Balancing Control of Bicyrobo by Particle Swarm Optimization-Based Structure-Specified Mixed H2/H∞ Control

Bui Trung Thanh and Manukid Parnichkun
School of Engineering and Technology,
Asian Institute of Technology,
P.O. Box 4, Klong Luang,
Pathumthani 12120, Thailand
st104181@ait.ac.th, manukid@ait.ac.th

Abstract: *In this paper, a structure-specified mixed H2/H∞ controller design using particle swarm optimization (PSO) is proposed for control balancing of Bicyrobo, which is an unstable system associated with many sources of uncertainties due to un-model dynamics, parameter variations, and external disturbances. The structure-specified mixed H2/H∞ control is a robust and optimal control technique. However, the design process normally comes up with a complex and non-convex optimization problem which is difficult to solve by the conventional optimization methods. PSO is a recently useful meta-heuristic search method used to solve multi-objective and non-convex optimization problems. In the method, PSO is used to search for parameters of a structure-specified controller which satisfies mixed H2/H∞ performance index. The simulation and experimental results show the robustness of the proposed controller in compared with the conventional proportional plus derivative (PD) controller, and the efficiency of the proposed algorithm in compared with the genetic algorithm (GA).*

Keywords: *Bicyrobo; gyroscopic stabilizer; mixed H2/H∞ control; particle swarm optimization; structure-specified controller.*

1. Introduction

Development of autonomous bicycle robots has attracted many researchers in the recent years. An exciting example of bicycle robot is Murata Boy robot developed in Japan (Murata Boy, 2005). There are many methods used to control balancing of bicycle robots such as flywheel balancing (Beznos, A.V., et al., 1998 ; Gallaspy, J.M., 1999 ; Suprapto, S., 2006), mass balancing (Lee, S. & Ham, W., 2002), and steering balancing (Tanaka, Y. & Murakami, T., 2004). Among these methods, flywheel balancing method which uses a spinning wheel as a gyroscopic stabilizer is a good choice because the response time is short and the system can be stable even at stationary position. The balancing principle using flywheel can also be applied to many other systems, for example, balancing of a biped robot (Wong Terence, C.F. & Hung, Y.S., 1996). Various balancing control algorithms have been proposed, such as nonlinear control (Beznos, A.V., et al., 1998), compensator design using root locus approach (Gallaspy, J.M., 1999), and PD control (Suprapto, S., 2006). However, these control algorithms are not robust, the systems cannot carry loads with varied weights and cannot work in disturbance environments. Therefore, a robust control algorithm is necessary in real applications of bicycle robots.

Mixed H_2/H_∞ control is an advanced technique for designing robust and optimal controllers for systems associated with sources of uncertainties. It was firstly proposed by Bernstein (Bernstein, D.S. & Haddad, W.M., 1989), and has been further developed by many researchers (Khargonekar, P.P. & Rotea, M.A., 1991; Rotea, M.A. & Khargonekar, P.P., 1991; Scherer, C.W., 1995; Pereira, G.J. & Araujo, H.X., 2004; Wu, B.L., et al., 2006). The mixed H_2/H_∞ controller design aims to design controllers to attain both robust stability and good performance, for instances, small tracking error, less control energy, etc. Although the mixed H_2/H_∞ control is an advanced method, it is not widely used like PID and lead-lag controllers due to its complex design procedures and the obtained high order controllers. The controllers are as high order as augmented plants by solving Riccati-like equations method (Bernstein, D.S. & Haddad, W.M., 1989; Khargonekar, P.P. & Rotea, M.A., 1991; Rotea, M.A. & Khargonekar, P.P., 1991), using linear matrix inequality (LMI) method (Pereira, G.J. & Araujo, H.X., 2004; Wu, B.L., et al., 2006), or using Youla parameterization method (Scherer, C.W., 1995). Structure-specified mixed H_2/H_∞ control, which has less order, has recently received a great deal of attentions (Chen, B.S., et al., 1995; Krohling R.A., 1998; Chang Y.F., 2005; Ho S.J., et al., 2004; Ho S.J., et al., 2005).

It is well-known that the structure-specified mixed H_2/H_∞ controller design normally generates a complex and non-convex optimization problem which is difficult to solve by the conventional optimization methods. Various stochastic searching algorithms have been proposed to solve this problem. Chen (Chen B.S., et al., 1995) used GA to search for parameters of a mixed H_2/H_∞ optimal PID controller. Krohling (Krohling R.A., 1998) used two real-coded GAs to search for parameters of a fixed-structure controller that minimized the integral of squared error with robust stability of constraint of plant uncertainty. Chang (Chang Y.F., 2005) used GA to synthesize a robust PD controller to control gap on die-sinking electric discharge machines (EDMs). Ho used an intelligent genetic algorithm (IGA) (Ho S.J., et al., 2004) and an orthogonal simulated annealing algorithm (OSA) (Ho S.J., et al., 2005) to synthesize the structure-specified controllers for multi-input multi-output (MIMO) systems that minimized both the integral of squared error (ISE) and the H_∞-norm of some system transfer functions simultaneously.

Although GA is a useful tool for solving optimization problems and has been applied successfully in many control systems (Fleming, P.J. & Purshouse, R.C., 2002), it still has limitations due to its stochastic searching characteristic and complex computation that make it slow convergence to global optimum. PSO is a recent meta-heuristic search firstly proposed by Kennedy (Kennedy, J. & Eberhart, R., 1995). It is a powerful method for solving complex and ill-defined optimization problems because of its oriented searching and simple computation (Song, M.P. & Gu, G.C., 2004; Panda, S. & Padhy, N.P., 2008). PSO has been recently used by many researchers to design controllers for various applications. Mukherjee (Mukherjee, V. & Ghoshal, S.P., 2007) used PSO to design a fuzzy PID controller for an automatic voltage regulator system. Kao (Kao C.C., et al., 2006) used PSO to design a self-tuning PID controller for a slider-crank mechanism. Chang (Chang W.D., 2007) used PSO to design a PID controller for chaotic synchronization.

In this paper, we propose a method to design the structure-specified mixed H_2/H_∞ controllers by using PSO algorithm. In the method, model uncertainty of the system is represented by multiplicative uncertainty, and the system is assumed to be affected by external disturbances. A structure-specified controller is then defined. Finally, PSO is used to search for parameters of an admissible structure-specified controller that minimizes the integral of squared error (H_2 norm) subjected to robust stability constraints (H_∞ norm) against model uncertainty and external disturbances. The proposed algorithm is practically applied to control balancing of Bicyrobo, a bicycle robot with gyroscopic stabilizer. By neglecting forces generated from moving forward and steering, a simplified dynamics model of Bicyrobo is derived using Lagrange method. The un-modeled dynamics, parameter variations, and external disturbances make the system complicate, and require a robust controller. The simulation and experimental results show the robustness of the proposed controller in compared with the conventional PD controller and the efficiency of the proposed algorithm in compared with the GA-based algorithm.

The remainder of this paper is organized as follows. Section 2 describes the hardware configuration and dynamics model of Bicyrobo. Section 3 explains a systematic procedure of the proposed controllers design algorithm. Simulation and experimental results on Bicyrobo are presented in Section 4. Section 5 finally concludes the paper.

2. Bicyrobo

2.1 Hardware configuration of Bicyrobo

A bicycle robot, Bicyrobo, has been developed at Mechatronics Laboratory, Asian Institute of Technology (AIT), as a platform to test performance of the proposed control algorithm. The system is modified from a regular size bicycle. Fig. 1 shows a photograph of Bicyrobo. The robot is designed so that it can carry loads, go forward or backward, and turn left or right without falling down.

Bicyrobo is equipped with the following components: a flywheel with the weight of 8.1kg and diameter of 380mm for creating precession torque; a 48V-200W-3600rpm DC servo motor for rotating the flywheel around its spinning axis; a 48V-200W-3600rpm DC servo motor with 5:1 chain transmission system for controlling the flywheel control axis; a 12V-15W-10rpm gear box DC motor for steering Bicyrobo; a 12V-35 W-100rpm gear box DC motor for driving Bicyrobo to move forward or backward; a PCM-3350 embedded PC running at 300MHz as a central controller with extended A/D and D/A boards; a VG400CC vertical-gyro sensor for measuring the lean angle of Bicyrobo; an EB6-CWZ encoder for measuring angular position of the flywheel control axis; and signal conditioning circuits. The dimension of Bicyrobo is about 1.65m in length, 0.43m in width, and 1.14m in height.

The principle for balancing using flywheel is explained as follows: When the flywheel rotates with a constant speed around Y_1 axis (Fig. 3), if we control angular position of the flywheel around X_1 axis, angular momentum on Z_1 axis generates a torque. This torque is called precession torque generated by gyroscopic effect, and it is used to keep Bicyrobo balanced.

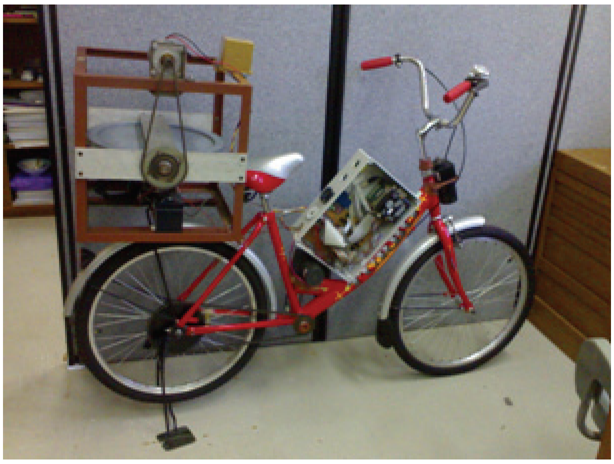


Fig. 1. Photograph of Bicyrobo

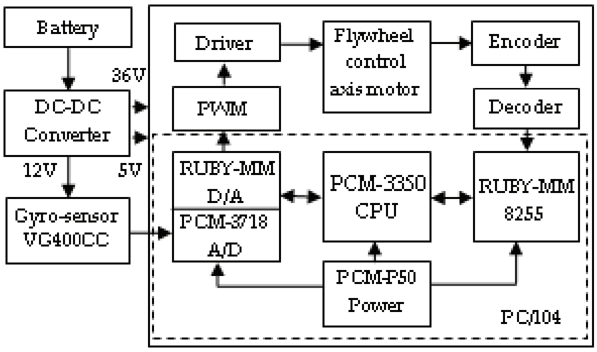


Fig. 2. Hardware configuration of balancing control system

The hardware configuration of the balancing control system is shown in Fig. 2. Program for implementing control algorithms is written in C and run on PC/104 under DOS to ensure real time control.

2.2 Dynamics model of Bicyrobo

Several assumptions are made to simplify the system as follows:

liên kết cứng

- The system is simplified to two rigid body links. The first link is the bicycle frame which has one degree of freedom (DOF) on leaning angle only, rotation around Z axis. The second link is the flywheel which has three DOFs including rotations around X₁, Y₁, and Z axes (Fig. 3).
- The flywheel is assumed to have a constant speed ω. Center of gravity of the flywheel is fixed relative to the bicycle frame.

trục tọa độ

Diagram of the simplified reference coordinates is shown in Fig. 3, where B and F denote bicycle and flywheel centers of gravity, respectively. The lean angle of the bicycle around Z axis is defined as θ, and the angular position of the control axis of the flywheel around X_1 axis is defined as φ. The angular velocity of the bicycle around Z axis is defined as $\dot{\theta}$, and the angular velocity of the flywheel around its control axis (X_1 axis) is defined as $\dot{\varphi}$.

Since the flywheel center of gravity does not move relative to the bicycle center of gravity, absolute velocities of B and F are given by

$$|v_b| = \dot{\theta}h_b \tag{1}$$

$$|v_f| = \dot{\theta}h_f \tag{2}$$

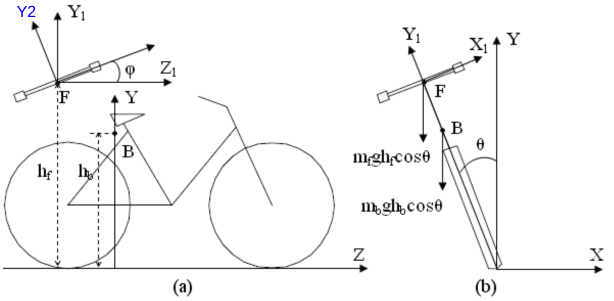


Fig. 3. Diagram of reference coordinates of Bicyrobo: (a) side view; (b) front view

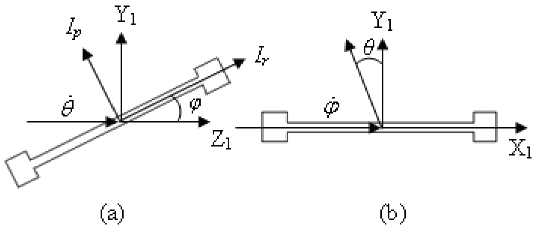


Fig. 4. Moments of inertia assignment for the flywheel : (a) side view ; (b) front view

where h_b, h_f are the height of bicycle center of gravity, and the height of flywheel center of gravity, respectively. To derive the dynamics model of the system, Lagrange equation is used

$$\frac{d}{dt}\left\{\frac{\partial T}{\partial \dot{q}_i}\right\} - \frac{\partial T}{\partial q_i} + \frac{\partial V}{\partial q_i} = Q_i \tag{3}$$

where T is system total kinetic energy, V is system total potential energy, Q_i is external forces, and q_i is generalized coordinate. V and T are determined, and represented by the following equations.

$$V = m_b g h_b cos\theta + m_f g h_f cos\theta \tag{4}$$

$$\begin{aligned} T &= \frac{1}{2}m_b(\dot{\theta}^2 h_b^2) + \frac{1}{2}m_f(\dot{\theta}^2 h_f^2) + \frac{1}{2}I_b\dot{\theta}^2 \\ &+ \frac{1}{2}\left[I_r\dot{\varphi}^2 + I_p(\dot{\theta}sin\varphi)^2 + I_r(\dot{\theta}\cos\varphi)^2\right] \end{aligned} \tag{5}$$

where I_p is flywheel polar moment of inertia and I_r is flywheel radial moment of inertia. m_g and m_f are bicycle and flywheel masses, respectively. I_b is bicycle moment of inertia.

For $q_i = \theta$, using equations (3)-(5), the following equation is derived.

Theo chiều của X

$$\begin{aligned} &\ddot{\theta}\left[m_b h_b^2 + m_f h_f^2 + I_b + I_p sin^2\varphi + I_r cos^2\varphi\right] \\ &+2sin\varphi cos\varphi(I_p - I_r)\dot{\theta}\dot{\varphi} \\ &-g(m_b h_b + m_f h_f)sin\theta = I_p\omega\dot{\varphi}cos\varphi \end{aligned} \tag{6}$$

Vế phải là thành phần ngoại lực gây ra bậc tự do

For $q_i = \varphi$, the following equation is derived.

theo chiều của Z

$$\ddot{\varphi}I_r - \dot{\theta}^2(I_p - I_r)sin\varphi cos\varphi = T_m - I_p\omega\dot{\theta}cos\varphi - B_m\dot{\varphi} \tag{7}$$

where B_m is DC motor viscosity coefficient. The dynamics of DC motor with a 5:1 ratio chain transmission system follows the equations.

$$T_m = 5K_m i \tag{8}$$

$$U = L\frac{di}{dt} + Ri + K_e\dot{\varphi} \tag{9}$$

where K_m, K_e are torque and back emf constants of the motor, respectively. R and L are armature resistance and inductance of the motor, respectively. T_m is torque generated by the motor. By substitution of equation (8) into equation (7), and linearization (6) and (7) around the equilibrium point, the following equations are obtained.

thay phi = 0 vào cả 2 phương trình 6 và 7.
thay theta = 0 vào phương trình 7 và tiến gần tới 0 vào phương trình 6 ???

$$\ddot{\theta}\left[m_b h_b^2 + m_f h_f^2 + I_b + I_r\right] - g(m_b h_b + m_f h_f)\theta - I_p\omega\dot{\varphi} = 0 \quad (10)$$

$$\ddot{\varphi} I_r + I_p\omega\dot{\theta} + B_m\dot{\varphi} - 5K_m i = 0 \quad (11)$$

Define $x = [\theta \quad \dot{\theta} \quad \dot{\varphi} \quad i]'$, $y = \theta$, and $u = U$. The dynamics model of the system in state-space representation by combining (9), (10), and (11) is shown by the following equation.

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases} \quad (12)$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ \dfrac{g(m_b h_b + m_f h_f)}{m_b h_b^2 + m_f h_f^2 + I_b + I_r} & 0 & \dfrac{I_p\omega}{m_b h_b^2 + m_f h_f^2 + I_b + I_r} & 0 \\ 0 & -\dfrac{I_p\omega}{I_r} & -\dfrac{B_m}{I_r} & \dfrac{5K_m}{I_r} \\ 0 & 0 & -\dfrac{K_e}{L} & -\dfrac{R}{L} \end{bmatrix} \quad (13)$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 1/L \end{bmatrix}', \; C = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}, \text{ and } D = \begin{bmatrix} 0 \end{bmatrix} \quad (14)$$

3. PSO-based structure-specified mixed H2/H∞ control

3.1 Structure-specified mixed H2/H∞ control

Consider a single-input single-output (SISO) controlled system as shown in Fig. 5. Assume that the uncertainty of the plant is modeled by multiplicative uncertainty. Where *P(s)* is nominal plant model, *ΔP(s)* is plant perturbation, *K(s)* is controller, *r(t)* is reference input, *e(t)* is tracking error, *d(t)* is external disturbance, and *y(t)* is output of the system (Ho S.J., et al., 2005). The perturbed plant, $\tilde{P}(s)$, is therefore represented by

$$\tilde{P}(s) = P(s)\left(1 + \Delta P(s)\right) \quad (15)$$

with

$$\Delta P(s) = \left(\frac{\tilde{P}(s)}{P(s)} - 1\right) \quad (16)$$

Assume that the plant perturbation, *ΔP(s)*, is upper bound by a known stable weighting function *W1(s)*

$$\left\|\Delta P(s)\right\|_\infty \le \left\|W_1(s)\right\|_\infty \quad (17)$$

where the H∞-norm in (17) is defined as

$$\left\|A(s)\right\|_\infty = \sup_\omega \left|A(j\omega)\right| \quad (18)$$

It is proved that if a controller, *K(s)*, is designed so that:

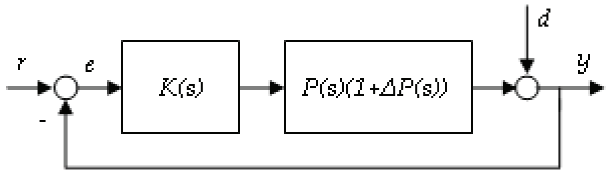


Fig. 5. Control system with plant perturbation and external disturbance

- The nominal closed-loop system ($\Delta P(s) = 0$ and *d(t)* = 0) is asymptotically stable.
- The robust stability performance against plant perturbation satisfies the following inequality

$$J_{\infty,a} = \left\|W_1(s)T(s)\right\|_\infty < 1 \quad (19)$$

The robust stability performance against external disturbance satisfies the following inequality

$$J_{\infty,b} = \left\|W_2(s)S(s)\right\|_\infty < 1 \quad (20)$$

Then, the closed-loop system is also asymptotically stable with $\Delta P(s)$ and *d(t)*, where *W2(s)* is an upper bound stable weighting function of external disturbances *d(t)*, *S(s)* and *T(s)* are sensitivity and complementary sensitivity functions of the system, respectively.

$$S(s) = (1 + P(s)K(s))^{-1} \quad (21)$$

$$T(s) = P(s)K(s)(1 + P(s)K(s))^{-1} \quad (22)$$

In many control systems, not only the robust stability against plant perturbation and external disturbances, but also small tracking error is also important. The problem of minimizing the tracking error of a system can be defined as minimizing the cost function, called the integral of the squared error (ISE)

$$J_2 = \int_0^\infty e^2(t)dt = \left\|E(s)\right\|_2^2 \quad (23)$$

where

$$E(s) = \left(1 + P(s)K(s)\right)^{-1} R(s) \quad (24)$$

In this paper, the structure-specified mixed H2/H∞ control design is defined as finding an admissible structure-specified controller that minimizes the cost function (23) subjected to both constraints (19) and (20).

3.2 PSO algorithm

PSO is one of the most recent evolutionary techniques. The method was developed by simulation of a simplified social model, where each population is called a swarm. In PSO, multiple solutions are together and collaborate simultaneously. Each candidate, called a particle, flies through problem space to look for the optimal position, similar to food searching of bird swarm. A particle adapts its position based on its own knowledge, and knowledge of neighboring particles. The algorithm is initialized with a population of random particles. It searches for the optimal solution by updating particles in generations. Fig. 6 shows the flowchart of PSO algorithm.

Let the search space be *N*-dimensional, then the particle *i* is represented by an *N*-dimensional position vector, $x_i = (x_{i1}, x_{i2}, \ldots, x_{iN})$. The velocity is represented also by an *N*-dimensional velocity vector, $v_i = (v_{i1}, v_{i2}, \ldots, v_{iN})$. The fitness of particles is evaluated by the objective function of the optimization problem. The best previously

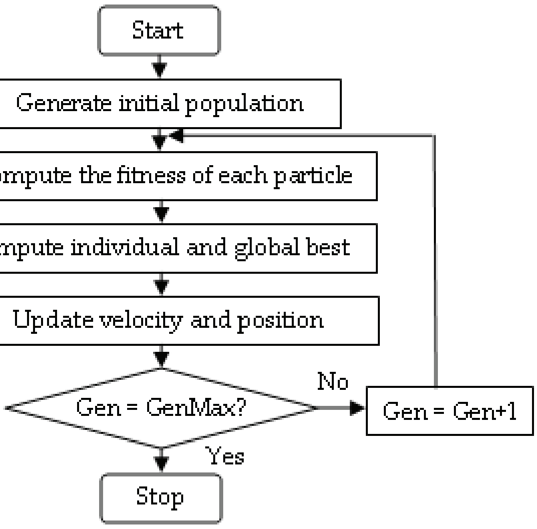

Start
Generate initial population
Compute the fitness of each particle
Compute individual and global best
Update velocity and position
Gen = GenMax?
No
Gen = Gen+1
Yes
Stop

Fig. 6. Flowchart of PSO

function of the optimization problem. The best previously visited position of particle i is noted as its individual best position, $P_i = (p_{i1}, p_{i2}, ..., p_{iN})$. The position of the best individual of the whole swarm is noted as the global best position, $G = (g_1, g_2, ..., g_N)$. At each step of searching process, the velocity of particle and its new position are updated according to the following two equations (Song M.P. & Gu, G.C., 2004).

$$\begin{aligned} v_i(k+1) &= w.v_i(k) + c_1.r_1.(P_i(k) - x_i(k)) \\ &+ c_2.r_2.(G(k) - x_i(k)) \end{aligned} \tag{25}$$

$$x_i(k+1) = x_i(k) + v_i(k) \tag{26}$$

where w, called inertia weight, controls the impact of previous velocity of the particle. r_1, r_2 are random variables in the range of [0,1]. c_1, c_2 are positive constant parameters called acceleration coefficients. The value of each component in v is limited to the range [-v_{max},v_{max}] to control excessive roaming of particles outside the search space.

3.3 PSO-based structure-specified mixed H2/H∞ control

A procedure for designing PSO-based structure-specified mixed H2/H∞ controllers for the problem defined in Section 3.1 is presented below.

Step 1: Define a structure-specified controller of the form

$$K(s) = \frac{N_k(s)}{D_k(s)} = \frac{a_m s^m + a_{m-1} s^{m-1} + ... + a_0}{s^n + b_{n-1} s^{n-1} + ... + b_0} \tag{27}$$

and specify the upper bound of plant uncertainty, *W1(s)*, weighting function for disturbance rejection, *W2(s)*.

Step 2: Set particle i to $x_i = (x_{i1}, x_{i2}, ..., x_{iN}) = (a_0, a_1, ..., b_0, b_1, ..)$, the number of parameters of the controller in equation (27) is the dimension of particle, $N = m + n + 1$. Define maximum number of iterations as GenMax.

Step 3: Initialize a random swarm of H particles as $[x_1 \;\; x_2 \;\; ... \;\; x_H]$, when the swarm size is set to H.

Step 4: For each generation, evaluate objective function of each particle using the objective function expreesed in (23), and also evaluate the constraints (19) and (20). These norms are easily evaluated using MATLAB μ-Analysis and Synthesis Toolbox. The cost function can then be calculated as following:

- If *E(s)* has right half-plane poles, then set $J_2 = \infty$
- If $Max(J_{\infty,a}, J_{\infty,b}) \geq 1$ then set $J_2 = \infty$ else $J_2 = \|E(s)\|_2^2$

Determine the individual best, $P_i(k)$, and the global best, $G(k)$.

Step 5: Update the velocity of particle and its new position using (25) and (26).

Step 6: When the maximum number of iterations is arrived, stop the algorithm. Otherwise go to Step 4.

4. Simulation and experimental results

4.1 Simulation results

Parameters of Bicyrobo are identified as shown in Table 1. By substitution of these parameters into equations (12) – (14), the nominal transfer function of the balancing system of Bicyrobo is described as

$$P(s) = \frac{\theta(s)}{U(s)} = \frac{4887}{s^4 + 683.3s^3 + 1208s^2 + 109700s - 6949} \tag{28}$$

where U is input voltage to the DC motor that controls flywheel control axis, θ is output lean angle of Bicyrobo. Assume that the system is affected by the following parametric uncertainties:

Case 1: The load is added with an additional 10kg, and the flywheel speed is reduced to 147rad/s. The system model thus becomes

$$\tilde{P}_1(s) = \frac{3784}{s^4 + 683.3s^3 + 1162s^2 + 78290s - 6857} \tag{29}$$

Case 2: The load is added with an additional 10kg, and the flywheel speed is increased to 167rad/s. The system model thus becomes

$$\tilde{P}_2(s) = \frac{4299}{s^4 + 683.3s^3 + 1197s^2 + 102300s - 6857} \tag{30}$$

The Bode diagram of transfer functions *ΔP1(s)*, *ΔP2(s)*, and the selected upper bound weighting function *W1(s)* are shown in Fig. 7, when the transfer function of *W1(s)* which meets the condition in (17) is selected as

$$W_1(s) = \frac{0.27(s^2 - 100s + 64)}{s^2 + 11.8s + 50} \tag{31}$$

Suppose that the system is encountering with the external disturbances with a center frequency at 25Hz due to vibration created from rotation of the flywheel at the speed of 157.08 rad/s, thus *W2(s)* with Bode diagram shown in Fig. 8 is selected as

$$W_2(s) = \frac{0.2s + 0.02}{s^2 + 0.4s + 625} \tag{32}$$

To synthesize a PSO-based structure-specified mixed H2/H∞ controller, a first order controller is selected as the following form

Parameters	Value	Unit
m_f	8.1	kg
m_b	43.1	kg
h_f	0.86	m
h_b	0.8	m
I_b	27.584	kg.m^2
I_p	0.215926	kg.m^2
I_r	0.112304	kg.m^2
ω	157.08	rad/s
L	0.0006	H
R	0.41	Ω
B_m	0.000253	kg.m^2/s
K_m	0.119	Nm/A
K_e	0.1184	V.s
g	9.81	m/s^2

Table 1. Parameters of Bicyrobo

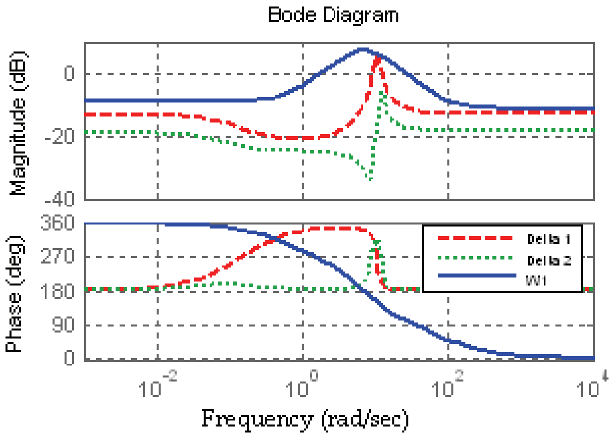


Fig. 7. Bode diagram of weighting function, $W_1(s)$, and plant perturbations

$$K_1(s) = \frac{a_0}{s + b_0} \qquad (33)$$

The following parameters are selected for optimization: Swarm size = 20, the dimension of each particle in the first order controller = 2 (a_0, b_0), $c_1 = c_2 = 2$, GenMax = 100. In the PSO algorithm, the weight, *w*, is dynamically changed so that the algorithm converges slowly to the optimal solution at the end of searching progress to avoid premature convergence. The initial weight is set to *w* = 0.95, and the final weight is set to *w* = 0.4. Velocity limit $[-v_{\max}, v_{\max}]$ is set to [-100,100].

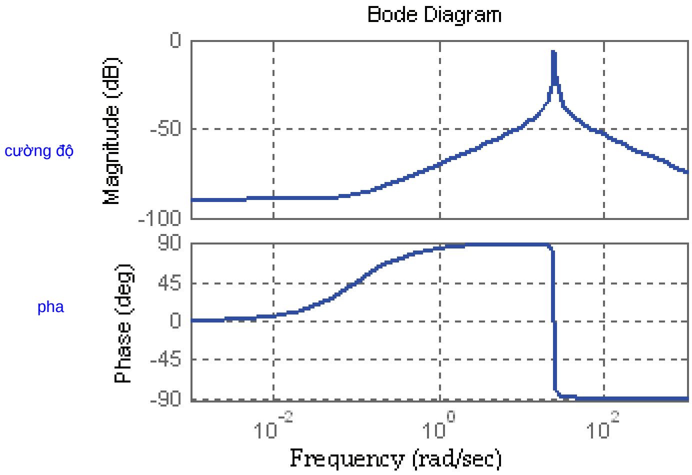


Fig. 8. Bode diagram of weighting function, $W_2(s)$

Select $r(t) = e^{-0.001t}$ instead of step as reference input to avoid pole on the imaginary axis of *E(s)* so that the cost function is finite. The algorithm is run on ten trials, it shows good convergent performance to optimal value. The best value of the cost function $J_2 = 1.08$ is obtained with $J_{\infty,a} = 0.9974$ and $J_{\infty,b} = 0.4977$. The obtained controller in the best case is shown in equation (34).

$$K_{1\_PSO}(s) = \frac{196}{s + 4.88} \qquad (34)$$

To compare PSO-based algorithm with GA-based algorithm, MATLAB Genetic Algorithm and Direct Search Toolbox is used to search for the parameters of the first order controller with the following set up: Population size = 20, parameters initial range = [0,100], GenMax = 100, crossover fraction = 0.8, Gaussian mutation. The algorithm is run on ten trials and the best value of the cost function is $J_2 = 1.085$ with $J_{\infty,a} = 0.9985$ and $J_{\infty,b} = 0.4977$. The obtained controller is shown in equation (35). Fig. 9 shows the convergence of the best trial from both PSO-based and GA-based algorithms. It can be seen that PSO-based algorithm is convergent at about 10 generations whereas GA-based algorithm takes about 70 generations. The comparison of ten runs is shown in Table 2.

$$K_{1\_GA}(s) = \frac{197.33}{s + 4.91} \qquad (35)$$

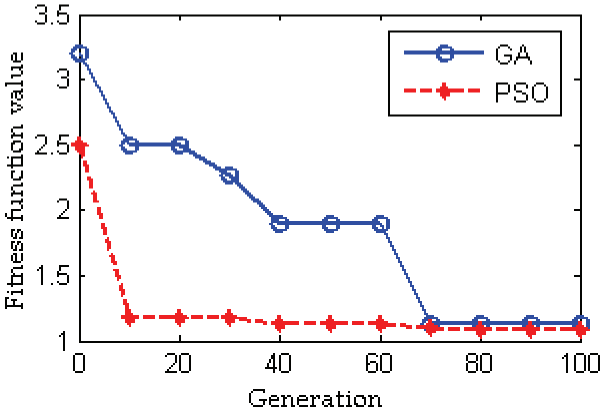


Fig. 9. Convergence of the best fitness for PSO-based and GA-based optimization algorithms

Cost function value (J_2)	PSO-based algorithm	GA-based algorithm
Best	1.0800	1.0850
Worst	1.0921	1.1818
Average	1.0833	1.1274
Standard deviation	0.0034	0.0353

Table 2. Comparison of 10 runs

To compare the computational time of PSO-based and GA-based algorithms, the simulation is conducted on a Pentium 4, 2.8GHz, 512MB RAM computer, in the MATLAB 7.0. The average computational time of ten trials from PSO-based algorithm is 105 seconds whereas it takes 268 seconds for GA-based algorithm.

By tuning parameters K_P and K_D of a PD controller, a step response with about the same response time as the proposed structure-specified controller is obtained. This PD controller is expressed by equation (36). Fig. 10 shows the step responses of closed-loop system using PSO-based controller and GA-based controller in compared with the PD controller. It can be seen that the step responses of PSO-based controller and GA-based controller are similar, they are both better than the step response of the PD controller.

$$K_{pd}(s) = \frac{30s + 2.5}{s} \qquad (36)$$

In order to show that the proposed controller is robust to parameter variations, step responses of the closed-loop system using the proposed controller for the perturbed plants in Case 1 and Case 2 are shown in Fig. 11. It is shown that in both Cases of parameter variations, the system is robustly stable.

4.2 Experimental results

Several experiments are conducted on Bicyrobo with the hardware setup as explained in Section 2.1. The diagram for implementing the controllers on Bicyrobo is shown in Fig. 2. The program is coded in C and implemented on PC/104, a 300MHz CPU embedded computer, with the sampling time of 12.5ms. The lean angle of Bicyrobo is read from the vertical gyro-sensor VG400CC via an A/D extended board. The calculated output voltage from a D/A extended board is sent to an external PWM and Driver circuit to control DC motor of the flywheel control axis. The angular position of the flywheel control axis (φ) is also read from an E6B-CWZ encoder. In the case that the measured lean angle is zero but Bicyrobo is not at the balancing position, the reference must be adjusted to eliminate a continuous rotation in one direction of the flywheel.

bien do

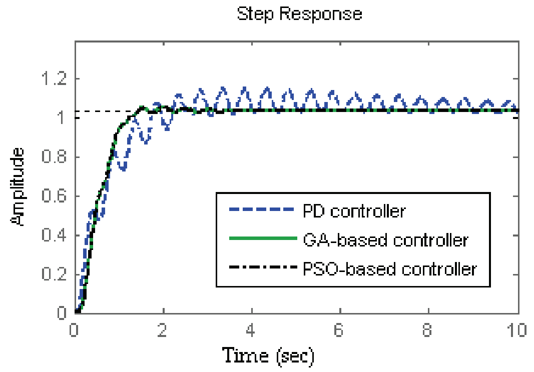


Fig. 10. Step responses of the PD, PSO and GA controllers

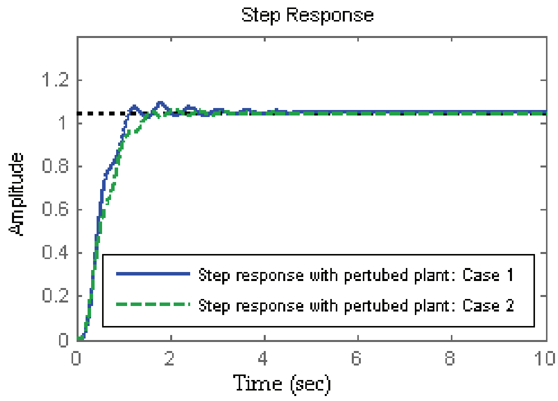


Fig. 11. Robustness against plant perturbations

Various experiments are conducted to evaluate balancing performance and robustness of the proposed controller. The first set of the experiments is tested on the system using the PD and the proposed controllers at a zero forward speed of Bicyrobo without applied masses. The controllers in (34) and (36) are converted to discrete forms (K(z)) and then coded in C on the embedded PC for testing. The lean angles of the system are saved in a file while the program is running. Fig. 12 shows balancing performance from the lean angle of Bicyrobo. The results show that the proposed controller has better balancing performance than the conventional PD controller.

The robust tests to parameter variations are then conducted using the proposed controller. Iron masses of 4kg and 8kg are applied on the system at a zero forward speed of Bicyrobo. The experimental results are shown in Fig. 13. In both cases, the system controlled by the proposed controller is stable against these parameter variations. Photographs of Bicyrobo taken during the experiments with 8kg applied mass are shown in Fig. 14.

5. Conclusion

This paper proposed a PSO-based algorithm for designing structure-specified mixed H_2/H_∞ controllers for a complex and non-convex optimization control problem. The PSO-based algorithm is shown superior to GA-based algorithm for this optimization problem in terms of computational effort, computational time, and convergent speed. The proposed algorithm is successfully applied to design a first order controller for control balancing of Bicyrobo, which is an unstable SISO system associated with many sources of uncertainties due to un-model dynamics, parameter variations, and external disturbances. The algorithm can be extended to other MIMO systems. The simulation and experimental results show the robustness and efficiency of the proposed controller in compared with the conventional PD controller and the GA-based structure-specified mixed H_2/H_∞ controller.

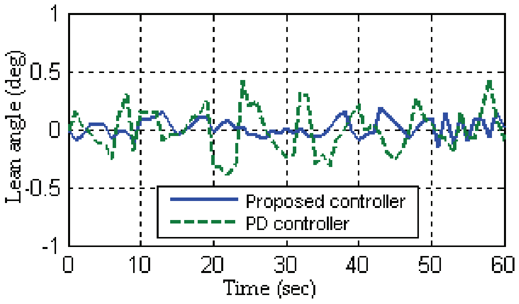


Fig. 12. Balancing performance without applied masses

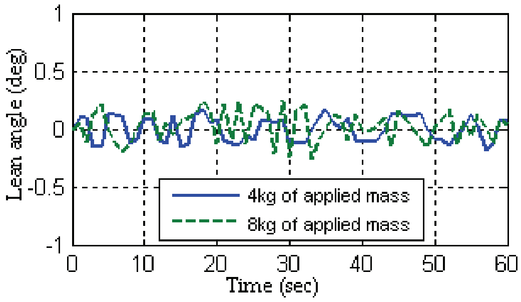
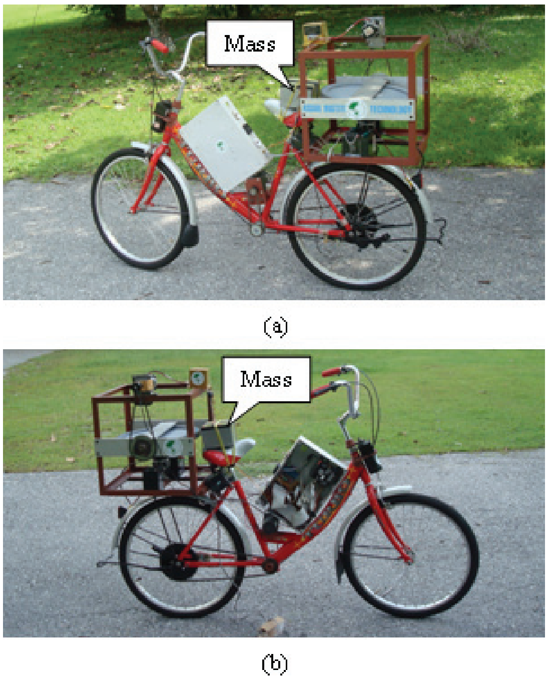


Fig. 13. Balancing performance with applied masses



(a)

(b)

Fig. 14. Photographs during experiment with 8kg of applied mass: (a) left view; (b) right view.

6. References

Bernstein, D.S. & Haddad, W.M. (1989). LQG control with a H_∞ performance bound: A Riccati equation approach. *IEEE Transactions on Automatic Control*, 34(3), pp. 293-305.

Beznos, A.V.; Formalsky, A.M.; et al. (1998). Control of autonomous motion of two-wheel bicycle with gyroscopic stabilization. Proceedings of the 1998 IEEE International Conference on Robotics and Automation, pp. 2670-2675, Leuven, Belgium.

Chang, W.D. (2007). PID control for chaotic synchronization using particle swarm optimization. *Chaos Solitons & Fractals*, In press, Available online 8 April 2007.

Chang, Y.F. (2005). Mixed H_2/H_∞ optimization approach to gap control on EDM. *Control Engineering Practice*, 13(1), pp. 95-104.

Chen, B.S.; Cheng, Y.M.; et al. (1995). A genetic approach to mixed H_2/H_∞ optimal PID control. *IEEE Control System Magine*, 15(5), pp. 51-60.

Fleming, P.J. & Purshouse, R.C. (2002). Evolutionary algorithms in control systems engineering: A survey. *Control Engineering Practice*, 10(9), pp. 1223-1241.

Gallaspy, J.M. (1999). *Gyroscopic stabilization of an unmanned bicycle*, M.S. Thesis, Auburn University.

Ho, S.J.; Ho, S.Y. ; et al. (2004). OSA: Orthogonal simulated annealing algorithm and its application to designing mixed H_2/H_∞ optimal controllers. *IEEE Transactions on Systems, Man and Cybernetics*, 34(5), pp. 588-600.

Ho, S.J.; Ho, S.Y.; et al. (2005). Designing structure-specified mixed H_2/H_∞ optimal controllers using an intelligent genetic algorithm IGA. *IEEE Transactions on Control Systems Technology*, 13(6), pp. 1119-1124.

Kao, C.C.; Chuang, C.W. ; et al. (2006). The self-turning PID control in a slider-crank mechanism system by applying particle swarm optimization approach. *Mechatronics*, 16(8), pp. 513-522.

Kennedy, J. & Eberhart, R. (1995). Particle swarm optimization. Proceedings of the IEEE International Conference on Neural Networks, pp. 1942-1948, Australia.

Khargonekar, P.P. & Rotea, M.A. (1991). Mixed H_2/H_∞ control: A convex optimization approach. *IEEE Transaction on Automatic Control*, 36(7), pp. 824-837.

Krohling, R.A. (1998). Genetic algorithms for synthesis of mixed H_2/H_∞ fixed-structure controllers. Proceedings of the 1998 IEEE ISIC/CIRA/ISAS Joint Conference, pp. 30-35, Maryland.

Lee, S. & Ham, W. (2002). Self-stabilizing strategy in tracking control of unmanned electric bicycle with mass balance. Proceedings of the IEEE/RSJ International Conference on Intelligent Robotics Systems, pp. 2200-2205, Switzerland.

Mukherjee, V. & Ghoshal, S.P. (2007). Intelligent particle swarm optimized fuzzy PID controller for AVR system. *Electric Power Systems Reseach*, 77(12), pp. 1689-1698.

Murata Boy http://www.murataboy.com/en/index.html

Panda, S. & Padhy, N.P. (2008). Comparison of particle swarm optimization and genetic algorithm for FACTS-based controller design. *Applied Soft Computing*, In press, Available online 26 October 2007.

Pereira, G.J. & Araujo, H.X. (2004). Robust output feedback controller design via genetic algorithms and LMIs: The mixed H_2/H_∞ problem. Proceedings of the 2004 American Control Conference, pp. 3309-3314.

Rotea, M.A. & Khargonekar, P.P. (1991). H_2 optimal control with an H_∞ constraint: The state feedback case. *Automatica*, 27(2), pp. 301-316.

Scherer, C.W. (1995). Multi-objective H_2/H_∞ control. *IEEE Transaction on Automatic Control*, 40(6), pp. 1054-1062.

Song, M.P. & Gu, G.C. (2004). Research on particle swarm optimization: A review. Proc. of the third Int Conf Machine Learning Cyber, pp. 2236-2241, Shanghai.

Suprapto, S. (2006). *Development of a gyroscopic unmanned bicycle*. M.Eng. Thesis, AIT, Thailand.

Tanaka, Y. & Murakami, T. (2004). Self sustaining bicycle robot with steering controller. Proceedings of the Int Works Adv Motion Control, pp. 193-197, Japan.

Wong Terence, C.F. & Hung, Y.S. (1996). Stabilization of biped dynamic walking using gyroscopic couple. Proc. of the IEEE Int Syst, pp. 102-108, Rockville.

Wu, B.L.; He, D.L. & Cao, X.B. (2006). Multi-objective output feedback control for satellite formation keeping: An LMI approach. Proc. of the fifth Int Conf Machine Learning Cyber, pp. 528-523, Taiwan