ornl

OAK RIDGE NATIONAL LABORATORY

MARTIN MARIETTA

ORNL/TM-12732

# Prediction of Parallel NIKE3D Performance on the KSR1 System

P. S. Su
R. E. Fulton
T. Zacharia

MANAGED BY
MARTIN MARIETTA ENERGY SYSTEMS, INC.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

## DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

ORNL/TM-12732

Metals and Ceramics Division

# PREDICTION OF PARALLEL NIKE3D PERFORMANCE ON THE KSR1 SYSTEM

T. Zacharia, P. S. Su, and R. E. Fulton

Date Published: May 1995

NOTICE: This document contains information of a preliminary nature. It is subject to revision or correction and therefore does not represent a final report.

Prepared for
U.S. Department of Energy
Office of Basic Energy Sciences
KC 02 01 05 0

Prepared by the
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831-6285
managed by
MARTIN MARIETTA ENERGY SYSTEMS, INC.
for the
U.S. DEPARTMENT OF ENERGY

## TABLE OF CONTENTS

## LIST OF FIGURES

## LIST OF TABLES

# PREDICTION OF PARALLEL NIKE3D PERFORMANCE ON THE KSR1 SYSTEM*

Philip S. Su†, Robert E. Fulton‡, and Thomas Zacharia

## ABSTRACT

Finite element method is one of the bases for numerical solutions to engineering problems. Complex engineering problems using finite element analysis typically imply excessively large computational time. Parallel supercomputers have the potential for significantly increasing calculation speeds in order to meet these computational requirements. This paper predicts parallel NIKE3D performance on the Kendall Square Research (KSR1) system. The first part of the prediction is based on the implementation of parallel Cholesky ($U^TDU$) matrix decomposition algorithm through actual computations on the KSR1 multiprocessor system, with 64 processors, at Oak Ridge National Laboratory. The other predictions are based on actual computations for parallel element matrix generation, parallel global stiffness matrix assembly, and parallel forward/backward substitution on the BBN TC2000 multiprocessor system at Lawrence Livermore National Laboratory. The preliminary results indicate that parallel NIKE3D performance can be attractive under local/shared-memory multiprocessor system environments.

## 1. INTRODUCTION

Finite element method is one of the bases for numerical solutions to engineering problems. Solving the associated simultaneous equations can take up to 50% of the total computation time. Realistic engineering problems using such analysis typically require excessively large amounts of computation time. However, the emergence of parallel computers has revolutionized approaches to numerical solutions of large-scale engineering problems. New algorithms are being developed by researchers in engineering and computer science in order to exploit the tremendous potential of parallel computing. Parallel computers

*This research was supported in part by an appointment to the Oak Ridge National Laboratory Postdoctoral Research Associates Program administered jointly by the Oak Ridge National Laboratory and the Oak Ridge Institute for Science and Education and was sponsored by the Division of Materials Sciences, U.S. Department of Energy, under contract DE-AC05-84OR21400 with Martin Marietta Energy Systems, Inc.

†Postdoctoral Research Associate, Oak Ridge National Laboratory, Oak Ridge, TN 37831-6140.

‡Professor, School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0405.

have become important tools for solving complex and computationally intensive science and engineering problems. Parallel supercomputers have the potential for significantly increasing calculation speeds in order to meet these computational requirements.

This paper predicts parallel NIKE3D performance on the Kendall Square Research (KSR1) system. The first part of the prediction is based on the implementation of parallel Cholesky ($U^TDU$) matrix decomposition algorithm through actual computations on the KSR1 multiprocessor system, with 64 processors, at Oak Ridge National Laboratory (ORNL).[1] The other predictions are based on actual computations for parallel element matrix generation, parallel global stiffness matrix assembly, and parallel forward/backward substitution on the BBN TC2000 multiprocessor system at Lawrence Livermore National Laboratory (LLNL).[2] The preliminary results indicate that parallel NIKE3D performance can be attractive under local/shared-memory multiprocessor system environments.

### 1.1 NIKE3D FINITE ELEMENT CODE

NIKE3D (ref. 3) is an implicit finite element code for analyzing the finite strain static and dynamic response of three-dimensional (3-D) inelastic solids, shells, and beams. The finite element formulation accounts for both material and geometric nonlinearities. The code is fully vectorized and available on several computer platforms. A number of material models are incorporated to simulate a wide range of material behavior, including elastoplasticity, anisotropy, creep, thermal effects, and rate dependence. Slideline algorithms model gaps and sliding along material interfaces, including interface friction and single surface contact. Interactive graphics and rezoning are available for analyses with large mesh distortions. Several nonlinear solution strategies are available, including full-, modified-, and quasi-Newton methods. The resulting system of simultaneous linear equations is either solved iteratively by an element-by-element method or directly by a factorization method, for which case bandwidth minimization is optional. Data may be stored either in or out of core memory to allow for large analysis.

As a Lagrangian implicit code, NIKE3D is ideally suited for quasi-static and low-rate dynamic problems of solids, shells, and beams. NIKE3D uses a small number of relatively large time steps. A coupled system of nonlinear algebraic equations is solved at each time step by a linearization and iteration procedure. A robust and efficient nonlinear solution strategy is an essential capability for the severely nonlinear problems typically solved by NIKE3D.

Ongoing parallel computation research and development activities at ORNL have focused on exploring parallelization of NIKE3D's application on local/shared-memory multiprocessor systems such as the KSR1. Since the equation solver can use up to one-half of the total computation time, parallelization of the popular direct matrix factorization algorithm on the multiprocessor system for nonlinear problems has been tested.[1] The parallel Cholesky ($U^TDU$) matrix decomposition algorithm and skyline data storage scheme were used for the implementation.

### 1.2 KSR1 MULTIPROCESSOR SYSTEM

The KSR1 architecture is a multiprocessor system composed of a hierarchy of rings. The lowest level, ring:0, consists of a 34-slot backplane connecting 32 processing cells and 2 cells responsible for routing to the next higher layer ring, ring:1. A fully populated ring:1 is composed of the interconnecting cells from 32 ring:0 rings. A fully configured KSR1 multiprocessor system is composed of two layers containing 1024 processing cells along with two ring interconnecting cells on each ring:0. Figure 1 shows the hierarchical ring structure of the KSR1 multiprocessor.

Each processor contains a 256-KB data cache and a 256-KB instruction cache. The on-board data and instruction caches are referred to as *subcaches*. A daughter board connected to each processing cell contains 32 MB of memory referred to as *local cache*. The word size of the KSR1 multiprocessor system is 64 bits, and all functional units are based on 64-bit operands. All of the local caches make up the ALLCACHE memory system. The KSR1 architecture and chip set are designed specifically to support a local/shared-memory multiprocessor. Reference 4 provides more detail on the actual implementation. Each processor has the peak performance of 40 MFLOPS. The KSR1 multiprocessor system at ORNL has 64 (2 ring:0 rings) processors.

## 2. NUMERICAL EXAMPLES

Serial and parallel NIKE3D were implemented for two nonlinear, dynamic, two-dimensional, and 3-D finite element problems. There are four major steps in NIKE3D: matrix decomposition, element matrix generation, global stiffness matrix assembly, and forward/backward substitution. The NIKE3D applications were run in core mode to eliminate the disk input/output timing. The prediction is based on computations that were performed

ORNL-DWG 95-5858

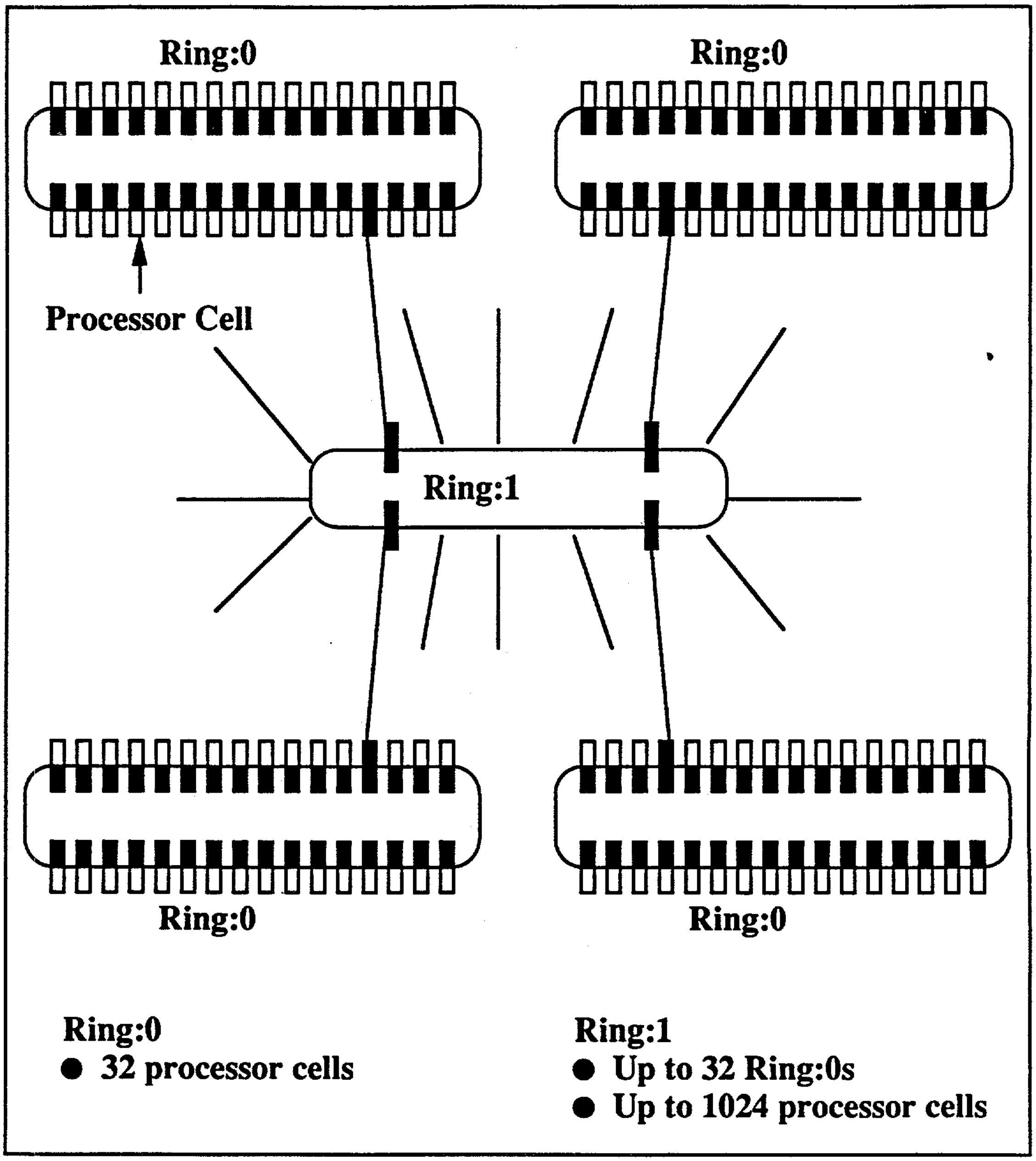


Fig. 1. The architecture of the KSR1 multiprocessor system.

either on the KSR1 multiprocessor system, with 64 processors, at ORNL or the BBN TC2000 multiprocessor system, with 104 processors, at LLNL.

### 2.1 PIPE WHIP TEST PROBLEM

This transient dynamic analysis simulates the impact of two steel pipes.[3] The pipes have 3.3125-in. OD, 0.432-in. wall thickness, and are 50 in. long. The target pipe is supported with a fixed boundary condition at each end. The second pipe swings freely about one end with an angular velocity at impact of 50 rad/s. One-half of this symmetric problem is modeled. Shell elements are used in the model, and a slide surface (type 3) is included between the pipes. There are 1484 shell elements, 8985 D. F., and 20 time steps are used with a fixed step size of $5.0 \times 10^{-5}$ s. All subroutines were compiled with the KSR1 optimization option up to the first and second levels. The central processing unit (CPU) time for the serial version is 2362 s. Figure 2 (ref. 3) shows the deformed geometry of the pipe whip test problem at several stages of analysis. The CPU times for each of the major steps and their percentages are shown in Table 1.

### 2.2 BAR IMPACTING RIGID WALL TEST PROBLEM

This transient dynamic analysis simulates a copper bar of 6.4 mm diam and 32.4 mm length impacting a rigid wall with an initial axial velocity of 0.227 mm/μs (ref. 3). The bilinear elastic-plastic material model is used with isotropic hardening. A 90° segment of this axisymmetric cross section is modeled using 972 solid elements, 3552 D. F., and 80 time steps with a fixed step size of $1.0 \times 10^{-6}$ s. None of the subroutines were compiled with the KSR1 optimization option. The CPU time for the serial version is 6060 s. Figure 3 (ref. 3) shows the stages of deformation at different time periods. The CPU times for each of the major steps and their percentages are shown in Table 1. The unusually high percentage (68.2%) of the element matrix generation step may be due to the elasto-plastic material and lack of the optimization option.

## 3. PREDICTION OF PARALLEL NIKE3D PERFORMANCE

There are four major steps in the parallelization of NIKE3D: parallel matrix decomposition, parallel element matrix generation, parallel global stiffness matrix assembly, and parallel forward/backward substitution. The prediction is based on actual computations of

ORNL-DWG 95-5859

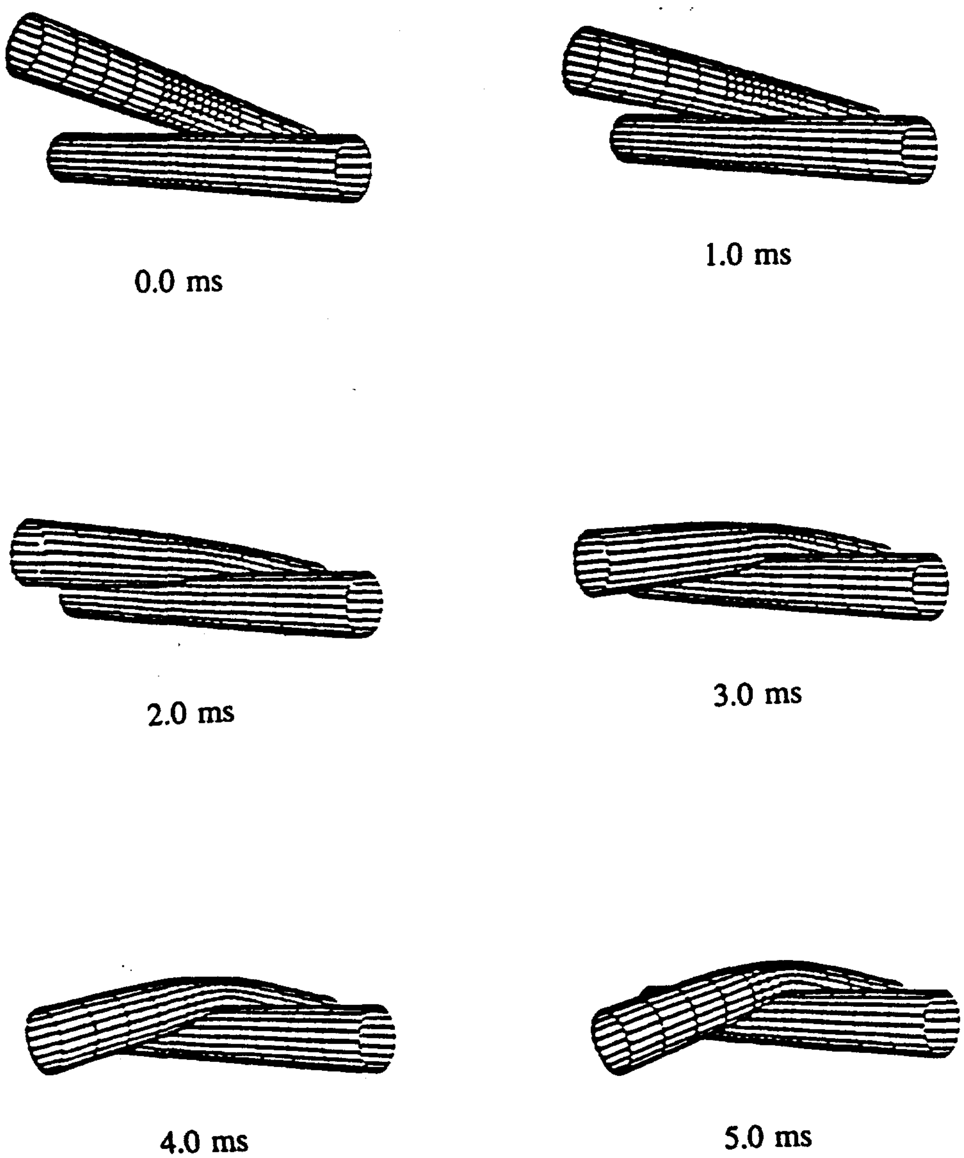


Fig. 2. The pipe whip test problem.

Table 1. Central processing unit (CPU) time for each major step in the serial version

| Major step | Pipe whip problem, CPU time (%) | | Bar impacting problem, CPU time (%) | |
|---|---|---|---|---|
| Matrix decomposition | 1096 s | (46.4) | 1224 s | (20.2) |
| Element matrix generation | 1065 s | (45.1) | 4131 s | (68.2) |
| Global stiffness matrix assembly | 67 s | (2.8) | 283 s | (4.7) |
| Forward/backward substitution | 69 s | (2.9) | 128 s | (2.1) |
| Others | 65 s | (2.8) | 294 s | (4.8) |
| Total | 2362 s | (100.0) | 6060 s | (100.0) |

ORNL-DWG 95-5860

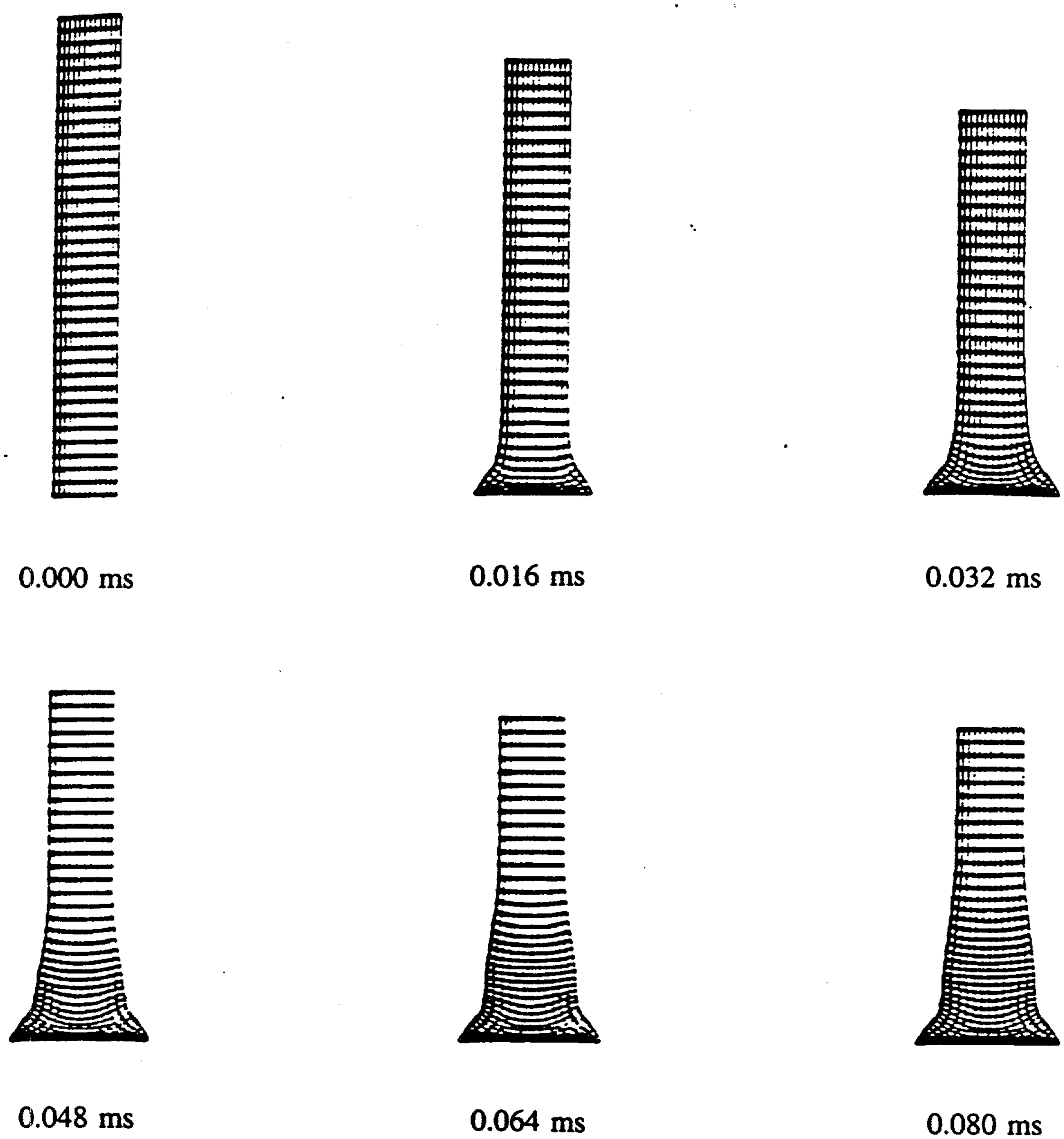


Fig. 3. The bar impacting wall test problem.

parallel matrix decomposition on the KSR1 multiprocessor system at ORNL and of parallel element matrix generation, parallel global stiffness matrix assembly, and parallel forward/backward substitution on the BBN TC2000 multiprocessor system at LLNL.

### 3.1 PARALLEL MATRIX DECOMPOSITION

Parallel matrix decomposition of pipe whip and bar impacting rigid wall test problems was tested on the KSR1 multiprocessor system.[1] This parallelization can reduce the CPU time of matrix decomposition from 1096 to 270 s for the pipe whip test problems and 1224 to 53 s for the bar impacting test problems. The higher speed-up performance for the bar impacting test problem is due to the fact that the serial version does not implement the compiler's optimization option, while the parallel version does.

### 3.2 PARALLEL ELEMENT MATRIX GENERATION

Parallel element matrix generation performance was tested on the BBN TC2000 multiprocessor system at LLNL.[2] The speed-up performance is over 90 for 104 processors under interleaved shared-memory environments. NIKE3D is a vectorized finite element code which is different from the tested code on the BBN TC2000 multiprocessor system. The element matrix generation is a highly parallelized step for a regular finite element code. It is believed that the vectorized version of element matrix generation should have the same highly parallel performance, as well as the regular version. However, due to the higher overhead, the predicted parallel element matrix generation performance for the NIKE3D should be around 30 to 40 for 64 processors on the KSR1 system, depending on the implementation of the compiler's optimization option for the serial version. The predicted CPU time for the parallel element matrix generation should be around 36 s for the pipe whip test problem and 103 s for the bar impacting test problem.

### 3.3 PARALLEL GLOBAL STIFFNESS MATRIX ASSEMBLY

The parallel global stiffness matrix assembly was tested on the BBN TC2000 multiprocessor system at LLNL.[2] The speed-up performance is over 55 for 104 processors under interleaved shared-memory environments. The predicted parallel global stiffness matrix assembly performance for the NIKE3D should be around 20 to 30 for 64 processors on the KSR1 system, depending on the implementation of the compiler's optimization option for the serial version. The predicted CPU time for the parallel global stiffness matrix assembly should be around 4 s for the pipe whip test problem and 10 s for the bar impacting test problem.

### 3.4 PARALLEL FORWARD/BACKWARD SUBSTITUTION

Forward/backward substitution with single right-hand side is not suitable for parallel computation due to the waiting steps in the forward/backward substitution procedure.[2] This forward/backward step will keep in serial and have the same CPU time, that is, 69 s for the pipe whip test problem and 128 s for the bar impacting test problem.

### 3.5 PREDICTION OF PARALLEL NIKE3D PERFORMANCE

Tables 2 and 3 show the predicted parallel NIKE3D CPU time for each major step and overall performance with respect to the serial version for pipe whip and bar impacting test problems. There are some overheads when the parallel version of an algorithm is implemented on the serial version. This overhead may be up to around 50% of the serial version CPU time. The CPU time for parallel matrix decomposition with a single processor is 1659 s compared to 1096 s for the serial version in the pipe whip test problem. This situation will reduce the overall speed-up performance.

The compiler's optimization option can reduce the tremendous CPU time required for serial computation on the KSR1 system. It affects the CPU time and percentage for each step. The optimized serial version has a higher CPU time percentage for matrix decomposition, as the pipe whip test problem in Table 1 shows. The parallel version is considered to be optimized for the comparison. That is the reason why the bar impacting test problem has much higher speed-up performance than the pipe whip test problem. It is believed that the parallel performance of the bar impacting wall test problem will be in the same range as the pipe whip test problem if the compiler's optimization option is implemented in its serial version.

The speed-up performance for banded matrix decomposition was limited. This is due to the fact that for local/shared-memory systems, there is a saturation point in the number of dedicated processors for any half bandwidth size.[5] The performance remains the same above this saturation number of processors. The number of processors and the higher level of ring, ring:1, for the KSR1 system may also affect the overall speed-up performance for NIKE3D. It is predicted that the overall parallel speed-up performance may be in the range of 5 to 10 for NIKE3D on the KSR1 system with up to 64 processors.

Table 2. Predicted parallel performance for the pipe whip test problem on the KSR1 system

| Major step | Serial version, CPU time (%) | Parallel version, CPU time (%) | Speed-up performance |
|---|---|---|---|
| Matrix decomposition | 1096 s (46.4) | 270 s (60.8) | 4.1 |
| Element matrix generation | 1065 s (45.1) | 36 s (8.1) | 29.6 |
| Global stiffness matrix assembly | 67 s (2.8) | 4 s (0.9) | 16.8 |
| Forward/backward substitution | 69 s (2.9) | 69 s (15.5) | 1.0 |
| Others | 65 s (2.8) | 65 s (14.6) | 1.0 |
| Total | 2362 s (100.0) | 444 s (100.0) | 5.3 |

Table 3. Predicted parallel performance for the bar impacting wall test problem on the KSR1 system

| Major step | Serial version, CPU time (%) | Parallel version, CPU time (%) | Speed-up performance |
|---|---|---|---|
| Matrix decomposition | 1224 s (20.2) | 53 s (9.0) | 23.1 |
| Element matrix generation | 4131 s (68.2) | 103 s (17.5) | 40.1 |
| Global stiffness matrix assembly | 283 s (4.7) | 10 s (1.7) | 28.3 |
| Forward/backward substitution | 128 s (2.1) | 128 s (21.8) | 1.0 |
| Others | 294 s (4.8) | 294 s (50.0) | 1.0 |
| Total | 6060 s (100.0) | 588 s (100.0) | 10.3 |

## 4. CONCLUSIONS

Predicted parallel performance of NIKE3D finite element code on the KSR1 multiprocessor system has been presented in this paper. The results show that the parallel NIKE3D is attractive for a multiprocessor computer architecture with local/shared-memory configuration. The compiler's optimization option can reduce the tremendous execution time of serial codes on the KSR1 system, and it is highly recommended to implement this function.

It is believed the results indicate that the parallel NIKE3D can provide an attractive strategy for high-performance finite element computations on multiprocessor, local/shared-memory systems. The results can be used to predict the parallel performance of the commercial version of NIKE3D (i.e., LS-NIKE3D) on the local/shared-memory multiprocessor systems. Further experience with parallel sparse matrix storage algorithm would be desirable as well as the iterative method for equation solver.

## 5. REFERENCES


1. P. S. Su, R. E. Fulton, and T. Zacharia, *Implementation of Parallel Matrix Decomposition for NIKE3D on the KSR1 System*, ORNL/TM-12733, Martin Marietta Energy System, Inc., Oak Ridge Natl. Lab., December 1993.

2. P. S. Su, "Parallel Subdomain Method for Massively Parallel Computers," Ph.D. thesis, Georgia Institute of Technology, Atlanta, 1992.

3. B. N. Maker, R. M. Ferencz, and J. O. Hallquist, *NIKE3D: A Nonlinear, Implicit, Three-Dimensional Finite Element Code for Solid and Structural Mechanics - User's Manual*, Lawrence Livermore Natl. Lab., Livermore, Calif., 1991.

4. *Kendall Square Research, KSR1 Principles of Operations,* KSR 8/1/91, Rev. 5.5, Kendall Square Research, Waltham, Mass., 1991.

5. R. E. Fulton, S. C. Chuang, P. S. Su, and S. Synn, "Structural Dynamics Methods for Parallel Supercomputers," Phase V, Report for MacNeal-Schwendler Corporation, Georgia Institute of Technology, Atlanta, 1991.

## INTERNAL DISTRIBUTION

1-2. Central Research Library
3. Document Reference Section
4-5. Laboratory Records Department
6. Laboratory Records, ORNL RC
7. ORNL Patent Section
8-10. M&C Records Office
11. H. W. Hayden, Jr.
12. E. Lara-Curzio
13. S. Viswanathan
14. H. W. Foglesong (Consultant)
15. E. L. Menger (Consultant)
16. J. G. Simon (Consultant)
17. K. E. Spear (Consultant)

## EXTERNAL DISTRIBUTION

18. GEORGIA INSTITUTE OF TECHNOLOGY, School of Mechanical Engineering, Atlanta, GA 30332-0405

    R. E. Fulton

19. KENDALL SQUARE RESEARCH CORPORATION, 170 Tracer Lane, Waltham, MA 02154-1379

    M. Kaufman

20. LAWRENCE LIVERMORE NATIONAL LABORATORY, Method Development Group, P.O. Box 808, L-122, Livermore, CA 94550

    B. N. Maker

21. LIVERMORE SOFTWARE TECHNOLOGY CORPORATION, 2876 Waverley Way, Livermore, CA 94550

    J. O. Hallquist

22. NASA LANGLEY, Research Center, Mail Stop 240, Hampton, VA 23681

    O. O. Storaasli

23. UNIVERSITY OF TENNESSEE, JICS, Room 104, South College, Knoxville, TN 37996-1301

    C. Halloy

24-25. DOE, OAK RIDGE OPERATIONS OFFICE, P.O. Box 2001, Oak Ridge, TN 37831

E. E. Hoffman, National Materials Programs (MS 6269)
Deputy Assistant Manager for Energy Research and Development (MS 6269)

26-27. DOE OFFICE OF SCIENTIFIC AND TECHNICAL INFORMATION, P.O. Box 62, Oak Ridge, TN 37831

For distribution as shown in DOE/OSTI-4500, Distribution Category UC-404 (Materials)