# Genetic Algorithm with Redundant Chromosome and its Application to Control Systems Design

**Ken-ichirou Komatsu, Tadashi Ishihara, and Hikaru Inooka**

**Graduate School of Information Sciences**
**Tohoku University**
**Sendai, 980-8579, JAPAN**

komatsu@control.is.tohoku.ac.jp, ishihara@control.is.tohoku.ac.jp, and inooka@control.is.tohoku.ac.jp

## Abstract

In solving engineering problem by an optimization technique, designers are required to choose an appropriate structural model since conventional optimization techniques, including standard genetic algorithms, do not possess structural model selection ability. To achieve more efficient design, we propose a new genetic algorithm with a model structure selection ability. The proposed algorithm utilizes individuals that have redundant chromosomes. The ability of the proposed algorithm is illustrated by using a simple multiobjective problem. An application to control systems design is discussed. It is shown that the proposed algorithm can effectively be used for a design based on the method of inequalities and the principle of matching

## 1 Introduction

In the process of solving engineering problems, designers are often required to choose an appropriate structural model of solutions before perfoming parametric optimizaion. Conventional optimization techniques including standard genetic algorithms seek an optimal solution for a fixed structural model specified by the designer. For example, in the optimization-based control systems design, a designer is usually required to specify the structure of controller. Then numerical search procedure is performed to find the best parameter values for the specified controller structure. If the parameter search fails, the designer must specify another structural model. This process has to be repeated until satisfactory solution is obtained. If we can develop an optimization algorithm with structural model selection ability, the burden for the designer can greatly be reduced.

As an optimization procedure with structural model selection ability, we propose a new genetic algorithm in which individuals have redundant chromosomes. In the new algorithm, each phenotype is constructed with some operators, which are encoded into genes. Genes are composed of redundant chromosomes, but an individual has sufficiently redundant genes to construct its phenotype.

Since the combination of operators or genes is adjustable and its types are rich, the individuals have potentialities for various phenotypes. The richness of structural models is supported by the various phenotypes. On the other hand, the richness introduces the difficulty in controlling the combination of genes.We introduce a selection scheme in which each individual has the innerstate which is acquired through its history and generates the essential information.the combination of genes and that of a phenotype of individual deffectively determined.

This paper is organized as follows. In Section2, we propose the new genetic algorithm and explain the methods for handling redundant chromosomes. Using a simple optimization problem, we discuss the effectiveness of the proposed algorithm in Section 3. In Section 4, we apply the proposed algorithm to control systems design. Conclusions are given in Section 5.

## 2 New Genetic Algorithm

In engineering problems, the search utilizing standard genetic algorithm is usually performed on parameters of one specified structural model. For example, a structural model $f$ is defined by a parameter vector $(p_1, p_2, \cdots, p_m)$ and represented as

$$f(p_1, p_2, \cdots, p_m). \tag{1}$$

The notation $f(p_1, p_2, \cdots, p_m)$ is used to represent a model of solutions in the search. If the search fails, the designer must specify another structural model and restart, the parameter search. This process has to be repeated until satisfactory solution is obtained.

To reduce the designer's interruption for the design process, we propose a new genetic algorithm with the model selection ability. The proposed algorithm utilizes individual that has the chromosomes composed with surplus genes and the group of individuals has various phenotypes.

Suppose that a structural model is constructed as

$$f_{\mathbf{l}} = f_{l_m} f_{l_{m-1}} \cdots f_{l_2} f_{l_1} \tag{2}$$

where $f_{l_i}$ is the operator, $\mathbf{l} = \{l_1, l_2, \cdots, l_m\}$ is the array of indexes of genes, and $m$ is the number of genes which are selected from chromosomes. The operator $f_{l_i}$ is described by the genes $g_i$, $(i = 1, 2, \cdots, n)$. $n$ is the number of genes which compose chromosomes of individual. Such genes compose individual's chromosomes $\mathbf{g} = \{g_1, g_2, \cdots, g_n\}$.

0-7803-5731-0/99$10.00 ©1999 IEEE

Note that a phenotype of individual is decided by the combination of the genes. The evaluations for each individuals depends on indexes of selected genes $\mathbf{l}$. Therefore the evaluations is given as

$$\phi(\mathbf{l}) = \{\phi_1(\mathbf{l}), \phi_2(\mathbf{l}), \cdots, \phi_n(\mathbf{l})\}. \quad (3)$$

In the search of solutions, we minimize the objective functions.

The surplus of genes over operators produces the redundancy of chromosomes. Due to the surplus, individuals can change their phenotype but this introduces difficulty in fixing their phenotypes.

To reduce the redundancies of individuals, we introduce inner states for individuals. The inner states of individual are represented five variables $\mathbf{c}_{\mathbf{promote}}$, $\mathbf{c}_{\mathbf{repress}}$, $\mathbf{a}$, $\mathbf{A}$, and $\Phi$. $\mathbf{c}_{\mathbf{promote}}$ and $\mathbf{c}_{\mathbf{repress}}$ are effective in controlling selection of genes. $c_{\mathrm{p}i}$ promotes the selection of $i$th gene. $c_{\mathrm{r}i}$ represses the selection of $i$th gene. $\mathbf{a}$ and $\mathbf{A}$ are memories about selections of genes. $\mathbf{a}$ stores the state of genes of individual. $\mathbf{A}$ memorize the history of $\mathbf{a}$. $\Phi$ memorize the history of evaluation $\phi$

Through trial and error, individuals get the information about the causal relationship between the combination of the genes and the evaluations of phenotype. Such information is useful in controlling the combination of genes. The information modifies inner states. Each individul utilizes its information to the next trial and searches for gene combinations which are superior to the previous trials. Through a result of trial and error, useful combinations of the gene strengthen $\mathbf{c}_{\mathbf{promote}}$ and useless combinations of the gene strangthen $\mathbf{c}_{\mathbf{present}}$.

In addition, inner states are useful when new children are reproduced. By using these memories, we can segregate useful genes and useless ones. In the alternation of generations, we can preserve useful genes in chromosomes of children and eliminate useless ones from chromosomes of children.

We explain our genetic algorithm more concretely. First, we explain the sequence of operations in the hierarchy of group level.

**Step 1:** Create individuals and complete the prescribed population.

**Step 2:** Fix individuals their own phenotypes.

**Step 3:** Evaluate phenotypes of individual and give evaluations to each individual.

**Step 4:** Check evaluations of individual. If some evaluations satisfy the requirements to terminate search then terminate operations. If any evaluations does not satisfy the requirements then go to the next operation.

**Step 5:** Make individuals memorize the causal relationship between their own phenotype and its effect.

**Step 6:** Compare individuals' most desirable evaluations each other, and select individual which is more desirable evaluation than the others.

**Step 7:** Reproduce individuals, and complete the prescribed population to reorganize the group.

**Step 8:** Go to **Step 2**.

Next, we explain the sequence of operations in the hierarchy of individual level and that of inner processes level.

**Step 1:** Create individual.

- **1.1:** Generate $n$ symbols at rundom, and construct $n$ genes $g_i$, $(i = 1, 2, \cdots, n)$ with them.
- **1.2:** Construct chromosomes $\mathbf{g} = \{g_1, g_2, \cdots, g_n\}$ with generated genes.
- **1.3:** Generate $n$ sysmbols at rundom, and construct $n$ gene promoting facrors $c_{\mathbf{promote}\,i}$, $(i = 1, 2, \cdots, n)$ with them.
- **1.4:** Construct $\mathbf{c}_{\mathbf{promote}} = \{c_{\mathrm{p}\,1}, c_{\mathrm{p}\,2}, \cdots, c_{\mathrm{p}\,n}\}$ with generated gene promoting facrors.
- **1.5:** Generate $n$ sysmbols at rundom, and construct $n$ gene repressing facrors $c_{\mathbf{repress}\,i}$, $(i = 1, 2, \cdots, n)$ with them.
- **1.6:** Construct $\mathbf{c}_{\mathbf{repress}} = \{c_{\mathrm{r}\,1}, c_{\mathrm{r}\,2}, \cdots, c_{\mathrm{r}\,n}\}$ with generated gene repressing facrors.

**Step 2:** Select genes and construct phenotype.

- **2.1:** Decide $\mathbf{a}$ with $\mathbf{c}_{\mathbf{promote}}$ and $\mathbf{c}_{\mathbf{repress}}$. If the effects of $c_{\mathrm{p}\,i}$ are superior to that of $c_{\mathrm{r}\,i}$ then $\mathbf{a}$ have a tendecy to promote appearing operator $f_i$ in the phenotype. If the effects of $c_{\mathrm{p}\,i}$ are inferior to that of $c_{\mathrm{r}\,i}$ then $\mathbf{a}$ have a tendecy to repress appearing operator $f_i$ in the phenotype.
- **2.2:** Decide $\mathbf{l}$ relying on $\mathbf{a}$.
- **2.3:** Select $\mathbf{g}_{\mathbf{l}}$ indicated by $\mathbf{l}$. If $\mathbf{l}$ includes a gene index $i$ then $g_i$ is selected and put in $\mathbf{g}_{\mathbf{l}}$.
- **2.4:** Decide phenotype $f_{\mathbf{l}}$. Decode operator $f_{l_i}$ from $g_{l_i}$ and construct composited function $f_{\mathbf{l}} = f_{l_m} f_{l_{m-1}} \cdots f_{l_1}$ with the operators.
- **2.5:** Decide $\mathbf{a} = \{a_1, a_2, \cdots, a_n\}$ relying $\mathbf{l} = \{l_1, l_2, \cdots, l_m\}$ where $m$ is the number of selected genes and $m \leq n$. If $\mathbf{l}$ includes a gene index $i$ then $a_i = 1$. If $a_i$ is not indexed by $\mathbf{l}$ then $a_i = 0$.

**Step 3:** Evaluate phenotype.

- **3.1:** Evaluate phenotype $f_{\mathbf{l}}$ with objective functions, and regard the evaluation as the gene combination's evaluation $\phi(\mathbf{l})$.

**Step 4:** Check individual.

**4.1:** Check objective function $\phi(\mathbf{l})$ and the other conditions. If $\phi(\mathbf{l})$ satisfies the requirements to terminate search then terminate operations. If $\phi(\mathbf{l})$ does not satisfy the requirements then continue next operation.

**Step 5:** Modify inner state.

**5.1:** Record $\phi(\mathbf{l})$ on $\Phi$.

**5.2:** Record $\mathbf{a}$ on $\mathbf{A}$.

**5.3:** Refer $\mathbf{A}$ to $\Phi$. If a gene $g_i$ contributes to desirable evaluations then its $c_{\mathrm{p}\,i}$ is emphasized. If a gene $g_i$ obstructes to desirable evaluations then its $c_{\mathrm{r}\,i}$ is emphasized.

**Step 6:** Reproduct individual.

**6.1:** Check several conditions of individual. If the condition is desirable then reproduct new individuals. If the condition is undesirable then continue the next operation.

**Step7:** Go to **Step 1**.

In reproducting individual, operations are done as the following sequence. Suppose that the child originates in two individuals. The one individual is described parent X and another is described parent Y.

**Step 2:** Decide the gene indexes $\mathbf{l}_X$ of parent X and $\mathbf{l}_Y$ of parent Y, the decision is based on $\mathbf{c}_{\mathbf{promote}}$ of parent X, $\mathbf{c}_{\mathbf{repress}}$ of parent X, $\mathbf{c}_{\mathbf{promote}}$ of parent Y, and $\mathbf{c}_{\mathbf{repress}}$ of parent Y.

**Step 3:** Copy $\mathbf{g}$ down from $\mathbf{g}$ of parent X and parent Y. The copied $\mathbf{g}$ is corresponding to indexes $\mathbf{l}_X$ and $\mathbf{l}_Y$.

**Step 4:** Reconstruct chromosomes with the copied $\mathbf{g}$.

**Step 5:** Copy $\mathbf{c}_{\mathbf{promote}}$ down from $\mathbf{c}_{\mathbf{promote}}$ of parent X and parent Y. The copied $\mathbf{c}_{\mathbf{promote}}$ is corresponding to indexes $\mathbf{l}_X$ and $\mathbf{l}_Y$.

**Step 6:** Copy $\mathbf{c}_{\mathbf{repress}}$ down from $\mathbf{c}_{\mathbf{repress}}$ of parent X and parent Y. The copied $\mathbf{c}_{\mathbf{repress}}$ is corresponding to indexes $\mathbf{l}_X$ and $\mathbf{l}_Y$.

**Step 7:** Make genes to mutate. If the effects of $c_{\mathrm{p}\,i}$ are superior to that of $c_{\mathrm{r}\,i}$ then $g_i$ is given a low probability ob mutation. If the effects of $c_{\mathrm{p}\,i}$ are inferior to that of $c_{\mathrm{r}\,i}$ then $g_i$ is given a high probagbility of mutation.

## 3 Illustrative Example

We present a simple example for our genetic algorithm. Consider the point $\mathbf{p}_{s_o}$ described as

$$\mathbf{p}_{s_o}(\theta,\varphi) = \begin{pmatrix} 5\cos\theta\sin\varphi \\ 5\sin\theta\cos\varphi \\ 5\cos\varphi \end{pmatrix} \tag{4}$$

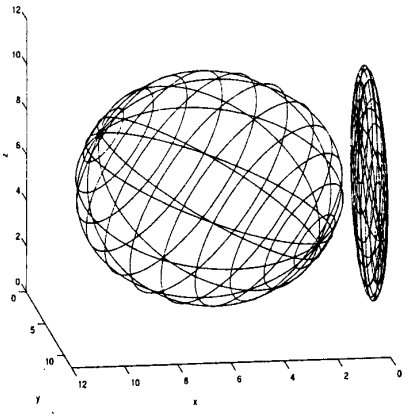

Figure 1: projection of a spherical surface of $\theta-\varphi$ to plane of $\phi_1-\phi_2$

where $\theta$ and $\varphi$ decide coordinates of the point. The point $\mathbf{p}_{s_o}$ is trasformed to $\mathbf{p}_s = {}^t(p_x,p_y,p_z)$ by the following equation.

$$\begin{aligned}\mathbf{p}_s(\theta,\varphi) &= {}^t(p_x(\theta,\varphi),p_y(\theta,\varphi),p_z(\theta,\varphi)) \\ &= \mathbf{R}\mathbf{p}_{s_o}(r,\theta,\varphi) + {}^t(6,6,6)\end{aligned} \tag{5}$$

where ${}^t(p_x(\theta,\varphi),p_y(\theta,\varphi),p_z(\theta,\varphi))$ is transported $(p_x(\theta,\varphi),p_y(\theta,\varphi),p_z(\theta,\varphi))$, $\mathbf{R}$ is a rotation matrix, and ${}^t(6,6,6)$ is transported (6,6,6). The $\mathbf{R}$ is apparently given as

$$\begin{aligned}\mathbf{R}(\alpha,\beta,\gamma) &= \begin{pmatrix} \cos\pi/6 & \sin\pi/6 & 0 \\ -\sin\pi/6 & \cos\pi/6 & 0 \\ 0 & 0 & 1 \end{pmatrix} \\ &\times \begin{pmatrix} \cos\pi/4 & 0 & -\sin\pi/4 \\ 0 & 1 & 0 \\ \sin\pi/4 & 0 & \cos\pi/4 \end{pmatrix} \\ &\times \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\pi/3 & \sin\pi/3 \\ 0 & -\sin\pi/3 & \cos\pi/3 \end{pmatrix}.\end{aligned} \tag{6}$$

Figure 1 shows the projection of the spherical surface of $\theta-\varphi$ to plane of $\phi_1-\phi_2$. Figure 1 shows that the set of $\{\theta,\varphi\}$ minimizing objective functions $\{\phi_1(\theta,\varphi),\phi_2(\theta,\varphi)\}$ is Pareto optimal [6].

In the example problem, we consider a pair of the objective function

$$\phi(\mathbf{i}_\theta,\mathbf{l}_\varphi) = \{\phi_1(\mathbf{i}_\theta,\mathbf{l}_\varphi),\phi_2(\mathbf{i}_\theta,\mathbf{l}_\varphi)\} \tag{7}$$

$$= \{p_y(\theta_{\mathbf{l}_\theta},\varphi_{\mathbf{l}_\varphi}),p_z(\theta_{\mathbf{l}_\theta},\varphi_{\mathbf{l}_\varphi})\}. \tag{8}$$

$\theta_{\mathbf{l}_\theta}$ is given as

$$\begin{aligned}\theta_{\mathbf{l}_\theta} &= f_{l_{m_\theta}} f_{l_{m_\theta-1}} \cdots f_{l_1} \\ &= \theta_{l_{\theta\,m_\theta}} \times \theta_{l_{\theta\,m_\theta-1}} \times \cdots \theta_{l_{\theta\,1}}\end{aligned} \tag{9}$$

where $m_\theta$ is the numbers of the genes selected from the chromosomes describing $\theta$ and

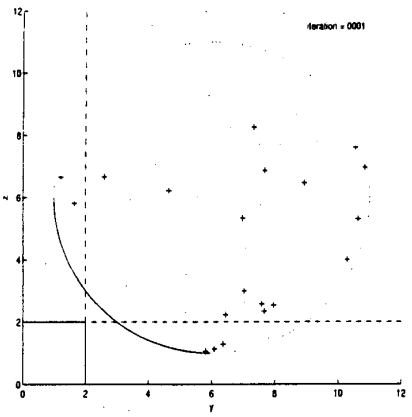


Figure 2: distribution map of individuals before search

$\mathbf{l}_\theta = \{\mathbf{l}_{\theta\,1}, \mathbf{l}_{\theta\,2}, \cdots, \mathbf{l}_{\theta\,\mathbf{m}_\theta}\}$ is a sets of indexes of genes. $\varphi_{\mathbf{l}_\varphi}$ is given as

$$\begin{aligned} \varphi_{\mathbf{l}_\varphi} &= f_{l_{m_\varphi}} f_{l_{m_\varphi - 1}} \cdots f_{l_1} \\ &= \varphi_{l_{\varphi\, m_\varphi}} \times \varphi_{l_{\varphi\, m_\varphi - 1}} \times \cdots \varphi_{l_{\varphi\, 1}} \end{aligned} \quad . \tag{10}$$

where $m_\varphi$ is the numbers of the genes selected from the chromosomes describeing $\varphi$ and $\mathbf{l}_\varphi = \{l_{\varphi\,1}, l_{\varphi\,2}, \cdots, l_{\varphi\,m_\varphi}\}$ are sets of indexes of selected genes. We have an object to confirm the effectiveness of proposed genetic algorithm for search of indexes $\mathbf{l}_\theta$ and $\mathbf{l}_\varphi$ which are indexes of minizing $\{\phi_1(\mathbf{l}_\theta), \phi_2(\mathbf{l}_\varphi)\}$. We minimize $\{\phi_1(\mathbf{l}_\theta), \phi_2(\mathbf{l}_\varphi)\}$ with the Pareto ranking scheme [6].

First, we confirm that individuals have the ability to appear various phenotypes. We generate a group in which individuals have completely homogeneous chromosomes. We show the initial distribution map of individuals of the group. Figure 2 shows that the group allows being various type of individuals in spite of their genetic homogeneity.

Next, we confirm that individuals have the ability to minimize $\{\phi_1(\mathbf{l}_\theta), \phi_2(\mathbf{l}_\varphi)\}$. We minimize the objective functions without reproducing any new child and operate the group without the alternation of generations. The results of the search of solutions are given in Figure 3. Comparing Figure 2 and Figure 3, the latter group moves against minimizing $\{\phi_1(\mathbf{l}_\theta), \phi_2(\mathbf{l}_\varphi)\}$. Figure 3 shows that individuals have tendencies to change their phenotypes and minimize $\{\phi_1(\mathbf{l}_\theta), \phi_2(\mathbf{l}_\varphi)\}$.

Furthermore, we minimize the objective functions using the group which given genetic variety, reproducts of new children and the alternation of generations. The results of the operation for the groups are given as Figure 4 proves that individuals are brought near to the trade-off surface of $\phi_2$ and $\phi_2$ by introducing Pareto ranking scheme to our genetic algorithm [6]

Figure 2, Figure 3, and Figure 4 prove that we can utilize our genetic algorithm for a solvent of some problems.

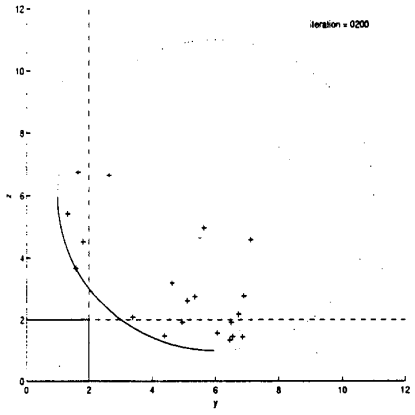


Figure 3: distribution map of individuals after search

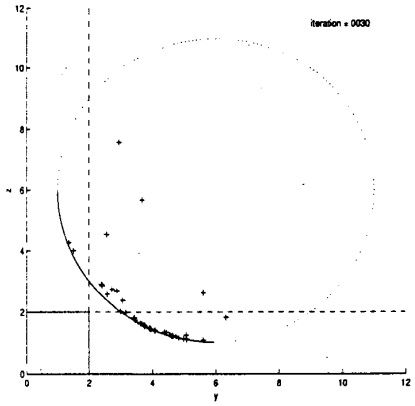


Figure 4: Result of minimizing $\{\phi_1, \phi_2\}$

## 4 Application to Control Systems Design

In this section, we discuss an application of the proposed algorithm to control systems design.

In the standard control systems design using a numerical search, the transfer function of a controller is assumed to be either

$$f(\mathbf{p},s)=\frac{p_{m_d+1}s^{m_n}+p_{m_d+2}s^{m_n-1}+\cdots+p_{m_d+m_n+1}}{s^{m_d}+p_1s^{m_d-1}+\cdots+p_{m_d}} \tag{11}$$

or

$$f(\mathbf{p},s)=\frac{p_{m_n+m_d+1}(s-p_{m_d+1})\cdots(s-p_{m_d+m_n})}{(s-p_1)\cdots(s-p_{m_d})} \tag{12}$$

where $m_d$ is the order of polynomial of the demoninator and $m_n$ is the order of polynomial of the numerator. Then the transfer function $f(\mathbf{p},s)$ is evaluated by objective functions $\phi_i(\mathbf{p})$. In the method of inequalities, the design problem is expressed as a set of algebraic inequalities consisting of the objective functions $\phi_i(\mathbf{p})$ and the largest tolerable values $\varepsilon_i$. A parameter vector $\mathbf{p}$ that simultaneously satisfies the set of inequalities is found by a numerical search.

In the proposed genetic algorithm, operators have to be specified. For control systems design based on the method of inequalities, we define operators as

$$f_{l_{\alpha i}}(s)=1/(s-\alpha_{l_{\alpha i}}) \tag{13}$$

$$f_{l_{\beta j}}(s)=s-\beta_{l_{\beta j}} \tag{14}$$

$$f_{l_{\gamma k}}(s)=\gamma_{l_{\gamma k}} \tag{15}$$

where $\alpha_{l_{\alpha i}}$ is a parameter of an operator corresponding to a pole of transfer functions, $\beta_{l_{\beta j}}$ is a parameter of an operator composing to a zero of a transfer functions, $\gamma_{l_{\gamma k}}$ is an operator corresponding to a part of gains of transfer function, $l$ are the indexes of the genes which describe the operators. Suppose that (13), (14), and (15) are given by the following characters.

$$f_j(s,f_i(s))=f_j\,f_i(s) \tag{16}$$

The transfer functions of controllers are given as

$$\begin{aligned} C(\mathbf{l},s) &= f_{\mathbf{l}}(s) \\ &= \prod_{i=1}^{m_\alpha} f_{l_{\alpha i}}(s)\prod_{j=1}^{m_\beta} f_{l_{\beta j}}(s)\prod_{k=1}^{m_\gamma} f_{l_{\gamma k}}(s) \\ &= \frac{\prod_{k=1}^{m_\gamma}\gamma_{l_{\gamma k}}\prod_{j=1}^{m_\beta}(s-\beta_{l_{\beta j}})}{\prod_{i=1}^{m_\alpha}(s-\alpha_{l_{\alpha i}})} \end{aligned} \tag{17}$$

where $m$ is the number of the genes selected from chromosome. Then the transfer functions $f_{\mathbf{l}}(s)$ is evaluated by the performance index and the result is given to the individual by the objective functions $\phi_i(\mathbf{l})$. In the method of inequalities, the design problem is expressed as a set of algebraic inequalities described by the objective functions $\phi_i(\mathbf{l})$ and the largest tolerable values $\varepsilon_i$. search is to find $\mathbf{l}=(l_i)$ satisfing $\phi_i(\mathbf{l})$ Some solutions of the problem are obtained by searchiing the combinations of index $\mathbf{l}$ that simultaneously satisfies the set of inequalities.

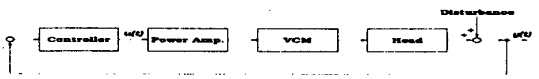


Figure 5: Block diagram of head drive control

We apply our design method to the control problem for the following mode of head of a hard disk drive (HDD). The block diagram of the control system is shown in Figure 5. The transfer function from the command of the electrical current of the voice coil motor to the position of the head is given by the following equation.

$$P(s)=\frac{-8.3297\times10^2}{s^2(s+10000)} \tag{18}$$

To keep bit error rate low, it is necessary to keep tracking error below $\pm45\mu$inch. The tracking error is caused by temperature changes, vibration of the beam of head, noise, and so on. Limitation on the power supply restricts the maximum absolute value of electrical current to 2.0A. Therefore, the design problem can be formulated by the principle of matching.

The tolerable set $T_2$ is defined by

$$T_2\triangleq\{F:\|e_i(f)\|_\infty\le\varepsilon_i,i=1,2\}. \tag{19}$$

where $e_i(t)$ is a response of HDD for input $f(t)$ and $\varepsilon_i$ is a largest tolerable values of $e_i$.

$$\left.\begin{aligned} e_1(t,f) &= -y(t) \\ e_2(t,f) &= u(t) \end{aligned}\right\}. \tag{20}$$

Note that the environment is characterized by $\mathbf{D}$. If the possible set $\mathbf{P}=F(\mathbf{D})$ belongs to the tolerable set (19) then the control system is matched. The necessary and sufficient condition for the match is given as

$$F(\mathbf{D})\subseteq T_2\Longleftrightarrow\phi_i(\mathbf{D},\mathbf{l})\le\varepsilon_i\ \text{for all}\ i\in\{1,2\} \tag{21}$$

where $\phi_i(\mathbf{D},\mathbf{l})$ is a scalar performance depending on environmental indexes $\mathbf{D}$ and gene indexes $l$. The aim of the problem is to find a combination of genes that simultaneously satisfies

$$\phi_i(\mathbf{D},\mathbf{l})\le\varepsilon_i\ \text{for all}\ i\in\{1,2\}. \tag{22}$$

$\varepsilon_i$ is given as

$$\left.\begin{array}{rcll} \varepsilon_1 & = & 4.5 \times 10^{-5} & \text{inch} \\ \varepsilon_2 & = & 2.0 & \text{A} \end{array}\right\}. \quad (23)$$

The detailed explanations of $\phi_i(\mathbf{D}, \mathbf{l})$ are given in [1].

The set of inequalities (22) should be satisfied for a successful design. The method of inequalities is effective in solving the problem based on the principle of matching [5]. Originally, the aim of the method of inequalities is to find a parameter vector satisfying the set of inequalities. Our aim is to find a combination of genes that simultaneously satisfies (22). We introduce a new objective function defined as

$$\lambda_i(\mathbf{D}, \mathbf{l}, \varepsilon_i) \triangleq \begin{cases} \phi_i(\mathbf{D}, \mathbf{l}, \varepsilon_i) - \varepsilon_i & \text{if } \phi_i(\mathbf{D}, \mathbf{l}) > \varepsilon_i \\ 0 & \text{if } \phi_i(\mathbf{D}, \mathbf{l}) \leq \varepsilon_i \end{cases} \quad i = 1, 2. \quad (24)$$

Minimizing $\{\lambda_1(\mathbf{D}, \mathbf{l}, \varepsilon_i), \lambda_2(\mathbf{D}, \mathbf{l}, \varepsilon_2)\}$ and finding vector $\{\lambda_1(\mathbf{D}, \mathbf{l}, \varepsilon_i), \lambda_2(\mathbf{D}, \mathbf{l}, \varepsilon_2)\} = \{0, 0\}$ are equivalent to finding solutions of (22) [2][3][4]. We attempt to find solutions of (22) by applying the proposed genetic algorithm for (24).

As a result of search, we obtain the following controller, satisfying the design specifications (21).

$$C_1(s) = \frac{-492.511(s + 9105.72)(s + 8227.59)}{(s + 598.155 \pm j421.731)} \times \frac{(s + 6.18817)}{(s + 548.741)} \quad (25)$$

This controller satisfies design specifications with the performance values

$$\left.\begin{array}{rcl} \phi_1(\mathbf{D}, \mathbf{p}) & = & 3.8546 \times 10^{-5} \\ \phi_2(\mathbf{D}, \mathbf{p}) & = & 0.421128 \end{array}\right. . \quad (26)$$

The order of the controller (25) is slightly higher than the order of the plant transfer function (18). Including a restriction on the order of the controller in the specifications, we can find another controller transfer function which is the less order than (26).

$$C_2(s) = \frac{-999.189(s + 13874.2)(s + 14.6789)}{(s + 277.738 \pm j160.055)} \quad (27)$$

The performance values of the above controller are given as

$$\left.\begin{array}{rcl} \phi_1(\mathbf{D}, \mathbf{p}) & = & 4.2219 \times 10^{-5} \\ \phi_2(\mathbf{D}, \mathbf{p}) & = & 0.408378 \end{array}\right. . \quad (28)$$

The above solutions prove that our genetic algorithm is useful in finding solutions of control systems design.

In the process of the search to obtain the controller (27), the following controllers which do not satisfy the specification, were appeared.

$$C(s) = \frac{-813.268(s + 14524.8)(s + 14274)(s + 124.863)}{(s + 13774.3)(s + 1561.31 \pm 1561.31j)} \quad (29)$$

$$C(s) = \frac{-631.583(s + 7720.6)(s + 286.943)}{(s + 1995.15)(s + 1561.29 \pm j255.851)} \quad (30)$$

$$C(s) = \frac{-22.3488(s + 12903.4)(s + 124.864)}{(s + 5374.7 \pm j2291.88)} \quad (31)$$

$$C(s) = \frac{-262.433(s + 1625.02)}{s + 5835.79} \quad (32)$$

The appearance of these controllers shows the characteristics of our genetic algorithm. The polynomials of their transfer functions are different from each other. The variety of the controllers do not appear in the group which is given by standard genetic algorithm.

## 5 Conclusion

In this paper, we have proposed a new genetic algorithm using individuals with redundant chromosomes. The proposed genetic algorithm can be endowed the model selection ability. at the same time. An application to optimization-based control systems design has been discussed. Using the proposed genetic algorithm, we can perform the search on a large class of the controllers with various structures. The effectiveness of the proposed method has been demonstrated for a design example.

## REFERENCES


[1] T. Satoh, T. Ishihara, and H. Inooka, "Systematic Design Via the Method of Inequalities", *IEEE Control Systems*, pp.57-65, Oct., 1996.

[2] T.K. Liu, T. Satoh, T. Ishihara, and H. Inooka, "An Application of Genetic Algorithms to Control Systems Design", *Proceeding of the First Asian control Conference*, PP.701-704, Tokyo, pp.27-30, July, 1994.

[3] T.K. Liu, T. Ishihara, and H. Inooka, "Design of Discrete-Time Control System by Multiobjective Genetic Algorithms", *Proceedings of the 1996 IEEE IECON 22nd International Conference on Industrial Electronics, Control, and Instrumentation*, PP.1618-1623, Taipei, R.O.C., Aug. 6-9, 1996.

[4] T.K. Liu, T. Satoh, T. Ishihara. and H. Inooka, "Application of Multiobjective Genetic Algorithms to Control Systems Design", *Proceedings of the* $19^{th}$ *Society of Instrument and Control Engineers Symposium on Dynamical System Theory*, PP.321-326, Hamamatsu, Japan, Oct. 30-Nov. 1, 1996.

[5] V. Zakian, "Perspectives on the principle of matching and the method of inequalities", *INT. J. CONTROL*, Vol.65, No.1, pp.147-175, 1996.

[6] David E. Goldberg, *Genetic Algorithms in Search, Optimazation, and Machine Learning*, MA, Addition-Wesley, 1989.