# Optimal Placement of Actuators in Actively Controlled Structures Using Genetic Algorithms

Singiresu S. Rao* and Tzong-Shii Pan†
*Purdue University, West Lafayette, Indiana 47907*
and
Vipperla B. Venkayya‡
*Air Force Wright Aeronautical Laboratories, Wright Patterson Air Force Base, Ohio 45431*

## Abstract

THE discrete optimal actuator location selection problem in actively controlled structures is cast in the framework of a zero-one optimization problem. A genetic algorithmic approach is developed to solve this zero-one optimization problem. To obtain successive generations that yield the solution corresponding to the maximum fitness value, this approach involves three basic operations: reproduction, crossover, and mutation. It can produce a global-optimal solution or a near-global-optimal solution if a sufficient number of generations are considered. Simplicity and parallel processing properties are the two attractive features of genetic algorithms. An example is presented to demonstrate the approach.

## Contents

For flexible structures, the equation of motion can be expressed in state-space form as

$$\dot{x} = Ax + Bu \tag{1}$$

where $x$ is the state vector, $A$ the plant matrix, $B$ the input matrix, and $u$ the input vector, which can be expressed, using the optimal linear quadratic regulator method, as

$$u = -R^{-1}B^TPx \tag{2}$$

where $P$ satisfying the following matrix Riccati equation

$$A^TP + PA - PBR^{-1}B^T + Q = 0 \tag{3}$$

where $Q$ is a positive semidefinite output weighting matrix and $R$ is a positive definite input weighting matrix. Note that, since the input matrix $B$ is a function of locations of actuators, the system Eq. (1) will be changed if the locations of actuators are changed. The objective function (criterion) proposed to be used in the actuator location selection problem is the energy dissipated by the active controller, which can be written as

$$E_c = \frac{1}{2}\int_0^\infty \dot{q}^T D_c\, \dot{q}\, \mathrm{d}t \tag{4}$$

where $\dot{q}$ is the velocity vector and $D_c$ is the induced damping matrix by the active controller. Equation (4) can be rewritten as

$$E_c = \frac{1}{2}\left\{q_0{}^T \dot{q}_0{}^T\right\} \bar{P} \begin{Bmatrix} q_0 \\ \dot{q}_0 \end{Bmatrix} \tag{5}$$

where $\{q_0\dot{q}_0\}$ is the initial state and $\bar{P}$ is the solution of the Lyapunov equation

$$A_{cl}{}^T P + \bar{P}A_{cl} = \bar{D}_c \tag{6}$$

$$A_{cl} = A + BG = A - BR^{-1}B^TP \tag{7}$$

Equation (5) shows that the energy dissipation depends on the initial state, which is not available for practical problems. A simple way[1] to eliminate the dependence on the initial state is to average it out by assuming the initial state $\{q_0\dot{q}_0\}$ to be a random variable uniformly distributed on the surface of the $2n$-dimensional unit sphere. Hence, an upper bound on the dissipation energy can be expressed as

$$\bar{E}_c = \tfrac{1}{2}\mathrm{tr}(\bar{P}) \tag{8}$$

Let

$$R^{-1} = \mathrm{diag}[\bar{r}_1\bar{r}_2 \cdots \bar{r}_n] \tag{9}$$

be the inverse of the input weighting matrix, with $\bar{r}_i$ denoting a binary variable that indicates the presence or absence of an actuator at position $i$. The zero-one optimization problem for the actuator location selection problem can be expressed as follows:

$$\underset{\{\bar{r}_1\bar{r}_2\cdots\bar{r}_n\}}{\text{Maximize}}\ \bar{E}_c \tag{10}$$

subject to

$$\bar{r}_1 + \bar{r}_2 + \cdots + \bar{r}_n = m$$

$$\bar{r}_i = \epsilon(0,1),\ i = 1,2,\ldots,n$$

#### Genetic Algorithm

Genetic algorithms[2] basically are guided random search techniques derived from the natural genetics of populations. The decision variables (or design parameters) are coded as a string of binary bits that correspond to the chromosome in natural genetics. The objective function value corresponding to the design vector plays the role of fitness in natural genetics. The artificial recombination among the population of strings is based on the fitness and the accumulated knowledge. In every new generation, a new set of strings is created by using randomized parents selection and crossovers from the old set of strings (or old generation). Although randomized, genetic algorithms are not simple random search techniques. They efficiently explore the new combinations with the available

Received Nov. 28, 1989; synoptic received June 1, 1990; accepted for publication June 8, 1990. Full paper available from National Technical Information Service, Springfield, VA 22151, by title, at the standard price (available on request). 

*Professor, School of Mechanical Engineering.
†Graduate Research Assistant, School of Mechanical Engineering.
‡Aerospace Engineer.

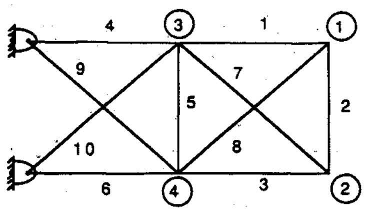


**Fig. 1a Two-bay truss.**

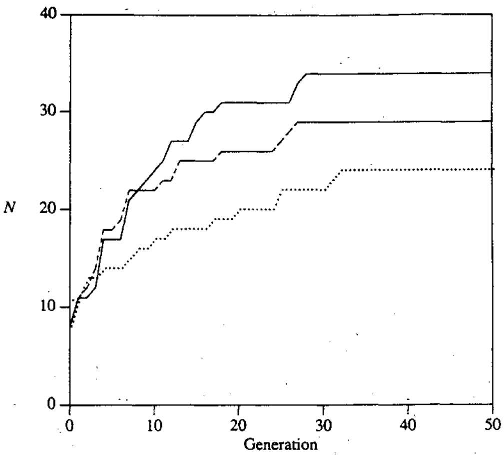


**Fig. 1b Accumulated genotypes (evaluations) for two-bay truss with three different initial conditions.**

knowledge to find a new generation with better fitness (or objective function value).

The procedure for genetic algorithms in optimization problems is described as follows:

1) An appropriate chromosome representation should be defined to represent the combinations of design parameters that correspond to the fitness or objective function values. The representation should be one-to-one mapping in order to have a normal coding and decoding processes.

2) The population size and maximum number of generations should be specified. The probabilities of crossover and mutation are selected. A set of initial population in the genetic system will also be generated.

3) Evaluate the fitness (or objective function) value of each individual in the current generation. The objective function plays the role of the environment to decide the fitness of a chromosome.

4) Apply the three operators, reproduction, crossover, and mutation, on the old generation to generate the new population for the next generation.

5) Repeat steps 3 and 4 until the maximum number of generations is achieved.

Several modifications are made to the genetic algorithm before it can be used for the solution of the actuator/sensor location selection problem:

1) The initial population is randomly generated with the restriction that no two individuals are allowed to have the same chromosome in order to get the maximal variety.

2) The successive flips of a biased coin were used to generate the initial population efficiently.

3) A normalization and scaling process has been applied to increase the differences between the fitness of different chromosomes. The normalization process is indicated by the relation:

$$f'(x) = \left[1 - \frac{f_{\max} - f(x)}{f_{\max} - f_{\min}}\right]^2 \tag{11}$$

where $f(x)$ is the objective function value with design vector $x$ and $f'(x)$ is the normalized fitness value for chromosome $x$.

4) A constraint is placed on the total number of actuators used.

**Table 1 Genetic algorithm for two-bay truss: population size = 8; number of generations = 15; and total number of genotypes = 56**

| $P_c$ | $P_m$ | Best final genotype | Total number of evaluations | Objective function | Maximum fitness found |
|---|---|---|---|---|---|
| 1.0 | 0.001 | 10101000[a] | 28 | 2.1149e + 5 | 1.0 |
| | 0.002 | 10101000 | 22 | 2.1149e + 5 | 1.0 |
| | 0.005 | 10101000 | 31 | 2.1149e + 5 | 1.0 |
| 0.9 | 0.001 | 10101000 | 35 | 2.1149e + 5 | 1.0 |
| | 0.002 | 10101000 | 35 | 2.1149e + 5 | 1.0 |
| | 0.005 | 10101000 | 35 | 2.1149e + 5 | 1.0 |
| 0.8 | 0.001 | 10100010[b] | 27 | 2.0654e + 5 | 0.9335 |
| | 0.002 | 10100010 | 28 | 2.0654e + 5 | 0.9335 |
| | 0.005 | 10101000 | 32 | 2.1149e + 5 | 1.0 |
| 0.7 | 0.001 | 10101000 | 25 | 2.1149e + 5 | 1.0 |
| | 0.002 | 10101000 | 26 | 2.1149e + 5 | 1.0 |
| | 0.005 | 10100010 | 32 | 2.0654e + 5 | 0.9335 |
| 0.6 | 0.001 | 10101000 | 23 | 2.1149e + 5 | 1.0 |
| | 0.002 | 10000101[c] | 18 | 1.8452e + 5 | 0.6659 |
| | 0.005 | 10101000 | 34 | 2.1149e + 5 | 1.0 |

[a]Global optimum. [b]Second global optimum. [c]Local optimum.

5) A variable population size is used in the actuator/sensor location selection problem. Since the genetic algorithm is highly dependent on the normalization technique used, the best chromosome that was found since the beginning of the process is not guaranteed to be included in the current generation. To preserve the best individual, the best individual in the old generation ($A$) is duplicated to the new generation if there is no other individual who is at least as good as $A$. Hence, the population size is increased by 1 if a duplication is made.

6) Although random mating is the most important mating system in many natural populations, there are certain departures from the random mating that can also be very important. Some of the important mating systems are the following: random mating—choice of mates independent of genotype and phenotype; positive assortative mating—mates phenotypically more similar than would be expected by chance; negative assortative mating—mates phenotypically more dissimilar than would be expected by chance; and inbreeding—mating between relatives.

**Results**

The feasibility of the method is demonstrated with the help of the two-bay truss shown in Fig. 1a. The number of actuators is assumed to be three. The optimal linear quadratic regulator is applied to solve the optimal control gain. The dissipation energy of active controller is used as the objective function for maximization. The output weighting matrix is assumed to be 1000*$I$ ($I$ is the identity matrix). The input weighting matrix has a fixed dimension of $n$ ($n$ is the total degree of freedom in the structure). Results obtained with different crossover probability (from 1.0 to 0.6) and mutation probability (from 0.001 to 0.005) are shown in Table 1. The number of function evaluations in 10 generations varies from 18 to 35. Most of them find the global optimal solution (1,3,5). A small number of them find the second optimal solution (1,3,7), which is the same branch (1,3) of the global optimal solution. Three initial conditions are used with $P_c = 0.7$ and $P_m = 0.001$. The results are shown in Fig. 1b. It can be seen that the total number of genotypes (configurations) generated are constant after generation 32, i.e., no new genotypes are generated.

## References

[1]Kleinman, D. L., and Athans, M., "The Design of Suboptimal Linear Time Varying Systems," *IEEE Transactions on Automatic Control*, Vol. AC-13, April 1968, pp. 150–158.

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