_m\eta_n)\right]\overline{\psi}_{n,g'}^{\,x}(x,y)\omega_n + Q_{g,i,j}\Bigg)\end{aligned} \tag{11}$$

$$\begin{aligned}\widetilde{\psi}_{m,g,i-\frac{1}{2},j} = \widetilde{\psi}_{m,g,i+\frac{1}{2},j} + \frac{h_{x,i,j}}{|\mu_m|}\Bigg(&- \widehat{L}_{m,g,i,j} - \sigma_{T,g,i,j}\overline{\psi}_{m,g,i,j}^{\,x} \\ &+ \frac{1}{4}\sum_{g'=1}^{G}\sum_{n=1}^{M}\left[\sigma_{S0,i,j}^{g'\to g} + 3\sigma_{S1,i,j}^{g'\to g}(\mu_m\mu_n + \eta_m\eta_n)\right]\overline{\psi}_{n,g'}^{\,x}(x,y)\omega_n + Q_{g,i,j}\Bigg)\end{aligned} \tag{12}$$

Analogously, the same procedure is done in the y coordinated axis, resulting in the equations to calculate the angular neutron flux in the upper and lower node-edges, respectively

$$\begin{aligned}\widehat{\psi}_{m,g,i,j+\frac{1}{2}} = \widehat{\psi}_{m,g,i,j-\frac{1}{2}} + \frac{h_{y,i,j}}{\eta_m}\Bigg(&- \widetilde{L}_{m,g,i,j} - \sigma_{T,g,i,j}\overline{\psi}_{m,g,i,j}^{\,y} \\ &+ \frac{1}{4}\sum_{g'=1}^{G}\sum_{n=1}^{M}\left[\sigma_{S0,i,j}^{g'\to g} + 3\sigma_{S1,i,j}^{g'\to g}(\mu_m\mu_n + \eta_m\eta_n)\right]\overline{\psi}_{n,g'}^{\,y}(x,y)\omega_n + Q_{g,i,j}\Bigg)\end{aligned} \tag{13}$$

$$\widehat{\psi}_{m,g,i,j-\frac{1}{2}} = \widehat{\psi}_{m,g,i,j+\frac{1}{2}} + \frac{h_{y,i,j}}{|\eta_m|}\Bigg(- \widetilde{L}_{m,g,i,j} - \sigma_{T,g,i,j}\overline{\psi}^{\,y}_{m,g,i,j} + \frac{1}{4}\sum_{g'=1}^{G}\sum_{n=1}^{M}\left[\sigma^{g'\to g}_{S0,i,j} + 3\sigma^{g'\to g}_{S1,i,j}(\mu_m\mu_n + \eta_m\eta_n)\right]\overline{\psi}^{\,y}_{n,g'}(x,y)\omega_n + Q_{g,i,j}\Bigg) \quad (14)$$

In order to perform the iterative process, four sweeping directions are made: $SW \to NE$, $SE \to NW$, $NE \to SW$ and $NW \to SE$, calculating the outgoing angular neutron fluxes as a function of the incoming ones, according to Eqs. (12–14). Before every sweeping step, the average neutron angular fluxes and scattering sources are updated in each node. After updating the angular scalar flux, the average scalar flux in each node is calculated, using the expression

$$\overline{\phi}^{\,u}_{g,i,j} = \frac{1}{4}\sum_{n=1}^{M}\overline{\psi}^{u}_{n,g,i,j}\omega_n,$$

where u represents any of the coordinated axis. With these values, the stopping criterion can be evaluated comparing the maximum relative deviation between the average scalar flux of two subsequent iterations

$$\left|\frac{\overline{\phi}^{\,u,k+1}_{g,i,j} - \overline{\phi}^{\,u,k}_{g,i,j}}{\overline{\phi}^{\,u,k}_{g,i,j}}\right| < \xi \quad (15)$$

where k represents the number of the iteration evaluated, and ξ is a predefined precision parameter.

4 NUMERICAL EXPERIMENTS

In this work, we present the solution of two model-problems solved with the *MSD-CN*, using the language C++, and compare the precision of the obtained results with ones found in the literature. Both models are two-dimensional, considering a linear anisotropy degree neutron scattering. A fine-mesh solution using the *Diamond Difference* method is used as a results precision reference.

4.1 Model-Problem 1

The first model-problem, consists in an homogeneous fixed-source problem within a 1×1 cm square domain divided in 4 regions, as seen in Figure 3 (Picoloto et al., 2017). The physical-material parameters are $\sigma_T = 0.8\ cm^{-1}$, $\sigma_{S0} = 0.4\ cm^{-1}$ and $\sigma_{S1} = 0.2\ cm^{-1}$. This model has reflective boundary conditions in the left and bottom sides, and vacuum boundary condition in the upper and right sides. An external source of neutrons $Q = 1.0$ is evenly distributed inside the first region. In the solution of this problem we used various sets of the LQ_N quadrature and grids, in order to calculate the average angular neutron fluxes in each spatial region, and test the precision of the results, by comparing it with a fine-mesh reference method.

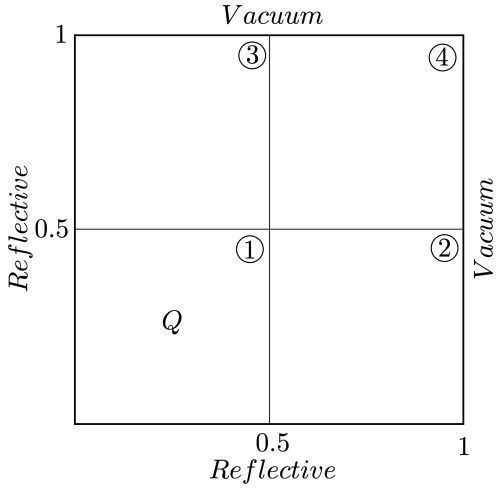


Fig. 3: Model-Problem 1 geometry

The average angular fluxes in regions 1, 2 and 4 are shown in Table 1.

Table 1: Model-problem 1 average angular fluxes numerical results

		$MSD-CN$			DD	Relative
Region	N	2×2	4×4	8×8	100×100	deviation (%)[a]
1	4	0.55673	0.55862	0.55933	0.55951	0.032
	8	0.57017	0.57091	0.57140	0.57148	0.015
	16	0.57388	0.57522	0.57551	0.57556	0.000
2	4	0.22504	0.21785	0.21557	0.21512	0.180
	8	0.22479	0.21740	0.21622	0.21596	0.120
	16	0.22374	0.21698	0.21615	0.21581	0.157
4	4	0.15268	0.15226	0.15221	0.15166	0.362
	8	0.13424	0.13523	0.13376	0.13362	0.104
	16	0.12855	0.12896	0.12832	0.12840	0.062

[a] Relative deviation between *MSD-CN* 8×8 and *Diamond Difference* 100×100.

The numerical results obtained with *MSD-CN* had a good precision within the stopping critterion used in this model, when comparing with the fine-mesh reference.

4.2 Model-problem 2

The second model-problem, adapted from (Curbelo and Barros, 2021), considers a two-layer heterogeneous shielding material around an external isotropic neutron source, as seen in Figure 4. This problem considers 10 groups of energy and linearly anisotropic neutron scattering. The physical-material parameters (cm^{-1}) of the 3 heterogeneous zones ($z = 1:3$), are given by

$$\sigma_{T,g,z} = \left(\frac{z+5}{21}\right)^5 \left(\frac{g}{10} - 0.15\delta_{5,g} - 0.15\delta_{10,g}\right),\ g = 1 \leq 10$$

and

$$\sigma_{Sl,z}^{g'\to g} = \left(\frac{z+20}{21}\right)\left(\frac{g'}{100(g-g'+1)}\right)\left(0.7-\frac{g+g'}{200}\right)^{l}, g=1\leq 10,\ g'=1\leq g,\ l=0\leq 1.$$

In the region that comprehends Zone 1, there is an evenly distributed neutron source

$$Q_g = 1.1 - 0.1g,\ \ g = 1 \leq 10.$$

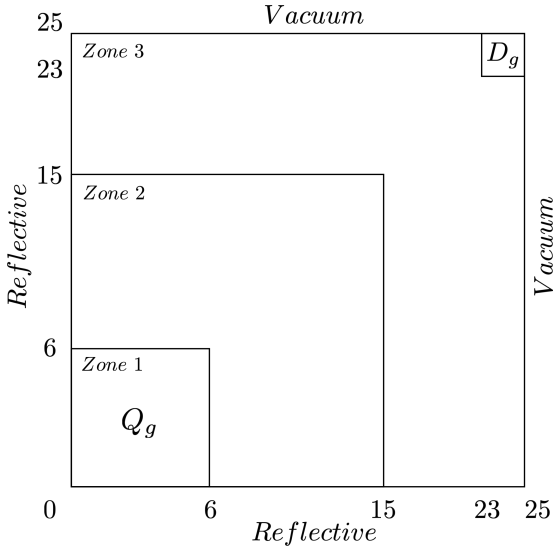


Fig. 4: Model-Problem 2 geometry

In the solution of this problem, we calculate the group absorption rate in a 2×2 cm detector D_g, located in the upper right position of the outer shielding materials region. It is considered a precision of $\xi = 10^{-6}$. The results obtained with the *MSD-CN* are compared to the fine-mesh method *DD* and a coarse-mesh one, denominated *SGF*. In this model, the precision of the results are compared in 3 different groups of energy, as shown in Table 2. The absortion rate is calculated as

$$T_{g,i,j}^{u} = \sigma_{a,g,i,j}\overline{\phi}_{g,i,j}^{u} h_{x,i} h_{y,j} \tag{16}$$

Table 2: Model-problem 2 absorption rate numerical results - 25×25 - S_4

Method	$g=1$	$g=5$	$g=10$
DD[a]	5.75851×10^{-4}	1.24058×10^{-5}	7.09877×10^{-6}
MSD-CN	5.96782×10^{-4}	1.29590×10^{-5}	7.50277×10^{-6}
SGF-CN	5.96783×10^{-4}	1.29590×10^{-5}	7.50278×10^{-6}
Relative Deviation (%)[b]	0.036	0.044	0.056

[a] Fine mesh reference 500×500.
[b] Relative deviation between *MSD-CN* and *DD*.

In the model-problem, the results obtained with *MSD-CN* also had good precision, when compared with the reference method and the *SGF-CN*.

5 CONCLUSIONS

In this work, we presented an extended study of a methodology to solve two-dimensional cartesian geometry neutron transport problems, named *Modified Spectral Deterministic-Constant Nodal* (*MSD-CN*). The problems solved in this paper considers a linearly anisotropic neutron scattering, in a stationary state. In the mathematical model, the angular variables are discretized using the discrete ordinates formulation, and the energetic variable is treated according to the multigroup theory. Two problems are solved, and the precision of the obtained results are compared with ones found in the scientific literature.

In the solution of the first model-problem, the *MSD-CN* was able to achieve results with good precision, when compared to the fine-mesh reference. As the MSD-CN is not free of truncation error, the refining of the mesh results in results that are closer to the fine-mesh reference. The second model-problem, as the first one, showed results close to the reference within the tested precision.

Regardless of obtaining precise results when compared to the literature, the *MSD-CN* required a lot of computational time in the tested model-problems. As these are the initial studies, we are primarily concerned with the methods precision, but further investigations must be done in the iterative process, in order to decrease the execution time.

For future works, the group intends to implement new kinds of approximation of the node-edge neutron angular flux, such as the linear one, and investigate ways to improve the computational time to solve fixed-source problems.

6 *Acknowledgements*

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001

REFERENCES

[1] Barros, R. C. and Larsen, E. W. (1992). "A spectral nodal method for one-group x, y-geometry discrete ordinates problems". *Nuclear Science and Engineering*, 111(1):34–45.

[2] Bell, G. I. and Glasstone, S. (1970). *Nuclear Reactor Theory*, volume 2. Krieger Pub Co.

[3] Curbelo, J. P. and Barros, R. C. (2021). "A spectral nodal method for the adjoint sn neutral particle transport equations in x,y-geometry: application to direct and inverse multigroup source-detector problems". *Annals of Nuclear Energy*, 150C.

[4] da Silva, O. P. (2018). *Um método de matriz resposta para cálculos de transporte multigrupos de energia na formulaçáo de ordenadas discretas em meios não-multiplicativos.* Tese de Doutorado, IPRJ/UERJ, Nova Friburgo.

[5] Duderstadt, J. J., Hamilton, L. J., et al. (1976). *Nuclear reactor analysis.* Wiley New York.

[6] Lewis, E. E. and Miller, W. F. (1993). "Computational methods of neutron transport". pages 400.

[7] Oliva, A. M. (2018). *Método Espectral Determinístico para a solução de problemas de transporte de nêutrons usando a formulação das ordenadas discretas.* Tese de Doutorado, IPRJ/UERJ, Nova Friburgo.

[8] Picoloto, C. B., da Cunha, R. D., Barros, R. C., and Barichello, L. B. (2017). "An analytical approach for solving a nodal formulation of twodimensional fixed-source neutron transport problems with linearly anisotropic scattering". *Progress in Nuclear Energy*, 98:193–201.

Anais do XIII Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 17 a 21 de Fevereiro de 2020.

Predição da aprovação do público para filmes utilizando aprendizado de máquina

Rafael de Souza Terra[1]

[1] *Laboratório Nacional de Computação Científica, Petrópolis/RJ, Brazil*

Abstract

Due to the film industry's growing, more and more movies are released per year. A movie that gets a bad acceptance by the public can bring a lot of loss to the producer. Hence, it's perceptible the rise in release of franchises. The franchises stablish a kind of "cake recipe" where, if a movie is successful, other movies like them will be released aiming for the same success. Because of this "cake recipe" that is created by various elements like cast; crew; genre and even movie rating, a question appears: would it be possible to predict the score of a movie using only these factors? This work realized the data mining of the movies from the Rotten Tomatoes website, the treatment of this data and the use of machine learning techniques to predict such score. However, the results showed using these techniques, despite of presenting a relative low error, weren't so satisfactory when the nature of the problem was accounted for.

Keywords: machine-learning, movies, regression, perceptron

1 INTRODUÇÃO

Com o crescimento da indústria cinematográfica, o número de filmes lançados por ano aumenta significativamente. É evidente que a má reputação de um filme resulta em um grande prejuízo para a produtora. Graças a isso, muitos dos filmes de sucesso atuais são franquias (filmes que são feitos seguindo "receitas de bolos"), ou seja, utilizando de forma exaustiva sempre os mesmos elementos que fazem sucesso com o público. Esse sucesso pode ser gerado por inúmeros fatores como: roteirista, diretor do filme, atores no elenco, gênero do filme, aprovação dos críticos e até mesmo a classificação indicativa.

Devido ao sucesso das grandes franquias, cada vez menos é notável o lançamento de filmes que não deixem o final em aberto, propondo assim uma continuação. Isso faz com que a indústria fique cada vez mais homogênea, pois gerar um filme diferente da "receita de bolo" pode significar prejuízo para a produtora, sendo um risco desnecessário.

Contato: Rafael de Souza Terra, rafaelst@lncc.br

Considerando os possíveis elementos de sucesso citados nessa seção, o presente trabalho tem como objetivo determinar, utilizando abordagens de aprendizado de máquina, qual seria a aprovação do público para um determinado filme. Para realizar esse objetivo é necessário realizar um conjunto de tarefas:

- Obtenção dos dados: extrair os dados dos filmes já lançados de algum *website* especializado;
- Tratamento dos dados: após extrair os dados, limpá-los e adaptá-los para servirem de entrada para o algoritmo de aprendizado de máquina;
- Escolha e treinamento do algoritmo de aprendizado de máquina: escolher uma abordagem para o problema, implementá-la e realizar o treinamento com os dados.

Esse trabalho está segmentado de forma que a próxima seção aborda todo esse conjunto de tarefas, a seção 3 exibe todos os testes feitos e a seção 4 aborda as considerações finais e possíveis trabalhos futuros.

2 TRABALHOS RELACIONADOS

Armstrong and Yoon (1995) utilizaram um conjunto de cerca de 1860 filmes obtidos do *website IMDb*[1](onde os filmes possuem uma avaliação de 0 a 10) para realizar a predição da aprovação do público com base em um conjunto de características de cada filme, como: diretores, atores, orçamento, roteiristas, ano do filme, entre outras. Para tentar obter a predição, foram utilizadas as técnicas de regressão *kernel* e árvore modelo, conseguindo um erro de $14,11\%$ para a regressão *kernel* e $14,40\%$ para a árvore modelo.

Marović et al. (2011) também fizeram uso de dados do *IMDb*, agregando informações sobre os usuários para tentar predizer a aprovação do público. Para realizar esse objetivo foram utilizados vários métodos como redes neurais artificiais, árvore de regressão, *K-nearest neighbors*, diagnósticos de personalidade, variáveis latentes e SVD-kNN. Com o melhor método (variáveis latentes) apresentando um erro de $1,739$ na métrica de raiz do erro quadrático médio.

3 METODOLOGIA

3.1 Obtenção dos dados

Quando dados são coletados para serem utilizados em análises estatísticas, normalmente são armazenados em uma matriz chamada de conjunto de dados (*data set*). Cada linha dessa matriz representa uma observação ou unidade desse conjunto de dados, recebendo o nome de instância e cada coluna representa uma característica do dado, sendo chamada atributo (Heumann et al., 2016).

Existem diversos *websites* especializados em críticas de filmes, um deles é o *Rotten Tomatoes*[2], que contém uma ampla base de dados com os filmes já lançados, juntamente com suas informações. Para utilizar esses dados foi aplicada a técnica de *web scraping*. *Web scraping* é uma técnica que utiliza algoritmos para obter dados de *websites* de forma

[1] `https://www.imdb.com/`
[2] `https://www.rottentomatoes.com/`

automática (Mitchell, 2018). Para esse trabalho foi utilizada a biblioteca *Selenium*[3] em sua implementação na linguagem *Python 3* (Van Rossum and Drake, 2011).

A biblioteca realiza a extração dos dados, utilizando para a busca dos mesmos expressões regulares ou o *XPath*. Expressões regulares são conjuntos de caracteres que representam padrões para a formação de novas sequências de caracteres, um exemplo pode ser visto na Fig. 1 (Sidhu and Prasanna, 2001).

O *XPath* consiste em uma árvore que modela um documento em formato *XML*, definindo um caminho no formato de uma sequência de caracteres para cada tipo de nó (nó de atributo, elemento ou texto) e provê vários tipos de expressões que podem ser construídas por palavras-chaves, símbolos e operadores (que geralmente são outras expressões). Além da possibilidade de aninhar essas expressões (Clark et al., 1999; Consortium et al., 2010).

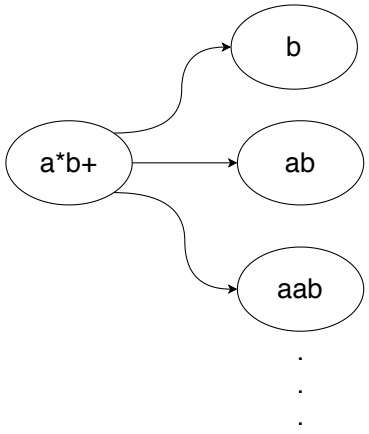


Fig. 1: Exemplo de uma expressão regular e as sequências formadas por ela. Essa expressão tem como padrão a geração de zero ou mais 'a' seguidos de um ou mais 'b'

O uso de expressões regulares complexas pode acarretar em uma perda de desempenho durante a extração dos dados, por isso a maioria das informações foram extraídas utilizando o *XPath*. Contudo, o *XPath* apresenta um pequeno empecilho: o *layout* da página precisa ser idêntico ao utilizado para a pesquisa do dado, caso o contrário o nó da árvore não é encontrado.

O *Rotten Tomatoes* apresenta uma página que contém todas as *url's* dos filmes. Para obter essa informação foi utilizado o *Selenium* passando como parâmetro de busca uma expressão regular (Eq. 1). Essa expressão consiste em encontrar, em linhas gerais, qualquer sequência de caracteres que comece com $href$, tenha "$/m/$" no meio e termine com o caractere ". Com isso foi possível obter todas as *url's* dos filmes presentes no *website*.

$$href\backslash s* = "/m/\backslash w*?" \tag{1}$$

De posse das *url's*, foram extraídas as informações de cada filme utilizando o *XPath* para a busca. Devido ao problema já mencionado do *XPath*, uma parcela dos filmes apresentou problemas durante a extração, resultando assim em filmes com dados faltantes. Toda a implementação do algoritmo para a extração pode ser encontrada em seu repositório do GitHub[4].

[3] `https://www.selenium.dev/documentation/en/webdriver/`
[4] `https://github.com/rafaelstjf/Tomato_Brusher`

3.2 Tratamento dos dados

Foi obtido um conjunto de dados de 8696 filmes, porém esse conjunto estava cheio de ruído, isso não é algo interessante para o algoritmo de aprendizado de máquina. Com isso, todos os filmes que apresentavam atributos faltantes foram removidos, resultando em um total de 7214 filmes que apresentam uma boa distribuição das avaliações, como está apresentado nos histogramas da Fig. 2. Em cada filme estão presentes os seguintes atributos: *title, tomatometer, user score, rating, genre, directed by, written by* e *cast*, onde apenas o *tomatometer* e o *user score* são atributos numéricos. Um exemplo de instância do conjunto pode ser visto na Tabela 1.

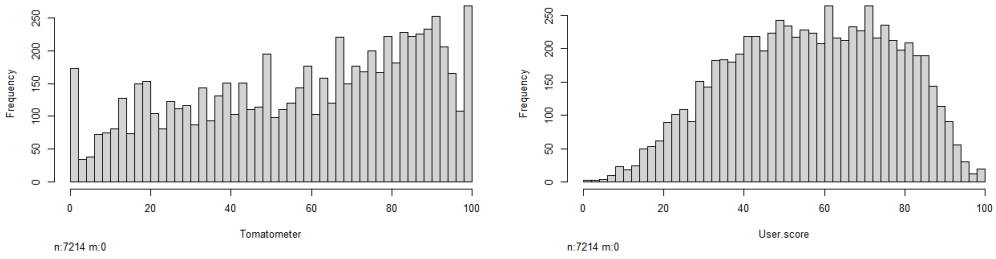


Fig. 2: Histogramas apresentando a distribuição das avaliações dos usuários e do *Tomatometer*

Tabela 1: Exemplo de instância do conjunto de dados

Title	Tomatometer	User score	Rating	Genre	Directed by	Written by	Cast
Mosul	100	100	NR	Documentary	Dan Gabriel	Dan Gabriel, Mike Tucker	Ali Mula

Os filmes possuem uma grande diversidade em seus atributos, alguns chegando a ter mais um elemento em cada. Além disso, cada atributo possui uma quantidade variável de elementos. Esse problema foi tratado em dois passos. O primeiro passo consistiu em separar os conjuntos de elementos, colocando apenas um por atributo e criando novos atributos quando necessário. Isso aumentou drasticamente a dimensionalidade do conjunto de dados e gerou um grande número de valores faltantes. No segundo passo, o número de atributos foi limitado (Tabela 2) na tentativa de amenizar o problema gerado pelo passo anterior. Para uso nos testes foi adotado o nome "conjunto padrão" para o conjunto de dados original e "conjunto limitado" para o conjunto de dados resultante do tratamento. A Tabela 3 exibe a representação de uma instância após o tratamento.

Tabela 2: Quantidade máxima dos atributos dos filmes

Atributo	Quantidade máxima
Genre	2
Directed by	2
Written by	2
Cast	5

Tabela 3: Instância da Tabela 1 após o tratamento dos dados. Os atributos Title, Tomatometer, User score e Rating foram removidos da tabela para uma melhor visualização

Genre 1	Genre 2	Directed by 1	Directed by 2	Written by 1	Written by 2	Cast 1	Cast 2	Cast 3	Cast 4	Cast 5
Documentary		Dan Gabriel		Dan Gabriel	Mike Tucker	Ali Mula				

3.3 Escolha e treinamento do algoritmo

O problema proposto é um típico problema de regressão, ou seja, um problema que dado um conjunto de dados deseja-se como saída um valor contínuo. Existem diversas abordagens de aprendizado de máquina para regressão, as utilizadas nesse trabalho foram a Árvore de Regressão e o *Perceptron* multicamadas.

A Árvore de Regressão é baseada em uma árvore de decisão. Em sua versão básica elas utilizam as mesmas métricas para a construção e poda da árvore, porém sua saída consiste em um número contínuo ao invés de um valor discreto (Breiman et al., 1984). Árvore de decisão é uma das principais abordagens quando se fala em classificação de um conjunto de dados. Ela classifica as instâncias, ordenando-as pelos seus atributos em uma estrutura de árvore, da raiz até as folhas, onde está localizada a classificação. Existem diversos algoritmos para construção de uma árvore de decisão. O algoritmo mais básico, o ID3, utiliza uma métrica de teoria da informação chamada Ganho de Informação para a escolha da raiz das subárvores. O Ganho de Informação mede o quão bem um determinado atributo consegue dividir o conjunto de dados, para isso ele faz o uso da entropia, uma métrica que mede a pureza de um determinado conjunto de dados. Outros algoritmos sucessores do ID3 são o C4.5, C5.0 e o CART (Mitchell, 1997).

Para a implementação de árvores de regressão nesse trabalho, foi utilizada a biblioteca *CatBoost* (Dorogush et al., 2018) disponível na linguagem de programação *Python 3*. Essa biblioteca traz como vantagem a possibilidade de utilizar conjuntos de dados com atributos categóricos e faz uso de um método chamado *gradient boosting* durante a construção das árvores. *Gradient boosting* é uma técnica de aprendizado de máquina utilizada para a obtenção de melhores resultados. Ela é realizada combinando iterativamente preditores fracos utilizando uma heurística gulosa que corresponde ao método do gradiente em um espaço de função (Dorogush et al., 2018).

O *CatBoost* utiliza o coeficiente de determinação R^2 para calcular o erro resultante da regressão, essa métrica é uma versão normalizada do Erro Quadrático Médio (EQM), apresentado na Eq. 2 (com a_i sendo o valor real da instância i e t_i o valor predito). Uma vantagem do R^2 é não depender da escala dos dados do conjunto. Contudo, no presente trabalho foi utilizado para a avaliação a raiz do EQM (conhecida como *rooted mean square error* em inglês, ou apenas RMSE). A utilização do RMSE garante que os erros não estão enviesados e seguem uma distribuição normal, garantindo assim uma melhor visualização de como o erro está distribuído ao longo da predição (Chai and Draxler, 2014).

$$EQM = \frac{1}{n}\sum_{i=1}^{n}(a_i - t_i)^2 \tag{2}$$

Outra abordagem é a Rede Natural Artificial (RNA), inspirada no aprendizado rea-

lizado pelos neurônios. Uma RNA é composta por um conjunto complexo de estruturas simples, cada estrutura recebe um conjunto de entradas (com valores reais) e entrega apenas uma saída (Haykin, 1999). Um exemplo dessas estruturas é o *Perceptron* (Fig. 3), que recebe um conjunto de números reais e retorna uma saída utilizando uma combinação linear e uma função de ativação, chamada função degrau. A saída gerada pelo *Perceptron* é 1 se a combinação linear for maior do que um determinado limite ou -1 se for menor (Eq. 3, onde w_i é uma constante real chamada peso e σ é uma constante chamada viés) (Mitchell, 1997).

$$-\sigma + \sum_{i=1}^{n} w_i * x_i \tag{3}$$

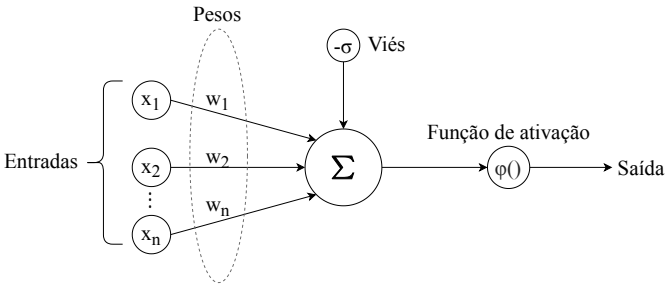


Fig. 3: Esquema do *Perceptron*

A maneira mais básica de aprender os pesos ideais do *Perceptron* é escolher valores aleatórios e iterativamente aplicar o *Perceptron* a cada instância do conjunto de treinamento, modificando os pesos cada vez que a instância não for classificada corretamente. Por ser iterativo, esse processo é repetido para cada instância de treinamento até que todas sejam classificadas corretamente ou até um limite máximo de iterações. Esses pesos são modificados utilizando a regra de treinamento do *Perceptron* (Eq. 4). Essa regra realiza a soma do peso atual com a diferença entre a classificação real da instância (α) e a classificação predita (β), multiplicada por uma taxa de aprendizado (η) e a entrada correspondente. A taxa de aprendizado é responsável por regular o grau de mudança do peso em cada iteração (Mitchell, 1997).

$$\begin{aligned} w_i &\leftarrow w_i + \Delta w_i \\ \Delta w_i &= \eta(\alpha - \beta)x_i \end{aligned} \tag{4}$$

As RNAs podem apresentar diferentes topologias (a forma que a rede está estruturada, como por exemplo, número de nós em uma camada) e arquiteturas (a forma que ocorre o aprendizado da rede) (Mitchell, 1997). O *Perceptron* é uma estrutura muito simples, não sendo capaz de obter um bom resultado em problemas complexos, que caso não sejam linearmente separáveis, a convergência não poderá ser alcançada em grande parte dos casos. Devido a isso uma alternativa é o uso do *Perceptron* multicamadas (*Multilayer Perceptron* ou MLP). O MLP consiste em uma camada de entrada, uma de saída e uma ou mais camadas internas de nós. Os valores da entrada se propagam pela rede em direção à camada de saída, uma arquitetura chamada *feedfoward* (Haykin, 1999).

Para o presente trabalho foram utilizadas três camadas internas com 64, 32 e 256 nós respectivamente. A função de ativação *Unidade Linear Retificada* (ou apenas *ReLu*) foi utilizada nas três camadas internas e na camada de saída uma função sigmoide. Essas parâmetros foram os melhores obtidos utilizando a técnica de busca em grade. Para implementação da rede foi utilizada a biblioteca *Tensorflow* (Abadi et al., 2016) em sua versão para a linguagem *Python*.

A RNA necessita receber valores numéricos normalizados, ou seja, entre 0 e 1. Com isso, o conjunto de dados "saída limitada" foi primeiramente codificado utilizando a função *Ordinal encoder* presente na biblioteca *Scikit-Learn*. Essa função converte os valores categóricos de um atributo para números inteiros (no intervalo $[0, A)$, onde A é o número de categorias presente no atributo).

Após a codificação, o conjunto foi normalizado utilizando a normalização *Min-max*. Sendo x_{min} e x_{max} o menor e o maior valor de um atributo presente no conjunto de dados, a Eq. 5 exibe o cálculo da normalização. Isso foi feito para todos os atributos.

$$x_{normalizado} = \frac{x - x_{min}}{x_{max} - x_{min}} \tag{5}$$

Uma estratégia para verificar o desempenho de um algoritmo de aprendizado de máquina é separar o conjunto de dados em 80% dos dados para treinamento e 20% para teste. Contudo, a cada execução os dados podem estar presentes em partes diferentes da divisão, o que pode ocasionar em resultados diferentes.

Para estimar a performance do algoritmo de forma mais satisfatória foi utilizada a técnica de *k-fold cross validation*. Essa técnica consiste em dividir o conjunto de dados de forma aleatória em k conjuntos (C) mutualmente exclusivos de tamanho similar. O algoritmo de aprendizado de máquina é executado k vezes, onde em cada execução o conjunto $C - C_k$ é utilizado para treinamento e o conjunto C_k para teste. No caso da regressão, a cada iteração é calculado o valor do RMSE da regressão. Ao fim da execução do *cross validation* é calculado um valor médio, esse valor mostra o quão estável é o algoritmo de aprendizado de máquina para o problema tratado (Kohavi et al., 1995).

4 RESULTADOS

4.1 Árvore de regressão

Utilizando a técnica de árvore de regressão, foi realizado um extenso conjunto de testes. Primeiramente, foram utilizados quatro conjuntos de dados: padrão, limitado, uma modificação do conjunto padrão (onde somente o elenco estava dividido e limitado) e o conjunto limitado codificado. Para cada conjunto foram realizados 3 testes, como apresentado abaixo: treinamento da árvore com 80% dos dados do conjunto e teste com 20%, treinamento utilizando o *10-fold cross validation* e utilizando o *50-fold cross validation*. Para os testes utilizando *cross validation* foi calculado, além da média, o desvio padrão e o erro (RMSE) da predição do conjunto de dados ao utilizar a melhor árvore do teste.

O teste utilizando 80% dos dados para treinamento e 20% para teste apresentou RMSE acima de 14 para todos os conjuntos de dados, com o elenco limitado apresentando o menor erro (Fig. 4). Como apenas 20% dos dados são utilizados para teste o erro pode apresentar viés.

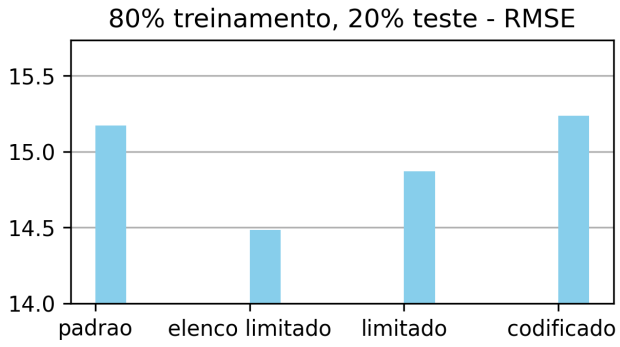


Fig. 4: Resultado utilizando 80% para treinamento e 20% para teste

Na tentativa de obter um resultado sem viés foram realizados testes utilizando o *cross validation* com 10 e 50 *folds* (Tabelas 4 e 5 respectivamente). Em ambos os testes o conjunto de dados limitado obteve o menor erro médio. Contudo, ao utilizar a melhor árvore gerada para realizar a predição de todos os valores de cada, o Elenco limitado foi o que obteve melhor resultado para 10 *folds*, enquanto que para 50 *folds* o melhor foi o Limitado.

Tabela 4: Resultado do 10-Fold Cross Validation

10-Fold Cross Validation – RMSE			
Conjunto de dados	RMSE médio	Desvio Padrão	Predição do conjunto de dados
Padrão	14,6093	0,2162	12,9765
Elenco limitado	14,5218	0,4499	12,5939
Limitado	**14,4866**	0,3240	12,9480
Limitado codificado	14,8488	0,2641	13,8720

Tabela 5: Resultado do 50-fold Cross Validation

50-Fold Cross Validation – RMSE			
Conjunto de dados	RMSE médio	Desvio Padrão	Predição do conjunto de dados
Padrão	14,5054	0,9681	12,5962
Elenco limitado	14,4049	1,0331	12,7446
Limitado	**14,3795**	1,0054	13,0592
Limitado codificado	14,7128	0,9647	12,7640

4.2 Perceptron multicamadas

Ao executar um treinamento com 80% dos dados da "saída limitada codificada" normalizada para treino e 20% para teste, foi obtido um erro de aproximadamente $0,18$. Contudo, assim como a árvore de regressão, o teste pode estar enviesado. Devido a isso foi executado o *cross validation* com 10 e 50 *folds*, cujo resultado pode ser visualizado na Tabela 6.

A Tabela 7 apresenta uma comparação entre o *user score* real e o predito para alguns filmes. Essa comparação foi realizada utilizando os modelos treinados com a técnica de

Tabela 6: Resultado do cross validation aplicado no Perceptron multicamadas

Resultado do Perceptron multicamadas - RMSE (x100)			
K-fold cross validation	RMSE médio	Desvio Padrão	Predição do conjunto de dados
10-fold	17,1854	0,8684	14,2522
50-fold	17,1029	1,2428	15,1512

10-fold cross validation, onde para a árvore de regressão foi utilizado o conjunto de dados limitado e para o MLP o conjunto de dados limitado codificado.

Tabela 7: Comparação entre o score real e o predito para alguns filmes

Filme	Score real	Score Árvore de regressão	Score MLP
Avengers: Infinity War	91	79	90
Star Wars: Episode VII - The Force Awakens	86	85	90
The Last Airbender	30	36	28
Transformers: The Last Knight	44	45	36

5 CONSIDERAÇÕES FINAIS

Em síntese, esse trabalho apresentou a extração dos dados dos filmes presentes no *website Rotten Tomatoes* com o objetivo de predizer a avaliação dos usuários de cada filme utilizando técnicas de aprendizado de máquina. Esses dados foram extraídos utilizando a técnica de *web scraping*, resultando em um conjunto de dados com bastante ruído.

Após a extração dos dados, eles foram tratados. Foram removidos os filmes que possuíam atributos faltantes, os valores que possuíam mais de um elemento por atributo (como, por exemplo, roteirista) foram separados e novos atributos foram criados. Com a criação dos novos atributos, o número de atributos foi limitado para minimizar o número de atributos faltantes. Assim, um dos frutos do presente trabalho é a disponibilidade desse conjunto de dados, disponível para *download* em sua página do *website Kaggle*[5].

Para a predição da avaliação dos usuários de cada filme foram utilizadas duas técnicas de aprendizado de máquina: a árvore de regressão e a rede neural artificial, mais especificamente o *perceptron* multicamadas. Comparando as duas pode-se ver que ambas apresentaram um erro, que apesar de relativamente pequeno, é insatisfatório quando levado em consideração a natureza do problema, ou seja, uma pequena diferença entre o valor real e o valor predito pode significar a perda de uma grande quantia de dinheiro por parte da indústria do cinema. O resultado do *perceptron* multicamadas, ainda apresentou um valor de erro um pouco pior em relação à árvore de regressão. Isso se dá por uma união de diversos fatores como arquitetura da rede, a técnica de codificação e normalização utilizada no conjunto de dados e até a própria escolha pelo MLP em relação a outros tipos de redes neurais artificiais.

As limitações aqui apresentadas abrem espaço para um amplo leque de estudos futuros que vão desde a utilização de outros tipos de redes neurais como a Rede Neural Profunda (Bengio, 2009), outras técnicas como a Máquina de vetores de suporte (Cortes and Vapnik, 1995) e até a implementação de novas técnicas para codificar o conjunto de dados.

[5] `https://www.kaggle.com/rafaelterra/movies-metadata-from-rotten-tomatoes`

6 *Agradecimentos*

Agradeço à Profª Drª Mariza Ferro pelas discussões e aos revisores pelos comentários. O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

REFERÊNCIAS

[1] Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., Devin, M., Ghemawat, S., Irving, G., Isard, M., et al. (2016). "Tensorflow: A system for large-scale machine learning". pages 265–283.

[2] Armstrong, N. and Yoon, K. (1995). "Movie rating prediction". Technical report nº, *Citeseer*.

[3] Bengio, Y. (2009). *Learning deep architectures for AI*. Now Publishers Inc.

[4] Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. A. (1984). *Classification and regression trees*. CRC press.

[5] Chai, T. and Draxler, R. R. (2014). "Root mean square error (rmse) or mean absolute error (mae)?–arguments against avoiding rmse in the literature". *Geoscientific model development*, 7(3):1247–1250.

[6] Clark, J., DeRose, S., et al. (1999). *XML path language (XPath)*. World Wide Web Consortium.

[7] Consortium, W. W. W. et al. (2010). *Xml path language (xpath) 2.0*. World Wide Web Consortium.

[8] Cortes, C. and Vapnik, V. (1995). "Support-vector networks". *Machine learning*, 20(3):273–297.

[9] Dorogush, A. V., Ershov, V., and Gulin, A. (2018). "Catboost: gradient boosting with categorical features support". *arXiv preprint arXiv:1810.11363*.

[10] Haykin, S. (1999). *Neural networks : a comprehensive foundation*. Pearson Education, Delhi.

[11] Heumann, C., Schomaker, M., et al. (2016). *Introduction to statistics and data analysis*. Springer.

[12] Kohavi, R. et al. (1995). "A study of cross-validation and bootstrap for accuracy estimation and model selection". In: *Ijcai*, Montreal, Canada, pages 1137–1145.

[13] Marović, M., Mihoković, M., Mikša, M., Pribil, S., and Tus, A. (2011). "Automatic movie ratings prediction using machine learning". IEEE, pages 1640–1645.

[14] Mitchell, R. (2018). *Web scraping with Python: Collecting more data from the modern web*. "O'Reilly Media, Inc.".

[15] Mitchell, T. (1997). *Machine Learning*. 1 ed.

[16] Sidhu, R. and Prasanna, V. K. (2001). "Fast regular expression matching using fpgas". IEEE, pages 227–238.

[17] Van Rossum, G. and Drake, F. L. (2011). *The python language reference manual*. Network Theory Ltd.

Anais do XIII Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 17 a 21 de Fevereiro de 2020.

Modelagem de um Dataflow para Detecção, Classificação e Predição Temporal de Anomalias

Thiago Moeda[1], Mariza Ferro[1], Eduardo Ogasawara[2] e Fabio Porto[1]

[1] *Laboratório Nacional de Computação Científica (LNCC), Petrópolis/RJ, Brazil*
[2] *CEFET, Rio de Janeiro/RJ, Brazil*

Abstract

In recent years, the classification of time series has gained great relevance in significant sectors and segments of society. Machine Learning Techniques makes it possible to interpret the behavior of anomalous phenomena in multivariate data sets. This work proposes an embryonic study of three methods from the perspective of its ability to provide relevant information for the detection, classification, validation and prediction of anomalous events in time series data. To achieve this objective, a case study was carried out exploring algorithms based on neural networks and inductive symbolic learning applied to a real problem of detecting anomalies associated with the oil well drilling process. The main results indicate that this dataflow can be a promising way to treat anomalies.

Keywords:Unsupervised Learning, Recurrent Neural Network and Self-Organizing Maps.

1 INTRODUÇÃO

O processo de exploração e extração de óleo e gás impõem desafios de diferentes naturezas para os agentes econômicos deste segmento industrial. Durante as operações de perfuração de poços de petróleo, corre-se o risco de a tubulação (coluna) ficar imobilizada dentro do poço, não podendo ser movimentada por razões mecânicas indesejadas ou não controláveis [1]. Este evento é um problema crítico que pode ser ocasionado por diferentes razões como, por exemplo, pressão diferencial, ou seja, quando uma parte da coluna de perfuração fica embutida em uma camada de lama; problemas de instabilidade geológica em poços de perfuração, onde uma porção da abertura do poço não mantém seu tamanho e forma e/ou sua integridade estrutural ou por acúmulo excessivo de cascalho no espaço anular causado pela limpeza inadequada do poço [1]. A detecção prévia destes eventos anômalos chamados de prisão de coluna é extremamente importante em técnicas de perfuração e operação de poços de petróleo já que causa uma parada repentina na produção.

Contato: Moeda T., tmoeda@lncc.br

O objetivo deste trabalho foi revelar a ocorrência destes eventos anômalos, avaliar a qualidade desta detecção e predizer a ocorrência desta anomalia. Para isto, modelamos um processo de fluxo de dados, por meio de métodos supervisionados e não supervisionados de Aprendizado de Máquina (AM), para extrair comportamentos anômalos subjacentes e realizar a previsão de tendências destas anomalias a partir de uma determinada série temporal multivariada. Neste estudo foram utilizados sete conjuntos de dados reais que correspondem a informações que foram registradas pelos sensores acoplados na broca de perfuração dos poços. Cada conjunto de dados possui medições referentes a dez variáveis, sendo todas numéricas de natureza física ou de controle processual. Sendo uma série temporal uma sequência de observações coletadas ao longo do tempo, um evento é definido como uma ocorrência dinâmica, cujo comportamento desperta interesse por variar ao longo do tempo [2]. No entanto, quando os eventos têm durações de tempo (intervalos de estado), as relações se tornam mais complexas [3]. Abordamos o problema utilizando o algoritmo Mapas Auto Organizáveis (SOM) para identificação e rotulagem manual das anomalias já que os conjuntos de dados disponibilizados não continham exemplares classificados correspondentes aos eventos; em seguida, foi implementada uma Árvore de Decisão para atuar como um validador dos rótulos pré-definidos, tornando possível obter um melhor ajuste das instâncias através da redução do erro majoritário; e por fim, treinamos modelos utilizando um algoritmo de aprendizado profundo, Rede Neural Recorrente de Memória de Longo Prazo (LSTM), capaz de aprender a predizer a ocorrência de comportamentos anômalos nas sequências de dados classificados. Os principais resultados indicam que: Em uma abordagem mais ampla, é possível detectar e classificar a ocorrência ou não de um evento anômalo através do processamento SOM; os resultados obtidos através da árvore de decisão mostram uma maneira eficiente e robusta de se obter a validação da classificação, possibilitando assim um processo de ajuste para minimização do erro; os modelos treinados e testados com a rede LSTM, tanto no mesmo conjunto de dados, quanto em outros conjuntos de dados não utilizados no processo de treinamento e teste, obtiveram resultados promissores quanto a predição da ocorrência do evento de prisão de coluna.

2 CONJUNTO DE DADOS

Durante a perfuração de um determinado poço, as informações são registradas pelos sensores acoplados na broca de perfuração e disponibilizadas em tempo real para processamento e armazenamento. Neste trabalho, foram selecionados sete poços, sendo cada poço correspondente a um único conjunto de dados no formato atributo-valor. Para cada conjunto de dados, as medições apresentadas são referentes a 10 variáveis (dimensões), sendo todas contínuas (numéricas) e nenhuma discreta (categórica). A seguir a descrição do dicionário com as 10 dimensões utilizadas:

operation_mode *(-)*: Parâmetro que infere o modo de operação da perfuração; ***bit_depth*** *(m)*: Profundidade da broca; ***weight_on_hook*** *(N)*: Peso no gancho; ***weight_on_bit*** *(N)*: Peso sobre broca; ***standpipe_pressure*** *(Pa)*: Pressão de bombeio do fluido; ***hole_depth*** *(m)*: Profundidade do poço; ***rotary_speed*** *(revolução/s)*: Rotação da coluna de perfuração; ***torque*** *(N.m)*: Torque da coluna; ***block_position*** *(m)*: Altura do bloco; ***fluid_flow*** *(m/s)*: Vazão de entrada na coluna.

A Tabela 1 resume os conjuntos de dados utilizados e suas principais características:

Tabela 1: Informações sobre os conjuntos de dados utilizados

Conjunto de Dados	Total de Instâncias	Número de Atributos (sem classe)	Tempo total de operação (dias)	Profundidade total da perfuração (metros)	Intervalo Absoluto de Perfuração (metros)	Maior tempo parado (horas)
7-10-rjs	54554	10	25.14	1777.37	[3073-4850]	50.70
7-15d-rjs	47494	10	17.55	2045.92	[2810-4856]	273.65
7-48d-rjs	23427	10	9.11	4487.69	[350-4837]	18.87
8-35d-rjs	60985	10	11.16	1914.28	[3494-5409]	26.40
8-37d-rjs	33427	10	68.60	2044.27	[3088-5132]	1379.48
8-38d-rjs	32651	10	4.00	716.12	[2665-3381]	1.30
8-54d-rjs	25442	10	118.82	2988.03	[2761-5749]	2476.36

3 ALGORITMOS

Os algoritmos de Aprendizado de Máquina podem ser definidos como programas de computador que absorvem uma nova experiência em função de alguma classe de tarefas, se o seu desempenho melhora em função de uma determinada medida de performance [4]. A aprendizagem pode ser categorizada como aprender com professor ou aprender sem professor, sendo que a primeira se refere ao aprendizado supervisionado, enquanto que a última pode ser sub-categorizada em aprendizagem não supervisionada e aprendizagem por reforço [5]. Neste trabalho foram implementados três diferentes paradigmas de Aprendizagem de Máquina: O Mapa Auto Organizável (SOM) é um algoritmo não supervisionado que foi elaborado com base no funcionamento dos neurônios do córtex do cérebro humano, onde na prática, o aprendizado absorvido pelo modelo é obtido através de um processo de competição, cooperação e adaptação entre seus neurônios em função de cada instância de dado fornecida ao modelo. Obtém-se como resultado um mapa topológico 2D que proporciona visualmente a correlação multivariada dos dados, através do seu agrupamento ou distanciamento, em função de uma determinada teoria de similaridade [6]. Vale ressaltar que o bom ajuste dos parâmetros para treinamento e calibração dos modelos depende, sobretudo, da análise estatística, mineração dos dados e do entendimento da natureza do domínio. O algoritmo foi implementado com a biblioteca *sompy* [7]. Árvore de Decisão (AD) é definida como um modelo de aprendizado supervisionado indutivo simbólico. A compreensão do resultado da estrutura da árvore possibilita a extração de conhecimento sobre o domínio do problema além da construção de preditores. O algoritmo de AD utilizado neste trabalho é uma versão otimizada do algoritmo *CART* disponibilizada na biblioteca *sklearn* [9], cujo critério de divisão dos nós da árvore visa minimizar a entropia [8], isto é, em AM uma grandeza que mede a desordem na informação. A arquitetura LSTM é um aprimoramento das redes neurais recorrentes (RNN), que são uma classe de redes neurais projetadas para analisar o comportamento das sequências de dados ao longo do tempo, mas possuem um problema de desaparecimento do gradiente. Como afirmado em Hochreiter e Schmidhuber [10], LSTM aborda o problema do desaparecimento do gradiente, incorporando funções (válvulas) em sua dinâmica de estado para

manter ou descartar informações. A formulação original da LTSM apresenta três portões: *input*, *forget* e *output*. O algoritmo foi implementado com a biblioteca *Keras* [11]. Todos os algoritmos foram implementados com a linguagem *python* 3.8.

4 METODOLOGIA

Diferentes aspectos do processamento de dados foram aplicados a cada conjunto de dados para a extração de informações relevantes. Inicialmente, foi realizado um estudo sobre o domínio do problema, possibilitando uma maior compreensão sobre a física do problema e das relações entre os dados. Em seguida, foi feita uma análise para seleção dos conjuntos de dados com as menores porcentagens de valores faltantes em cada dimensão. Na sequência, a etapa de pré-processamento consistiu da normalização e da interpolação dos valores faltantes através do valor médio. Não houve nenhum pré-processamento para remoção de ruído nos dados. A normalização dos dados de entrada para um intervalo entre [0,1] foi obtida através do método MinMaxScaler [12]. Trabalhar com os dados em uma mesma escala é uma boa prática quando se trata de redes neurais. Primeiro porque o algoritmo SOM é baseado em distância e segundo porque acelera o processamento de cálculo do encadeamento da retropropagação dos modelos LSTM. A etapa de modelagem foi dividida em três partes: **Detecção e Classificação**: O processo de detecção compreende a interpretação visual dos resultados obtidos através da modelagem SOM e a análise exploratória dos dados dos atributos. Após este processo, cada conjunto de dados utilizado foi rotulado criteriosamente visando a máxima correlação entre a análise dos resultados provenientes do SOM e da análise do comportamento dos valores dos atributos. **Validação da Classificação**: Nesta etapa, a AD possuiu duas funções especificas: A formação de preditores para extração de conhecimento e também, como uma forma de validar a classificação determinada no processo anterior. **Predição Temporal**: Foram estabelecidos modelos de redes LSTM com o objetivo de gerar tendências de comportamentos anômalos ou não nas sequências de dados.

5 TRABALHOS RELACIONADOS

Diversos estudos utilizaram métodos baseados no aprendizado métrico e no aprendizado profundo para abordar o problema de detecção e classificação de anomalias em séries temporais. Porém, após extensa pesquisa bibliográfica não foram encontrados trabalhos que utilizem essas abordagens para área de óleo e gás e para o problema específico tratado neste artigo. Por exemplo, em [13] os autores utilizaram o algoritmo SOM para identificar os vizinhos mais próximos dos neurônios principais e definiram como um indicador de anomalias o erro mínimo de quantização produzido pelo SOM para cada janela de dados fornecida ao modelo. Seguindo a mesma linha [14] propôs um método baseado em distância para a detecção de anomalias em séries temporal. Um algoritmo baseado em janelas deslizantes que emprega uma estrutura de dados baseada em um ponto central que armazena diferentes relações concêntricas com os pontos vizinhos, que por sua vez, este ponto central pertence a um conjunto maior que é composto por pontos centrais. Este algoritmo tem o foco na redução do espaço de busca de vizinhos através de técnicas de poda multi-distância e na identificação dos *inliers*, que nesta abordagem são pontos que possuem relações com seus vizinhos. Já em [15] os autores propuseram um método de aprendizagem instantânea para detecção de anomalias através da classificação binária

por similaridade, com base no cálculo da distância absoluta entre pares de sequências fornecidas a um modelo de inferência siamesa. Em [16] os autores também utilizaram o aprendizado profundo através de uma rede LSTM para classificação de subtipos da doença de Parkinson, que está associada a diversas manifestações clínicas sendo bastante heterogênea. Sendo assim, um modelo LSTM foi treinado com os prontuários dos pacientes com o objetivo de fornecer representações integradas de sequências multivariadas. Com isso, foi possível utilizá-lo para definir semelhanças entre os pacientes, tornando possível discernir subtipos de progressão da doença. Uma arquitetura que combina duas redes [17] foi proposta para a classificação de séries temporais através da LSTM e da rede convolucional unidimensional CONV1D. Adicionalmente, foi desenvolvido um refinamento para a técnica de aprendizagem por transferência com o objetivo de melhorar o desempenho de um modelo pré-treinado. Um *framework* foi desenvolvido por Salles R. et al. [18] para detecção de eventos. Neste trabalho os autores realizaram combinações de resultados obtidos por diferentes métodos de detecção possibilitando uma melhor compreensão da natureza dos eventos.

6 EXPERIMENTOS

A fim de tornar o *dataflow* o mais genérico possível, todos os modelos foram treinados e aplicados com os mesmos parâmetros de inicialização em todos os sete conjuntos de dados utilizados. Para cada conjunto de dados, os experimentos foram divididos em três etapas: **I**) Detecção e Classificação: Este processo de análise foi subdividido em três partes: **modelagem e interpretação subjetiva dos mapas gerados através do algoritmo SOM**: Cada mapa gerado pelo SOM corresponde ao resultado do processamento de uma determinada janela, intervalo de instâncias, de forma que cada janela subsequente incrementa 500 instâncias. Por exemplo, para o conjunto de dados *8-37d-rjs* que possui 33427 instâncias de dados, foram produzidos 67 mapas de forma que o primeiro mapa representa o resultado do processamento do modelo SOM utilizando as primeiras 500 instâncias de dados, o segundo mapa foi gerado com as instâncias [0, 999] e assim por diante. Os principais parâmetros utilizados para gerar os modelos SOM foram: para cada janela foram gerados 300 hipóteses aleatórias; o tamanho do mapa gerado possui dimensões de 60x90 e isto corresponde a quantidade de neurônios, vetores principais, da malha de discretização; e a função, *kernel*, responsável pelo ajuste dos vetores de referência da vizinhança em relação aos neurônios foi a gaussiana. Dos diversos modelos gerados aleatoriamente para cada janela, foi selecionado o que possuía o menor erro de quantização, ou seja, a média das distâncias entre os entre os neurônios de referência e seus vizinhos. Através da análise visual dos mapas, foi possível detectar de forma segmentada, padrões de comportamento como agrupamentos e dispersões dos dados ao longo do tempo, no espaço topográfico produzido pelo algoritmo SOM. Estes comportamentos foram rotulados manualmente nas instâncias de dados; **análise exploratória dos dados**: Uma vez determinado os intervalos de dados com diferentes padrões de comportamento, foi realizada uma análise conjunta entre os resultados obtidos anteriormente com o SOM e a análise exploratória dos dados dos atributos. Padrões dos quais, alguns, corroboraram com a análise realizada no passo anterior. Tais fatores, ainda que incipientes, contribuíram de maneira substancial no processo de ajuste dos intervalos de instâncias registrados anteriormente; **atualização do conjunto de dados**: Neste processo, os conjuntos de dados foram classificados cri-

teriosamente visando a máxima correlação entre a análise dos resultados provenientes do SOM e da análise exploratória dos dados. Para cada conjunto de dados, as instâncias foram classificadas em duas classes: **(0)**: corresponde a operação normal do equipamento; e **(1)**: correspondente a ocorrência de anomalia. **II**) Validação da Classificação: Com o intuito de validar a precisão da classificação realizada na etapa anterior ou refazê-la, buscando a redução da subjetividade, foi aplicado um algoritmo de árvore de decisão e avaliado o resultado para cada conjunto de dados. Inicialmente, as instâncias dos dados foram distribuídas em 70% utilizados para treinamento do modelo e 30% utilizados para teste. Em seguida, diversos estudos dos parâmetros foram realizados na expectativa de atingir resultados satisfatórios, ou seja, obter uma boa acurácia e, posteriormente, prover modelos que possuam um bom grau de generalização, proporcionando, por sua vez, poder preditivo para a classificação. Os parâmetros utilizados foram determinados baseados em sua capacidade preditiva. Foi feita a análise de densidade e das impurezas em todos os modelos propostos para investigar a normalidade e a ausência de autocorrelação. Após o ajuste, os modelos finais foram utilizados em seus respectivos conjuntos de teste de forma a avaliar sua acurácia preditiva em um conjunto de dados não utilizados no processo de treinamento. Também foi realizada a validação cruzada em k subconjuntos, treinados em *k-1* desses subconjuntos sendo o último subconjunto de dados mantido para teste. Este processo ocorreu 10 vezes com diferentes divisões a cada vez e calculada a média e o desvio padrão. **III**) Predição de Eventos: Um modelo de rede neural LSTM foi treinado, para cada conjunto de dados classificado e validado nas etapas anteriores, com o objetivo de estimar tendências de comportamentos conhecidos e representá-los através da classificação entre [0,1] para um determinado intervalo de instâncias a frente no tempo. Assim como na segunda etapa, as instâncias dos dados também foram distribuídas em 70% utilizados para treinamento e 30% para teste. A arquitetura da rede foi definida com uma única camada interna com 300 blocos LSTM empilhados. Foi aplicado o otimizador *Adam*, o número de iterações de dez épocas e o lote com tamanho 32. A função de perda utilizada foi a *binary_crossentropy*. Foi utilizada uma camada *Embedding* de forma a tornar as sequências de dados contínuas e densas. A fim de reduzir o *overfitting*, foram utilizadas duas camada de *Dropout* com o valor de 0.2. Finalmente, uma camada de saída de um vetor empregando a função *Sigmoid* proporcionando o resultado da predição em um intervalo entre [0, 1].

7 RESULTADOS E DISCUSSÕES

Na Figura 1 são apresentados os resultados do algoritmo SOM, utilizado na primeira etapa para detecção dos eventos nos conjuntos de dados. As regiões com tonalidades claras devem ser interpretadas como separadores de aglomerados. A Figura 1 (a) representa um único mapa gerado a partir de todas as instâncias do conjunto de dados *8-37d-rjs*. Já a Figura 1 (b) representa uma fração de mapas que foram gerados a partir dos dados processados de forma acumulativa em janelas de 500 instâncias utilizando o mesmo conjunto de dados. Nesta figura, os dois primeiros mapas foram interpretados como um estado de operação normal do equipamento, enquanto que do terceiro ao oitavo mapa ocorre a incidência de um evento, na sequência o equipamento volta a operar normalmente conforme os dois últimos mapas. Analisando estas imagens fracionadas em relação a Figura 1 (a), pode-se deduzir que as anomalias se encontram de forma subjacente em relação ao todo.

Portanto, a abordagem de gerar os mapas fracionados e de maneira incremental indica ser um caminho promissor para que se possa detectar e aferir os eventos anômalos. Desta maneira, a classificação das instâncias foi determinada. Na Tabela 2 é apresentado o resultado obtido na primeira etapa, onde cada linha representa a realização do experimento, individualmente, para cada conjunto de dados. Suas colunas correspondem a quantidade total de instâncias, a quantidade de instâncias que foram classificadas como "Normal"ou "Anomalia", o rótulo da classe predominante, a porcentagem e o erro majoritário para aquele conjunto de dados. Pode-se observar o desbalanceamento da classificação nos conjuntos de dados.

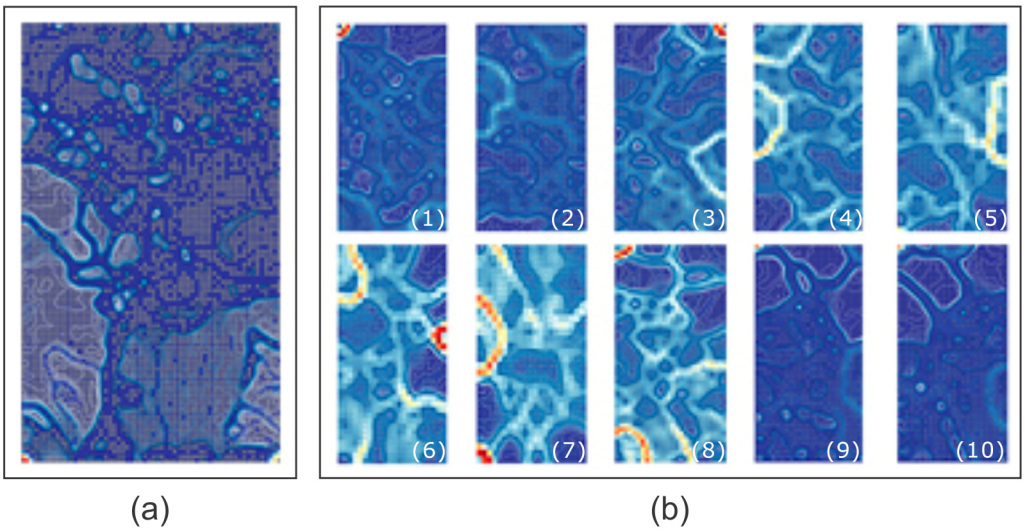


Fig. 1: Resultados do algoritmo SOM aplicado ao conjunto de dados *8-37d-rjs*. (a) Único mapa gerado a partir do processamento de todos os dados. (b) Uma pequena fração dos mapas gerados a partir do processamento de janelas incrementais.

Tabela 2: Resultado da primeira etapa: rotulagem dos conjuntos de dados

Conjunto de Dados	Qtde. Total de Instâncias	Qtde. de Instâncias Rotuladas como Normal	Qtde. de Instâncias Rotuladas como Anomalia	Classe Majoritária (rótulo)	Classe Majoritária (%)	Erro Majoritário (%)
7-10-rjs	54554	31996	22558	Normal	58.65%	41.35%
7-15d-rjs	47494	32992	14502	Normal	69.47%	30.53%
7-48d-rjs	23427	20000	3427	Normal	85.37%	14.63%
8-35d-rjs	60985	19999	40986	Anomalia	67.21%	32.79%
8-37d-rjs	33427	23500	9927	Normal	70.30%	29.70%
8-38d-rjs	32651	9500	23151	Anomalia	70.90%	29.10%
8-54d-rjs	25442	6999	18443	Anomalia	72.49%	27.51%

Já na segunda etapa, foi realizada a análise do grau de generalização do classificador. Esta etapa também foi fundamental na validação da rotulagem que ocorreu na primeira etapa. Os resultados obtidos encontram-se na Tabela 3:

Tabela 3: Resultado da validação da classificação utilizando a árvore de decisão

Conjunto de Dados	Erro (teste)	Erro Aparente	Erro (10CV)	Desvio Padrão
7-10-rjs	0.60%	1.00%	0.00%	0.00%
7-15d-rjs	1.30%	5.20%	6.00%	1.00%
7-48d-rjs	15.80%	5.20%	5.00%	1.00%
8-35d-rjs	0.20%	0.10%	0.00%	0.00%
8-37d-rjs	5.51%	1.80%	2.00%	1.00%
8-38d-rjs	8.12%	5.70%	4.00%	2.00%
8-54d-rjs	0.20%	0.20%	0.00%	0.00%

Para uma análise mais criteriosa da árvore de decisão, é imprescindível que um especialista do domínio participe deste processo. Comumente, esta verificação é realizada a partir da técnica de duplo cego, no qual se descreve atributos presentes na árvore e o especialista emite sua opinião sobre qual classe aquela regra deveria se encaixar [8]. É interessante notar que existe uma clara tendência de que a maior parte dos resultados obtidos estão notadamente em função de um subconjunto específico de atributos (*hole_depth*, *bit_depth*, *rotary_speed*, *weight_on_hook* e *standpipe_pressure*). Finalmente, na Tabela 4 estão os resultados da acurácia dos modelos que correspondem a terceira etapa. O modelo da rede LSTM foi treinado com o objetivo de estimar a classe de um determinado intervalo de instâncias no futuro. Basicamente, fornece como saída a informação sobre o estado do processo de operação do equipamento, se será normal ou anômalo, com uma determinada acurácia para um intervalo de instâncias a frente, a partir de uma sequência multivariada de instâncias fornecidas como entrada ao modelo treinado. Os resultados foram obtidos a partir dos conjuntos de dados classificados e validados nas etapas anteriores.

Tabela 4: Resultado da classificação utilizando a rede neural LSTM

Conjunto de Dados	Erro (teste)	Erro Aparente
7-10-rjs	23.41%	23.42%
7-15d-rjs	24.36%	24.76%
7-48d-rjs	14.20%	14.2%
8-35d-rjs	33.01%	32.74%
8-37d-rjs	29.48%	7.14%
8-38d-rjs	28.35%	28.36%
8-54d-rjs	27.21%	27.68%

A Tabela 5 exibe a comparação entre os desempenhos das previsões. Cada modelo treinado com um determinado conjunto de dados foi aplicado aos demais conjuntos de dados. A primeira coluna corresponde aos modelos gerados a partir do conjunto de dados entre parênteses.

Tabela 5: Resultado do erro da aplicação dos modelos LSTMs

Modelos / Dados	7-10-rjs	7-15d-rjs	7-48d-rjs	8-35d-rjs	8-37d-rjs	8-38d-rjs	8-54d-rjs
M(7-10-rjs)	-	30.64%	**29.08%**	65.95%	30.48%	70.53%	64.01%
M(7-15d-rjs)	40.97%	-	**14.68%**	66.94%	**29.81%**	70.53%	72.36%
M(7-48d-rjs)	40.97%	30.65%	-	66.94%	**29.81%**	70.53%	72.36%
M(8-35d-rjs)	59.03%	69.35%	85.32%	-	70.19%	**29.47%**	**27.64%**
M(8-37d-rjs)	40.97%	40.97%	**14.68%**	66.94%	-	70.53%	72.36%
M(8-38d-rjs)	59.03%	69.35%	85.32%	33.06%	70.19%	-	**27.64%**
M(8-54d-rjs)	59.03%	69.35%	85.32%	33.06%	70.19%	**29.47%**	-

Os Melhores resultados, com o menor erro, destacados em negrito, foram obtidos pelos modelos **M(7-15d-rjs)** e **M(8-37d-rjs)** quando aplicados ao conjunto de dados *7-48d-rjs*. Diversos fatores contribuíram para a heterogeneidade dos resultados, como os ajustes dos parâmetros de inicialização dos modelos, a utilização da mesma topologia de rede LSTM para treinamento de todos os modelos e a rotulagem visual realizada na primeira etapa através do SOM.

8 CONCLUSÃO E TRABALHOS FUTUROS

Neste trabalho foram aplicados métodos de Aprendizagem de Máquina na abordagem do problema de prisão de coluna, que ocorre durante o processo de perfuração de poços de petróleo. Foram utilizadas metodologias supervisionadas e não supervisionadas através dos algoritmos Mapas Auto Organizáveis (SOM), Árvore de Decisão e da Rede Neural Recorrente de Memória de Longo Prazo (LSTM). Técnicas de mineração de dados foram elaboradas na medida em que o entendimento da física do problema era adquirido, contribuindo para aumentar o nível de resolução na classificação das anomalias. Quanto ao processo de predição da classificação através da rede neural LSTM foram obtidos resultados favoráveis. Contudo, a automatização, refinamento dos modelos, avaliação e a minimização do erro de classificação ficam como desafios a serem desenvolvidos. Este foi um estudo inicial que se baseou na formulação de uma teoria para tratamento de anomalias através da integração de diferentes paradigmas de Aprendizado de Máquina cujo foco foi a obtenção de resultados promissores quanto a classificação de séries temporais.

9 *Reconhecimentos*

Agradeço à Profa. Dra. Mariza Ferro que leciona a disciplina de Aprendizado de Máquina de maneira brilhante. Gostaria de agradecer também ao Prof. Dr. Fabio Porto e ao Prof. Dr. Eduardo Ogasawara por me orientarem ao longo desta jornada.

Referências

[1] Robert Mitchell and Society of Petroleum Engineers. (2006). “Petroleum engineering handbook”. Vol. II. English. OCLC: 254332812. Richardson, Tex: SPE, 2006. ISBN: 978-1-55563-129-1.

[2] Guralnik V; Srivastava J. (1999). "Event Detection from Time Series Data". Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '99. pp. 33–42. ISBN : 1-58113-143-7. DOI: 10.1145/312129.312190.

[3] I. Batal et al. (2012). "Mining Recent Temporal Patterns for Event Detection in Multivariate Time Series Data". Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining - KDD '12. DOI: 10.1145/2339530.2339578.

[4] Mitchell, T. (1997). "Machine Learning", pp 2. ISBN: 0070428077.

[5] Haykin, S. (2008). "Neural networks and learning machines - 3rd ed."pp 34-37. ISBN-13: 978-0-13-147139-9.

[6] Kohonen et al. (2001). "Self-Organizing Maps". Springer Berlin Heidelberg.

[7] SOMPY: numpy based SOM (Self Organizing Map) Library. https://github.com/ttlg/sompy (acesso em 06/02/2021).

[8] Ferro, M; Huei, L; Sandro, C. (2002). "Intelligent Data Analysis: A Case Study of the Diagnostic Sperm Processing". CSITeA'02 - ACIS International Conference.

[9] Scikit-learn: Machine Learning in Python. A decision tree classifier. https://scikit-learn.org/stable/modules/generated/sklearn.tree.DecisionTreeClassifier.html (acesso em 06/02/2021).

[10] Hochreiter, S; Schmidhuber, J. (1997). "Long Short-Term Memory". Neural Computation 9(8):1735-1780.

[11] Keras: Deep Learning API in Python. LSTM class. https://keras.io/api/layers/recurrent_layers/lstm/ (acesso em 06/02/2021).

[12] MinMaxScaler. https://scikit-learn.org/stable/modules/generated/ sklearn.preprocessing.MinMaxScaler.html (acesso em 06/02/2021).

[13] Tian J. et al. (2014). "Anomaly Detection Using Self-Organizing Maps-Based K-Nearest Neighbor Algorithm". European Conference of the Prognostics and Health Management Society.

[14] Tran L. et al. (2021). "Real-Time Distance-Based Outlier Detection in Data Streams". PVLDB.

[15] Ullah A. et al. (2020). "One-Shot Learning for Surveillance Anomaly Recognition using Siamese 3D CNN". IEEE.

[16] Zhang X et al. (2019). "Data-Driven Subtyping of Parkinson Disease Using Longitudinal Clinical Records: A Cohort Study". Nature.

[17] Karim F et al. (2018). "LSTM Fully Convolutional Networks for Time Series Classification ". IEEE.

[18] Salles R. et al. (2020). "Harbinger: Um framework para integração e análise de métodos de detecção de eventos em séries temporais". SBBD - Brazilian Symposium on Databases.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Identificação e análise de potenciais alvos moleculares para doenças negligenciadas causadas por tripanossomatídeos

Victória Cruz de Barros[1], Caroline Leles Amaral[1] e Gregório Kappaun Rocha[1]

[1] *Instituto Federal Fluminense (IFF), Macaé/RJ, Brazil*

Abstract

Neglected Diseases portrays a group of endemic tropical diseases that affect approximately one billion people worldwide. These diseases mainly affect populations in underdeveloped regions and do not arouse the interest of the pharmaceutical industry. The process of discovering new drugs has been benefiting by computational biology tools, which stand out for their low cost, the speed in carrying out the tests and the expressive results recently achieved. The present work seeks to identify and analyze potential molecular targets for the development of chemotherapeutic agents for neglected diseases caused by trypanosomatids (in this case, Chagas Disease and Leishmaniasis). Ten potential targets present in different metabolic pathways were identified. So far, the identification and structural comparison of ten targets present in trypanosomatids and also in different pathogens of the human species has been carried out, seeking to indicate paths for future studies of drug repositioning. Identifying common and similar targets among different pathogens is interesting, as it allows the same drug to be used to treat different diseases. TM-Score values were low among all the related enzymes that were studied, however, a more refined comparative study is needed, mainly with a focus on the residues of the active site.

Keywords: Doenças Negligenciadas, Tripanossomatídeos, Alvos Moleculares, Reposicionamento, Fármacos.

1 INTRODUÇÃO

Doenças negligenciadas representam um conjunto de doenças infecciosas tropicais e endêmicas que afetam majoritariamente as populações mais pobres e vulneráveis do planeta [15]. A denominação "negligenciada" foi adotada devido ao investimento precário em pesquisa e desenvolvimento de novos medicamentos, testes diagnósticos, vacinas e outras tecnologias para prevenção e controle [28, 5]. Doenças negligenciadas afetam cerca de

Contato: Gregório Kappaun Rocha, D.Sc., gregorio.rocha@iff.edu.br

um bilhão de pessoas no mundo [15], contribuindo para a perpetuação dos ciclos de pobreza e desigualdade social, como consequência de seu impacto na saúde, na redução da produtividade da população e na promoção do estigma social [28]. É pertinente ressaltar que, frente à omissão de empresas privadas, cabe ao Estado financiar iniciativas que visem o melhor entendimento dessas doenças, com enfoque na prevenção, tratamento e cura.

Grande parte destas doenças continuam sem tratamento ou com tratamentos pouco eficazes e/ou com efeitos colaterais severos. Um projeto recente, chamado G-finder [4], que rastreia o investimento anual na área de pesquisa e desenvolvimento acerca da saúde, revelou que menos de 5.0% dos recursos do financiamento mundial de inovação para doenças negligenciadas foram investidos no grupo das doenças extremamente negligenciadas, ou seja, doença do sono, leishmaniose visceral e doença de Chagas, embora mais de 500 milhões de pessoas sejam ameaçadas por estas três doenças parasitárias [26].

Motivado por esta grande ausência de investimentos, o presente trabalho foca em duas destas doenças extremamente negligenciadas, que são causadas por tripanossomatídeos: a Doença de Chagas e a Leishmaniose. A Doença de Chagas é causada pelo protozoário *Trypanossoma cruzi* e, no mundo, cerca de sete milhões de pessoas estão infectadas com a doença [9]. A transmissão do patógeno pode ocorrer pelas vias vetorial, oral, congênita ou sanguínea. No Brasil, a principal forma de contaminação é através da via oral, por meio do consumo de açaí ou cana-de-açúcar contaminados por serem triturados junto ao inseto vetor, popularmente conhecido como barbeiro [8]. As leishmanioses são causadas por parasitas do gênero *Leishmania* e afetam 1,8 milhão de pessoas por ano no mundo e 25 mil no Brasil [28]. As leishmanias podem ser transmitidas através da picada de mosquitos hematófagos e constituem um complexo de enfermidades causadas por diferentes espécies morfologicamente semelhantes, sendo diferenciadas apenas por métodos bioquímicos, imunológicos ou mesmo patológicos [26]. Estima-se que o Brasil foi o responsável por 70% das mortes no mundo por Doenças de Chagas em 2017 e 98% dos casos de leishmaniose visceral do continente [14].

A Biologia Computacional refere-se ao emprego de ferramentas computacionais no estudo de problemas biológicos e vem sendo empregada com sucesso em aplicações relacionadas à saúde humana, desde a interpretação de dados genômico até o planejamento de fármacos [20]. Estudos *in silico* podem ser bastante úteis no estudo de doenças negligenciadas, pois permitem a realização de pesquisas de forma rápida, barata e eficiente.

A identificação de candidatos a alvos moleculares representa um passo crucial para o desenvolvimento de agentes quimioterápicos que possam levar à cura ou reduzir os sintomas de uma doença [11]. Além disso, identificar alvos comuns e similares entre diferentes patógenos é interessante pois permite que uma mesma droga possa ser usada para o tratamento de diferentes doenças e, ainda, encurtar o tempo de desenvolvimento do medicamento [17, 10].

Provocar a morte do parasita, bloquear sua ação patogênica, impedir sua replicação são algumas das possibilidades de ação de um candidato a fármaco. Tal objetivo pode ser alcançado por meio da inibição de uma enzima-chave de uma determinada via metabólica, por exemplo [27]. Faz-se necessário, então, investigar as diferenças entre as vias e as enzimas presentes no hospedeiro e no parasita, buscando prever possíveis efeitos colaterais [13]. Nesse sentido, ferramentas computacionais de análise de sequências [1] e de comparação estrutural [29] são fundamentais para realizar a exploração quantitativa

e analisar as diferenças entre o metabolismo do hospedeiro e do parasita, possibilitando, assim, a seleção de um melhor alvo [27].

Desta forma, o projeto tem como objetivo aplicar estratégias de biologia computacional no estudo de doenças negligenciadas no Brasil. Busca-se, especificamente, identificar potenciais alvos moleculares para o desenvolvimento de agentes quimioterápicos para doenças causadas por tripanossomatídeos (no caso, Doença de Chagas e Leishmaniose) através da revisão de literatura e analisar tais alvos com foco na similaridade de sequência e estrutural entre diferentes espécies.

2 METODOLOGIA

Uma revisão da literatura foi realizada para buscar alvos moleculares candidatos para o tratamento da Doença de Chagas e da Leishmaniose. Dez alvos moleculares, identificados como promissores em outros estudos [24, 23, 13] foram selecionados para investigação neste trabalho (Tabela 1).

Tabela 1: Potenciais alvos moleculares para o desenvolvimento de agentes quimioterápicos selecionados a partir da revisão de literatura.

Enzima-alvo	PDB ID	Espécie de referência
Arginase (ARGI)	4ITY	*Leishamania mexicana*
Cruzaína (CRUZ)	1EWP	*Trypanossoma cruzi*
Di-hidrofolato redutase (DHFR)	3QFX	*Trypanosoma brucei*
Gliceraldeido-3-fosfato desidrogenase (GA3P)	1I32	*Leishmania mexicana*
Glicerol-3-fosfato desidrogenase (GL3P)	1EVY	*Leishamania mexicana*
Hipoxantina-guanina fosforibosiltransferase (HGFT)	5JV5	*Trypanosoma brucei*
Ornitina descarboxilase (ORND)	1QU4	*Trypanosoma brucei*
Superóxido dismutase (SODM)	4F2N	*Leishmania major*
Topoisomerase I (TOPI)	2B9S	*Leishmania donovani*
Tripanotiona redutase (TRED)	2JK6	*Leishmania infantum*

Dentre as enzimas selecionadas, encontram-se duas enzimas presentes na via glicolítica, rota metabólica essencial para o suprimento de energia (ATP - adenosina trifosfato) ao parasita. As enzimas da via glicolítica dos tripanossomatídeos eleitas foram: a gliceraldeido-3-fosfato desidrogenase e a glicerol-3-fosfato desidrogenase. A gliceraldeido-3-fosfato desidrogenase é um alvo de suma importância, pois é responsável por catalisar a fosforilação do substrato gliceraldeído-3-fosfato em 1,3-bifosfoglicerato na presença de NAD+ e fosfato inorgânico. O 1,3-bifosfoglicerato é altamente energético, permitindo que haja produção de ATP. Por conseguinte, a inibição dessa enzima reduz a oferta de energia do parasita, ocasionando sua morte. Entretanto, devido a esta via também estar presente nos seres humanos, pode ocasionar graves efeitos colaterais caso a enzima do parasita seja similar à enzima do ser humano. A outra enzima selecionada desta via, a glicerol-3-fosfato desidrogenase, catalisa a oxidação de glicerol-3-fosfato em 1,3-bifosfofoglicerato, sendo também

importante para o aporte de ATP nestes parasitas.

As enzimas da biossíntese de poliaminas são consideradas, também, potenciais alvos moleculares. As poliaminas são moléculas que atuam regulando o crescimento e diferenciação celular [13]. A espermidina é um composto de poliamina encontrados em ribossomos e tecidos vivos, apresentando várias funções metabólicas dentro dos organismos, inclusive de empacotamento do DNA [13]. Os tripanossomatídeos dependem da espermidina para o crescimento e sobrevivência [13]. Dentre as diversas enzimas presentes nessa via, destaca-se a ornitina descarboxilase, a arginase e a tripanotiona redutase.

Ademais, a via dos folatos é de suma importância para os tripanossomatídeos [13]. A enzima di-hidrofolato redutase faz a redução do ácido dihidrofólico a ácido tetrahidrofólico. A inibição desta via impede a formação do tetra-hidrofolato, que é essencial para a síntese das bases nitrogenadas, sendo que a deficiência desses compostos purínicos e pirimidínicos pode levar à inibição da síntese de DNA, RNA e proteínas [13].

Observou-se que, além da di-hidrofolato redutase, a hipoxantina-guanina fosforibosiltransferase também é uma enzima essencial para síntese de bases nitrogenadas, sendo responsável pela conversão de bases purínicas a ribonucleotídeos [13]. Sendo assim, os tripanossomatídeos são totalmente dependentes desta enzima para recuperação das purinas durante seu ciclo de vida.

Outrossim, a enzima superóxido dismutase catalisa a transformação do radical superóxido em oxigênio molecular e peróxido de hidrogênio. A inibição desta enzima provoca um estresse oxidativo no parasita pelo acúmulo do íon superóxido [13].

Outra enzima candidada é a topoisomerase I, que pode ser encontrada em procariotos e eucariotos. A enzima possui a função de relaxar o DNA superenrolado para remover restrições helicoidais que podem impedir a replicação e a transcrição do DNA [12].

Por fim, a enzima cruzaína, que é fundamental para a nutrição e desenvolvimento do parasita, para a evasão do sistema imune e invasão celular do hospedeiro, também foi selecionada para o estudo [24].

Para cada um dos alvos selecionados foi obtida a sequência de aminoácidos dispostas no *Protein Data Bank* (PDB - www.rcsb.org) [2] (Tabela 1).

O BLASTp (*Basic Local Alignment Search Tool for Proteins*) [1] foi utilizado para a busca de sequências relacionadas às enzimas-alvo. Através dos métodos de alinhamento é possível obter informações a respeito da relação evolutiva entre sequências de diferentes espécies [27]. Três filtros de identidade de sequência (30%, 50% e 70%) foram adotados para comparação no banco de dados de proteínas do NCBI (www.ncbi.nlm.nih.gov/protein/). Esta etapa tem a finalidade de comparar sequências de aminoácidos, por intermédio do alinhamento local e, assim, encontrar proteínas similares em espécies distintas. Desta forma, caso as proteínas possuam um alto nível de similaridade, o resultado poderá ser utilizado para verificações da possibilidade de um mesmo fármaco atuar no tratamento de mais de uma doença.

Embora exista uma regra comum de que duas sequências são homólogas se forem mais de 30% idênticas em todo o seu comprimento, tal critério pode implicar na perda ou na inclusão de sequências como homólogas, e não deve ser visto como uma faixa limítrofe inquestionável [16]. Avaliar a presença de sequências relacionadas em faixas acima deste valor, aumenta a garantia da sequência encontrada ser evolutivamente relacionada.

Espécies que apresentam enzimas com percentual de identidade maior que 30% aos

alvos selecionadas e que causam patologias na espécie humana foram destacadas para os estudos de comparação estrutural.

O TM-Score [29] foi usado como métrica de comparação estrutural entre as enzimas selecionadas de diferentes espécies.

3 RESULTADOS E DISCUSSÃO

Observa-se, na Tabela 2, por intermédio do alinhamento, o número de espécies distintas que possuem sequências de enzimas similares com as dos alvos moleculares iniciais. Verificou-se, para a maior parte dos alvos, que ao utilizar o filtro de 70% a quantidade de espécies que apresentam enzimas relacionadas foi reduzida. Dentre os resultados obtidos, observou-se que a enzima cruzaína atua exclusivamente em espécies do gênero *Trypanosoma* [24]. Por este motivo, este alvo está sendo amplamente estudado por atuar no tratamento da Doença de Chagas.

Tabela 2: Número de espécies que apresentam sequências similares à enzima-alvo. Três filtros de identidade de sequência foram adotados para comparação no banco de dados de proteínas do NCBI.

Enzima-alvo	Filtro 30%	Filtro 50%	Filtro 70%
Arginase	100	72	19
Cruzaína	8	8	8
Di-hidrofolato redutase	54	21	3
Gliceraldeído-3-fosfato desidrogenase	50	50	50
Glicerol-3-fosfato desidrogenase	112	29	6
Hipoxantina-guanina fosforibosiltransferase	71	17	3
Ornitina descarboxilase	109	109	8
Superóxido dismutase	89	81	9
Topoisomerase I	76	74	26
Tripanotiona redutase	41	41	16

Observou-se para alguns outros alvos (e.g., Arginase, Topoisomerase I, Gliceraldeído-3-fosfato desidrogenase, Tripanotiona redutase) a presença de enzimas com alta identidade de sequência presentes em organismos de gêneros e espécies distintas (Tabela 2). Este resultado indica que estes são bons alvos, pois enzimas que se repetem com alta similaridade em várias espécies apresentam um papel altamente conservado evolutivamente. Ademais, enzimas com alta similaridade e recorrentes entre espécies distintas são potenciais alvos para o desenvolvimento de fármacos, visto que o mesmo medicamento poderia ser utilizado para o tratamento de inúmeras doenças negligenciadas.

Outrossim, é imperioso salientar que as espécies mais recorrentes entre os alvos foram: *Trypanosoma cruzi*, *Leishmania mexicana*, *Leishmania major*, *Trypanosoma brucei* e *Leishmania infantum*.

Dentre as espécies encontradas através do alinhamento de sequências, selecionou-se aquelas que apresentam enzimas com percentual de identidade maior que 30% aos alvos

selecionadas e que causam patologias na espécie humana (Tabela 3). Essas enzimas foram destacadas para os estudos de comparação estrutural usando como métrica de comparação o TM-Score (Tabela 4). As enzimas-alvos superóxido dismutase, ornitina descarboxilase, topoisomerase I, arginase e cruzaína não apresentaram enzimas relacionadas presentes em espécies causadoras de patologias humanas dentro do filtro utilizado.

Observa-se, na Tabela 4, a comparação estrutural entre as enzimas-alvo e as enzimas das espécies selecionadas que afetam os seres humanos. Dessa maneira, é possível verificar se há possibilidade de um medicamento que iniba mais de uma das enzimas supracitadas. Os valores do TM-Score foram baixos para todas as enzimas estudadas, indicando baixa similaridade estrutural. Entretanto, é necessário um estudo comparativo mais refinado entre as enzimas, principalmente com foco nos resíduos componentes do sítio ativo.

Tabela 3: Espécies selecionadas por conter enzimas com alta similaridade ao alvo e por causar patologias na espécie humana.

Espécie	Cobertura	*E-value*	Identidade (%)	PDB ID
HIPOXANTINA-GUANINA FOSFORIBOSILTRANSFERASE - 5JV5				
Bacillus anthracis	83%	2e-36	40.00%	6D9Q
Vibrio cholerae	78%	9e-32	38.24%	3OHP
DI-HIDROFOLATO REDUTASE - 3QFX				
Candida glabrata	58%	8e-24	40.71%	3CSE
Mycobacterium tuberculosis	58%	2e-17	39.29%	1DF7
TRIPANOTIONA REDUTASE - 2JK6				
Bartonella henselae	95%	5e-99	38.65%	3O0H
GLICEROL-3-FOSFATO DESIDROGENASE - 1EVY				
Coxiella burnetti	88%	4e-50	32.72%	3K96
GLICERALDEÍDO-3-FOSFATO DESIDROGENASE - 1I32				
Naegleria fowleri	96%	4e-128	56.29%	5UR0

Cobertura: o quanto as sequências foram sobrepostas; *E-value*: probabilidade do alinhamento ter sido ao acaso; Identidade (%): porcentagem referente ao quanto as sequências são idênticas.

A seguir, segue uma breve descrição das patologias causadas pelas espécies relacionadas identificadas no estudo e que podem ser usadas como partida para estudos de reposicionamento de fármacos. A infecção causada pelo protozoário *Naegleria fowleri*, popularmente conhecida como "ameba comedora de cérebros", progride rapidamente, podendo levar a óbito em um período de 10 dias [18]. A *Vibrio cholerae* é a bactéria responsável pela cólera, uma doença bacteriana infecciosa intestinal aguda, podendo ser transmitida por contaminação fecal-oral direta ou pela ingestão de água ou alimentos contaminados. Caso não seja tratado rapidamente, pode ocorrer graves complicações e até mesmo óbito. Os membros do gênero *Candida* são os fungos mais frequentemente recuperados da infecção fúngica humana [22]. Com o crescente aumento do número de pessoas imunodeficientes, inferiu-se que a *Candida glabrata* é um fungo agressivo, extremamente oportunista e resistente que ataca principalmente os indivíduos com comorbidades [19]. A bactéria conhecida como *Mycobacterium tuberculosis* é responsável por causar a tuberculose [3].

A *Bacillus anthracis* é a causadora da moléstia conhecida como antraz. Consiste em infecções, sendo três formas que acometem os humanos: cutânea, gastrintestinal e pulmonar; esta última apresenta maior índice de mortalidade [25, 21]. A *Bartonella henselae* é uma proteobacteria que é o agente causador da doença por arranhões em gatos, a chamada angiomatose bacilar. Os gatos são reservatórios importantes desses microrganismos [7]. A febre Q, causada pela *Coxiella burnetii*, é uma zoonose de ampla distribuição mundial, apesar de pouco relatada no Brasil. A *C. Burnetii* geralmente está presente na urina e fezes de animais infectados. Nos casos onde a doença acaba evoluindo de forma crônica, a endocardite é a manifestação mais frequente [6].

Tabela 4: Comparação estrutural (via TM-score) entre as enzimas selecionadas nas espécies causadoras de outras patologias em humanos e a enzima-alvo.

Enzimas	Espécie	TM-Score
Gliceraldeido-3-fosfato desidrogenase	*Naegleria fowleri*	0.2177
Hipoxantina-guanina fosforibosiltransferase	*Baccilus anthracis*	0.1890
Hipoxantina-guanina fosforibosiltransferase	*Vibrio cholerae*	0.1814
Di-hidrofolato redutase	*Candida glabrata*	0.1795
Di-hidrofolato redutase	*Mycobacterium tuberculosis*	0.1880
Tripanotiona redutase	*Bartonella henselae*	0.2429
Glicerol-3-fosfato desidrogenase	*Coxiella burnetti*	0.1908

4 CONCLUSÃO

Estudos de biologia computacional, por também representarem uma alternativa mais rápida e acessível economicamente, vem contribuindo para a identificação de novos alvos enzimáticos, para o estudo molecular de tais estruturas e para o desenvolvimento de novos agentes quimioterápicos, sendo uma importante arma para o enfrentamento às doenças negligenciadas. O investimento em novas formas de prevenção, tratamento e cura para doenças negligenciadas no Brasil e no mundo continua sendo fundamental para a redução dos ciclos de pobreza e desigualdade social, bem como para evitar a morte de milhares de pessoas por ano.

No presente estudo, a partir de duas doenças negligenciadas iniciais - a Doença de Chagas e a Leishmaniose - selecionou-se dez enzimas indicadas como potenciais alvos moleculares. Foi realizado, até o momento, a identificação e a comparação estrutural de alvos similares presentes em diferentes patógenos da espécie humana, o que pode indicar caminhos para futuros estudos de reposicionamento de fármacos. Esta estratégia (o reposicionamento) de busca de fármacos a partir de medicamentos já aprovados é bastante utilizada e implica em menor custo e tempo de desenvolvimento.

5 *Agradecimentos*

Agradecemos ao suporte financeiro da FAPERJ, através do programa Jovens Talentos, e ao IFFluminense *Campus* Macaé pelo suporte.

Referências

[1] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman. Basic local alignment search tool. *Journal of molecular biology*, 215(3):403–410, 1990.

[2] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000.

[3] E. P. Bethlem. Manifestações clínicas da tuberculose pleural, ganglionar, geniturinária e do sistema nervoso central. *Pulmão RJ*, 21(1):19–22, 2012.

[4] N. Chapman, L. Abela-Oversteegan, V. Chowdhary, A. Doubell, U. Gurjav, and M. Ong. *Neglected Disease Research & Development: A Pivotal Moment for Global Health*. Health Policy Division, The George Institute for International Health, 2016.

[5] M. da Saúde. Doenças negligenciadas: estratégias do ministério da saúde. *Rev Saúde Pública*, 44(1):200–2, 2010.

[6] I. A. d. M. Damasceno and R. C. Guerra. Coxiella burnetii e a febre q no brasil, uma questão de saúde pública. *Ciência & Saúde Coletiva*, 23:4231–4239, 2018.

[7] G. F. de Souza. Doença da arranhadura do gato: relato de caso. *Rev Med Minas Gerais*, 21(1):75–78, 2011.

[8] J. C. P. Dias, V. Amato Neto, and E. J. d. A. Luna. Mecanismos alternativos de transmissão do trypanosoma cruzi no brasil e sugestões para sua prevenção. *Revista da Sociedade Brasileira de Medicina Tropical*, 44(3):375–379, 2011.

[9] R. Exame. A "silenciosa" doença de chagas faz 7 milhões de vítimas no mundo. *Exame: Ciência. Disponível em https://exame.com/ciencia/a-silenciosa-doenca-de-chagas-faz-7-milhoes-de-vitimas-no-mundo/. Acesso em: 17 de Outubro de 2020*, 1, 2020.

[10] R. J. M. d. Fonseca et al. *Estudo de reposicionamento de fármacos para doenças negligenciadas causadas por protozoários através da integração de bases de dados biológicas usando Web Semântica*. PhD thesis, Instituto Oswaldo Cruz, 2013.

[11] R. V. Guido, A. D. Andricopulo, and G. Oliva. Planejamento de fármacos, biotecnologia e química medicinal: aplicações em doenças infecciosas. *Estudos avançados*, 24(70):81–98, 2010.

[12] M. Li and Y. Liu. Topoisomerase i in human disease pathogenesis and treatments. *Genomics, proteomics & bioinformatics*, 14(3):166–171, 2016.

[13] J. L. Melos and A. Echevarria. Sistemas enzimáticos de tripanossomatídeos como potenciais alvos quimioterápicos. *Revista Virtual de Quimíca*, 4(4):374–392, 2012.

[14] L. Mori. As doenças negligenciadas pela indústria farmacêutica que afetam milhões de pessoas no mundo e no brasil. *BBC News Brasil em São Paulo. Disponível em https://www.bbc.com/portuguese/geral-46961306. Acesso em: 17 de Outubro de 2020*, 1, 2019.

[15] W. H. Organization et al. *Neglected tropical diseases, hidden successes, emerging opportunities*. Number WHO/HTM/NTD/2009.2. World Health Organization, 2009.

[16] W. R. Pearson. An introduction to sequence similarity ("homology") searching. *Current protocols in bioinformatics*, 42(1):3–1, 2013.

[17] S. Pushpakom, F. Iorio, P. A. Eyers, K. J. Escott, S. Hopper, A. Wells, A. Doig, T. Guilliams, J. Latimer, C. McNamee, et al. Drug repurposing: progress, challenges and recommendations. *Nature reviews Drug discovery*, 18(1):41–58, 2019.

[18] A. A. Queiroz. *Meningoencefalite Amebiana Primária causada por Naegleria fowleri*. PhD thesis, Faculdade Método de São Paulo, 2016.

[19] C. F. Rodrigues, M. E. Rodrigues, S. Silva, and M. Henriques. Candida glabrata biofilms: how far have we come? *Journal of fungi*, 3(1):11, 2017.

[20] G. D. A. Scheila de Avila e Silva, Daniel Luis Notari. Bioinformática contexto computacional e aplicações. 2020.

[21] L. J. d. Silva. Guerra biológica, bioterrorismo e saúde pública, 2001.

[22] S. Silva, M. Negri, M. Henriques, R. Oliveira, D. W. Williams, and J. Azeredo. Candida glabrata, candida parapsilosis and candida tropicalis: biology, epidemiology, pathogenicity and antifungal resistance. *FEMS microbiology reviews*, 36(2):288–305, 2012.

[23] R. J. Soares-Bezerra, L. Leon, and M. Genestra. Recentes avanços da quimioterapia das leishmanioses: moléculas intracelulares como alvo de fármacos. *Revista Brasileira de Ciências Farmacêuticas*, 40(2):139–149, 2004.

[24] M. L. d. Souza. *Identificação de novos inibidores da enzima cruzaína de Trypanosoma cruzi candidatos a fármacos contra a doença de Chagas*. PhD thesis, Universidade de São Paulo, 2012.

[25] R. C. Spencer. Bacillus anthracis. *Journal of clinical pathology*, 56(3):182–187, 2003.

[26] R. Valverde. Agência fiocruz de notícias. *Doenças Negligenciadas. Disponível em https://agencia.fiocruz.br/print/4740. Acesso em: 17 de Outubro de 2020*, 1, 2013.

[27] H. Verli. Bioinformática: da biologia à flexibilidade molecular. 2014.

[28] G. L. Werneck, M. H. Hasselmann, and T. G. Gouvêa. Panorama dos estudos sobre nutrição e doenças negligenciadas no brasil. *Ciência & Saúde Coletiva*, 16:39–62, 2011.

[29] Y. Zhang and J. Skolnick. Scoring function for automated assessment of protein structure template quality. *Proteins: Structure, Function, and Bioinformatics*, 57(4):702–710, 2004.

Anais do XIV Encontro Acadêmico de Modelagem Computacional.
Laboratório Nacional de Computação Científica, 08 a 11 de Fevereiro de 2021.

Identifying strong gravitational lenses using deep learning techniques

Viviane M. Matioli[1], Rafael S. Pereira[1] e Fabio Porto[1]

[1] *Laboratorio Nacional de Computação Cientifica, Petrópolis/RJ, Brazil*

Abstract

The production of astronomical data is expected to increase dramatically with large telescopes and surveys planned for the next decade. It is estimated that such projects will observe a number of strong gravitational lens candidates greater than the volume currently known by possibly three orders of magnitude, in particular galaxy-scale lenses. Such lens systems are traditionally identified through methods involving visual inspection of images, which would not be feasible in this near future scenario. Thus, different attempts have been made recently to develop more efficient and automated methods for the task, as we also seek to do in this paper. For this purpose we use deep learning techniques, more specifically siamese neural networks, training the method on simulated examples and evaluating its performance on real and simulated images. Results obtained in some scenarios indicate the possibility of its use as an initial candidate filtering step, eliminating half of the false candidates, while correctly classifying most lens examples.

Keywords: Deep Learning, Strong Gravitational Lensing, Artificial Neural Networks

1 INTRODUCTION

Gravitational lenses are phenomena observed when the light from a distant source has its path deviated due to the curvature of space-time caused by the presence of another massive body between source and observer. Strong lenses are cases in which the two objects are sufficiently aligned in order to lead to the formation of multiple images associated with the same source. In these cases we usually have a galaxy or quasar as the source, while the deflecting object is another galaxy or cluster.

Strong lens systems are extremely useful tools, with many astrophysical and cosmological applications. They can be used to measure the Hubble constant and other cosmological parameters [25, 28, 31], to determine the mass of objects acting as deflectors and study

Contato: Viviane Matioli, vivianemm@lncc.br

the distribution of dark matter in galaxies and clusters [18, 30, 32, 11]. Moreover, they can function as natural telescopes, magnifying high redshift sources that would not be detected without the lens effect [2, 26, 19, 10].

Despite their importance, the number of known lens candidates today is relatively small, in the order of a few thousand systems [22]. The traditional identification process involves visual inspection of a large volume of images selected according to parameters such as brightness or color. However, the increasing volume of astronomical data being produced makes evident the need for more efficient and automated lens finding methods. It is estimated that projects planned for the near future, such as the Legacy Survey of Space and Time (LSST), the Euclid telescope and Square Kilometer Array (SKA) should observe around 10^5 strong lenses [7, 20].

In this scenario, several methods have been developed recently in attempts to automate the search for lenses in astronomical surveys, many of them using machine learning techniques [24, 3, 16, 14, 23]. After training, such methods are able to quickly classify a large number of images in a short period of time, which makes them more suitable for the task. Following this line of research, we seek to develop an automated method to assist in the strong lens finding problem using deep learning techniques, more specifically siamese neural networks, together with the K-Nearest Neighbors algorithm (KNN).

In this paper we describe the development of such method. In Section 2 we briefly describe training and test data sets and the tools used to develop the method. In Section 3 we present the results of different experiments performed using real and simulated images. In Section 4 we conclude with a quick recapitulation of results, summarizing our conclusions and future possibilities for further improvement of the method.

2 METHODOLOGY

2.1 Model

Siamese neural networks were introduced in the 1990s to deal with the signature verification problem [4], and have since been employed in different tasks, such as dimensionality reduction [13], face recognition and verification [6, 29, 15] and image, patch and point descriptors [5, 33, 27]. They typically consist of two identical subnets with shared weights which accept different input examples and act as feature extractors. The twin networks are then joined at their outputs by a layer which computes a distance metric between the embeddings. The main goal is to map data into a lower dimensional embedding space where examples belonging to the same class are closer to each other than to examples of different classes.

This type of deep siamese structure presented better results than traditional convolutional neural networks (CNNs) in our experiments in the strong lens finding problem. Therefore, the method described in this work is based on a siamese neural network and the KNN algorithm, consisting of a binary classifier, which after trained, classifies input images as lenses or non-lenses.

More precisely, we use the network architecture presented in Figure 1, with ReLU activation function in all intermediate layers, sigmoid function in the output layer, and binary cross entropy loss function. The KNN algorithm is trained on the embeddings generated by the network, classifying new images according to their distance to training

examples.

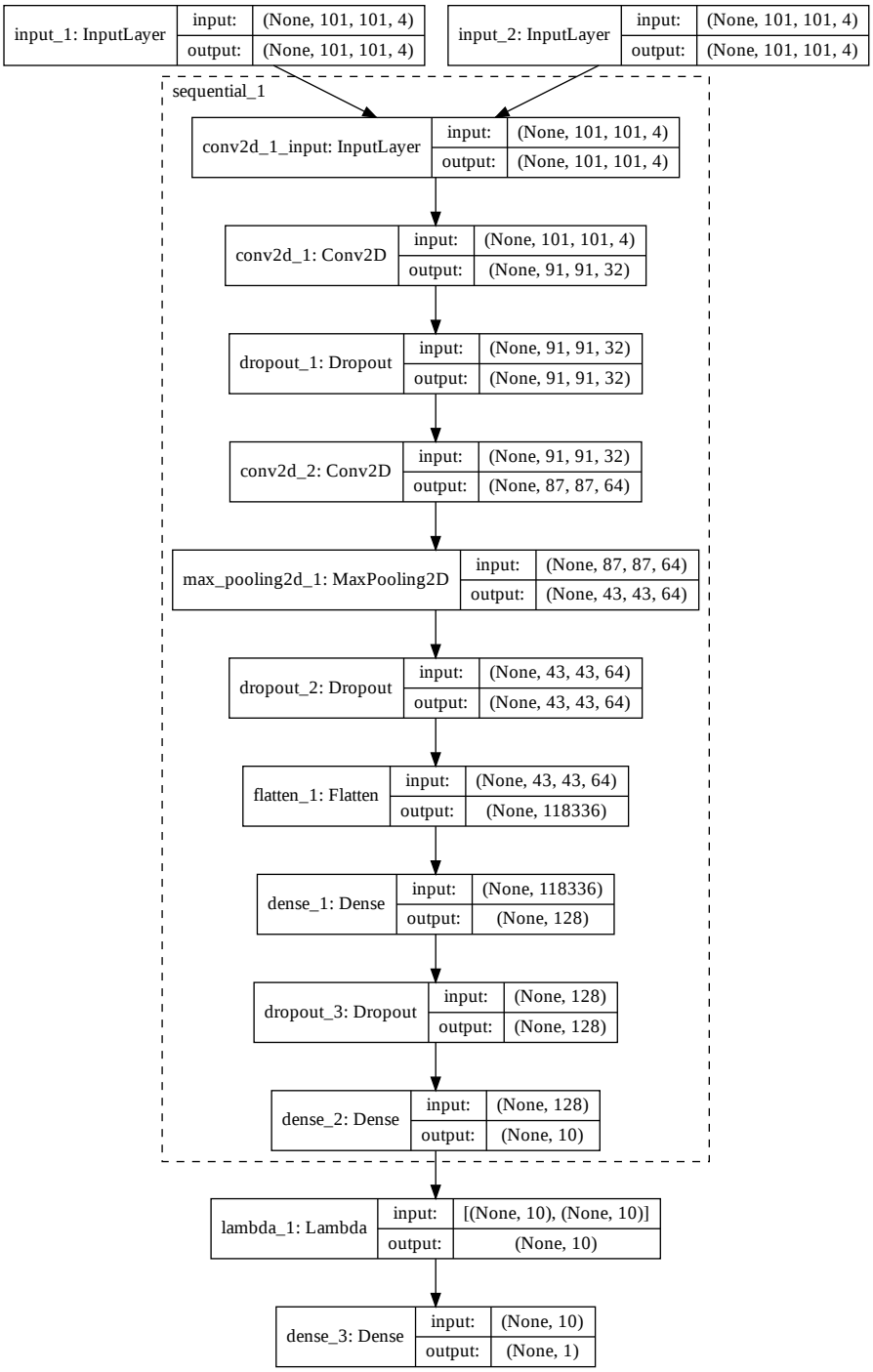


Fig. 1: Siamese network architecture

2.2 Data

The number of real lenses known today is relatively low, as mentioned in section 1, and presents heterogeneous conditions, since the candidates come from different surveys, were observed in different filters and identified through various projects. Therefore, in this paper we use simulated training images, available in greater quantity, as has been done in other recent papers.

The simulated data set used consists of images of galaxy-galaxy lenses and non-lenses, part of the first Bologna Strong Gravitational Lens Finding Challenge [21, 22]. We used approximately 17000 simulated objects, each of them having images in four filters, simulating the bands u, g, r and i of the Kilo Degree Survey (KiDS) [9].

In this project we choose to make use of publicly available simulated images in order to verify how well the method performs without the cost necessary to produce a simulation of our own. Nevertheless, the development of an optimized simulation is of interest and can be performed in the future.

Thus, some factors which may have an impact on the results must be taken into account, such as the difference in quality and noise levels between the training simulation and real test images, in addition to their filters, available in different quantities and frequencies of the spectrum. Due to these differences, the distribution learned by the network during training does not correspond exactly to the real distribution, but can be considered an approximation.

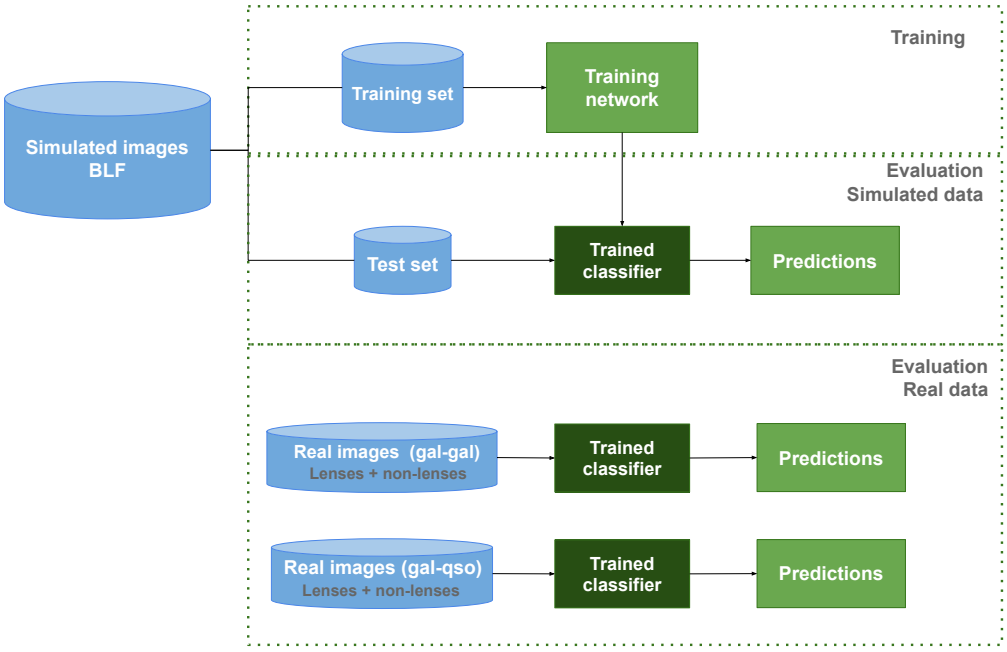


Fig. 2: Flowchart indicating different sets used for testing. The trained classifier consists of the siamese network together with the KNN algorithm.

In order to evaluate the performance of the method, the following data sets were also used: real images of galaxy-galaxy lenses from different surveys, gathered in the *Master*

Lens Database [1]; real images of galaxy-quasar lenses observed in the CASTLES survey, carried out with the Hubble Space Telescope (HST) [8, 12]; and real images of galaxies as non-lens examples, also observed with the HST. The training and testing steps using these different sets are illustrated in Figure 2.

3 RESULTS AND DISCUSSION

Evaluating the model's performance classifying simulated images, we obtain the confusion matrix shown in Figure 3, and the probability distribution histogram of classifications in Figure 4.

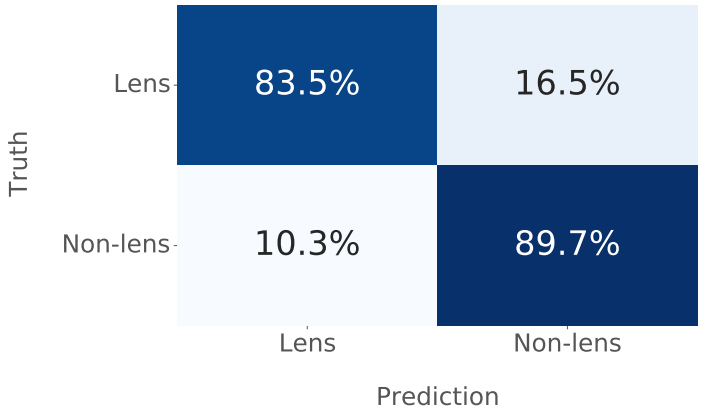


Fig. 3: Confusion matrix - Simulated galaxy-galaxy lenses

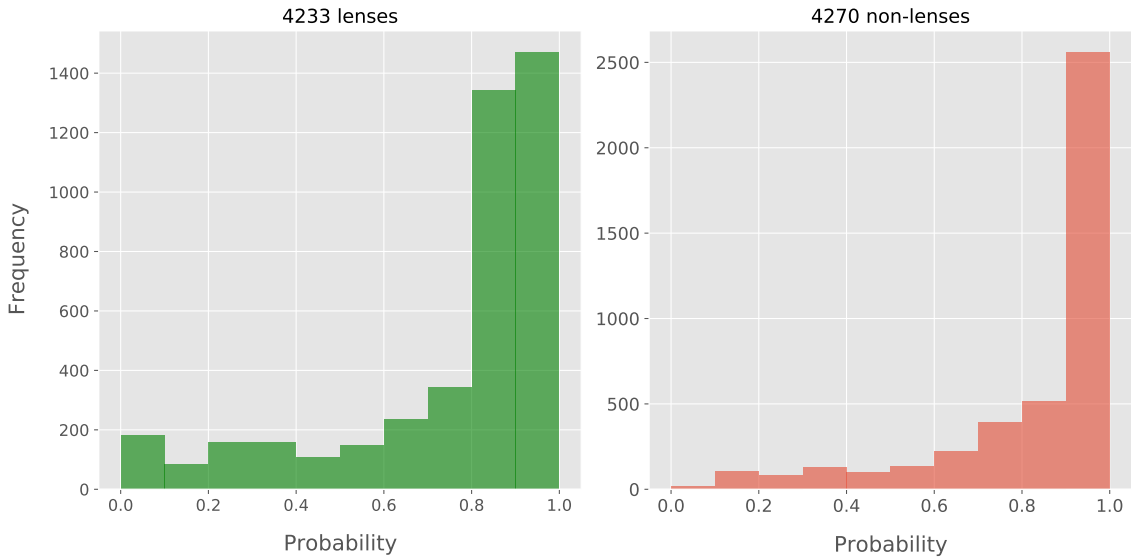


Fig. 4: Probability distribution - Simulated galaxy-galaxy lenses

According to the results in Figure 3, we observe that 87% of all classifications are correct (accuracy). The network identifies 83% of the test lenses (recall), over 60% of them with probability higher than 80%. The probability histograms also exhibit configurations

close to ideal, indicating that most examples are correctly classified with high probability, while the number of classifications with lower probabilities quickly decays. The metrics are presented in Table 1.

3.1 Evaluation on real images

Before evaluating the method on real images, it was necessary to perform an initial pre-processing step. We removed texts and symbols located over the images, reduced them all to the same size and normalized them. Moreover, they were also organized in such a way that observations of an object in different filters of the telescope correspond to different channels.

After this process, during the evaluation phase, the recall metric represents the proportion of true lenses correctly identified, while accuracy, precision and F1 also depend on the method's performance when classifying non-lenses. As our method was developed to function as an initial step, filtering candidates, we are primarily interested in its performance in the lens class, where high recall solutions make sure that real candidates are not discarded in the process.

3.1.1 Galaxy-Galaxy Lenses

Evaluating the method on images of real galaxy-galaxy lenses, the same type as those used for training, we obtain the confusion matrix shown in Figure 5 and probability distribution in Figure 6.

Although this is a very heterogeneous set, containing lens images from different surveys, we verify that over 95 % of them are classified correctly, most with high probability, as seen in the histogram. The evaluation metrics are presented in Table 1.

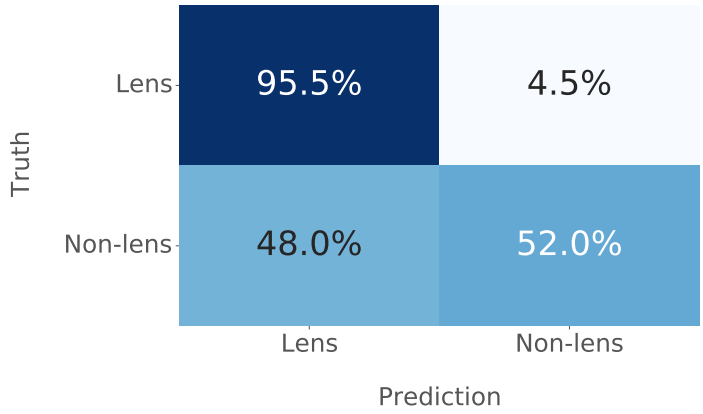


Fig. 5: Confusion matrix - Real galaxy-galaxy lenses

As for the classification of real non-lenses, the model does not perform as well, misclassifying nearly half of them, while correct classifications present probabilities lower than 70%.

As also verified before in other papers [17], factors such as the morphological classification of galaxies used to generate the simulation and those included in the test set can influence the results obtained. In general, simulations only contain elliptical galaxies,

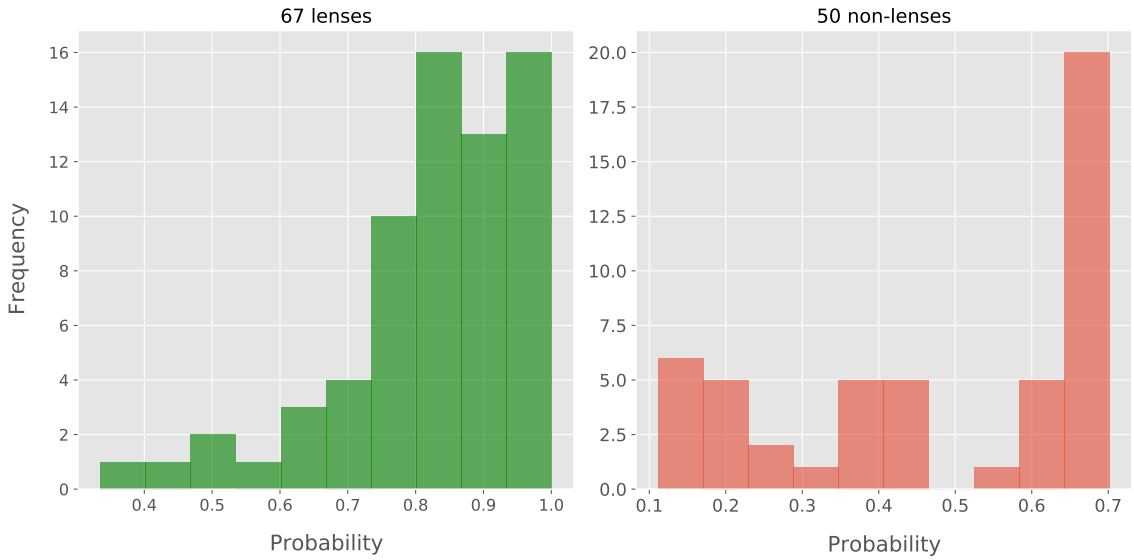


Fig. 6: Probability distribution - Real galaxy-galaxy lenses

while our test data set is also comprised of spiral galaxies, for instance, which tend to be mistaken for lenses very often by this kind of method. Possible solutions for this problem would be the production of a simulation of our own containing different types of non-lens galaxies, or the use of real galaxy images together with the simulation for training.

3.1.2 Galaxy-quasar lenses

Evaluating the performance of the method classifying real images of galaxy-quasar lenses from the CASTLES survey, we obtain the confusion matrix shown in Figure 7 and the probability distribution of classifications in Figure 8.

We observe that all real lenses are correctly classified, with probability higher than 90%, despite being of a different type than those used for training, in addition to being observed in different filters. Table 1 shows the evaluation metrics obtained.

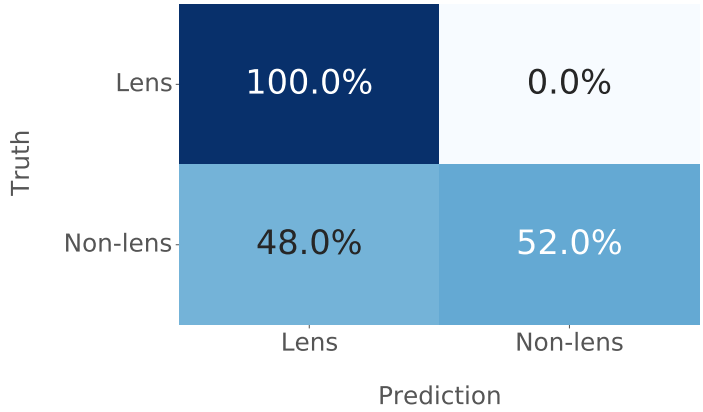


Fig. 7: Confusion matrix - Real galaxy-quasar lenses

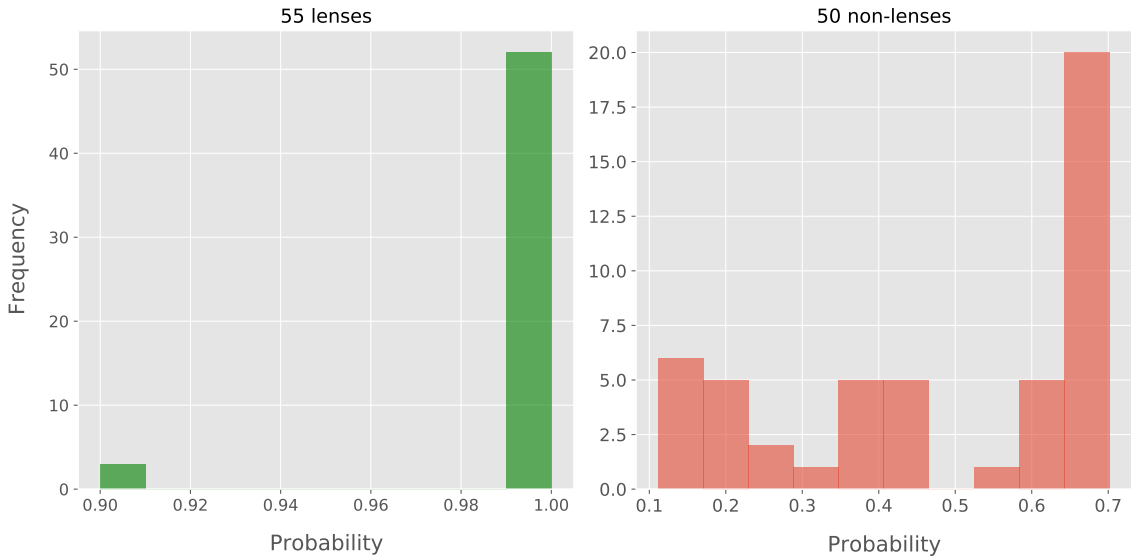


Fig. 8: Probability distribution - Real galaxy-quasar lenses

Tabela 1: Evaluation metrics

Metrics	Simulation	Real (gal-gal)	Real (gal-qso)
Accuracy	0.87	0.77	0.77
Precision	0.89	0.73	0.70
Recall	0.83	0.95	1.0
F1	0.86	0.82	0.82

4 CONCLUSIONS

In the course of this work we seek to develop an automated method to assist in the identification of strong gravitational lenses, selecting possible candidates. After experimenting with different architectures, we decided on a method consisting of a siamese neural network together with the KNN algorithm, which presented better results than traditional CNNs when classifying input images as lenses and non-lenses.

The method is trained on simulated images containing galaxy-galaxy lens and non-lens examples. The evaluation is performed on simulated images and real images of both galaxy-galaxy and galaxy-quasar lenses. In these three cases, the method is able to identify 83%, 95% and 100% of lenses, respectively.

Even though the method does not show the same generalization power for the classification of real non-lenses, it eliminates half of them, maintaining good performance when classifying lenses, which is the class of interest. The method could thus be used as an initial step on a set of lens candidate images, reducing it by half, without eliminating a large number of true lenses.

This method also presents several possibilities for further development in order to reduce the number of false positives, for instance. It is important to perform a closer

investigation of the effect of different objects used as non-lens examples for tests and how each type is classified. We could then use real training examples, together with the simulation, and even develop a simulation of our own, containing different types of non-lens examples. It would also be interesting to analyze the effect of a preprocessing step in real test images, such as the one performed in the CASTLES survey. Finally, this initial method can be integrated with another lens searching algorithm and applied to real surveys.

5 *Acknowledgements*

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.

REFERENCES

[1] Master Lens Database. `http://masterlens.astro.utah.edu`, 2020. Access: 2 oct. 2020.

[2] Bellagamba. Zooming into the cosmic horseshoe: new insights on the lens profile and the source shape. *Monthly Notices of the Royal Astronomical Society*, page stw2726, 2016.

[3] Bom. A neural network gravitational arc finder based on the mediatrix filamentation method. *Astronomy & Astrophysics*, 597:A135, 2017.

[4] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, and R. Shah. Signature verification using a "siamese" time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(04):669–688, 1993.

[5] N. Carlevaris-Bianco and R. M. Eustice. Learning visual feature descriptors for dynamic lighting conditions. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2769–2776. IEEE, 2014.

[6] S. Chopra, R. Hadsell, and Y. LeCun. Learning a similarity metric discriminatively, with application to face verification. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 539–546. IEEE, 2005.

[7] Collett. The population of galaxy–galaxy strong lenses in forthcoming optical imaging surveys. *The Astrophysical Journal*, 811(1):20, 2015.

[8] C.S. Kochanek. Castles survey. `https://www.cfa.harvard.edu/castles/`. Access: 1 oct. 2020.

[9] de Jong, Jelte TA and Kleijn, Gijs A Verdoes and Kuijken, Konrad H and Valentijn, Edwin A and others. The kilo-degree survey. *Experimental Astronomy*, 35(1-2):25–44, 2013.

[10] Deane. The preferentially magnified active nucleus in iras f10214+ 4724–iii. vlbi observations of the radio core. *Monthly Notices of the Royal Astronomical Society*, 434(4):3322–3336, 2013.

[11] Dye. Decomposition of the visible and dark matter in the einstein ring 0047–2808 by semilinear inversion. *The Astrophysical Journal*, 623(1):31, 2005.

[12] Falco. The castles gravitational lensing tool. *Gravitational Lensing: Recent Progress and Future Go*, 237:25, 2001.

[13] R. Hadsell, S. Chopra, and Y. LeCun. Dimensionality reduction by learning an invariant mapping. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pages 1735–1742. IEEE, 2006.

[14] Hartley. Support vector machine classification of strong gravitational lenses. *Monthly Notices of the Royal Astronomical Society*, 471(3):3378–3397, 2017.

[15] J. Hu, J. Lu, and Y.-P. Tan. Discriminative deep metric learning for face verification in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1875–1882, 2014.

[16] Jacobs. Finding strong lenses in cfhtls using convolutional neural networks. *Monthly Notices of the Royal Astronomical Society*, 471(1):167–181, 2017.

[17] C. Jacobs, T. Collett, K. Glazebrook, C. McCarthy, A. Qin, T. Abbott, F. Abdalla, J. Annis, S. Avila, K. Bechtol, et al. Finding high-redshift strong lenses in des using convolutional neural networks. *Monthly Notices of the Royal Astronomical Society*, 484(4):5330–5349, 2019.

[18] Kochanek. The implications of lenses for galaxy structure. *The Astrophysical Journal*, 373:354–368, 1991.

[19] Marshall. Superresolving distant galaxies with gravitational telescopes: Keck laser guide star adaptive optics and hubble space telescope imaging of the lens system sdss j0737+ 3216. *The Astrophysical Journal*, 671(2):1196, 2007.

[20] McKean. Strong gravitational lensing with the ska. *arXiv preprint arXiv:1502.03362*, 2015.

[21] Metcalf. Gravitational lens finding challenge. `http://metcalf1.difa.unibo.it/blf-portal/gg_challenge.html`, 2018. Access: 2 oct. 2020.

[22] Metcalf. The strong gravitational lens finding challenge. *Astronomy & Astrophysics*, 625:A119, 2019.

[23] Ostrovski. Vdes j2325- 5229 az= 2.7 gravitationally lensed quasar discovered using morphology-independent supervised machine learning. *Monthly Notices of the Royal Astronomical Society*, 465(4):4325–4334, 2017.

[24] Petrillo. Finding strong gravitational lenses in the kilo degree survey with convolutional neural networks. *Monthly Notices of the Royal Astronomical Society*, 472(1):1129–1150, 2017.

[25] Refsdal. On the possibility of determining hubble's parameter and the masses of galaxies from the gravitational lens effect. *Monthly Notices of the Royal Astronomical Society*, 128(4):307–310, 1964.

[26] Shu. The boss emission-line lens survey. iv. smooth lens models for the bells gallery sample. *The Astrophysical Journal*, 833(2):264, 2016.

[27] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer. Discriminative learning of deep convolutional feature point descriptors. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 118–126, 2015.

[28] Suyu. Two accurate time-delay distances from strong lensing: Implications for cosmology. *The Astrophysical Journal*, 766(2):70, 2013.

[29] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[30] Tessore. Lensed: a code for the forward reconstruction of lenses and sources from strong lensing observations. *Monthly Notices of the Royal Astronomical Society*, 463(3):3115–3128, 2016.

[31] Treu. Time delay cosmography. *The Astronomy and Astrophysics Review*, 24(1):11, 2016.

[32] Vegetti. Bayesian strong gravitational-lens modelling on adaptive grids: objective detection of mass substructure in galaxies. *Monthly Notices of the Royal Astronomical Society*, 392(3):945–963, 2009.

[33] S. Zagoruyko and N. Komodakis. Learning to compare image patches via convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4353–4361, 2015.