Fractional order differential calculus applied on decision making system to smart grid management via decision trees

Cálculo diferencial de ordem fracionária aplicado em sistema de tomada de decisão para gerenciamento de rede inteligente via árvores de decisão

Cálculo diferencial de orden fraccional aplicado en el sistema de toma de decisiones a la gestión de redes inteligentes a través de árboles de decisión

Heictor Alves de Oliveira Costa
ORCID: https://orcid.org/0000-0003-3611-3675
Estacio de Sa University, Brazil
E-mail: heictor8@gmail.com
Larissa Luz Gomes
ORCID: https://orcid.org/0000-0002-9086-2821
Estacio de Sa University, Brazil
E-mail: larissa.gomes@estacio.br
Denis Carlos Lima Costa
ORCID: https://orcid.org/0000-0003-3207-6934
Federal Institute of Education, Science and Technology of Pará, Brazil
E-mail: denis.costa@ifpa.edu.br
Erick Melo Rocha
ORCID: https://orcid.org/0000-0002-5039-9967
Estacio de Sa University, Brazil
E-mail: Erick.rocha@estacio.br
Carlos Renato Francês
ORCID: https://orcid.org/0000-0003-0305-7662
Federal University of Pará, Brazil
E-mail: rfrances@ufpa.br
Sergio Henrique Andrade
ORCID: https://orcid.org/0000-0003-2587-4516
Estacio de Sa University, Brazil
E-mail: Andrade.Sergio@estacio.br

Abstract
This article portrays the relationship between fractional order differential calculus and the computational intelligence method, applying it to the improvement of intelligent systems. The Kirchhoff Laws, represented by second order differential equations, were solved via non-integer order differential calculus. The results obtained were used in the implementation of decision trees, which allowed the decision rules to be incorporated into the controllers. The results obtained by mathematical modeling did magnify the information extracted from Kirchhoff's Laws. Due to the gain magnitude of this information, the decision trees were obtained with greater precision and accuracy. In this way, it was achieved to build a hybrid system capable of being used in the development of controllers automata that has the lower response time and highest efficiency.
Keywords: Decision tree; Fractional order differential calculus; Smart grid; Kirchhoff Laws.

Resumo
O presente artigo retrata a relação entre o cálculo diferencial de ordem fracionária e o método de inteligência computacional, aplicando-a no aprimoramento de sistemas inteligentes. As Leis Kirchhoff, representadas por equações diferenciais de segunda ordem, foram solucionadas via cálculo diferencial de ordem não-inteira. Os resultados obtidos foram utilizados na implementação de inúmeras árvores de decisão, que por sua vez subsidiaram regras de decisão incorporadas aos controladores. Os resultados obtidos pela modelagem matemática magnificaram as informações extraídas das Leis de Kirchhoff. Devido a magnitude dessas informações, obteve-se árvores de decisão com maior precisão e acurácia. Dessa forma, foi alcançado um sistema híbrido capaz de ser utilizado no desenvolvimento de controladores autômatos que apresentarão menor tempo de resposta e maior eficiência.
Palavras-chave: Árvore de decisão; Cálculo diferencial de ordem fracionária; Rede inteligente; Leis Kirchhoff.

Resumen

Este artículo describe la relación entre el cálculo diferencial de orden fraccionario y el método de inteligencia computacional, aplicándolo a la no mejora de los sistemas inteligentes. Las leyes de Kirchhoff, representadas por ecuaciones diferenciales de segundo orden, se resolvieron mediante cálculo diferencial de orden no entero. Los resultados obtenidos se utilizaron en la implementación de árboles de decisión, que a su vez apoyaron las reglas de decisión incorporadas en los impulsores. Los resultados obtenidos mediante modelos matemáticos magnificaron la información extraída de las Leyes de Kirchhoff. La magnitud de esta información obtenida, los árboles de decisión se obtuvieron con mayor precisión y exactitud. De esta manera, se logró un sistema híbrido capaz de ser utilizado en el desarrollo de controladores de autómatas que tiene el menor tiempo de respuesta y la mayor eficiencia.

Palabras clave: Árbol de decisión; Cálculo diferencial de orden fraccionario; Red inteligente; Leyes de Kirchhoff.

1. Introduction

In the past 20 years, humanity's care about Ambiental impacts gained even more weight on decision making, mainly because of extreme climatic changes and its impact on nature, economy, and life quality of humans. Unfortunately, this caution about the consequence of humanity's impact took much time do bring practical solutions, and because of that, society began living a hard time with resources production and distribution.

The resource scarcity got accentuated over the last 2 years, mainly because of covid-19 pandemic, being one of the most affected areas the technological industry, as seen in Baraniuk (2021) and Wilmot. (2021) with the chips shortage, affecting from the smartphones production to the new smart automobiles manufacturing. Another sector struggling because of the pandemic and lack of previous prepare is the energy section, both power generation and power distribution, as shown in Hoskins. (2021) with some provinces in China going through a power shortage.

In Brazil, there's currently a similar problem of power generation happening because of a drastic reduction in rainfall over the last year as seen in Mota (2021), and as most of Brazil's electrical power is produced by hydroelectric systems, the country began an energy rationing and increasing the usage of thermoelectric systems, causing an increase on population's power bill.

The pandemic also brought a new era based on home office to reduce infection rates, and as this experience allowed some professional to keep on working from home, workers started to improve their own domestic offices for productivity and comfort. Nonetheless, the need of caring about the energetical consumption got even higher because of world's situation described previously on this paper, therefore this new generation of home officers lack a trustful, efficient, and simple implementation system capable of managing their inhouse office room energetical usage.

Managing inhouse power grid allows not only energy saving, but also Ambiental and living comfort for the residents, (Hagenbeck et al. 2016) presented a smart home control system to save energy using Fuzzy Techniques to perform automatic decision making, however its implementation requires some advanced analysis by individual sensors in the smart home, and manual defined Fuzzy Rules to operate the system based on conditionals. This paper, analogously, proposes a method that applies automation since the development of the conditional rules, by utilizing computational intelligence to trace the behavior patterns and return the different scenarios and later run the real time decision making.

In (Costa H. A. O. et al. 2021a), the authors apply a hybrid model using Genetic Algorithm (GA) and Newton's method to optimize the energy distribution and transmission over a power grid, possible thanks to the mathematical-computational modeling of the circuit. On (Costa H. A. O. et al. 2021b), it's presented a composed model by the union of GA and Particle Swarm Optimization (PSO), this model allows quickly finding great solutions with optimal values for better operating the components in the 14-bus electrical system.

Because of modern devices and various activities over the day, the home office turns the domestic office circuit in a dynamic power grid, therefore the scenarios vary according to the combinations of active devices. This dynamic profile demands a control system appropriate to not only classify the scenarios but also to perform the decision making while the control system

is online.

For this paper the authors chose to operate the circuit via a microcontroller device responsible, because it's affordable device and simply operated via multiple easy interfaces, so it fits greatly for the necessary activities. The device will operate based on an algorithm developed to work with multiple conditions described based on the multiple home circuits behaviors.

The electrical phenomena, described by the Kirchhoff Laws, were implemented by way of the Non-Integer Order Differential Equations solving results. Such strategy allowed to extract unhackneyed information about the behavior of RCL autonomous electrical circuits. It's considered that, the energy storers capacitors and inductors, such as the circuits in which they are included, do not depend only on their power sources, but also on the voltages and initial charge from the capacitors, and the current and initial flows from the inductors, since the differential equations the describe those circuits are obtained adopting the relation voltage-current from the storer devices.

In the analysis of characteristic equations, obtained by the application of differential equations with non-integer orders, it was perceived, with higher resolution, the behavior of damping intensity related to the undamped natural frequency.

The results generated by the optimized Decision Tree, trained with the input data received from the characteristic equations, shown 92.2% accuracy, ratifying the hybrid system applied is efficient and reliable.
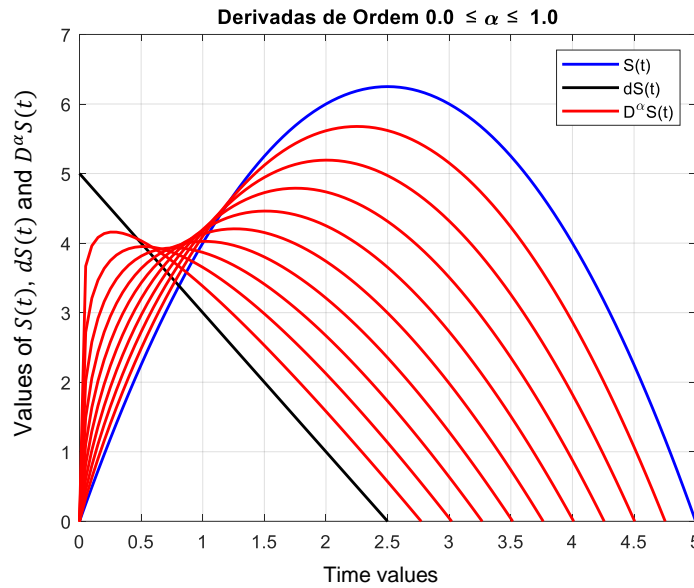
2. Methodology

This paper joints multiple fields of study, such as mathematical modeling, circuit analysis, computational intelligence, and microcontrollers. Since it applies techniques from various areas, the following topics will describe every step from this research that compounds this paper's methodology. It is also important to emphasize that the methodology implemented in this paper, takes as a great reference the guidelines proposed by Pereira et al. (2018), in which it is presented a set of well-structured steps to the organization of scientific papers.

2.1 Mathematical modeling

This paper will apply Fractional Order Differential Calculus (FODC) in its mathematical modeling, following the process presented in (Costa, D. C. L. et al, 2021a). The integer order derivatives reflect the variation rate at which some phenomenon changes, the real order derivatives allows one to extract more information about such phenomenon, an example in (Costa, D. C. L. et al, 2021b) shows a fractional order derivative outcomes when applied to a space function, shows not only the variation of an object through space, but also brings to the user the information about the eminence of the movement, in other words, it allows one to perceive the position change before the action properly occurs as seen in Figure 1.

Figure 1 – Fractional Order Derivative Curves.

Source: Costa, D. C. L. et al (2021b).

Where $S(t) = f(x) = 5x - x^2$ (blue line on Figure 1), its derivative $dS(t) = f'(x) = x - 2x$ (black line on Figure 1), and $D^\alpha S(t)$; $0.0 \leq \alpha \leq 1.0$ (red lines on Figure 1) are the fractional order derivatives. It is noteworthy that the fractional derivatives outcomes provide much more information between the space and speed function, that is, the time interval from an object to leave its state of rest until initializing the proper movement. It is also important to highlight that in Figure 1, the red lines closer to the left margin are the ones in which $\lim_{a \to 1} D^\alpha S(t) \sim dS(t)$, and the red lines dislocated to the right side of the graph is where $\lim_{a \to 0} D^\alpha S(t) \sim S(t)$, therefore one may comprehend this as an entropy inversion when dealing with the information of a phenomenon.

Through time, multiples mathematicians and physicians studied and developed the fractional order derivatives, the demonstration and structural differences occur from the diverse oriented scenarios in which each author aimed their method. These variations, yet bringing some decision difficulties, are mostly a good characteristic, because it allows researchers to choose the best method for their situation. Between multiples possibilities, this research considered some methods to apply, but to decide it is needed to evaluate those method's definitions and applicabilities.

One of the considered methods is the Riemann-Liouville, where its derivative of fractional order α of a function x(t) is shown in Equation 1.

$$D_{0,t}^\alpha x(t) = \frac{d^m}{dt^m} D_{0,t}^{-(m-\alpha)} x(t) = \frac{1}{\Gamma(m-\alpha)} \frac{d^m}{dt^m} \int_0^t (t-\tau)^{m-\alpha-1} x(\tau) d\tau \qquad (1)$$

Where $m - 1 \leq \alpha < m \in Z^+$, and Γ is the Euler-Gamma function.

Another method this paper considered is Caputo's, its derivative of fractional order α of a function x(t) is defined as presented in Equation 2.

$$D_{0,t}^{\alpha} x(t) = D_{0,t}^{-(m-\alpha)} \frac{d^m}{dt^m} x(t) = \frac{1}{\Gamma(m-\alpha)} \int_0^t (t-\tau)^{m-\alpha-1} x^m(\tau) d\tau \qquad (2)$$

Where $m - 1 < \alpha < m \, \epsilon \, Z^+$. From this definition, is noticeable that in Caputo's method $D_{0,t}^{\alpha} x^{(n)}(t) = D_{0,t}^{\alpha+n} x(t)$.

Both methods are mathematically applicable to the problems this paper works with, however, as seen in (Li, C. et al, 2011), the Riemann-Liouville method grants a wider horizon of possibilities since Caputo's method works in a shorter operational range. Another important feature to help deciding the most appropriate Fractional Order Derivative method is its computational implementation capability and easiness.

As described in the next topic, this project relies on computational tasks to better precision and real-time analysis, thus the compatibility of the mathematical method and modeling shall fit this project's necessities. In (Li, C. et al, 2011) and (Brzeziński, 2016), Riemann-Liouville was the main solution method applied because of its balance between low implementation complexity, high rates of precision and accuracy, keeping the method's characteristic of being applicable in a wide quantity of functions.

2.2 Computational intelligence modeling

As said in the introduction, this research seeks to develop an algorithm that allows complete automation, since its implementation until the later execution, to accomplish that, it is inevitable to utilize some computational intelligence method. The chosen technique, between various methods, was the Decision Tree (DT), this algorithm of computational intelligence grants great flexibility since it allows multiples utilizations as classification, regression, and prediction. (Azancort et al, 2021) presented an application of DT to improve the selection of areas for building a landfill, by discovering the control variables in multiple scenarios thanks to this computational technique. Among other possibilities, DT is also a low computational cost algorithm when compared to Artificial Neural Networks (ANN), (Ekiz et al, 2020) shows DT is also more efficient for prediction and for studies relying on small quantity of data.

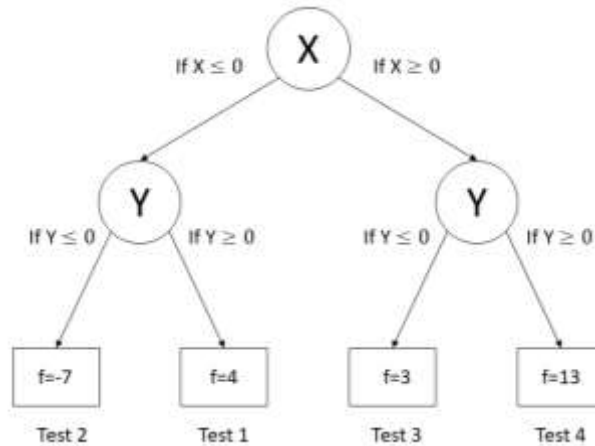To exemplify the DT working process, consider the data in Table 1.

Table 1 - Data Input Example

Test	Attribute 1 (X)	Attribute 2 (Y)	Result (f)
1	-1	1	4
2	-4	-1	-7
3	+3	-2	3
4	1	4	13

Source: Authors.

In Table 1 there are 4 Tests, also known as "observations", in which the table provides the inputs - or attributes - values of X and Y in given system, and in the last column the output values f of the system with respect to each input. Even with these few data, if one uses it as an input to a DT, one possible outcome (since the tree model may vary depending on the programming and setup utilized) is the model presented on Figure 2.

Figure 2 - Decision Tree Classification Outcome.

Source: Authors.

The DT analyzes the input data and provides a model capable of describing the behavior of this dataset, the resultant model can predict the results of inedited inputs, as it can also guess the input combinations from a given output. Based on the conditional relations the DT established, like seen in Figure 2 (If $Y \leq 0, X \geq 0,…$), the DT may also work as decision-making mechanism over the patterns the algorithm learned. Notice this example is simpler than the scenarios this paper will work with, but the intention is to easily explain the DT operation.

In this study the DT will be utilized in two different ways, the first is to analyze the circuit and elaborate a model based on the electrical grid architecture, DT is a classifier technique which its rules are produced by itself, and by so it compounds more of this paper's automation essence, so the classification property of the algorithm will be used, besides that the algorithm will also compare the attributes described in the next topic to evaluate its importance and impact on the circuit. The second application of the DT uses the algorithm decision making property, based on the possible scenarios produced by the DT at the previous step. The DT will use the FODC information gain to provide real time decisions, the computational platform will solve the differential equations developed on the mathematical modeling step according to the input and choose the better solution for this input based on the computational model developed by the classification tree.

To choose a programming language which fits the software requirements for this methodology some features must be available in the platform, as the data treatment, tools compatibility and intercommunication, besides computational cost, and results precision. (Costa, D. C. et al, 2019) and (Costa, D. C. et al, 2020a) utilize Python Programming Language as main tool for mathematical methods, mostly because of its libraries that allow symbolic mathematics and interactive visualizations. Nonetheless, (Costa, D. C. et al, 2021b) present MATLAB as a more efficient working tool, mainly in applied situations that need a high quantity of variables included in their analysis, as in (Costa, D. C. et al, 2020b) and (Silva et al, 2019).

The software chosen to develop the DT and connect it to the later explained controllers is MATLAB, which name stands for Matrix Laboratory, this platform works data in matrix format to achieve higher precision in problem solving based on geometric calculus and geometric algebra, as differential geometry problems. The geometric sustenance allows MATLAB to calculate via Graphical Processing Unit (GPU), instead of relying only on Central Processing Unit (CPU), which grants better performance, in (Semenenko et al, 2020) it is presented that, in average, a GPU realizes calculations 9 to 11 times faster than a CPU.

MATLAB provides toolboxes, which are specialized groups of functions, for the DT implementation, these toolboxes optimize the development of the computational intelligence method used in this paper since it drastically reduces the quantity of

commands, and consequently the amount of code lines. Another important feature is the toolbox for system control management provided by MATLAB, better explained in the Microcontroller Functionality topic, which enables the communication between hardware and software.

2.3 Physical attributes setup

For the DT to work properly, it is needed to set a mathematical meaning to correlate the attributes, the attributes on the Tree are the variables analyzed on the circuit. One of the attributes is the Current, defined by the rate at which charge moves past a point over time, as shown in Equation 3.

$$i = \frac{dq}{dt} \tag{3}$$

In which i is the current (in amperes) and dq is (in coulombs) the amount of charge moving past a point through a time interval dt (in seconds). Another important attribute is the potential difference (tension) between two points a and b, defined in Equation 4.

$$V_b - V_a = iR \tag{4}$$

Where V_a is the tension on point a, V_b the tension on point b, i the current defined in Equation 3, and R is the resistance which may be defined as shown in Equation 5.

$$R = \rho \frac{L}{A} \tag{5}$$

In which L is the length of the wire, A is the cross-sectional area of the wire, and ρ is material's resistivity. Nonetheless, the resistivity property varies with temperature, for metals in general the variation is approximately linear over a broad temperature range (Resnick & Halliday, 2011). A material's resistivity variation is defined in this paper as presented in Equation 6.

$$\rho - \rho_0 = \rho_0 \alpha (T - T_0) \tag{6}$$

In which T_0 is a selected reference temperature, and ρ_0 is the resistivity at such temperature. The α is known as the temperature coefficient of resistance, it means the change factor per degree of temperature change. For copper, the resistivity $\rho_0 = 1.69 \times 10^{-8} \Omega \cdot m$ when $T_0 = 293K$, and the $\alpha = 0.00404 K^{-1}$.

Another attribute included in the analysis is the energy transfer, known as power, and defined as shown in Equation 7.

$$P = iV \tag{7}$$

Being P the rate at which energy is transferred from one point to another. The unit of power is volt-ampere, that may be describe as

$$1V \cdot A = \left(1\frac{J}{C}\right)\left(1\frac{C}{s}\right) = 1\frac{J}{s} = 1W.$$

As shown above, Power unit may also be represented in Watts(W).

Since this paper seeks to improve energy saving while considering the resident comfort, it is essential to include the thermal energy emitted from the domestic office electrical grid, with this objective the analysis will include the dissipated power as thermal energy. Considering a device with resistance R, the resistance may also be described as Equation 8.

$$R = \frac{V}{i} \tag{8}$$

A device's resistance from Equation 8 may be combined with Equation 7 to provide the resistive dissipation as shown in Equation 9 and Equation 10.

$$P = i^2 R \tag{9}$$

$$P = \frac{V^2}{R} \tag{10}$$

It is important to notice that Equation 9 and Equation 10 applies only to the transfer of electric potential energy to thermal energy in devices with resistance.

This paper will also utilize the concepts from Kirchhoff's laws of circuits, using multichannel devices on the electrical grid. These laws are the strategies responsible for connecting the multiples variables and attributes here analyzed, the mathematical method of Kirchhoff's Laws is represented in Equation 11.

$$L\frac{d^2q}{dt^2} + R\frac{dq}{dt} + \frac{1}{C}q = V(t) \tag{11}$$

Where L is the inductance defined by Equation 12.

$$L = \frac{N\phi}{i} \tag{12}$$

In which N is the number of turns in the coil, the windings of the inductor are connected by the shared flux, and the product $N\phi$ is called magnetic flux linkage. The inductance (L) is, therefore, a measure of the flux linkage produced by the inductor per unit of current. The unit of inductance is tesla-square meter per ampere ($T \cdot m^2/A$), most often worked as henry (H), as presented in Equation 13.

$$1\, T \cdot \frac{m^2}{A} = 1H \tag{13}$$

The C in Equation 11, represents the rate of electrical potential between conductors caused by their stored electrical charge, also known as capacitance, this attribute described in Equation 14.

$$C = \frac{q}{V} \tag{14}$$

Capacitance mainly adopted unit is Farad (F), after the English physicist Michael Faraday.

When combining Equation 11 with Equation 1, one applies the fractional order derivative principles of Riemann-Liouville to the Kirchhoff Laws of circuits, resulting in Equation 15.

$$L\frac{d^m q}{dt^m} + R\frac{d^{m-1}q}{dt^{m-1}} + \frac{1}{C}q = V(t) \tag{15}$$

Where $1 \leq m \leq 2 \, \epsilon \, \mathbb{Q}^+$.

These attributes will contribute to the management of the analyzed electrical grid, so was necessary to prove their mathematical relation as strategy to provide physical meaning to the DT results, otherwise the computational outcome would not be enough to guarantee the model efficiency. Furthermore, the fractional order methods will provide more information about the circuit behavior, increasing the smart metering.

2.4 Microcontroller functionality

To effectively implement the mathematical and physical models to the electrical grid through the computational system developed, it is necessary a hardware control device capable of receiving the grid inputs and correctly obey the conditionals implemented on its programming. Both Arduino and Raspberry Pi devices work with the MATLAB tools, and that's a key
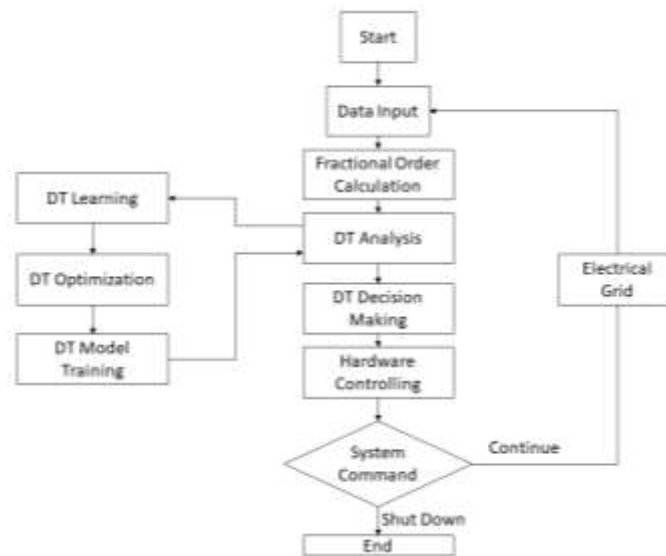
feature to develop this project over the same software, since it creates a broad horizon of possibilities.

In (González & Calderón, 2019), it is presented the application of Arduino platform in automation of Smart Grid systems, ensuring low cost, because of Arduino's opensource technology. (Li, Z. et al, 2020) presents a model of intelligent office lighting system using Arduino as its main controller, combined with illuminance sensors to deal with China's office lighting energy waste. (Khairudin et al, 2019) shows the possibility of developing fuzzy logic systems in MATLAB and converting it into C/Arduino scripts, achieving immediate hardware control application in an Arduino device.

The importance of connecting the Arduino device to MATLAB represents an easier path to data and commands transfer between these platforms, since MATLAB provides a toolbox for directly converting its scripts to C codes, the communication may be executed in low computational cost, since there's no need for heavy encoding, consequently, low time delay between input and response's system. This MATLAB feature carries great importance for bringing the DT conditional structures to Arduino's grid management and controlling. It is noteworthy that the Arduino physical implementation and I/O setup is out of the scope of the paper.

The workflow of the presented methodology is represented in Figure 3.

Figure 3 – Methodology Flowchart.

Source: Authors.

At this moment, the methodology workflow is well defined in all the addressed aspects, the flowchart illustrates the union of the steps well described on the previous topics, therefore the methodology proposed is consistent and cohesive. The next step is to analyze the practical execution of the algorithm on the running program, to evaluate parameters as precision, accuracy, computational cost, and time response.

3. Results and Discussion

This paper's whole computational system was implemented on MATLAB platform, since it offers all tools needed to accomplish the methodology presented on Figure 3. The DT methods were implemented using Classification Leaner tools, however the FODC codes were completely developed by the authors.

The first result comes from the application of the FODC techniques described in the mathematical modeling section, using the input data being Capacitance (C) = 0.001 Farad, Voltage (V) = 3 Volts, Resistance (R) = 50 Ohms, Inductance (L) =

10 Henry and Current (*i*) = 17 Ampere from (Abubakar et al, 2019), in which the attributes values are shown in Table 2.

Table 2 - Circuit Data From FODC Application.

	α = .1	α = .2	α = .3	α = .4	α = .5	α = .6	α = .7	α = .8	α = .9	DAMPING
T 00	2.957515	4.373448	6.46727	9.563525	14.14214	20.91279	30.92495	45.73051	67.62433	WEAK
T 01	2.660065	3.496247	4.523375	5.732623	7.062947	8.353107	9.259726	9.119841	6.723601	WEAK
T 02	2.477475	3.07342	3.736806	4.426574	5.061719	5.499541	5.505643	4.71435	2.579682	WEAK
T 03	2.318031	2.759792	3.209608	3.620085	3.914122	3.97352	3.625414	2.627458	0.65407	STRONG
T 04	2.1692	2.497605	2.79975	3.028624	3.114904	2.960781	2.433704	1.361225	-0.47097	CRITICAL
T 05	2.029642	2.27015	2.463432	2.565393	2.515198	2.232111	1.613095	0.532252	-1.15629	CRITICAL
T 06	1.900044	2.070871	2.18175	2.192565	2.050793	1.690207	1.030697	-0.01994	-1.56098	WEAK
T 07	1.781139	1.896257	1.94418	1.889159	1.686455	1.282248	0.614747	-0.3828	-1.77625	CRITICAL
T 08	1.673276	1.743778	1.743593	1.641369	1.399475	0.974683	0.320025	-0.61163	-1.86158	CRITICAL
T 09	1.576402	1.611236	1.574495	1.439046	1.173633	0.74403	0.115302	-0.74444	-1.85828	CRITICAL
T 10	1.490137	1.49657	1.43233	1.274217	0.996606	0.572879	-0.02215	-0.80853	-1.79583	CRITICAL
T 11	1.413878	1.397819	1.313193	1.14039	0.8587	0.447888	-0.10937	-0.82396	-1.69556	CRITICAL
T 12	1.346882	1.313127	1.213689	1.032182	0.752143	0.35865	-0.15932	-0.80565	-1.57284	CRITICAL
T 13	1.288334	1.240764	1.130859	0.945094	0.670641	0.296969	-0.18194	-0.76468	-1.43869	CRITICAL
T 14	1.237401	1.179138	1.062128	0.875357	0.609082	0.256369	-0.18487	-0.70929	-1.30092	CRITICAL
T 15	1.193263	1.126805	1.005266	0.819814	0.563307	0.231731	-0.17395	-0.64554	-1.1649	CRITICAL
T 16	1.155137	1.08247	0.95835	0.77583	0.529941	0.219019	-0.15365	-0.57785	-1.03424	CRITICAL
T 17	1.122293	1.044983	0.919731	0.74121	0.506252	0.215063	-0.12733	-0.50938	-0.91122	CRITICAL
T 18	1.094062	1.01333	0.888004	0.714133	0.490035	0.217384	-0.09752	-0.44231	-0.79714	CRITICAL
T 19	1.069838	0.986629	0.861976	0.693097	0.479521	0.224062	-0.06608	-0.37815	-0.69263	CRITICAL
T 20	1.049077	0.964112	0.840641	0.676865	0.473297	0.233617	-0.03438	-0.31782	-0.59783	CRITICAL
T 21	1.031299	0.945117	0.823151	0.664425	0.47024	0.244924	-0.00337	-0.26188	-0.51253	CRITICAL
T 22	1.016078	0.929077	0.808798	0.654953	0.469468	0.257137	0.02629	-0.21058	-0.4363	CRITICAL
T 23	1.003043	0.915508	0.796992	0.647779	0.470287	0.26963	0.054178	-0.16396	-0.36858	CRITICAL
T 24	0.991872	0.903998	0.787244	0.642366	0.47216	0.28195	0.080038	-0.12192	-0.30871	CRITICAL
T 25	0.982284	0.894198	0.77915	0.638281	0.474678	0.29378	0.103745	-0.08428	-0.25603	CRITICAL
T 26	0.974039	0.885815	0.772377	0.63518	0.477527	0.304903	0.125268	-0.05075	-0.20986	CRITICAL
T 27	0.966929	0.878602	0.766652	0.632792	0.480477	0.315185	0.144641	-0.02106	-0.16953	CRITICAL
T 28	0.960777	0.872352	0.761753	0.630902	0.483358	0.32455	0.161945	0.005105	-0.13442	CRITICAL
T 29	0.955432	0.866892	0.757501	0.629345	0.48605	0.332966	0.177291	0.028069	-0.10395	CRITICAL
T 30	0.950766	0.86208	0.75375	0.627994	0.488472	0.340437	0.190807	0.048135	-0.07758	CRITICAL
T 31	0.94667	0.857796	0.750383	0.626751	0.490573	0.346985	0.202632	0.065596	-0.05481	CRITICAL
T 32	0.943051	0.853943	0.747308	0.625546	0.492322	0.352652	0.212907	0.080729	-0.0352	CRITICAL
T 33	0.939832	0.850439	0.744449	0.624324	0.493709	0.357488	0.221773	0.09379	-0.01837	CRITICAL
T 34	0.936947	0.847219	0.74175	0.62305	0.494732	0.36155	0.229365	0.105013	-0.00394	CRITICAL
T 35	0.934342	0.844227	0.739165	0.621698	0.495401	0.364898	0.23581	0.114614	0.008385	STRONG
T 36	0.931971	0.84142	0.736659	0.620253	0.495732	0.367593	0.24123	0.122785	0.018892	STRONG
T 37	0.929794	0.838762	0.734206	0.618706	0.495745	0.369694	0.245736	0.129702	0.027821	STRONG
T 38	0.927781	0.836224	0.731785	0.617053	0.49546	0.371259	0.24943	0.135519	0.035387	STRONG
T 39	0.925904	0.833783	0.729382	0.615295	0.494903	0.372342	0.252405	0.140375	0.041777	STRONG
T 40	0.92414	0.83142	0.726987	0.613435	0.494096	0.372995	0.254747	0.144393	0.047153	STRONG
T 41	0.922472	0.829121	0.724593	0.61148	0.493064	0.373263	0.256531	0.147681	0.051657	STRONG
T 42	0.920884	0.826873	0.722195	0.609436	0.491828	0.373193	0.257826	0.150336	0.055411	STRONG
T 43	0.919364	0.824668	0.71979	0.60731	0.490412	0.372822	0.258693	0.152441	0.058522	STRONG
T 44	0.917901	0.822497	0.717377	0.605111	0.488836	0.372188	0.259186	0.15407	0.061082	STRONG
T 45	0.916486	0.820357	0.714956	0.602846	0.487119	0.371324	0.259355	0.155287	0.063169	STRONG
T 46	0.915113	0.818242	0.712529	0.600525	0.485278	0.370259	0.259242	0.15615	0.064853	STRONG
T 47	0.913776	0.81615	0.710096	0.598154	0.483331	0.36902	0.258885	0.156708	0.066191	STRONG
T 48	0.91247	0.814078	0.707659	0.595741	0.481292	0.36763	0.258317	0.157003	0.067235	STRONG
T 49	0.911191	0.812025	0.70522	0.593294	0.479174	0.366111	0.257568	0.157072	0.068029	STRONG
T 50	0.909935	0.809989	0.70278	0.590818	0.47699	0.364481	0.256664	0.156949	0.068608	STRONG

Source: Authors.

Table 2 shows the behavior of the Differential Equation from Kirchhoff Laws applied to the chosen circuit with the proper input data. Table 2 structure consists of the analysis over time, discretized on the first column, from 0 to 5 seconds with steps of one tenth of a second (1s∕10), resulting on 51 rows of time progression. Columns 2 to 10 present the circuit behavior on the same time instant but with dissimilar orders of the Non-Integer Order Derivative (NIOD), these orders values are symbolized by α, in which the relation of α and m from Equation 15 is set by the relation below.

$$\alpha = m + 1$$

This relation allows simplifying the quantity of alpha (α) values in Table 2 to work with the first and second order derivatives needed in Equation 15. One can understand that every value in the range between rows 2 to 52 and the columns 2 to 10, is the Voltage value in the circuit at a T time at a α order in the NIOD. The 11th column shows the damping rate classification of the circuit at a T time, and it's defined by the following relations.

If $i < \frac{\sqrt{LC}}{2vC}$, then the damping rate is considered STRONG,

If $i > \frac{\sqrt{LC}}{2vC}$, means the damping rate is WEAK,

An if $i = \frac{\sqrt{LC}}{2vC}$, the damping rate is classified as WEAK.

The Damping Rate attribute defines the circuit behavior, so it is extremely important for the construction of the DT, working as its labels, as shown in the following steps of the workflow.

With the data in Table 2, it is possible to create an initial tree as a manner to allow the computational method to first learn the conduct of the dataset, this initial execution returns a fitted binary classification decision tree according to the attribute values (voltage) in table, and each moment's response (damping rate classification). By then, the tree was implemented to execute a data classification only, and the result of this process is presented in Figure 4.

Figure 4 – Structure of Classification Tree.

Source: Authors.

The classification tree in Figure 4 may also be represented in algorithm structure as shown in Figure 5.

Figure 5 – Algorithm of Classification Tree.


```
Command Window                                                              ⊛
  >> load('MyWorkspace.mat')

  Decision tree for classification
  1  if x9<0.0022225 then node 2 elseif x9>=0.0022225 then node 3 else CRITICAL
  2  if x1<1.84059 then node 4 elseif x1>=1.84059 then node 5 else CRITICAL
  3  if x1<2.39775 then node 6 elseif x1>=2.39775 then node 7 else STRONG
  4  class = CRITICAL
  5  if x1<1.96484 then node 8 elseif x1>=1.96484 then node 9 else CRITICAL
  6  class = STRONG
  7  class = WEAK
  8  class = WEAK
  9  class = CRITICAL

fx >>
```
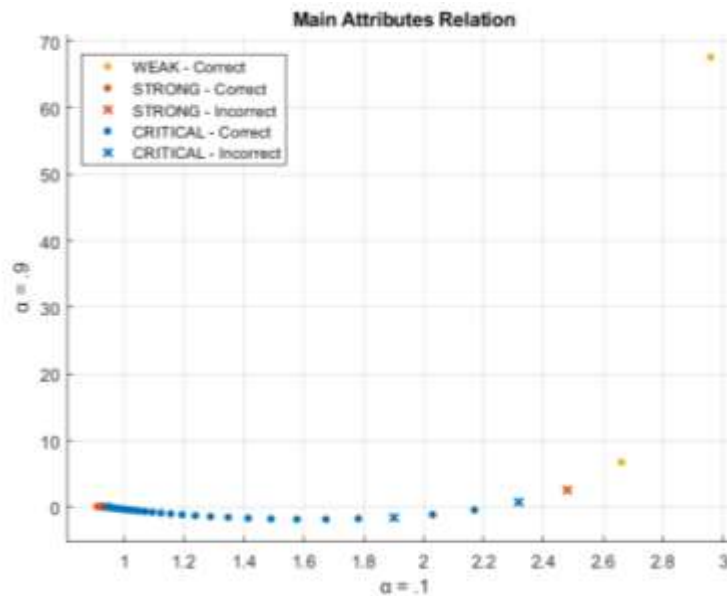
Source: Authors.

In which x is the equivalent to the m element value in the Equation 15, the decision nodes are the NIOD orders and leaf nodes are the damping behavior classification. This classification tree brings to this research the information that, even though the NIOD extracts more information about a phenomenon, for this dataset it is only necessary the data from the second (α=.1) and tenth (α=.9) columns do describe the damping from the circuit at a given moment, so these variables may be considered the main attributes. This conclusion is important for allowing optimization of the computational cost, since it shows no need for loading the whole dataset to deal with future decision making, by reaching a reasonable classification tree with approximately 22.2% of data available data. The relation between the main attributes is shown in Figure 6.

Figure 6 – Relation of the main attributes.

Source: Authors.

Since this paper seeks to build optimal solutions for real time smart grid management, the optimization process is indispensable for reducing input-output delay, so the next step was improving the tree for quicker results. To achieve this goal was necessary to utilize classification leaner features on MATLAB programming language, in which the authors were able to train 3 DT models, with the main difference being the Maximum Number of Splits (MNS). The models applied were Coarse

Tree (MNS=4), Medium Tree (MNS=20), and Fine Tree (MNS=100), in the comparative were also included the evaluation more parameters as presented on Table 3.
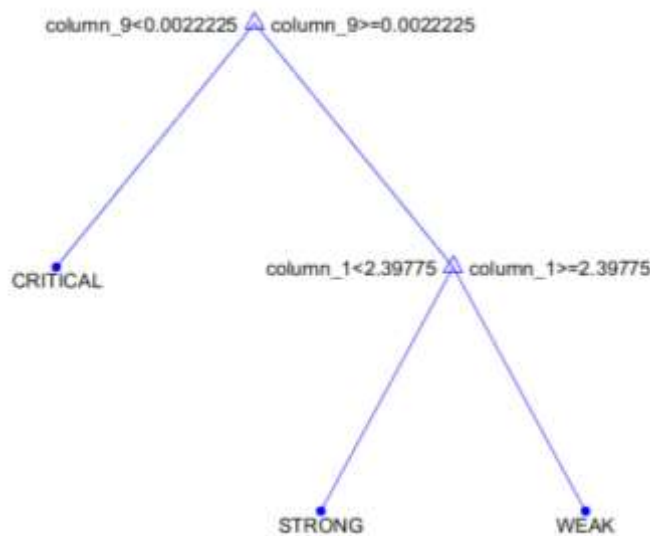
Table 3 – Trees Performance

Parameters	Fine Tree	Medium Tree	Coarse Tree
Accuracy	92.2%	92.2%	92.2%
Misclassification Cost	4	4	4
Prediction Speed	~3700 obs/sec	~4500 obs/sec	~5300 obs/sec
Training Time	1.3448 sec	0.8059 sec	0.45676 sec
MNS	100	20	4

Source: Authors.

As every method achieved the same accuracy rate and misclassification cost, to choose the best technique for this scenario it was considered the Training Time as the selection parameter, mainly for the persistence of this study to develop an efficient real time analysis system. From this perspective, as Coarse Tree was the best performance method, achieving the same results of the finer methods but with lower execution time, it was chosen to be trained model. The Coarse Tree structure is presented in Figure 7.
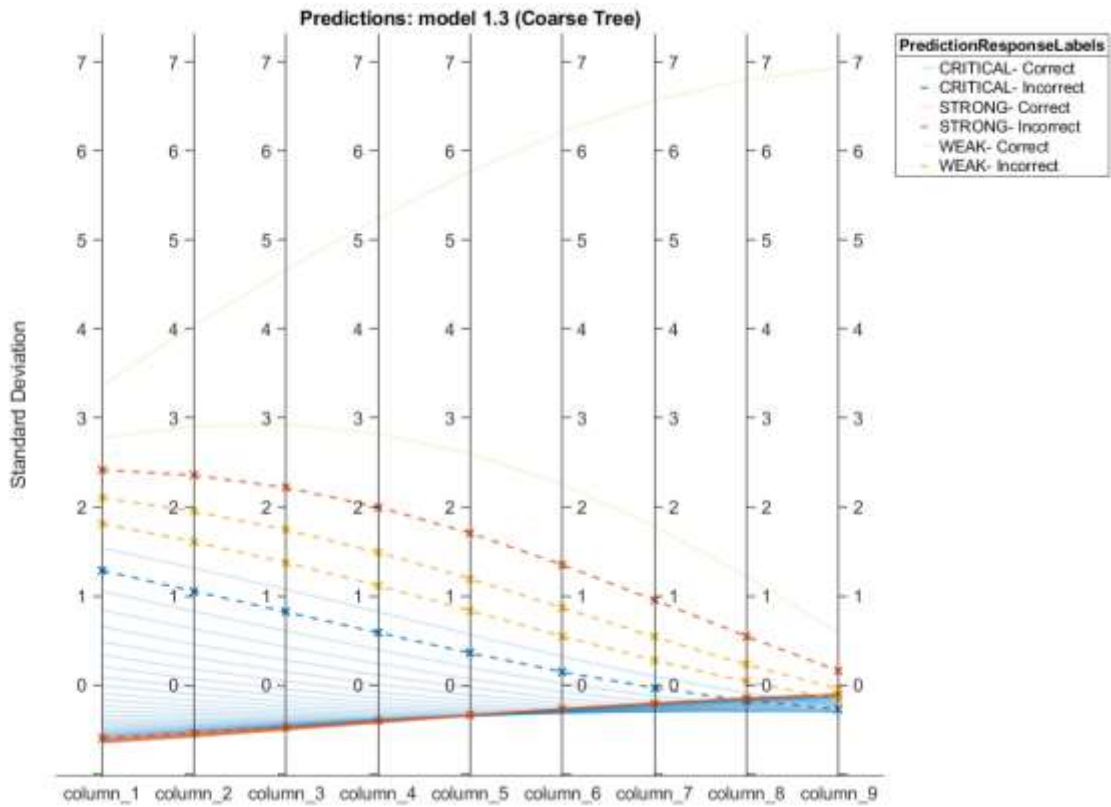
Figure 7 – Structure of Coarse Tree.

Source: Authors.

This Coarse Tree is an optimization of the Classification Tree presented on Figure 4, reducing the quantity of decision nodes based on the estimated probabilities from each leaf node results. With the Coarse Tree, this research achieved a trained model for current and future data evaluation, this model brought many results, being the DT structure one of them, and the behavior representations of the input data, one of these representations is shown on Figure 8.
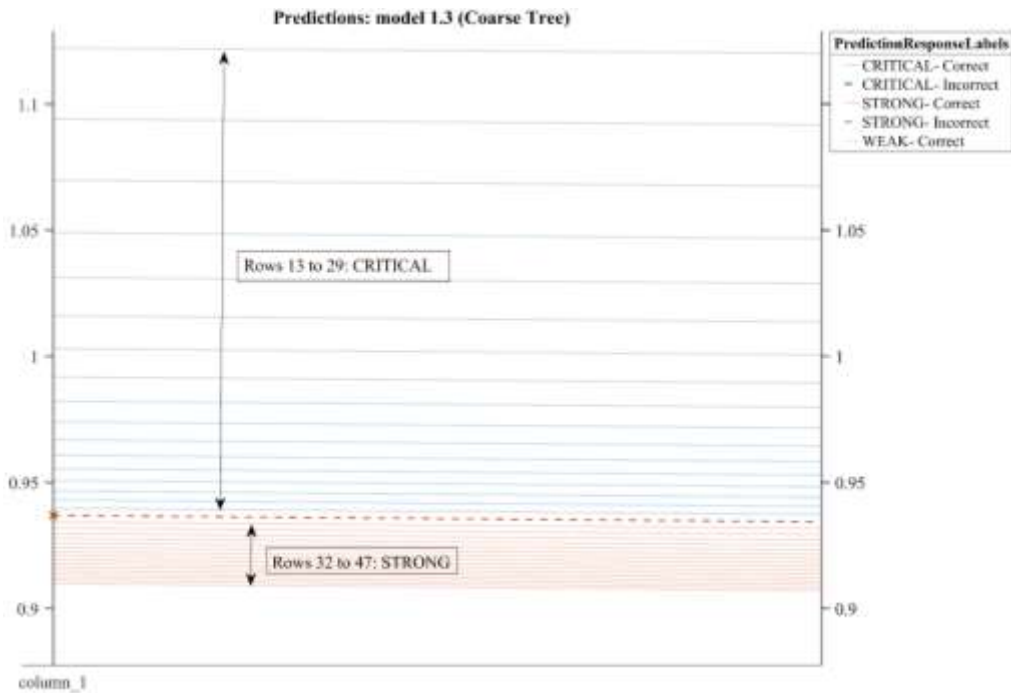
Figure 8 – Trained Model Parallel Coordinate Plot (Z-Score).

Source: Authors.

Figure 8 is a Parallel Coordinates Plot (PCP), each column (attribute) from this line is symbolized by a point, and each row from the table becomes a line in the plot connecting the points, line colors represent its classification label (response), misclassified points are illustrated by dashed lines, so the dataset is brought to a representation according to its numerical behavior. This figure displays the plotting on a z-score scaling, thus presents how far below or above a score is from the dataset mean, this "distance" is measured in units of standard deviations, as noticeable in the plot vertical axis. This illustration method didactic makes it easier to comprehend medium or large datasets, as example when zooming in the Figure 8, one can see the different damping rate classification over time instants in the analyzed time interval, this zooming generates Figure 9.

Figure 9 – Trained Model Parallel Coordinate Plot (No Scaling).
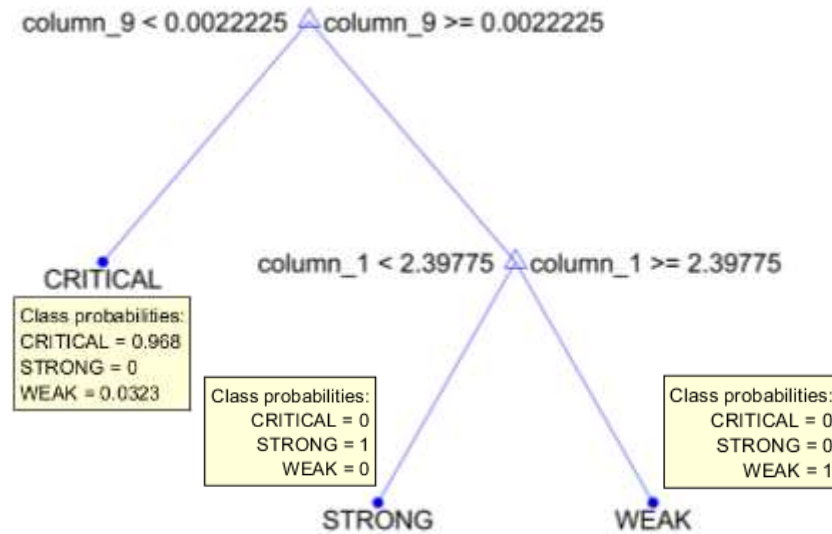

Source: Authors.

A zoomed in observation increases precision at cost of resolution, however the figure interface generated on MATLAB allows zooming in and zooming out at any moment, without any lost in the trained model data. The improved phenomenon data extraction feature surely improves more precise and complex analysis, however when uploading the software functionality to the microcontroller, it shall be able to performance decision making in the best time possible.

To achieve great processing time, it was used the observation made after analyzing the Classification Tree figure, in which is said that data from the second (α=.1) and tenth (α=.9) columns are enough to represent the studied system, reducing the data load every time a calculation is needed. Relying on this information, and combining with the different tree techniques comparative, the authors trained a model that uses Coarse Tree architecture and only the (α=.1) and (α=.9) data, resulting in an Optimized Coarse Tree, with structure presented in Figure 10.

Figure 10 – Optimized Coarse Tree.

Source: Authors.

The yellow rectangles in Figure 10 show the estimated probabilities of every circuit damping rate classification in each leaf node, thus the system does not disregard a probabilistic analysis, since deterministic proposals are often rough approximations. Even though the models in Figure 7 and Figure 10 are visually similar, their difference becomes explicit when comparing their training results, as shown in Table 4.
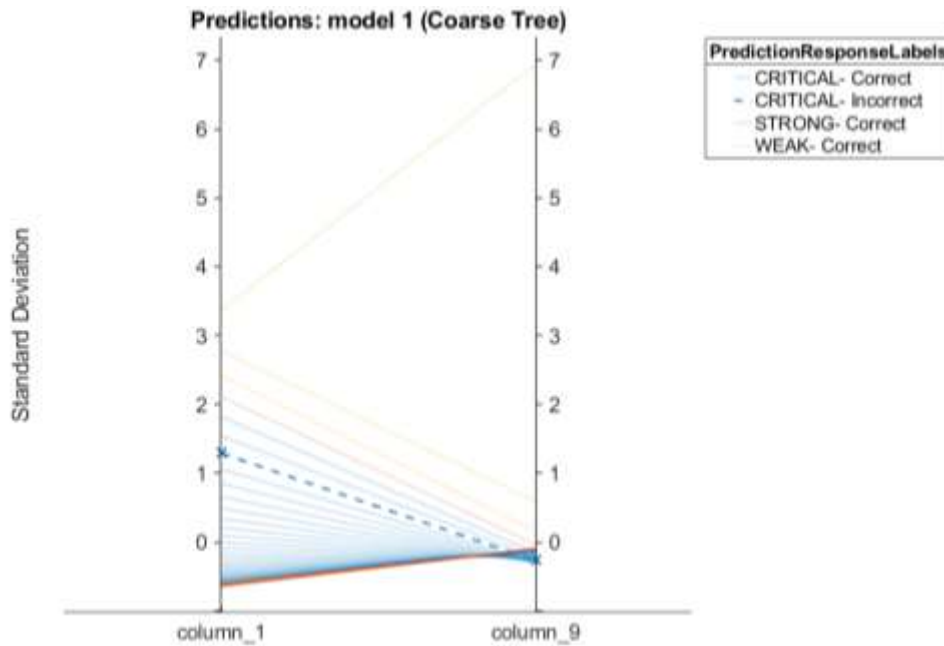
Table 4 – Coarse Trees Performance.

Parameters	Coarse Tree	Optimized Coarse Tree
Accuracy	92.2%	98.0%
Misclassification Cost	4	1
Prediction Speed	~5300 obs/sec	~13000 obs/sec
Training Time	0.45676 sec	0.86332 sec
MNS	4	4

Source: Authors.

Comparing the same parameters used in Table 3, one can notice the superior results of the Optimized Coarse Tree, as the higher accuracy rate and the 4 times smaller Misclassification Cost. Even thou the Training Time was approximately 1.89 times longer, it's a fair cost over the 2.45 times faster Prediction Speed, which was able to perform nearly 13000 observations per second. It is also noticeable that, in Table 3 comparative analysis, the Training Time was the selection parameter because the other tree models performed the same Accuracy and Misclassification Cost, however the Optimized Coarse Tree results are arguably better, compensating the longer training time. The Parallel Coordinates Plot of the Optimized Coarse Tree is shown in Figure 11.

Figure 11 – Optimized Trained Model Parallel Coordinate Plot (Z-Score).

Source: Authors.

As decide, only the columns 1 and 9 were used, reducing the quantity of misclassification elements to 1 and increasing the Accuracy Rate. Loading selected data instead of the whole dataset reduce the quantity of analyzed values from 459 to 102, resulting in datasheet files 3517 bytes lighter. This reduction of data loading may not look overly impressive by itself, but it shall be considered this system will be in continuous processing, repeatedly calculating the new values from the circuit, therefore this carrying out less information will directly impact on energy saving and resources usage, as CPU and RAM.

4. Conclusion

It is declared over this paper that its main objective was to propose and effectively build, a well-structured methodology to support the combination of mathematical and computational methods over a hybrid modeling. This proposal goal was to search for a modern and robust solution for the current energy management problems brought by recent changes in society's behavior. Following that goal, this paper brought up a mathematical-computational modeling by using Fractional Order Differential Calculus along with Decision Tree computational intelligence model techniques, this modeling process compared different techniques of both mathematical and computational tools, as the FODC solutions methods and the DT structures.

After elaborating the whole functionality of the proposed model, this paper shows the results that prove its model viability and efficiency, the authors of this research are pleased with the built methodology, for its capability of solving the analyzed scenario of smart power management. It is expected that this methodology works as reference for other researchers, in the common goal to improve power management efficiency.

For future research, the authors of this paper plan to adjust this methodology for greater and more complex scenarios. It is also intended to increase the algorithm's performance by the application of different computational intelligence techniques, which may provide an improvement on the Internet of Things (IOT) made possible by the 5G technology. Looking for more probabilistic analysis, future works will include Bayesian Networks implementation to better comprehend modern stochastic systems.

Acknowledgments

References

Abubakar, A., Grace, A., & Akinola, S. (2019). The Development of a Basic Electricity Trainer. *International Journal of Advanced Engineering Research and Science*, 6(5), 36-43, https://doi.org/10.22161/ijaers.6.5.5.

Azancort, Neto. J. L., Gonçalves, A. L. S., Cruz, B. C. C. da, Gomes, L. L., & Costa, D. C. L. (2021). Artificial Intelligence implemented to recognize patterns of sustainable areas by evaluating the database of socioenvironmental safety restrictions. *Research, Society and Development.* 10(10), e212101018841. https://doi.org/10.33448/rsd-v10i10.18841.

Baraniuk, C. (2021, August 27). "Why is there a chip shortage?". *BBC News*, Business, Retrieved from https://www.bbc.com/news/business-58230388.

Brzeziński, D. W. (2016). Accuracy Problems of Numerical Calculation of Fractional Order Derivatives and Integrals Applying the Riemann-Liouville/Caputo Formulas. *Applied Mathematics and Nonlinear Sciences*, 1(1), 23-44, https://doi.org/10.21042/AMNS.2016.1.00003.

Costa, D. C. L., Costa, H. A. de O., & Neves, L. P. (2019). *Métodos Matemáticos Aplicados nas Engenharias via Sistemas Computacionais*. (2nd ed.). Belém, PA, SINEPEM-IFPA.

Costa, D. C. L., Costa, H. A. de O., Castro, A. P. S., Cruz, E. C., Azancort Neto, J. L., & Cruz, B. C. C. da. (2020b). The dimensions of Mathematical and Computational Modeling prescribed to Environmental Management. *Research, Society and Development*, 9(10), e6939109013, https://doi.org/10.33448/rsd-v9i10.9013.

Costa, D. C. L., Costa, H. A. de O., Silva, H. C. M. da, & Silva, S. T. T. da. (2021b). *Matemática Computacional Aplicada à Ciência e Tecnologia*. (1st ed.). Belém, PA, SINEPEM-IFPA.

Costa, D. C. L., Costa, H. A. de O., Silva, S. T. T., (2021a). Quantum de Informação: A Entropia do Movimento sob a Óptica das Derivadas de Ordem Não Inteira. In D. M. Rosário, D. C. L. Costa, & J. C. Suzuki (Eds.), *Ensino-Aprendizagem em Ciências, Matemática e Tecnologia,* pp. 20-50, edusp.

Costa, D. C. L., Costa, H. A. O. de, Oliveira Neto, J. C. de & Castro, W. de O. (2020a). Análise e Decisão Sobre a Otimização de Preços em Função da Taxa de Deterioração De Produtos, in *Linguagens, Culturas, Tecnologias e Inclusão*, vol. 1, FFLCH/USP, pp. 321-354.

Costa, H. A. de O., Costa, D. C. L., & Meneses, L. A. de. (2021a). Interdisciplinarity Applied to the Optimized Dispatch of Integrated Electricity and Natural Gas Networks using the Genetic Algorithm, *Research, Society and Development*, 10(2), e42110212641, https://doi.org/10.33448/rsd-v10i2.12641.

Costa, H. A. de O., Gomes, L. L., & Costa, D. C. L. (2021b). Genetic algorithm and particle swarm applied in electric system optimization. *Research, Society and Development*, 10(10), e166101018871, https://doi.org/10.33448/rsd-v10i10.18871.

Ekiz, B., Baygul, O., Yalcintan, H., & Ozcan, M. (2020). Comparison of the decision tree, artificial neural network and multiple regression methods for prediction of carcass tissues composition of goat kids, *Meat Science*, 161(1), 1-45, https://doi.org/10.1016/j.meatsci.2019.108011.

González, I., & Calderón, A. J. (2019). Integration of open source hardware Arduino platform in automation systems applied to Smart Grids/Micro-Grids, *Sustainable Energy Technologies and Assessments*, 36(100557), 1-12, https://doi.org/10.1016/j.seta.2019.100557.

Hagenbeck, E., Santos, N. B., Cavalcante, R. & Macedo, H. (2016, April 1-29). *Using fuzzy logic in smart homes lighting controllers*, 8th Euro American Conference on Telematics and Information Systems (EATIS). Washington, DC. https://doi.org/10.1109/EATIS.2016.7520129.

Hoskins, P. (2021, September 30). "China power cuts: What is causing the country's blackouts?". *BBC News*, Business, https://www.bbc.com/news/business-58733193.

Khairudin, M., Wijaya, H. A. & Muslikhin, H. A. (2019, SEP 14-15). *Converter matlab fuzzy inference to arduino Csystem*, The 5th International Conference on Technology and Vocational Teachers (ICTVT 2019), Yogyakarta, Indonesia. https://iopscience.iop.org/issue/1742-6596/1456/1.

Li, C., Qian, D., Chen, Y. (2011). On Riemann-Liouville and Caputo Derivatives. *Discrete Dynamics in Nature and Society*, 2011(562494), 1-15, https://doi.org/10.1155/2011/562494.

Li, Z., Li, J., Li, X., Yang, Y., Xiao, J., & Xu, B. (2020). Design of Office Intelligent Lighting System Based on Arduino, *Procedia Computer Science*, 166(1), 1-5, https://doi.org/10.1016/j.procs.2020.02.035.

Mota, C. (2021, May 31). Como seca histórica no Brasil traz risco de inflação e racionamento de energia. *BBC Brasil*. Retrieved from https://www.bbc.com/portuguese/brasil-57290389.

Pereira, A. S., Shitsuka, D. M., Parreira, F. J. & Shitsuka, R. (2018). *Metodologia da pesquisa científica*. Núcleo de Tecnologia Educacional –UFSM.

Resnick, R., & Halliday, D. (2011). *Fundamentals of Physics*. (9th ed. pp.690-691). Wiley.

Semenenko, J., Kolesau, A., Starikovičius, V., Mackūnas, A., & Šešok, D. (2020). Comparison of GPU and CPU efficiency while solving heat conduction problems. *Mokslas – Lietuvos ateitis / Science – Future of Lithuania*, 12(24), 1-5, https://doi.org/10.3846/mla.2020.13500.

Silva, G. D., Costa, D. C. L., & Costa, H. A. de O. (2019). Método de inteligência computacional aplicado na otimização da produção por meio da detecção do sistema de coordenadas do centro de massa de uma estrutura, in *Gestão da Produção em Foco*. (pp170-180). Poisson. https://doi.org/10.5935/978-85-7042-059-6.

Wilmot, S. (2021, November 17). "The Great Car-Chip Shortage Will Have Lasting Consequences.". *The Wall Street Journal*, Heard On The Street, https://www.wsj.com/articles/the-great-car-chip-shortage-will-have-lasting-consequences-11632737422.