Optimization of a Composite Laminated Structure by Network-Based Genetic Algorithm

Jungsun Park*
School of Aerospace and Mechanical Engineering, Hankuk Aviation University, Kyunggi-do 412-791, Korea
Seokbong Song
Graduate School of Aeronautical Engineering, Hankuk Aviation University, Kyunggi-do 412-791, Korea

Genetic algorithm (GA), compared to the gradient-based optimization, has advantages of convergence to a global optimized solution. The genetic algorithm requires so many number of analyses that may cause high computational cost for genetic search. This paper proposes a personal computer network programming based on TCP/IP protocol and client-server model using socket, to improve processing speed of the genetic algorithm for optimization of composite laminated structures. By distributed processing for the generated population, improvement in processing speed has been obtained. Consequently, usage of network-based genetic algorithm with the faster network communication speed will be a very valuable tool for the discrete optimization of large scale and complex structures requiring high computational cost.

Key Words : Composite Structure, Discrete Optimization, Genetic Algorithm, Network, Client-Server, TCP/IP

1. Introduction

Genetic algorithm (GA) is a probabilistic optimization method based on the Darwin theory of evolution, "the survival of the fittest". Holland (1975) implemented originally theoretical analysis of genetic search method. Goldberg (1989) solved structural optimization problems by genetic algorithm.

Genetic algorithm has advantages of excellent convergence to a global optimized solution, which is much better than gradient-based classical methods. Despite of their attractive features, a serious problem is the high computational cost of the genetic search, which may require a large number of analyses (sometimes in the range of thousands or millions iterative analyses for population formed by generated strings in every generation). Therefore, improvements in the efficiency has been needed to make optimizing process more speedy and effective in discrete optimization using the genetic algorithm. Nowadays, many efforts have been made to speed up convergence by taking advantage of improved genetic operators, or mixed method with other optimization methods (Kogiso, et al., 1994; Park, et al., 2000).

Recently, rapid progress in the area of computer network and internetworking has enabled large-scale and sophisticated problems to be solved effectively. Networked multiprocessors has been widely applied in the various areas, for example, computer-aided design and finite-element structure analysis (Kumar and Hojjat, 1995; Regli, 1997; Regli, Gupta and Nau, 1997). Most of the distributed processing requires specific hardwares and softwares. Consideration of more

* Corresponding Author,
E-mail : jungsun@mail.hangkong.ac.kr
TEL : +82-2-300-0283; **FAX** : +82-3158-3189
School of Aerospace and Mechanical Engineering, Hankuk Aviation University, Kyunggi-do 412-791, Korea.
(Manuscript **Received** May 8, 2001; **Revised** May 8, 2002)

effective and economical method using current network system and software is needed. TCP/IP is a most popular protocol for networking. Also, client-server communication model using Windows socket can be utilized under the environments of personal computer networking (Comer and Stevens, 1997; Comer and Stevens, 1999; Shay, 1999).

The objective of the present paper is to improve processing speed of genetic algorithm for optimization of a composite laminated structure by personal computer network programming based on TCP/IP and client-server communication model using Windows socket. Iterative analysis for the generated population in every generation from genetic algorithms will be distributed to the networked personal computers.

2. Genetic Algorithm

A genetic algorithm is inspired by the basic mechanism of natural evolution and particularly suited for optimization problems such as discrete optimization (Goldberg, 1989). The genetic algorithm generates a population of random trial strings (design alternatives), analyzes the strings with the fitness function reflecting constraint condition, and then constructs new improved populations of strings by genetic operation until the optimal solution is obtained.

The operators applied successively to the parent generation for creating a new generation are usually reproduction (selection), crossover and mutation. Reproduction is a process in which individual strings are copied according to their fitness function values. Crossover allows selected individuals to trade characteristics of parent designs and create offsprings by exchanging parts of the parents' strings. Mutation performs random changes in an individual string with a low probability. It protects the population against becoming uniform.

3. Client-Server Model

Client-server model means various types of distributed computing on more than two cooperative processing environments by several related application programs. In other words, client-server computing is a method getting desired results through cooperative works between a client requesting service and a sever processing client's request.

A server is a process that is waiting to be contacted by a client process. A server uses ***bind*** to specify the local protocol port it will use, calls ***listen*** to set the length of the connection queue. The client creates a socket, calls ***connect*** to con nect to the server, and then interacts using ***send*** to send a request across the network to the server requesting some service. The server executes a command for the client on the server's system. The server provides its service to the client. The client interacts using ***recv*** to receive replies. When it finishes using the connection, it calls ***closesocket***. The server waits for the next client request to arrive (Fig. 1) (Comer and Stevens, 1997; Comer and Stevens, 1999).

There are some categories to design server software such as ***iterative vs. con-current*** server implementations, ***connectionless vs. connection-oriented*** server access, and ***stateless vs. stateful*** applications.

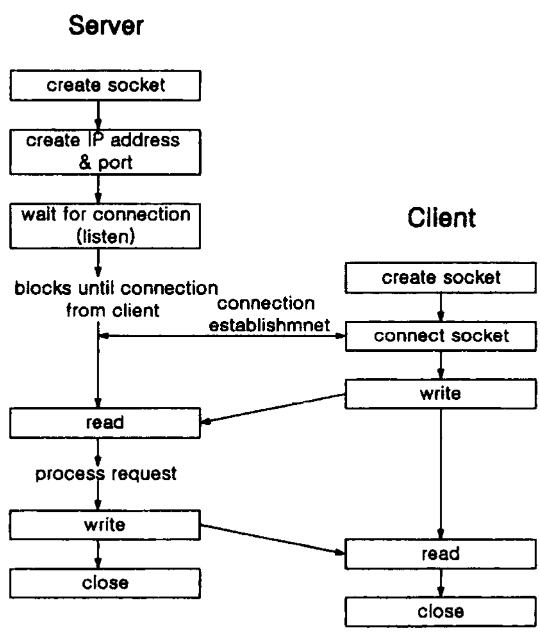

Fig. 1 Client-server model

4. Optimization of Composite laminated Structures

In order to proceed the optimization of composite laminated structures by network-based genetic algorithm, a main personal computer and sub personal computers are networked by TCP/IP protocol and client-server model under the Microsoft Windows NT environment using Windows Sockets, programmed by Microsoft Visual C^{++}, version 6. An iterative, ***connection- oriented and stateless*** sever is used. All computers are linked to 100 MBPS Ethernet.

The main computer controls genetic algorithm. A population of design variables which are generated by genetic operators and pop size in each generation is distributed to the sub computers. Each sub computer processes finite element analysis of composite laminated structures with design variables sent by the main computer, calculates fitness from a fitness function, and sends the result of fitness to the main computer. The main computer compares results of fitness which are provided by sub computers (Fig. 2).

To implement the proposed optimization model of network-based genetic algorithm, a composite laminated structure of a cantilevered curved shell shown in Fig. 3 is optimized. Cross ply $[0/90]_s$ and quasi-isotropic $[0/90/45/-45]_s$ laminates are employed. Material is Graphite/Epoxy T300/976 and mechanical properties are described in Table 1. Eight-node shell element based on Mindlin's shell theory is used for finite element analysis with half model of 8 elements using geometric symmetry. Ply numbers of each element are considered as the design variables.

To apply genetic algorithm for the optimization of the given composite structure, probabilities of crossover (P_c) and mutation (P_m) are $P_c=0.75$ and $P_m=0.01$. Objective function (F) is considered as minimizing volume of the composite laminated structure with constraints as the deflection at designated nodes and Tsai-Wu failure criterion (Reddy, 1997). To handle constraints, penalty functions are applied and pseudo objective functions are defined (Haftka and Gurdal, 1992). Formulations of optimization problem are given in Eqs. (1) ~ (4).

Minimize

$$F=\sum_{i=1}^{n} A_i \cdot X_i \cdot t \tag{1}$$

Subject to

$$C_1=\frac{W}{W_{\max}}-1\leq 0 \tag{2}$$

$$C_2=\sum_{i=1}^{n} TWF(i)-1\leq 0 \tag{3}$$

$$4\leq X_i\leq 80 \tag{4}$$

Table 2 Mechanical properties of T300/976 Gr/Ep (unit: MPa)

	E_{11}	E_{22}	G_{12}	G_{23}	ν_{12}
Fiber	230,126	15,984	8,957	6,148	0.3
Matrix	3,651.7	3,651.7	1,352.48	1,352.48	0.35

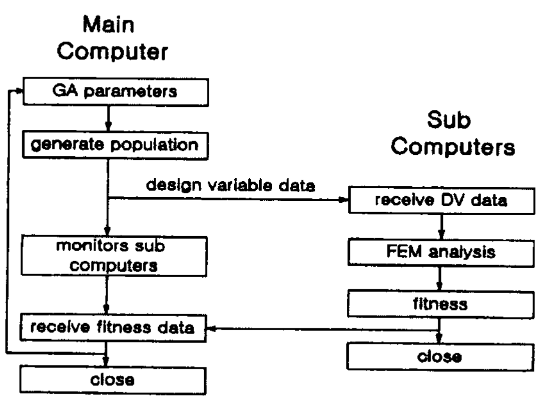

Fig. 2 Optimization model of network-based GA

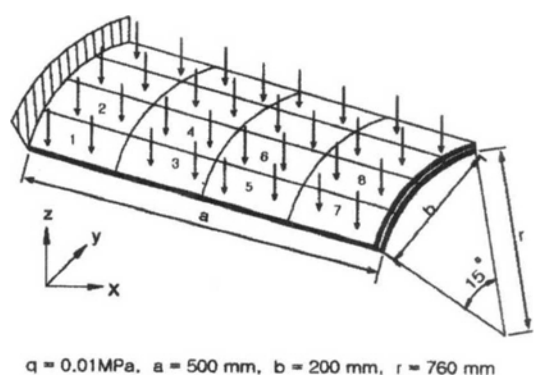

Fig. 3 Cantilevered curved shell under pressure load

where A_i is the lamina area of i-th design domain, t is lamina thickness, W is deflection at designated node, W_{max} is maximum allowable deflection, $TWF(i)$ is the failure index of Tsai-Wu criterion, X_i is ply number in the i-th design domain, and n is the number of design domains.

5. Optimization Result of a Composite Laminated Structure

To compare processing speed of one computer with that of distributed processing, speedup rate S_u is defined as follows:

$$S_u = \frac{T_1}{T_n} \tag{5}$$

where T_i is processing time with one computer and T_n is processing time with networked computers.

5.1 Case of population size=5

To optimize cross ply and quasi-isotopic laminated structure, one main computer and five sub computers have been networked. Processing time by one computer took 7,467 seconds for optimizing cross plies. By distributed optimization processing, it took 2,718 seconds. The speedup is 2.7 (Table 2). In case of quasi-isotropic plies, the speedup is 2.9 (Table 3).

Optimization results are shown in Fig. 4 and Fig. 5. 80% and 78% of initial weight with maximum laminated ply constraints were reduced.

Table 2 Processing time and speedup of cross ply (pop size=5)

1 Computer	With 5 Sub Computers	SPEED UP
7,467 sec	2,718 sec	2.7

Table 3 Processing time and speedup of quasi-isotropic ply (pop size=5)

1 Computer	With 5 Sub Computers	SPEED UP
8,139 sec	2,796 sec	2.9

5.2 Case of population size=10

In case of population size=10, two situations were performed. First, one main computer and five sub computers were networked to evaluate two strings by one sub computer for each generation. Second, one main computer and ten sub computers were networked in order to evaluate a single string by one sub computer for each generation.

In the first case, the speedups of 3.4 times at cross ply (Table 4) and 3.6 times at quasi-isotropic (Table 5) were acquired in comparison with one computer processing. Optimization results are shown in Fig. 6 and Fig. 7. 80% and

Table 4 Processing time and speedup of cross ply (pop size=10)

1 Computer	With 5 Sub Computers	SPEED UP
15,606 sec	4,609 sec	3.4

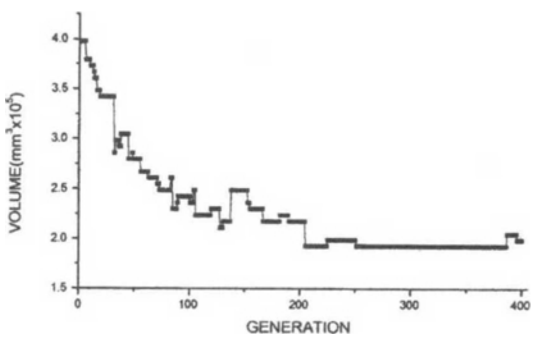

Fig. 4 Optimization of Cross ply (pop size=5)

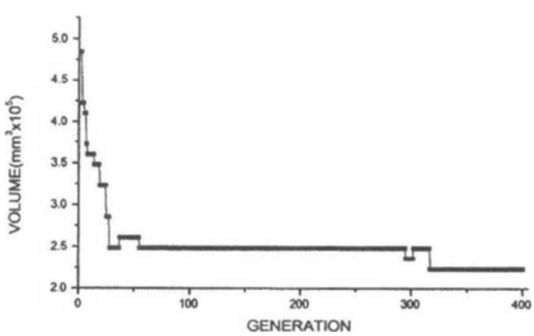

Fig. 5 Optimization of Quasi-isotropic ply (pop size=5)

Table 5 Processing time and speedup of quasi-isotropic ply (pop size=10)

1 Computer	With 5 Sub Computers	SPEED UP
15,903 sec	4,741 sec	3.6

Table 6 Processing time and speedup of cross ply (pop size=10)

1 Computer	With 10 Sub Computers	SPEED UP
15,606 sec	2,998 sec	5.2055

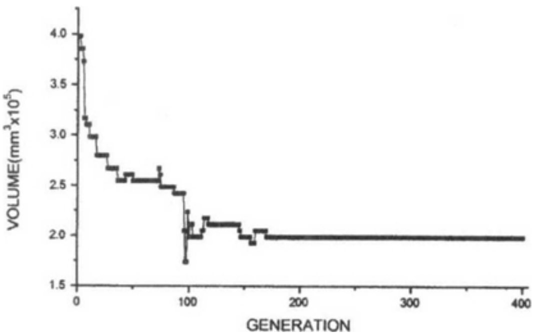

Fig. 6 Optimization of Cross ply (pop size=10)

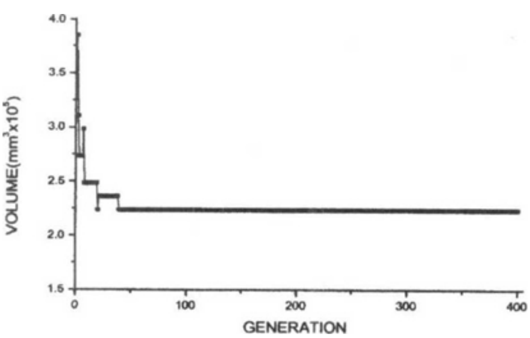

Fig. 7 Optimization of Quasi-isotropic ply (pop size=10)

78% of initial weight with maximum laminated ply constraints were reduced.

In the second case using 10 sub computers, the speedups of 5.2 times at cross ply (Table 6) and 5.3 times at quasi-isotropic (Table 7) in comparison with one computer processing were obtained.

Table 7 Processing time and speedup of quasi-isotropic ply (pop size=10)

1 Computer	With 10 Sub Computers	SPEED UP
04:25:03 (15,903 sec)	00:50:13 (3,013 sec)	5.2781

6. Conclusions

In this paper, we performed the discrete optimization of a composite laminated structure by computer network programming based on client-server communication model and TCP/IP protocol under the Microsoft Windows Sockets. Iterative calculation processes for the population of each generation were distributed to the networked computers. Improvement in processing speed of genetic algorithm was achieved.

When communication speed of network becomes faster in the future, the usage of network-based genetic algorithm for the optimization of large scale and complex structures can be an very valuable tool.

References

Comer, D. E. and Stevens, D. L., 1997, *Internetworking with TCP/IP, Volume III, Client-Server Programming and Applications, Windows Socket Version*, 2nd Edition, Prentice Hall.

Comer, D. E. and Stevens, D. L., 1999, *Internetworking with TCP/IP, Volume I, Principles, Protocols, and Archietectures*, 3rd Edition, *Prentice Hall.*

Goldberg, D. E., 1989, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley Publishing Company.

Haftka, R. T. and Gurdal, Z., 1992, *Element of Structural Optimization*, Kluwer Academic Publishers.

Holland, J. H., 1975, *Adaptation in Natural and Artificial Systems*, University of Michigan Press.

Kumar S. and Hojjat, 1995, "Distributed Finite-Element Analysis on Network of Workstations-Implementation and Applications," *Journal of Structural Engineering*, ASCE, Vol. 121, No. 10, pp. 1456~1462.

Kogiso, N., Watson, L. T., Gurdal, Z. and Haftka, R. T., 1994, "Genetic Algorithms with Local Improvement Design," *Structural Optimization*, Vol. 7, pp. 207~218.

Park, J. S., Lee, S. Y., Song, S. B. and Ryu, S. K., 2000, "Optimization of Composite Laminated Structures by the Mixed Method," KSAS journal, Vol. 28, No. 7, pp. 74~80.

Reddy, J. N., 1997, *Mechanics of Laminated Composite Plates*, CRC Press, Inc.

Regli, W. C., 1997, "Internet-Enabled Computer-Aided Design," *IEEE Internet Computing*, Vol. 1, No. 1, pp. 39~50.

Regli, W. C., Gupta, Satyandra K. and Nau, Dana S., 1997, "Towards multiprocessor feature recognition," Computer-Aided Design, Vol. 29, No. 1, pp. 37~51.

Shay, W. A., 1999, *Understanding Data Communication and Networks*, PWS Pub. Co.