DAA/LANGLEY

IN-CAT. 61

76280

P4

GRANT TITLE:
Parallel Software Support for
Computational Structural Mechanics

SEMIANNUAL REPORT

PRINCIPAL INVESTIGATOR:
Harry F. Jordan

PERIOD COVERED:
12/1/86 through 5/31/87

GRANTEE INSTITUTION:
University of Colorado
Boulder, Colorado 80309-0425

GRANT NUMBER:
NAG-1-640

(NASA-CR-180963) PARALLEL SOFTWARE SUPPORT FOR COMPUTATIONAL STRUCTURAL MECHANICS Semiannual Report, 1 Dec. 1986 - 31 May 1987 (Colorado Univ.) 4 p Avail: NTIS HC A02/MF A01 CSCL 09B G3/61

N87-27425

Unclas
0076280

Semiannual Report

Harry F. Jordan

During the second half year of the grant for Parallel Software Support for Computational Structural Mechanics attention was turned to the application of the parallel programming methodology known as the Force. Two application issues were addressed. The first involves the efficiency of the implementation and its completeness in terms of satisfying the needs of other researchers implementing parallel algorithms. Support for, and interaction with, other CSM researchers using the Force was the main issue here, but some independent investigation of the Barrier construct, which is extremely important to overall performance, was also undertaken. Another efficiency issue which was addressed was that of relaxing the strong synchronization conditions imposed on the self-scheduled parallel DO loop. Processes can now start on such a loop before all have arrived and exit before all have completed their work, but enough synchronization is maintained to ensure that the execution of such a loop does not overlap the execution of another self-scheduled construct or a subsequent execution of this one. The Force was extended by the addition of logical conditions to the cases of a parallel case construct and by the inclusion of a self-scheduled version of this construct.

The second issue involved applying the Force to the parallelization of finite element codes such as those found in the NICE/SPAR testbed system. Consideration of a very small and independent subroutine gave an indication of some of the problems to be encountered and attention then moved to the INV processor of SPAR, which accounts for considerable execution time and promises to reward parallelization effort. One of the more difficult problems encountered is the determination of what information in COMMON blocks is actually used outside of a subroutine and when a subroutine uses a COMMON block merely as scratch storage for internal temporary results. Assuming the worst case behavior, i.e. that any result in a COMMON block is used elsewhere, can often lead to a large reduction in possible parallelism over the more reasonable assumptions which are virtually certain to be correct, but which are hard to prove unequivocally.

Reports generated during this period and supported in whole or in part by this grant include a complete description of the Force [1], the philosophy of parallel programming which it embodies, its current state as a macro preprocessor, proposed extensions and implementation issues arising on the four machines on which it has been implemented. This report will appear as a chapter of the book *The Characteristics of Parallel Algorithms* edited by Leah Jamieson, Dennis Gannon, and Robert Douglass to be published by MIT Press. A revision of the Force User's Manual [2] was also prepared in order to include descriptions of the changes in the self-scheduled parallel DO and in the parallel case statement. A report on the study of the barrier implementation was also produced [3].

Effort begun during this period and still in progress includes an analysis of the NICE/SPAR system from the point of view of data dependencies. To obtain significant gains in performance, the program must be parallelized at a large scale. This second level of attack requires a thorough understanding of the structure of SPAR as a Fortran program. This structure is not well documented at the level required for parallelization, and the required analysis of data dependencies will contribute to any effort to implement SPAR on a multiple instruction stream computer.

BIBLIOGRAPHY

[1] H. F. Jordan, "The Force," *ECE Tech. Rept. 87-1-1,* Computer Systems Design Group, Dept. of Electrical and Computer Engineering, University of Colorado, Boulder, CO, 80309-0425 (Jan. 1987).

[2] H. F. Jordan, M. S. Benten and N. S. Arenstorf and A. V. Ramanan, "Force User's Manual," *ECE Tech. Rept. 86-1-4R,* Computer Systems Design Group, Dept. of Electrical and Computer Engineering, University of Colorado, Boulder, CO, 80309-0425 (June 1987).

[3] N. S. Arenstorf and H. F. Jordan, "Comparing Barrier Algorithms," *ECE Tech. Rept. 87-1-2,* Computer Systems Design Group, Dept. of Electrical and Computer Engineering, University of Colorado, Boulder, CO, 80309-0425 (June 1987).