International Journal of Systems Science
volume 35, number 13–14, 20 October–15 November 2004, pages 775–785

Minimax control design for nonlinear systems based on genetic programming: Jung's collective unconscious approach

J. Imae*, N. Ohtsuki, Y. Kikuchi and T. Kobayashi

When it comes to the minimax controller design, it would be extremely difficult to obtain such controllers in nonlinear situations. One of the reasons is that the minimax controller should be robust against any kind of disturbances in the nonlinear situations. In this paper, we propose a new type of design method of minimax control problems. More precisely, based on the genetic programming and the collective unconscious of Jung, this paper presents a simple design technique to solve the minimax control problems, where the minimax controller may be constructed only paying attention to the minimization process. It would be surprising if the maximization process is not needed in the construction of minimax controllers. Some simulations are given to demonstrate the effectiveness of the proposed design technique with the identification problem and minimax control problems.

1. Introduction

We deal with the optimal control design for the minimax problems of the nonlinear systems (Isidori and Astolfi 1992, Van der Schaft 1992). Generally speaking, there exist two ways to attack such minimax problems. One is the so-called 'indirect approach', where the solutions of partial differential equations, such as Hamilton–Jacobi–Isaacs (HJI) equations, are used for obtaining such controllers 'indirectly', and the other is the so-called 'direct approach', where both of the minimization and maximization processes are done numerically for obtaining such controllers 'directly'.

Unfortunately, both approaches have drawbacks. In the former, it would be very difficult to solve the nonlinear partial differential equations such as HJI equations. In the latter, it would be very difficult to perform the minimization and maximization calculations in obtaining the solutions of the minimax problems. In this paper, we focus on the direct approach and, based on a GP technique, propose a new design method where no maximization calculation could be required. That is, we could omit the maximization process in deriving the controllers of the minimax problems. This idea comes from the fusion between GP and the collective unconscious of Jung (Jung 1959, Koza 1992).

Our approach is mainly based on a GP technique. What is the GP technique? In the 1990s, J. R. Koza extended J. Holland's genetic algorithm (GA) to the genetic programming (GP) in which the population consists of computer programs of varying size and shapes (Holland 1975, Koza 1992). The individual in the GP has the structure of the tree. In the evolution process, mutation, inversion and crossover are used as the genetic operators, which are called Gmutation, Ginversion and Gcrossover, respectively. The GP can generate computer programs by itself, which means the GP has the emergent property. The proposed approach is mainly based on the emergent property of the GP, and allows the computers to generate the optimal/robust controllers of nonlinear systems.

Of course, GP-based methods are already reported as one of the promising approaches in the field of nonlinear control problems (Imae and Takahashi 1999a, b, Imae *et al.* 1999, Koza *et al.* 1999). However, it should be noted that our approach is completely different from the existing GP-based methods in the minimax problems because no maximization process is required.

Received 1 January 2004. Accepted 1 August 2004.

Graduate School of Engineering, Osaka Prefecture University, 1-1, Gakuen-cho, Sakai 599-8531, Japan.

*To whom correspondence should be addressed.
e-mail: jimae@mecha.osakafu-u.ac.jp

International Journal of Systems Science ISSN 0020–7721 print/ISSN 1464–5319 online © 2004 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/00207720412331303705

The paper is organized as follows. In Section 2, the minimax control problems are formulated. In Section 3, the property of the GP and Jung's idea of the collective unconscious are described in detail. They play key roles in constructing the controllers of the minimax problems, i.e. minimax controllers. In Section 4, the new design method is proposed as a result of the fusion between the GP and the collective unconscious. We then describe the design procedure in detail, focusing on the minimax control problems. In Section 5, some numerical examples such as the identification problem and minimax control problems are given to show the effectiveness of our approach.

2. Problem formulation

In the field of control engineering, H_∞ control problems are popular in deriving robust controllers, where the H_∞ controllers are known to be the minimax controllers. In this paper, we adopt the H_∞ control problems as the minimax problems in question.

2.1. *Minimax control problem*

One of the important control design problems is as follows. Consider the dynamical system described by the ordinary differential equation with the performance equation:

$$\begin{aligned} \dot{x} &= f(x, u, w) \\ z &= h(x, u, w), \end{aligned} \tag{1}$$

where x is the state, u is the control w is the disturbance and z is the error. We here assume that the design specifications can be written with the to-be-minimized performance index J:

$$J = \max_{w} \tilde{J} \tag{2}$$

$$\tilde{J} = \frac{\int_{t_0}^{t_1} z^T z \, \mathrm{d}t}{\int_{t_0}^{t_1} w^T w \, \mathrm{d}t}, \tag{3}$$

where t_0 and t_1 are the initial and terminal time, respectively. Then, the control design problem is to find the optimal feedback controller minimizing the performance index (2).

The above problems are called 'robust control problems' because once the controllers are obtained, the resultant control systems are expected to be robust against any type of disturbances given to the systems. Therefore, the minimax problem is considered to be one of the important issues to be solved in the field of control engineering. However, these problems seem to be extremely difficult to solve, because we have to find the controller that is robust against any kinds of disturbance. This means we have to find the controller minimizing the performance index J, where the value of J is to maximize the value of $\tilde{J}$ over all kinds of disturbances. The calculations of minimization and maximization seem to be complicated. Is it possible to obtain the so called minimax controllers in the complicated situation? One of the answers could be given in this paper.

3. Preliminaries

In this section, we describe the genetic programming and Jung's idea of the collective unconscious. Here, only the flavor is given. For more details, see Jung (1959).

3.1. *Genetic programming*

A genetic algorithm (GA) has been extended to the so-called genetic programming (GP), where the population consists of computer programs of varying sizes and shapes. The individual in the GP has the tree structure. See figure 1, for example. The tree structure in figure 1(a) is equivalent to the symbolic expression in figure 1(b), which represents the function of '$(X1 + X2 - 2.5)\ X1 - 0.4\sin(T)$'. For more details, see Koza (1992).

In the evolutionary process of the GP, the genetic operators, such as Gcrossover, Gmutation, and Gcopy, are used. See figure 2, for example. Gmutation consists of four types of mutation process, such as terminal-to-function type, function-to-terminal type, terminal-to-terminal type and function-to-function type. In figure 2, F_i $(i = 1, 2, \ldots)$ are called 'functions', and T_i $(i = 1, 2, \ldots)$ are called 'terminals'.

Here is the procedure of the GP algorithm, which consists of an iterative process.

step 0. Select GP parameters.

step 1. Generate the initial population.

step 2. Evaluate the fitness of individuals.

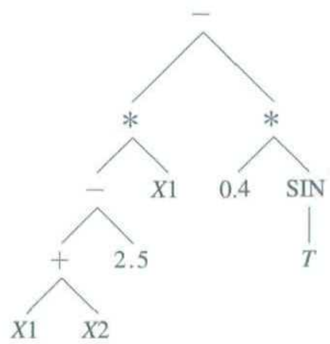
step 3. Execute the criterion on the convergence.

step 4. Based on the fitness, perform the operations of Gcrossover, Gmutation, and Gcopy.

step 5. Go back to step 2.

3.2. *Collective unconscious*

When it comes to the minimax problems, both of the minimization and maximization processes are usually needed to obtain the minimax controllers. From a computational point of view, one of the easy ways to obtain the minimax solution is to carry out the minimization and maximization calculations



(a)

```
(-
  (*
    (-
      (+
        X1
        X2)
      2.5)
    X1)
  (*
    0.4
    (SIN
      T)))
```

(b)

Figure 1. Individual in GP for '$(X1+X2-2.5)X1-0.4\sin(T)$': (a) tree structure; and (b) symbolic expression.

alternately, which is described in section 4. In this paper, we propose a relatively simpler way to tackle such minimax problems, based on the Jung's idea of the collective unconscious. Due to the Jung's idea, the proposed approach only depends on the minimization process, which does not need the maximization process. We describe Jung's idea of the collective unconscious as follows.

Carl Jung was a student of Sigmund Freud, and he accepted the existence of a conscious and unconscious mind. However, Jung gave a unique interpretation of the unconscious. He insisted that the unconscious consists of two parts, the personal unconscious and the collective unconscious. Although the personal unconscious is almost the same as Freud's and depends mainly on environments, the collective unconscious is beyond Freud's. It does not depend on environments, and it would rather depend on the evolutionary process of human beings. That is to say, the offspring would unconsciously share the experiences the ancestor had. It would be important to note here that the collective unconscious exists within the evolutionary process. Anyway, if we pay attention to this idea, we could omit the maximization process in the minimax approach of the design problems. Because if we regard disturbances as materials of the collective unconscious, we could deal with any disturbance 'unconsciously' in the evolutionary process. This leads to the idea that we need not pay attention to all disturbances in finding the minimizing controller in the evolutionary process, where the controllers evolve. That is to say, in such an evolutionary process, the offspring would unconsciously share all the disturbances the ancestor had.

4. Design method

The minimax control problems seem to be extremely difficult to solve because we have to find the controller that is robust against any kinds of disturbances. Generally speaking, both of the minimization and maximization operations are usually needed to obtain the minimax controllers. One of the computationally easy ways to obtain the minimax solution is to perform the minimization and maximization processes 'alternately'. We here revisit such a conventional procedure of the GP-based design method for the minimax problems. For more details, see Imae *et al.* (1999).

[Conventional GP design method]

step 1. Select the GP parameters.

step 2. Generate randomly two initial populations: One for the control population, and the other for the disturbance population.

step 3 (Evaluation of the fitness at the initial generation).

step 3-1 (Control population). Evaluate the fitness $\tilde{J}$ of individuals of the control population, with the disturbance being zero ($w=0$). Then, select the best one of all individuals based on the fitness, according to the smaller-is-better philosophy.

step 3-2 (Disturbance population). Evaluate the fitness $\tilde{J}$ of the individuals of the disturbance population, with the best one selected in step 3-1. Then, select the best one of all individuals based on the fitness, according to the larger-is-better philosophy.

step 4. Execute the convergence criterion.

step 5. Proceed with the Gcrossover, Gmutation, and Gcopy. Create the new control/disturbance generations.

step 6 (Evaluation of the fitness at the generation, k).

step 6-1 (Control population). Evaluate the fitness $\tilde{J}$ of individuals of the control population, with the best individual of the disturbance population at the $k-1$ generation. Then, select the best one of all individuals based on the fitness.

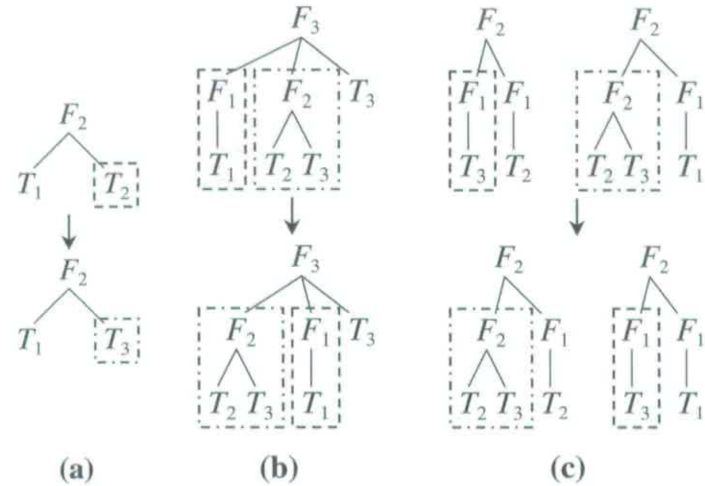


Figure 2. GP operators: (a) Gmutation; (b) Ginversion and (c) Gcrossover.

step 6-2 (Disturbance population). Evaluate the fitness $\tilde{J}$ of the individuals of the disturbance population, with the best one selected in step 6-1. Then, select the best one of all individuals based on the fitness.

step 7. Execute the convergence criterion.

step 8. Proceed with the Gcrossover, Gmutation, and Gcopy. Create the new control/disturbance generations.

step 9. Go to step 6.

Although the conventional GP design method seems to be the computationally easy technique for obtaining the numerical minimax solutions, both the minimization and maximization processes are necessary. In this section, we propose a new approach to the GP design method of the minimax problems, where the maximization process is not necessary explicitly, based on Jung's idea of the collective unconscious.

Let us go back to the section of the problem formulation. What we have to do is to find the robust controller against any kinds of disturbances, that is, to find the controller minimizing J, where the value of J is given by maximizing the value of $\tilde{J}$ over all kinds of disturbances. Now recall the collective unconscious. It tells us that if we regard disturbances as materials of the collective unconscious, we could deal with any disturbance unconsciously in minimax problems. Therefore, there would be no need to deal with all the disturbances 'consciously', i.e. explicitly. A few disturbances might be enough in deriving the minimax controller, because other disturbances are expected to be experienced implicitly in the evolutionary process. It should be noted that if we take only one disturbance in the evaluation of the fitness, we can omit the maximization process in minimax problems. This gives birth to a new type of GP design method based on the collective unconscious, as follows.

At first, we determine the terminals, functions and fitness:

(1) Terminals: the input of the controller is the state of the plant system. So, we use the state variable x as the terminal. The terminal set also contains real value $R(-5.0 \leq R \leq 5.0)$.
(2) Functions: this set is the non-terminal set. A lot of functions are proposed so far. For the purpose of simplicity in control design, we focus on the basic arithmetic operators such as '+', '−', '*' and '/'.
(3) Fitness: the fitness function is determined minimizing the performance index (3). It should be noted that no maximization operation is required.

Now, we are in a position to propose the following procedure of the new GP-based design method including no maximization process, which is much simpler than the conventional one.

[Proposed GP design method]

step 0. Select GP parameters.

step 1. Generate randomly an initial population, consisting of functions and terminals, with the size N of the population.

step 2. Evaluate the fitness of the individual, calculating the performance index (3) subject to the dynamical equation (1). Then, due to Jung's idea of the collective unconscious, only one arbitrary disturbance is adopted. One of the schemes to generate such a disturbance is given in Remark 3.

step 3. Based on the fitness, select the best individual of the population (say, best-of-generation individual).

step 4. Preserve the best individual over all the best-of-generation individuals (say, best-so-far individual). Here, we adopt another criterion to choose the best-so-far individual. For more details, see Remark 4.

step 5. Perform genetic operations, such as Gcrossover, Gmutation, and Gcopy. And create a new population.

step 6. Go back to step 2.

Remark 1: It should be noted that we adopt the performance index (3) in the evaluation of the individual, not the performance index (2). That is to say, no maximization process is given in step 2. Because, due to Jung's idea, the offspring could experience all disturbances the ancestor experienced, it might not be necessary for each individual to experience more than one kind of the disturbance.

Remark 2: GP parameters in step 0 are as follows. The deepness of the tree structure is 6 in the first generation and 4 in Gmutation. The rate of Gcrossover is 0.1 in functions, and 0.7 in terminals, respectively. The rate of Gmutation or Gcopy is 0.1. The size of the population is 256. Besides, 'grow' method in generating an initial random population and 'tournament' selection method are adopted.

Remark 3: We generate two populations; one is for the minimax controller and the other is for the disturbance. However, it should be noted that the population for the disturbance does not evolve by itself, and therefore does not make any effect on the evolutionary process in the design method. The individual of the disturbance population is used only in the evaluation of the fitness of step 2.

Remark 4: In choosing the best-so-far individual of step 4, we again evaluate only the best-of-generation individual by applying the 256 kinds of the disturbances to that individual, and compare it with the best-so-far individual of the one-generation-before's.

5. Simulations

Our design method based on the collective unconscious is heuristic. There is no theoretical proof given. In this section, we give some numerical simulations to demonstrate the effectiveness of our design approach. We tackle five identification/control problems, where the identification problem is given in example 5.1 and minimax control problems are given in examples 5.2–5.5. In examples 5.1 and 5.2, we try to show that our unique idea is implementable, and in examples 5.3–5.5, we demonstrate how well our design method works in a comparison with the conventional one.

5.1. *Identification problem*

Consider the following function to be identified:

$$y = 3x^2 + 4x + 6. \tag{4}$$

Generally, the identification problem falls in the category of the minimization problem. However, to check out whether or not our new approach is implementable, this identification problem is converted into the minimax one, as follows: find a function $\hat{y}$ with the index:

$$\min_{\hat{y}} \max_{x} J_1, \tag{5}$$

where $J_1 = (\hat{y}/y - 1)^2$. In this section, we focus on the minimax formulation (5). Note that it could be formulated as a minimization problem, as follows: find a function $\hat{y}$ with the index:

$$\min_{\hat{y}} \int (\hat{y}/y - 1)^2 \, \mathrm{d}t. \tag{6}$$

The proposed method is applied with 100 generations. After 13 generations, we obtained the best-so-far individual. In simulations the range of the variable x is given with $0.2 \le x \le 20.0$. The variable x plays a key role in the identification situation, as the disturbance. Terminals, functions, fitness, and GP parameters are the same as in Section 4, except that one of the arithmetic operations '/' is omitted. The GP solution is:

$$\hat{y} = 2.9122x^2 + 5.587x + 2.2009. \tag{7}$$

See figure 3 for the optimal and GP solutions. Both functions are extremely close. The computational results tell us that Jung's unconscious approach might be implementable in a kind of minimax problem. It is important to note that due to Jung's idea, there is no need to apply all kinds of disturbances to the indivi-

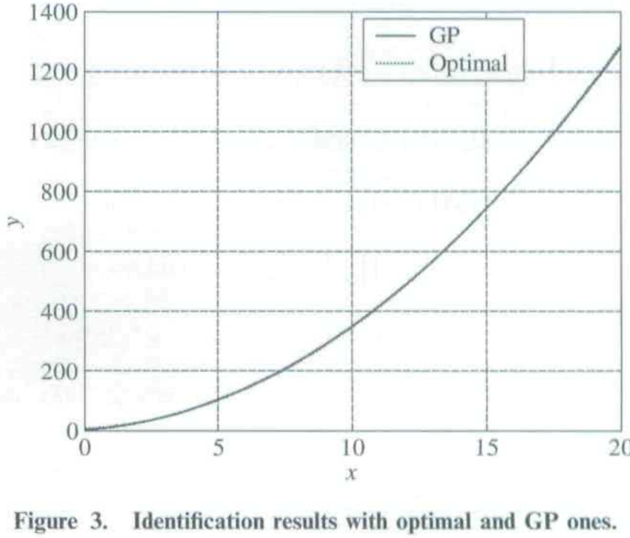


Figure 3. Identification results with optimal and GP ones.

dual, that is, one kind of disturbance is sufficient in obtaining the GP solution of the minimax problems.

5.2. *Minimax control problem I (Linear one-dimensional system)*

Consider the H_∞ control problem with the following equations:

$$\dot{x} = -x + u + w \quad z = [x \quad u]^{\mathrm{T}}. \tag{8}$$

The performance index is given as follows.

$$\tilde{J} = \frac{\int_O^\infty z^T z \, \mathrm{d}t}{\int_O^\infty w^T w \, \mathrm{d}t}. \tag{9}$$

That is

$$\tilde{J} = \frac{\int_O^\infty (x^2 + u^2) \, \mathrm{d}t}{\int_O^\infty w^2 \, \mathrm{d}t}. \tag{10}$$

Through this simple example, we demonstrate our new approach is also implementable in the H_∞ control problems. Because the optimal solution is analytically given as:

$$u_{\mathrm{op}} = -x, \tag{11}$$

it is possible to compare the result of our method with the optimal one.

The problem has been tackled with 200 iterations. The range of the disturbance is set to $|w| \le 1$ through trial-and-error. After 147 generations, we obtained the best-so-far individual. The minimization value of the GP solution is 0.466048 with respect to the performance index $\tilde{J}$, which is close to the optimal value 0.5. The resultant GP controller is derived as follows:

$$\begin{aligned} u_{\mathrm{GP}} = \Big\{ & x(0.023944 + 0.011391x + 0.9475x^2 + 1.2971x^3 \\ & + 2.4468x^4 + 1.7909\,x^5 - 0.99182x^6 - 0.95245x^7 \\ & + 0.15544x^8 + 0.15037x^9 - 0.0026144x^{10} \\ & - 0.004175x^{11})\Big\} \Big/ \Big\{(0.22063 + 0.04398x) \\ & \times (-1.5015 - 1.5983x - 4.1924x^2 - 2.2558x^3 \\ & + 0.54897x^4)(0.072277 - 0.05696x + 2.7248x^2 \\ & + 0.50026x^3 - 1.3195x^4 - 0.04797x^5 + 0.1722x^6)\Big\} \end{aligned} \tag{12}$$

The structure is so complicated, and seems to be far from the optimal one. However, see figure 4 for comparison with the optimal solution. It would be surprising that the GP solution is extremely close to the optimal one, which means our design approach is applicable to a kind of the simple H_∞ control problems.

Remark 5: We cannot adopt the infinite control-time interval in simulations, and so all we have to do is to adopt the finite one, which is expected to be sufficiently large.

5.3. *Minimax control problem II (Linear two-dimensional system)*

In the next three examples, we demonstrate how well our design method works in comparison with the conventional one. First, consider the following two-dimensional dynamical system, which is one of the linear H_∞ control problems:

$$\dot{x} = \begin{bmatrix} -1 & 0 \\ -1 & -1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \quad z = \begin{bmatrix} 0 & 0 \\ 0 & -1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u. \tag{13}$$

In this case, the performance index is given as follows:

$$\tilde{J} = \frac{\int_0^\infty (x_2^2 + u^2) \mathrm{d}t}{\int_0^\infty w^2 \, \mathrm{d}t}. \tag{14}$$

An optimal solution is analytically given in a simple fashion.

$$u_{\mathrm{op}} = x_1 - x_2. \tag{15}$$

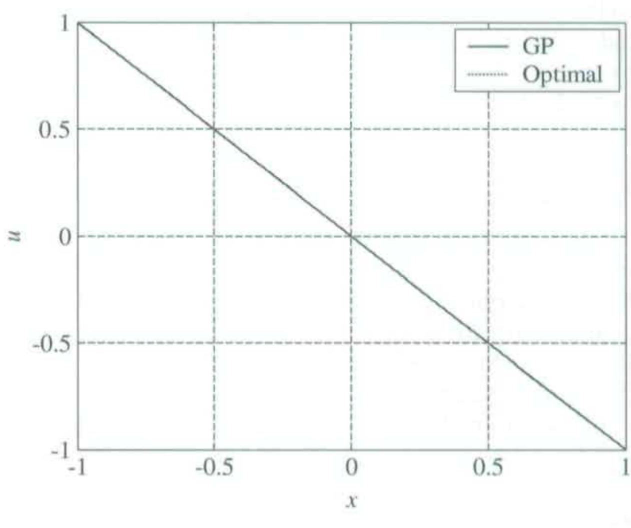


Figure 4. Optimal and GP solutions for example 5.2.

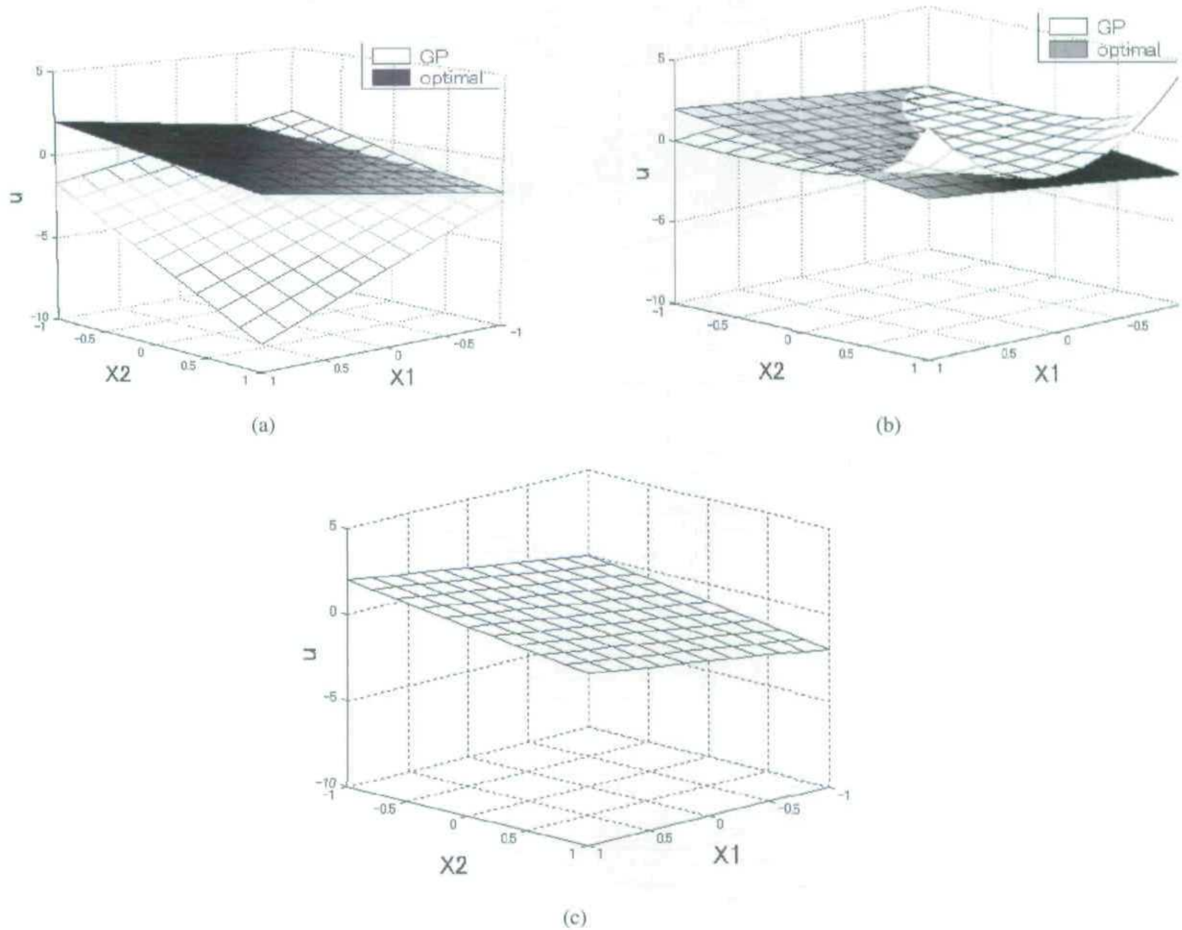


Figure 5. Optimal and GP solutions for example 5.3: (a) generation 1; (b) generation 3; and (c) generation 5.

In simulations, the control interval is set to [0, 10] and the range of the disturbance is constrained to $|w| \leq 5.0$ through trial and error. Note that the GP design methods are a bit sensitive to the range of the disturbance in the conventional and proposed approaches. We have performed the calculations with 100 generations. Arithmetic operations are restricted to '+', '−' and '*' because the operation '/' did not work well. After five generations with initial states $x_1(0) = x_2(0) = 0$, we have obtained the GP solution, where the CPU time is 11.208 s. When it comes to the CPU time, our approach is far superior to the conventional approach, because in the conventional approach the solution is obtained after 16 generations spending 100.718 s. The resultant controller is as follows:

$$u_{GP}(x) = x_1 - x_2. \tag{16}$$

This is exactly the same as the optimal one. Figure 5 shows the evolutionary process of GP controllers, where the controllers of the first, third and fifth generations are given.

Remark 6: In the conventional GP method, both the minimization and maximization operations are carried out alternately. That is why the proposed one is superior to the conventional one in the CPU time, even though the number of generations in the former is smaller than that in the latter.

5.4. *Minimax control problem III (Non-linear one-dimensional system)*

We attack the following nonlinear system:

$$\begin{aligned} \dot{x} &= -x\sqrt{2x^4 + 4x^2 + 1} + (1 + x^2)w + u \\ z &= \begin{bmatrix} 1 \\ 0 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u. \end{aligned} \tag{17}$$

Generally speaking, for nonlinear problems, it is difficult to obtain analytically the optimal solutions. However,

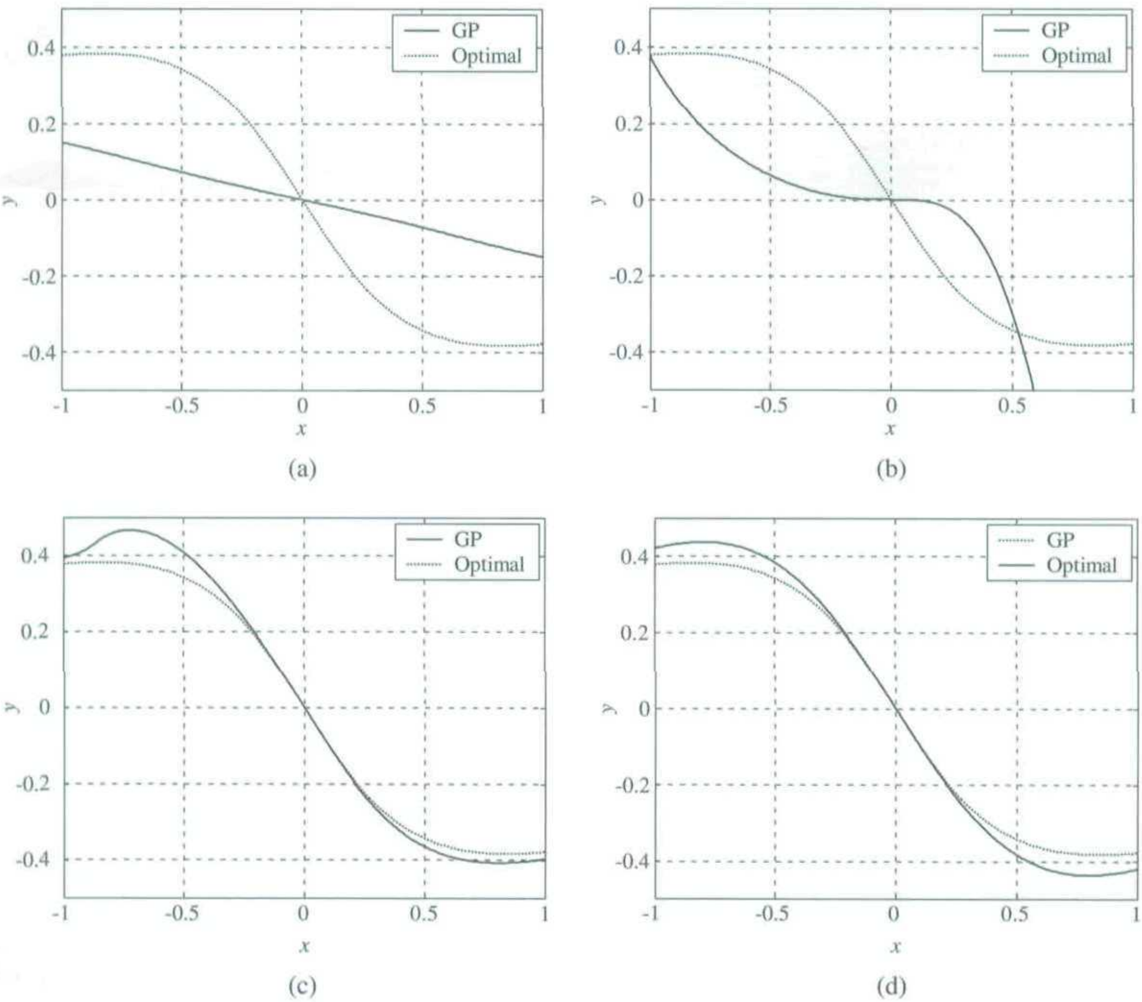


Figure 6. Optimal and GP solutions for example 5.4: (a) generation 2; (b) generation 10; (c) generation 15; and (d) generation 21.

in this problem, the optimal solution is given in the reference (Van der Schaft 1992). The optimal solution is:

$$u_{\mathrm{op}} = -\frac{x}{\sqrt{2x^4 + 4x^2 + 1}}. \tag{18}$$

The proposed method is applied to this nonlinear problem with 100 generations. The range of the disturbance is fixed to $|w| \leq 5.0$ through trial and error. For the function set, we adopted 'sin', 'cos', 'atan' and 'exp' in addition to the basic arithmetic operations such as '+', '−' , '*' and '/', because we could not obtain the numerically satisfactory solutions with such basic arithmetic operations. After 21 generations, we have obtained the best-so-far individual. The obtained GP solution is as follow:

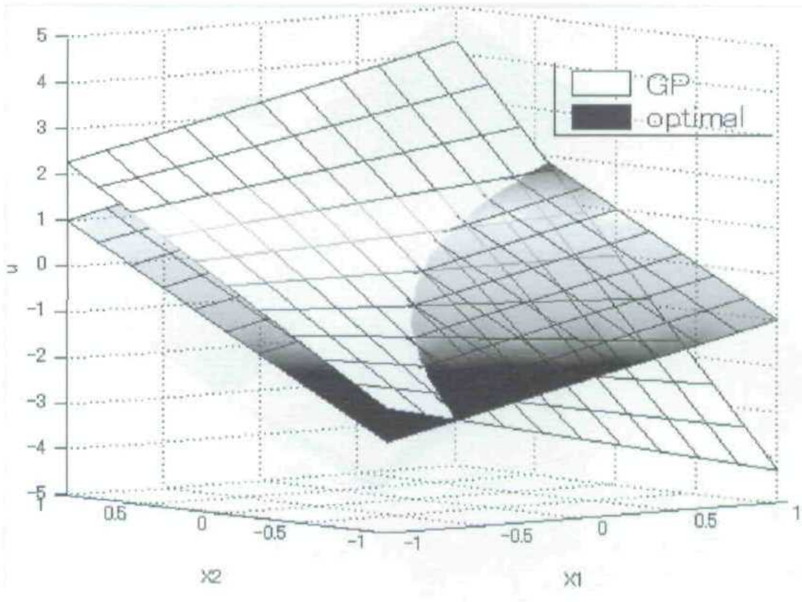
$$u(x) = \frac{\sin x}{x^2 - \exp(-19.5865 - 3.269x) + 0.9983}. \tag{19}$$

See figure 6(d). The GP solution is close to the optimal one. Speaking of CPU time, our GP method used 184.418 s while the conventional approach used 328.477 s. Figure 6 demonstrates how well the proposed method works even with nonlinear problems, where the results of the second, 10th, 15th and 21st generation are given. Note that in the conventional one, the solution is obtained after 31 generations.

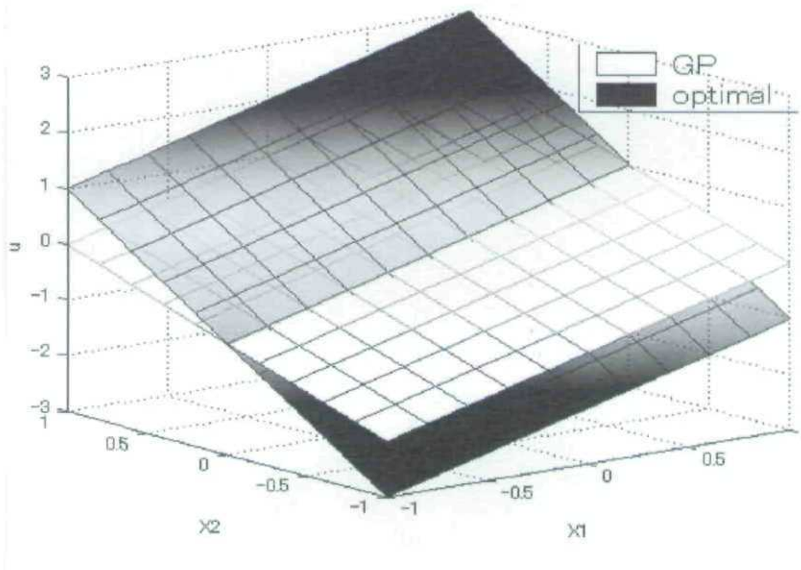
5.5 *Minimax control problem IV (Non-linear two-dimensional system)*

We consider the following nonlinear system:

$$\begin{aligned} \dot{x} &= \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w + \begin{bmatrix} x_2 & -1 \\ -x_1 & -2 \end{bmatrix} u \\ z &= [x_1 \quad x_2 \quad u]^{\mathrm{T}} \\ J &= \frac{\int_0^\infty (x_1^2 + x_2^2 + u^2)\,\mathrm{d}t}{\int_0^\infty w^2\,\mathrm{d}t}. \end{aligned} \tag{20}$$



(a)



(b)

Figure 7. Optimal and GP solutions for example 5.5: (a) generation 8; (b) generation 20; (c) generation 40; and (d) generation 56.

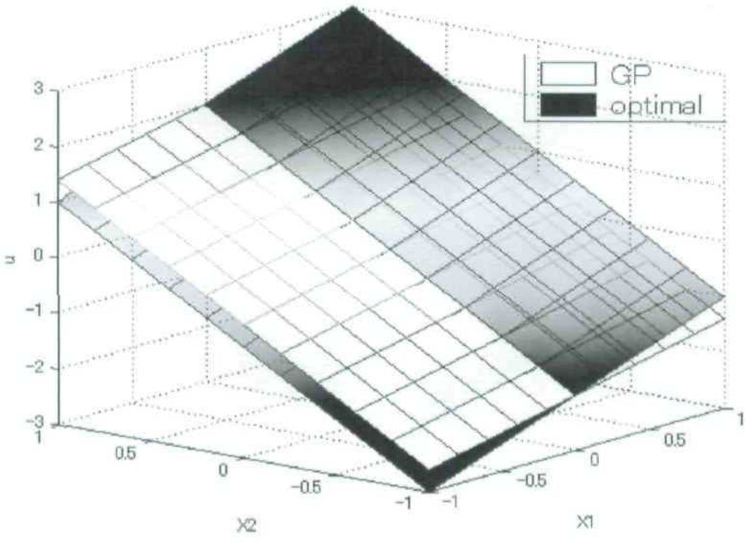
Let us show the analytical/optimal solution:

$$u_{\mathrm{op}} = x_1 + 2x_2. \tag{21}$$

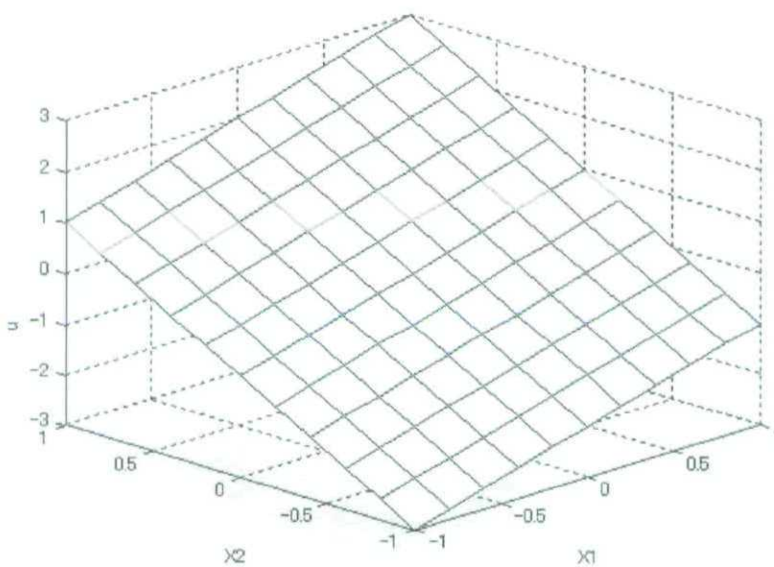
Computational results are as follows. The control interval [0, 10] is chosen for this problem. The computational conditions are the same as in example 5.3, except that arithmetic operation '/' is included. The proposed method is applied to this nonlinear problem with 100 generations. After 56 generations, we obtained the best-so-far individual. The GP solution is obtained as follows:

$$u_{\mathrm{GP}} = x_1 + 2x_2. \tag{22}$$

This is exactly the same as the optimal one. See figure 7, where the results of the eigth, 20th, 40th and 56th generation are given. The evolutionary process is found in this figure. Moreover, the CPU time of our GP method



(c)



(d)

Figure 7. Continued.

is 600.705 s while that of the conventional approach is 1095.729 s. In this example, our approach is also superior to the conventional GP approach. Also note that the solution of our GP method is obtained after 56 generations while that of the conventional one is after 65 generations.

6. Conclusions

In the field of the minimax control problems, the new type of GP-based design method is proposed, where the maximization process is not required in the minimax calculation process. It is due to Jung's idea of the collective unconscious.

Our design method based on the collective unconscious is heuristic, and there is no theoretical proof given. We have given some numerical simulations to demonstrate the effectiveness of our design approach. Six simulations are given to demonstrate the efficiency of the proposed GP-based design method. We have tackled five identification/control problems, where the identification problem is given in example 5.1 and the H_∞ control problems are given in examples 5.2–5.5. In examples 5.1 and 5.2, we have tried to show that our unique idea is implementable, and in examples 5.3–5.5, we have demonstrated how well our design method works in a comparison with the conventional one. With respect to the CPU time, our method has turned out dramatically superior to the conventional GP method.

References

HOLLAND, J. H., 1975, *Adaptation in Natural and Artificial Systems* (Ann Arbor: University of Michigan Press).

IMAE, J., NAKATANI, S., and TAKAHASHI, J., 1999, GP based flight control in the windshear. In *Proceedings of the 1999 IEEE International Conference on Systems, Man, and Cybernetics*, Part II, pp. 650–653.

IMAE, J., and TAKAHASHI, J., 1999a, GP based design method for control systems via Hamilton–Jacobi–Bellman equations. In *Proceedings of 1999 ACC*, San Diego, USA, pp. 3001–3002.

IMAE, J., and TAKAHASHI, J., 1999b, A design method for nonlinear H-∞ control systems via Hamilton–Jacobi–Isaacs equations: A genetic approach. In *Proceedings of the 38th CDC*, Phoenix, USA, pp. 3782–3783.

ISIDORI, A., and ASTOLFI, A., 1992, Disturbance attenuation and H_∞ control via measurment feedback in nonlinear systems. *IEEE Transactions on Automatic Control*, **37**, 1283–1293.

JUNG, C. G., 1959, *The Archetypes and the Collective Unconscious*, trans. R. F. C. Hull (London: Routledge & Kegan Paul).

KOZA, J. R., 1992, *Genetic Programming* (Cambridge, MA: MIT Press).

KOZA, J. R., KEANE, M. A., YU, J., and Bennett III, F. H., 1999, Automatic synthesis of both the topology and parameters for a robust controller for a non-minimal phase plant and three-lag plant by means of genetic programming. In *Proceedings of the 38th CDC*, Phoenix, USA, pp. 5292–5300.

VAN DER SCHAFT, A. J., 1992, L2-gain analysis of nonlinear systems and nonlinear state feedback H_∞ control. *IEEE Transactions on Automatic Control*, **37**, 770–784.

VAN DER SCHAFT, A. J., 1994, Nonlinear state space H_∞ control theory. Workshop (No.7) of the 33rd CDC, Orlando, USA.