International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056
Volume: 03 Issue: 07 | July-2016 www.irjet.net p-ISSN: 2395-0072

# Optimal Control of Microgrid Networks Using Gradient Descent and Differential Evolution Methods.

**Abdul Shariq and K. Vaisakh**

[1] *M.Tech Student,Department of Electrical Engineeiring, A.U.C.E(A), Andhra University, Visakhapatnam, A.P.*
[2] *Professor, Department of Electrical Engineering, Andhra University, Visakhapatnam, A.P.*

---

**Abstract -** The given thesis puts forth the application of various methods for optimal control of power flow in a network of microgrids(MGs). The methods used for solving the mathematical formulation of Microgrid control are Steepest Descent method, Newton method and Differential Evolution method. In the above first three come under a group of gradient descent methods where as the differential evolution is an Evolutionary Algorithm. A comparative study is performed by calculating the cost function for each of the three methods. We have taken a case of four Microgrids collaborating in a network. In the proposed work, the optimal control microgrid system with Energy storage systems are considered. The gradient descent methods and evolutionary algorithm are applied and an optimized result for energy storage systems are obtained. It is observed that the results obtained are good and comparable.

***Key Words***: Differential Evolution, Microgrid Network, Gradient Descent Methods, Optimal Control, Newton Methods, Steepest Descent Methods, Optimization, Energy Storage Systems.

## I. INTRODUCTION

Classical gradient methods and evolutionary algorithms represent two very different classes of optimization techniques. In optimization, a problem is typically specified by a set of parameters and an objective function, which is also called a fitness function in the context of evolutionary algorithms. The goal of the optimization process is to find a set of variables such that the objective function is optimum. In the special case of continuous parameter optimization in which all parameters are real valued, Newton developed the gradient method, which is also known as the method of steepest descent. In unimodal functions, the optimum can be found by moving along the local gradients, which leads to the following formulation of the steepest-descent method. It is obvious that steepest-descent algorithms can be applied only to continuously differentiable objective functions. If either the objective function is not continuously differentiable or if the function is not (completely) given due to limited knowledge, which often occurs in real-world applications, the designer has to resort to other methods, such as evolutionary algorithms. Evolutionary algorithms are a class of stochastic optimization and adaptation techniques that are inspired by natural evolution. Each evolutionary algorithm is designed along a different methodology. Despite their differences, all evolutionary algorithms are heuristic population-based search procedures that incorporate random variation and selection.

The optimal control of Microgrid Networks is an active field for research and development. There are several papers and books related to the Microgrids and optimal control concepts. The basic concepts and non-classical gradient methods are learnt from [1]. The example fictional Microgrid network was developed with reference to [15]. [3]-[7] are deal with various views and applications of numerical optimization techniques. [12] is used to study and develop the mathematical formulation of optimal control problems. Differential evolution is one of the population based algorithm. It is a stochastic method. [13]-[14] are the discussions regarding the application of evolutionary algorithms and differential evolution in specific towards the global optimization of optimal control problems. These methods to the best of our knowledge, it has never been applied to control Microgrid Network. The objective is to minimize the cost and energy flow of the Microgrid Network. The problem is solved considering constraints related to each State vector. The rest of this work is organized as follows. Section II presents Steepest Descent method. The Newton method is described in Section III. The Differential Evolution method is discussed in section IV. In Section V, an application on microgrid network problem is presented. The results are Finally, conclusions are drawn in Section VI.

## II. Gradient descent methods

Gradient descent is a first-order optimization algorithm. To find a local minimum of a function using gradient descent, one takes steps proportional to the *negative* of the gradient(or of the approximate gradient) of the function at the current point. If instead one takes steps proportional to the *positive* of the gradient, one approaches a local maximum of that function; the procedure is then known as gradient ascent.

Gradient descent is also known as steepest descent, or the method of steepest descent. Gradient descent should not be confused with the method of steepest descent for approximating integrals.

Gradient Descent Method is a first-order optimization algorithm. To find a local minimum of a function, one takes a step proportional to the negative of the gradient of the function at the current point. Gradient is the slope of a function. Optimization is finding the “best” value of a function which is the minimum value of the function. The number of “turning points” of a function depend on the order of the function. Not all turning points are minima. The least of all the minimum points is called the “global” minimum. Every minimum is a “local” minimum.

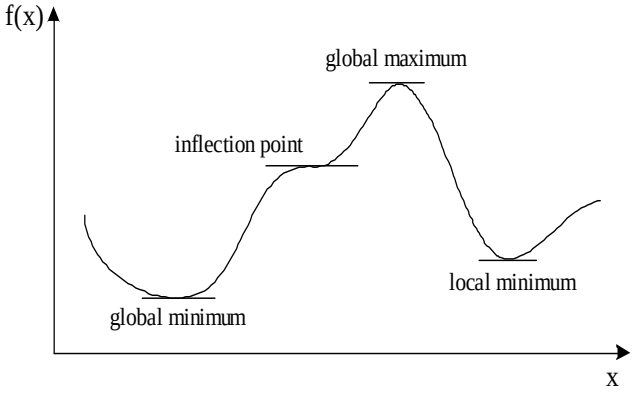


Figure 1 : Global Optimum.

Basic principle is to minimize the N-dimensional function by a series of 1D line-minimizations:

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{p}_k \quad (1)$$

The steepest descent method chooses $p_k$ to be parallel to the gradient

$$\mathbf{p}_k = -\nabla f(\mathbf{x}_k) \quad (2)$$

Step-size $\alpha_k$ is chosen to minimize $f(x_k + \alpha_k p_k)$.

For quadratic forms there is a closed form solution:

$$\alpha_k = \frac{\mathbf{p}_k^T \mathbf{p}_k}{\mathbf{p}_k^T \mathbf{H} \mathbf{p}_k} \quad (3)$$

The gradient is everywhere perpendicular to the contour lines. After each line minimization the new gradient is always *orthogonal* to the previous step direction (true of any line minimization). Consequently, the iterates tend to zig-zag down the valley in a very inefficient manner.

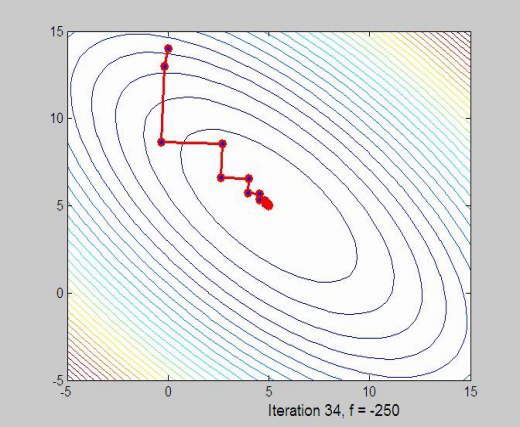


Figure 2 Steepest Descent Paths

### III. Newton method

Newton Method is one of the direct root methods. The necessary condition for f (λ) to have a minimum of λ∗ is that f ′ (λ∗) = 0. The direct root methods seek to find the root (or solution) of the equation, f ′(λ) = 0.

Consider the quadratic approximation of the function f (λ) at λ = λi using the Taylor's series expansion:

$$f(\lambda) = f(\lambda_i) + f'(\lambda_i)(\lambda - \lambda_i) + \frac{1}{2} f''(\lambda_i)(\lambda - \lambda_i)^2 \quad (4)$$

By setting the derivative of above equation equal to zero for the minimum of f (λ), we obtain

$$f'(\lambda) = f'(\lambda_i) + f''(\lambda_i)(\lambda - \lambda_i) = 0 \quad (5)$$

If λi denotes an approximation to the minimum of f (λ), above equation can be rearranged to obtain an improved approximation as

$$\lambda_{i+1} = \lambda_i - \frac{f'(\lambda_i)}{f''(\lambda_i)} \quad (6)$$

Thus the *Newton method*, above equation is equivalent to using a quadratic approximation for the function f (λ) and applying the necessary conditions. The iterative process given by above equation can be assumed to have converged when the derivative, f′(λi+1), is close to zero:

$$|f'(\lambda_{i+1})| \leq \varepsilon \quad (7)$$

where ε is a small quantity. The convergence process of the method is shown graphically in below figure.

Expand $f(\mathbf{x})$ by its Taylor series about the point $\mathbf{x}_k$

$$f(\mathbf{x}_k + \delta\mathbf{x}) \approx f(\mathbf{x}_k) + \mathbf{g}_k^T\delta\mathbf{x} + \frac{1}{2}\delta\mathbf{x}^T\mathbf{H}_k\delta\mathbf{x}$$

where the gradient is the vector

$$\mathbf{g}_k = \nabla f(\mathbf{x}_k) = \left[\frac{\partial f}{x_1} \cdots \frac{\partial f}{x_N}\right]^T \quad (8)$$

and the Hessian is the symmetric matrix

$$\mathbf{H}_k = \mathbf{H}(\mathbf{x}_k) = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_N \partial x_1} & \cdots & \frac{\partial^2 f}{\partial x_N^2} \end{bmatrix} \quad (9)$$

For a minimum we require that $\nabla f(\mathbf{x}) = 0$, and so with solution $\nabla f(\mathbf{x}) = \mathbf{g}_k + \mathbf{H}_k\delta\mathbf{x} = 0$. This gives the iterative update $\delta\mathbf{x} = -\mathbf{H}_k^{-1}\mathbf{g}_k$

If *f*(**x**) is quadratic, then the solution is found in one step.

$$\mathbf{x}_{k+1} = \mathbf{x}_k - \mathbf{H}_k^{-1}\mathbf{g}_k \quad 10$$

The method has quadratic convergence (as in the 1D case). The solution $\delta\mathbf{x} = -\mathbf{H}_k^{-1}\mathbf{g}_k$ is guaranteed to be a downhill direction. Rather than jump straight to the minimum, it is better to perform a line minimization which ensures global convergence $\mathbf{x}_{k+1} = \mathbf{x}_k - \alpha_k\mathbf{H}_k^{-1}\mathbf{g}_k$

General non linear optimization Algorithm :

- Minimize *f*(**x**) subject to: $a_i(\mathbf{x}) = 0$ $c_j(\mathbf{x}) \geq 0$ where the objective function and constraints are nonlinear.
- For a given $\{\mathbf{x}_k, \boldsymbol{\lambda}_k, \boldsymbol{\mu}_k\}$ approximate Lagrangien by Taylor series → QP problem
- Solve QP → descent direction $\{\boldsymbol{\delta}_x, \boldsymbol{\delta}_\lambda, \boldsymbol{\delta}_\mu\}$
- Perform line search in the direction $\boldsymbol{\delta}_x$ → $\mathbf{x}_{k+1}$
- Update Lagrange multipliers → $\{\boldsymbol{\lambda}_{k+1}, \boldsymbol{\mu}_{k+1}\}$
- Repeat from Step 1.

### IV. Differential Evolution (DE)

This method is proposed by Price and Storn in 1995. It is considered as one of the most powerful evolutionary algorithms for real number function optimization nowadays. DE's Main Idea: (DE/rand/1). Generate trial vectors (v) using the following formula:

$$v_i = x_{r1} + F\cdot(x_{r2} - x_{r3})$$

It elegantly replaces the two operations:

1. Crossover
2. Mutation

It has less parameter to be tuned and self-organizing ability. Given an optimization problem, traditional optimization algorithms can be applied to obtain a optimum. However, in the real world, we are often interested in not only a single optimum, but also other possible global and local optima.

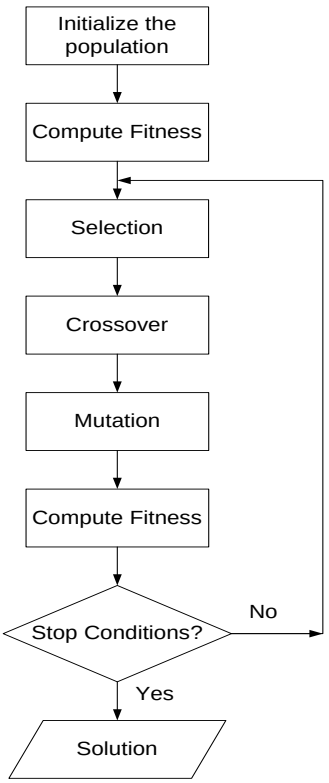


Figure 3 : Flowchart for DE

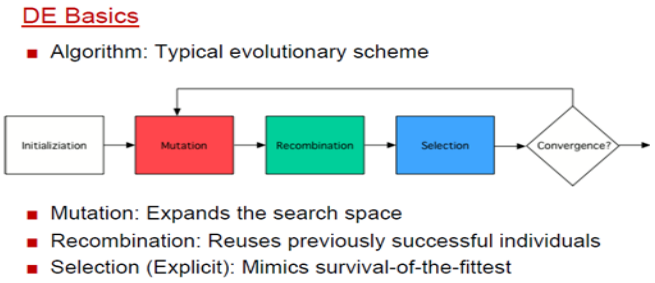


Figure 4 : Algorithm for DE

MUTATION

Each of the N parameter vectors undergoes mutation, recombination and selection. Mutation expands the search space. For a given parameter vector xi,G randomly select three vectors xr1,G, xr2,G and xr3,G such that the indices i, r1, r2 and r3 are distinct. Add the weighted difference of two of the vectors to the third vi,G+1 = xr1,G + F(xr2,G − xr3,G) .

The mutation factor F is a constant from [0, 2] • vi,G+1 is called the donor vector .

Recombination incorporates successful solutions from the previous generation. The trial vector ui,G+1 is developed from the elements of the target vector, xi,G, and the elements of the donor vector, vi,G+1. Elements of the donor vector enter the trial vector with probability CR. The target vector xi,G is compared with the trial vector vi,G+1 and the one with the lowest function value is admitted to the next generation. Mutation, recombination and selection continue until some stopping criterion is reached

The control parameters of Differential Evolution :

Differential Evolution in general has three control parameters.

1) Population Size(NP)
2) Crossover Rate (CR)
3) Mutation Scale Factor(F)

Population size (NP):

Population Size (NP) plays a crucial role in the efficiency and effectiveness of Differential Evolution. Large population size potentially increases the population diversity and helps Differential Evolution to sample more regions, simultaneously. However, when computational budget is limited (which in practice usually is), increasing the population size will decrease the number of iterations (i.e. generations) and may result in early termination. In other words, Differential Evolution may be terminated before the population converges to a desirable point. The smaller population sizes work well. We can have faster and deeper convergence.

Crossover rate (CR):

In discrete recombination or crossover, CR value determines the number of decision variables of each target vector which must be interchanged with the corresponding variables of mutant vector. As a rule of thumb, small CR values can boost convergence speed when a few decision variables are interacting with each others. In turn, large CR values are more effective when lots of decision variables are interacting.

Mutation scale factor (F):

In Differential Evolution, the exploration exploitation balance is controlled by F value. As a rule of thumb, too small F values increase the risk of premature convergence (i.e. converge to an undesirable point), while too large F values decrease the convergence speed that degrades DE efficiency and may result in early termination.

So the solution can be improved by changing the mutation factor F. We can obtain a faster and deeper convergence. It has a good adaptivity.

**PROBLEM FORMULATION**

The microgrid is supposed to be smart, in the sense that all the processes are measurable and the related information can be transmitted to a centralized decision maker in real time. It is assumed that each microgrid contains a local energy storage system (ESS), and can produce RESs power, supplying the consumption of a certain number of households. It is supposed that the instantaneous information on the generated and consumed power, as well as on the stored energy, can be sent in a negligible amount of time toward a central controller. After the acquisition of all the information from all microgrid and the specific computations, the central controller can send the information on the optimal control of the power flows to the control unit of each microgrid. In a microgrid, the ESS is mostly used to compensate the power fluctuations in the local microgrid itself. It is assumed that an ESS is available in each microgrid and that the ESS state operates in connection with other Microgrids. The evolution over time of the energy stored in the microgrids network is supposed to be described, for the i microgrid, by the following continuous state equation:

$$X_i^{'}(t) = \alpha_i X_i(t) + b_i u_i(t) \qquad (11)$$

$$X_i(t_0) = X_{0i}(t)$$

Where

1) $X_i(t)$ is the ESS state at instant t in the $i^{th}$ MICROGRID [kWh], defined as the storage system level with respect to an optimal working level reference value. It is supposed that each element of $X_i(t)$ may assume positive as well as negative values;

2) $\alpha_i$ is a parameter related to the efficiency of the ESS at the $i^{th}$ MICROGRID;

3) $X_{0_i}$ is the ESS state at the initial time $t_0$ ;

4) $b_i$ is a vector which expresses the connections that are established with the $i^{th}$ MICROGRID;

5) $u_i(t)$ [kWh] is the control vector whose dimension is equal to the degree of the power links connected to the $i^{th}$ MICROGRID, which designates the energy exchanged;

The ESS charge at the $i^{th}$ MICROGRID is limited by the following constraint:

$$X_{\min,i} \le X_i(t) \le X_{\max,i} \quad (12)$$

where Xmin,i and Xmax,i are, respectively, the lower and the upper bounds of the state variable at the ith MICROGRID, completely known constant and strongly dependent on the storage technology.

**2.5 Optimal Control Theory on Microgrids :**

Optimal control theory provides a modern, direct, and systematic approach to a large variety of control design problems including constrained optimization with interconnected variables. In this work, the optimal control problem is formulated using the gradient descent method and differential evolution. In gradient descent method we solved in terms of dual variables (Hamiltonian co state and multipliers of constraints) using the Euler–Lagrange approach. Its main innovation is the use and the exchange of information and forecast of power production and consumption on the whole set of MGs, to improve the overall quality of the power management, and energy storage. In addition, the PMP decreases the computational loads as the number of nonlinear second order differential equations increases in a linear manner with the dimension of the state variables.

**C. State Variables Constraints**

While, in the case of state constraints, the mathematical formalization of the optimal control problem must consider the constraints before developing necessary conditions for the optimality. The state constraint that appears in can be converted into two equality constraints. The reason for such a transformation is to consider the variation of the state in the Hamiltonian. The inequality constraint related to the state that is expressed can be written as follows:

$$g_2(t) = S(t) - S_{\min} \quad (13)$$

$$g_1(t) = S_{\max} - S(t) \quad (14)$$

**d. Objective Function**

The problem statement is an optimal control of energy in a network of microgrids. This can be explained as a transfer of energy of the state of the system from any initial state that is different from 0 to a certain value of time that is known. The problem also considers the minimization of energy transfer and storage costs. The aim is to find the optimal control $u^*(t)$ and the state $X_i^*(t)$ [where (*) designates the optimal value] that minimize the performance measure (cost function), assuming that there is a perfect knowledge of the state of each ESS, and under a cooperative strategy among the MGs. The cooperative strategy aims to maintain an optimal level of energy in the distributed ESS, as well as to achieve a low power flow among the MGs. In the following, a vector/matrix notation for the state ($X(t) \in R^I$) and control $U(t) \in R^W$ variables are assumed, with the same meaning of the scalar notation as defined before. The objective function is expressed as follows:

$$F(x,u,t) = \int_0^{t_f} [\sum_{i=0}^{n} (S(t)' M_{t_f} S(t)) + u(t)' N u(t)]dt \quad (15)$$

*M, M* > 0, is a *I* × *I* matrix, related to the cost of an exceeding/lacking quantity of energy stored in each energy storage device;

*N* is a *W* ×*W* matrix, *N* > 0, related to the cost of the power sent on each edge of the network.

The Microgrid system optimal control equations are as follows:

$$\dot{X}(t) = A^* X(t) + B^* U(t)$$

As discussed X is ESS

Here since I = 4,

$$X(t) = \begin{matrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \end{matrix} \quad (16)$$

A is efficiency matrix of each microgrid. Let us consider each microgrid has 100%efficiency , so A would turn out to be an identity matrix of order 4.

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (17)$$

The state equations of the microgrid system is as follows :

$$\dot{x}_1(t) = x_1(t) - u(t) \quad (18)$$

$$\dot{x}_2(t) = x_2(t) + 2u(t) \quad (19)$$

$$\dot{x}_3(t) = x_3(t) + u(t) \quad (20)$$

$$\dot{x}_4(t) = x_4(t) + 2u(t) \quad (21)$$

The cost function of the network of microgrids is

$$f(x,u,t) = \int_0^t (x_1(t)^2 + x_2(t)^2 + x_3(t)^2 + x_4(t)^2 + 4u(t)^2)dt \quad (22)$$

**RESULTS**

**Steepest Descent Method**

The optimum cost function is obtained and the value is 0.8196p.u.

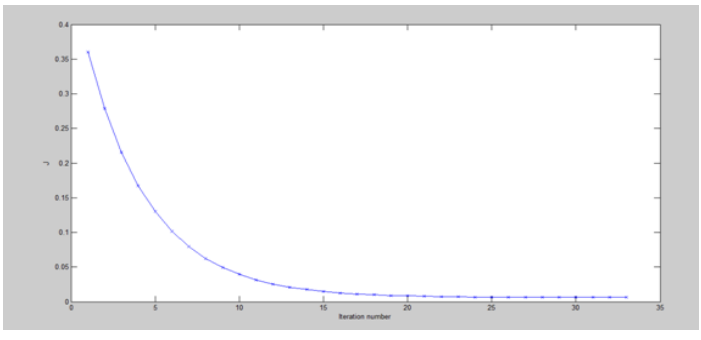

Figure 5 : Cost Function of SD

The above plot shows the cost function versus number of iterations.

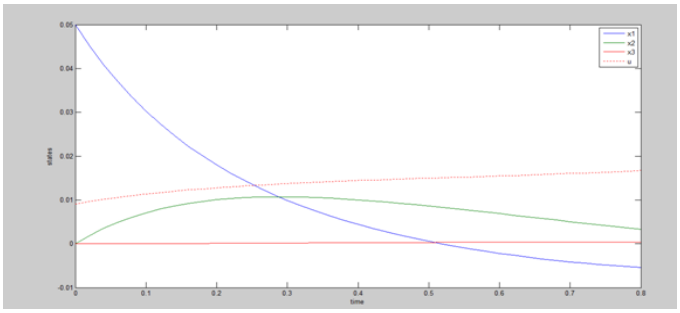

Figure 6 : State functions of SD method.

The above plot shows the variation of state variables during time of execution.

Differential Evolution :

The optimum cost function is obtained and the value is 0.7923p.u

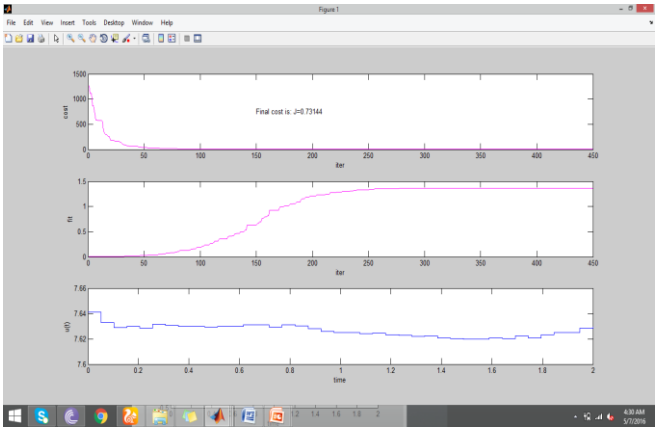

Figure 7 Cost function using DE

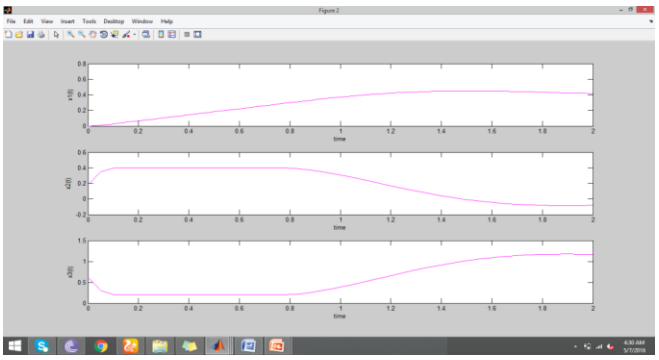

Figure 8 State functions using DE

**Conclusion**

Comparing the final cost functions after executing differential evolution and steepest descent method, we observe that differential evolution obtains a better optimum result. The problem is solved using the continuous

mathematical formalization of the optimal control based on both deterministic and stochastic methods. The wish is to open a new modeling approach also for recent evolutions of traditional approaches. An interesting development of the proposed approach should be devoted to the application of the control approach using a real case study and real data. This approach can be further extended to renewable energy turbines like wind and solar turbines.

**REFERENCES**

[1] Rao, S. S. Engineering optimization : theory and practice / Singiresu S. Rao.–4th ed.

[2] J. Ortega and W. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Academic Press, New York, 1970.

[3] R. Horst, H. Tuy, Global Optimization, Deterministic Approaches, Springer-Verlag, Berlin, 1990.

[4] J. Nocedal and S.J. Wright. *Numerical Optimization. Second edition*. Springer-Verlag, 2006.

[5] E. Polak. *Computational Methods in Optimization*. Academic Press, New York, N.Y., 1971.

[6] B.D.O. Anderson and J.B.Moore. *Optimal Control: Linear Quadratic Methods*. Prentice Hall, 1990.

[7] J.K. Hale. *Ordinary Differential Equations*. Wiley-Interscience, 1969.

[8] H.K. Khalil. *Nonlinear Systems*. Prentice Hall, 2002. Third Edition.

[9] I. Androulakis, C.D. Maranas, C.A. Floudas, $\alpha$BB: A global optimization method for general constrained non-convexproblems, J. Global Optim. 7 (1995) 337–363.

[10] M. Locatelli, Convergence of simulated annealing algorithmfor continuous global optimization, J. Global Optim. 18 (3)(2000) 219–233.

[11] M.D. Canon, JR. C.D. Cullum, and E. Polak. *Theory of Optimal Control an Mathematical Programming*. McGRAW-HILL BOOK COMPANY, New York, 1970.

[12] F.H. Clarke. *Optimization and Nonsmooth Analysis*. Wiley Interscience, 1983.

[13] R. Storn, Differential Evolution—a simple and efficient heuristic for global optimization over continuous spaces, J.Global Optim. 11 (4) (1997) 341–359.

[14] D. Corne, M. Dorigo, F. Glover (Eds.), New Ideas in Optimization, McGraw-Hill, New York, 1999.

[15] Optimal control of a network of power microgrids using the pontryagin's minimum principle, hanane dagdougui ; department of informatics, bioengineering, robotics and systems engineering, university of genoa, genoa, italy ; ahmed ouammi ; roberto sacile