

PREDICTION ERROR'S MINIMIZATION THROUGH A CONTROLLER BASED ON A METAHEURISTIC ALGORITHM

Mînză Viorel

Control and Electrical Engineering Department, "Dunarea de Jos" University of Galati, Romania

Abstract: A Metaheuristic Algorithm (MA) can be a realistic method to solve a given Optimal Control Problem (OCP), but the result is an open-loop solution. If the Metaheuristic Algorithm is integrated within the Model Predictive Control (MPC) structure, a closed-loop solution can be achieved. The controller works using a prediction technique and prediction error's minimization. On the other side, the MA optimizes (minimizes or maximizes) the OCP's objective function. The controller is faced with two optimization tasks. This paper proves through theoretical analysis and simulations that the prediction error's minimization is implicitly accomplished.

Keywords: Metaheuristic Algorithm, optimal control problem, Model Predictive Control, Evolutionary Algorithm.

1. INTRODUCTION

A closed-loop control structure can be achieved using the Receding Horizon Control (RHC). This one uses a process model (MP) and organizes the moving of the prediction horizon. A particular case of the RHC is the well-known Model Predictive Control (MPC) that makes at each step a specific action: *the minimization of the prediction errors*.

Many works address the MPC from different points of view such as: Theoretical works - (Clarke, 1994; Hiskens, and Gong, 2006; Zheng, 2010), tutorial reviews - (Christofides, 2013), or surveys of industrial applications - (Qin, and Badgwell, 2003; Yang, *et al.*, 2014; Lopez-Franco, 2018).

The MPC structure has a controller that can integrate a Metaheuristic Algorithm (MA) to solve a given Optimal Control Problem (OCP). On the one side, usually, the controller has to minimize *the prediction errors*. On the other side, the MA has to optimize the performance index specific to the OCP under consideration. The objective of this paper is to show that, in this case -MPC with MA- the minimization of the prediction error is no longer

necessary. Section 2 outlines the proof that the optimization of the performance index is equivalent to the prediction error's minimization. A case study is presented in section 3, which address an OCP whose theoretical solution is already known. A version of the Evolutionary Algorithm (EA) is used to solve the problem in section 3.2. The result is an open-loop solution. That is why section 3.3 constructs a closed-loop solution using the MPC structure. A quantitative comparison between these solutions is presented in section 4, and some conclusions are drawn in section 5.

2. MINIMIZATION OF THE PREDICTION ERROR USING METAHEURISTIC ALGORITHMS

In the sequel, let us consider $t_0=0$ and the discrete moments $t_k=k \cdot T$ will be specified simply by k . The control horizon is the interval $[0, H]$. We recall the following notations:

- $[k, k + h]$ is the prediction horizon, with $h < H$,
 $k = 0, 1, \dots, H-h$

- $U(k+i/k)$, $i=0, \dots, h-1$ is the predicted value for $U(k+i)$ based on knowledge up to moment k .

- $X(k+i/k)$, $i=1, \dots, h$ is the predicted value for $X(k+i)$ based on knowledge up to moment k ;

Note that

$$(1) U(k+i/k) \neq U(k+i), \quad k > 0, i=1, \dots, h-1,$$

because $U(k+i/k)$ is the future value of the control input predicted at the present moment, whereas $U(k+i)$ is the future real value of the control input at the moment $k+i$, which is not known at the present moment. The same thing can be asserted for the state variables.

Fig. 1 suggests how MPC achieves the predictive control technique. The control input is calculated through the minimization of a performance index $I(U, k)$ depending on the prediction error. At present moment k , when the state variable is $X(k)$, the performance index $I(U, k)$ for the interval $[k, k+h]$ is minimized subject to constraints through an optimal control sequence

$$(2) \langle u(k/k), \dots, u(k+h-1/k) \rangle.$$

The first element $U(k) = u(k/k)$ of this sequence is applied to the system. Then, the horizon is shifted by one sample, and the minimization is restarted for the interval $[k+1, k+h+1]$. The minimization is made within the Model Predictive Controller.

The MA is conceived to solve a specific OCP having a characteristic objective function, $J(X(k), k)$. The main aspect of our implementation is the integration of the MA under consideration into the Model Predictive Controller. Using the PM based on knowledge up to time k , the MA tries to optimize the objective function $J(X(k), k)$ on the control horizon $[k, k+h]$ starting from the initial state $X(k)$. Therefore, the MA yields the best control sequence that it can find during the current sampling period. This sequence is, in fact, the predicted control sequence (pcs).

$$(3) pcs = \langle U(k/k), \dots, U(N-1/k) \rangle.$$

In the sequel, it will be shown that pcs can replace sequence (2). In other words, the minimization of the performance index $I(U, k)$ is no longer necessary. A special definition of the performance index will determine the equivalence between pcs and control sequence (2).

Let us note that the optimal process of the considered environment is generally unknown. However, for this discussion, we introduce the optimal control sequence (ocs) and optimal states sequence (oss) expressed as

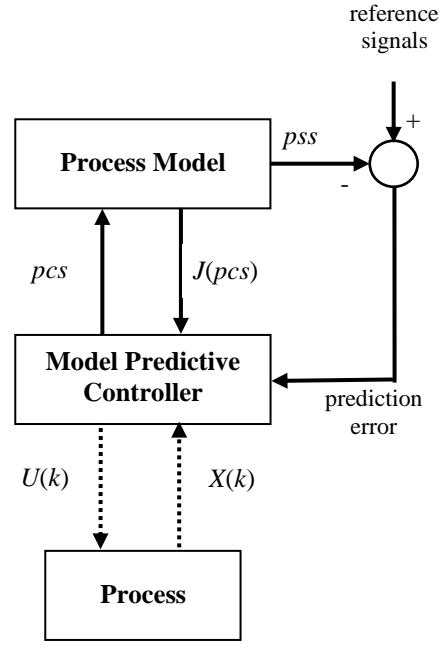


Fig.1. Model Predictive Control structure

$$(4) ocs = \langle U^*(k+1/k), \dots, U^*(k+h/k) \rangle.$$

$$(5) oss = \langle X^*(k+1/k), \dots, X^*(k+h/k) \rangle.$$

Because MA has a stochastic character, pcs is generally quasi-optimal, i.e., it is near but different from the ocs .

Remark 1: Being an MA, it generates the control inputs randomly. While MA generates the pcs , it also calculates the corresponding state variables' values by the integration of the PM. Obviously, the state variables depend on the control inputs. Then, the pcs and the state variables' values are used together to calculate the objective function J .

The predicted state sequence (pss) arranges these values in increasing order of time:

$$(6) pss = \langle X(k+1/k), \dots, X(k+h/k) \rangle.$$

For the same reason, pss is generally quasi-optimal, i.e., it is near but different from the oss . If oss is known, it would be the ideal reference for the state trajectory of the dynamic system. That is why we may consider (only theoretically) oss as the reference signal for the current moment k :

$$(7) R(k+i/k) = X^*(k+i/k), \quad i=1, \dots, h.$$

Therefore, the prediction error along the prediction horizon will be

$$(8) e(k+i/k) = X^*(k+i/k) - X(k+i/k), \quad i=1, \dots, h.$$

In the spirit of MPC structure, we may consider the following performance index:

$$(9) I(U, k) = \sum_{i=1}^h e(k+i|k)^2 = \sum_{i=1}^h \left(X^*(k+i|k) - X(k+i|k) \right)^2,$$

that has to be minimized

$$(10) \min_U I(U, k)$$

Even if the control inputs are not present in equation (9), the state variables are implicitly dependent on the control inputs (see Remark 1), and therefore the minimization (10) makes sense.

A crucial aspect is that the MA produces the *pcs* as a quasi-optimal solution on the prediction horizon. MA, which is a stochastic algorithm, yields a single solution after a single execution. Therefore, it generates a single realization of a stochastic process. Despite this fact, if the convergence speed is good, the unique realization of the stochastic process is close to the optimal process. We may consider that the chosen metaheuristic solves the OCP efficiently, especially the good convergence speed of the algorithm has already been proven and tested. Hence, the *pcs* is very close to the *ocs*, and the *pss* is very close to the *oss*. Under these conditions, the *pcs* predicted by the MA will generate a *pss* implicitly very close to the optimal trajectory of the dynamic system. The value of the performance index (9) will be very close to zero, and there is no need to further minimization in (10).

Of course, this statement is a general qualitative argument for the fact that the objective function (J)'s optimization can replace the performance index minimization. The analysis by simulation of some concrete OCPs will show the veracity of this argument.

3. CASE STUDY

In this section, we consider a specific OCP whose solution is theoretically known, in order to prove that an MPC structure can be implemented using an MA, avoiding the minimization of the prediction error. This approach is possible because of the equivalence, stated in the previous section, between the control input minimizing the prediction error and that which optimizes the performance index.

In section 3.1, the theoretical solution is computed to serve as a comparative element, when we shall analyze the solution given by the MA or the MPC structure.

This OCP has also been solved using an MA. Section 3.2 describes an Evolutionary Algorithm (EA) finding another solution that is very close to the theoretical one. The two solutions, one optimal and the other quasi-optimal are unfortunately open-loop solutions. For a real implementation, only closed-loop solutions can be accepted.

Section 3.3 describes an implementation of the MPC structure using the same EA slightly modified. The yielded closed-loop solution will be compared to the previous two solutions. Its quasi-optimal character proves that MPC can provide a realistic implementation.

3.1 Finite-horizon, discrete-time Linear-quadratic Problem

This section recalls a well-known OCP whose solution is also very well known: finite-horizon, discrete-time Linear-quadratic Problem (LQP). The paper (Michalewicz, *et al.*, 1992) presents an instance of this problem, which is recalled hereafter. The discrete system has the following state equation, where x_0 is given:

$$(11) x_{k+1} = a \cdot x_k + b \cdot u_k, \quad k = 0, 1, \dots, N-1.$$

The objective function and the optimum criterion are respectively:

$$(12) J = q \cdot x_N^2 + \sum_{k=0}^{N-1} (s \cdot x_k^2 + r \cdot u_k^2)$$

$$(13) J^* = \min_{u_k, k=0, 1, \dots, N-1} J$$

The constant values characterizing our problem's instance are:

$$a=1, b=1, q=1, r=10, s=1, x_0=1, N=45.$$

The solution of this OCP consists of the following sequence:

$$(14) u_k, k=0, 1, \dots, N-1,$$

which meets equations (11) and (13).

The theoretical solution states that the value of the performance index (12) subject to (11) is

$$(15) J_{star} = K_0 \cdot x_0^2,$$

where K_k is the solution of the next Riccati equation.

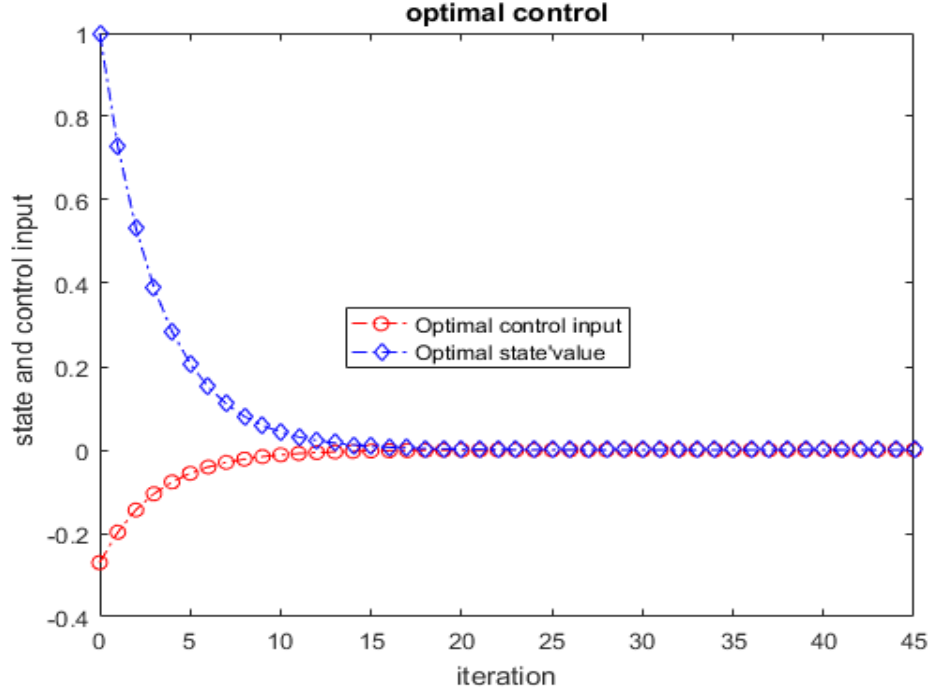


Fig.2. System evolution using theoretical optimal control

$$(16) K_k = s + ra^2 \frac{K_{k+1}}{r + b^2 K_{k+1}}, K_N = q.$$

The optimal control values are computed using the equations below:

$$(17) u_k^* = -F_k \cdot x_k$$

$$(18) F_k = a \cdot b \cdot \frac{K_{k+1}}{r + b^2 \cdot K_{k+1}}$$

Firstly, the values K_k are found iteratively backwards in time using equation (16). Afterwards, the optimal control inputs and state variables are computed using equations (11), (17) and (18). These values are depicted in Fig. 2. Using (15), the performance index is

$$(19) J_{\text{star}} = 3.702$$

3.2 Quasi-optimal solution using an Evolutionary Algorithm

This paper addresses OCPs for which theoretical solutions do not exist considering different reasons. We have emphasized why we have chosen to solve LQP. Our main objective is to test the ability of metaheuristic algorithms and MPC structure to solve difficult problems. The LQP was also solved using a version of the EA (Kruse, *et al.*, 2016; Talbi, 2009) described in the sequel, called LQP-EA. For LQP, every chromosome of the solutions'

population encodes the sequence U_k , $k = 0, 1, \dots, N-1$. So, the length of the solution vector is $h=N$ (control horizon). LQP-EA uses a direct encoding with real values and has some usual characteristics listed below:

- The population of each generation has μ individuals;
- The offspring population has λ individuals ($\lambda < \mu$);
- NGen is the number of generations in which the population is evolving;
- The selection strategy is based on Stochastic Universal Sampling using the rank of individuals, which is scaled linearly using selection pressure;
- Linear Blend Alpha Crossover operator, with $\alpha = 0.4$;
- The mutation operator uses the global variance adaptation of the mutation step. The adaptation is made according to the "1/5 success rule".
- The replacement strategy: the offspring replace the λ worst parents of the generation.

Table 1: The main parameters of LQP-EA

λ	μ	NGen	h	s
30	50	300	45	1.8

The parameters of LQP-EA are given in Table 1.

A typical result is depicted in Fig. 3 and 4 that present the control input and state evolution, the red and blue curves, respectively. The approximation of the theoretical solution is very good. The performance index of the quasi-optimal solution is greater with only 2.6%: $J_{AE}=3.799$

3.3A closed-loop solution using MPC structure

The two solutions of the OCP under consideration constructed in the previous sections are useless for a real implementation. From this point of view, only the closed-loop solutions are taken into account. The solutions described before are important for our case study as comparison elements allowing us to evaluate at what extent the MPC provides close solutions.

The Model Predictive Controller integrates the MA that makes the optimal prediction. In our case, it is about the LQP-EA slightly modified to agree with the closed-loop structure. This adaptation is described systematically in (Minzu and Serbencu, 2020).

The particularity of the objective function (12), which has an integral term and a terminal penalty as well, needs a specification. Let us consider the current moment k when the controller has to calculate the current control input $U(k)$. To do this, it has to calculate J for a predicted control sequence:

$$(20) pcs = \langle U(k/k), \dots, U(N-1/k) \rangle.$$

At this moment, the controller has already "sent" toward the Process the following control input sequence:

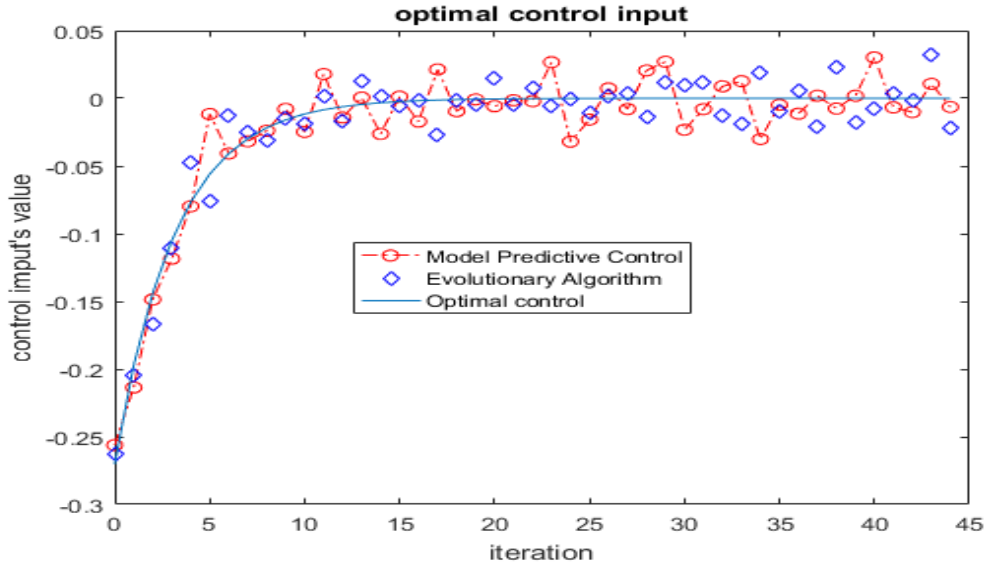


Fig.3. Comparative evolution of the control input

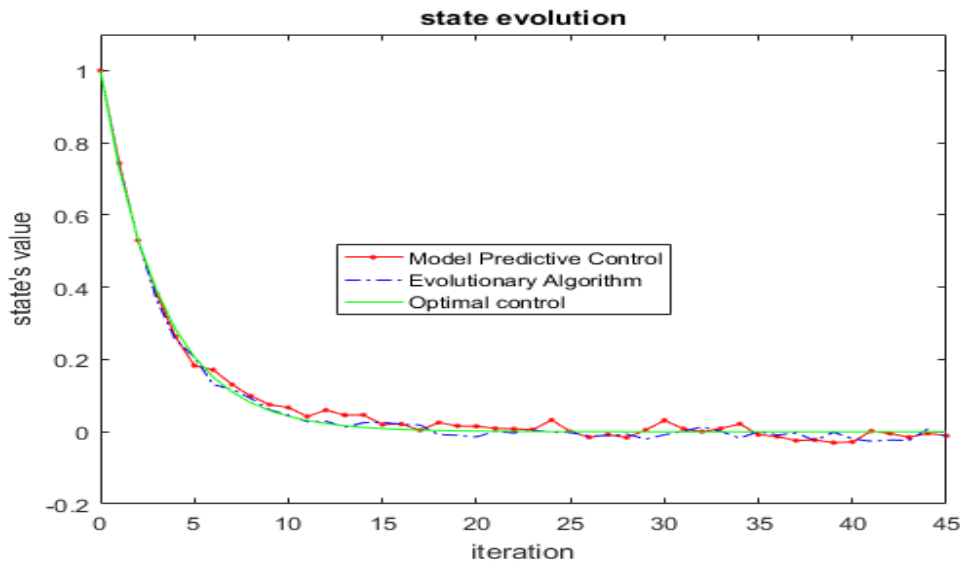


Fig.4. Comparative evolution of the state variable

$$(21) \langle U(0), \dots, U(k-1) \rangle.$$

The calculation of J needs the concatenation of the two sequences. In Fig. 1, the notation $J(pcs)$ means that the sequence (21) is implicitly taken into consideration as past control values. The EA calculates the value J for each solution of the current population. Equation (11) is used repeatedly to calculate the predicted states (6). Afterwards the objective function can be calculated using equation (12). Equation (13) determines simultaneously the value J^* and the best pcs . This sequence's first element, $U(k)=U(k/k)$, is the current control input value that is "sent" to the Process.

As we have already mentioned, the minimization of the performance index (10) is no longer necessary. The good convergence of the EA is a sufficient condition to ensure the minimization of the prediction error.

4. QUANTITATIVE COMPARISON BETWEEN SOLUTIONS

The theoretical solution calculated in section 3.1 has allowed us to know the LQP's optimal solution. The algorithm's implementation has been made using the MATLAB language and system. The simulation's results are already presented in Fig. 2, 3 and 4.

Table 2: Performance index's value

J_{star}	J_{AE}	$\varepsilon_{AE} [\%]$	J_{MPC}	$\varepsilon_{MPC} [\%]$
3.702	3.799	2.6	3.832	3.5

Table 2 gives the values of the performance index for the three solutions. It also shows the relative errors of the two solutions against the theoretical one:

$$(22) \quad \varepsilon_{AE} = \frac{J_{AE} - J_{star}}{J_{star}} \quad \varepsilon_{MPC} = \frac{J_{MPC} - J_{star}}{J_{star}}$$

Although it holds

$$\varepsilon_{MPC} > \varepsilon_{AE},$$

the value ε_{MPC} is quite small, meaning that the RHC structure still implements a quasi-optimal solution for the LQP.

Fig. 5 shows the error of trajectory yielded by LQP-EA (or MPC) as against the theoretical one. A greater error of the MPC structure is the price to pay for the closed-loop implementation.

In our case, the degradation of the performance index is acceptable. Therefore one may consider that the MPC structure yields a quasi-optimal solution in closed-loop.

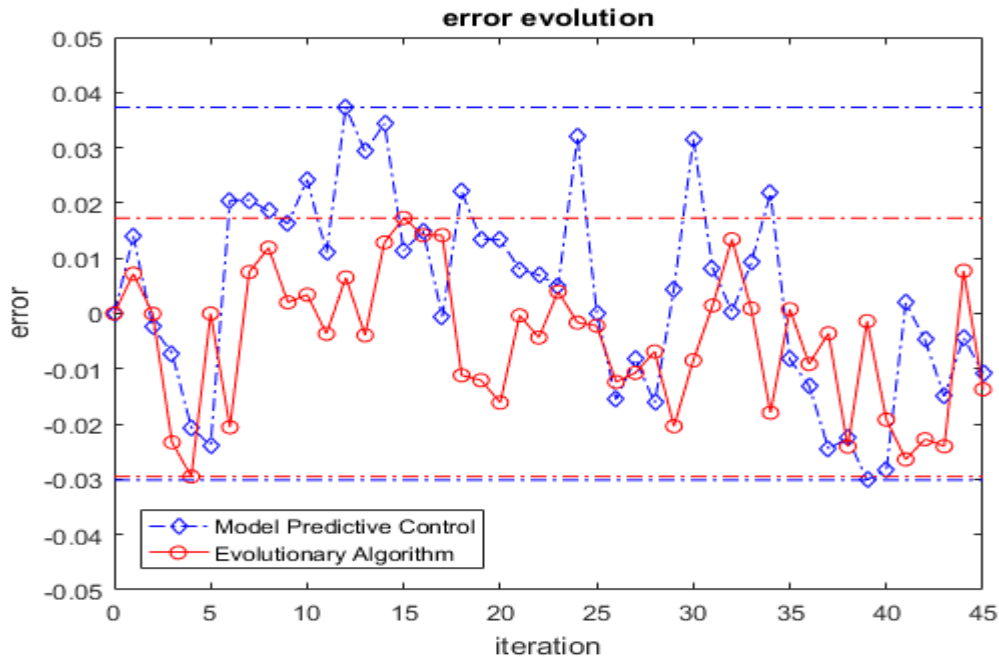


Fig.5. Errors of the open and closed-loop trajectories.

5. CONCLUSION

The MPC structure whose controller integrates an MA can implement a closed-loop solution for a given OCP. The analysis from section 2 and many simulations have proved that the optimization of the performance index is equivalent to the prediction error's minimization. Section 3 has presented a particular OCP that is not the main objective of this work. LQP has a theoretical solution that can be constructed and compared with the MAs' solutions.

REFERENCES

- Christofides, P.D., (2013). Distributed model predictive control: A tutorial review and future research directions. *Computers and Chemical Engineering*, **Vol 51**, 21-41.
- Clarke, D.W., (1994). *Advances in Model-based Predictive Control*. Oxford University Press.
- Hiskens, I.A. and B. Gong, (2006). Voltage Stability Enhancement Via Model Predictive Control of Load. *Intelligent Automation & Soft Computing*, **Vol. 12(1)**, 117-124.
- Kruse, R., C. Borgelt, C. Braune, S. Mostaghim and M. Steinbrecher (2016). *Computational Intelligence - A Methodological Introduction*, second edition, Springer
- Lopez-Francol, C., (2018). Robot Pose Estimation Based on Visual Information and Particle Swarm Optimization, *Intelligent Automation & Soft Computing*, **Vol 24(2)**, 431-442.
- The case study from section 3 and also other implementations achieved in this framework have proved that our result from section 2 is realistic.
- An MA within the controller is a realistic solution to implement a closed-loop structure. The price to pay is the deterioration of the performance index to some extent.
- The OCP solution works in a degraded mode but still could have a quasi-optimal character. The designer of the control system must decide through off-line simulations whether the degraded mode is acceptable for a real implementation.
- Michalewicz, Z., C.Z. Janikov and J.B. Krawczyk, (1992). A modified genetic algorithm for optimal control problems. *Computers Math. Applic.* **Vol. 23, No. 12**, pp. 83-94.
- Minzu, V. and A. Serbencu, (2020), Systematic Procedure for Optimal Controller Implementation using Metaheuristic Algorithms. *Intelligent Automation & Soft Computing*, **Vol. 26, no.4** ,653-667, in press.
- Qin, S.J., Badgwell, T.A., (2003). A survey of industrial model predictive control technology. *Control Engineering Practice*, **Vol 11**, 733-764
- Talbi, E.G., (2009). *METAHEURISTICS-From Design to Implementation*. ISBN 978-0-470-27858-1, WILEY
- Yang, Y., (2014). Predictive Control Strategy Based on Extreme Learning Machine for Path-Tracking of Autonomous Mobile Robot. *Intelligent Automation & Soft Computing*, **Vol 21(1)**, 1-19.
- Zheng, T., (2010). *Model Predictive Control*. (Edited by Tao Zheng), ISBN 978-953-307-102-2, Publisher: Sciyo.