PID Controller Tuning Based on the Covariance Matrix Adaptation Evolution Strategy

Yuji Wakasa Member (Yamaguchi University, wakasa@yamaguchi-u.ac.jp)
Shinji Kanagawa Non-member (Yamaguchi University, n011vk@yamaguchi-u.ac.jp)
Kanya Tanaka Senior Member (Yamaguchi University, ktanaka@eee.yamaguchi-u.ac.jp)
Yuki Nishimura Member (Yamaguchi University, yunishi@yamaguchi-u.ac.jp)

Keywords: PID control, Covariance matrix adaptation evolution strategy, Optimization method

A proportional-integral-derivative (PID) controller is widely used in industrial control systems because of its effectiveness and easy implementation. In general, however, the design problem for optimal robust PID controllers results in a nonconvex optimization problem which suffers from computational intractability and conservatism. To cope with this problem, a tuning method for PID controllers based on the particle swarm optimization (PSO) has been proposed recently.

On the other hand, the covariance matrix adaptation evolution strategy (CMA-ES) has been recently proposed as a particularly reliable and highly competitive optimization method for nonconvex optimization problems. In the CMA-ES, search points, i.e., candidate solutions are sampled according to a multivariate normal distribution, and the covariance matrix of the distribution is adaptively updated through the evaluation of the search points. Since the main idea of the CMA-ES was introduced, it has been improved and extended continuously. It is shown that the CMA-ES algorithm significantly outperforms other algorithms such as a PSO algorithm through benchmark problems. While the CMA-ES algorithm includes some heuristic rules and parameters, these are automatically chosen and determined in the algorithm. Therefore, the CMA-ES is easily tractable for implementation, and has much potential for various applications that need optimization. However, there are few control applications of the CMA-ES except for the only one paper.

In this paper, we investigate the applicability of the CMA-ES algorithm to control problems. To this end, we deal with a typical PID control problem with constraints on sensitivity and complementary sensitivity functions, and propose a PID controller tuning method based on the CMA-ES. More concretely, we consider the PID control system in Fig. 1. Let $S(s)$ and $T(s)$ be the sensitivity function and the complementary sensitivity function, respectively. The design problem is to minimize

$$\lambda_{\max}(k_p, t_i, t_d, N_d) = \arg\max_i \{\mathrm{Re}(\lambda_i(k_p, t_i, t_d, N_d))\}$$

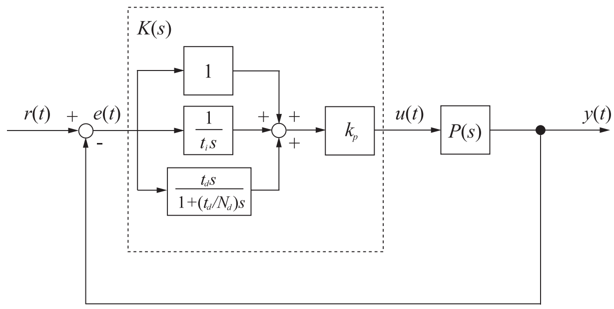


Fig. 1. PID feedback control system.

subject to

$$\sup_{\omega\geq 0} |W_S(j\omega)S(j\omega)| < 1,$$
$$\sup_{\omega\geq 0} |W_T(j\omega)T(j\omega)| < 1,$$

where λ_i denotes the ith pole of the closed-loop system $T(s)$, and $W_S(s)$ and $W_T(s)$ are stable frequency-dependent weighting functions.

Since the original CMA-ES is developed for an optimization problem without constraints, we transform the PID control problem into an optimization problem without constraints by using a kind of penalty method. We show the effectiveness of the proposed method by comparing it with the PSO-based method through three numerical examples.

As a result, we have obtained the following observations through the three examples.

- The CMA-ES algorithm gave a better performance than the PSO one from the view point of convergence speed and searchability of an optimal value.
- The CMA-ES algorithm is apparently more complicated than the PSO one. However, the computation time by the CMA-ES was shorter than or comparable to that by the PSO.

From the above observations, we have concluded that the CMA-ES is applicable and effective to the PID control problems.

Paper

PID Controller Tuning Based on the Covariance Matrix Adaptation Evolution Strategy

Yuji Wakasa* Member
Shinji Kanagawa* Non-member
Kanya Tanaka* Senior Member
Yuki Nishimura* Member

The covariance matrix adaptation evolution strategy (CMA-ES) is a kind of stochastic optimization such as particle swarm optimization (PSO), and has been shown to have a good performance. However, there are few control applications of the CMA-ES except for only one paper. This paper deals with a PID control problem with constraints on sensitivity and complementary sensitivity functions, and proposes a PID controller tuning method based on the CMA-ES. Numerical examples are given to show the effectiveness of the proposed method in comparison with the recently proposed PSO-based method.

Keywords: PID control, Covariance matrix adaptation evolution strategy, Optimization method

1. Introduction

A proportional-integral-derivative (PID) controller is widely used in industrial control systems because of its effectiveness and easy implementation. In general, however, the design problem for optimal robust PID controllers results in a nonconvex optimization problem which suffers from computational intractability and conservatism. To cope with this problem, a tuning method for PID controllers based on the particle swarm optimization (PSO) has been proposed recently [1] [2].

On the other hand, the covariance matrix adaptation evolution strategy (CMA-ES) has been proposed as a particularly reliable and highly competitive optimization method for nonconvex optimization problems [3]. In the CMA-ES, search points, i.e., candidate solutions are sampled according to a multivariate normal distribution, and the covariance matrix of the distribution is adaptively updated through the evaluation of the search points. Since the main idea of the CMA-ES [4] was introduced, it has been improved and extended continuously [5]–[7]. In [8] [9], it is shown that the CMA-ES algorithm significantly outperforms other algorithms such as a PSO algorithm through benchmark problems. While the CMA-ES algorithm includes some heuristic rules and parameters, these are automatically chosen and determined in the algorithm. Therefore, the CMA-ES is easily tractable for implementation, and has much potential for various applications that need optimization. However, there are few control applications of the CMA-ES except for the only one paper [10]. Whereas, in [10], a method for handling uncertainty in optimization has been proposed in the framework of the CMA-ES and applied to an experimental combustion control system, there is no comparison with other optimization-based control methods, and therefore, we cannot see the superiority of the CMA-ES in control problems.

In this paper, we investigate the applicability and effectiveness of the CMA-ES algorithm to control problems. To this end, we deal with a typical PID control problem with constraints on sensitivity and complementary sensitivity functions, and propose a PID controller tuning method based on the CMA-ES. Since the original CMA-ES is developed for an optimization problem without constraints, we transform the PID control problem into an optimization problem without constraints by using a kind of penalty method. We show the effectiveness of the proposed method by comparing it with the PSO-based method [2] through numerical examples.

2. PID Control Problem

Consider the standard PID feedback control system shown in Fig. 1, where $r(t)$ is the reference, $u(t)$ is the control input, $y(t)$ is the control output, $e(t)$ is the control error, and $P(s)$ is the linear time-invariant system.

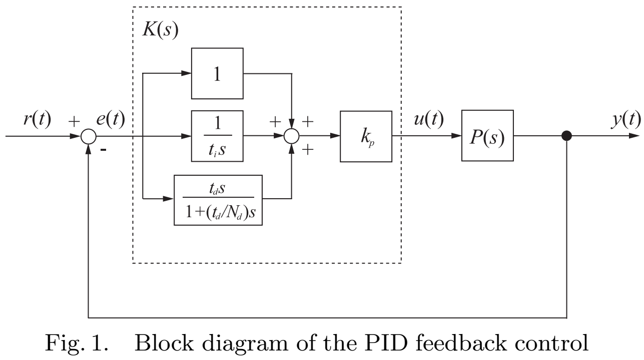


Fig. 1. Block diagram of the PID feedback control system.

* Graduate School of Science and Engineering, Yamaguchi University
2-16-1, Tokiwadai, Ube, Yamaguchi 755-8611

$K(s)$ denotes the PID controller which is augmented by a low pass filter on the derivative part and is given as

$$K(s) = k_p \left(1 + \frac{1}{t_i s} + \frac{t_d s}{1 + (t_d/N_d)s}\right), \cdots\cdots\cdots \quad (1)$$

where k_p, t_i, t_d, and t_d/N_d are the proportional gain, the integral time, the derivative time, and the filter time constant, respectively. For the system in Fig. 1, the loop transfer function is $L(s) = P(s)K(s)$, and then the sensitivity function and the complementary sensitivity function are defined as $S(s) = 1/(1 + L(s))$ and $T(s) = L(s)/(1 + L(s))$, respectively.

In this paper, the robust performance criteria based on $S(s)$ and $T(s)$ are given as

$$\sup_{\omega \geq 0} |W_S(j\omega)S(j\omega)| < 1, \cdots\cdots\cdots \quad (2)$$

$$\sup_{\omega \geq 0} |W_T(j\omega)T(j\omega)| < 1, \cdots\cdots\cdots \quad (3)$$

where $W_S(s)$ and $W_T(s)$ are stable frequency-dependent weighting functions concerning the required stability and performance specifications. Taking into account the stability and transient response of the closed-loop system, we adopt the following objective function

$$\lambda_{\max}(k_p, t_i, t_d, N_d) = \max_i \{\mathrm{Re}(\lambda_i(k_p, t_i, t_d, N_d))\} \cdots\cdots\cdots \quad (4)$$

where λ_i denotes the ith pole of the closed-loop system $T(s)$. Denoting the optimization variable by $\boldsymbol{x} := (k_p, t_i, t_d, N_d)^T$, we formulate a PID control problem as follows:

$$\min_{\boldsymbol{x} \in \mathbb{D}} f(\boldsymbol{x}) := \lambda_{\max}(\boldsymbol{x}) \cdots\cdots\cdots \quad (5)$$

subject to

$$\boldsymbol{h}(\boldsymbol{x}) := \begin{bmatrix} \|W_S(s)S(s;\boldsymbol{x})\|_\infty - 1 \\ \|W_T(s)T(s;\boldsymbol{x})\|_\infty - 1 \\ \lambda_{\max}(\boldsymbol{x}) \end{bmatrix} < \boldsymbol{0}, \cdots\cdots \quad (6)$$

where $\|\cdot\|_\infty$ denotes the H_∞ norm, and $\mathbb{D} \subseteq \mathbb{R}^4$ is a search space. To apply the CMA-ES to the problem (5), we use the technique proposed in [2], and transform the objective function as follows:

$$f_v(\boldsymbol{x}) := \arctan\{f(\boldsymbol{x})\} - \frac{\pi}{2}. \cdots\cdots\cdots \quad (7)$$

Then, based on the above virtual objective function $f_v(\boldsymbol{x})$, the original constrained optimization problem in (5) is modified as the following unconstrained one:

$$\min_{\boldsymbol{x} \in \mathbb{D}} \tilde{f}(\boldsymbol{x}) \cdots\cdots\cdots \quad (8)$$

with

$$\tilde{f}(\boldsymbol{x}) := \begin{cases} h_{\max}(\boldsymbol{x}) & \text{if } h_{\max}(\boldsymbol{x}) \geq 0 \\ f_v(\boldsymbol{x}) & \text{otherwise,} \end{cases} \cdots\cdots\cdots \quad (9)$$

where $h_{\max}(\boldsymbol{x}) := \max\{h_1(\boldsymbol{x}), h_2(\boldsymbol{x}), h_3(\boldsymbol{x})\}$. This technique is the so-called exact penalty function method with a sufficiently large penalty parameter [11].

3. CMA-ES

We use the CMA-ES algorithm [3] to solve the above PID control problem. The CMA-ES is an evolution strategy where search points are generated according to a multivariate normal distribution, and the corresponding covariance matrix is updated through the evaluation of the search points. The algorithm mainly consists of the generation of search points, the update of the mean of the search points, the update of the covariance matrix of the search distribution, and the update of the step size parameter.

The dimension of the search space is $n = 4$ for the problem settings in the previous section. The outline of the CMA-ES algorithm is as follows.

CMA-ES algorithm for the PID control problem

Step 0. (Setting of parameters) Set the number of search points N, the number of selected search points μ, the weights $w_i, i = 1, \ldots, \mu$, and the other parameters c_σ, d_σ, c_c, $\mu_\mathrm{cov}(= \mu_\mathrm{eff})$, c_cov.

Step 1. (Initialization) Initialize the generation number $g = 0$, the covariance matrix $\boldsymbol{C}^{(0)} = \boldsymbol{I}$, the step size parameter $\sigma^{(0)} = 0.5$, and the evolution path parameters $\boldsymbol{p}_\sigma = \boldsymbol{0}$, $\boldsymbol{p}_c = \boldsymbol{0}$. Set the initial mean $\boldsymbol{m}^{(0)} \in \mathbb{D}$.

Step 2. (Generation of search points) Search points are generated according to the normal distribution as follows:

$$\boldsymbol{x}_k^{(g+1)} \sim \mathcal{N}\left(\boldsymbol{m}^{(g)}, (\sigma^{(g)})^2 \boldsymbol{C}^{(g)}\right) \text{ for } k = 1, \ldots, N. \cdots\cdots\cdots \quad (10)$$

Step 3. (Update of the mean) Evaluate the function values $\tilde{f}(\boldsymbol{x}_k^{(g+1)})$ of the search points, and select μ best points $\boldsymbol{x}_{i:N}^{(g+1)}, i = 1, \ldots, \mu$. Update the mean of the search distribution as follows:

$$\boldsymbol{m}^{(g+1)} = \sum_{i=1}^{\mu} w_i \boldsymbol{x}_{i:N}^{(g+1)}. \cdots\cdots\cdots \quad (11)$$

Step 4. (Update of the covariance matrix) Update the covariance matrix as follows:

$$\boldsymbol{C}^{(g+1)} = (1 - c_\mathrm{cov})\boldsymbol{C}^{(g)} + c_\mathrm{cov}\left(1 - \frac{1}{\mu_\mathrm{cov}}\right) \cdot \sum_{i=1}^{\mu} w_i \left(\frac{\boldsymbol{x}_{i:N}^{(g+1)} - \boldsymbol{m}^{(g)}}{\sigma^{(g)}}\right)\left(\frac{\boldsymbol{x}_{i:N}^{(g+1)} - \boldsymbol{m}^{(g)}}{\sigma^{(g)}}\right)^T + \frac{c_\mathrm{cov}}{\mu_\mathrm{cov}}\left(\boldsymbol{p}_\mathrm{c}^{(g+1)} \boldsymbol{p}_\mathrm{c}^{(g+1)^T} + \delta(H_\sigma)\, \boldsymbol{C}^{(g)}\right), \cdots\cdots\cdots \quad (12)$$

where

$$c_\mathrm{cov} = \frac{1}{\mu_\mathrm{cov}} \frac{2}{(n + \sqrt{2})^2} + \left(1 - \frac{1}{\mu_\mathrm{cov}}\right) \cdot \min\left(1, \frac{2\mu_\mathrm{cov} - 1}{(n + 2)^2 + \mu_\mathrm{cov}}\right), \cdots\cdots\cdots \quad (13)$$

$$\boldsymbol{p}_c^{(g+1)} = (1-c_c)\,\boldsymbol{p}_c^{(g)} + H_\sigma \sqrt{c_c(2-c_c)\mu_{cov}}\,\frac{\boldsymbol{m}^{(g+1)}-\boldsymbol{m}^{(g)}}{\sigma^{(g)}}, \quad \cdots\cdots (14)$$

$$\delta(H_\sigma) = (1-H_\sigma)c_c(2-c_c), \quad \cdots\cdots (15)$$

$$H_\sigma = \begin{cases} 1 & \text{if } \frac{\|\boldsymbol{p}_\sigma\|}{\sqrt{1-(1-c_\sigma)^{2(g+1)}}} < (1.4+\frac{2}{n+1})\mathrm{E}\|\mathcal{N}(\boldsymbol{0},\boldsymbol{I})\| \\ 0 & \text{otherwise,} \end{cases} \quad \cdots\cdots (16)$$

$$\mathrm{E}\|\mathcal{N}(\boldsymbol{0},\boldsymbol{I})\| = \sqrt{n}\left(1-\frac{1}{4n}+\frac{1}{21n^2}\right). \quad \cdots\cdots (17)$$

Step 5. (Update of the step size parameter) Update the step size parameter as follows:

$$\sigma^{(g+1)} = \sigma^{(g)} \exp\left(\frac{c_\sigma}{d_\sigma}\left(\frac{\|\boldsymbol{p}_\sigma^{(g+1)}\|}{\mathrm{E}\|\mathcal{N}(\boldsymbol{0},\boldsymbol{I})\|}-1\right)\right), \quad \cdots\cdots (18)$$

where

$$\boldsymbol{p}_\sigma^{(g+1)} = (1-c_\sigma)\,\boldsymbol{p}_\sigma^{(g)} + \sqrt{c_\sigma(2-c_\sigma)\mu_{\mathrm{eff}}}\left(\boldsymbol{C}^{(g)}\right)^{-\frac{1}{2}} \cdot \frac{\boldsymbol{m}^{(g+1)}-\boldsymbol{m}^{(g)}}{\sigma^{(g)}}. \quad \cdots\cdots (19)$$

Step 6. (Stopping criterion) If the stopping criterion is satisfied, stop. Otherwise, set $g = g+1$ and go to Step 2.

Although the above algorithm includes some parameters to be set in Step 0, these parameters including even the number of search points N are automatically determined in the CMA-ES algorithm developed from theoretical and empirical viewpoints [3]. Therefore, note that the CMA-ES algorithm does not need so much effort for parameter tuning. Basically, we adopt such parameter settings in the CMA-ES algorithm [3]. However, the number of search points N is an exception because we fairly compare the CMA-ES-based method with the PSO-based method by using the same number of search points in the following examples.

4. Numerical Examples

We consider the three examples shown in [1] where an augmented Lagrangian PSO (ALPSO) is used. As stated in [2], the more recently proposed PSO-based method [2] has a better performance than the ALPSO-based method [1]. Therefore, we compare our method with the PSO-based method in [2]. The two algorithms are run on a computer with 2.4GHz Core Duo CPU and 1024MB RAM.

4.1 Example 1

The plant model $P(s)$ and the weighting functions $W_S(s)$, $W_T(s)$ are as follows:

$$P(s) = \frac{7.147}{(s-22.55)(s+20.9)(s+13.99)}, \quad \cdots (20)$$

$$W_S(s) = \frac{5}{s+0.1}, \quad \cdots\cdots (21)$$

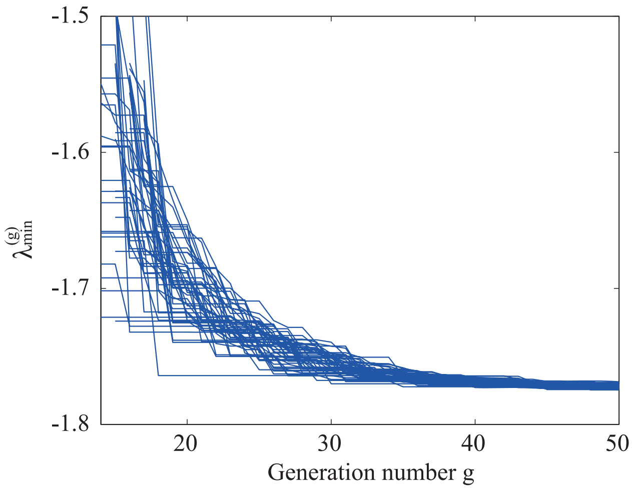


Fig. 2. Convergence property of the CMA-ES in Example 1.

$$W_T(s) = \frac{(s+0.066)(s+31.4)(s+88)}{2.2796\times10^{-5}(s+10^4)^3}. \quad \cdots\cdots (22)$$

As in [1] [2], we use the following variable transformation for a broader search space

$$\boldsymbol{x} := (x_1,x_2,x_3,x_4)^T = (\log_{10}k_p, \log_{10}t_i, \log_{10}t_d, \log_{10}N_d)^T. \quad \cdots\cdots (23)$$

This means that the PID controller is expressed as

$$K(s) = 10^{x_1}\left(1+\frac{1}{10^{x_2}s}+\frac{10^{x_3}s}{1+10^{(x_3-x_4)}s}\right). \quad \cdots\cdots (24)$$

Let the number of search points be $N = 100$ and the search space be

$$\mathbb{D} := \left\{\boldsymbol{x}\in\mathbb{R}^4 \mid (2,-1,-1,2)^T \le \boldsymbol{x} \le (4,1,1,3)^T\right\}. \quad \cdots\cdots (25)$$

These settings correspond to the conditions of Example 1 in [1] [2]. The number of selected search points is set as $\mu = 50$. The maximum number of generations $g = 50$ is used as a stopping criterion. The other parameters are set to the default values in the CMA-ES algorithm which can be downloaded as a MATLAB program from [3]. We run the CMA-ES algorithm 50 times with different initial search points of $\boldsymbol{x} \in \mathbb{D}$. Then, we succeed in obtaining feasible solutions (i.e., controllers satisfying the constraints are obtained) perfectly (100% success rate). The convergence property of $\lambda_{\min}^{(g)}$ is illustrated in Fig. 2, where $\lambda_{\min}^{(g)}$ is defined as

$$\lambda_{\min}^{(g)} := \min_{j=1,\dots,g}\ \min_{k=1,\dots,N}\{\lambda_{\max}(\boldsymbol{x}_k^{(j)}) | \boldsymbol{x}_k^{(j)} \text{is feasible}\}. \quad \cdots\cdots (26)$$

For comparison, we rerun the PSO algorithm proposed in [2] 50 times and show its result in Fig. 3. Note that the number of function evaluations at each iteration (generation) in the PSO is 100, which is the same as in the

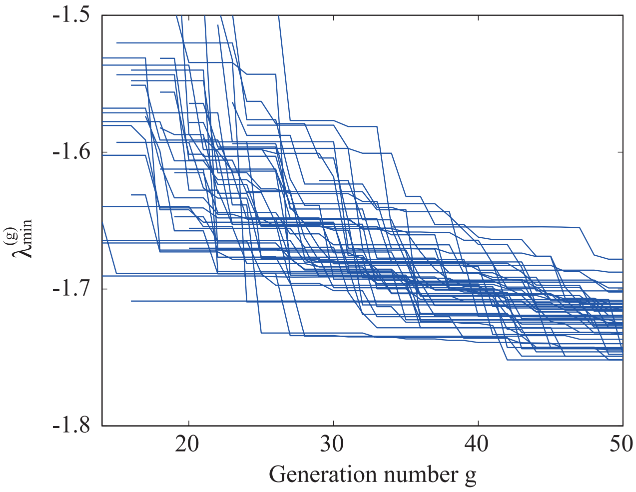


Fig. 3. Convergence property of the PSO in Example 1.

Table 1. Statistical result of Example 1.

$\lambda_{\min}^{(g)}$	Best	Mean	Worst
CMA-ES	-1.7745	-1.7719	-1.7687
PSO	-1.7519	-1.7238	-1.6782

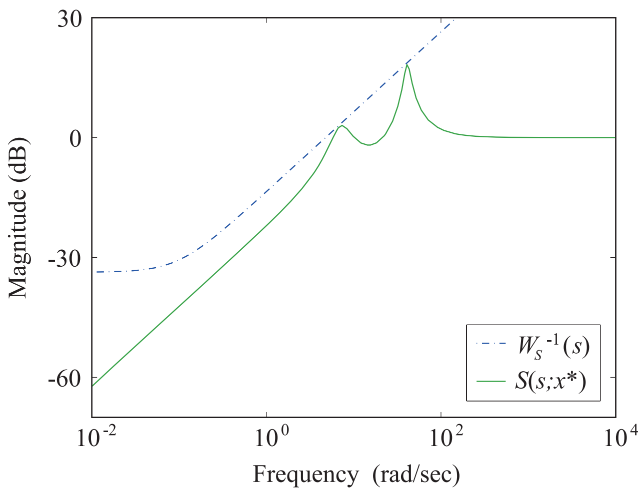


Fig. 4. Bode plot of the sensitivity function $S(s;\boldsymbol{x}^*)$ in Example 1.

CMA-ES. The average computation time of the CMA-ES is 262 seconds while that of the PSO is 274 seconds. From Figs. 2 and 3, we see that the CMA-ES algorithm converges faster than the PSO one. Moreover, to compare the performances of the two algorithms, we show the statistical result in Table 1. We see from the table that the CMA-ES algorithm gives a better performance than the PSO one. Even when the PSO converges sufficiently by 400 iterations as shown in [2], the success rate is 93%, which shows that the CMA-ES is superior to the PSO from the viewpoints of searchability.

The best design parameter obtained in the experiments is $\boldsymbol{x}^* = (3.2560, -0.8321, -0.7595, 2.3323)^T$, and the corresponding PID controller is obtained from (24). The plots of the sensitivity function $S(s;\boldsymbol{x}^*)$ and the complementary sensitivity function $T(s;\boldsymbol{x}^*)$ are shown in Figs. 4 and 5, respectively. These figures demonstrate that the given constraints are satisfied in a non-conservative way.

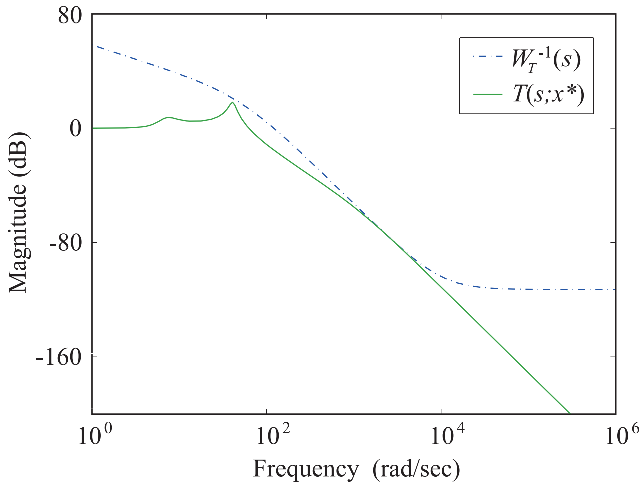


Fig. 5. Bode plot of the complementary sensitivity function $T(s;\boldsymbol{x}^*)$ in Example 1.

4.2 Example 2 Consider the system in [1][12], where the plant and the PID controller are given as follows:

$$P(s) = \frac{s-1}{s^2+0.8s-0.2}, \quad \cdots\cdots (27)$$

$$K(s;\boldsymbol{x}) = x_1 + \frac{x_2}{s} + x_3 s, \quad \cdots\cdots (28)$$

where $\boldsymbol{x} := (x_1, x_2, x_3)^T = (k_p, k_i, k_d)^T \in \mathbb{R}^3$ denotes the search point. As in [1][12], we consider the constraint only on the complementary sensitivity function as shown by $\|W_T(s)T(s;\boldsymbol{x})\|_\infty < 1$ where

$$W_T(s) = \frac{s+0.1}{s+1}. \quad \cdots\cdots (29)$$

For the above system, we apply the CMA-ES algorithm to find the optimal gains $\boldsymbol{x}^* = (k_p^*, k_i^*, k_d^*)^T$. All parameters in the CMA-ES algorithm are set identically to those of Example 1. The search space is

$$\mathbb{D} := \left\{\boldsymbol{x} \in \mathbb{R}^3 | (-1.8, -1, -1)^T \le \boldsymbol{x} \le (-0.2, 1, 1)^T\right\}. \quad \cdots\cdots (30)$$

The optimal gains are found as $\boldsymbol{x}^* = (k_p^*, k_i^*, k_d^*)^T = (-0.5519, -0.0446, -0.4583)^T$ for which $\lambda_{\max}(\boldsymbol{x}^*) = -0.4343$, and $\|W_T(s)T(s;\boldsymbol{x}^*)\|_\infty < 1$ is guaranteed. The plot of the resulting complementary sensitivity function $T(s;\boldsymbol{x}^*)$ is shown in Fig. 6. This figure also demonstrates that the given constraint is satisfied in a non-conservative way.

As in Example 1, we show the convergence properties by the CMA-ES and the PSO in Figs. 7 and 8, respectively. The average computation time of the CMA-ES is 105 seconds while that of the PSO is 106 seconds. Also, we show the statistical result by the two algorithms in Table 2. From these figures and table, we see that the CMA-ES algorithm gives a better performance than the PSO one.

4.3 Example 3 Consider a mixed sensitivity

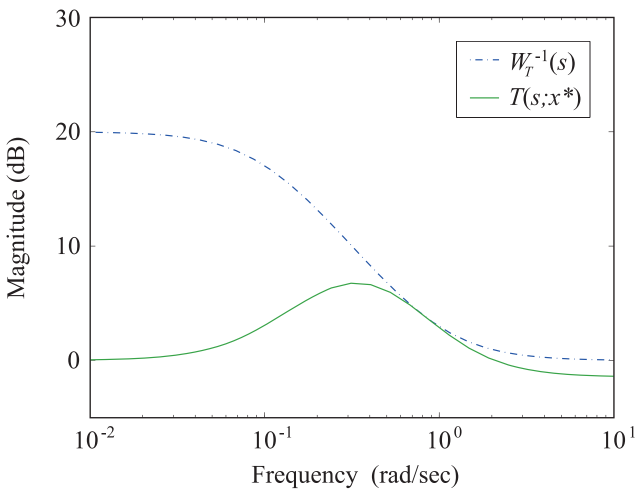


Fig. 6. Bode plot of the complementary sensitivity function $T(s;\boldsymbol{x}^*)$ in Example 2.

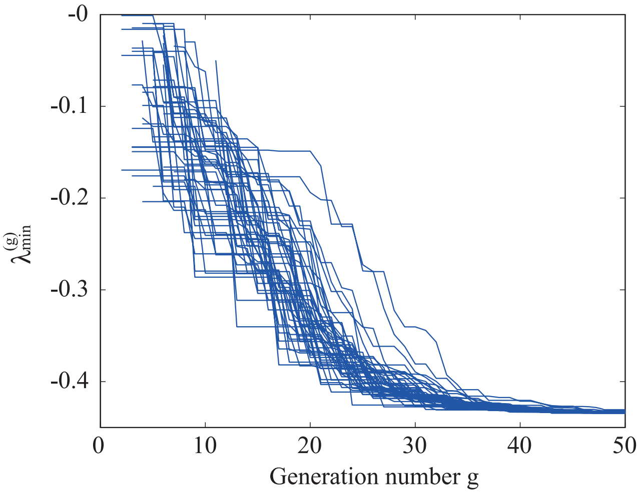


Fig. 7. Convergence property of the CMA-ES in Example 2.

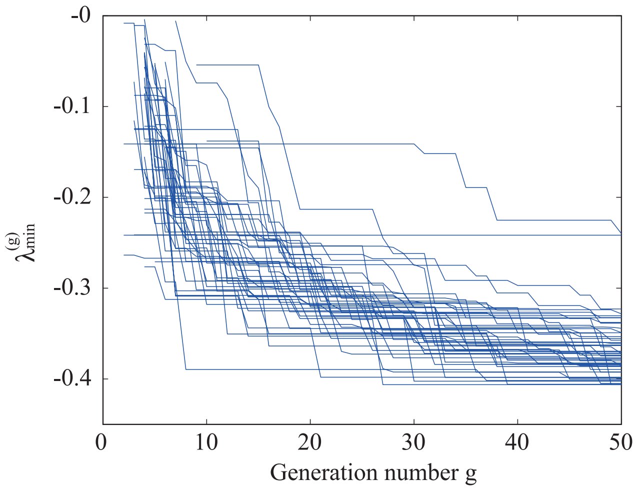


Fig. 8. Convergence property of the PSO in Example 2.

control problem in Fig. 9 presented in [1], where

$$P(s)=\begin{bmatrix}1/(s+1) & 0.2/(s+3)\\ 0.1/(s+2) & 1/(s+1)\end{bmatrix},\cdots\cdots\cdots (31)$$

Table 2. Statistical result of Example 2.

$\lambda_{\min}^{(g)}$	Best	Mean	Worst
CMA-ES	-0.4346	-0.4337	-0.4207
PSO	-0.4063	-0.3673	-0.2400

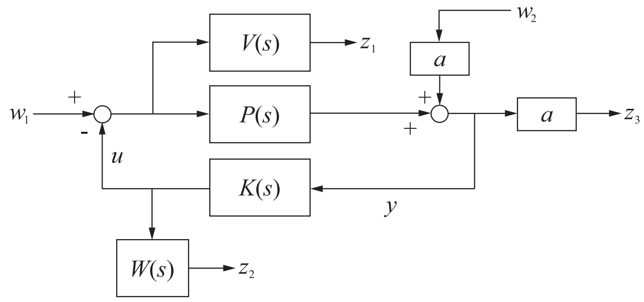


Fig. 9. Mixed sensitivity control problem.

$V(s)=v_1(s)\boldsymbol{I}_2$, $W(s)=v_2(s)\boldsymbol{I}_2$, $v_1(s)=(s+3)/(3s+0.3)$, $v_2(s)=(10s+2)/(s+40)$, and $a=0.01$. For the above system, the design problem is to develop a decentralized PID controller having the following structure:

$$K(s)=\begin{bmatrix}k_{p_1} & 0\\ 0 & k_{p_2}\end{bmatrix}+\begin{bmatrix}k_{i_1} & 0\\ 0 & k_{i_2}\log\end{bmatrix}\frac{1}{s}$$
$$+\begin{bmatrix}k_{d_1} & 0\\ 0 & k_{d_2}\end{bmatrix}\frac{s}{1+0.01s}\cdots\cdots\cdots\cdots\cdots (32)$$

which should satisfy $\|T_{zw}(s)\|_\infty<1$ where $T_{zw}(s)$ denotes the closed-loop transfer function from w to z. Let $\boldsymbol{x}:=(x_1,x_2,x_3,x_4,x_5,x_6)^T=(\log_{10}k_{p_1},\log_{10}k_{i_1},\log_{10}k_{d_1},\ \log_{10}k_{p_2},\ \log_{10}k_{i_2},\ \log_{10}\ k_{d_2})^T$ denote the search point. In this example, the PID control problem shown in Section 2 is slightly modified. The objective function is defined as $f(\boldsymbol{x})=\|T_{zw}(s;\boldsymbol{x})\|_\infty$, and the virtual objective function $f_v(\boldsymbol{x})$ is defined as

$$f_v(\boldsymbol{x})=-\|T_{zw}(s;\boldsymbol{x})\|_\infty^{-1}.\cdots\cdots\cdots\cdots\cdots\cdots\cdots (33)$$

Also, the following constraint is introduced to guarantee the stability of the system in Fig. 9

$$h(\boldsymbol{x}):=\lambda_{\max}(\boldsymbol{x})+\epsilon<0,\cdots\cdots\cdots\cdots\cdots\cdots (34)$$

where $\epsilon=10^{-4}$. All design parameters in the CMA-ES algorithm are set identically to those of Examples 1 and 2. The search space is set as

$$\mathbb{D}:=\left\{\boldsymbol{x}\in\mathbb{R}^6\,|-(3,3,\ldots,3)^T\leq\boldsymbol{x}\leq(3,3,\ldots,3)^T\right\}.$$
$$\cdots\cdots\cdots\cdots\cdots\cdots\cdots (35)$$

In this example, we perform the CMA-ES algorithm once. As a result, we obtain the best controller gains $(x_1^*,x_2^*,x_3^*,x_4^*,x_5^*,x_6^*)^T=(0.25405, 0.22484, -1.689, 0.25843, 0.3161, -1.5712)^T$. The objective function value $f(\boldsymbol{x}^*)=\|T_{zw}(s;\boldsymbol{x}^*)\|_\infty$ corresponding to the best controller gains is 0.58719. We also run the PSO algorithm under the same condition as in the CMA-ES one. The resulting best objective function value is $f(\boldsymbol{x}^*)=0.59085$, which means that our approach is not worse than the result of the PSO.

The computation time of the CMA-ES is 11976 seconds while that of the PSO is 13038 seconds. Therefore, we see that the CMA-ES algorithm gives a better performance than the PSO one.

5. Conclusion

We have proposed a PID controller tuning method based on the CMA-ES, and have compared it with the PSO algorithm [2] which have recently proposed as an effective technique. As a result, we have obtained the following observations through the three examples.

- The CMA-ES algorithm gave a better performance than the PSO one from the viewpoints of convergence speed and quality of the final solution.
- The CMA-ES algorithm is apparently more complicated than the PSO one. However, the computation time by the CMA-ES was shorter than or comparable to that by the PSO.

In addition, we see that we need to set just a few parameters in the CMA-ES in spite of its complicated calculation, which leads to easy implementation. From the above observations, we have concluded that the CMA-ES is applicable and effective to the PID control problems.

(Manuscript received Aug. 28, 2009,
revised Jan. 27, 2010)

References

(1) T.-H. Kim, I. Maruta, and T. Sugie: "Robust PID Controller Tuning Based on the Constrained Particle Swarm Optimization", *Automatica*, Vol.44, pp.1104–1110 (2008)

(2) I. Maruta, T. H. Kim, and T. Sugie: "Fixed-structure H∞ Controller Synthesis: A Meta-heuristic Approach Using Simple Constrained Particle Swarm Optimization", *Automatica*, Vol.45, pp.553–559 (2009)

(3) N. Hansen: "The CMA Evolution Strategy: A Tutorial", (2009) http://www.lri.fr/˜hansen/cmatutorial.pdf

(4) N. Hansen: "On the Adaptation of Arbitrary Normal Mutation Distributions in Evolution Strategies: The Generating Set Adaptation", Proc. 6th International Conference on Genetic Algorithms, pp.57–64 (1995)

(5) N. Hansen and A. Ostermeier: "Adapting Arbitary Normal Mutation Distributions in Evolution Strategies: The Covariance Matrix Adaptation", Proc. 1996 IEEE International Conference on Evolutionary Computation, pp.312–317 (1996)

(6) N. Hansen and A. Ostermeier: "Convergence Properties of Evolution Strategies with the Derandomized Covariance Matrix Adaptation: The $(\mu/\mu_I, \lambda)$-CMA-ES", Proc. 5th European Congress on Intelligent Techniques and Soft Computing, pp.650–654 (1997)

(7) N. Hansen and A. Ostermeier: "Completely Derandomized Self-Adaptation in Evolution Strategies", Evolutionary Computation, Vol.9, No.2, pp.159–195 (2001)

(8) A. Auger and N. Hansen: "A Restart CMA Evolution Strategy with Increasing Population Size", Proc. IEEE Congress on Evolutionary Computation 2005, pp.1769–1776 (2005)

(9) A. Auger and N. Hansen: "Performance Evaluation of an Advanced Local Search Evolutionary Algorithm", Proc. IEEE Congress on Evolutionary Computation 2005, pp.1777–1784 (2005)

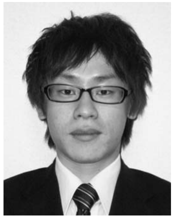
(10) N. Hansen, A. S. P. Niederberger, L. Guzzella, and P. Koumoutsakos: "A Method for Handling Uncertainty in Evolutionary Optimization with an Application to Feedback Control of Combustion", *IEEE Trans. Evolutionary Computation*, Vol.13, No.1, pp.180–197 (2009)

(11) W. I. Zangwill: "Non-linear Programming via Penalty Functions", *Management Science*, Vol.13, No.5, pp.344–358 (1967)

(12) M. Ho: "Synthesis of H_∞ PID Controllers: A Paramtric Approach", *Automatica*, Vol.39, No.6, pp.1069–1075 (2003)

Yuji Wakasa (Member) received the B.S. and M.S. degrees in engineering and the degree of doctor of informatics all from Kyoto University, Japan, in 1992, 1994 and 2000, respectively. He was a research associate at Okayama University and Kyoto University, from 1994 to 1998 and from 1998 to 2002, respectively. Since October 2002, he has been with the Graduate School of Science and Engineering, Yamaguchi University. He is currently an associate professor. His current research interests include control system design via mathematical programming and its applications. He is a member of SICE, ISCIE, ORSJ, and IEEE.

Shinji Kanagawa (Non-member) graduated from the Department of Electrical and Electronic Engineering, Yamaguchi University in 2009. He is currently a student at the Graduate School of Science and Engineering, Yamaguchi University. His current research interests include control system design via evolutionary computation.



Kanya Tanaka (Senior Member) graduated from the Science and Engineering program at the National Defense Academy in 1982. From 1990 through 1992, he was an associate professor at the Department of Electronic Engineering, the Hiroshima Denki Institute of Technology. From 1992 through 1998, he was an associate professor at the Department of Mechanical Engineering, Ehime University. He is currently a professor of the Graduate School of Science and Engineering at Yamaguchi University. His main interests are in the theory of intelligent control and related applications. He is a member of SICE, ISCIE, JSME, JSAEM, and IEEE.

Yuki Nishimura (Member) received his B.S., M.S., and Ph.D. degrees from Hokkaido University, Japan, in 2002, 2005, and 2009, respectively. In 2009, he joined the faculty of Yamaguchi University, where he is currently an assistant professor of the Graduate School of Science and Engineering. His research interests include nonlinear control theory. He is a member of SICE, ISCIE, and IEEE.