Research Article

Pareto Design of Decoupled Sliding-Mode Controllers for Nonlinear Systems Based on a Multiobjective Genetic Algorithm

M. J. Mahmoodabadi,[1] A. Bagheri,[1] N. Nariman-Zadeh,[1,2] A. Jamali,[1] and R. Abedzadeh Maafi[3]

[1] *Department of Mechanical Engineering, Faculty of Engineering, University of Guilan,*
  P.O. Box 3756, Rasht, Iran
[2] *Intelligent-Based Experimental Mechanics Center of Excellence, School of Mechanical Engineering,*
  Faculty of Engineering, University of Tehran, Tehran, Iran
[3] *Department of Mechanical Engineering, Takestan Branch, Islamic Azad University, Takestan, Iran*

Correspondence should be addressed to M. J. Mahmoodabadi, mahmoodabadi@guilan.ac.ir

This paper presents Pareto design of decoupled sliding-mode controllers based on a multiobjective genetic algorithm for several fourth-order coupled nonlinear systems. In order to achieve an optimum controller, at first, the decoupled sliding mode controller is applied to stablize the fourth-order coupled nonlinear systems at the equilibrium point. Then, the multiobjective genetic algorithm is applied to search the optimal coefficients of the decoupled sliding-mode control to improve the performance of the control system. Considered objective functions are the angle and distance errors. Finally, the simulation results implemented in the MATLAB software environment are presented for the inverted pendulum, ball and beam, and seesaw systems to assure the effectiveness of this technique.

1. Introduction

There are many control techniques that have been used to investigate the control behavior of the nonlinear systems [1–4]. A variable structure control with sliding mode, which is commonly known as sliding-mode control, is a nonlinear control strategy that is well-known for its guaranteed stability, robustness against parameter variations, fast dynamic response, and simplicity in implementation [1]. Although the sliding mode control method gives a satisfactory performance for the second-order systems, its performance for a fourth-order

coupled system is questionable. For example, in an inverted pendulum system controlled by the sliding-mode control, either the pole or cart can be successfully controlled, but not both. A remedy to this problem is to decouple the states and apply a suitable control law to stabilize the whole system. Recently, a decoupled sliding-mode control has been proposed to cope with this issue. It provides a simple way to decouple a class of fourth-order nonlinear systems in two second-order subsystems such that each subsystem has a separate control objective expressed in terms of a sliding surface [5, 6]. An important consequence of the using decoupled sliding-mode control is that the second subsystem is successfully incorporated into the first one via a two-level decoupling strategy.

It is very important to note that for design of the sliding-mode control and decoupled sliding-mode control, the sliding surface parameters should be determined, properly. This point is very crucial for the performance of the control system. The problem can be solved using evolutionary optimization techniques such as the genetic algorithm [7–10]. In this paper, a new intelligent decoupled sliding-mode control scheme based on an improved multiobjective genetic algorithm is proposed. Using this optimization algorithm, the important parameters of the decoupled sliding mode controller are optimized in a way to decrease the errors of the position and angle, simultaneously. The results obtained from this study illustrate that there are some important optimal design facts among objective functions which have been discovered via the Pareto optimum design approach. Such important design facts could not be found without using the multiobjective Pareto optimization process. In the end, simulations are presented to show the feasibility and efficiency of the proposed Pareto optimum decoupled sliding-mode control for the nonlinear systems.

2. Sliding-Mode Control

Sliding-mode controller is a powerful robust control strategy to treat the model uncertainties and external disturbances [11]. Furthermore, it has been widely applied to robust control of nonlinear systems [12–18]. In this section we recall the general concepts of sliding mode control for a second-order dynamic system. Suppose a nonlinear system is defined by the general state space equation as follows:

$$\dot{x} = f(x, u, t), \tag{2.1}$$

where $x \in R^n$ is the state vector, $u \in R^m$ the input vector, n is the order of the system, and m is the number of inputs. Then, the sliding surface $s(e, t)$ is given by the following:

$$s(e, t) = \left\{ e : H^T e = 0 \right\}, \tag{2.2}$$

where $H \in R^n$ represents the coefficients or slope of the sliding surface. Here,

$$e = x - x_d \tag{2.3}$$

is the negative tracking error vector.

Usually a time-varying sliding surface $s(t)$ is simply defined in the state-space R^n by the scalar equation as the following:

$$s(e,t) = \left(\frac{d}{dt} + \lambda\right)^{n-1} e = 0, \tag{2.4}$$

where λ is a strictly positive constant that can also be explained as the slope of the sliding surface. For instance, if $n = 2$ (for a second-order system) then,

$$s = \dot{e} + \lambda e, \tag{2.5}$$

and hence, s is simply a weighted sum of the position and velocity error from (2.4). The nth-order tracking problem is now being replaced by a first-order stabilization problem in which scalar s is to be kept at zero by a governing reaching condition. By choosing Lyapunov function $V(x) = (1/2)s^2$, the following equation can guarantee that the reaching condition is satisfied,

$$\dot{V}(x) = s\dot{s} < 0. \tag{2.6}$$

The existence and convergence conditions can be rewritten as follows:

$$s\dot{s} \leq -\eta|s|. \tag{2.7}$$

This equation permits a nonswitching region. Here, η is a strictly positive constant, and its value is usually chosen based on some knowledge of disturbances or system dynamics in terms of some known amplitudes.

In this control method, by changing the control law according to certain predefined rules which depend on the position of the error states of the system with respect to sliding surfaces, those states are switched between stable and unstable trajectories until they reach the sliding surface.

It can be shown that the sliding condition of (2.6) is always satisfied by the following:

$$u = u_{eq} - k \cdot \text{sgn}(s), \tag{2.8}$$

where u_{eq} is called equivalent-control input which is obtained by $\dot{s} = 0$. k is a design parameter and $k \geq \eta$.

Function sgn makes the high frequency chattering in control command. Using a proper definition of a thin boundary layer around the sliding surface, the chattering can be eliminated (Figure 1). This is accomplished by defining a boundary layer of thickness Φ, and replacing function sgn with function sat. This function is as the following and shown in Figure 2,

$$\text{sat}\left(\frac{s}{\Phi}\right) = \begin{cases} \text{sgn}\left(\frac{s}{\Phi}\right) & \text{if } \left|\frac{s}{\Phi}\right| \geq 1 \\ \left(\frac{s}{\Phi}\right) & \text{if } \left|\frac{s}{\Phi}\right| < 1 \end{cases}. \tag{2.9}$$
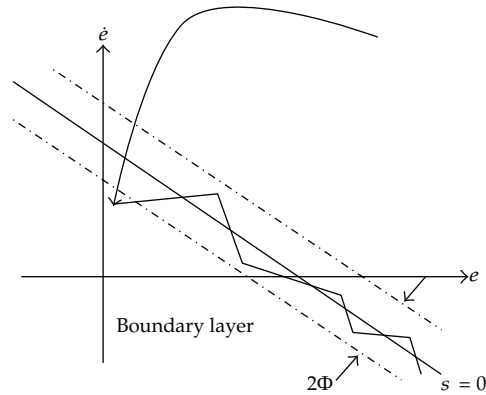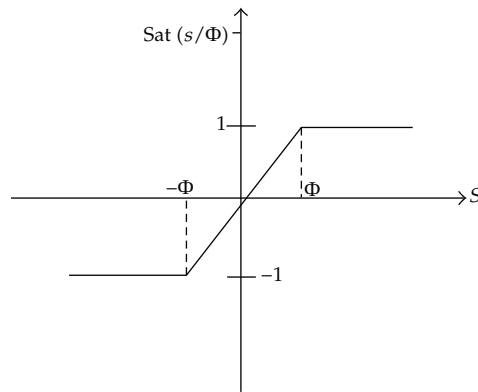
Figure 1: Sliding plant of a smooth controller.

Figure 2: Function Sat (s/Φ) to eliminate the chattering phenomena in the sliding mode controller.

3. Inverted Pendulum System

In this section, the model of an inverted pendulum is recalled. In fact, the work deals with the stabilization control of a complicated, nonlinear, and unstable system. A pole, hinged to a cart moving on a track, is balanced upwards by motioning of the cart via a DC motor. The system observable state vector is $x = [x_1, x_2, x_3, x_4]^T$, including, respectively, the position of the cart, the angle of the pole with respect to the vertical axis, and their derivatives. The force to motion the cart may be expressed as $F = \alpha u$, where u is the input that is the limited motor supply voltage. The system dynamic model is as follows:

$$
\begin{aligned}
\dot{x}_1 &= x_3, \\
\dot{x}_2 &= x_4, \\
\dot{x}_3 &= f_1(x) + b_1(x)u, \\
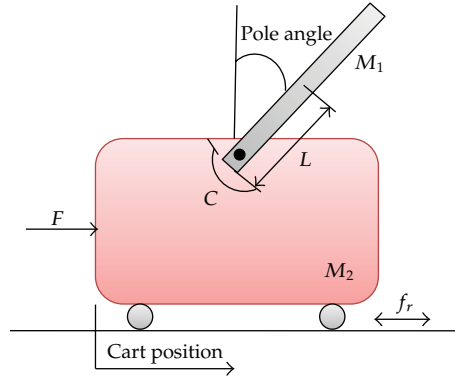\dot{x}_4 &= f_2(x) + b_2(x)u,
\end{aligned}
\tag{3.1}
$$

Figure 3: Inverted pendulum system.

where

$$f_1(x) = \frac{a\left(-f_r - \mu x_4^2 \sin x_2\right) - l\,\cos(x_2)\left(\mu g \sin x_2 - Cx_4\right)}{J + \mu l \sin^2 x_2},$$

$$b_1(x) = \frac{a\alpha}{J + \mu l \sin^2 x_2},$$

$$f_2(x) = \frac{l\,\cos(x_2)\left(-f_r - \mu x_4^2 \sin x_2\right) + \mu g \sin x_2 - Cx_4}{J + \mu l \sin^2 x_2},$$

$$b_2(x) = \frac{l\,\cos x_2\,\alpha}{J + \mu l \sin^2 x_2},$$

(3.2)

That is

$$l = \frac{LM_1}{2(M_2 + M_1)}, \qquad a = l^2 + \frac{J}{M_2 + M_1}, \qquad \mu = (M_2 + M_1)l.$$

(3.3)

Masses of the cart and pole are, respectively, M_2 and M_1, g represents the gravity acceleration, L is the half length of the pole, and J is the overall inertia moment of the cart and pole with respect to the system centre of mass. C is the rotational friction coefficient of the pole, and f_r is the horizontal friction coefficient of the cart (Figure 3). This system is a nonlinear fourth-order system that includes two second-order subsystems in the canonical form with states $[x_1, x_3]^T$ and $[x_2, x_4]^T$.

4. Ball and Beam System

The ball and beam system is one of the most enduringly popular and important laboratory models for teaching control systems engineering. Because it is very simple to understand as a system, and control techniques that can stable it cover many important classical and modern design methods. The system has a very important property, it is open-loop unstable. The system is very simple, a steel ball rolling on the top of a long beam. The beam is mounted on
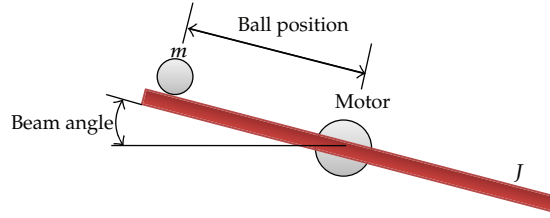
Figure 4: Ball and beam system.

the output shaft of an electrical motor, and so the beam can be tilted about its center axis by applying an electrical control signal to the motor amplifier. The control job is to automatically regulate the position of the ball on the beam by changing the angle of the beam. This is a difficult control task because the ball does not stay in one place on the beam, and moves with acceleration that is approximately proportional to the tilt of the beam. In control terminology, the system is open-loop unstable because the system output (the ball position) increases without any limitation for a fixed input (beam angle). Feedback control must be used to stabilize the system and to keep the ball in a desired position on the beam.

Consider a ball and beam system depicted in Figure 4 and its dynamic is described below:

$$\dot{x}_1 = x_3,$$
$$\dot{x}_2 = x_4,$$
$$\dot{x}_3 = f_1(x) + b_1(x)u,$$
$$\dot{x}_4 = f_2(x) + b_2(x)u,$$

$$(4.1)$$

where

$$f_1(x) = -\frac{5}{7}\left(g\sin(x_2) - x_1 x_4^2\right),$$

$$b_1(x) = 0,$$

$$f_2(x) = \frac{-mx_1\left(2x_3 x_4 - g\cos(x_2)\right)}{mx_1^2 + J},$$

$$b_2(x) = \frac{1}{mx_1^2 + J}.$$

$$(4.2)$$

The mass of the ball is m, g represents the gravity acceleration, and J is the inertia moment of the beam (Figure 4). The system observable state vector is $x = [x_1, x_2, x_3, x_4]^T$, including, respectively, the position of the ball, the angle of the beam with respect to the horizontal axis, and their derivatives. This system is a nonlinear fourth-order system that includes two second-order subsystems in the canonical form with states $[x_1, x_3]^T$ and $[x_2, x_4]^T$.
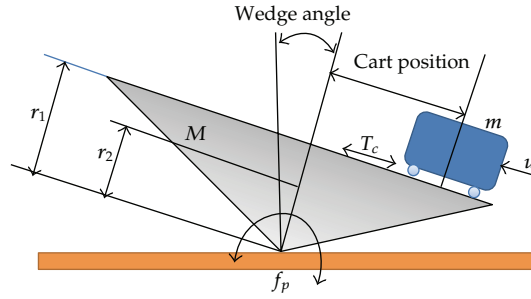
Wedge angle

Cart position

r_1

r_2

M

T_c

m

u

f_p

Figure 5: Seesaw system.

5. Seesaw System

According to the basic physical concepts, in the seesaw mechanism, if the vertical line along the centre of gravity of the inverted wedge is not passing through the fulcrum perpendicularly, then the inverted wedge will result in a torque and rotates until reaching the stable state. If we want to balance the inverted wedge, we have to put an external force to produce an appropriate opposite torque. For this reason, the inverted wedge is equipped with a cart to balance the unstable system. The cart can move to produce the appropriate torque against the internal force (Figure 5).

The observable state vector is $x = [x_1, x_2, x_3, x_4]^T$, including, respectively, the cart position, the wedge angle with respect to the vertical axis, and their derivatives. The system dynamic model is as the following:

$$
\begin{aligned}
\dot{x}_1 &= x_3, \\
\dot{x}_2 &= x_4, \\
\dot{x}_3 &= f_1(x) + b_1(x)u, \\
\dot{x}_4 &= f_2(x) + b_2(x)u,
\end{aligned}
\tag{5.1}
$$

where

$$
\begin{aligned}
f_1(x) &= g\sin(x_2) - \frac{T_c}{m}x_3, \\
b_1(x) &= \frac{1}{m}, \\
f_2(x) &= \frac{Mgr_2\sin(x_2) + mg\sqrt{x_1^2 + r_1^2}\sin(x_2 + \alpha) - f_p x_4}{J}, \\
b_2(x) &= \frac{r_1}{J},
\end{aligned}
\tag{5.2}
$$

that is $\alpha = \tan^{-1}(x_1/r_1)$.

The cart and wedge masses are, respectively, m and M, g represents the gravity acceleration, r_1 is the height of the wedge, r_2 is the height of mass centre, J is the inertia

moment of the wedge, f_p is the rotational friction coefficient of the wedge, and T_c is the friction coefficient of the cart. This system is a nonlinear fourth-order system that includes two second-order subsystems in the canonical form with states $[x_1, x_3]^T$ and $[x_2, x_4]^T$.

6. Decoupled Sliding-Mode Control

Consider the nonlinear fourth-order coupled system expressed as the following.

$$
\begin{aligned}
\dot{x}_1 &= x_3, \\
\dot{x}_2 &= x_4, \\
\dot{x}_3 &= f_1(x) + b_1(x)u, \\
\dot{x}_4 &= f_2(x) + b_2(x)u.
\end{aligned}
\tag{6.1}
$$

This system includes two second-order subsystems in the canonical form with states $[x_1, x_3]^T$ and $[x_2, x_4]^T$, and the sliding-mode control mentioned in the Section 2 can only control one of these subsystems. Hence, the basic idea of the decoupled sliding-mode control is proposed to design a control law such that the single input u simultaneously controls two coupled subsystems to accomplish the desired performance [5, 6, 19]. To achieve this goal, the following sliding surfaces are defined:

$$
s_1(x) = \lambda_1 \cdot (x_2 - x_{2d} - z) + x_4 - x_{4d} = 0
\tag{6.2a}
$$

$$
s_2(x) = \lambda_2 \cdot (x_1 - x_{1d}) + x_3 - x_{3d} = 0.
\tag{6.2b}
$$

Here, z is a proportional value of s_2 and has a proper range with respect to x_2. A comparison of (6.2a) with (2.5) shows the meaning of (6.2a): the control objective in the first subsystem of (6.1) changes from $x_2 = x_{2d}$ and $x_4 = x_{4d}$ to $x_2 = x_{2d} + z$ and $x_4 = x_{4d}$. On the other hand, (6.2b) has the same meaning of (2.5) and its control objectives are $x_1 = x_{1d}$ and $x_3 = x_{3d}$. Now, let the control law for (6.2a) be a sliding mode with a boundary layer, then:

$$
u_1 = \hat{u}_1 - G_{f_1}\operatorname{sat}(s_1(x)b_2(x)G_{s_1}), \quad G_{f_1}, G_{s_1} > 0,
\tag{6.3}
$$

with

$$
\hat{u}_1 = -b_1^{-1}(x)\left(f_2(x) - \ddot{x}_{2d} + \lambda_1 x_4 - \lambda_1 \dot{x}_{2d}\right).
\tag{6.4}
$$

So

$$
z = \operatorname{sat}(s_2 \cdot G_{s_2}) \cdot G_{f_2}, \quad 0 < G_{f_2} < 1,
\tag{6.5}
$$

where G_{s_2} represents the inverse of the width of the boundary layer for s_2, G_{f_2} transfers s_2 to the proper range of x_2. Notice, in (6.5) z is a decaying oscillation signal since $G_{f_2} < 1$. Moreover, in (6.2a), if $s_1 = 0$, then $x_2 = x_{2d} + z$ and $x_4 = x_{4d}$.

Now, the control sequence is as follows: when $s_2 \neq 0$, then $z \neq 0$ in (6.2a) causes (6.3) to generate a control action that reduces s_2; as s_2 decreases, z decreases too. Hence, at the limit $s_2 \rightarrow 0$ with $x_1 \rightarrow x_{1d}$, then $z \rightarrow 0$ with $x_2 \rightarrow x_{2d}$; so, $s_1 \rightarrow 0$, and the control objective would be achieved [19].

7. Genetic Algorithm

Optimization in engineering design has always been of great importance and interest particularly in solving complex real-world design problems. Basically, the optimization process is defined as finding a set of values for a vector of design variables so that it leads to an optimum value of an objective or cost function. In such single-objective optimization problems, there may or may not exist some constraint functions on the design variables, and they are, respectively, referred to as constrained or unconstrained optimization problems. There are many calculus-based methods including gradient approaches to search for mostly local optimum solutions and these are well documented in [20, 21]. However, some basic difficulties in the gradient methods such as their strong dependence on the initial guess can cause them to find a local optimum rather than a global one. This has led to other heuristic optimization methods, particularly genetic algorithms (GAs) being used extensively during the last decade. Such nature-inspired evolutionary algorithms [22, 23] differ from other traditional calculus based techniques. The main difference is that GAs work with a population of candidate solutions, not a single point in search space. This helps significantly to avoid being trapped in local optima [24] as long as the diversity of the population is well preserved.

One of complex real-world problems is the controller design, because it is necessary to assign the control parameters. This parameter tuning is traditionally based on the trial and error procedure; however, this problem can be solved via evolutionary algorithms, for example, genetic algorithms. In the existing literature, several previous works have considered the evolutionary algorithms for control design. For an overview of evolutionary algorithms in the control engineering, [25] is appropriate. In particular, the pole placement procedure to design a discrete-time regulator in [26] and the observer-based feedback control design in [27] are formulated as multiobjective optimization problems and solved via genetic algorithms. Moreover, in [28], two decoupled sliding-mode control configurations are designed for a scale model of an oil platform supply ship while the genetic algorithm is used for optimization.

A simple genetic algorithm includes individual selection from population based on the fitness, crossover, and mutation with some probabilities to generate new individuals. With the genetic operation going on, the individual maximum fitness and the population average fitness are increased, steadily. When applied to a problem, GA uses a genetics-based mechanism to iteratively generate new solutions from currently available solutions. It then replaces some or all of the existing members of the current solution pool with the newly created members. The motivation behind the approach is that the quality of the solution pool should improve with the passage of time [22, 23].

8. Multiobjective Optimization

In multiobjective optimization problems which is also called multi-criteria optimization problems or vector optimization problems, there are several objective or cost functions

(a vector of objectives) to be optimized (minimized or maximized), simultaneously. These objectives often conflict with each other so that as one objective function improves, another deteriorates. Therefore, there is no single optimal solution that is best with respect to all the objective functions. Instead, there is a set of optimal solutions, well-known as Pareto optimal solutions [29–32], which distinguishes significantly the inherent natures between single-objective and multiobjective optimization problems.

In fact, multiobjective optimization has been defined as finding a vector of decision variables satisfying constraints to give acceptable values to all objective functions. Such multiobjective minimization based on Pareto approach can be conducted using some definitions [33].

8.1. Definition of Pareto Dominance

A vector $\vec{U} = [u_1, u_2, \ldots, u_n]$, is dominance to vector $\vec{V} = [v_1, v_2, \ldots, v_n]$ (denoted by $\vec{U} \prec \vec{V}$) if and only if for all $i \in \{1, 2, \ldots, n\}$, $u_i \leq v_i \wedge \exists j \in \{1, 2, \ldots, n\} : u_j < v_j$.

8.2. Definition of Pareto Optimality

A point $X^* \in \Omega$ (Ω is a feasible region in R^n) is said to be Pareto optimal (minimal) if and only if there is not $X \in \Omega$ which is dominance to X^*. Alternatively, it can be readily restated as following. For all $X \in \Omega$, $X \neq X^*$, $\exists i \in \{1, 2, \ldots, m\} : f_i(X^*) < f_i(X)$.

8.3. Definition of Pareto Set

For a given multiobjective optimization problem, a Pareto set P^* is a set in the decision variable space consisting of all the Pareto optimal vectors. $P^* = \{X \in \Omega \mid \nexists X' \in \Omega : F(X') \prec F(X)\}$.

8.4. Definition of Pareto Front

For a given multiobjective optimization problem, the Pareto front PT^* is a set of vectors of objective functions which are obtained using the vectors of decision variables in the Pareto set P^*, that is $PT^* = \{F(X) = (f_1(X), f_2(X), \ldots, f_m(X)) : X \in P^*\}$. In other words, the Pareto front PT^* is a set of the vectors of objective functions mapped from P^*.

In fact, evolutionary algorithms have been widely used for multiobjective optimization because of their natural properties suited for these types of problems. This is mostly because of their parallel or population-based search approach. Therefore, most of the difficulties and deficiencies within the classical methods in solving multiobjective optimization problems are eliminated. For example, there is no need for either several runs to find all individuals of the Pareto front or quantification of the importance of each objective using numerical weights. In this way, the original nondominated sorting procedure given by Goldberg [22] was the catalyst for several different versions of multiobjective optimization algorithms [29, 30]. However, it is very important that the genetic diversity within the population be preserved sufficiently. This main issue in multiobjective optimization problems has been addressed by many related research works [34]. Consequently, the premature convergence of multiobjective optimization evolutionary algorithms is prevented, and the

solutions are directed and distributed along the true Pareto front if such genetic diversity is well provided. The Pareto-based approach of NSGAII [33] has been used recently in a wide area of engineering multiobjective optimization problems because of its simple yet efficient non-dominance ranking procedure in yielding different level of Pareto frontiers. However, the crowding approach in such state-of-the-art multiobjective optimization problems [35] is not efficient as a diversity preserving operator [36]. In this paper, a new diversity preserving algorithm called ε-elimination diversity algorithm [36], as a multiobjective tool, searches the definition space of decision variables and returns the optimum answers in Pareto form. In this ε-elimination diversity approach that is used to replace the crowding distance assignment approach in NSGAII [33], all the clones and/or ε-similar individuals based on Euclidean norm of two vectors are recognized and simply eliminated from the current population. Therefore, based on a predefined value of ε as the elimination threshold ($\varepsilon = 0.01$ has been used in this paper) all the individuals in a front within this limit of a particular individual are eliminated. It should be noted that such ε-similarity must exist both in the space of objectives and in the space of the associated design variables. This will ensure that very different individuals in the space of design variables having ε-similarity in the space of objectives will not be eliminated from the population. Evidently, the clones or ε-similar individuals are replaced from the population with the same number of new randomly generated individuals. Meanwhile, this will additionally help to explore the search space of the given multiobjective optimization problems more efficiently [36].

9. Multiobjective Optimization of Decoupled Sliding Mode Control

As mentioned before this, it is necessary for the practical engineering applications to solve the optimization problems involving multiple design criteria which are also called objective functions. Furthermore, the design criteria may conflict with each other so that improving one of them will deteriorate since another. The inherent conflicting behavior of such objective functions lead to a set of optimal solutions named Pareto solutions. These types of problems can be solved using evolutionary multiobjective optimization techniques. Here, for multiobjective optimization of the decoupled sliding mode controller, vector $[G_{f_1}, G_{s_1}, \lambda_1, G_{f_2}, G_{s_2}, \lambda_2]$ is the vector of selective parameters of the decoupled sliding mode controller. G_{f_1} and G_{s_1} are positive constant. λ_1 and λ_2 are coefficients of sliding surfaces, and G_{s_2} represents the inverse of the width of the boundary layer of s_2. G_{f_2} transfers s_2 to the proper range of x_2. The error of the position and the error of the angle are functions of this vector's components. This means that by selecting various values for the selective parameters, we can make changes in the position and angel errors. In this paper, we are concerned in choosing values for the selective parameters to minimize above two functions. Clearly, this is an optimization problem with two object functions (errors of position and angle) and six decision variables $[G_{f_1}, G_{s_1}, \lambda_1, G_{f_2}, G_{s_2}, \lambda_2]$. The regions of the selective parameters are as follows:

$G_{s_2}, G_{f_1}, G_{s_1}$: positive constant, $G_{s_2}, G_{f_1}, G_{s_1} > 0$,

λ_1, λ_2: coefficients of the sliding surface, $\lambda_1, \lambda_2 > 0$,

G_{f_2}: transfers s_2 to a proper range of x_2, $0 < G_{f_2} < 1$.

The following parameters of the genetic algorithm are considered.
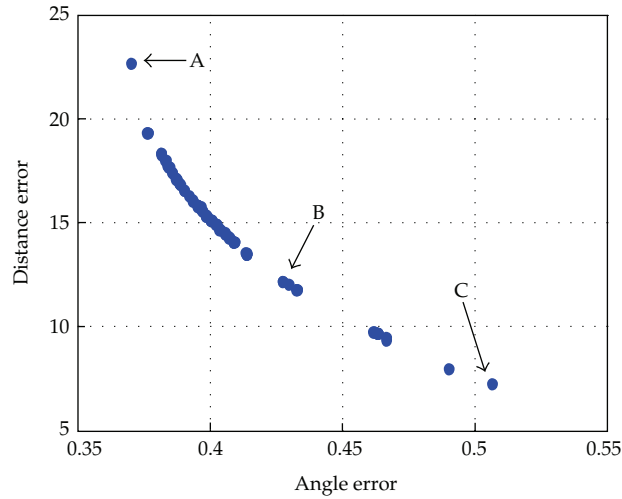
Figure 6: Pareto front of the angle error and distance error for the inverted pendulum.
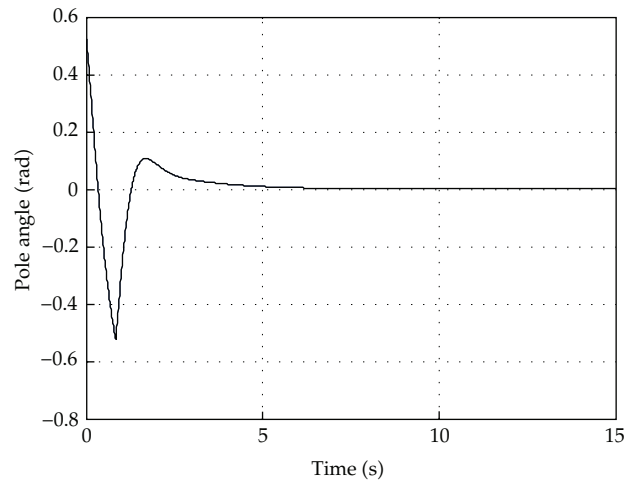

Figure 7: Simulation results for the pole angle.

Population size = 100, chromosome length = 48, generations = 300, crossover probability = 0.8, and mutation probability = 0.02. Also, the stopping criterion for this algorithm is the maximum number of generations.

10. Simulation and Results for the Inverted Pendulum System

The simulation for the inverted pendulum system considered here is carried out by MATLAB software. The initial values are as the following:

$$x_1(0) = 0, \quad x_2(0) = \frac{\pi}{6} \text{ rad}, \quad x_3(0) = 0, \quad x_4(0) = 0. \tag{10.1}$$

The system parameters and constants used in the simulation are given in Table 1.
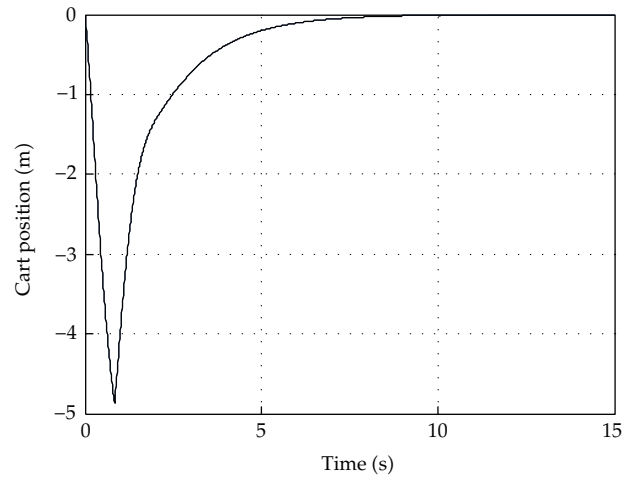
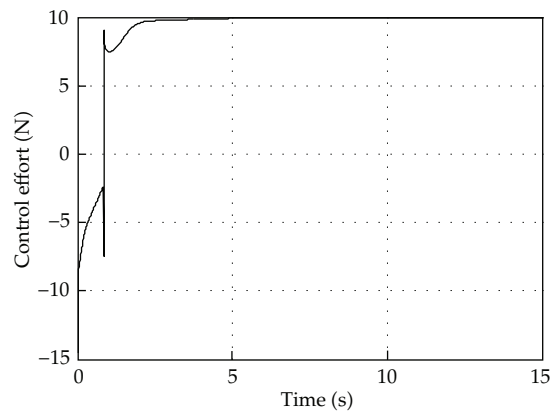**Figure 8:** Simulation results for the cart position.

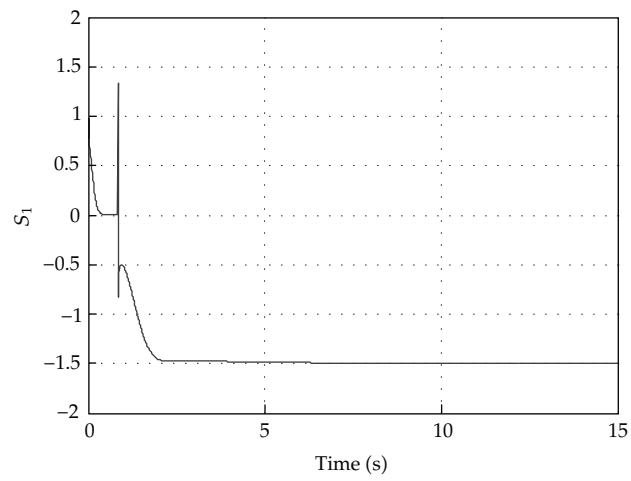**Figure 9:** Simulation results for the control action.

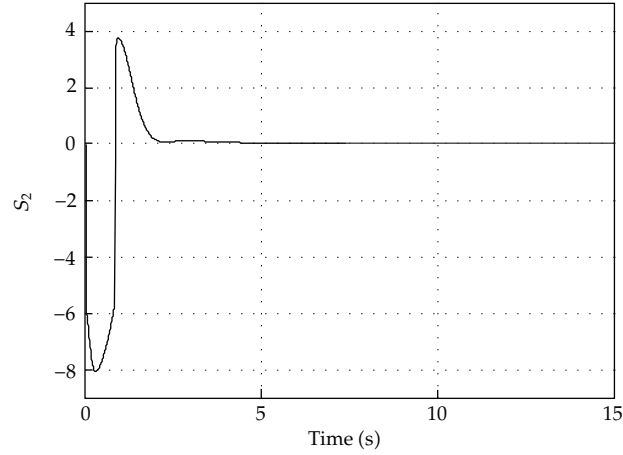
Figure 10: Sliding surface $s_1(x)$.
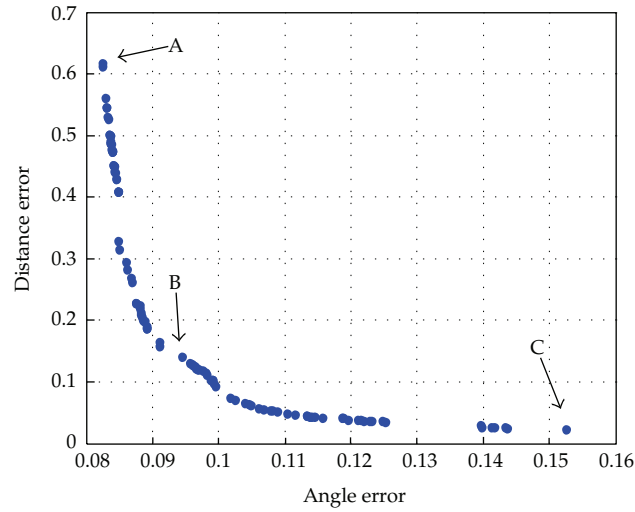
Figure 11: Sliding surface $s_2(x)$.

Figure 12: Pareto front of the angle error and distance error for the ball and beam system.

Table 1: Inverted pendulum parameters.

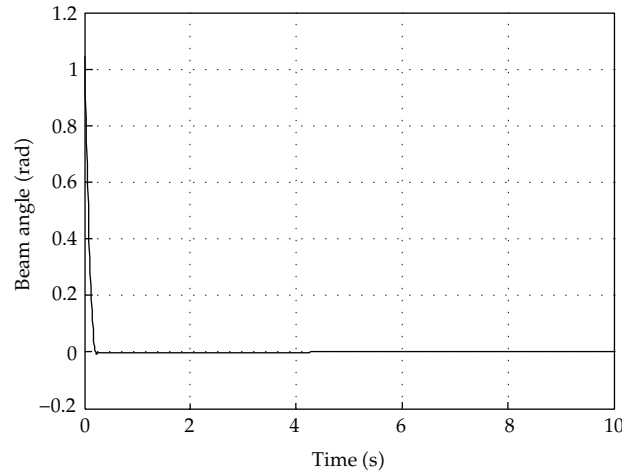|		
The mass of the pole	M_1	0.5
The mass of the cart	M_2	2
The half length of the pole	L	0.5
The inertia moment of the cart and pole	J	0.4
The friction constant of the pole	C	0.1
The friction constant of the cart	f_r	0.25
The gravity acceleration	g	9.81
The force coefficient	α	3

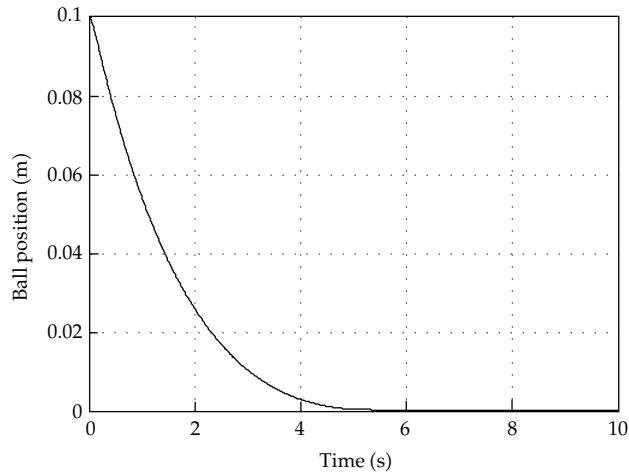**Figure 13:** Simulation results for the beam angle.

Figure 14: Simulation results for the ball position.

When we apply the multiobjective genetic algorithm, we achieve a Pareto front of the angle error and distance error as demonstrated in Figure 6.

Figure 6 is the chart resulted from multiobjective optimization which all the presented points are nondominated to each other. Each point in this chart is a representative of a vector of selective parameters which if we choose it for the decoupled sliding-mode controller, the analysis tends to objective functions corresponding to that point of chart. The design variables and objective functions of the optimum design points A, B, and C are presented in Table 2.

Achieving several solutions, all of which are considered optimum is a unique property of multiobjective optimization. Designer in facing to Pareto charts, among several different optimum points can choose a suitable multisided design point, easily. According to the Pareto chart, we applied point C for simulation, as shown in Figures 7, 8, 9, 10, and 11.

The simulation results (Figures 7, 8, 9, 10, and 11) show that the pole and the cart can be stabilized to the equilibrium point.
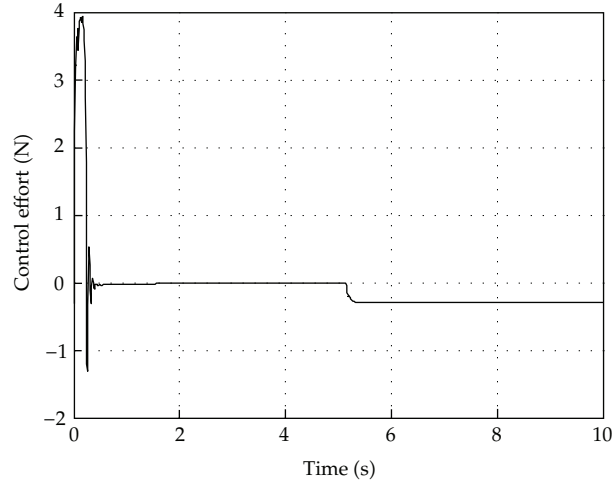
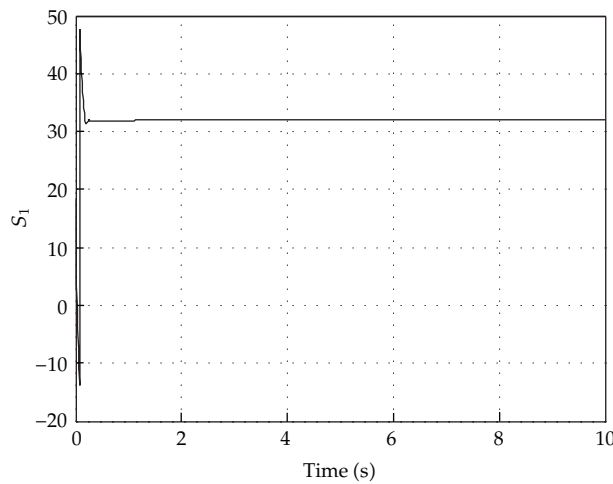**Figure 15:** Simulation results for the control action.

Figure 16: Sliding surface $s_1(x)$.

The numerical results show that the control action is bounded between -15 and 10 (N), and sliding surface $s_2(x)$ reaches to zero during the simulation.

11. Simulation and Results for the Ball and Beam System

The initial values of the ball and beam system are considered in the following form:

$$x_1(0) = 0.1\,\text{m}, \quad x_2(0) = \frac{\pi}{3}\,\text{rad}, \quad x_3(0) = 0, \quad x_4(0) = 0. \tag{11.1}$$

The system parameters and constants used in the simulation are given in Table 3.
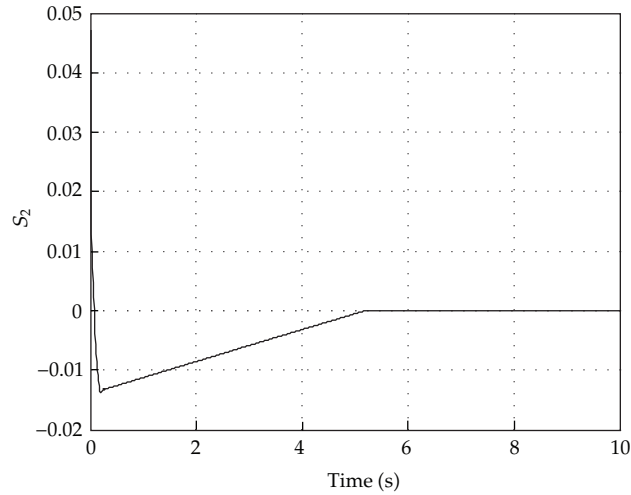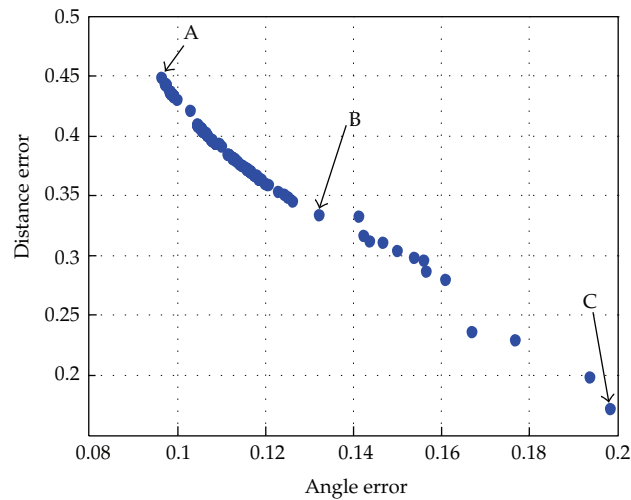
Figure 17: Sliding surface $s_2(x)$.

Figure 18: Pareto front of the angle error and distance error for the seesaw system.

When the multiobjective genetic algorithm is applied, a Pareto front of the angle error and distance error would be achieved (Figure 12).

Figure 12 shows the Pareto front obtained from the modified NSGAII algorithm in an arbitrary run for the ball and beam system. In this figure, points A and C stand for the best distance error and angle error, respectively. Furthermore, point B could be a trade-off optimum choice when considering minimum values of both angle error and distance error. Table 4 illustrates the design variables and objective functions corresponding to the optimum design points A, B, and C.

The time responses of the ball and beam system related to point B are shown in Figures 13, 14, 15, 16, and 17. These figures demonstrate that the ball and beam system can be stabilized to the equilibrium point.
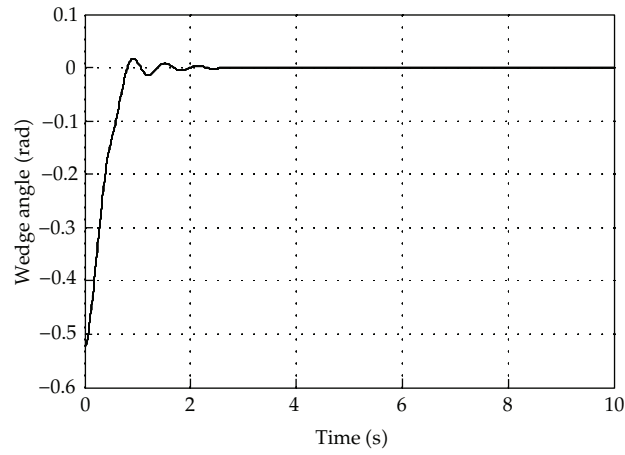
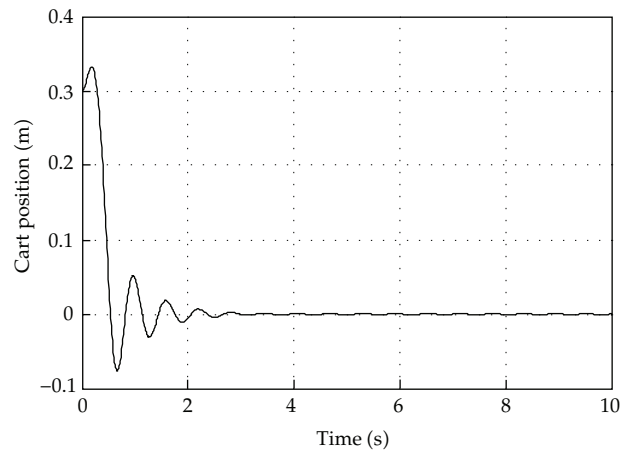**Figure 19:** Simulation results for the wedge angle.
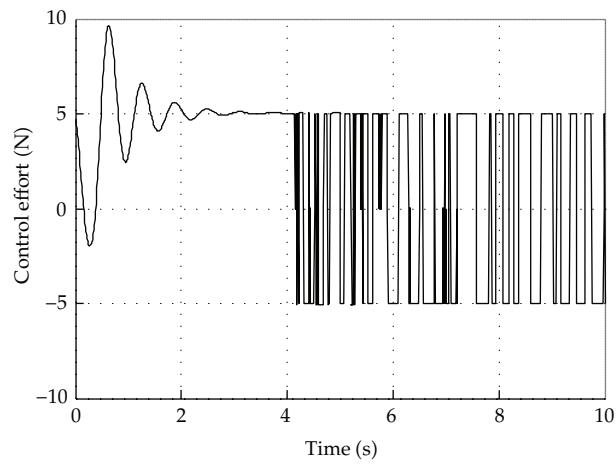

Figure 20: Simulation results for the cart position.

Figure 21: Simulation results for the control action.

Table 2: Comparison among points A, B, and C for Figure 6.

Point	G_{f_1}	G_{s_1}	λ_1	G_{f_2}	G_{s_2}	λ_2	Angle error	Distance error
A	9.9294	9.8941	1.4941	0.9608	0.1851	0.1459	0.3699	22.6927
B	9.9294	9.8941	1.4941	0.9608	0.1851	0.2987	0.4271	12.2111
C	9.8941	9.8941	1.4941	1.0000	0.1772	0.4946	0.5065	7.2827

Table 3: Ball and beam system parameters.

The mass of the ball	m	0.05
The inertia moment of the beam	J	0.0833
The gravity acceleration	g	9.81

Table 4: Comparison among points A, B, and C for Figure 12.

Point	G_{f_1}	G_{s_1}	λ_1	G_{f_2}	G_{s_2}	λ_2	Angle error	Distance error
A	29.8443	1.5451	48.4345	0.6823	46.8690	0.1074	0.0824	0.6174
B	0.2957	0.4158	48.2388	0.6647	41.5855	0.4725	0.0944	0.1402
C	9.8843	1.0040	41.5855	0.6612	2.0569	1.5008	0.1525	0.0225

Table 5: Seesaw system parameters.

The mass of the cart	m	0.46
The mass of the wedge	M	1.52
The height of the wedge	r_1	0.148
The height of center of mass	r_2	0.123
The inertia moment of the wedge	J	0.044
The friction coefficient of the wedge	f_p	0.3
The friction coefficient of the cart	T_c	0.7
The gravity acceleration	g	9.8

Table 6: Comparison among points A, B, and C for Figure 18.

Point	G_{f_1}	G_{s_1}	λ_1	G_{f_2}	G_{s_2}	λ_2	Angle error	Distance error
A	9.9612	7.6706	9.7671	0.0088	0.0060	0.0245	0.0963	0.4489
B	9.9612	0.2165	5.1082	0.0088	0.0557	0.0010	0.1319	0.3345
C	5.0306	0.29416	8.75766	0.60046	0.09306	0.9491	0.1984	0.1729

Furthermore, the simulation shows that the control action is bounded between -1.2 and 4 (N), and sliding surface $s_2(x)$ reaches to zero during simulation.

12. Simulation and Results for the Seesaw System

In this section, the simulation results for seesaw system are investigated. The initial values of this system are described by the following equations:

$$x_1(0) = 0.3\,\text{m}, \quad x_2(0) = -\frac{\pi}{6}\,\text{rad}, \quad x_3(0) = 0, \quad x_4(0) = 0. \tag{12.1}$$
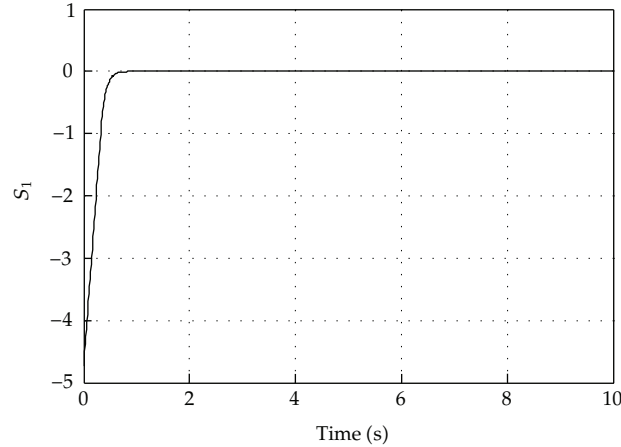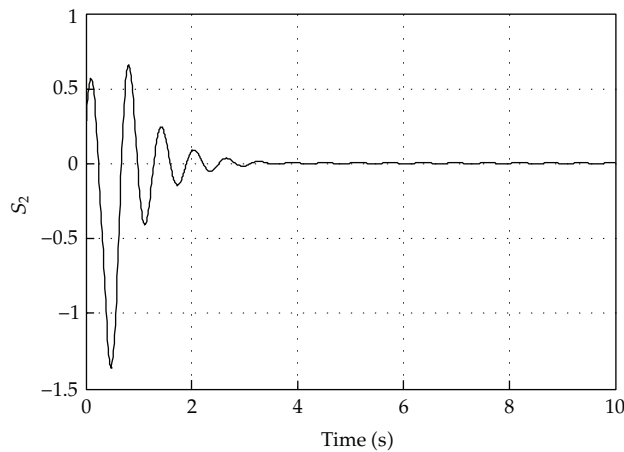
Figure 22: Sliding surface $s_1(x)$.

Figure 23: Sliding surface $s_2(x)$.

The system parameters used in the simulation are given in Table 5.

Figure 18 demonstrates a Pareto front of two objective functions (angle error and distance error) which is achieved of the multiobjective genetic algorithm (e.g. modified NSGAII).

It is clear that all points in Figure 18 are nondominated to each other, and each point in this chart is a representative of a vector of selective parameters for the decoupled sliding mode controller. Moreover, choosing a better value for any objective function in the Pareto front would cause a worse value for another objective function. Here, point B has been chosen from Figure 18 to design an optimum decupled sliding mode controller (Figures 19, 20, 21, 22, and 23). Design variables and objective functions related to the optimum design points A, B, and C are detailed in Table 6.

The simulations (Figures 19, 20, 21, 22, and 23) shows that the seesaw system is stabilized to the equilibrium point after 3 seconds, and the control effort is bounded between -5 and 10 (N).
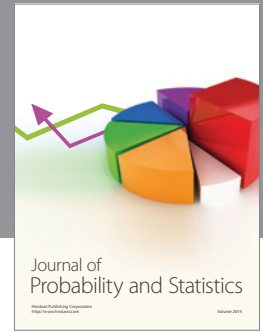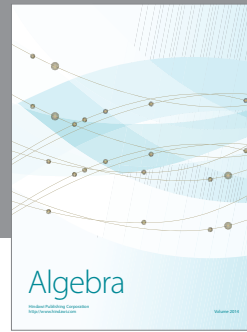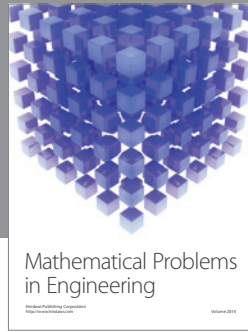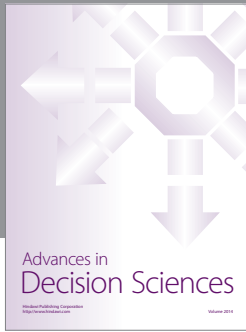
13. Conclusion

This paper proposes the decoupled sliding-mode technique for stabilising the coupled nonlinear systems while the multiobjective genetic algorithm is employed in order to optimize two objective functions. This method is a universal design method and suitable to various kinds of control objects. Usage this method includes two steps. The first step is to design the decoupled sliding-mode controller for the nonlinear system. The second step is to apply the multiobjective optimization tool to search the definition space of decision variables and to return the optimum answers in the Pareto form. The simulation results on three different and typical control systems show good control and robust performance of the proposed strategy.

References

[1] J. J. E. Slotine and W. Li, *Applied Nonlinear Control*, Prentice-Hall, Englewood Cliffs, NJ, USA, 1991.

[2] Z. Gao and S. X. Ding, "Actuator fault robust estimation and fault-tolerant control for a class of nonlinear descriptor systems," *Automatica*, vol. 43, no. 5, pp. 912–920, 2007.

[3] Z. Gao, X. Shi, and S. X. Ding, "Fuzzy state/disturbance observer design for T-S fuzzy systems with application to sensor fault estimation," *IEEE Transactions on Systems, Man, and Cybernetics B*, vol. 38, no. 3, pp. 875–880, 2008.

[4] M. J. Mahmoodabadi, A. Bagheri, S. Arabani Mostaghim, and M. Bisheban, "Simulation of stability using Java application for Pareto design of controllers based on a new multi-objective particle swarm optimization," *Mathematical and Computer Modelling*, vol. 54, no. 5-6, pp. 1584–1607, 2011.

[5] J. C. Lo and Y. H. Kuo, "Decoupled fuzzy sliding-mode control," *IEEE Transactions on Fuzzy Systems*, vol. 6, no. 3, pp. 426–435, 1998.

[6] A. Bagheri and J. J. Moghaddam, "Decoupled adaptive neuro-fuzzy (DANF) sliding mode control system for a Lorenz chaotic problem," *Expert Systems with Applications*, vol. 36, no. 3, pp. 6062–6068, 2009.

[7] N. H. Moin, A. S. I. Zinober, and P. J. Harley, "Sliding mode control design using genetic algorithms," in *Proceedings of the 1st IEE/IEEE International Conference on Genetic Algorithms in Engineering Systems: Innovations and Applications (GALESIA '95)*, vol. 414, pp. 238–244, September 1995.

[8] C. C. Wong and S. Y. Chang, "Parameter selection in the sliding mode control design using genetic algorithms," *Tamkang Journal of Science and Engineering*, vol. 1, no. 2, pp. 115–122, 1998.

[9] P. C. Chen, C. W. Chen, and W. L. Chiang, "GA-based suzzy sliding mode controller for nonlinear systems," *Mathematical Problems in Engineering*, vol. 2008, Article ID 325859, 16 pages, 2008.

[10] J. Javadi-Moghaddam and A. Bagheri, "An adaptive neuro-fuzzy sliding mode based genetic algorithm control system for under water remotely operated vehicle," *Expert Systems with Applications*, vol. 37, no. 1, pp. 647–660, 2010.

[11] H. K. Khalil, *Nonlinear Systems*, MacMillan, New York, NY, USA, 1992.

[12] N. Yagiz and Y. Hacioglu, "Robust control of a spatial robot using fuzzy sliding modes," *Mathematical and Computer Modelling*, vol. 49, no. 1-2, pp. 114–127, 2009.

[13] W. S. Lin and C. S. Chen, "Robust adaptive sliding mode control using fuzzy modelling for a class of uncertain MIMO nonlinear systems," *IEE Proceedings: Control Theory and Applications*, vol. 149, no. 3, pp. 193–202, 2002.

[14] J. Jing and Q. H. Wuan, "Intelligent sliding mode control algorithm for position tracking servo system," *International Journal of Information Technology*, vol. 12, no. 7, pp. 57–62, 2006.

[15] V. I. Utkin and H. C. Chang, "Sliding mode control on electro-mechanical systems," *Mathematical Problems in Engineering*, vol. 8, no. 4-5, pp. 451–473, 2002.

[16] N. F. Al-Muthairi and M. Zribi, "Sliding mode control of a magnetic levitation system," *Mathematical Problems in Engineering*, vol. 2004, no. 2, pp. 93–107, 2004.

[17] Z. L. Wan, Y. Y. Hou, T. L. Liao, and J. J. Yan, "Partial finite-time synchronization of switched stochastic Chua's circuits via sliding-mode control," *Mathematical Problems in Engineering*, vol. 2011, Article ID 162490, 13 pages, 2011.

[18] C. Pukdeboon, "Optimal sliding mode controllers for attitude stabilization of flexible spacecraft," *Mathematical Problems in Engineering*, vol. 2011, Article ID 863092, 20 pages, 2011.

[19] M. Dotoli, P. Lino, and B. Turchiano, "A decoupled fuzzy sliding mode approach to swing-up and stabilize an inverted pendulum, The CSD03," in *Proceedings of the 2nd IFAC Conference on Control Systems Design*, pp. 113–120, Bratislava, Slovak Republic, 2003.

[20] J. S. Arora, *Introduction to Optimum Design*, McGraw-Hill, New York, NY, USA, 1989.

[21] S. S. Rao, *Engineering Optimization: Theory and Practice*, Wiley, NewYork, NY, USA, 1996.

[22] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, Mass, USA, 1989.

[23] T. Back, D. B. Fogel, and Z. Michalewicz, *Handbook of Evolutionary Computation*, Institute of Physics Publishing, New York, NY, USA, Oxford University Press, Oxford, UK, 1997.

[24] G. Renner and A. Ekárt, "Genetic algorithms in computer aided design," *Computer Aided Design*, vol. 35, no. 8, pp. 709–726, 2003.

[25] P. J. Fleming and R. C. Purshouse, "Evolutionary algorithms in control systems engineering: a survey," *Control Engineering Practice*, vol. 10, no. 11, pp. 1223–1241, 2002.

[26] C. M. Fonseca and P. J. Fleming, "Multiobjective optimal controller design with genetic algorithms," in *Proceedings of the International Conference on Control*, vol. 1, pp. 745–749, March 1994.

[27] G. Sánchez, M. Villasana, and M. Strefezza, "Multi-objective pole placement with evolutionary algorithms," *Lecture Notes in Computer Science*, vol. 4403, pp. 417–427, 2007.

[28] E. Alfaro-Cid, E. W. McGookin, D. J. Murray-Smith, and T. I. Fossen, "Genetic algorithms optimisation of decoupled Sliding Mode controllers: simulated and real results," *Control Engineering Practice*, vol. 13, no. 6, pp. 739–748, 2005.

[29] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms," *Evolutionary Computation*, vol. 2, no. 3, pp. 221–248, 1994.

[30] C. M. Fonseca and P. J. Fleming, "Genetic algorithms for multi-objective optimization: formulation, discussion and generalization," in *Proceedings of the 5th International Conference On genetic Algorithms*, S. Forrest, Ed., pp. 416–423, Morgan Kaufmann, San Mateo, Calif, USA, 1993.

[31] C. A. Coello and A. D. Christiansen, "Multiobjective optimization of trusses using genetic algorithms," *Computers and Structures*, vol. 75, no. 6, pp. 647–660, 2000.

[32] C. A. Coello Coello, D. A. Van Veldhuizen, and G. B. Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 2002.

[33] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[34] A. Toffolo and E. Benini, "Genetic diversity as an objective in multi-objective evolutionary algorithms," *Evolutionary Computation*, vol. 11, no. 2, pp. 151–167, 2003.

[35] C. A. Coello Coello and R. L. Becerra, "Evolutionary multiobjective optimization using a cultural algorithm," in *Proceedings of the IEEE Swarm Intelligence Symposium*, pp. 6–13, IEEE Service Center, Piscataway, NJ, USA, 2003.

[36] K. Atashkari, N. Nariman-Zadeh, A. Pilechi, A. Jamali, and X. Yao, "Thermodynamic Pareto optimization of turbojet engines using multi-objective genetic algorithms," *International Journal of Thermal Sciences*, vol. 44, no. 11, pp. 1061–1071, 2005.

Advances in
Operations Research

Advances in
Decision Sciences

Mathematical Problems
in Engineering

Algebra

Journal of
Probability and Statistics

The Scientific
World Journal

International Journal of
Differential Equations

International Journal of
Combinatorics

Advances in
Mathematical Physics

Submit your manuscripts at
http://www.hindawi.com

Hindawi

Journal of
Complex Analysis

Journal of
Mathematics

International Journal of
Stochastic Analysis

Abstract and
Applied Analysis

Discrete Dynamics in
Nature and Society

International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Applied Mathematics

Journal of
Function Spaces

Journal of
Optimization