

FRP Fracture Simulation Using Parallel Processing on a PC Cluster

Yoshinobu Shimamura^{1, a}, Yosuke Yamamura² and Akira Todoroki³

¹Shizuoka University, 3-5-1 Johoku, Hamamatsu, Shizuoka 432-8561, Japan

²SMG Co., Ltd., 3-18-14 Shinyokohama, Kohoku, Yokohama, Kanagawa 222-0033, Japan

³Tokyo Institute of Technology, 2-12-1 O-okayama, Meguro, Tokyo 152-8552, Japan

^atysimam@ipc.shizuoka.ac.jp

Keywords: Fracture Simulation; Parallel Processing; Finite Element Method; Domain Decomposition Method; Personal Computer Cluster.

Abstract. Recently, applications of integrated large composite structures have been attempted to many structures of vehicles. In order to improve the cost performance and reliability, it is necessary to judge the structural integrity of composite structures. Fracture simulation techniques using FEM have been developed for the purpose. Since a number of iterations of finite element analysis are required in the fracture simulation, the simulation techniques consume many memory resources and much calculation time. In this study, a personal computer cluster (PC cluster) and the domain decomposition method were incorporated into a fracture simulation system. Calculations using a Windows PC cluster were carried out to confirm the efficiency of the proposed simulation system. As a result, it is concluded that adopting the domain decomposition method and the computer cluster is remarkably efficient to reduce calculation time.

Introduction

Recently, integrated large composite structures have been applied to many structures of vehicles. To improve the cost performance and reliability of integrated composite structures, it is necessary to judge the structural integrity of composite structures with defects that were found by nondestructive inspection. Fracture simulation techniques are expected to be useful for the purpose. Many researches on the fracture simulation method using FEM have been reported [1-4]. Since a number of iterations of finite element analysis are required in the fracture simulation based on FEM, the simulation techniques consume many memory resources and much calculation time.

Recent advance on personal computer and computer network technology enables us to construct a personal computer cluster (PC cluster) as a parallel computing system [5, 6]. A PC cluster consists of several PCs and Ethernet. It does not depend on operating system. By constructing a system using computers in the office, you can use a parallel computing system at small cost.

In this study, a personal computer cluster and the domain decomposition method were incorporated into a fracture simulation system. Efficiency of the fracture simulation system was confirmed by conducting fracture simulations of a CFRP plate with bolt holes for different numbers of domain decompositions and computers on a Windows PC cluster. As a result, it is concluded that adapting a parallel computing system is remarkably efficient to reduce calculation time.

Simulation Method

Parallel Computing System. A PC cluster is a system with several personal computers connected to Ethernet. It is classified into MIMD (Multiple Instruction stream, Multiple Date stream) system [7]. PC cluster as a parallel computing system has better cost performance and scalability than commercial parallel computers.

In this study, the domain decomposition method [5, 6] is adopted as a solver of parallel FEM because large granularity is required to achieve higher computational performance for PC cluster. MPI [7], which is de facto standard of message-passing library, is used for parallel programming. A

PC Cluster used in this study consists of a maximum of 17 Windows NT workstation (Pentium II 233 MHz: 64MByte) and 100Base-TX Ethernet. If necessary, it is also possible to use computers with different operating system. Porting of the program to commercial parallel computers is also feasible.

Fracture Simulation Procedure. The simulation procedure is described as follows.

- (1) The stress-stain matrix is calculated based on the classical lamination theory (CLT).
- (2) Strain components of each element are determined by FEM analysis.
- (3) Stress components of each lamina of each element are calculated from CLT.
- (4) A fractured element is estimated one by one according to Yamada's method [8].
- (5) Degradation of stiffness is applied to the fractured element.
- (6) The above process is repeated until a given displacement is achieved.

In this simulation, triangle plane stress elements are adopted. Intralaminar fractures, which mean matrix cracking and fiber breaking, are included as fracture modes. Maximum stress criterion is adopted to discriminate fiber breaking and matrix cracking.

Degradation of stiffness after fracture is given by the following equations.

$$E_T^* = \frac{E_T}{100}, G_{LT}^* = \frac{G_{LT}}{100} \text{ (Matrix cracking)} \quad (1)$$

$$E_L^* = \frac{E_L}{100}, E_T^* = \frac{E_T}{100}, G_{LT}^* = \frac{G_{LT}}{100}, \nu_{LT}^* = \frac{\nu_{LT}}{100} \text{ (Fiber breaking)} \quad (2)$$

where a subscript L means longitudinal, a subscript T means transverse and a superscript $*$ means the elastic modulus after fracture. Detailed discussion on simulation methods compared with experimental results is in Refs. [3] and [4].

A server-client model was adopted. A server controls the convergence of equilibrium of inter-subdomain boundaries and judges an element with r_{\min} in Yamada's method. Each client carries out finite element analysis of the assigned subdomain and estimates r in Yamada's method. Time-consuming parts to solve the simultaneous equations in FEM and to estimate r are assigned to clients. Thus this method is expected to accelerate simulation.

Efficiency of Simulation Method

Analytical Model. Fracture of a CFRP plate of stacking sequence $[0/45/90/45]_s$ with bolt holes as shown in Fig.1 was simulated to confirm the efficiency of the simulation method. Load is applied by using titanium rivets. The degree of freedom is 13122. The analytical domain was divided into 4, 8 and 16 subdomains, and each subdomain was assigned to each client. Displacement control was used, and the same displacement was given at each titanium rivet. Material properties used in the analysis are listed in Table 1. Elastic analyses and Fracture simulations were carried out on conditions of 1domain/1CPU, 4subdomains/4CPUs, 8subdomains/8CPUs and 16subdomains/16CPUs. Fracture simulations were conducted up to 200 iterations of finite element analysis. A direct method solver was used in FEM for 1domain/1CPU, and an iterative method solver was used in FEM for the other conditions. In order to prevent tensile fracture of the composite plate adjacent to unloaded half of titanium rivets, Young's modulus of the unloaded half of titanium rivets is set to 1/1000 of that in Table 1.

Simulated load-displacement curves and fracture process are shown in Fig.2 and Fig.3, respectively.

Efficiency of Simulation Method. Fig. 4 shows reduction of memory used at elastic analysis. Memory consumption at fracture simulation was quantitatively similar to one at elastic analysis. Fig.4 clearly shows that the domain decomposition method is effective to reduce memory consumption.

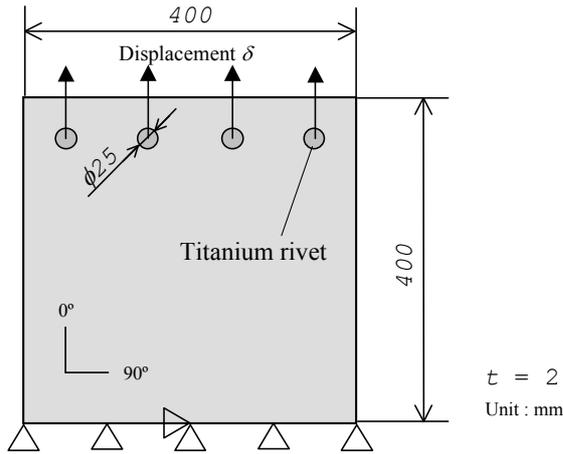


Fig.1 Analytical model

Table 1 Material properties	
CFRP	
Longitudinal modulus [GPa]	181
Transverse modulus [GPa]	10.3
Shear modulus [GPa]	7.17
Major Poisson ratio	0.28
Longitudinal tension strength [MPa]	1500
Longitudinal compress strength [MPa]	1500
Transverse tension strength [MPa]	40
Transverse compress strength [MPa]	246
Shear strength [MPa]	68
Titanium	
Young's modulus [GPa]	107
Poission ratio	0.3

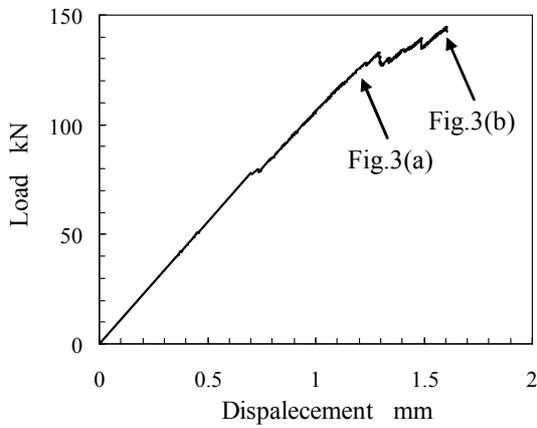
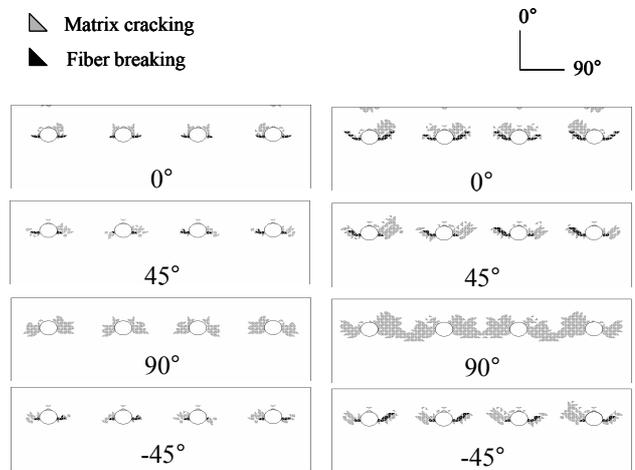


Fig.2 Load-displacement curve



(a) $\delta=1.20$ mm

(b) $\delta=1.62$ mm

Fig.3 Simulated fracture process

The efficiency of parallel computing can be discussed by using speedup ratio S , which is defined as follows.

$$S = T_1/T_n \tag{3}$$

where T_1 and T_n is the total calculation time at 1 CPU and n CPUs, respectively. An upper limit of speedup ratio equals to n .

Fig.5 shows the relationship between speedup ratio and the number of CPUs. Broken line means an upper limit of speedup ratio. For elastic analysis, parallel computing is not effective because iterative method solver consumes essentially much time. On the contrary, speedup ratios at fracture simulation are about 50% of the upper limit. In other words, the efficiency for fracture simulation is much better than that of elastic analysis. This is because initialization of inter-subdomain boundaries by using the solution of the previous step is effective to reduce the number of iterations in conjugate gradient method routine of DDM as shown in Fig.6.

As a result, it is concluded that the fracture simulation method using a PC cluster is remarkably efficient to reduce total calculation time.

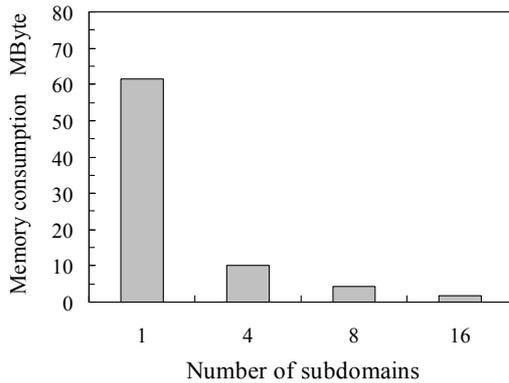


Fig.4 Memory used at elastic analysis

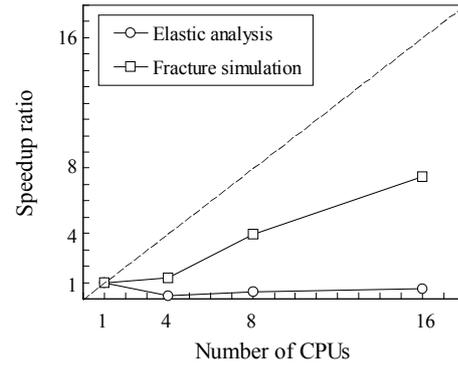


Fig.5 Speedup ratio

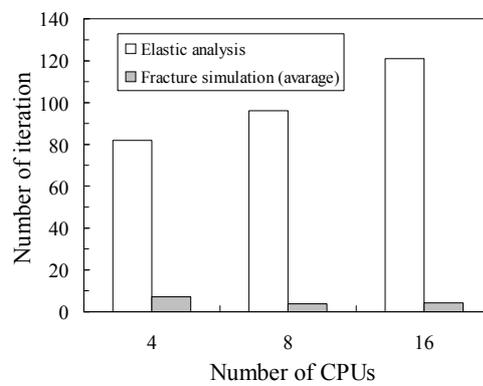


Fig.6 Number of iterations in conjugate gradient method routine of DDM

Summary

A fracture simulation method using the domain decomposition method and a personal computer cluster was proposed, and the efficiency was confirmed in view of speedup ratio. As a result, it is concluded the fracture simulation method has good computational and cost performance.

References

- [1] M. Zako and T. Tsujikami, J.Soc.Mat.Sci.,Japan, Vol.43, No.489 (1994), p.684 (in Japanese).
- [2] S. Kamiya, H. Sekine and Y. Yagishita, J.Comp.Mater. Vol.32, No.8 (1998), p.744
- [3] Y. Shimamura, A. Todoroki, H. Kobayashi, H. Nakamura and T. Inada, Proceedings of Computer Methods in Composite Materials VI, (1998), p.237
- [4] Y. Shimamura, S. Mukai, A. Todoroki, H. Kobayashi and H. Nakamura, Proceedings of JSME 12th Computational Mechanics Conference, Vol.12, (1999), pp.361-362 (in Japanese)
- [5] G. Yagawa, N. Soneda and S. Yoshimura, Computers&Structures Vo.38, No.5-6 (1991), p.615
- [6] T. Horie and H. Kuramae, Microcomputers in Civil Engineering, Vol.12 (1997), p.129
- [7] P. S. Pacheco, Paralle Programming with MPI, Morgan Kaufmann Publishers, Inc. (1997)
- [8] T. Miyoshi, M. Shiratori, M. Zako and S. Sakata, Yuugenyousohou, Jikkyousyuppan (1976), pp.39 (in Japanese)