Advanced Metaheuristics-based Tuning of Effective Design Parameters for Model Predictive Control

Mohamed Derouiche, Soufiene Bouallègue, Joseph Haggège, Guillaume Sandou

HAL Id: hal-02861054

https://centralesupelec.hal.science/hal-02861054

Submitted on 8 Jun 2020

Advanced Metaheuristics-based Tuning of Effective Design Parameters for Model Predictive Control Approach

Mohamed Lotfi Derouiche[1], Soufiene Bouallègue[*2], Joseph Haggège[3], Guillaume Sandou[4]

Laboratoire de Recherche en Automatique (LA.R.A), École Nationale d'Ingénieurs de Tunis (ENIT)[1, 3]
Université de Tunis EL Manar, BP 37, Le Belvédère, 1002 Tunis, Tunisia
Institut Supérieur des Systèmes Industriels de Gabès, Université de Gabès, Rue Salaheddine EL Ayoubi, 6011 Gabès, Tunisia[2]
L2S, CentraleSupélec, CNRS, Université Paris Sud, Université Paris-Saclay[4]
3 rue Joliot Curie, 91192 Gif-sur-Yvette Cedex, France

Abstract—**This paper presents a systematic tuning approach for Model Predictive Control (MPC) parameters' using an original LabVIEW-implementation of advanced metaheuristics algorithms. Perturbed Particle Swarm Optimization (pPSO), Gravitational Search Algorithm (GSA), Teaching-Learning Based Optimization (TLBO) and Grey Wolf Optimizer (GWO) metaheuristics are proposed to solve the formulated MPC tuning problem under operational constraints. The MPC tuning strategy is done offline for the selection of both prediction and control horizons as well as the weightings matrices. All proposed algorithms are firstly evaluated and validated on a benchmark of standard test functions. The same algorithms were then used to solve the formulated MPC tuning problem for two dynamical systems such as the magnetic levitation system MAGLEV 33-006, and the three-tank DTS200 process. Demonstrative results, in terms of statistical metrics and closed-loop systems responses, are presented and discussed in order to show the effectiveness and superiority of the proposed metaheuristics-tuned approach. The developed CAD interface for the LabVIEW implementation of the proposed metaheuristics is given and freely accessible for extended optimization puposes.**

Keywords—Model predictive control; parameters tuning; advanced metaheuristics; MAGLEV 33-006; DTS200 three-tank process; LabVIEW implementation

I. INTRODUCTION

In recent decades, the Model Predictive Control (MPC) has emerged as a leading control strategy due to its effectiveness and robust performance on complex systems under operational constraints. Such a control approach is one of the most used and successfully implemented in a large variety of industrial applications. As an advanced robust control strategy, the MPC approach is used to drive renewable energy systems [1]. By controlling the bidirectional buck-boost converters of the PV-battery based AC micro-grid system, the fluctuating output from the solar energy sources is smoothed. In [2], Rahimi & Moghaddam proposed an extended MPC approach for maximizing the absorbed power of a point absorber wave energy converter. In [3], a new information-theoretic based MPC approach has been developed and applied to autonomous vehicles in the aggressive driving around a dirt test track. Worthmann et al. [4] proposed a tailored nonquadratic stage cost-based MPC sheme is proposed to the steering problem of

the nonholonomic unmanned ground vehicles. Other several various developments and applications of MPC appraoch can be found in [5-8].

In the MPC formalism, the tuning of effective design parameters, i.e. control and prediction horizons as well as the weighting matrices [6,7], remains a hard problem and a serious drawback. Indeed, the successful implementation of a MPC algorithm in practical applications requires appropriate tuning of the controller parameters which specify the performance of the closed-loop dynamics. This parameters' tuning, difficult and not systematic, becomes more and more tedious and time-consuming. To overcome such a problem, various techniques have been proposed and investigated. In [9], an intelligent deep-learning based mechanism is proposed for the implementation of a MPC algorithm for a mode-locked fiber laser system. The introduced recurrent neural network allows the classic MPC predicting of the birefringence and the laser states task. Yamashita, Zanin & Odloak [10] formulated the MPC tuning task as a mutli-objective optimization problem. Two methods based on the lexicographic and compromize optimization algorithms have been proposed and succesfully applied to a shell heavy oil fractionator benchmark. Another similar tuning approach based on the Pareto multi-objective optimization has been proposed for drinking water networked systems [11]. In [12], the authors present a reverse-engineering tuning method for the MPC strategy and applied for a binary distillation column system. In a pole placement framework, an analytical MPC tuning strategy has been proposed for a class of industrial systems described by first order plus dead time models [13]. Shah & Engell [14] present a semi-definite programming based approach to determine MPC parameters for MIMO systems thanks to a specification of the desired behavior of the closed-loop for small changes.

All these related works lack the systematic aspect and simplicity of the MPC parameters tuning. Indeed, most of these techniques are restrictive, and/or time-consuming in real-world implementation scenarios. Far from these analytical and restrictive tuning methods, the use of soft computing concepts, in particular the optimization by metaheuristics algorithms, seems a promising solution for this kind of hard problems, often non-convex and non-smooth [15-17]. Up to date, there

have been a few research works that successfully integrated the metaheuristics-based optimization and MPC approach. Most of these given works use classic and old metaheuristics such as the Genetic Algorithms (GA), standard Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO) and so on [18-23]. Unfortunately, all these algorithms for the formulated problems present limitations in terms of premature convergence and unbalanced exploitation/exploration mechanism. These drawbaks affect the quality and optimality of the found solutions.

Recently, many advanced and global metaheuristics have been proposed in the literature with easier to implement algorithms and less control parameters in comparison with the old ones. The perturbed Particle Swarm Optimization (pPSO) [24], Gravitational Search Algorithm (GSA) [25], Teaching-Learning Based Optimization (TLBO) [26] and Grey Wolf Optimizer (GWO) [27] are have attracted considerable interest due to their effectiveness and wide range of applicability. Their suitability to solve the MPC parameters' tuning, formulated as a constrained optimization problem, presents a promising alternative for reducing the complexity of the MPC strategy. In this paper, an original LabVIEW-based implementation of these advanced metaheuristics algorithms is investigated. The tuning stage of all effective MPC parameters, i.e. weighting filters and prediction/control horizons, is formulated as a nonlinear optimization problem under operational constraints. Numerical validation and comparison studies are given for two different benchmarks process such as the suspended sphere MAGLEV-33-006 and the DTS200 three interconnected tanks DTS200 systems. All the proposed metaheuristics have been firstly experimented and validated on a benchmark of standard test problems [15,16].

The main contributions of the paper are, on the one hand, the development of a systematic metaheuristics-based method of easy and fast tuning of the MPC parameters' for dynamical systems. The classical trials-errors methods are no longer used and the design time is further reduced. On the other hand, a software CAD interface for the implementation of the proposed metaheuristic algorithms, in particular for the resolution of the MPC tuning problem, is developed under the LabVIEW graphical programming environment. Such a software tool is freely accessible.

The remainder of this paper is organized as follows. In Section II, a preliminary survey on the MPC approach is firstly presented. The effective control parameters' tuning problem is formulated as a constrained optimization problem. Section III presents a theoretical background of the proposed pPSO, GSA, TLBO and GWO metaheuristics as well as their original implementation under the graphical programming LabVIEW software. All given algorithms are numerically validated through various test functions from the optimization literature. Section IV is dedicated to the application of the proposed metaheuristics-tuned MPC approach for the position control of a suspended sphere in the didactic MAGLEV-33-006 benchmark as well as the level control of the three DTS200 plants. All demonstrative simulation results are presented and discussed. Section V concludes this paper.

II. TUNING PROBLEM FORMULATION

A. Formalism and basic Concepts

In the MPC framework, the human behavior is reproduced according to which it is a question of selecting control actions for the system to be controlled on a finite horizon [5-7]. To achieve the control performances under operational constraints, a digital model of the plant is used and decisions are constantly updated as depicted in Fig. 1.

In Fig. 2, both prediction and control horizons, denoted as N_p and N_c, respectively, are shown as ones of the main and effective design parameters of the MPC approach. The discrete-time control signals, system outputs and reference trajectories are u, y, and r, respectively. The MPC algorithm assumes that $u(k-1+i) = u(k+N_c-1)$ for $N_c < i \leq N_p$.

A MPC algorithm leads to compute the control laws $\{u(k-1+i), i=1,2,\ldots,N_c\}$ where only the first element $u^*(k)$ of such an optimized control sequence is applied to the system. These control laws are updated at each sampling time k in order to minimize the cost function (1) under various operational constrains [5,6,23]:

$$J(k) = \sum_{i=1}^{N_p} \hat{e}^T(k+i \mid k) \boldsymbol{Q} \hat{e}(k+i \mid k)$$
$$+ \sum_{i=0}^{N_c-1} \left[\Delta u^T(k+i \mid k) \boldsymbol{R} \Delta u(k+i \mid k) \right] \quad (1)$$

where $r(k+i \mid k)$, $\hat{y}(k+i \mid k)$ and $\Delta u(k+i \mid k)$ denote respectively the setpoint trajectories, predicted outputs and increments of control laws at the time $k+i$, given all measurements up to and including those at sampling-time k. The terms $\boldsymbol{Q} = \boldsymbol{Q}^T > 0$ and $\boldsymbol{R} = \boldsymbol{R}^T > 0$ are the weighting matrices of the MPC technique, and $\hat{e}(.) = \hat{y}(.) - r(.)$.
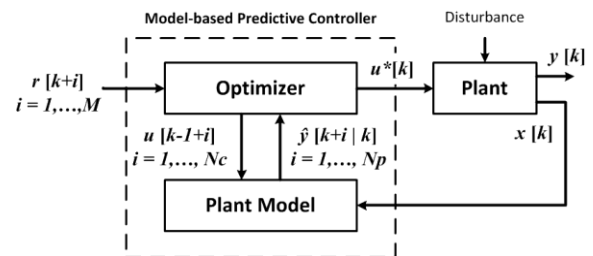

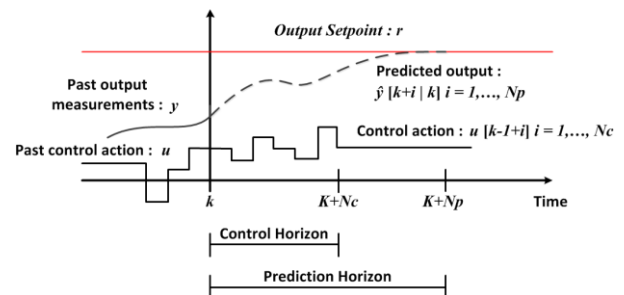Fig. 1. Model Predictive Control Structure.

Fig. 2. Prediction and Control Horizons in MPC Framework.

Such an optimization process is repeated at the next sampling-time based on the measured system state $x(k)$ and under operational constraints specified on the control actions, plant output signals and states as follows:

$$u_{\min} \le u(k) \le u_{\max}$$
$$\Delta u_{\min} \le \Delta u(k) \le \Delta u_{\max}$$
$$y_{\min} \le y(k) \le y_{\max}$$
$$x_{\min} \le x(k) \le x_{\max} \tag{2}$$

B. Optimization Problem Formulation

Since the operational constraints of the MPC are specified, the effective design parameters, i.e. prediction/control horizons (N_p, N_c) and the weighting matrices (Q, R), need to be tuned appropriately to improve the performances and robustness of the predictive controllers. In this paper, all these control parameters are considered as decision variables of the following formulated optimization problem:

$$\begin{cases} J_1(N_p, N_c, Q, R) = \sum_{\tau=0}^{+\infty} \left[y(\tau) - r(\tau) \right]^2 \\ s.t: \quad N_c \le N_p \\ \quad u_{\min} \le u(\tau) \le u_{\max}, \\ \quad \Delta u_{\min} \le \Delta u(\tau) \le \Delta u_{\max}, \\ \quad y_{\min} \le y(\tau) \le y_{\max}, \\ \quad x_{\min} \le x(\tau) \le x_{\max} \end{cases} \tag{3}$$

where τ is the simulation time.

Schematically, the principle of the proposed metaheuristics-tuned MPC parameters' is illustrated in Fig. 3.

The problem formulation (3) takes into account the operational constraints given by Equation (2). The prediction horizon N_p is always higher than the control one N_c. For simplicity purposes, only constraints on the control laws are considered as follows:

$$\max_{0 \le \tau < +\infty} |u(\tau)| \le u_{\max} \tag{4}$$

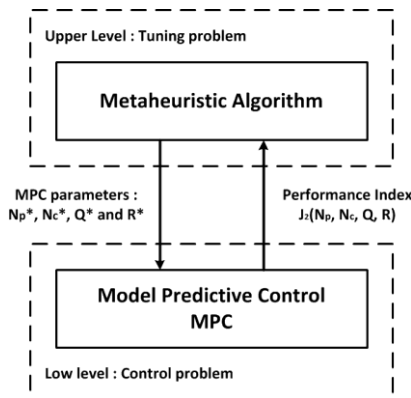where $u_{\max}$ is the maximum value of the control signal.

Fig. 3. Proposed Metaheuristics-based MPC Tuning Approach.

Since the optimization problem (3) is constrained, the following static penalty-based method is proposed to handle with the inequality type of operational constraints:

$$J_2(N_p, N_c, Q, R) = \sum_{\tau=0}^{+\infty} \left[y(\tau) - r(\tau) \right]^2$$
$$+ \exp\left(1000 \frac{N_c - N_p}{N_p} \right)$$
$$+ \exp\left(1000 \frac{\max_{0 \le \tau < +\infty} |u(\tau)| - u_{\max}}{u_{\max}} \right) \tag{5}$$

III. PROPOSED METAHEURISTICS

A. Perturbed PSO Algorithm

Originally proposed by Xinchao [24], the perturbed variant of the PSO algorithm, abbreviated pPSO, maintains a population of n_{PART} particles in the D-dimensional research space. The ith particle of the swarm is characterized by its current position $x_t^i = (x_t^{i,1}, x_t^{i,2}, \dots, x_t^{i,D})$ and velocity $v_t^i = (v_t^{i,1}, v_t^{i,2}, \dots, v_t^{i,D})$ at the iteration $t \in [1, n_{GENE}]$. In the pPSO formalism, the trajectory of the ith particle is updated according to the following motion equations:

$$x_{t+1}^i = x_t^i + v_{t+1}^i \tag{6}$$
$$v_{t+1}^i = w v_t^i + c_1 r_{1,t}^i \left(p_t^i - x_t^i \right) + c_2 r_{2,t}^i \left(\hat{p}_t^g - x_t^i \right) \tag{7}$$

where w is the inertia factor, c_1 and c_2 are the cognitive and social factors respectively, $r_{1,t}^i$ and $r_{2,t}^i$ are random numbers uniformly distributed in the interval $[0,1]$, and p_t^i is the best previously position, obtained by the ith particle in the swarm and the global best position $\hat{p}_t^g$ is now defined by:

$$\hat{p}_t^g = N\left(p_t^g, \sigma \right) \tag{8}$$

where σ is the degree of uncertainty about the optimality of the global best position p_t^g in the standard PSO algorithm.

B. Gravitational Search Algorithm

The Gravitational Search Algorithm (GSA), initially proposed by Rashedi, Nezamabadi & Saryazdi [25], considers agents as objects of different masses to solve difficult and hard optimization problems. Thanks to the gravity forces, each object will be influenced by their neighbors. Every object position is updated using the concepts of Newton's laws of gravity and motion, as follows:

$$F_t^{ij} = G_t \cdot \left(\frac{m_{a,t}^i \cdot m_{p,t}^j}{r_{ij,t}^2} \right); \quad a_t^i = \frac{F_t^i}{m_t^i} \tag{9}$$

where F_t^{ij} denotes the gravitational force between two entities, G_t is the gravitational constant, $m_{a,t}^i$ is the active mass and $m_{p,t}^i$ is the passive mass, $r_{ij,t}$ is the distance between the ith and jth entities, and F_t^i, a_t^i and m_t^i represent the force applied to the ith particle, its acceleration and mass, respectively.

C. Teaching-Learning based Optimization Algorithm

Introduced Rao, Savsani & Vakharia [26], the Teaching-Learning Based Optimization (TLBO) method is based on the influence effect of a teacher on the output of learners, i.e. the potential solutions of the optimization problem. The TLBO algorithm is divided in two phases: teaching and learning. In the first stage, the teacher tries to bring his learners up to his level in terms of knowledge. The new vector of decision variables is updated as follows:

$$x_{t+1}^i = x_t^i + Difference\_Mean_t \tag{10}$$

where $Difference\_Mean_t$ denotes the difference between the existing and new means M_t and M_{t+1}, computed as follows:

$$Difference\_Mean_t = rand_t \left(M_{t+1} - T_F M_t \right) \tag{11}$$

where $rand_t$ is a random number in the interval $[0,1]$ and T_F is the teaching factor that selects the value of mean as follows:

$$T_F = round \left[1 + rand(0,1) \right] \tag{12}$$

In the second phase, the increasing knowledge is done randomly through the interaction between learners using the following computation steps:

If $f\left(x_t^i\right) < f\left(x_t^j\right)$, then $x_{t+1}^i = x_t^i + rand_t\left(x_t^i - x_t^j\right)$, else
$$x_{t+1}^i = x_t^i + rand_t\left(x_t^j - x_t^i\right) \tag{13}$$

D. Grey Wolf Optimizer Algorithm

Recently proposed by Mirjalili, Mirjalili & Lewis [27], the Grey Wolf Optimizer (GWO) tries to mimic the social hierarchy and behavior of wolves where the fittest solution, denoted as α, is the leader of the troupe. Subsequently, the second and third best solutions are named as β and δ, respectively. All remaining candidate solutions are presumed to be ω. The related algorithm has the following four phases:

- Encircling prey sub-model:

$$D_t^i = \left| C_t^i x_t^p - x_t^i \right| \tag{14}$$

$$x_{t+1}^i = x_t^p - A_t^i D_t^i \tag{15}$$

where x_t^p is the prey position, and x_t^i indicates the position of a grey wolf, A_t^i and C_t^i are control parameters defined as:

$$A_t^i = 2a_t r_{1,t}^i - a_t \tag{16}$$

$$C_t^i = 2r_{2,t}^i \tag{17}$$

where a_t are linearly decreasing in the interval [2, 0] and $r_{1,t}^i$ and $r_{2,t}^i$ are random vectors in $[0,1]$.

- Hunting sub-model:

$$\begin{cases} D_{\alpha,t}^i = \left| C_{1,t}^i \, x_{\alpha,t}^i - x_t^i \right| \\ D_{\beta,t}^i = \left| C_{2,t}^i \, x_{\beta,t}^i - x_t^i \right| \\ D_{\delta,t}^i = \left| C_{3,t}^i \, x_{\delta,t}^i - x_t^i \right| \end{cases} \tag{18}$$

$$\begin{cases} x_{1,t}^i = x_{\alpha,t}^i - A_{1,t}^i \, . \, D_{\alpha,t}^i \\ x_{2,t}^i = x_{\beta,t}^i - A_{2,t}^i \, . \, D_{\beta,t}^i \\ x_{3,t}^i = x_{\delta,t}^i - A_{3,t}^i \, . \, D_{\delta,t}^i \end{cases} \tag{19}$$

$$x_{t+1}^i = \frac{x_{1,t}^i + x_{2,t}^i + x_{3,t}^i}{3} \tag{20}$$

- Attacking prey step: this is done by decreasing a_t over the course of iterations which enhances the exploitation capacity of the algorithm.

- Search for prey: this is done by giving A_t^i random values greater than 1 or less than -1 which enhances the exploration mechanism.

E. Numerical Experimentation and Analysis

All proposed pPSO, GSA, TLBO and GWO metaheuristics algorithms are graphically implemented and executed under LabVIEW software in such a manner to insure modularity and scalability using the Sub-Virtual Instruments (VIs) formalism (see Fig. A1 and A2 of Appendix A). All these algorithms have been coded in MATLAB 7.8 and executed on a PC computer with Core 2 Duo-2.20 GHz CPU and 2.00 GB RAM. After the software implementation phase, various optimization tests are used to compare the obtained results with the published ones. To do so, the optimization test functions of Appendix B have been implemented as new libraries in the LabVIEW Functions palette.

A benchmark of eight standard test functions [15-17], with various properties for the optimization framework, is adopted for the numerical experimentation stage. Each of these functions has a different set of features representative of a different class of single-objective optimization problems. All proposed metaheuristics algorithms have been evaluated and validated thanks to this benchmark. The related statistical results in terms of the best, mean and worst cases of the optimization as well as the standard deviation (STD) are summarized in Table I for independent 30 runs. From Table I, the validity of the LabVIEW-based implementation is checked. The convergence of all algorithms is usually guaranteed in a reasonable computation time and with remarkable superiority. Roughly, the STD metric has a small value which means that the according algorithms are repeatable over the independent 30 runs, especially for the GWO one. From such a graphical-based implementation of advanced metaheuristics, it is observed that there is no algorithm that excels in solving all considered functions. This is already mentioned by the well-known "No Free Lunch" theorem which stipulates that "for any algorithm, any elevated performance over one class of problems is offset by performance over another class".

TABLE I. NUMERICAL OPTIMIZATION RESULTS OVER 30 RUNS

Function		pPSO	GSA	TLBO	GWO
f_1	Best	5.55e-06	1.30e-16	1.45e-06	1.44e-30
	Mean	7.50e-06	4.03e-16	1.75e-04	1.07e-27
	STD	8.97e-07	3.97e-16	3.69e-04	1.10e-27
	Median	7.48e-06	2.51e-16	3.75e-05	5.92e-28
f_2	Best	1.09e-02	4.48e-08	1.85e-06	1.07e-17
	Mean	6.87e-01	1.13e-01	4.74e-04	1.08e-16
	STD	1.51e+00	3.77e-01	1.30e-03	7.59e-17
	Median	9.39e-02	9.27e-08	3.54e-05	7.56e-17
f_3	Best	3.75e-03	4.62e+02	5.44e+01	9.02e-09
	Mean	9.69e-01	1.02e+03	2.42e+02	1.26e-05
	STD	1.29e+00	3.22e+02	2.14e+02	2.98e-05
	Median	3.82e-01	1.01e+03	1.78e+02	1.34e-06
f_4	Best	1.82e+01	2.74e+01	1.28e+01	2.64e+01
	Mean	2.70e+01	4.24e+01	1.08e+02	2.81e+01
	STD	3.37e+00	2.95e+01	3.86e+01	7.61e-01
	Median	2.68e+01	3.00e+01	9.45e+01	2.81e+01
f_5	Best	1.00e+00	0.00e+00	4.20e+01	0.00e+00
	Mean	2.57e+01	1.32e+01	1.59e+02	0.00e+00
	STD	3.07e+01	2.57e+01	1.14e+02	0.00e+00
	Median	1.65e+01	4.00e+00	1.26e+02	0.00e+00
f_6	Best	3.87e-03	4.05e-02	2.33e-02	6.76e-04
	Mean	1.11e-02	8.19e-02	5.59e-02	2.12e-03
	STD	3.62e-03	3.26e-02	2.52e-02	1.08e-03
	Median	1.05e-02	8.04e-02	5.13e-02	1.96e-03
f_7	Best	4.08e+01	1.49e+01	1.79e+01	0.00e+00
	Mean	8.82e+01	2.93e+01	2.80e+01	3.44e+00
	STD	2.69e+01	8.62e+00	4.88e+00	3.70e+00
	Median	8.61e+01	2.59e+01	2.74e+01	2.21e+00
f_8	Best	2.38e-07	1.74e+01	1.54e-06	0.00e+00
	Mean	1.21e-02	2.74e+01	8.59e-02	3.37e-03
	STD	1.50e-02	6.97e+00	1.15e-01	7.33e-03
	Median	7.40e-03	2.60e+01	3.95e-02	0.00e+00

IV. APPLICATIONS TO THE CONTROL OF DYNAMIC SYSTEMS

A. Case-Study 1: Magnetic Levitation System

The magnetic levitation system MAGLEV 33-006 from the Feedback Company (see Fig. 4) is a SISO process example used to validate the proposed metaheuristics-tuned MPC approach. Demonstrative results and analyses are given and discussed through this subsection.
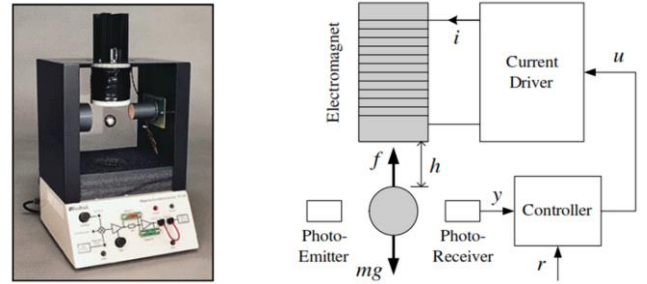

Fig. 4. Schematic Set-up of the MAGLEV 33-006 System.

The position dynamics of the metal sphere of MAGLEV system is modeled as follows [28, 29]:

$$m\frac{d^2 y}{dt^2} = \gamma mg_a - K\frac{(\rho u + i_0)^2 \gamma^3}{(y - y_0)^2} \tag{21}$$

where y is the voltage reflecting the distance between the coil and the metal sphere, $\gamma > 0$, $\rho > 0$ and K are the sensor gain, coil resistor and electromechanical conversion gain, respectively, y_0 and $i_0 > 0$ are the voltage and the offsets current, respectively, and g_a is the acceleration of gravity.

Taking $x = \begin{bmatrix} y & \dot{y} \end{bmatrix}^T$ as a state vector, the continuous-time state-space representation of the system is given by:

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = \gamma g_a - K\frac{(\rho u + i_0)^2 \gamma^3}{m(x_1 - y_0)^2} \end{cases} \tag{22}$$

With a sampling period of 5ms and while using the numeric values of the physical model parameters, a discrete-time model of the MAGLEV 33-006 plant is derived as follows:

$$\begin{cases} x_{k+1} = \begin{bmatrix} 1.0108 & 0.0050 \\ 4.3185 & 1.0108 \end{bmatrix} x_k + \begin{bmatrix} -0.0142 \\ -5.6779 \end{bmatrix} u_k \\ y_k = \begin{bmatrix} 1 & 0 \end{bmatrix} x_k \end{cases} \tag{23}$$

Since that the MAGLEV 33-006 system is SISO type, the unknown weighting matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ of the MPC approach are scalars. Therefore, the formulated optimization problem (5) has four decision variables, i.e. two horizons (N_p, N_c) and two weighting coefficients ($\boldsymbol{Q} = q_1$, $\boldsymbol{R} = r_1$). For all proposed pPSO, GSA, TLBO and GWO algorithms, a maximum number of iterations equal to 100 and a population size of 40 are considered as common control parameters. For the algorithms with specific control parameters, i.e. pPSO and Genetic Algorithm (GA), the following set of coefficients is used:

- pPSO [24]: $c_1 = 0.5$; $c_2 = 0.3$; $w = 0.9$; $\sigma_{max} = 0.15$ $\sigma_{min} = 0.001$ and $\alpha = 0.5$;

- GA [30]: crossover probability 0.85; mutation rate 0.005; Min mutation rate 0.0005 and Max rate 0.25.

TABLE II.   OPTIMIZATION RESULTS FOR THE MAGLEV SYSTEM OVER 30 RUNS

	GA	pPSO	GSA	TLBO	GWO
Best	1.95e+01	7.68e+00	7.68e+00	7.67e+00	7.67e+00
Mean	3.02e+01	7.68e+00	7.68e+00	7.68e+00	7.68e+00
STD	1.62e+01	9.33e-04	1.98e-03	1.35e-03	1.27e-03
Median	2.37e+01	7.68e+00	7.68e+00	7.68e+00	7.68e+00
Duration	01:53:14	06:10:42	00:44:00	01:30:55	02:20:54
N_p^*	2.00e+01	2.00e+02	8.30e+01	2.70e+01	2.60e+01
N_c^*	1.10e+01	5.20e+01	3.90e+01	1.40e+01	1.40e+01
$Q^* = q_1$	3.21e+00	1.85e+02	9.59e+01	5.77e+01	8.56e+01
$R^* = r_1$	2.57e+00	1.72e+02	8.92e+01	5.26e+01	7.74e+01
u_{max}	2.29e+00	2.29e+00	2.29e+00	2.29e+00	2.29e+00

The resolution of the constrained optimization problem (5) leads to the results of Table II.

From a practical point of view, when exceeding the value $u_{max} = 2.3\ V$, the ball will stack immediately to the coil which creates a constraint on the control action. For all numerical experimentations, a comparison with the GA metaheuristic [30] is investigated. From these results, all metaheuristics improve high performances in terms of stability, trajectories tracking and handling of constraints. All proposed algorithms have almost reached the same solutions quality in terms of time computation and optimality, except the GA one. The plant output and control action signals have similar shapes and one representation is shown in Fig. 5 and Fig. 6, respectively.

An evaluation of the time-domain performances of the metaheuristics-tuned MPC for the MAGLEV process is given in Table III. It is shown that the TLBO-based method leads to high performance in terms of fastness and damping responses as well as the precision of the steady-state dynamics.

B. Case-Study 2: Hydraulic System

As depicted in Fig. 7(a), the DTS200 three-tank system is used to validate the proposed metaheuristics-tuned MPC approach for MIMO systems.
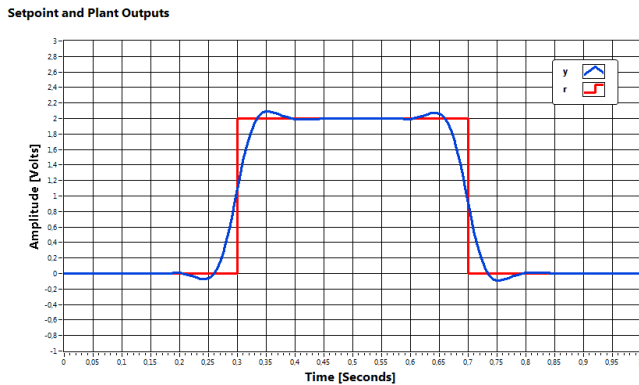

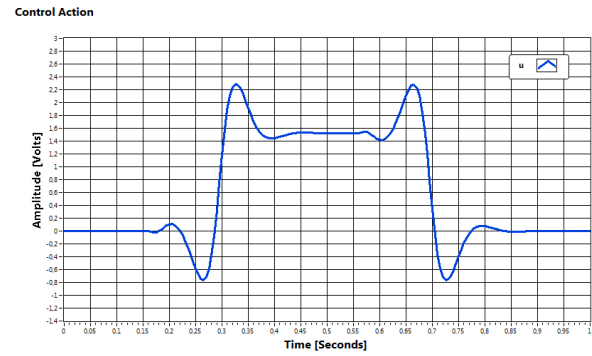Fig. 5.   MAGLEV 33-006 System Response: TLBO-based Approach.

Fig. 6.   Metaheuristics-Tuned MPC Signal: TLBO-based Approach.

TABLE III.   TABLE. PERFORMANCE EVALUATION OF THE TUNED MPC OF MAGLEV SYSTEM

Algo.	Settling time (ms)	Overshoot (%)	Steady-state error
pPSO	135.00	10	0.00
GSA	150.00	5	0.00
TLBO	100.00	5	0.00
GWO	120.00	8	0.00
GA	090.00	12	0.00

A simplified dynamic model for such a system is given as follows [31-33]:

$$\begin{cases} \dfrac{dh_1}{dt} = Q_1 - k_1\sqrt{|h_1 - h_3|}\,sign(h_1 - h_3) \\[2mm] \dfrac{dh_2}{dt} = Q_2 - k_3\sqrt{|h_2 - h_3|}\,sign(h_2 - h_3) - k_2\sqrt{h_2} \\[2mm] \dfrac{dh_3}{dt} = k_1\sqrt{|h_1 - h_3|}\,sign(h_1 - h_3) + k_3\sqrt{|h_2 - h_3|}\,sign(h_2 - h_3) \end{cases} \tag{24}$$

where h_1, h_2 and h_3 are the liquid heights inside the tanks. The interconnection parameters k_i and Q_i are defined as:

$$k_i = a_i\,\frac{S_n\sqrt{2g_a}}{A}; \quad Q_j = \frac{Q'_j}{A}, \forall i = 1,2,3 \text{ and } j = 1,2 \tag{25}$$

where A and S_n are the cross section of the tanks and the valves, respectively, a_i are the outflow coefficients, Q'_1 and Q'_2 are the flow rates of the two pumps P_1 and P_2.

A linearization of (24) around the point ($h_{1op} = 0.4m$, $h_{2op} = 0.2m$, $h_{3op} = 0.3m$, $Q'_{1op} = 32.24e-6\ m^3/sec$, $Q'_{2op} = 27.91e-6\ m^3/sec$) leads to the discrete-time model:

$$\begin{cases} x_{k+1} = \begin{bmatrix} 0.9503 & 0.0012 & 0.0484 \\ 0.0012 & 0.9065 & 0.0462 \\ 0.0484 & 0.0462 & 0.9041 \end{bmatrix} x_k + \begin{bmatrix} 316.472 & 0.1359 \\ 0.1359 & 309.2 \\ 8.0659 & 7.7682 \end{bmatrix} u_k \\[3mm] y_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} x_k \end{cases} \tag{26}$$

To achieve a smooth pump control, the saturation behavior is avoided. The operational constraint $u_{max} = 10^{-4} \ m^3/\sec$ is used as the maximum pumps flow rate. Since the DTS200 is a MIMO system, the decision variables $\boldsymbol{Q}$ and $\boldsymbol{R}$ of problem (5) are matrices and chosen with the following expressions:

$$\boldsymbol{Q} = \begin{bmatrix} q_{11} & 0 \\ 0 & q_{22} \end{bmatrix} > 0 \quad \boldsymbol{R} = \begin{bmatrix} r_{11} & 0 \\ 0 & r_{22} \end{bmatrix} > 0 \tag{27}$$

So, the formulated MPC tuning problem for the system (24) is solved to optimize the following decision variables:

$$\boldsymbol{x} = \left(N_c, N_p, q_{11}, q_{22}, r_{11}, r_{22} \right)^T \in \mathrm{R}_+^6 \tag{28}$$

While using the same control parameters of the proposed algorithms, 30 independent runs of problem (5) lead to the statistical results of Table IV and dynamical responses of Fig. 7(b), Fig. 8 and Fig. 9. These demonstrative results show the superiority and effectiveness of the proposed metaheuristics in terms of damping responses, tracking capabilities and robustness under the operational constraints. All proposed algorithms have almost reached the same solution qualities. In the GA-based strategy, the MPC algorithm creates pump saturation, i.e. at pump 2, by reaching the constraint u_{max} as shown in Fig. 8 and Fig. 9. However, the GWO-based method achieves more efficient and smooth control without using an anti-windup mechanism.

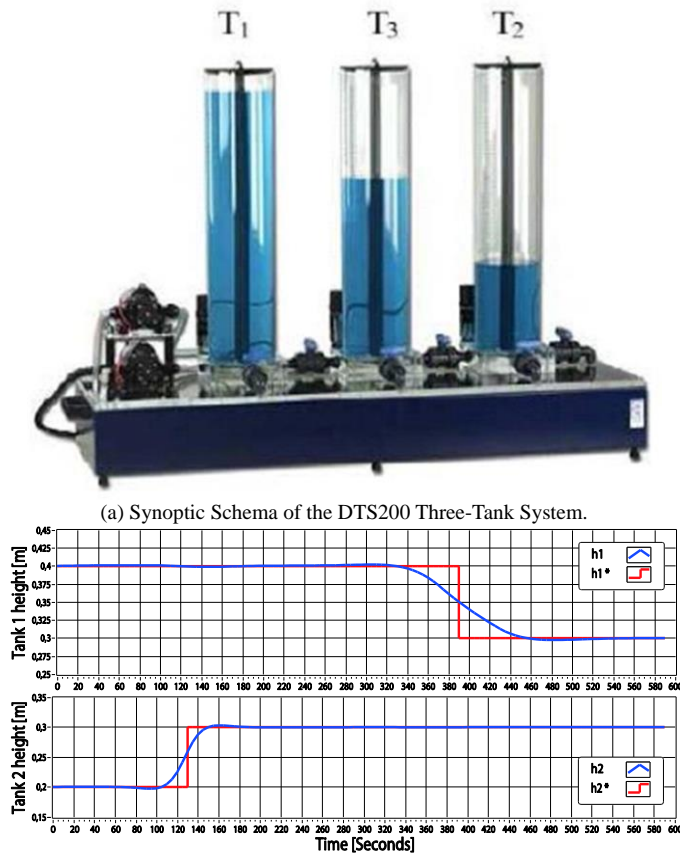(a) Synoptic Schema of the DTS200 Three-Tank System.

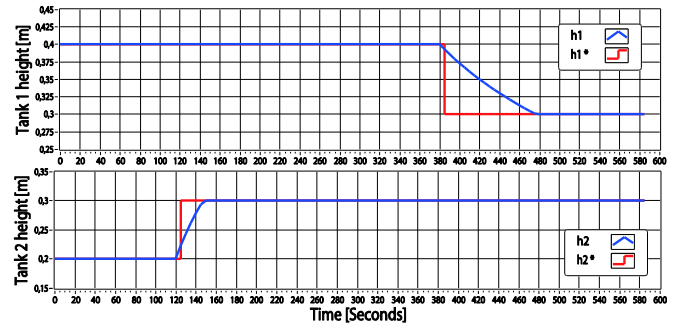Fig. 7.   (b) DTS200 Process Responses: GWO-based Approach.

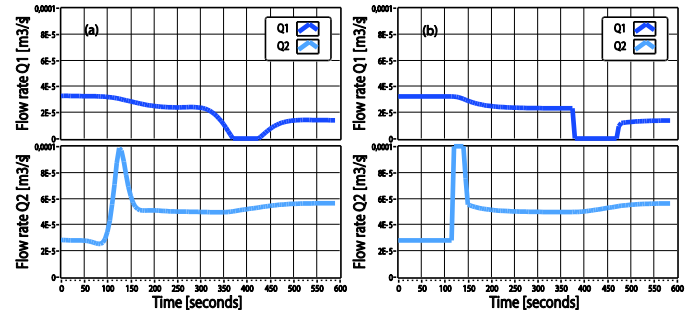Fig. 8.   DTS200 Process Responses: GA-based Approach.

Fig. 9.   Metaheuristics-Tuned MPC Signals: (a) GWO-based Approach and (b) GA-based Approach.

The performances of the metaheuristics-tuned MPC for the DTS200 process are summarized in Table V. One can observe the superiority of the GWO-tuned MPC in terms of fastness and smoothness of time-domain responses. The steady-state errors remain null for all algorithms. For the GA-tuned MPC case, the predictive behavior of the control law is not observed. The system outputs' variations are not smooth which presents a major drawback in the real-world implementation. In addition, the steady-state error for this tuning algorithm is not null in comparison with all others tuning metaheuristics.

TABLE IV.   NUMERICAL OPTIMIZATION RESULTS FOR THE DTS200 SYSTEM OVER 30 RUNS

	GA	pPSO	GSA	TLBO	GWO
Best	5.51e+00	3.58e-02	2.85e-02	2.13e-02	2.13e-02
Mean	5.95e+00	3.58e-02	4.50e-01	2.62e-02	2.81e-02
STD	2.88e-01	7.69e-07	4.12e-01	3.37e-03	5.49e-03
Median	5.98e+00	3.58e-02	2.50e-01	2.85e-02	2.85e-02
Duration	04:47:28	03:38:04	04:11:17	10:02:29	04:52:11
N_p^*	1.00e+00	1.10e+01	6.00e+00	9.50e+01	1.00e+02
N_c^*	1.00e+00	1.00e+00	1.00e+00	2.80e+01	2.70e+01
q_{11}^*	5.27e+00	5.12e+01	4.64e+01	1.00e-08	3.14e-08
q_{22}^*	1.72e+00	7.88e+00	9.16e+01	1.00e-08	1.11e-07
r_{11}^*	9.04e+01	2.22e+01	4.21e+01	5.56e-01	1.66e+00
r_{22}^*	2.85e-02	4.88e+00	3.54e+01	1.56e-02	1.73e-01
Q_{max}	1.00e-04	6.60e-05	8.35e-05	9.88e-05	9.88e-05

TABLE V.   TABLE. PERFORMANCE EVALUATION OF THE TUNED MPC OF DTS200 SYSTEM

Algo.	Settling time (sec)	Overshoot (%)	Steady-state error
pPSO	98/T1; 27/T2	0	0.00
GSA	60/T1; 52/T2	4	0.00
TLBO	64/T1; 58/T2	5	0.00
GWO	70/T1; 50/T2	1	0.00
GA	92/T1; 23/T2	0	0.01

V. CONCLUSIONS

In this paper, an advanced metaheuristics-based approach for MPC parameters' tuning has been proposed and successfully applied for SISO and MIMO systems. The proposed pPSO, GSA, TLBO and GWO algorithms are originally implemented under LabVIEW graphical software. These algorithms are firstly evaluated on a benchmark of standard test functions in order to be numerically validated for the formulated MPC tuning problems. All optimization results are discussed and compared in order to show the validity of such a LabVIEW-based implementation. The MPC tuning problem, involving the choice of the well-known prediction and control horizons as well as the weighting matrices, is formulated as a constrained optimization problem and solved using the proposed advanced metaheuristics. All demonstrative results are compared with those obtained by the classical GA-based implementation. Applications to the position control of the MAGLEV system and level regulation of the three-tank DTS200 plants are successfully achieved.

Future works deal with the real-world implementation of the metaheuristics-tuned MPC approaches using a compatible CompactRIO-RT board. The formulation of the proposed metaheuristics algorithms in an online tuning framework is also investigated for the hydraulic DTS200 process.

REFERENCES

[1] Y. Shan, J. Hu, Z. Li, and J.M. Guerrero, "A Model Predictive Control for Renewable Energy Based AC Microgrids Without Any PID Regulators", IEEE Transactions on Power Electronics, vol. 33, no. 11, pp. 9122-9126, 2018.

[2] N. Rahimi, R.K. Moghaddam, "Maximizing the Absorbed Power of a Point Absorber using an FA-based Optimized Model Predictive Control", China Ocean Engineering, vol. 32, no. 6, pp. 696-705, 2018.

[3] G. Williams, P. Drews, B. Goldfain, J.M. Rehg, and E.A. Theodorou, "Information-Theoretic Model Predictive Control: Theory and Applications to Autonomous Driving", IEEE Transactions on Robotics, vol. 34, no. 6, pp. 1603-1622, 2018.

[4] K. Worthmann, M.W. Mehrez, M. Zanon, G.K. I. Mann, R.G. Gosine, and M. Diehl, "Model Predictive Control of Nonholonomic Mobile Robots Without Stabilizing Constraints and Costs", IEEE Transactions on Control Systems Technology, vol. 24, no. 4, pp. 1394-1406, 2016.

[5] D. Bao-Cang, Modern Predictive Control, 1st ed. USA CRC Press, 2010.

[6] D. Q. Mayne, "Model predictive control: Recent developments and future promise", Automatica, vol. 50, no. 12, pp. 2967–2986, 2014.

[7] J. B. Rawlings and D. Q. Mayne, Model Predictive Control: Theory and Design. Madison, USA,: Nob Hill Publishing, 2013.

[8] S. Bouallègue and R. Fessi, "Rapid Control Prototyping and PIL Co-Simulation of a Quadrotor UAV Based on NI myRIO-1900 Board", International Journal of Advanced Computer Science and Applications, vol. 7, no. 6, pp. 26-35, 2016.

[9] T. Baumeister, S.L. Brunton, and J.N. Kutz, "Deep Learning and Model Predictive Control for Self-Tuning Mode-Locked Lasers", Journal of the Optical Society of America B, vol. 35, no. 3, pp. 617-626, 2018.

[10] A. S. Yamashita, A. C. Zanin, and D. Odloak, "Tuning of Model Predictive Control with Multi-objective Optimization", Brazilian Journal of Chemical Engineering, vol. 33, no. 02, pp. 333-346, 2016.

[11] Q. N. Tran, R. Octaviano, L. Ozkan and A.C.P.M. Backx, "Generalized Predictive Control tuning by controller matching", In Proceedings of the 2014 American Control Conference, pp. 4889-4894, Portland, Oregon, USA, 2014.

[12] P. Bagheri, A.K. Sedigh, "Analytical approach to tuning of model predictive control for first-order plus dead time models", IET Control Theory & Applications, vol. 7, no. 14, pp. 1806-1817, 2013.

[13] R. Toro, C.Ocampo-Martinez, F. Logist, J.V. Impe and V. Puig, "Tuning of Predictive Controllers for Drinking Water Networked Systems", In Proceedings of the 18th World Congress of the International Federation of Automatic Control, pp. 14507-14512, Milano-Italy, 2011.

[14] G. Shah and S. Engell, "Tuning MPC for Desired Closed-Loop Performance for MIMO Systems", In Proceedings of the 2011 American Control Conference, San Francisco, CA, USA pp. 4404-4409, 2011.

[15] X.-S.Yang, Engineering optimization :an introduction with metaheuristic applications, 1st ed. New Jersey, USA: John Wiley & Sons, Inc., 2010.

[16] J. Dréo, A. Pétrowski, P. Siarry, and E. Taillard, Metaheuristics for Hard Optimization, 1st ed. Berlin-Heidelberg: Springer-Verlag, 2006.

[17] M. Gendreau and J.-Y. Potvin, Handbook of Metaheuristics, 2nd ed., vol. 146. Boston, MA: Springer US, 2010.

[18] G. Sandou and S. Olaru, Particle Swarm Optimization Based NMPC: An Application to District Heating Networks, in Nonlinear Model Predictive Control, D.M.R.L. Magni and F. Allgöwen, Eds. Berlin Heidelberg: Springer Verlag, 2009.

[19] G. Sandou and S. Olaru, Ant Colony and Genetic Algorithm for Constrained Predictive Control of Power Systems, in Hybrid Systems: Computation and Control, A. Bemporad and A. B. Buttazzo, Eds. Berlin Heidelberg: Springer Verlag, 2007.

[20] G. Sandou, Metaheuristic strategy for the hierarchical predictive control of large scale energy networks, Control Eng. Appl. Informatics, vol. 11, no. 3, pp. 32–40, 2009.

[21] H.M. Albeahdili, T. Han and N.E. Islam, Hybrid Algorithm for the Optimization of Training Convolutional Neural Network, Int. Jour.of Advanced Comp. Science and Applications, vol.6, no.10,pp.79-85,2015.

[22] R. Suzuki, F. Kawai, H. Ito, C. Nakazawa, Y. Fukuyama, and E. Aiyoshi, Automatic Tuning of Model Predictive Control Using Particle Swarm Optimization, In Proceedings of the 2007 IEEE Swarm Intelligence Symp. Honolulu, pp. 221–226, 2007.

[23] M. L. Derouiche, S. Bouallègue, J. Haggège, and G. Sandou, LabVIEW Perturbed Particle Swarm Optimization Based Approach for Model Predictive Control Tuning, In Proceedings of the 4th IFAC International Conference on Intelligent Control and Automation Sciences, vol. 49, no. 5. Reims, pp. 353–358, 2016.

[24] X. Zhao, A perturbed particle swarm algorithm for numerical optimization, Appl. Soft Comput., vol. 10, no. 1, pp. 119–124, 2010.

[25] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, GSA: A Gravitational Search Algorithm, Inf. Sci. (Ny)., vol. 179, no. 13, pp. 2232–2248, 2009.

[26] R. V. Rao, V. J. Savsani, and D. P. Vakharia, Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems, Comput. Des., vol. 43, no. 3, pp. 303–315, 2011.

[27] S. Mirjalili, S. M. Mirjalili, and A. Lewis, Grey Wolf Optimizer, Adv. Eng. Softw., vol. 69, pp. 46–61, 2014.

[28] H. Yaghoubi, The Most Important Maglev Applications, J. Eng., vol. 2013, pp. 1–19, 2013.

[29] M. Santos, R. K. H. Galvao, and T. Yoneyama, Robust Model Predictive Control for a Magnetic Levitation System Employing Linear Matrix Inequalities, ABCM Symp. Ser. Mechatronics, vol.4, pp.147–155, 2010.

[30] W. Golebiowski, Waptia-genetic optimization algorithm-General-LAVA. [Online]. Available: https://lavag.org/files/file/94-wap tia-genetic-optimization-algorithm/. [Accessed: 08-Jan-2017].

[31] J. Ivanka and P. Navratil, Multiestimation Scheme for Adaptive Control of Three Tank System DTS200, In Proceedings of the 48th International Scientific Conference on Experimentalni Analyza Napeti 2010 Experimental Stress Analysis. Velke Losiny, pp. 123–130, 2010.

[32] P. Chalupa and J. Novak, Modeling and model predictive control of a nonlinear hydraulic system, Comput. Math. with Appl., vol. 66, no. 2, pp. 155–164, 2013.

[33] L. Wang, Model Predictive Control System Design and Implementation Using MATLAB, 1st ed. London: Springer London, 2009.

APPENDIX A: LABVIEW FRONT PANELS FOR MPC PARAMETERS' TUNING
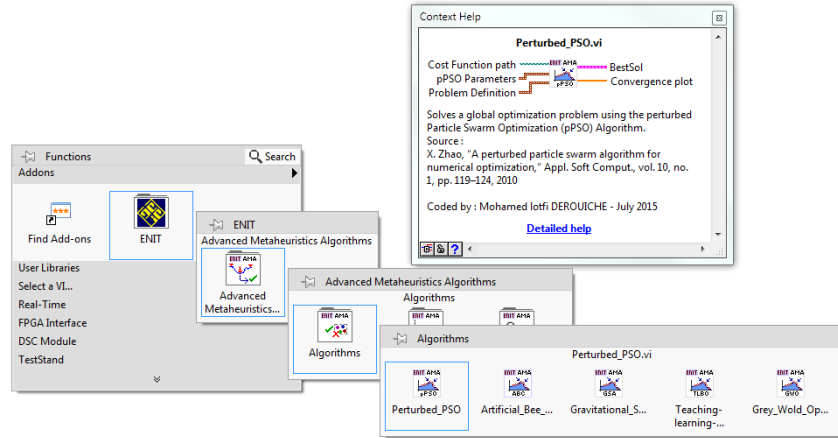

Fig. A1 Advanced Metaheuristic Algorithms as Part of the Lab view Functions Palette.
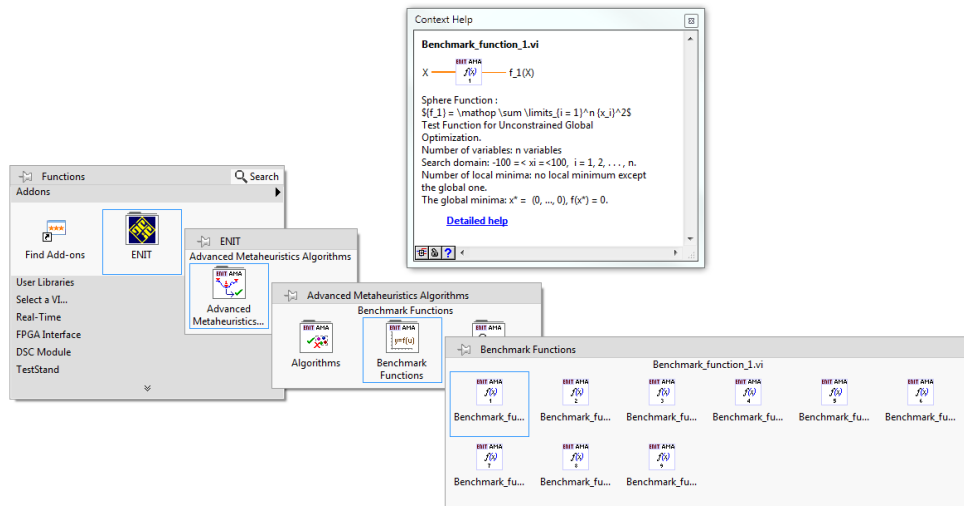

Fig. A2 Benchmark Functions as Part of the Lab view Functions Palette.

APPENDIX B: BENCHMARK OF TEST FUNCTIONS

All LabVIEW implementations of optimization functions are available for free access and download through the following National Instruments web site: HTTP://SINE.NI.COM/NIPS/CDS/VIEW/P/LANG/EN/NID/216789

Optimization functions	Domain	f_{min}	Optimization functions	Domain	f_{min}
$f_1 = \sum_{i=1}^{n} x_i^2$	$[-100,100]^n$	0	$f_5 = \sum_{i=1}^{n} \left(x_i + 0.5 \right)^2$	$[-100,100]^n$	0
$f_2 = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i|$	$[-10,10]^n$	0	$f_6 = \sum_{i=1}^{n} i x_i^4 + random[0,1]$	$[-1.28,1.28]^n$	0
$f_3 = \sum_{i=1}^{n} \left(\sum_{j=1}^{i} x_j \right)^2$	$[-100,100]^n$	0	$f_7 = \sum_{i=1}^{n} \left[x_i^2 - 10 \cdot \cos\left(2\pi x_i\right) + 10 \right]$	$[-5.12,5.12]^n$	0
$f_4 = \sum_{i=1}^{n-1} \left[100 \cdot \left(x_{i+1} - x_i^2\right)^2 + \left(x_i - 1\right)^2 \right]$	$[-30,30]^n$	0	$f_8 = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	$[-600,600]^n$	0