24th ABCM International Congress of Mechanical Engineering
December 3-8, 2017, Curitiba, PR, Brazil

COBEM-2017-1531
USING MULTIOBJECTIVE OPTIMIZATION DESIGN FOR DRIVEN PENDULUM SYSTEM CONTROL

Alisson Ravaglio Santos
Younic Solutions, Rua Anne Frank, 2210 – Hauer, Curitiba, Brazil and
Electrical Engineering Graduate Program (PPGEE)
Federal University of Parana (UFPR), Curitiba, Brazil
alisson@younicsolutions.com.br

Gabriel Herman Bernardim Andrade
Electrical Engineering Graduate Program (PPGEE)
Federal University of Parana (UFPR), Curitiba, Brazil
gabrielhbandrade@gmail.com

Gilberto Reynoso-Meza
Industrial and Systems Engineering Graduate Program (PPGEPS)
Pontifical Catholic University of Parana (PUCPR), Curitiba, Brazil
g.reynosomeza@pucpr.br

Leandro dos Santos Coelho
Electrical Engineering Graduate Program (PPGEE), Federal University of Parana (UFPR), Curitiba, Brazil and
Industrial and Systems Engineering Graduate Program (PPGEPS)
Pontifical Catholic University of Parana (PUCPR), Curitiba, Brazil
leandro.coelho@pucpr.br

Abstract. *Almost all of the optimization problems found in science and engineering are represented by complex mathematical models with multiple parameters that require simultaneous optimization. This eventually leads to the issue of several constraints and conflicting objectives, requiring the use of advanced optimization techniques. Multiobjective optimization is the class of optimization problems which involves more than one objective function to be optimized simultaneously. It increases the problem complexity by requiring not only the computation of the optimal value for each objective but also analyze the tradeoff between each objective best value in order to compute the real global value for the required problem. The main contribution of this paper is to analyze the performance of Multiobjective Optimization Design techniques when applied to the optimization of the input values of a controller. We study the optimization of a multiple parameterized DMC (Dynamic Matrix Control) Controller, a model-based predictive control approach, applied to the control of a pendulum system. The controller parameters are optimized through the application of the sp-MODE (Spherical Pruning Multiobjective Differential Evolution), a Differential Evolution based algorithm which applies concepts of spherical pruning in order to improve diversity in the solution search space. Our approach validates the application of such optimization methods for control applications, since the optimized parameters of the controller worked neatly, making the pendulum system completely stable. The obtained resulting parameters were close to the truly ideal values.*

Keywords: *Multiobjective optimization, pendulum system, dynamic matrix control, m-dimensional visualization, decision making.*

1. INTRODUCTION

Multiobjective Optimization Problems (MOPs) represent realistic statements for many complex optimization instances found in scientific and engineering fields. Usually, this kind of problem is nonlinear and demand the consideration of many conflicting objectives and several constraints at the same time, causing the application of usual optimization methods for a single objective nearly ineffective (Marler and Arora, 2004; Ulungu and Teghem, 1994).

Multiobjective optimization is a field of multiple criteria decision making, that concerns with optimization problems involving more than one objective function to be improved simultaneously (Marler and Arora, 2004). A reasonable solution to a multiobjective problem is to investigate a set of solutions, each of which satisfies the objectives at an acceptable level without being dominated by any other solution option. This set is known as the Pareto front (or frontier), and represent a group of the best possible solutions for the problem. During design optimization, one often needs to consider several design criteria or objective functions simultaneously (Knowles and Nakayama, 2008).

The DMC (Dynamic Matrix Control) (Cutler and Ramaker, 1980) was the first Model Predictive Control (MPC) algorithm and still one of the most researched and popular among this set of advanced control schema. This kind of controllers works by predicting the value of the output in a short time, instead of using the past error between the real output of the system and the desired value. It is especially powerful when applied to control systems that have multiple inputs and multiple outputs (MIMO). Due to the difficulty to find a balanced set of parameters for the controller, the application of Multiobjective Optimization (MOO) algorithms presents itself as a good alternative to classical tuning techniques (Reynoso-Meza, Blasco, Sanchis, and Martinez, 2014).

The main contribution of this paper is to evaluate the performance of a Multiobjective Optimization Design algorithm during the task of optimization of multiple parameters of a DMC Controller, applied to the stabilization of a simple driven pendulum system. This study purposes the demonstration of a distinct technique to adjust controller parameters through the usage of recent evolutionary optimization techniques. We validate this approach through the use of a MOO algorithm already used in control systems: the sp-MODE algorithm (Reynoso-Meza et al., 2010).

The evaluation procedure consists of three parts. After defining the MOP, running the MOO algorithm and retrieve the Pareto front with the best parameter values. Second, evaluate the obtained set of results using TOPSIS (Technique for Order of Preference by Similarity to Ideal Solution) RANK multi-criteria decision analysis tool and last, we plot the obtained data through some visualization methods (Scatter plots and Level Diagrams), for better data comprehension and evaluation. This overall procedure has been identified as a Multiobjective Optimization Design (MOOD) procedure (Reynoso-Meza, Blasco, Sanchis, and Martinez, 2014).

The reminder of this paper is organized as follows. Section 2 presents the components used to perform the study, such as the optimization algorithm, the controller and the problem in which we evaluate our approach. Section 3 discusses the experiments and demonstrates the validity of the optimization technique. Finally, Section 4 concludes the paper.

2. MATERIALS AND METHODS

The steps we took for this research were to run the sp-MODE algorithm in order to get the input values, which represent the input parameters of the DMC Controller, that produces the best output for the pendulum system. Then, we use the TOPSIS decision making tool and some visualization tools to evaluate the multiple sets of combined parameters which makes the Pareto front of best results. This will help to evaluate which of the dominant concurrent points gives us a feasible true best set of controller parameters.

2.1 The Problem

The pendulum system, studied in this paper, is a simple mechanical system that exhibits periodic motion as on Fig. 1 (Raju et al., 2012). It represents a good research opportunity due to its varied applications, and is considered an interesting benchmark for an extensive range of control techniques.

The equation which represents this system on continuous time is Eq. (1) given by

$$\frac{\Theta(s)}{V(s)} = \frac{2.7922}{s^2 + .7191s + 9.9989} \quad (1)$$

Applying a Z-transform in order to obtain the discrete time equivalent equation results in Eq. (2).

$$\frac{\Theta(s)}{V(s)} = \frac{0.0001393z + 0.0001389}{z^2 + 1.992z + 0.9928} \quad (2)$$

From this, it is possible to apply this equation to DMC Controller.

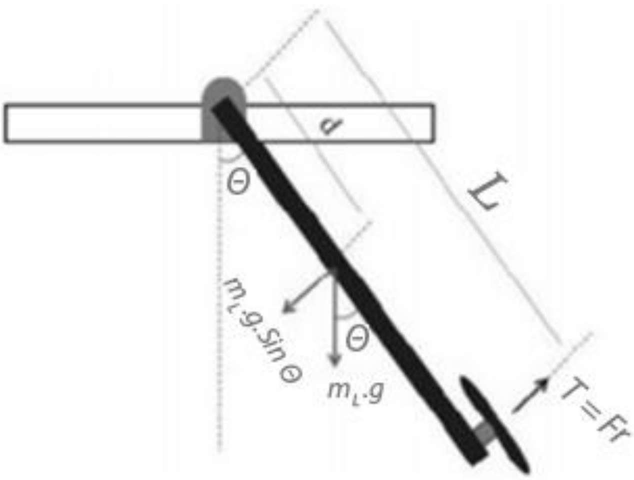

Figure 1. Driven Pendulum System.

2.2 Fundamentals of the Dynamic Matrix Control (DMC) Controller

The DMC (Cutler and Ramaker, 1980) is one of the most popular methods of model predictive control. It is especially powerful when applied to control systems that have multiple inputs and multiple outputs (MIMO). It has a linear response to a step input as a process model. Also, it has an objective quadratic function in a finite prediction horizon. The DMC algorithm models the process output $y_{(t)}$ at a discrete time through a discrete step response $g_{(j)}$ (García et al., 1989; Morari and Lee, 1999). In Fig. 2 it is possible observe the general DMC controller schematic.

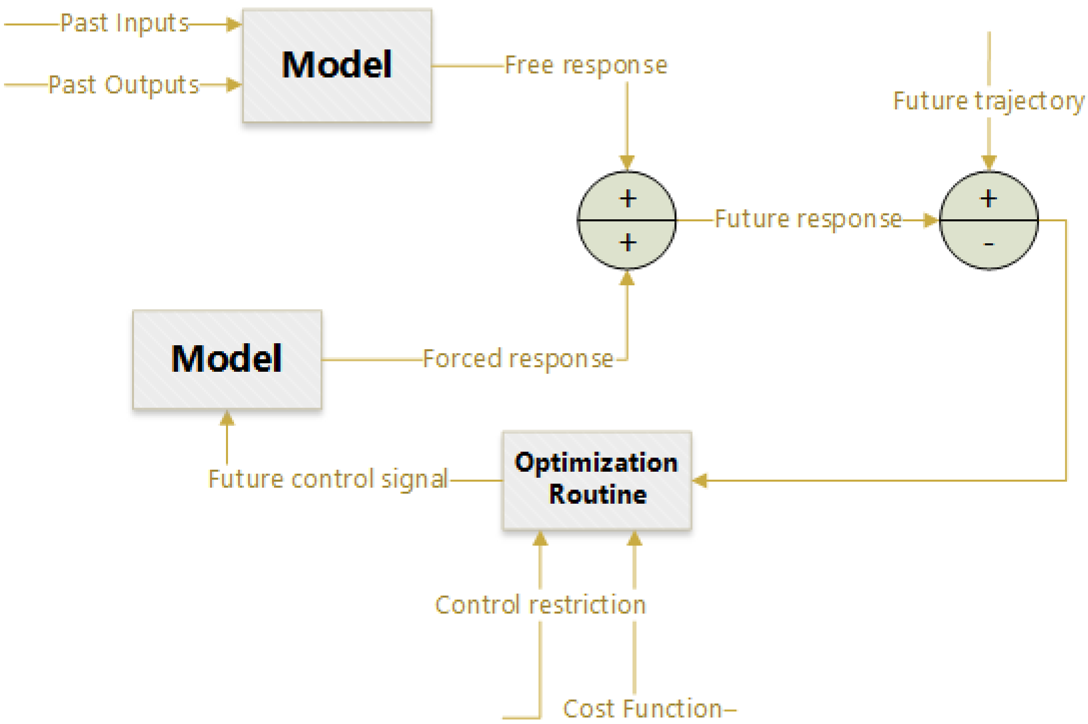

Figure 2. General Schematic for the DMC Controller.

2.3 The Differential Evolution Algorithm

The definition of Evolutionary Algorithms (EA) is not a simple one, but, instead, an imprecise and context-dependent concept. Authors throughout literature spread themselves around many different facets that are covered by this area. EA are mostly stochastic search and optimization heuristics derived from the classic evolution theory, they may be derived from genetics concepts, based on the evolution of populations of possible solution candidates, inspired on the behavior of the nature or the evolution of different species of animals, for example. Which all of this have in common is the fact that the algorithms evolve through the solution over many interactions, called generations (Simon, 2013).

Because of the diversity of EA methods, it is easy to select a method that is especially well suited for given problem, regarding the data types that are to be processed, the representation of the solution and the search space topology. The major advantage of its usage compared to other methods is, that they only need little problem specific knowledge and additional information can be easily incorporated into the method's heuristic.

In this paper we consider the group of EA to be algorithms that consider a population of possible solution candidates and uses mechanisms inspired by biological evolution, such as reproduction, mutation, recombination, and selection to combine these in order to generate new solutions with slightly different characteristics. Then, a natural selection like process determines which individuals of the current population are fit to participate in the new population. Figure 3 shows the basic workflow of this kind of algorithm. Evolutionary algorithms often perform well approximating solutions to all types of problems because they ideally do not make any assumption about the underlying fitness landscape.

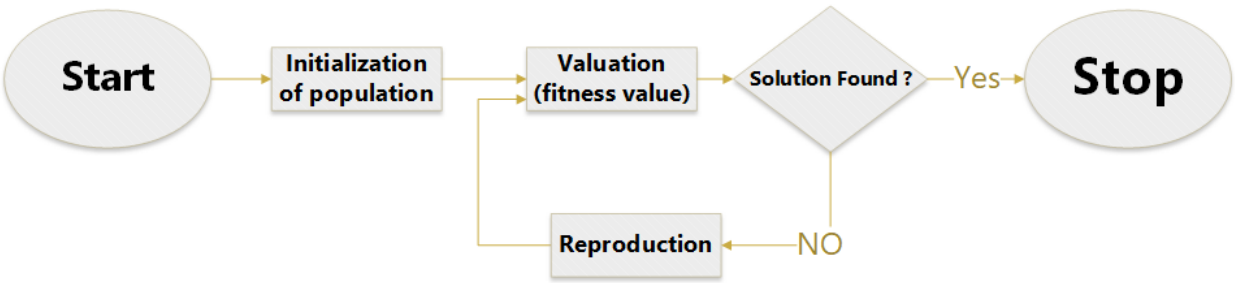

Figure 3. Simple flow chart of an evolutionary algorithm.

The Differential Evolution (DE) (Storn and Price, 1997) method is a subclass of evolutionary algorithms. It is a stochastic, population-based search strategy that uses very few or no assumptions at all about the problem that it is applied to, being classified as a metaheuristic, and can be used to search very large solution spaces.
While it shares many similarities with other evolutionary algorithms, the way it works to generate and evaluate the population at each new iteration is significantly different, as its acquiring of direction and distance information is done though the current population, combining it according to a simple formula. Mutation is applied in order to generate a trial vector, which is used within the crossover operator, generating multiple new candidate solutions. It then keeps whichever candidate solution has the best score or fitness on the optimization problem. Also, mutation step sizes are not sample from a prior known probability distribution, but rather influenced by the difference between individuals of the current population, hence the name.

In this paper we use the sp-MODE algorithm (Reynoso-Meza et al., 2010) as the MOO algorithm chosen to optimize the DMC Controller. The sp-MODE algorithm is based on the DE algorithm, but takes advantage of some modifications in order to find a set of solutions with enough diversity in order to have a useful representation of the whole Pareto front and avoid premature convergence into suboptimal solutions. The algorithm is based on the concept of spherical pruning, in order to make the population of candidate solution widespread throughout the objective space, reducing the cardinality of Θ^*p and making it less sensitive to losing relevant non-dominant solutions in the evolution process due to the unknown geometry of the Pareto front.

The main flow of sp-MODE algorithm with constraints, defined in (Reynoso-Meza et al., 2010), is:

1. Initialize the initial population $P(0)$ with N_p individuals randomly selected from the searching space.
2. Evaluate initial population $P(0)$
3. Look for non-dominated solutions on $P(0)$ to get the initial dominance $D(0)$
4. Perform a spherical pruning for feasibility in $D(0)$ to get initial auxiliary file $A(0)$ and store the solutions.
5. For k =1: **MaxGen** or convergence criterion reached
    a. Select randomly a subpopulation of NS(k) individuals with proposed solutions on P(k) and A(k).
    b. Apply the fundamental DE operators on subpopulation NS(k) to get the offspring O(k):
        i. Perform mutation operator.
        ii. Apply fixing rule for boundary constraint violations if needed.

iii. Perform crossover operator

c. Evaluate offspring *O*(k); if child ≺ parent, the parent is substituted by its child.
d. Apply dominance on *A*(k) U *O*(k) to get *D*(k)
e. Apply spherical pruning for feasibility on *D*(k) to get *A*(k + 1)
f. Store the solution *A*(k + 1)

6. Algorithm terminates. The solution in A with the lower $N_\infty\,(J^{i}_{2:m})$ will be the proposed solution for a single run of the optimization algorithm.

2.4 Evaluation Process

In order to control the pendulum system, the MOO algorithm will try to generate some of the input parameters for the DMC Controller which improves the performance of the system. We evaluate the parameters *N* (Number of terms), *Ny* (Forecast horizon), *Nu* (Control horizon) and λ (Control effort weighting), through the analysis of the output parameters, *ITAE* (Integral of Time multiplied by Absolute Error), represented by equation 3, *IAE* (Integral of Absolute Error), equation 4, and Total Variation (*TV*) which is represented by equation 5. These variables will represent the quality of the DMC controller influence over the system. As this combination of input variables gets improved, the output values should be lowered, until the lowest possible values are reached and the corresponding solution is chosen.

$$ITAE = \sum_{k=1}^{N} k.\,|e(K)| \tag{3}$$

$$IAE = \sum_{k=1}^{N} |e(K)| \tag{4}$$

$$TV = \sum_{k=1}^{N} (Y_k - \bar{Y})^2 \tag{5}$$

Where *e(k)* is the error in kth, Y_k the real output and $\bar{Y}$ the expected output.

To have a good approximation using MOOD, it is necessary to study the input parameters in order to analyze and choose the range in which the algorithms will work, defining the range of the solution search space. Table 1 shows the range for the input parameters of the controller.

Table 1. Range of input values for DMC controller.

Parameter	Range
N	[10, 1000] (ℤ)
Ny	[1, 9] (ℤ)
Nu	[1, 5] (ℤ)
λ	(0,1) (ℝ)

The sp-MODE algorithm, implemented in MATLAB, was adapted to run along the mathematical representation of the pendulum and DMC Controller system. The following tune parameters and initialization values were selected: *NOBJ* = 3 (number of objectives), *NRES* = 0 (number of constraints) and *NVAR*=4 (number of decision variables) for DMC Controller function.

The initial values, in which will represent the condition where the algorithm will start were [10; 6; 1.5; 0.1].
After the initial configuration, the sp-MODE algorithm has been run and generated the matrix with all values found. Then, the decision making and visualization algorithms have been run and the best values were founded, based on *ITAE*, *IAE* e *TV* error, to control de pendulum system.

Since the Pareto front, that is output from the MOOD algorithm, generates a set of solutions, multiple group of input parameters that are able to generate very similar output values for the observed variables, we can make use of multi-criteria decision analysis tools to help evaluate and decide between the multiple conflicting solution sets. In this paper we use the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) Rank algorithm, developed by Hwang and Yoon (1981), which selects the alternative set which gives shortest geometric distance from the positive ideal solution and the longest geometric distance from the negative ideal solution. It is a method of compensatory aggregation that compares a set of alternatives by identifying weights for each criterion, normalizing scores for each criterion and calculating the geometric distance, which is the best score in each criterion. Thus, TOPSIS Ranking was necessary to see in order of a rank, which are the best output values obtained considering the criteria mentioned.

Finally, we plot the data obtained through visualization tools, which are important because allow us to view the data graphically and helps in the decision-making process of the best values of the pareto front and pareto set. Therefore, in this paper we analyze the data using both Scatter plots and Level Diagrams.

Scatter plots shows the data using Cartesian coordinates to display values for a set of data. Normally it is displayed in a two-dimensional space, but it is possible to show in m-dimensional space if the points have different colors. In figure 1, there are plots based on study time *x* grade, and it is possible to see a pattern based on the plots (red line).

Level Diagrams (Blasco et al., 2008) generated the visualization based on the classification of the Pareto front approximation according to the proximity to the ideal point. This way, it is possible to figure out the best choice for an output variable and the best decision space choice.

3. RESULTS AND DISCUSSION

This session will summarize the results for the executed EA. The Fig. 4 represents the obtained Pareto front for the sp-MODE algorithm. It is represented by a three-dimensional graph where the shown points for values of Total Variation, *IAE* and *ITAE* are the set of optimal solutions that are not dominated by any other solution in the search space. This frontier of points represents a group of output values that cannot be improved inside the imposed constraints.

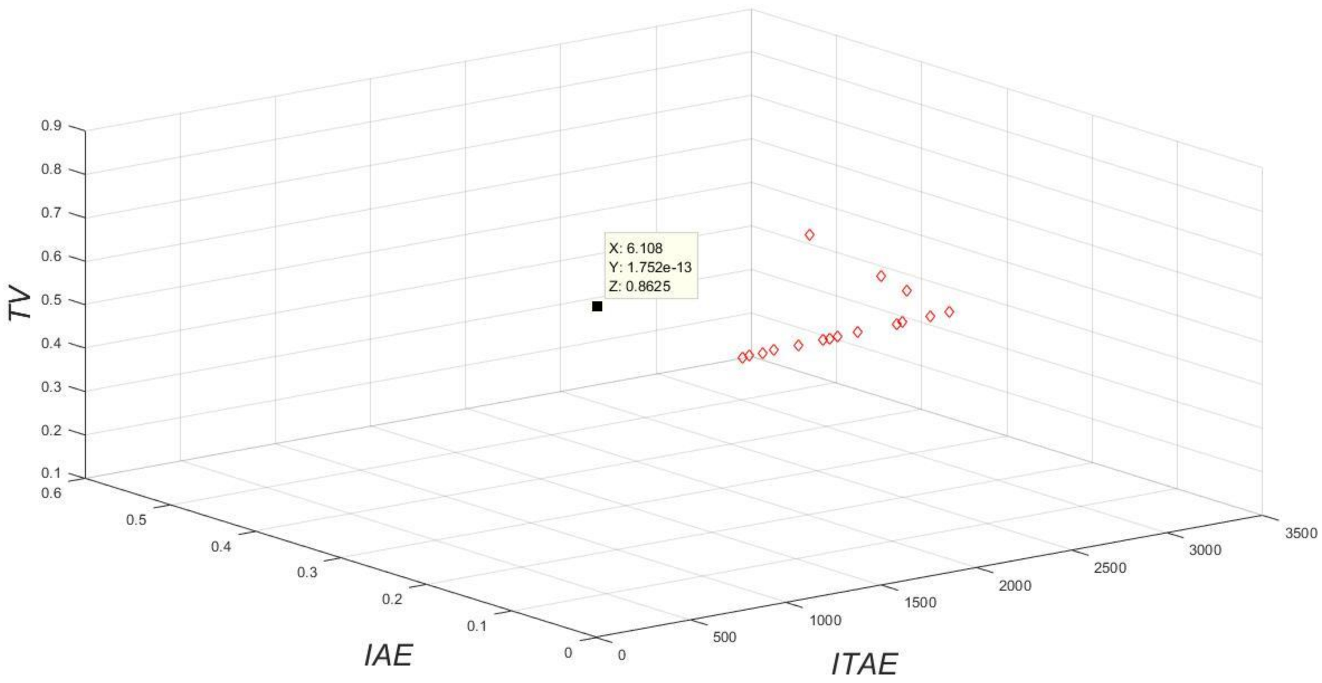

Figure 4. Pareto Front for sp-MODE Algorithm.

In order to acquire the truly best set of Pareto values, decision-making and data visualization techniques were used. Table 2 displays top 10 best values found by TOPSIS Ranking, in order of relevance. The set of values that are shown in the first line represent the candidate to be selected as the best optimization result.

Finally, to have a good visualization from the differential algorithm results, were used the Scatter Plots to visualize the data. Figs. 5 and 6 present the Pareto front's visualization graphic with scatter plots and Level Diagrams. In Pareto front it is possible to compare the three output variables among them.

The variable *J*1 indicates the *ITAE* error, *J*2 indicates *IAE* and *J3* the *TV*. As it is possible to observe, *J*1, *J*2 e *J*3 have approximate from the zero point and it indicates the booth variables have low values being easy to identify the best point just looking at the scatter plot graphic. Furthermore, looking at *J*1 *vs.* *J*3 plot, it is possible to notate the exact point which *J*1 has achieve the best value and how *J1* presented higher values it has to be consider as the main point to be observed.

Level Diagrams technique was the best one to visualize the best points in Pareto Front. In *J*1 it is possible to see, clearly, the isolated point near to zero which corresponds to one in Y axis, and them it is easy to see the other points achieved in *J*2 and *J*3 and then determine which is the best solution for this problem. Besides, it is a good solution to understand the output values behavior.

Table 2. Results obtained with TOPSIS Ranking for the DE algorithm.

Result Number	*ITAE*	*IAE*	*TV*
1	6.107688498	1.74E-13	0.863
9	1866.150783	0.000427039	0.7
8	1860.682495	0.004149273	0.704
10	1942.118752	0.028731651	0.671
11	1989.375388	0.045789406	0.654
12	2019.503087	0.057462327	0.643
7	1802.068801	0.02843949	0.728
13	2085.699947	0.084956054	0.615
14	2128.420848	0.099852241	0.6
15	2157.491454	0.112792255	0.587

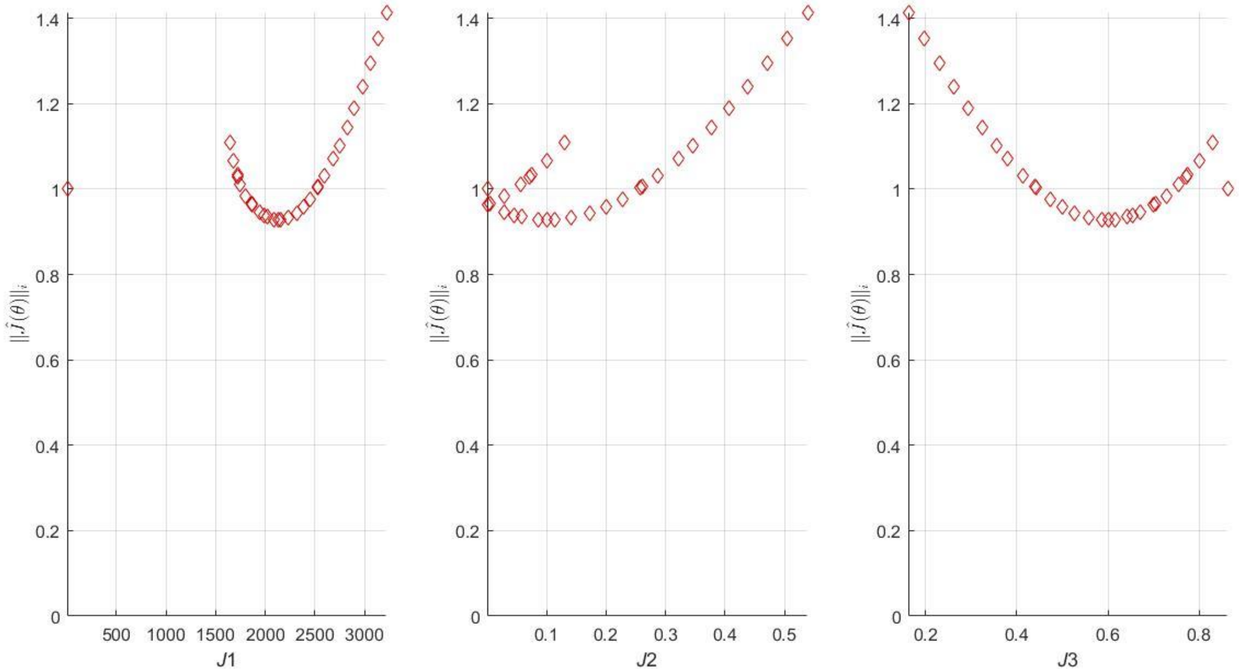

Figure 5. Pareto front visualized with scatter plots.

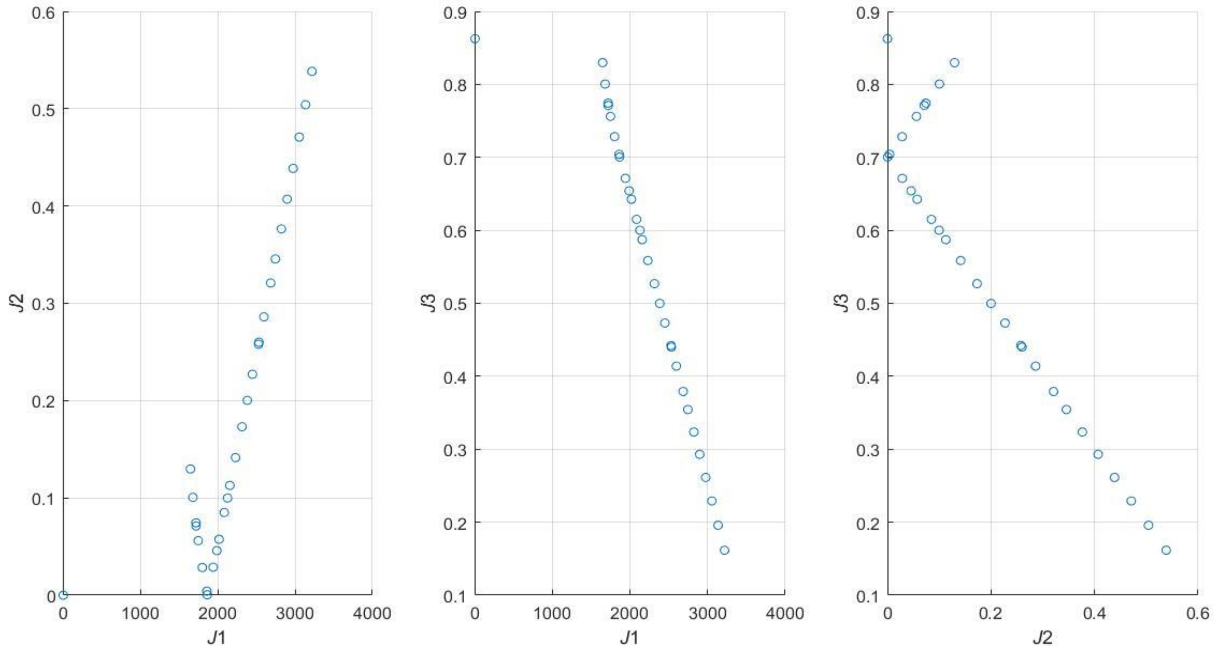

Figure 6. Pareto Front visualized with Level Diagrams.

The evaluation generated after analyzing these results, reinforced the same set of values that were made candidate by TOPSIS Rank as the best solutions. This set of values is shown in Tab 3. In order to achieve the goal of this paper we need to acquire the input parameters that lead us to the best output values found, which are displayed in Tab. 4.

Table 3. Best output values found using the approach of DE.

Output parameter	Value
ITAE	6.1077
IAE	1.7397e-13
TV	0.863

Table 4. Best input values founded using the approach of DE.

Parameter	Value
N	200
Ny	9
Nu	2
λ	0

Figure 7 represents the output signal of the DMC controller using the best values found by the optimization methods. As shown, the output signal of DMC Controller adjusted with the optimized input values is able to make the pendulum system stable, after some stabilization time, which means the appraised parameters for the DMC controller were indeed good values, appropriated for the pendulum system evaluated.

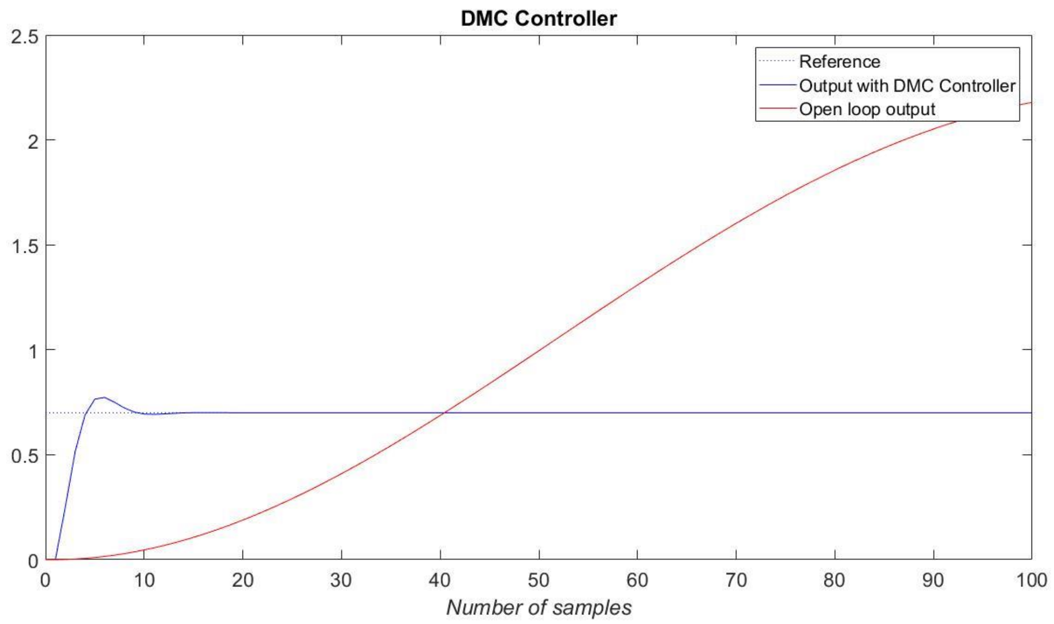

Figure 7. Output signals from DMC controller with the best parameters found.

4. CONCLUSION

In order to determine the best control for Pendulum System using a DMC Controller, we used the sp-MODE algorithm, which is an algorithm for MOO, and the techniques level diagrams, scatter plots and TOPSIS Ranking to visualize the best input attributes and obtain the minimum overall point for *ITAE*, *IAE* and *TV* errors. So, it was possible to reach a good control, with a short time of stabilization and low errors of *ITAE*, *IAE* and *TV*, for pendulum system with results which could be considered near the ideal parameters to control a nonlinear model as the one studied.

Nevertheless, there are a lot of others multiobjective optimization, decision making and visualization techniques which could be used and compared with the methods of this research to reach a good control for the pendulum system. Considering that, it was possible to conclude that MOOD techniques represent a valid procedure to find the best input parameters for a controller, in this case a DMC controller, minimizing overall errors.

In future works, it is possible to use other MOO algorithms mainly to improve the value of *ITAE*, which was the higher error found (6.1077) compared with *IAE* and *TV*. Besides, it is possible to use another visualization tools as parallel coordinates technique and compare TOPSIS ranking with others ranking techniques to help the decision making.

5. ACKNOWLEDGEMENTS

The authors would like to thank National Council of Scientific and Technologic Development of Brazil - CNPq (grants: CNPq/PQ2/304066/2016-8 and CNPq/PQ1D/30309/2015-7) for the financial support of this work.

6. REFERENCES

Blasco, X., Herrero, J. M., Sanchis, J., and Martínez, M. (2008). A new graphical visualization of n-dimensional Pareto front for decision-making in multiobjective optimization. *Information Sciences*. https://doi.org/10.1016/j.ins.2008.06.010

Cutler, C. R., and Ramaker, B. L. (1980). Dynamic matrix control: A computer control algorithm. *Cutler, Ramaker. 1980. Dynamic matrix control??A computer control algorithm. Joint Automatic Control Conference 17: 72.* https://doi.org/10.1109/JACC.1980.4232009

García, C., Prett, D., and Morari, M. (1989). Model predictive control: Theory and practice - a Survey. *Automatica*, *25*(3), 335–348.

Hwang, C.-L., and Yoon, K. (1981). Methods for Multiple Attribute Decision Making. In *Multiple Attribute Decision Making: Methods and Applications A State-of-the-Art Survey*. https://doi.org/10.1007/978-3-642-48318-9_3

Knowles, J., and Nakayama, H. (2008). Meta-Modeling in Multiobjective Optimization. In *Multiobjective Optimization* (p. 245–284).

Marler, R. T., and Arora, J. S. (2004). Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, *26*(6), 369–395. https://doi.org/10.1007/s00158-003-0368-6

Morari, M., and Lee, J. H. (1999). Model predictive control: past, present and future. *Computers and Chemical Engineering*, *23*(4–5), 667–682.

Raju, S. S., Darshan, T. S., and Nagendra, B. (2012). Design of Quadratic Dynamic Matrix Control for Driven Pendulum System, *5*(3), 363–370.

Reynoso-Meza, G., Blasco, X., Sanchis, J., and Martinez, M. (2014). Controller tuning using evolutionary multi-objective optimisation: Current trends and applications. *Control Engineering Practice*.

Reynoso-Meza, G., Blasco, X., Sanchis, J., and Martínez, M. (2010). Multiobjective optimization algorithm for solving constrained single objective problems. In *2010 IEEE World Congress on Computational Intelligence, WCCI 2010 - 2010 IEEE Congress on Evolutionary Computation, CEC 2010*. Barcelona, Spain. https://doi.org/10.1109/CEC.2010.5586408

Simon, D. (2013). Evolutionary Optimization Algorithms. *Soft Computing*. https://doi.org/10.1017/CBO9781107415324.004

Storn, R., and Price, K. (1997). Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. *Journal of Global Optimization*. https://doi.org/10.1023/A:1008202821328

Ulungu, E. L., and Teghem, J. (1994). Multi-objective combinatorial optimization problems: A survey. *Journal of Multi-Criteria Decision Analysis*. https://doi.org/10.1002/mcda.4020030204

7. RESPONSIBILITY NOTICE

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