18th Brazilian Congress of Thermal Sciences and Engineering
November 16-20, 2020 (Online)

ENC-2020-0475
OPTIMIZATION OF A THERMAL COGENERATION SYSTEM USING HYBRID ALGORITHMS

Bruno Seixas Gomes de Almeida
Manuel Ernani de Carvalho Cruz
Federal University of Rio de Janeiro – UFRJ, Mechanical Engineering Department
brunoseixas@poli.ufrj.br; manuel@mecanica.coppe.ufrj.br

Leonardo dos Santos Reis Vieira
Electric Energy Research Center – CEPEL
lsrv@cepel.br

Abstract. *Optimization has been used in countless works to reduce the cost or increase efficiency in thermal systems. Among the optimization algorithms, hybrid methods stand out, which are a combination of deterministic and heuristic methods, in order to maintain the advantage of each one. In this context, the objective of the present work is to optimize the cost function that describes the total cost of a cogeneration system and also to optimize the exergetic efficiency function using hybrid optimization algorithms. Six optimization algorithms were used combining the methods of Particle Swarm and Differentiated Evolution with Conjugated Gradient, Newton and Quasi-Newton. The methods were implemented in Matlab and a professional Thermal Systems simulator (IPSEpro) was used to solve the thermodynamic problem. The results found for the cost function had a performance compatible with the literature and for the exergetic efficiency the expected results were achieved.*

Keywords: *Optimization, Hybrid Methods, Cogeneration*

1. INTRODUCTION

Usually in thermal system projects, the design engineer is interested in defining the system components, the arrangement of these components, the best process conditions, besides defining other specifications of the thermal system. In order to make this decision, the engineer can formulate an optimization problem to be solved to achieve the desire goals. In this context, the criteria for optimization in thermal systems can be economic (total investment capital, net profit, return on investment etc.), technological (thermodynamic efficiency, reliability, weight etc.) or environmental (rate and emission of pollutants for example) (Bejan, A. et al., 1996).

In general, optimization methods are divided into deterministic and heuristic methods. In deterministic methods the algorithm establishes an iterative process involving a gradient that will converge to the minimum of the objective function. The iterative procedure for this type of method can be written as described in Eq. (1)

$$x^{k+1} = x^k + \alpha^k d^k \tag{1}$$

where x is the vector of variables, α is the step size, d is the direction of descent and k is the number of the iteration. For the deterministic gradient method the best that can be expected is its convergence to a stationary point (local minimum).

In contrast to deterministic methods, the heuristic methods do not use the gradient of the objective function as a direction of descent. As described in Colaço, M. J. et al. (2006) these methods tend to imitate nature, in order to find the minimum of the objective function, by selecting, in an elegant and organized way, the points where this function will be calculated.

Hybrid methods are a combination of deterministic and heuristic methods, in order to maintain the advantage of each one. Normally the algorithm uses a heuristic method to locate the region where the global minimum is found and then a deterministic method to get closer and faster to the minimum point.

There has been an increasing number of publications over the last decade related to hybrid formulations for optimization. Jourdan, L. et al. (2009) present a survey and classification of several approaches found in the literature which make use of hybrid algorithms. Such analysis validates the growing interest that this subject has been arousing in the scientific community.

Countless works have been published over the past few years showing the efficiency and effectiveness of hybrid formulations. Nery R. S. and Rolnik V. (2007) made a comparison between the pure application of a heuristic method (simulated annealing) with two hybrid formulations (simulated annealing with conjugated gradient and simulated annealing with maximum inclination method). It was found that the hybrid methods performed better than the pure simulated annealing method, achieving more accurate responses. This result pointed to the potential of hybrid methods, especially in complex functions.

Practice has shown that hybrid formulations show good results in different applications. Zadeh P. M. et al (2015) used a hybrid method, composed of genetic algorithm and sequential quadratic programming, in a parabolic solar collector system using nanofluids. In the analyzed problem, despite being non-convex, non-linear and computationally expensive, the hybrid method proved to be a good tool to be used in its methodology.

Dominkovic D. F. et al. (2015) applied an optimization using a hybrid method in a trigeneration system, using biomass, and coupled with an underground thermal energy store. The objective function used was the net present value (NPV) of the system. The results using the method were satisfactory, indicating that the energy store is an excellent option for renewable energy projects.

Therefore, due to the advantages of hybrid methods, in this work hybrid formulations will be used to optimize the studied thermal system. The deterministic and heuristic methods used to compose the hybrid methods of the present work are briefly presented below and more details can be found in Colaço, M. J. et al. (2006).

1.1 Deterministics Methods

Conjugated Gradient

The conjugate gradient method improves the convergence rate of the Steepest Descent method, choosing descent directions that are a linear combination of the gradient direction with the descent directions from previous iterations. The Eq. (2) and Eq.(3) indicates the method.

$$x^{k+1} = x^k + \alpha^k d^k \tag{2}$$

$$d^k = -\nabla(x^k) + \gamma^k d^{k-1} \tag{3}$$

where α is the step size and γ is the conjugation coefficient that works by adjusting the size of the vectors. Different versions of the method can be found in the literature. In the Fletcher-Reeves version, the conjugation coefficient is given by Eq. (4).

$$\gamma^k = \frac{\|-\nabla(x^k)\|^2}{\|-\nabla(x^{k-1})\|^2} \tag{4}$$

Newton's method

While the Steepest Descent and Gradient Conjugate methods use information from the first derivative, Newton's method also uses information from the second derivatives to accelerate the convergence of the iterative process. The algorithm used in this method is shown in Eq. (5) and Eq. (6).

$$x^{k+1} = x^k + d^k \tag{5}$$

$$d^k = -[H(x)]^{-1}\nabla \mathrm{U}(x^k) \tag{6}$$

where $H(x)$ is the Hessian of the function. In general, this method requires few iterations to converge, however, it needs a matrix that grows with the dimension of the problem. If the estimate is far from the minimum, the Hessian matrix may be poorly conditioned. In addition, the method involves the inversion of a matrix, which makes the method even more computationally expensive.

BFGS method

It is a type of Quasi-Newton methods. This method seeks to approximate the inverse of the Hessian using information from the function gradient. This approach is made in such a way that it does not involve second derivatives. Thus, this method has a slower convergence rate than Newton's Methods, although it is computationally faster. The algorithm is presented in Eq. (7-12).

$$x^{k+1} = x^k + \alpha^k d^k \tag{7}$$

$$d^k = -H^k \nabla \mathrm{U}(x^k) \tag{8}$$

$$H^k = H^{k-1} + M^{k-1} + N^{k-1} \tag{9}$$

$$M^{k-1} = \left[1 + \frac{(Y^{k-1})^T.H^{k-1}.Y^{k-1}}{(Y^{k-1})^T.d^{k-1}}\right] \frac{d^{k-1}.(d^{k-1})^T}{(d^{k-1})^T.Y^{k-1}} \quad (10)$$

$$N^{k-1} = -\left[\frac{d^{k-1}.(Y^{k-1})^T.H^{k-1} + H^{k-1}.Y^{k-1}(d^{k-1})^T}{(d^{k-1})^T}\right] \quad (11)$$

$$Y^{k-1} = \nabla \mathrm{U}(x^k) - \nabla \mathrm{U}(x^{k-1}) \quad (12)$$

1.2 Heuristics Methods

Differentiated evolution

The differentiated evolution method is based on Darwin's Theory of Evolution of Species. Following the theory, the strongest individuals in a population would be better able to survive under certain environmental conditions. The iterative process of the method is presented in Eq. (13)

$$x_i^{k+1} = \delta_1 x_i^k + \delta_2[\alpha + F(\beta - \gamma)] \quad (13)$$

where x_i is the ith vector of parameters, α, β and γ are three individuals in the population represented by a matrix P, F is the weight function, which influences the mutation ($0.5 < F < 1$), k is the number of iterations and δ_1 and δ_2 are Dirac deltas that define the mutation. In the process of $U(x_i^{k+1}) < U(x_i^k)$, x_i^{k+1} replaces x_i^k in the population of matrix P, otherwise x_i^k is maintained. The binomial crossover (genetic operator) is given by:

$$\delta_1 = \{0, se\ R < CR\ e\ 1\ se\ R > CR\}$$

$$\delta_2 = \{1, se\ R < CR\ e\ 0\ se\ R > CR\}$$

where CR is the factor that defines the crossover ($0.5 < CR < 1$) and R is a random number drawn at each crossover with uniform distribution between 0 and 1.

Particles Swarm

The Particle Swarm method was created in 1995 by an electrical engineer and a social psychologist as an alternative method to the genetic algorithm. This method is based on the social behavior of various species and attempts to balance the individuality and sociability of individuals in order to select the optimal point of interest. The iterative process is presented in Eq. (14) and Eq. (15).

$$x_i^{k+1} = x_i^k + V_i^{k+1} \quad (14)$$

$$V_i^{k+1} = \alpha V_i^k + \beta r_{1i}(p_i - x_i^k) + \beta r_{2i}(p_g - x_i^k) \quad (15)$$

where x_i is the nth vector of parameters, $V_i = 0$ for k = 0, r_{1i} and r_{2i} are random numbers with uniform distribution between 0 and 1, p_i is the best value found for the vector x_i and p_g is the best value found for the entire population, $0 < \alpha < 1$ and $1 < \beta < 2$. In the equation of V_i^{k+1}, the second term represents individuality and the third represents sociability.

2. DESCRIPTION OF THE CGAM SYSTEM

The system studied in this work is a reference system called the CGAM System. This system was developed to compare the solution of the optimization problem with different methodologies, which makes this system a benchmark among researchers (Frangopoulos, C. et al, 1993).This system was chosen because it is sufficient to achieve the objectives of the present work.

The CGAM system is a cogeneration system consisting of an air compressor (AC), a combustion chamber (CC), a gas turbine (GT), an air preheater (APH) and a Heat Recovery Steam Generator (HRSG), which in turn consists of an economizer to preheat the water and an evaporator. The purpose of such a cycle is the generation of 30MW of electrical energy and 14 kg/s of saturated steam at a pressure of 20 bar. Figure 1 illustrates this system.

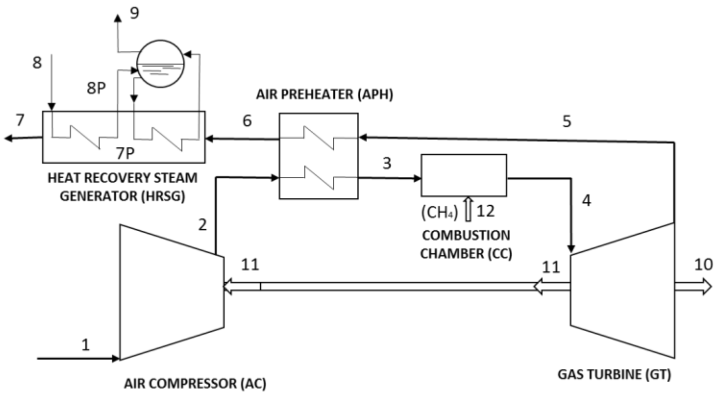

Figure 1. CGAM System

For this system the following approximations are made: Air and combustion gases behave like ideal gases, with constant specific heat; fuel is considered to be pure methane and complete combustion is adopted; all components are adiabatic, except the combustion chamber. In addition, methane has a calorific value of less than 50MJ / kg, the air entering the compressor is at a temperature of 25 ° C and a pressure of 1.013 bar. The original proposition of the thermodynamic model for this system can be found in Frangopoulos, C. et al (1993).

In the present work, the thermodynamic equations of the system were solved by the IPSEpro simulator. Figure 2 illustrates the representation of the CGAM system built in the simulator.

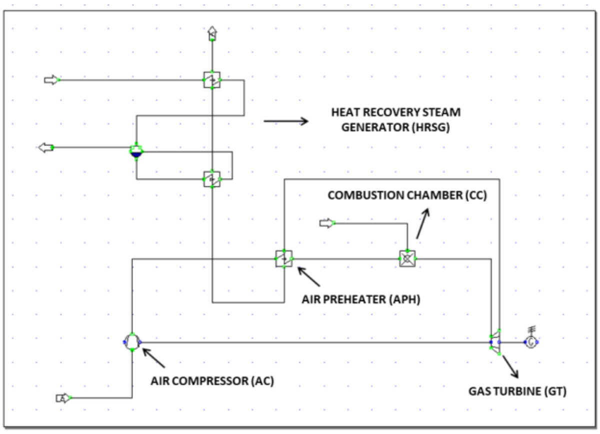

Figure 2. CGAM System built in IPSEpro.

The economic description of the system used in the present work is the same as that adopted in the original work (Frangopoulos, C. et al, 1993) and considers the annual fuel cost and the annual cost associated with the acquisition and operation of each equipment. These equations for Air Compressor (AC), Combustion Chamber (CC), Gas Turbine (GT), Air Preheater (APH) and Heat Recovery Steam Generator are presented in Eq. (16), Eq. (17), Eq. (18) and Eq. (19), respectively

$$Z_{AC} = \left(\frac{C_{11}\dot{m}_a}{C_{12}-\eta_{AC}}\right)\left(\frac{P_2}{P_1}\right) ln\left(\frac{P_2}{P_1}\right) \tag{16}$$

$$Z_{cc} = \left(\frac{C_{21}\dot{m}_a}{C_{22}-\frac{P_4}{P_3}}\right)[1+\exp(C_{23}T_4 - C_{24})] \tag{17}$$

$$Z_{APH} = C_{41}\left(\frac{\dot{m}_g(h_5-h_6)}{(U)(\Delta TLM)}\right)^{0,6} \tag{18}$$

$$Z_{HRSG} = C_{51}\left(\left(\frac{Q_{PH}}{(\Delta TLM)_{PH}}\right)^{0,8} + \left(\frac{Q_{PH}}{(\Delta TLM)_{PH}}\right)^{0,8}\right) + C_{52}\dot{m}_{st} + C_{53}\dot{m}_g{}^{1,2} \tag{19}$$

In the above equations, the variables ΔTLM and Q represent the logarithmic mean temperature difference and the heat transfer rate, respectively. The general expression for the cost rate ($/s) related to the investment of each component is given by Eq. (20).

$$\dot{Z}_{i,invest} = \frac{Z_i \varphi CRF}{N.3600} \tag{20}$$

where *CRF* is the capital recovery factor (18.2%), *N* the number of annual hours of operation of the plant (8000h) and φ is a maintenance factor (1.06). In addition, c_f is the cost of fuel per unit of energy ($ 0.004/MJ). Table 1 indicates the cost constants adopted for each component. Equation (21) represents the total operating cost rate and the equation to be optimized.

$$F = c_f \dot{m}_f PCI + \dot{Z}_{AC} + \dot{Z}_{APH} + \dot{Z}_{CC} + \dot{Z}_{GT} + \dot{Z}_{HRSG} \tag{21}$$

Table 1. Cost constants adopted for each component.

Air Compressor (AC)	$C_{11} = 39.5\ \$/(\frac{\text{kg}}{\text{s}})$ $C_{12} = 0.9$
Combustion Chamber (CC)	$C_{21} = 25.6\ \$/(\frac{\text{kg}}{\text{s}})$ $C_{22} = 0.995$ $C_{23} = 0.018\ (\text{K}^{-1})$ $C_{24} = 26.4$
Gas Turbine (GT)	$C_{31} = 266.3\ \$/(\frac{\text{kg}}{\text{s}})$ $C_{32} = 0.92$ $C_{33} = 0.036\ (\text{K}^{-1})$ $C_{34} = 54.4$
Air Preheater (APH)	$C_{41} = 39.5\ \$/(\text{m}^{1.2})$ $U = 0.018 k\text{W}/(\text{m}^2\text{K})$
Heat Recovery Stem Generator (HRSG)	$C_{51} = 3650\ \$/(\frac{\text{kW}}{\text{K}})^{0.8}$ $C_{52} = 11820\ \$/(\frac{\text{kg}}{\text{s}})$ $C_{53} = 658\ \$/(\frac{\text{kg}}{\text{s}})^{1.2}$

The second equation to be optimized is the equation referring to the overall exergetic efficiency of the system that is given the Eq. (22).

$$\varepsilon = \frac{30MW + \dot{m}_{st} e_{st@20bar}}{\dot{m}_f PCI_{CH_4}} \tag{22}$$

where $\dot{m}_{st}$ is the mass flow of steam, $e_{st@20bar}$ is the specific exergy of saturated steam at 20 bar, $\dot{m}_f$ is the mass flow of fuel and PCI_{CH_4} is the calorific value lower methane. In the numerator, the value of 30 MW refers to the electrical power produced by the system.

3. METHODOLOGY FOR OPTIMIZATION

To solve the problem's thermodynamic equations, the professional process simulator IPSEpro version 6.0 was adopted. IPSEpro is a process simulator used to model and simulate different thermal systems, through its thermodynamic equations. The routines of the optimization methods were written in Matlab® (MathWorks Inc.). In order for the code to be able to collect the results obtained with IPSEpro, it was necessary to use specific syntax to allow integration between both programs. To optimize the objective functions, six optimization routines with hybrid methods were used, combining Heuristic and Deterministic methods according to Tab. 2.

Table 2. Hybrid methods.

	Heuristic	Deterministic
Hybrid 1	Particle Swarm	Conjugate Gradient
Hybrid 2	Particle Swarm	Quasi-Newton
Hybrid 3	Particle Swarm	Newton
Hybrid 4	Differential Evolution	Conjugate Gradient

Hybrid 5	Differential Evolution	Quasi-Newton
Hybrid 6	Differential Evolution	Newton

At the beginning of the optimization methods, the chosen optimization criteria are established, such as the number of individuals in the population and the number of iterations of each method. Once these parameters are determined, the optimization routine starts, the thermodynamic variables are taken to IPSEpro that runs the simulation and provides the the obtained thermodynamic results to Matlab. Then, the objective function is calculated and it is verified if the stop criterion of the algorithm has been reached. If not reached, the program changes the population, according to the method's algorithm and continues the routine. If this criterion is achived, the program ends and provides the point of interest. It is worth noting that in cases where simulation errors in IPSEpro occur for the values of the variables provided, the objective function was penalized so that the solution moved away from these points. Figure 3 illustrates the flowchart of the optimization program.

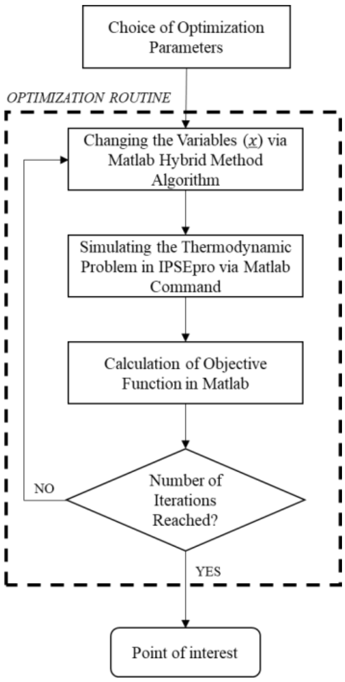

Figure 3. Flowchart of the optimization program.

To solve the optimization problem, the five decision variables adopted in the definition of the original problem are considered: the compression ratio (P_2/P_1), the isentropic efficiency of the compressor (η_{AC}), the isentropic efficiency of the turbine (η_{GT}), the temperature of the air at the exit of the preheater (T_3) and the temperature of the flue gas at the entrance to the turbine (T_4). To find the lowest cost condition, the algorithm minimizes the objective function of the problem given by the total cost rate, indicated in Eq. (21). In order to find the condition of greatest efficiency, the the algorithm maximizes the exergetic efficiency function, given by equation Eq. (22).

In order to carry out the optimization, the limits for the problem variables were established. The limits adopted are shown in Tab. 3 and were used in Padilha, R. S. (2006) and Pires, T. S. (2010). In addition, it was necessary to define the optimization parameters used in each of the optimization methods, as shown in Table 4.

Table 3. Limits of variables for optimization.

Boundaries
$7 \leq P_2/P_1 \leq 27$
$0{,}7 \leq \eta_{AC} \leq 0{,}9$
$0{,}7 \leq \eta_{GT} \leq 0{,}9$
$700 \leq T_3 \leq 1100$
$1100 \leq T_4 \leq 1500$

Table 4. Parameters chosen for optimization.

Parameters	Values
Population of individuals	60
Iterations (Heuristic)	20
Iterations (Determinístic)	20

4. RESULTS

Figure 4 indicates the iterative process of each method to optimize the cost function. Table 5 presents the final results for the decision variables in each method and the value of the Cost Fuction.

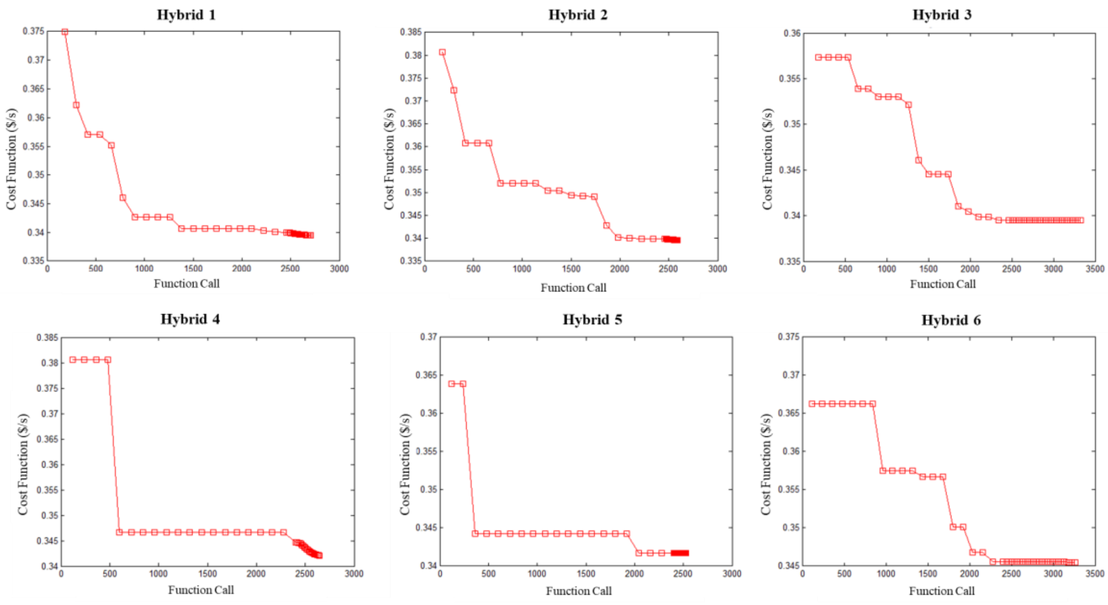

Figure 4. Iterative process of each method for cost function.

Table 5. Results for each hybrid method for cost function.

	Hybrid 1	Hybrid 2	Hybrid 3	Hybrid 4	Hybrid 5	Hybrid 6
P_2/P_1	9.46	9.04	8.29	8.72	9.61	8.90
η_{AC}	0.83	0.83	0.85	0.84	0.83	0.84
T_3	600.43	612.53	606.47	603.90	610.29	614.59
η_{GT}	0.88	0.88	0.88	0.87	0.87	0.85
T_4	1210.95	1212.67	1214.65	1186.64	1195.51	1188.56
Cost Function ($/s)	0.33948	0.33953	0.33949	0.34201	0.34167	0.34551

In order to evaluate the performance of each method, the number of calls from the objective function was verified. As can be seen in Fig. 5, the methods that called the objective function more often were hybrid methods 3 and 6. This performance was already expected, since the Newton method involves calculating the Hessian of a function, which requires calculating the objective function more often.

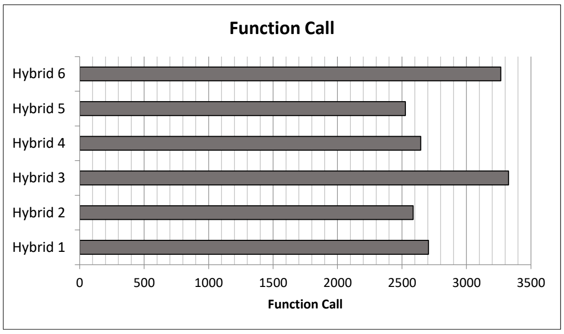

Figure 5. Comparison between the cost function call for each method.

A comparison was made between the results obtained in the present work with those obtained by Frangopoulos C. et al, (1993) and Pires T. S. (2010). It is worth noting that the thermodynamic formulation used by VALERO is slightly different from that built in the simulator, so some differences in relation to the final value of the objective function were already expected. In the work of Pires T. S. (2010), the CGAM system was also built in IPSEpro and the optimization was performed in Matlab® using the following optimization methods: Differentiated Evolution (ED), Particle Swarm (EP), Simulated Annealing (RS), Genetic Algorithm (AG) and Direct Search in Standards (BP). A comparison between the results is shown in Fig. 6.

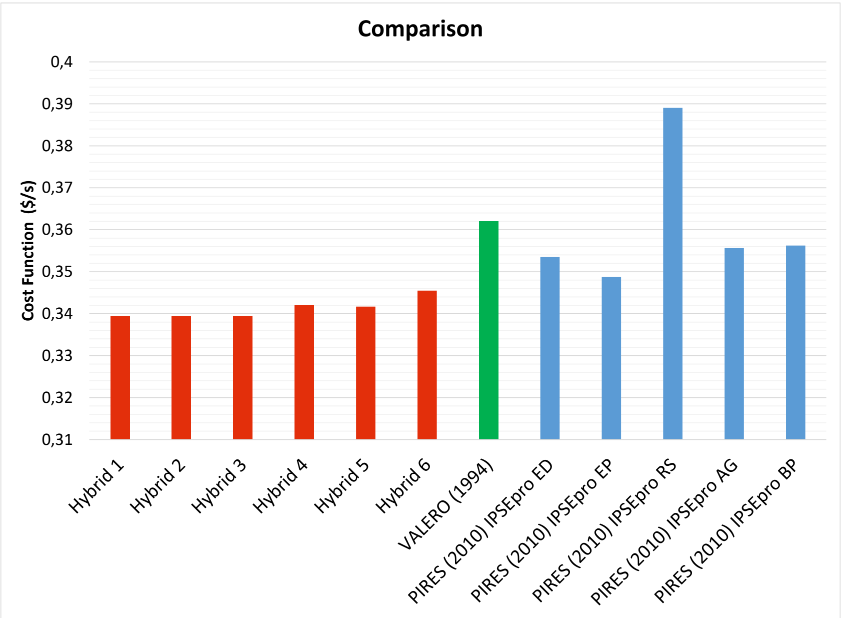

Figure 6. Comparison between the objective function call for each method.

As it is possible to notice, the hybrid methods used in this work have an excellent performance, being the values found compatible with the other references. The same procedure was applied to the exergetic efficiency function. Figure 7 indicates the iterative process of each method to optimize exergetic efficiency function and Tab. 6 presents the final results for the decision variables in each method and the value of the exergetic efficiency.

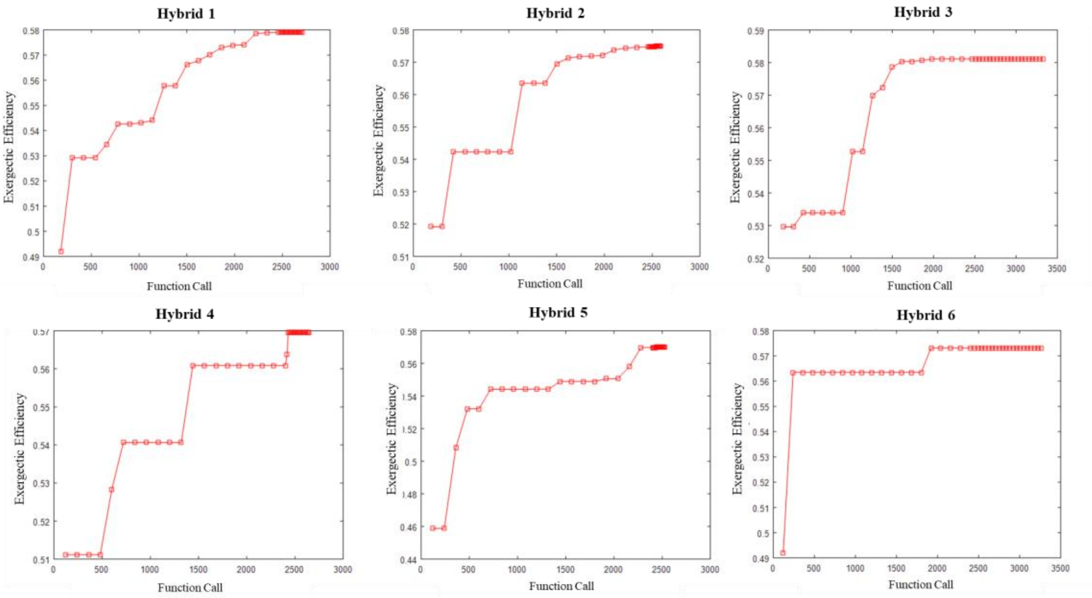

Figure 7. Iterative process of each method for exergetic efficiency.

Table 6. Results for each hybrid method for exergetic efficiency.

	Hybrid 1	Hybrid 2	Hybrid 3	Hybrid 4	Hybrid 5	Hybrid 6
P_2/P_1	12.87	23.06	19.80	16.14	21.76	20.57
η_{AC}	0.89	0.87	0.89	0.89	0.90	0.90
T_3	529.24	540.19	517.71	536.82	539.56	503.02
η_{GT}	0.90	0.90	0.90	0.88	0.88	0.90
T_4	1224.25	1219.96	1222.07	1211.77	1208.23	1186.14
Exergetic Efficiency	0.57910	0.57505	0.58113	0.56952	0.57006	0.57317

The number of calls from the objective function were also checked, as shown in Fig. 8. As in the optimization of the cost function, the methods that called the objective function more often were hybrid methods 3 and 6 whose algorithms involve the calculation of the function hessian.

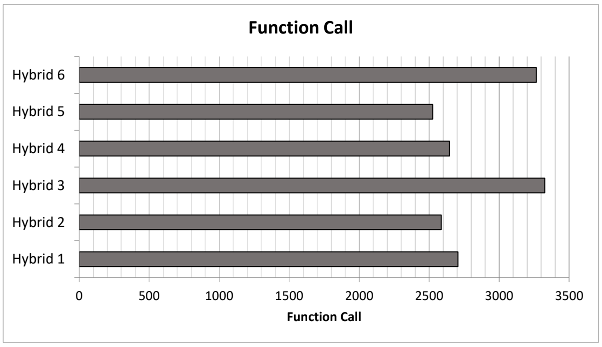

Figure 8. Comparison between the exergetic efficiency function call for each method.

5. CONCLUSIONS

It can be seen that the use of hybrid optimization methods showed satisfactory results for obtaining the conditions of lower cost and greater exergetic efficiency. Comparing the lower cost condition with those used in the literature, it was noticed that the hybrid methods performed better than the others methods and for the exergetic efficiency function the expected results were achieved. Thus, these results consolidated the hybrid formulations used to optimize the objective functions discussed in this work.

6. REFERENCES

Bejan, A., Tsatsaronis, G. and Moran, M., 1996. "Thermal Design & Optimization". *John Wiley & Sons*, Inc., 1th edition.

Colaço, M. J., Orlande, H. R. B. and Dulikravich G. S., 2006. "Inverse and Optimization Problems in Heat Transfer". *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Vol. 28, No.1, pp. 1-24, Brasil, 2006.

Dominkovic, D. F., Cosic, B., Medic, B. et al., 2015. "A hybrid optimization model of biomass trigeneration system combined with pit thermal energy storage", *Energy Conversion and Management*.

Frangopoulos, C., Tsatsaronis, G., Valero, A., et al, 1993. "CGAM Problem: Definition and Conventional Solution", *Energy*, Vol. 19, No. 3, pp. 279-286.

Jourdan, L., Basseur, M. and Talbi, E.G., 2009. "Hybridizing exact methods and metaheuristics: A taxonomy", *European Journal of Operational Research*, Vol. 199, No. 3, pp. 620-629.

Nery, R. S. and Rolnik, V., 2007. "Métodos Híbridos para Otimização global não-linear", *Congresso Nacional de Matemática Aplicada e Computacional,* Florianópolis, SC, Brasil.

Padilha, R. S., 2006; "Otimização de um Sistema de Cogeração de Referência Utilizado Métodos de Gradiente, Evolucionário e Híbrido: Um Estudo Comparativo", MSc. Thesis, COPPE-UFRJ, Rio de Janeiro, Brasil.

Pires, T. S., 2010. "Método de Superfície de Resposta Aplicado à Otimização Termoeconômica de Sistemas de Cogeração Modelados em um Simulador de Processos", MSc. Thesis, *COPPE-UFRJ*, Rio de Janeiro, Brasil.

Zadeh, P. M., Sokhansefat, T., Kasaeian, A. B. et al, 2015. "Hybrid Optimization algorithm for thermal analysis in a solar parabolic trough collector based on nanfluid", *Energy*, Vol. 1-8, 2015.

7. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.