

Evolution of Controllers under a Generalized Structure Encoding/Decoding Scheme with Application to Magnetic Levitation System

Bin Xin, *Member, IEEE*, Yipeng Wang, Wenchao Xue, *Member, IEEE*, Tao Cai, Zhun Fan, *Senior Member, IEEE*, Jiaoyang Zhan, and Jie Chen, *Fellow, IEEE*

Abstract—Evolutionary search has been widely implemented for the adjustment of controllers' parameters. Nevertheless, the structure of controllers, which has a more important role in control systems, has been seldom studied. To this end, an evolutionary design method of controllers is proposed to optimize both structures and parameters simultaneously in this paper. A controller is made up of a combination of some basic controller components and relevant parameters. The design of controllers can be transformed into an optimization problem involving the structure (represented by discrete vectors) and parameters (represented by real numbers). A generalized structure encoding/decoding scheme is developed. Guided by the performance indicators, intelligent algorithms for both combinatorial and numerical optimization are employed to iteratively and cooperatively evolve the controller structure and parameters, respectively. In order to effectively reduce some redundant or infeasible solutions, a set of generation rules for the controller structure are put forward, which also ensures the feasibility of the structure. Furthermore, this method is applied to a magnetic levitation ball system with nonlinear dynamics and external disturbance. Both simulation and experiment results demonstrate the effectiveness and practicability of the proposed method.

Index Terms—Automatic design of controllers, mixed-variable optimization, intelligent optimization algorithms, disturbance-rejection, magnetic levitation system.

I. INTRODUCTION

This work was supported in part by the National Outstanding Youth Talents Support Program under Grant 61822304; in part by Beijing Advanced Innovation Center for Intelligent Robots and Systems under Grant 2019IRS09; in part by the Basic Science Center Programs of NSFC under Grant 62088101; and in part by the National Nature Science Foundation of China under Grant 61973299. (Corresponding author: Tao Cai and Bin Xin.)

Bin Xin, Yipeng Wang, Tao Cai, Jie Chen, and Jiaoyang Zhan are with the School of Automation, Beijing Institute of Technology, Beijing 100081, China; Jie Chen and Bin Xin are also with the Beijing Advanced Innovation Center for Intelligent Robots and Systems, Beijing Institute of Technology, Beijing 100081, China. Yipeng Wang is also with the Peng Cheng Laboratory, Shenzhen 518055, China. Wenchao Xue is with the Key Laboratory of Systems and Control, Academy of Mathematics and Systems Science, CAS, Beijing 100190, China, and the School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing 100049, China. Zhun Fan is with the Guangdong Provincial Key Laboratory of Digital Signal and Image Processing and the Key Laboratory of Intelligent Manufacturing Technology, Ministry of Education, Shantou University, Shantou 515063, China (e-mail: caitao@bit.edu.cn, brucebin@bit.edu.cn).

THE work environment, control objects, and tasks of production fields have become more and more complex, which makes control requirements of the industrial production increasingly demanding. As a consequence, the controller design has long been one of the hot topics in the field of control engineering and a flood of control methods of different characteristics have arisen. Admittedly, the Proportion-Integration-Differentiation (PID) control is widely used in many automatic control processes of machinery manufacturing, metallurgy, petroleum chemical engineering, and other industries, owing to its easy-to-understand principle, simple structure, and mature parameter tuning methods [1]–[3].

For better control performance, parameter tuning is inevitable before a PID controller is applied and substantial research efforts have been devoted to this matter. Existing parameter tuning methods can be classified into two categories: conventional methods and optimization-based methods. The Ziegler-Nichols method [4], the Cohen-Coon technique [5], the relay auto-tuning method [6], and the internal model control based method [7] are among the most representative conventional tuning methods, which are highly dependent on human beings and time-consuming. Thus, many intelligent methods such as genetic algorithm [8], differential evolution (DE) algorithm [9], ant colony optimization algorithm [10], and particle swarm optimization algorithm [11] have been adopted to tune PID parameters.

Parameter tuning is an arduous process and the control performance may deteriorate, especially when applied to complex, non-linear objects whose properties are difficult to describe accurately. Therefore, a series of advanced control methods for complex plants have been proposed, including adaptive control [12], robust control [13], sliding mode control (SMC) [14], active disturbance rejection control (ADRC) [15]–[18], optimal control [19], predictive control [20], fuzzy control [21], neural network control [22], and so on.

However, designers have to analyze the characteristics of objects deeply to design specific controllers using the above methods for satisfactory expected performance indicators. In ADRC, designers need to design an appropriate extended state observer to transform the nonlinear uncertain object into an integral-chain system [16]. Besides, it is a tedious task to tune numerous parameters of ADRC. As for SMC, the design of appropriate sliding surface and control law specific to the object is crucial to the performance of SMC controllers

[14]. Moreover, it is ordinarily necessary to linearize the object model near the equilibrium point of the system, add linear transformation modules, or apply approximations in the process of some control methods [23]. Yet these approaches may lead to a loss of system dynamic characteristics, and influence the stability and practicability of controllers in real-time setups. Hence, it is better to propose a design method that directly acts on the object and takes the performance indicators of system output as design objective.

It should be noted that the studies to date have focused on the optimization or dynamic regulation of controller parameters with fixed structures mainly, but few scholars have considered the simultaneous optimization of the structure and parameters. On the other hand, it is evident that the optimized structure can greatly improve the performance of a controller. Thus, how to achieve the optimization of both the structure and parameters during the design process becomes an important and open problem.

In the previous research, Koza et al. [24] put forward an automatic design method of control laws by genetic programming. From the view of signal processing, the controller is decomposed into several signal processing modules. The type of signal processing modules, the topology and control parameters of the modules are determined by genetic programming. Taking a 3-Lag plant with 5s delay as an example, the topology and parameters of the controller were designed by genetic programming on a cluster of 1000 computers. The simulation results showed that the obtained controller has good control effect. However, the implementation of this method depends on a large cluster of computers, the entire process will consume a lot of resources and time, which cannot meet the design requirements.

With the rapid improvement of computer performance, Xin et al. realized the simultaneous design of the structure and parameters by evolutionary algorithms [25]. The obtained controllers were denoted as the structural optimization controller (SOC). Series-parallel connection relationships between components were considered and a binary encoding method was proposed to denote the structure. A bi-level optimization strategy was proposed to optimize the structure and parameters of controllers using two variants of DE algorithms respectively. However, many structure forms are not considered and the bi-level optimization strategy is time-consuming.

In this paper, the automatic design and optimization of controllers is achieved by evolution under a generalized structure encoding/decoding scheme. The structure and parameters of controllers are optimized simultaneously using competent operators/algorithms to search for controllers with satisfactory indicators. The property of the coupled mixed-variable aggravates the difficulty of the optimization of controller, reducing the chance of finding feasible solutions. The simulation-based evaluation in the proposed method is independent of the system model, which makes it become a general method but time-consuming. In this sense, we can only rely on very limited evaluations to find satisfactory desirable controllers. To validate the effectiveness of this method, it is applied to a magnetic levitation system with external disturbance. A number of independent simulations and experiments are

carried out and the results have demonstrated the effectiveness and practicability of the proposed method. For simplicity, the controller determined by the method with optimized structure and parameters is named COSP. The main contributions of this paper can be summarized as follows:

- A general automatic controller design method is proposed to achieve the optimization of expected performance indicators via flexibly combining basic controller components and tuning their parameters. The design of controllers is transformed into a mixed-variable optimization problem involving continuous and discrete decision variables. Driven by performance indicators, the structure and parameters are evolved iteratively and cooperatively using competent evolutionary algorithms respectively. Compared with traditional controller design methods, this method can not only realize the automatic tuning of parameters, but also search for a wealth of controller structures.
- Compared to the previous work [25], a generalized encoding/decoding scheme, which considers more abundant controller structures, is proposed to represent controller solutions incorporating integer and real-valued vectors. Integer vectors are employed to represent the connection relationship of the given components and real-valued vectors are adopted to represent the corresponding parameters of the given components. The proposed encoding/decoding scheme facilitates the application of intelligent optimizers in searching for high-quality solutions.
- Since the structure of the COSP is more abundant and comprehensive, a set of generation rules for the controller structure are put forward to exclude some unreasonable or infeasible structures. These rules can transform mixed-variable solutions into effective controllers while ensuring sufficient expression ability of solutions.

The rest of this paper is structured as follows: Section II presents the controller optimization problem; Section III presents the detailed description of the proposed controller design method; Section IV carries out simulations and experiments to validate the proposed method; and Section V concludes the whole paper.

II. PROBLEM DESCRIPTION

The controller design method proposed in this paper takes the performance indicators of the control system as the optimization objective, and designs the control law directly for the system in time domain. It can describe the dynamic characteristics of the system more completely, and has a clear design meaning and practical significance. The automatic design of a controller can be formulated as an optimization problem as follows:

$$\begin{aligned} \min_c \quad & J = f(p, c) \\ \text{s.t.} \quad & g_i(p, c) \leq 0, \quad i = 1, 2, \dots, m, \end{aligned} \quad (1)$$

where $f(p, c)$ represents the functional relationship among the objective function value J , the control object p , and the controller c . The control object p is usually depicted by linear or nonlinear dynamic equations. The objective function

f is usually chosen as the performance indicator of the control system, such as the integral of absolute error (IAE), the integral of the time-weighted absolute error (ITAE), the integral of the squared error (ISE), and the integral of the time-weighted squared error (ITSE) [26]. $g_i(p, c) \leq 0$ is the constraint in control system, and m is the number of the constraints. The structure of a controller is parameterized as a discrete vector to denote the connection relationship of the given components. Thus, Eq. (1) can be rewritten as follows:

$$\begin{aligned} \min_{K_{stru}, K_{para}} \quad & J = f(p, K_{stru}, K_{para}) \\ \text{s.t.} \quad & g_i(p, K_{stru}, K_{para}) \leq 0, \quad i = 1, 2, \dots, m, \end{aligned} \quad (2)$$

where K_{stru} stands for the structure variable of a controller and it is a discrete vector; K_{para} is the parameter vector of a controller. Thus, the design of controllers is transformed into a mixed-variable optimization problem involving continuous and discrete decision variables.

Remark 1: It should be noted that the calculation of a performance indicator itself may be very complicated for control systems, such as ITAE and ITSE. Thus, simulation-based numerical computation methods are applied in control performance evaluation. However, different indicators have various properties and their properties may be uncertain. A performance indicator function may be non-linear, non-analytic, non-convex, discontinuous, multi-modal, or lack of gradient information. As a result, there are plenty of non-numeric solutions caused by unstable controller performance and infeasible solutions in the controller space. Besides, the property of the mixed-variable also aggravates the difficulty of solving the problem, reducing the chance of finding analytic solutions. Hence, more general methods are required to find the optimal solution through iterative search.

III. PROPOSED CONTROLLER DESIGN METHOD

A. Controller Components

The selection of controller components plays an irreplaceable role in the controller design and should be designed properly before the application of the proposed method to ensure the quality of the obtained controllers. The controller component refers to the basic units of a controller and rich components contribute to the variety of controllers. However, the number of components will directly determine the dimension of the solution, which will affect the complexity of the solving process. Therefore, the number of components should be determined to maintain a trade-off between the diversity of structure and computational complexity. Besides, the selection of components is critical to control performance. Thus, the number and types of components should be determined first.

TABLE I

SOME COMMONLY USED ATOMIC AND COMPOSITE COMPONENTS.

Components	Proportional	Integral	Differential
Transfer function	k_p	k_i/s	$k_d s$
Components	Time-delay	Double integral	Double differential
Transfer function	$e^{-s\tau}$	s^2	s^{-2}
Components	First-order inertial element	...	
Transfer function	$\frac{1}{Ts+1}$...	

Apparently, since control objects are various in actual systems, it is often difficult to determine what components are needed and the choice of components is often determined by prior knowledge. Thus, the knowledge of the object is indispensable to the selection of controller components, which is usually highly dependent on human beings. Some commonly used components are listed in Table I which can be chosen to constitute a controller. Also, some complicated and integrated combinations of components such as ADRC [15] and CRTC [27] can be taken as an alternative component by experience.

B. Representation of Controller Solution

In practical applications, a controller can be regarded as a combination of basic controller components. The basic connection relationships between components mainly include the serial, parallel, and feedback connections. Besides, every component has at least one parameter. Three typical examples are presented in Fig.1. Once the number, types, parameters, and connection relationship of controller components are determined, the controller will be determined. Therefore, with given components, the connections among the input and output of each component and the input and output of the controller can represent the structure of a controller. The proposed encoding/decoding scheme provides a genetic metaphor for controllers and enables the evolution of controllers.

Experience suggests that the input to a controller is typically the desired input x and the output y of the system as feedback, which can be designed as needed. Here, the input of a controller is designed as the above two variables, and the output of a controller is u . For an SISO system, if k components $A_i (i = 1, 2, \dots, k)$ are selected and their connection relationships are arbitrary, then their connections can be presented by a table with $k+2$ rows and $k+1$ columns, as shown in Table II.

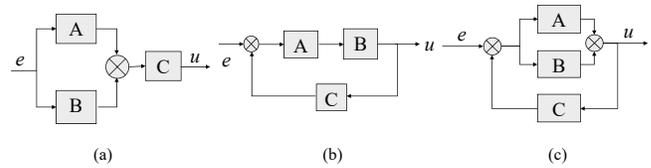


Fig. 1. Some Examples of the Controller Structure. (a) Parallel-series combination. (b) Series-feedback combination. (c) Parallel-feedback combination.

TABLE II

THE INPUT-OUTPUT CONNECTION RELATIONSHIPS OF k COMPONENTS.

Port	$A_{1,IN}$	$A_{2,IN}$...	$A_{k,IN}$	Output u
Input x	1\ -1\ 0	1\ -1\ 0	...	1\ -1\ 0	1\ -1\ 0
$A_{1,OUT}$	1\ -1\ 0	1\ -1\ 0	...	1\ -1\ 0	1\ -1\ 0
$A_{2,OUT}$	1\ -1\ 0	1\ -1\ 0	...	1\ -1\ 0	1\ -1\ 0
...
$A_{k,OUT}$	1\ -1\ 0	1\ -1\ 0	...	1\ -1\ 0	1\ -1\ 0
System output y	1\ -1\ 0	1\ -1\ 0	...	1\ -1\ 0	1\ -1\ 0

As shown in Table II, $A_{i,IN}$ and $A_{i,OUT}$ represent the input and output of the i th component, respectively. The horizontal head of the table is the Input x , the output of each component $A_{i,OUT}$ and the system output y . The vertical head of the

table is the input of each component $A_{i,IN}$ and the controller output u . Each element in the table represents the connection relationship of the two corresponding ports. Specifically, value 1 represents a positive connection, -1 indicates a negative connection, and 0 stands for no connection. The connection relationship of these ports can be determined row by row. There are totally $k^2 + 3k + 2$ effective elements. So that, once the values of these elements are fixed, the controller structure will be determined. That is, the structure of the controller can be represented by a discrete vector of $k^2 + 3k + 2$ dimensions. Hence, after the parameterization of the structure, a controller solution can be represented as follows:

$$\mathbf{X} = \underbrace{[X_1, X_2, \dots, X_{d1}]}_{\text{parameter variable}} \underbrace{[X_{d1+1}, X_{d1+2}, \dots, X_{d1+d2}]}_{\text{structure variable}} \quad (3)$$

where $d1$ is the dimension of the parameter variable, namely the total number of the parameters of the k components; $d2$ is the dimension of the structure variable and $d2 = k^2 + 3k + 2$.

TABLE III
THE INPUT-OUTPUT CONNECTION RELATIONSHIPS OF THE CONTROLLER SHOWN IN FIG.1(C).

Port	A_{IN}	B_{IN}	$A_{k,IN}$	Output u
Input e	1	1	0	0
A_{OUT}	0	0	1	1
B_{OUT}	0	0	1	1
C_{OUT}	-1	-1	0	0

As a simple example, the structure of the controller shown in Fig.1(c) can be presented by a table as shown in Table III. Moreover, the structure solution can be represented by a 16-dimensional discrete vector $X_{(struc)} = [1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 1, -1, -1, 0, 0]$.

C. Generation Rules for Controller Structure

As mentioned above, the structure variable of the controller is a discrete vector, and each element has three values of 0/1/-1. When the length of the vector is h , the size of the structure solution space is 3^h . It is apparent that the size of the solution space will increase exponentially with h , which will be extremely large. Besides, in this huge structure solution space, there may not be a connected channel between the input and output of the controller or not all the connections are significant. Thus, there are a large number of infeasible or unreasonable structures, and it is necessary to design a set of generation rules to remove these unnecessary structures and improve the solving efficiency. The generation rules cannot only compress the structure solution space, but also ensure the feasibility of generated structures.

TABLE IV
NOTATION DESCRIPTION

S_{Com} : Component state vector. If a component is connected, $S_{Com}(i) = 1$; otherwise, $S_{Com}(i) = 0$, $i = 1, 2, \dots, N_{Com}$.
S_{Edge} : Edge state vector. If there is a connection edge between certain two ports, $S_{Edge}(j) = 1$; otherwise, $S_{Edge}(j) = 0$, $j = 1, 2, \dots, N_{Edge}$.
$S_{ExpPort}$: Export port state vector. If it is an available export port, $S_{ExpPort}(k) = 1$; otherwise, $S_{ExpPort}(k) = 0$, $k = 1, 2, \dots, N_{Com} + 1$.
S_{InPort} : Input port state vector. If it is an available input port, $S_{InPort}(k) = 1$; otherwise, $S_{InPort}(k) = 0$.
S_{Cur} : Current connected components vector. If a component is connected in the current channel, $S_{Cur}(i) = 1$; otherwise, $S_{Cur}(i) = 0$.

In the proposed generation rules for controller structure, control channels of a controller are divided into the main

channel and several branch channels. The main channel is the signal path between the input and output ports of a controller and branch channels refer to the signal loops within the controller. The following rules are adopted to construct the structure of a controller: (1) The main channel will be constructed first and then the branch channels; (2) The number of components will be constructed first and then the types of components; (3) Determine the components first and then the connection edges.

TABLE V
STEPS OF THE CONTROLLER STRUCTURE GENERATION.

Step 1:	Initialize the related state vectors. Set the number of the remaining edges as $NE_{re} = N_{Edge}$.
Step 2:	Determine the number of components in the main channel (N_{Cmc}) by roulette approach. Then select N_{Cmc} components randomly from the given components and connect them with the input and output ports end to end. Update the related state vectors. $NE_{re} = NE_{re} - N_{Cmc} - 1$.
Step 3:	If $NE_{re} > 0$, then select an available export port randomly as the start of the current edge. If $NE_{re} > 1$, go to Step 4. Otherwise, skip to Step 7.
Step 4:	Generate a random real number (α) in $[0, 1]$. If $\alpha > 0.5$, skip to Step 5. Otherwise, skip to Step 6.
Step 5:	Select an unconnected component randomly to build a branch channel, then go to Step 7. $NE_{re} = NE_{re} - 1$.
Step 6:	Select an available input port randomly as the end of the current edge. Determine the value of the connected edge according to the connection probability matrix and update the related state vectors. $NE_{re} = NE_{re} - 1$.
Step 7:	If $NE_{re} > 1$, then return to Step 4. If $NE_{re} = 1$, then return to Step 6. If $NE_{re} = 0$, then stop.

Once the number of components (N_{Com}) is determined, the total number of connection edges (N_{Edge}) of the controller can be determined as aforementioned and $N_{Edge} = d2$. During the construction of the controller structure, some state vectors are used to record the state of each component and each edge as listed in Table IV. With the given components, the structure of a controller can be determined as listed in Table V.

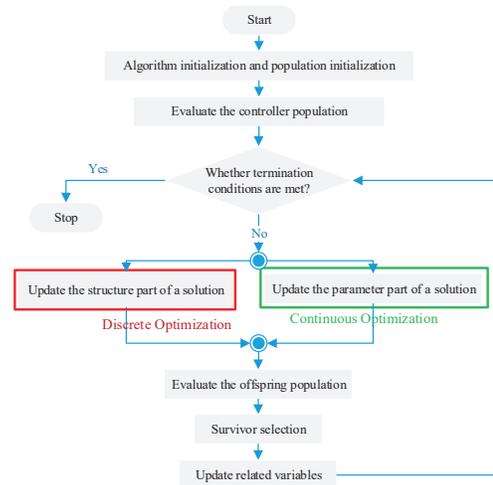


Fig. 2. The process of the optimization of controllers.

D. Process of the COSP Design

Before the implementation of this method, the number and types of components should be determined based on the property of the control object, considering both the structure diversity and computational complexity. Then, the structure and parameters of a controller are evolved synchronously

by intelligent algorithms, while viewing the structure and parameters as an organic whole during the evaluation process of solutions (as shown in Fig. 2). Firstly, the population will be initialized randomly. Then, JADE (mainly including mutation and crossover operators) will be carried out to evolve the parameter part of solutions and EDA will be executed to update the structure part of solutions iteratively using the generation rules proposed in Sec. III. C. Then, the better half of the parent and offspring solutions will be propagated into the next generation. Furthermore, the proposed method will rebuild the population when the genetic diversity of the current population reduces to a certain level. The population will be evolved iteratively until the termination criterion is met.

IV. SIMULATION AND EXPERIMENT RESULTS

In order to validate the effectiveness of the proposed design method, the method is applied to a magnetic levitation system (MLS), which is a typical nonlinear system, with external disturbance. The structure diagram of MLS is shown in Fig. 3. Moreover, both simulations and physical experiments of the MLS are carried out in this paper. Since the parameters of the actual plants vary from one to another, controllers designed by the proposed method are compared with advanced control algorithms only in the simulation.

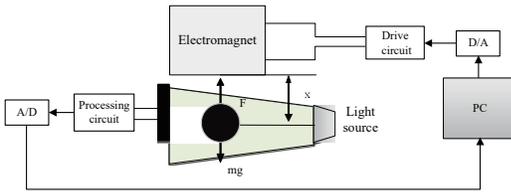


Fig. 3. The structure diagram of the MLS.

A. Mathematical Model of the Controlled Object

The magnetic levitation system (MLS) is a kind of single-degree-of-freedom control system. As a nonlinear system, it is a typical object to verify various control algorithms, especially nonlinear control algorithms [28], [29]. In order to control a magnetic levitation system, it is usually demanded to get a linear model by linearizing the model near the equilibrium point of the system, and then a wide variety of controllers have been designed [27], [30], [31]. Hence, the method proposed in this paper is applied to MLS to verify its performance. The following dynamic model of MLS is considered in this paper:

$$\begin{cases} m \frac{d^2 x(t)}{dt^2} = F(i, x) + mg \\ mg + F(i_0, x_0) = 0 \\ F(i, x) = K \left(\frac{i}{x}\right)^2 \end{cases} \quad (4)$$

where $K = -\mu_0 K_f A N^2$, x is the air gap between the center of mass of the ball and the magnetic pole of the electromagnet, m is the mass of the ball, $F(i, x)$ is the electromagnetic force, μ_0 is magnetic permeability of vacuum, A is the cross-sectional area of the magnetic circuit, and N is the number of the electromagnet coils. Besides, i_0 and x_0 are the coil current and the air gap at the time when the ball is in equilibrium.

B. Setting of the Controller Design Method

For a successful application of the proposed design method to MLS, some key elements in this method are determined for the controlled object as follows:

1) *Component Selection*: In the controller design for MLS, the CRTC [27] component (A_1), the proportion component (A_2), the integral component (A_3), and the differential component (A_4) are selected to constitute the controller. The CRTC component is mainly used to enhance the robust performance of the control system. Among them, the input of the CRTC component is fixed, and its output is directly connected to the output of the controller. Besides, the input of the controller includes input signal $Input\ x$ and system output as Feedback signal f . So a solution of this controller optimization problem has 7 controller parameter elements and 24 structure elements. Use X_1, X_2, \dots, X_7 to denote the proportion, integral, differential components and the CRTC parameters respectively, and use X_8, X_9, \dots, X_{31} to represent structure variable.

2) *Selection of the Optimization Objective*: In order to validate the practicability and effectiveness of the proposed method, two different kinds of optimization objectives are designed to observe that whether the obtained COSPs can meet various design goals as follows:

- ITAE: $J = \int_0^\infty t |e(t)| dt$;
- the IAE and overshoot indicators of the control system output are used as constraints, and the rise time indicator is taken as the optimization objective.

3) *Optimizer Selection*: Two powerful intelligent methods, namely, the estimation of distribution (EDA) algorithm [32] and JADE [33], are employed to optimize the structure and parameter solutions, respectively. The population size is set as 100 and the maximum number of iterations is set as 800 for each algorithm. Besides, some specific parameters are set as listed in Table VI.

TABLE VI
PARAMETER SETTINGS OF INTELLIGENT ALGORITHMS.

Algorithms	Parameters	Values
JADE	Range of parameter optimization	$10^{[-3,3]}$
	The initial scaling factor F	0.5
	The initial crossover probability CR	0.9
EDA	Learning ratio	0.7

- The differential evolution (DE) algorithm is a very simple but very powerful population-based numerical optimization algorithm, which has been proved by great deals of applications and the performance testing and comparison [9]. JADE, which is an advanced adaptive variant of DE, introduces the DE/current-to-pbest/1 mutation scheme with the optional external archive and the adaptive mechanism of parameters into the basic DE. Therefore, JADE is employed to optimize the controller parameters in this application.
- The EDA algorithm is a widely-used and effective method to solve the complex combinatorial optimization problem [32]. It describes the distribution information of promising solutions by building probability models and evolves the population by repeatedly modeling and sampling. As a classical EDA, the population-based incremental learning

(PBIL) algorithm [34] has been proved to be highly effective on plenty of stationary benchmark and real-world problems. Considering that the controller structure variable is a discrete vector, PBIL is used to optimize the structure of the controller.

Remark 2: These two algorithms are taken as an example of the optimizer selection, and they are used to validate the effectiveness of the proposed controller design method. Other optimization algorithms can be used to replace them as needed. Due to limited space, we just present some brief introductions to these algorithms and the parameter setting. More details of these algorithms are available in [32], [33].

4) *Constraint Setting:* According to the requirements on the input and the output constraints of the actual system, set the limiting modules “Saturation1” of 0 ~ 10V and “Saturation2” of -10 ~ 0V, as shown in Fig. 5.

C. Simulation of the Magnetic Levitation System

To validate the effectiveness of the proposed controller design method, the following simulations are presented in 3 different cases. After the setting of components, objectives, and optimizers, the proposed controller design method is conducted with MATLAB/Simulink to generate the COSP controllers with satisfactory indicators automatically. In MATLAB/Simulink, the controller model for application to MLS is built as shown in Fig. 4. As shown this figure, 31 elements (i.e., k_1, k_2, \dots, k_{31}) need to be optimized to get a controller with satisfactory expected performance indicators. The algorithm will run multiple times independently for each design goal and the results will be statistically analyzed. As space is limited, we only present some typical results here, more detailed results are shown in the supplemental material.

In this section, some performance indicators are adopted to evaluate the tracking performance of controllers. Specifically, t_r indicates the rise time, t_s indicates the settling time, *overshoot* indicates the overshoot, *ess* indicates the steady state error, *ess1* indicates the mean steady state error of the entire signal, *ess2* indicates the average steady state error of second half in each step period of the square wave signal, *IAE* indicates the integral of the absolute value of the error with time, and *ITAE* indicates the integral of the product of the absolute value of the error and the time.

1) *Case 1:* To verify the validity of the novel generalized encoding/decoding scheme, a step-reference signal of -7V to -5V is taken as the reference signal (see Eq.(5)) and the design method proposed in [25] is taken as a competitor in this section. Besides, the 2-DOF-PID control [35] is also involved in the comparison.

$$y_{r1}(t) = \begin{cases} -5V, & t \geq 10; \\ -7V, & t < 10. \end{cases} \quad (5)$$

The simulation diagram of the control system is shown in Fig. 5 and the nonlinear model of MLS is shown in Fig. 6. The considered system model parameters are given in Table VII. The simulation settings for all these methods are set as the same as that in [25] for a fair comparison, namely, the Runge-Kutta method is utilized and the sampling time is 3ms.

The simulation and statistical results are presented in Table VIII where these COSPs are designed taking ITAE as the optimization objective. The output curves of COSP1 and the competitors are given in Fig.7. As shown in Table VIII, all the indicators of the COSPs are better than those of 2-DOF-PID and the controller designed in [25], except for *overshoot* and *ess*. Nevertheless, more than 86.7% of the indicators of the obtained COSPs are better than those of 2-DOF-PID. Besides, more than 83.7% of the indicators are better than those of the controller designed in [25]. It can also be found through simulations that COSP1 has better tracking performance than the competitors, which validates the effectiveness of the proposed encoding/decoding scheme.

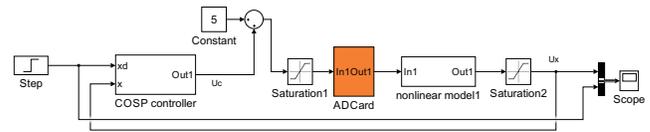


Fig. 5. The simulation diagram of the control system for Case 1.

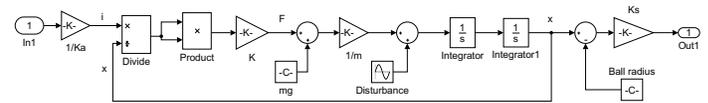


Fig. 6. The nonlinear model of MLS.

TABLE VII
MODEL PARAMETERS.

Parameter	Value	Parameter	Value
m	22g	x_0	20mm
Iron core diameter	$\Phi 22\text{mm}$	Enameled wire diameter	$\Phi 0.8\text{mm}$
Coil resistance	13.8 Ω	Ball radius	12.5mm
N	2450	K	$2.3142e - 4\text{Nm}^2/\text{A}^2$
i_0	0.6105A	K_f	0.25

TABLE VIII
STATISTICAL RESULTS OF PERFORMANCE INDICATORS FOR CASE 1.

controller	t_r/s	t_s/s	overshoot	ess	IAE	ITAE	T_c/h
2-DOF-PID [35]	1.7910	2.2260	0.0000	0.0000	7.3452	23.3659	N.A.†
SOC [25]	0.1200	0.2220	0.0637	0.0001	0.5548	1.6578	4.4019
COSP1	0.0480	0.1200	0.2193	0.0000	0.3601	0.5848	1.319
COSP2	0.1170	0.2040	0.0000	0.0000	0.3243	1.2720	1.312
COSP3	0.0510	0.0510	0.3925	0.0000	0.2762	0.4793	1.364
COSP4	0.0630	0.0750	0.0000	0.0034	0.4855	2.9470	1.315
COSP5	0.0510	0.0510	0.7084	0.0000	0.2553	0.4609	1.316

† Since the 2-DOF-PID control method was designed through theoretical analysis based on the plant characteristics, the time cost to obtain the controller is not available.

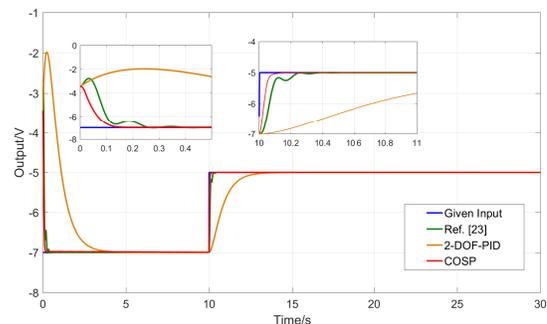


Fig. 7. The system output curves for Case 1.

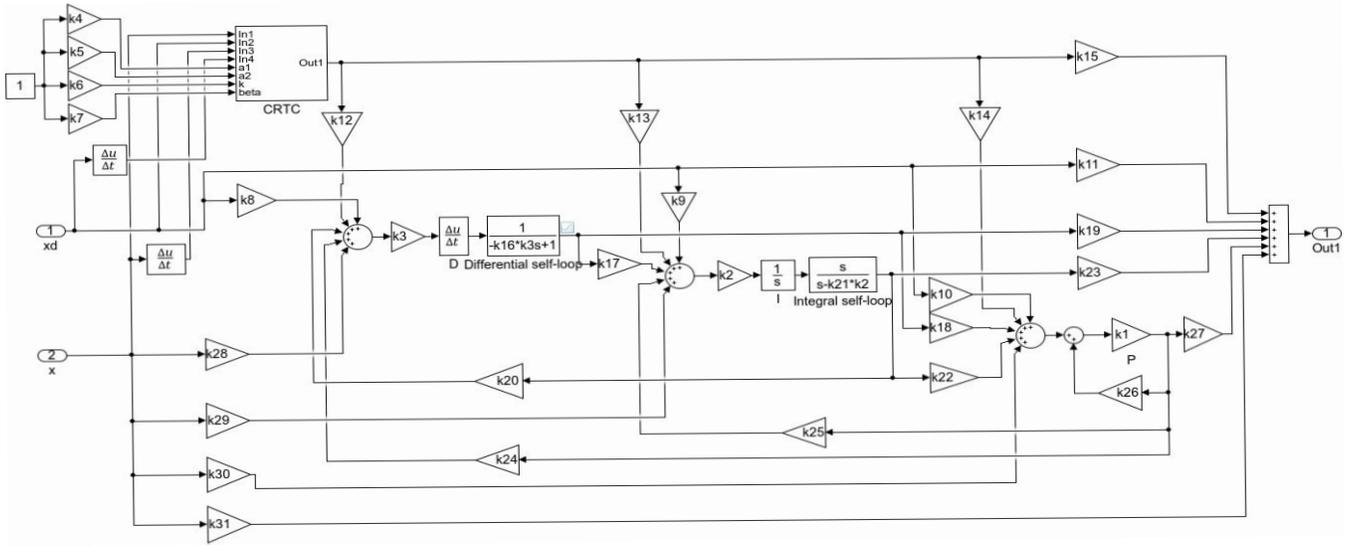


Fig. 4. The model diagram of 4-components controllers for MLS.

2) *Case 2*: To validate the superiority of the method in anti-interference performance, an existing state of art control, i.e., the adaptive linear ADRC method [18], is adopted as a competitor. A step signal from 0mm to 10mm is input as the reference signal(see Eq.(6)) and a 20N external interference force will be applied to the plant at 2s.

$$y_{r2}(t) = \begin{cases} 10\text{mm}, & t \geq 0; \\ 0\text{mm}, & t < 0; \end{cases} \quad (6)$$

The simulation diagram of the control system is shown in Fig.8. The control plant is implemented by Matlab S-function. All the model parameters and simulation settings are set as the same with those in [18] and the sampling time is set as 1ms.

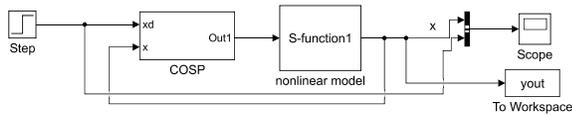


Fig. 8. The simulation diagram of the control system for Case 2.

TABLE IX
STATISTICAL RESULTS OF PERFORMANCE INDICATORS FOR CASE 2.

controller	IAE	ITAE	ITSE	T_c/h
LADRC	7.611E-04	7.251E-04	7.572E-07	n.a.†
A-LADRC	7.217E-04	1.348E-04	2.001E-07	n.a.†
COSP1	3.604E-05	5.595E-06	1.542E-07	1.522
COSP2	3.512E-05	7.105E-06	1.543E-07	1.486
COSP3	2.548E-05	4.167E-06	1.499E-07	1.635
COSP4	4.816E-05	1.144E-05	2.470E-07	1.818
COSP5	3.608E-05	6.882E-06	1.568E-07	1.553
COSP6	2.548E-05	5.307E-06	1.567E-07	1.621
COSP7	2.272E-04	1.483E-05	1.309E-06	1.339
COSP8	4.401E-05	5.439E-06	1.776E-07	1.382
COSP9	3.600E-05	6.749E-06	1.574E-07	1.591
COSP10	3.662E-05	6.757E-06	1.552E-07	1.372

† Since LADRC and A-LADRC were designed through theoretical analysis based on the plant characteristics, the time cost to obtain the controller is not available.

The simulation results are presented in Fig.9 and Table IX. These COSPs are designed by the proposed method taking ITAE as optimization objective. As can be seen in Table IX, all the IAE and ITAE indicators of the COSPs are better than those of the competitors. More than 93.3%

of the evaluation indicators of the obtained COSPs prevail over those of A-LADRC. Besides, more than 96.7% of the indicators of obtained COSPs are better than those of LADRC. Furthermore, the output curves of COSP1 and the competitors are given in Fig.9. It can be found that COSP1 has better anti-interference performance than LADRC and A-LADRC.

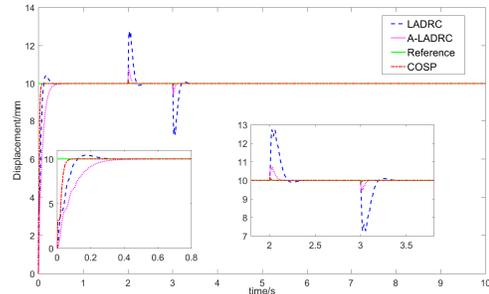


Fig. 9. The system output curves of Case 2.

3) *Case 3*: In this section, COSPs are designed by intelligent algorithms for an MLS with an external acceleration disturbance $d(t) = 2 \sin(4t) \text{ m/s}^2$. In the simulation, the parameters of the MLS, the desired input signal, and the initial conditions are set as the same as those in [27] ($m = 0.54\text{kg}$, $x_\infty = 0.007987\text{m}$, $Q = 0.001624\text{H}\cdot\text{m}$) for a fair comparison. The initial position of the ball is set as 0.02m, and the initial velocity is set as 0.01m/s. As what [27] did, the reference signal is generated by filtering a rectangular wave with a period of 4 s through a third-order filter $G_f(s)$ depicted as

$$\frac{60 \times 70^2}{(s + 60)(s + 70)^2}. \quad (7)$$

The rectangular wave is given as follows:

$$y_{r3}(t) = \begin{cases} 12\text{mm}, & 4k \leq t \leq 4k + 4 (k = 0, 2, 4, \dots); \\ 4\text{mm}, & 4k \leq t \leq 4k + 4 (k = 1, 3, 5, \dots). \end{cases} \quad (8)$$

The simulation diagram of the control system is similar with that of Case 1. The simulation and statistical results are shown in Table X, and Table XI corresponding to the aforementioned two optimization objectives, respectively. The bold part indicates that the index of COSP is better than that of CRTC, otherwise the part is shaded. To validate the effectiveness of the simultaneous optimization of the structure and parameters, a parallel combination of PID and CRTC (denoted by CRTC_PID) with fixed structure is considered in the comparison.

TABLE X
STATISTICAL RESULTS OF PERFORMANCE INDICATORS. CRTC* REFERS TO THE CRTC WITH THE OPTIMIZED PARAMETERS.

controller	tr/s	ts/s	overshoot	ess1	ess2	IAE	ITAE	T_c/h
COSP1	0.0826	0.2404	0.1459	1.4312E-04	1.2237E-06	4.2857E-03	2.1751E-03	3.3539
COSP2	0.0820	0.2120	0.1765	1.1810E-04	1.5593E-06	3.5351E-03	2.0036E-03	3.3162
COSP3	0.0816	0.2060	0.2263	1.0867E-04	2.3843E-06	3.2523E-03	2.3135E-03	3.1548
COSP4	0.0826	0.1353	0.0800	2.5024E-05	4.0544E-07	7.4275E-04	6.3519E-04	3.4342
COSP5	0.0816	0.1194	0.0060	2.0468E-05	1.0144E-07	6.0607E-04	3.4282E-04	3.3135
COSP6	0.0816	0.1164	0.0045	1.9471E-05	1.0847E-07	5.7614E-04	3.3742E-04	3.2730
COSP7	0.0817	0.1489	0.1095	4.0718E-05	7.9529E-07	1.2136E-03	9.0492E-04	3.4777
COSP8	0.0816	0.1830	0.1675	6.7071E-05	1.9395E-06	2.0042E-03	1.6472E-03	3.2510
COSP9	0.0816	0.1299	0.0340	3.0462E-05	2.4431E-07	9.0590E-04	5.0317E-04	3.8285
COSP10	0.0816	0.1180	0.0044	1.9650E-05	1.0481E-07	5.8152E-04	3.3645E-04	3.6785
COSP11	0.0823	0.1306	0.0944	2.1857E-05	6.1034E-07	6.4774E-04	7.2846E-04	3.5165
COSP12	0.0817	0.2664	0.2486	1.7182E-04	2.7311E-06	5.1468E-03	3.4569E-03	3.2958
COSP13	0.0816	0.1088	0.0032	1.8665E-05	1.1407E-07	5.5197E-04	3.3000E-04	3.4130
COSP14	0.0824	0.3044	0.1930	2.0600E-04	4.0833E-06	6.1723E-03	3.9655E-03	3.3820
COSP15	0.0816	0.1178	0.0052	2.0258E-05	1.8060E-07	5.9975E-04	3.7675E-04	3.3655
COSP16	0.0826	0.2414	0.1372	1.4538E-04	1.1132E-06	4.3535E-03	2.1558E-03	3.2148
COSP17	0.0816	0.1139	0.0031	1.8642E-05	1.0510E-07	5.5128E-04	3.2634E-04	3.4219
COSP18	0.0816	0.1086	0.0028	1.8316E-05	1.0735E-07	5.4151E-04	3.2422E-04	3.8609
COSP19	0.0816	0.1166	0.0008	1.3685E-05	5.0151E-08	4.0255E-04	2.2865E-04	3.5223
COSP20	0.0816	0.1649	0.2007	6.3971E-05	8.4342E-07	1.9112E-03	1.2798E-03	3.7142
CRTC_PID	0.0780	0.1085	0.5668	4.3657E-05	9.1288E-06	1.3018E-03	8.5892E-03	1.1933
CRTC*	0.0816	0.1290	0.1760	3.2458E-05	2.9154E-06	9.6578E-04	1.7701E-03	1.5406
CRTC	0.0764	0.5413	4.1767	8.8390E-05	1.6814E-05	2.6438E-03	2.6419E-02	N.A. †

† Since CRTC was designed through theoretical analysis based on the plant characteristics, the time cost to obtain the controller is not available.

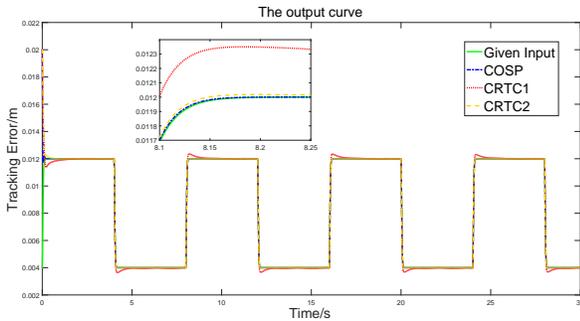


Fig. 10. The system output of COSP17 compared with CRTC. CRTC1 represents the original CRTC [27], while CRTC2 indicates the CRTC with the optimized parameters obtained using JADE.

It can be found from Table X that all the IAE , $ITAE$, t_s , $overshoot$, $ess2$ indicators of the COSPs are better than those of CRTC. In addition to t_r , only 6 COSPs have slightly worse $ess1$ and IAE . More than 77.14% of all the evaluation indicators of the obtained COSPs are superior to those of CRTC. Most indicators of COSPs outperform those of CRTC_PID, which shows the advantage of simultaneous optimization of structures and parameters. The output curves of COSP17 and CRTC are given in Fig. 10. Note that the yellow dashed line indicates the CRTC with the optimized parameters obtained using JADE (denoted by CRTC2). The performance of CRTC has been improved remarkably, except for t_r . Since $ITAE$ is a comprehensive indicator, which is calculated by time integration, it is easy to lead to a preference for the dynamic tracking characteristics of the system response, and ignoring

response speed in a certain degree. This is a normal result and phenomenon.

For better performance of COSPs, the objective function with constraints is adopted, in which IAE and $overshoot$ are used as the constraint and the rise time is used as the design goal. It is expected that controller solutions with both excellent comprehensive performance and dynamic indicators can be obtained. The statistical results with this objective function are shown in Table XI. As shown in Table XI, the t_r indicator of the solution has been improved while other indicators become slightly worse. Because there are contradictions among these indicators, when the rise time indicators become better, it will affect the real-time tracking performance of the system and cause some sacrifices of the overshoot indicator. In summary, 92.1% of the performance indicators of COSPs are superior to those of CRTC, which shows the effectiveness of the design of objective function.

TABLE XI
STATISTICAL RESULTS OF PERFORMANCE INDICATORS.

controller	t_r/s	t_s/s	overshoot	ess	IAE	ITAE	T_c/h	
COSP1	0.0631	0.2699	3.6884	9.6121E-05	3.7798E-06	2.8757E-03	1.6333E-02	3.5788
COSP2	0.0656	0.1734	4.0410	6.4670E-05	1.3968E-05	1.9321E-03	1.4604E-02	3.6451
COSP3	0.0606	0.1446	3.7625	6.4905E-05	1.6035E-05	1.9392E-03	1.5471E-02	3.7198
COSP4	0.0607	0.1318	2.8197	6.4687E-05	2.0033E-05	1.9327E-03	1.7104E-02	3.7116
COSP5	0.0647	0.1694	4.0246	4.5881E-05	6.0045E-06	1.3685E-03	1.2632E-02	3.6032
COSP6	0.0617	0.1474	3.8498	6.0745E-05	1.5007E-05	1.8144E-03	1.6249E-02	3.8658
COSP7	0.0664	0.1480	3.9031	6.7311E-05	3.2029E-05	2.0114E-03	2.6479E-02	3.6593
COSP8	0.0594	0.0825	1.0409	5.0144E-05	9.1603E-06	1.4964E-03	1.3592E-02	3.5133
COSP9	0.0647	0.1368	2.0859	5.9700E-05	1.9216E-05	1.7830E-03	1.5399E-02	3.4954
COSP10	0.0650	0.1924	4.0686	8.2156E-05	1.2597E-05	2.4568E-03	1.4524E-02	3.8478
COSP11	0.0660	0.2006	3.5094	9.4722E-05	3.3103E-05	2.8338E-03	2.2126E-02	3.6820
COSP12	0.0650	0.1831	4.0424	6.0778E-05	4.2924E-06	1.8154E-03	1.1571E-02	3.7918
COSP13	0.0609	0.1446	3.0447	7.2851E-05	1.9283E-05	2.1776E-03	1.6718E-02	3.7443
COSP14	0.0656	0.1568	2.9994	5.0644E-05	1.4416E-05	1.5114E-03	1.4331E-02	3.5744
COSP15	0.0650	0.1899	4.0972	6.8312E-05	1.2537E-05	2.0414E-03	1.4246E-02	3.5126
COSP16	0.0644	0.1703	3.5833	7.6124E-05	3.6361E-05	2.2758E-03	2.3027E-02	3.5337
COSP17	0.0651	0.1841	4.0738	6.9182E-05	8.3747E-06	2.0675E-03	1.2850E-02	3.5679
COSP18	0.0659	0.1779	4.0557	4.9320E-05	6.1491E-06	1.4716E-03	1.3245E-02	3.6282
COSP19	0.0666	0.2829	3.9442	6.2025E-05	1.5562E-05	1.8528E-03	1.6187E-02	3.7533
COSP20	0.0674	0.1730	3.5311	4.5405E-05	4.7320E-06	1.3542E-03	1.0948E-02	3.6584
CRTC	0.0764	0.5413	4.1767	8.8390E-05	1.6814E-05	2.6438E-03	2.6419E-02	N.A. †

† Since CRTC was designed through theoretical analysis based on the plant characteristics, the time cost to obtain the controller is not available.

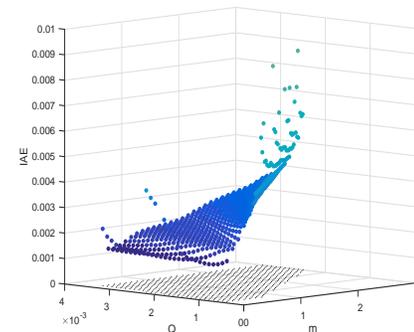


Fig. 11. The parameter sensitivity analysis of the plant parameters.

In general, when the design goal is determined, the method can stably get effective controllers by optimization with the objective as the search direction, as shown in these simulations. Thus, the setting of the objective function will directly affect the design results. As long as the objective function is set, the algorithms can automatically generate controller solutions that meet the design objectives well.

D. Experiments of the Magnetic Levitation System

To further validate the performance of the proposed controller design method, physical experiments are implemented

on a laboratory-scale MLS manufactured by Googol Technology Ltd as depicted in Fig. 12. The experiment setup consists of the mechanical unit (an electromagnet, a metallic ball, and a LED-optoelectronic sensor) and the control interface.

In this setup, the feedback signal from the LED-optoelectronic sensor to the computer and control signals from the computer to the physical system are sent via PCI 1711 A/D card made by Advantech Technologies. The real-time control algorithm in the experiment is implemented by Matlab/simulink software, which contains Real-Time Workshop (RTW) tool set. The real-time experiment environment of the MLS testbed is presented in Fig. 13. The parameters of the actual MLS are the same as listed in Table VII. The environment of the practical control system of MLS is set as follows.

- 1) Software version: MATLAB R2009a.
- 2) Algorithm settings: fixed step, ode4 (Runge-Kutta), and the sampling time is 0.003s.
- 3) Run time: 0 seconds start, manual stop.

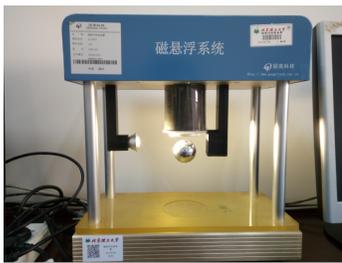


Fig. 12. MLS testbed from Googol Technology Ltd.

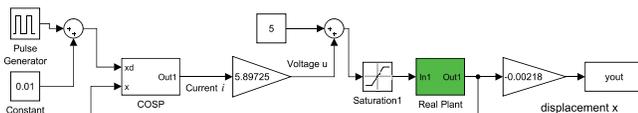


Fig. 13. Experiment environment of the MLS testbed.

First, COSPs are designed by the proposed method for the actual control plant taking IAE as optimization objective. Then, apply the COSP obtained to the experimental platform where the ball is expected to move in a range of [0.01m, 0.014m] and the given input signal is a rectangular wave with a period of 9s. An obtained COSP is taken as an example and applied to the actual plant. The selected solution is

$$X = [0.002, 0, 0.0245, 27.7534, 33.6203, 1.1714, 25.8047, 1, 0, 1, 0, -1, 0, 0, -1, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, -1, -1].$$

Thus, the corresponding simulation model built in MATLAB/Simulink for this solution is presented in Fig. 14.

Fig. 15 demonstrates the distinction of the response curves in the simulation and experiment. More specifically, in the actual control system, $t_r = 0.0045s$, $t_s = 1.5780s$, $ess = 1.9692E - 05$, $IAE = 4.5345E - 03$, and $ITAE = 7.1301E - 02$. Most of the indicators are of the same order as the simulation system. Comparing the simulation and experi-

ment output results within 0 to 36s, the mean absolute error of them can be calculated as shown in the following formula.

$$Err_{mean} = \sum_{i=1}^n |Y_{i,sim} - Y_{i,real}| / n \approx 1.6507E - 04$$

where n is the sample size, Y_{sim} is the simulation output data, and Y_{real} is the experiment output data.

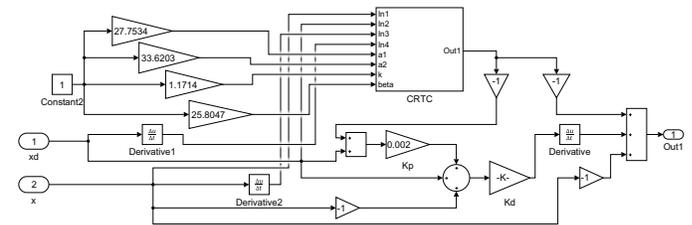


Fig. 14. The COSP controller applied in the MLS testbed.

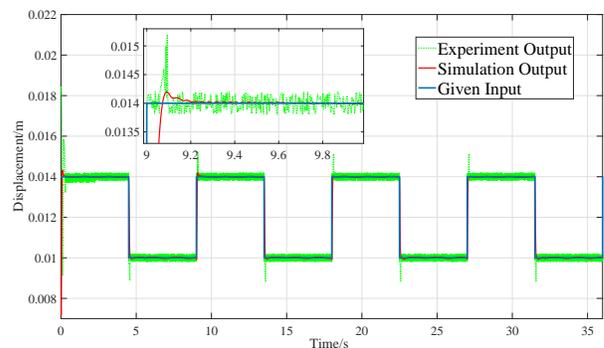


Fig. 15. The output of the simulation system and the actual system.

As can be seen in Fig. 15, the output curve of the actual system is basically the same with that of the simulation system. The main difference between the simulation output and experiment output is that some minor jitter can be found in the experiment output and the overshoot of the experiment output is larger. The reason behind this phenomenon might be that there are some internal and external disturbances in the actual system. Note that the controller by simulation optimization proposed in this paper can be directly applied to the actual control system without adjustment, which verifies the feasibility and practicability of the proposed method. Besides, when the disturbance is imposed on the object, the experimental result is shown in the video available at <https://github.com/Ovinton32roc/COSP>.

V. CONCLUSION

In this paper, controllers are designed automatically in an evolutionary manner under a generalized structure encoding/decoding scheme. The design of controllers is formulated as a mixed-variable optimization problem. A generalized encoding and decoding scheme is proposed to represent a controller. The structure and parameters of a controller are optimized simultaneously under the framework of evolution. Besides, a set of generation rules for the controller structure is proposed to remove some unreasonable and infeasible structures, which can also ensure the feasibility of solutions. Moreover, the proposed design method is applied to a typical

nonlinear system, namely a magnetic levitation system (MLS) with external disturbance. The validity of the proposed controller design method has been proved by the simulations and experiments about MLS. Comparing the numerical results with those of the state-of-the-art control methods, the performance indicators have been remarkably improved, which proves the practicability and effectiveness of the proposed design method. The proposed method can not only shorten the design cycle, but can also reduce the dependence of controller design on human beings.

The proposed method can easily find a controller with desirable performance through automatic design. Besides, the introduction of the knowledge of control objects and the design experience in the selection of components is conducive to improving the solving efficiency of the proposed method. The number of components should be moderate so as to find controllers with satisfactory expected indicators within the acceptable computing time. The proposed method relies on the design experience of human beings to some extent. The dependence mainly lies in the selection of candidate components to constitute a controller. To further reduce the dependence on human beings, automatic selection of candidate components deserves further research in the future. It is worth noting that the choice of intelligent optimization algorithms is not the focus of this paper. The EDA and JADE algorithms used in the simulation and experiment are taken as a choice with good performance after a comparison. Undoubtedly, they can be replaced by other competent algorithms as needed. In addition, more applications to diverse control objects will be considered in future work.

ACKNOWLEDGMENT

The authors would like to thank Prof. Kuangang Fan for providing the source code of A-LADRC for comparison.

REFERENCES

- [1] K. J. Åström and T. Hägglund, *PID controllers: theory, design, and tuning*. Research Triangle Park, NC, USA: Instrument Society of America, 1995.
- [2] K. H. Ang, G. Chong, and Y. Li, "Pid control system analysis, design, and technology," *IEEE Transactions on Control Systems Technology*, vol. 13, no. 4, pp. 559–576, 2005.
- [3] P. Shah, S. Agashe, and A. J. Kulkarni, "Design of a fractional pid controller using the cohort intelligence method," *Frontiers of Information Technology and Electronic Engineering*, vol. 19, no. 3, pp. 437–445, 2018.
- [4] S. A. Bhatti, S. A. Malik, and A. Daraz, "Comparison of pi and ip controller by using ziegler-nichols tuning method for speed control of dc motor," in *Proceedings of the 2016 International Conference on Intelligent Systems Engineering (ICISE)*, pp. 330–334. Islamabad, Pakistan: IEEE, Jan. 2016.
- [5] W. Ho, O. Gan, E. Tay, and E. Ang, "Performance and gain and phase margins of well-known pid tuning formulas," *IEEE Transactions on Control Systems Technology*, vol. 4, no. 4, pp. 473–477, 1996.
- [6] W. Ho, Y. Hong, A. Hansson, H. Hjalmarsson, and J. Deng, "Relay auto-tuning of pid controllers using iterative feedback tuning," *Automatica*, vol. 39, no. 1, pp. 149–157, 2003.
- [7] D. E. Rivera, M. Morari, and S. Skogestad, "Internal model control: Pid controller design," *Industrial & Engineering Chemistry Process Design and Development*, vol. 25, no. 1, pp. 252–265, 1986.
- [8] J. J. Grefenstette, "Optimization of control parameters for genetic algorithms," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 16, no. 1, pp. 122–128, 1986.
- [9] S. Das and P. N. Suganthan, "Differential evolution: A survey of the state-of-the-art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4–31, 2011.
- [10] M. Blondin, J. Sanchis, P. Sicard, and J. Herrero, "New optimal controller tuning method for an avr system using a simplified ant colony optimization with a new constrained nelder-cmead algorithm," *Applied Soft Computing*, vol. 62, pp. 216–229, 2018.
- [11] A. Moharam, M. A. El-Hosseini, and H. A. Ali, "Design of optimal pid controller using hybrid differential evolution and particle swarm optimization with an aging leader and challengers," *Applied Soft Computing*, vol. 38, pp. 727–737, 2016.
- [12] S. Soyguder, M. Karakose, and H. Alli, "Design and simulation of self-tuning pid-type fuzzy adaptive control for an expert hvac system," *Expert Systems with Applications*, vol. 36, no. 3, pp. 4566–4573, 2009.
- [13] S. Mathavaraj and R. Padhi, "Optimally allocated nonlinear robust control of a reusable launch vehicle during re-entry," *Unmanned Systems*, vol. 8, no. 1, pp. 33–48, 2020.
- [14] S. A. Vimala and S. Sathiyavathi, "Design of sliding mode controller for magnetic levitation system," *Computers & Electrical Engineering*, vol. 78, pp. 184–203, 2019.
- [15] J. Q. Han, "From pid to active disturbance rejection control," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 3, pp. 900–906, 2009.
- [16] W. Xue, W. Bai, S. Yang, K. Song, Y. Huang, and H. Xie, "Adrc with adaptive extended state observer and its application to air/cfuel ratio control in gasoline engines," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 9, pp. 5847–5857, 2015.
- [17] J. Yao and W. Deng, "Active disturbance rejection adaptive control of uncertain nonlinear systems: theory and application," *Nonlinear Dynamics*, vol. 89, pp. 1611–1624, 2017.
- [18] Q. Ouyang, K. Fan, Y. Liu, and N. Li, "Adaptive ladrc parameter optimization in magnetic levitation," *IEEE Access*, vol. 9, DOI 10.1109/ACCESS.2021.3062797, pp. 36 791–36 801, 2021.
- [19] G. Rigatos, P. Siano, P. Wira, K. Busawon, and R. Binns, "A nonlinear h-infinity control approach for autonomous truck and trailer systems," *Unmanned Systems*, vol. 08, no. 01, pp. 49–69, 2020.
- [20] P. Cortes, M. P. Kazmierkowski, R. M. Kennel, D. E. Quevedo, and J. Rodriguez, "Predictive control in power electronics and drives," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 12, pp. 4312–4324, Dec. 2008.
- [21] A. T. Hafez and M. A. Kamel, "Cooperative task assignment and trajectory planning of unmanned systems via hflc and pso," *Unmanned Systems*, vol. 7, no. 2, pp. 65–81, 2019.
- [22] K. S. Narendra and K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Transactions on Neural Networks*, vol. 1, no. 1, pp. 4–27, 1990.
- [23] A. S. Chopade, S. W. Khubalkar, A. Junghare, M. Aware, and S. Das, "Design and implementation of digital fractional order pid controller using optimal pole-zero approximation method for magnetic levitation system," *IEEE/CAA Journal of Automatica Sinica*, vol. 5, no. 5, pp. 977–989, 2018.
- [24] J. R. Koza, M. A. Keane, J. Yu, W. Mydlowec, and F. H. Bennett, "Automatic synthesis of both the control law and parameters for a controller for a three-lag plant with five-second delay using genetic programming and simulation techniques," in *Proceedings of the 2000 American Control Conference*, vol. 1, no. 6, pp. 453–459. Chicago, IL, USA: IEEE, Jun. 2000.
- [25] J. Zhan, B. Xin, and J. Chen, "Evolutionary design of controllers with optimized structure and its application in a maglev ball control system," in *Proceedings of the 36th Chinese Control Conference (CCC)*, pp. 2545–2550. Dalian, China: IEEE, Jul. 2017.
- [26] S.-L. Jamsa-Jounela, R. Poikonen, Z. Georgiev, U. Zuehlke, and K. Halmevaara, "Evaluation of control performance: methods and applications," in *Proceedings of the International Conference on Control Applications*, vol. 2, pp. 681–686, 2002.
- [27] Y. Zhang, B. Xian, and S. Ma, "Continuous robust tracking control for magnetic levitation system with unidirectional input constraint," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 9, pp. 5971–5980, 2015.
- [28] A. T. Tran, S. Suzuki, and N. Sakamoto, "Nonlinear optimal control design considering a class of system constraints with validation on a magnetic levitation system," *IEEE Control Systems Letters*, vol. 1, no. 2, pp. 418–423, 2017.
- [29] C.-M. Lin, Y.-L. Liu, and H.-Y. Li, "Sopc-based function-link cerebellar model articulation control system design for magnetic ball levitation systems," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 8, pp. 4265–4273, 2014.

- [30] J. de Jesús Rubio, L. Zhang, E. Lughofer, P. Cruz, A. Alsaedi, and T. Hayat, "Modeling and control with neural networks for a magnetic levitation system," *Neurocomputing*, vol. 227, pp. 113–121, 2017.
- [31] S. Yadav, S. Verma, and S. Nagar, "Optimized pid controller for magnetic levitation system," *IFAC-PapersOnLine*, vol. 49, no. 1, pp. 778–782, 2016.
- [32] P. Larrañaga and J. A. Lozano, *Estimation of distribution algorithms: A new tool for evolutionary computation*, vol. 2. Springer Science & Business Media, 2001.
- [33] J. Zhang and A. C. Sanderson, "Jade: adaptive differential evolution with optional external archive," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 945–958, 2009.
- [34] S. Baluja, "Population-based incremental learning: A method for integrating genetic search based function optimization and competitive learning," Carnegie-Mellon University Pittsburgh Pa Department Of Computer Science, USA, Tech. Rep., 1994.
- [35] A. Ghosh, T. Rakesh Krishnan, P. Tejaswy, A. Mandal, J. K. Pradhan, and S. Ranasingh, "Design and implementation of a 2-dof pid compensation for magnetic levitation systems," *ISA Transactions*, vol. 53, no. 4, pp. 1216–1222, 2014.



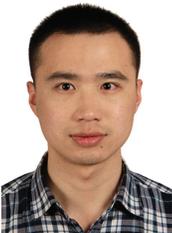
Bin Xin (Member, IEEE) received the B.S. degree in information engineering and the Ph.D. degree in control science and engineering, both from the Beijing Institute of Technology, Beijing, China, in 2004 and 2012, respectively. He was an academic visitor at the Decision and Cognitive Sciences Research Centre, the University of Manchester, from 2011 to 2012.

He is currently a Professor with the School of Automation, Beijing Institute of Technology. His current research interests include search and optimization, evolutionary computation, unmanned systems, and multiagent systems.



Yipeng Wang received the B.S. degree in automation from the Beijing Institute of Technology, Beijing, China, in 2016. He is currently pursuing the Ph.D. degree with the School of Automation, Beijing Institute of Technology, Beijing.

His current research interests include evolutionary computation and hybrid optimization algorithms.



Wenchao Xue (Member, IEEE) received the B.S. degree in applied mathematics from Nankai University, Tianjin, China, in 2007, and the Ph.D. degree in control theory from the Academy of Mathematics and Systems Science (AMSS), Chinese Academy of Sciences (CAS), Beijing, China, in 2012.

He is currently an Associate Professor with the Key Lab of System and Control, AMSS, CAS. His research interests include nonlinear uncertain systems control, nonlinear uncertain

systems filter, and distributed filter.



Tao Cai received the M.S. degree and the Ph.D. degree in control science and engineering from the Beijing Institute of Technology, China, in 1999 and 2011, respectively.

Since July 2004, he has been an Associate Professor with the Beijing Institute of Technology. His research interests include nonlinear control, motion control, cooperative control, sensors, and measurement.



Zhun Fan (Senior Member, IEEE) received the B.S. and M.S. degrees in control engineering from the Huazhong University of Science and Technology, Wuhan, China, in 1995 and 2000, respectively, and the Ph.D. degree in electrical and computer engineering from Michigan State University, Lansing, MI, USA, in 2004.

He is a Full Professor with the Department of Electronic and Information Engineering, College of Engineering, Shantou University, Shantou, China. His major research interests include robotic systems, robot vision and cognition, computational intelligence, design automation, and machine learning.



Jiaoyang Zhan received the B.S. degree in automation from Jilin University, Jilin, China, in 2016, and the M.A. degree in control science and engineering from the Beijing Institute of Technology, Beijing, China, in 2019. Her research included intelligent control, nonlinear system control, and evolutionary computing.



Jie Chen (Fellow, IEEE) received the B.S., M.S., and Ph.D. degrees in control theory and control engineering from the Beijing Institute of Technology, in 1986, 1996, and 2001, respectively. From 1989 to 1990, he was a visiting scholar in the California State University, U.S.A. From 1996 to 1997, he was a research fellow with the school of E&E, the University of Birmingham, U.K.

He is currently a Professor of control science and engineering, Beijing Institute of Technology and Tongji University, Shanghai, China. His main research interests include intelligent control and decision of complex systems, multi-agent systems, and optimization methods.

Prof. Chen served as the Editor-in-Chief for the Journal of Systems Science and Complexity and the journal Unmanned Systems from 2019 to present, and an Associate Editor for the IEEE Transactions on Cybernetics from 2016 to present and many other international journals. He is also an Academician of the Chinese Academy of Engineering and an IFAC Fellow.