

Forecasting Chemical Characteristics of Aircraft Fuel Using Artificial Neural Networks

Felipe Valverde Rocha^{1,*} , Koshun Iha¹ , Thiago Antonio Grandi de Tolosa² 

1. Departamento de Ciência e Tecnologia Aeroespacial – Instituto Tecnológico de Aeronáutica – Departamento de Química – São José do Campos/SP – Brazil. **2.** Instituto Mauá de Tecnologia – Departamento de Engenharia Elétrica – São Caetano do Sul/SP – Brazil.

*Corresponding author: felipe_rocha_tri@hotmail.com

ABSTRACT

Aircraft fuels, called jet propulsion, are used in several areas of activity within aeronautics. There are jet fuels based on kerosene, that is, those obtained commercially, and there are synthetics produced in the laboratory. All of these fuels are included within the so-called propellants. In this article, Jet propulsion-8 (JP 8) fuel was used as the basis for data analysis, and thus two temperature ranges were analyzed. The first range, from 300 to 2500 K, was analyzed for specific heat, enthalpy and entropy. Based on theoretical and experimental data, artificial neural networks (ANNs) were developed to identify these properties in other working conditions, that is, at other temperatures.

Keywords: Fuel; Temperature; Enthalpy; Entropy; Heat.

INTRODUCTION

Fuel is a substance that, when reacts with oxygen, releases energy. Therefore, there are several fuels, and among the best-known are those derived from petroleum, such as gasoline and diesel, which are normally used in land motor vehicles. Another type of fuel used in motor vehicles is nonpolluting fuels, such as electric cars.

In this context, there are also fuels that are used in aircraft. Most are based on kerosene, which is also a derivative of oil, and they have uses and characteristics that can be very different. Such characteristics are linked to the fact that these fuels are classified as propellants, that is, they are substances that, alone or in combination with others, are burned in a large mass of gaseous products at high temperature, which, escaping through the chamber discharge piston combustion, provide by reaction, the engine impulse.

These propellants may have commercial and/or military uses. Those for commercial use are employed in commercial or transport aircrafts and are based on kerosene, derived from petroleum, which has the name QAV-1. Those for military use have other names, such as jet propulsion-8 (JP-8). The jet propulsion-4 (JP-4) was used by the United States Air Force until 1995 and, after that, replaced by JP-8, as it is less flammable and less dangerous (Center 2008).

The JP-8 has 81.20% isoparaffins, 0.30% oils, 4.90% naphthalene and 13.00% aromatic compounds (Pires et al 2018). Among them, 16.90% is C_9H_{12} , 62.55% is $C_{11}H_{24}$ and 21.05% is $C_{10}H_{20}$ (Xu *et al.* 2015).

Received: Oct. 9, 2020 | Accepted: Mar. 16, 2021

Peer Review History: Single Blind Peer Review.

Section Editor: T John Tharakan



This is an open access article distributed under the terms of the Creative Commons license.

State-of-the-art

The study of propellants is something that is being done a lot nowadays, because more and more are looking for cheap and easily manufactured fuels. As they are flammable fuels, it is not possible to work at high temperatures in the laboratory. Therefore, the study at high temperatures simulates certain fuels at very high combustion rates, where in a laboratory it would not be able to reach.

Therefore, it is possible to study a variety of temperatures and select within them the one that is the most interesting. The use of neural networks helps in this process, where it is possible to train a network to analyze and give results close to the experimental ones and also at temperatures that it is not possible to work with.

So, with that the range of temperatures to be studied for future fuels is huge.

THEORY

Artificial neural network

The work on artificial neural networks, usually called “neural networks”, has been motivated since the beginning by the recognition that the human brain processes information in an entirely different way from the conventional digital computer (Haykin 2001). They allow the projection of values on a laboratory scale and on pilot scale in the various areas studied worldwide. Thus, the human nervous system can be seen as a three-stage system, as show in Fig. 1 (Arbib 1987).

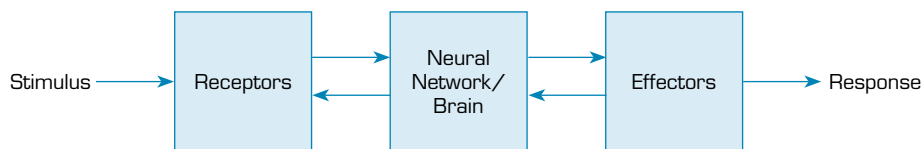


Figure 1. Block diagram representation of nervous system.

The center of the system is the brain, represented by the neural network, which continuously receives information, perceives and makes appropriated decisions. Two sets of arrows are shown in Fig. 1. Those from left to right indicate the forward transmission of the information-carrying signal through the system. Arrows pointing from right to left indicate the presence of feedback in the system. The receptors convert stimuli from the human body or the external environment into electrical impulses that transmit information to the neural network. The actuators convert electrical impulses generated by the neural network into discernible responses as outputs from the system (Haykin 2001).

In practice, however, neural networks cannot provide a solution working individually. Instead, they need to be integrated into a consistent system engineering approach (Haykin 2001). Thus, a complex problem is decomposed into a number of relatively simple tasks, and a subset of tasks is assigned to neural networks that coincide with their ability to solve such problem.

The use of neural networks offers the following useful properties and capabilities:

- Nonlinearity;
- Input-output mapping;
- Adaptability;
- Responses to evidence;
- Contextual information;
- Fault tolerance;
- Very large-scale integration;
- Uniformity of analysis and design;
- Neurobiological analogy.

Activation function

There are several activation functions that can be used and the choice depends on the purpose and complexity of the neural network. For this, it is necessary to analyze the problem involved in order to choose the best activation function. Table 1 shows the most frequently used.

Table 1. Activation functions.

Linear	$f(x) = ax$
Sigmoidal	$f(x) = \frac{1}{1+e^x}$
Hyperbolic tangent	$f(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$
Softmax	$f(x) = \frac{e^{x_j}}{\sum_j e^{x_j}}$
Exponential Rectilinear Unit	$f(x) = \max(0, x)$
Softplus	$f(x) = \ln(1 + e^x)$

Learning/training algorithms

Learning and training algorithms serve to “teach” the neural network how to work with the data. Thus, it is able to predict values and make approximations closest to the real value. There are two types of learning algorithms, the so-called supervised and the unsupervised.

Unsupervised learning is used when the desired answer is not known, that is, there is no output value. It is possible to map the input data and bring it close to the expected output value. The most cited algorithms to date in this category are:

- Additive Grossberg (AG) (Haykin 2001);
- Adaptive resonance theory (ART) (Carpenter and Grossberg 2010);
- Symmetric and asymmetric Hopfield (Sousa 2014);
- Bidirectional associative memory (BAM) (Sousa 2014);
- Temporal associative memory (TAM) (Vieira and Araujo 1997);
- Competitive learning – SOM by Kohonen (Haykin 2001).

There is also supervised learning, where the output values are known. This type of learning is widely used when there is a theoretical or experimental data, so it is possible to compare and even calculate errors between those values. It also allows to map the input data to get close to or equal to the expected output data. The most cited algorithms in this category are:

- Boltzmann machine (BM) (Haykin 2001);
- Recurring real-time learning (Oliveira and Siqueira 2013);
- Adaptive logic network (ALN);
- Back propagation (BP) (Haykin 2001);
- Cauchy machine (CM);
- Kalman filter (EKF) (Aguirre 2015).

Multilayer artificial neurons (multilayer perceptron)

Typically, the network consists of a set of sensory units (source nodes) that constitute the input layer, one or more hidden layers of computational nodes and an output layer of computational nodes. The input signal propagates forward through the network layer-by-layer (Haykin 2001). Figure 2 shows the representation of a multilayer perceptron (Coutinho *et al.* 2016).

Multilayer perceptron has been successfully applied to solve several complex problems through their supervised training with a very popular algorithm, known as error propagation algorithm. This algorithm is based on the error

correction-learning rule (Haykin 2001). One of the mostly used algorithms nowadays (and that will be used in this article) is the multilayer perceptron network trained by the retro propagation algorithm. The learning by retro propagation of error consists of two steps.

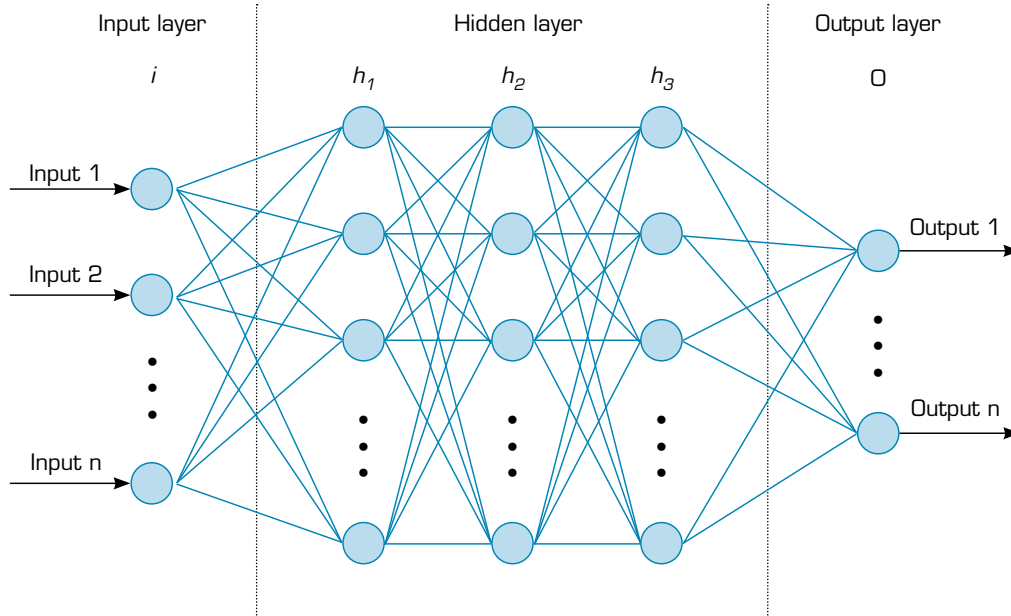


Figure 2. Representation of a multilayer perceptron.

The descending gradient

The descending gradient is a technique used in learning algorithms that involves calculating derivatives of the objective function to find the point with the greatest possible slope in the direction of the minimum error. Such variables, to be optimized, are shifted in a negative direction, that is, this will reduce the value of the objective function and give the best possible optimization. This method can have two points, the gradient with and without the moment factor (Rodríguez 2011).

MATLAB

MATLAB is a high-performance interactive software focused on numerical calculation. It also integrates numerical analysis, matrix calculation, signal processing and graph construction in an easy-to-use environment, where problems and solutions are expressed only as they are written mathematically, unlike traditional programming.

EXPERIMENTAL

The experimental part was based on the article by Xu *et al.* (2015). In the mentioned work, the authors obtained, experimentally, physical-chemical properties, such as enthalpy, entropy, specific heat, variation of enthalpy, heat of formation and Gibbs free energy for several types of aircraft fuels, such as jet fuel-A (JET-A), jet propulsion-5 (JP-5) and JP-8, and also alternative fuels from some companies, such as Gevo and Virent. The percentage of components that each fuel has is also determined. The experimental data are shown in Table 2.

Table 2. Jet propulsion-8 fuel experimental data. Retrieved from Xu R, Wang H, Colket M, Edwards M (2015) Thermochemical properties of jet fuels [Internet]. Stanford: Stanford University, p. 6.

T (K)	Cp (cal/mol-K)	S (cal/mol-k)	H(T)-H(298) (kcal/mol)	Hf (kcal/mol)	Gf (kcal/mol)
298	54.325	121.203	0	-65.114	5.628
300	54.646	121.567	0.109	-65.201	6.103
400	70.216	139.447	6.362	-69.298	30.505
500	84.305	156.662	14.102	-72.747	55.869
600	96.474	173.139	23.158	-75.53	81.865
700	106.672	188.8	33.331	-77.719	108.278
800	115.244	203.619	44.438	-79.415	134.97
900	122.926	217.64	56.35	-80.682	161.848
1000	130.845	230.993	69.031	-81.49	188.844
1100	135.847	243.702	82.369	-81.894	215.899
1200	140.409	255.721	96.185	-82.066	242.981
1300	144.555	267.126	110.437	-82.037	270.068
1400	148.31	277.979	125.083	-81.836	297.146
1500	151.7	288.329	140.087	-81.489	324.205
1600	154.75	298.218	155.412	-81.021	351.237
1700	157.485	307.684	171.026	-80.452	378.236
1800	159.931	316.756	186.899	-79.802	405.199
1900	162.115	325.462	203.004	-79.088	432.125
2000	164.064	333.828	219.314	-78.324	459.011
2100	165.803	341.876	235.809	-77.525	485.858
2200	167.361	349.626	252.469	-76.699	512.667
2300	168.766	357.097	269.277	-75.855	539.437
2400	170.044	364.307	286.218	-74.999	566.171
2500	171.224	371.272	303.282	-74.134	592.868

RESULTS AND DISCUSSION

Jet propulsion-8 was tested at three different temperatures for data prediction. The temperatures were chosen at random; the results were interpolated according to the values provided (Table 2), and the information for these tests is shown in Table 3.

Table 3. Neural network test data.

Test	1	2	3
T (K)	350	1050	1850
Cp (cal·mol ⁻¹ ·K)	63.63	138.62	167.53
S (cal·mol ⁻¹ ·K)	128.96	239.73	327.08
ΔH (kcal·mol ⁻¹)	3.29	78.55	202.91
Hf (kcal·mol ⁻¹)	-64.10	-81.17	-81.39
Gf (kcal·mol ⁻¹)	28.28	227.83	463.88

In the tests for neural networks, four prediction analyses were made, where the learning functions and the activation function of the first layer were analyzed, thus, 20 analyses were performed based on the number of neurons. These analyses followed a criterion, starting with 25 neurons, then 24 neurons, 23 neurons and so on until reaching 6 neurons, where the stop criterion for the analyses was made, since the minimum number of neurons must be equal to the sum of the output data plus one. In this case, the output data are: the specific heat, the entropy, the enthalpy variation, the formation enthalpy and the Gibbs free energy.

These data were obtained through predetermined factors and their performance was based on the calculation of the percentage of local and global errors, always based on the best global error with the fewest possible neurons as a stopping criterion.

For all tests, the same training and error function and the same number of layers were used, in this case, two layers. The training function used was *trainlm*, which refers to the Levenberg–Marquardt algorithm, considered the standard function in the MATLAB toolbox and evaluated by as the fastest method of training feedforward neural networks and with excellent performance (Friderichs *et al.* 2013).

To calculate the global error, the mean squared error (MSE) function was used, which corresponds to the mean square error. Local errors were calculated according to Eq. 1 (calculation of local error), where the interpolated values were adopted as experimental values and the theoretical value obtained by the MATLAB program.

$$Local\ error = \frac{ABS\ (Theoretical\ Value - Experimental\ Value)}{Experimental\ Value} \quad (1)$$

The other parameters were changed, such as the *learngdm* and *learngd* learning function, which correspond to the descending gradient function with and without moment factor and activation function in the first layer. In the second layer, the *purelin* function was maintained, which corresponds to a first-degree function, where these functions aim to guide and approximate the experimental result to the real result.

For the first prediction, the *learngdm* learning function and the activation functions *tansig* and *purelin* (Table 4) were used and the results obtained for the first test are shown in Table 5. Table 6 shows the percentage of local error.

Table 4. First prediction of neural networks.

Training function (algorithm)	Trainlm
Learning function	Learngdm
Error calculation	MSE
Number of layers	2
Activation function (1st layer)	Tansig
Activation function (2nd layer)	Purelin

Table 5. Results of the first forecast.

T (K)	350	1050	1850
Cp (cal·mol ⁻¹ ·K)	63.62	138.24	167.49
S (cal·mol ⁻¹ ·K)	128.23	239.63	327.24
ΔH (kcal·mol ⁻¹)	3.29	78.81	203.08
Hf (kcal·mol ⁻¹)	-64.47	-81.19	-81.39
Gf (kcal·mol ⁻¹)	28.09	228.34	464.10

Table 6. Local errors of the first forecast.

T (K)	350	1050	1850
Cp (cal·mol ⁻¹ ·K)	0.03%	0.28%	0.03%
S (cal·mol ⁻¹ ·K)	0.57%	0.04%	0.05%
ΔH (kcal·mol ⁻¹)	0.19%	0.33%	0.08%
Hf (kcal·mol ⁻¹)	0.58%	0.01%	0.25%
Gf (kcal·mol ⁻¹)	0.68%	0.22%	0.05%

For the second forecast, the learnsgdm learning function was maintained, but the activation function of the first layer was changed to logsig (Table 7), where the objective was to compare results in the exchange of a tangential function for a logarithmic function (Table 8) and the local errors in Table 9.

Table 7. Second forecast of neural networks.

Training function (algorithm)	Trainlm
Learning function	Learnsgdm
Error calculation	MSE
Number of layers	2
Activation function (1st layer)	Logsig
Activation function (2nd layer)	Purelin

Table 8. Results according to forecast.

T (K)	350	1050	1850
Cp (cal·mol ⁻¹ ·K)	63.55	138.63	167.50
S (cal·mol ⁻¹ ·K)	128.80	239.25	326.96
ΔH (kcal·mol ⁻¹)	3.21	78.48	203.02
Hf (kcal·mol ⁻¹)	-64.11	-81.18	-81.37
Gf (kcal·mol ⁻¹)	28.31	227.82	463.85

Table 9. Local errors according to forecast.

T (K)	350	1050	1850
Cp (cal·mol ⁻¹ ·K)	0.14%	0.00%	0.02%
S (cal·mol ⁻¹ ·K)	0.13%	0.20%	0.04%
ΔH (kcal·mol ⁻¹)	2.60%	0.10%	0.05%
Hf (kcal·mol ⁻¹)	0.01%	0.00%	0.23%
Gf (kcal·mol ⁻¹)	0.08%	0.00%	0.01%

When comparing the first forecast with the second forecast, it is noted that the local errors did not have a very large difference in values and, for the first forecast, the global error was 3.72×10^{-5} or, in percentage, 0.00372%, and the number of neurons used was seven. For the second forecast, the global error was 2.04×10^{-5} or 0.00204%, but the number of neurons needed for the best global error was eight. So, it can be said that when changing the activation function of the first layer, there was a decrease in the global error, but one more neuron was necessary for this.

For the third forecast, the learning function was changed to learngd and changes in the activation functions of the first layer were maintained, as in this case the results were observed in the change in the learning function and in the change in the activation function of the first layer (Table 10). The results are shown in Table 11 and the local errors in Table 12.

Table 10. Third forecast neural networks.

Training function (algorithm)	Trainlm
Learning function	Learnrd
Error calculation	MSE
Number of layers	2
Activation function (1st layer)	Tansig
Activation function (2nd layer)	Purelin

Table 11. Third forecast results.

T (K)	350	1050	1850
Cp (cal·mol ⁻¹ ·K)	63.42	138.86	167.33
S (cal·mol ⁻¹ ·K)	128.64	239.75	326.97
ΔH (kcal·mol ⁻¹)	3.26	78.40	203.18
Hf (kcal·mol ⁻¹)	-64.06	-81.30	-81.45
Gf (kcal·mol ⁻¹)	27.90	227.46	463.79

Table 12. Local errors third forecast.

T (K)	350	1050	1850
Cp (cal·mol ⁻¹ ·K)	0.34%	0.17%	0.12%
S (cal·mol ⁻¹ ·K)	0.25%	0.01%	0.03%
ΔH (kcal·mol ⁻¹)	1.17%	0.20%	0.13%
Hf (kcal·mol ⁻¹)	0.06%	0.14%	0.34%
Gf (kcal·mol ⁻¹)	1.35%	0.16%	0.02%

For the fourth and last forecast, the activation function was changed back to logsig (Table 13) and the results and local errors are shown in Table 14 and 15, respectively.

Table 13. Fourth forecast neural networks.

Training function (algorithm)	Trainlm
Learning function	Learngd
Error calculation	MSE
Number of layers	2
Activation function (1st layer)	Logsig
Activation function (2nd layer)	Purelin

Table 14. Fourth forecast results.

T (K)	350	1050	1850
Cp (cal·mol ⁻¹ ·K)	62.55	138.80	167.47
S (cal·mol ⁻¹ ·K)	128.37	239.97	327.17
ΔH (kcal·mol ⁻¹)	3.79	79.01	202.60
Hf (kcal·mol ⁻¹)	-63.42	-81.20	-81.27
Gf (kcal·mol ⁻¹)	28.18	228.60	463.21

Table 15. Local errors fourth forecast.

T (K)	350	1050	1850
Cp (cal·mol ⁻¹ ·K)	1.70%	0.13%	0.03%
S (cal·mol ⁻¹ ·K)	0.46%	0.10%	0.03%
ΔH (kcal·mol ⁻¹)	14.81%	0.58%	0.16%
Hf (kcal·mol ⁻¹)	1.06%	0.03%	0.11%
Gf (kcal·mol ⁻¹)	0.35%	0.34%	0.15%

Comparing the results of the third with the fourth forecast, similarities in the results of local errors are noted again. For the third forecast, the global error was 3.43×10^{-5} or 0.00343% with six neurons used. For the fourth forecast, the global error was 1.80×10^{-5} or 0.00180% with the same six neurons used. It can be seen then that the error was greater when the function of the first layer was changed from tansig to logsig.

CONCLUSION

The results obtained were satisfactory because they meet the stopping criteria with the best overall error in the 10⁻⁵ range and the fewest possible neurons. The number of neurons is very important due to the propagation of error that can exist, since the greater the number of neurons used, the greater the propagation of error; however, the neurons used do not exceed eight. Therefore, comparing the four simulations, an increase in local and global error was observed due to the change in the learning function, therefore, the descending gradient function without the moment factor raises the value of errors. It was also observed that the change in the learning functions of the first layer, for all tests, raised the value of the errors too, so changing a function of

a tangential character to a function of a logarithmic character also interferes with the increase of the error. The best result between the simulations, taking into account the best global error with the fewest possible neurons, was the first prediction.

AUTHORS' CONTRIBUTION

Conceptualization: Rocha FV, Koshun I and Tolosa TAG; **Methodology:** Rocha FV and Tolosa TAG; **Investigation:** Rocha FV, Koshun I and Tolosa TAG; **Writing – Original Draft:** Rocha FV, Koshun I and Tolosa TAG; **Writing – Review and Editing:** Rocha FV; **Supervision:** Koshun I.

DATA AVAILABILITY STATEMENT

All data sets were generated or analyzed in the current study.

FUNDING

Not applicable.

ACKNOWLEDGEMENTS

Not applicable.

REFERENCES

- Aguirre LA (2015) *Introdução à Identificação de Sistemas: Técnicas Lineares e não Lineares Aplicadas a Sistemas*. Minas Gerais: UFMG.
- Arbib MA (1987) *Brains, machines, and mathematics*. New York: Springer.
- Carpenter GA, Grossberg S (2010) Adaptive resonance theory. In: Sammut C, Webb GI, editors. *Encyclopedia of Machine Learning and Data Mining*. Boston: Springer. https://doi.org/10.1007/978-1-4899-7502-7_6-1
- Center USDES (2008) Turbine Fuel, Aviation, Kerosene Type, JP-8 (NATO F-34), NATO F-35, and JP-8+100 (NATO F-37). [S.l.]. [accessed 9 10 2020].
- Coutinho ER, Silva RM; Delgado ARS (2016) Utilização de técnicas de inteligência computacional na predição de dados meteorológicos. *Rev Bras Meteorol* 31(1):24-36. <https://doi.org/10.1590/0102-778620140115>
- Friderichs A, Cunha GA, Weber DP, Castro T, Hoffmann R, Salau NPG (2013) Uso de redes neurais na previsão do desempenho de um reator UASB para tratamento de vinhaça e geração de biogás. Paper presented XV Congresso Brasileiro de Energia (CBE). CBE; Rio de Janeiro, Rio de Janeiro, Brazil. [accessed Sep 18 2020]. http://coral.ufsm.br/cenergia/images/Use_de_redes_neurais_na_previs%C3%A3o_do_desempenho_de_um_reat_or_UASB_para_tratamento_de_vinha%C3%A7a_e_gera%C3%A7%C3%A3o_de_biog%C3%A1s_1.pdf
- Haykin S (2001) *Redes neurais: princípios e prática*. Brussels: Bookman.

Oliveira MA, Siqueira JO (2003) Aplicação do algoritmo de rede neural de aprendizagem recorrente em tempo real (RTRL) para previsão da série do Ibovespa. FEA. São Paulo: EAD/FEA/USP.

Pires APP, Han Y, Kramlich J, Garcia-Perez M (2018) Chemical composition and fuel properties of alternative jet fuels. *BioResources* 13(2):2632-2657. <https://doi.org/10.15376/biores.13.2.2632-2657>

Rodríguez RAM (2011) Teoria e aplicação do classificador baseado em segmentos de reta em problemas de multiclassificação (Master thesis). São Paulo: Universidade de São Paulo. In Portuguese.

Sousa FB (2014) Análise de modelo de Hopfield com topologia de rede complexa (Master thesis). São Carlos: Universidade de São Paulo. In Portuguese.

Vieira M, Araújo AFR (1997) Memória Associativa Multidirecional Temporal para Planejar Trajetória de um Manipulador. Paper presented 3° Simpósio Brasileiro de Automação Inteligente. SBAI; Vitória, Espírito Santo, Brazil.

Xu R, Wang H, Colket M, Edwards M (2015) Thermochemical properties of jet fuels [Internet]. Stanford: Stanford University. [accessed Sep 18 2020]. https://web.stanford.edu/group/haiwanglab/HyChem/approach/Report_Jet_Fuel_Thermochemical_Properties_v6.pdf