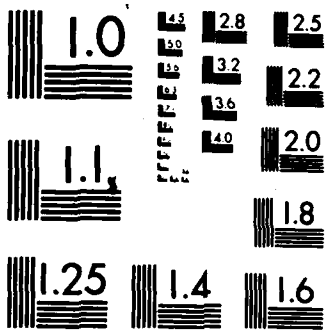
AD-A137 774 HIGH PERFORMANCE PARALLEL COMPUTING(U) TEXAS UNIV AT AUSTIN DEPT OF COMPUTER SCIENCES J C BROWNE 31 DEC 82 AFOSR-TR-84-0042 AFOSR-82-0091 1/1

UNCLASSIFIED F/G 9/2 NL

END
FILMED
3-84
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION: UNCLASSIFIED	1b. RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE: -	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S): AFOSR-TR- 84-0042
6a. NAME OF PERFORMING ORGANIZATION: University of Texas; 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION: Air Force Office of Scientific Research
6c. ADDRESS (City, State and ZIP Code): Department of Computer Sciences, Austin TX 78712	7b. ADDRESS (City, State and ZIP Code): Directorate of Mathematical & Information Sciences, Bolling AFB DC 20332
8a. NAME OF FUNDING/SPONSORING ORGANIZATION: AFOSR; 8b. OFFICE SYMBOL (If applicable): NM	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: AFOSR-82-0091
8c. ADDRESS (City, State and ZIP Code): Bolling AFB DC 20332	10. SOURCE OF FUNDING NOS.: PROGRAM ELEMENT NO. 61102F; PROJECT NO. 2304; TASK NO. A3; WORK UNIT NO.
11. TITLE (Include Security Classification): SEE REMARKS ON BACK	
12. PERSONAL AUTHOR(S): J.C. Browne	
13a. TYPE OF REPORT: Final; 13b. TIME COVERED: FROM 1/1/82 TO 31/12/82	14. DATE OF REPORT (Yr., Mo., Day): DEC 31, 1982; 15. PAGE COUNT: 11
16. SUPPLEMENTARY NOTATION	
17. COSATI CODES: FIELD / GROUP / SUB. GR.	18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)

DTIC SELECTE FEB 1 3 1984 S E

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

The three major research areas in this project were parallel structuring of computations, basic software for support of parallel computations, and parallel architectures and supporting hardware. The model problem selected for formulation for parallel execution during this period was a simplified Monte Carlo code obtained from Los Alamos National Laboratory. This code, called GAMTEB, is a simulation of the transport of photons through a carbon cylinder. There are three major components in this code --- 'bank' management, particle management and statistics gathering. Bank management deals with the initiation of particles, particle management moves particles and develops trajectories, and the statistics manager accumulates the properties of interest for the particles (cross-sections for departure, etc.) as they move through the carbon cylinder. The execution cost of each component and the communication requirements between components led to the definition of several alternative parallel formulations for this code. A linear speed up in total problem execution was found and documented. The study of parallel formulations of (CONTINUED)

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT: UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT ☒ DTIC USERS ☐	21. ABSTRACT SECURITY CLASSIFICATION: UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL: John P. Thomas, Jr., Capt, USAF	22b. TELEPHONE NUMBER (Include Area Code): (202) 767- 5026; 22c. OFFICE SYMBOL: NM

DD FORM 1473, 83 APR EDITION OF 1 JAN 73 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

AD A137774

DTIC FILE COPY

(2)

ITEM #11, TITLE: HIGH PERFORMANCE PARALLEL COMPUTING

ITEM #19, CONTINUED: numerical algorithms has covered three topics. The work of Kapur and Browne on parallel formulation of odd-even elimination has been the method of choice for sequential and scalar machines for solution of the block tridiagonal linear equations which result from discretization of Poisson's equation. It was established that the odd-even elimination method is preferable for parallel formulation where the degree of parallelism exceeds approximately 16. This work has now been formulated into a paper. Professor D. Scott analyzed parallel formulations of solutions for dense linear systems. This topic has not been extensively considered in the past because the communication geometry which must be established varies depending on the selection of pivots. It appears that the reconfigurability of communication paths offered by TRAC may make parallel formulation on TRAC an attractive option. There is a wide spectrum of algorithms from which to select for parallel formulation. Dr. Scott has identified those which are most promising in terms of both low operation count and communication geometry requirements.

AFOSR-TR- 84-0042

FINAL

High Performance Parallel Computing

J. C. Browne

The University of Texas at Austin

Austin, Texas 78712

AFOSR Grant AFOSR-82-0091

1.0 RESEARCH ACCOMPLISHMENTS FOR 1982

1.1 Overview

The three major research areas have been parallel structuring of computations, basic software for support of parallel computations and parallel architectures and supporting hardware.

The work on parallel structuring of computations falls into three categories. First of these is parallel structuring of complete programs or applications. The example to be discussed below is a Monte Carlo simulation of particle movement. The second category is parallel formulation of specific numerical algorithms. The third topic is parallel formulations of non-numeric algorithms, in particular radix sorting.

Design, development and analysis of basic software for parallel computing has received major emphasis. Basic software for support of parallel computations on

84 02 10 121

reconfigurable network architectures is an almost entirely unstudied problem area.

The third major area is continuing development of parallel architectures and hardware support for parallel architectures in the context of the Texas Reconfigurable Array Computer (TRAC).

1.2 Parallel Structuring Of Computations

The model problem which was selected for formulation for parallel execution during this year is a simplified Monte Carlo code which was obtained from Los Alamos Scientific Laboratories. This code, called GAMTEB, is a simulation of the transport of photons through a carbon cylinder There are three major components in this code -- "Bank" management, particle management and statistics gathering. Bank management deals with the initiation of particles, particle management moves the particles and develops trajectories and the statistics manager accumulates the properties of interest for the particles (cross-sections for departure, etc.) as they move through the carbon cylinder. The execution cost of each component and the communication requirements between components led to the definition of several alternative parallel formulations for this code. The processing time for the "bank" is very small compared to particle management and statistics gathering which are of approximately equal magnitude. The first and most obvious is to utilize n copies of the entire code. This approach requires the cascading of statistics gathering through a tree structure. It may be difficult to balance this approach. The second approach decomposes into bank managers, particle managers and statistics gathering, maps each to a separate set of processors and establishes communication paths between the three components. The communication requirements for each particle between the bank manager and the particle manager can be satisfied by an exchange of two packets of approximately 6-8 bytes apiece. The completion of processing for each particle requires the movement between the particle manager and the statistics manager of approximately 147 real numbers. Thus data movement between the bank manager and the statistics manager can be accomplished by packets whereas shared memories are required for efficient communication between the particle manager and the statistics manager. This structure and a related structure where a bank manager and the particle manager were combined have been mapped onto the TRAC architecture. A linear speed up in total problem execution is found. A report [GAJ82] describing this work is being prepared and will be available in the next few weeks. (A draft version of the report is now available in machine readable format.)

The study of parallel formulations of numerical algorithms has covered three topics. The work of Kapur and Browne [KAP81] on parallel formulation of odd-even elimination has been extended to include odd-even reduction. Odd-even reduction has been the method of choice for sequential and scalar machines for solution of the block tridiagonal linear equations which result from discretization of Poisson's equation. It was established that the odd-even elimination method is preferable for parallel formulation where the degree of parallelism exceeds approximately 16. This work has now been formulated into a paper [KAP82], a copy of which will be submitted as a report to AFOSR.

Professor D. Scott has begun an analysis of parallel formulations of solutions for dense linear systems. This topic has not been extensively considered in the past because the communication geometry which must be established varies depending on the selection of pivots. It appears that the reconfigurability of communication paths offered by TRAC may make parallel formulation on TRAC an attractive option. There is a wide spectrum of algorithms from which to select for parallel formulation. Dr. Scott has identified those which are most promising in terms of both low operation count and communication geometry requirements.

J. C. Browne has, together with D. Scott, begun development of a general specification of the interconnection geometry required for solution for arbitrary discretizations of partial differential equations containing only first and second partial derivatives. It appears that all of the known methods decompose into two stages. A discretization [5-point, 9-point,etc.] establishes a linear recurrence relation. The first stage applies a unit operation to the equations defined by the linear recurrence and generates a different recurrence relation. There are then two possibilities for the second stage. Either the new recurrences established by the unit operation of the method is mapped back to the original recurrence for repeated application of the unit computation or else the algorithm is modified to conform to the new recurrence. We choose the former. It then appears that while there is a specific n-neighbors relationship required for each discretization and equation that the mapping back to the initial recurrence can always be obtained by an inverse perfect shuffle. It appears that a unified characterization of communication geometries can be defined. This may lead to the formulation of an architecture which will be nearly optimal for a wide spectrum of solution methods for partial differential equations.

It was conjectured in the original proposal that if a method for key compression which preserves key order without excessive introduction of duplicates could be developed then parallel radix sort methods with linear speed-up could be developed. We have conducted experiments on ordered key compression techniques. The experiments utilized keys selected from a Zipf's Law distribution. A number of different order preserving compressions were tried (division, nth root, logx, nth root of a polynomial in the key)

No single transformation was adequate but a two phase scheme whereby the first k most significant sets were taken as a "segment" number and the balance of the key hashed proved quite successful. If the distribution of keys is known then duplicates can be kept to a very small percentage and are highly clustered around the "correct" value. For example, for keys of length 108 bits a duplicate ratio of 9×10^{-3} was obtained for Zipf's law distribution of keys, for a hashed key size of 30 bits. This work is being continued and extended. A preliminary report has been written [VAR82] and a paper will be prepared.

1.3 Basic Software For Parallel Computing

It is necessary in a reconfigurable parallel architecture that the processors themselves be virtualized. This virtualization of processors is akin, at the physical level, to the virtualization implemented on virtual machine monitors at a logical level. The work on basic software for parallel computing has included the implementation design of the virtualization monitor for the individual processors of the parallel architecture of TRAC. The basic machine monitor for accomplishing this task for TRAC has been designed. It will be the subject of the Ph.D. dissertation of Mr. Daniel Canas. A paper [BRO82] describing the principles upon which this work is founded will be presented at the International Workshop on High Level Language

Computer Architecture in December of 1982.

We have also implemented during this year a version of Pascal⁻ which includes both message and shared memories interprocess communication capabilities. This Pascal is being used as the implementation vehicle for PRM and Computation Structures Language CSL compiler and also as the vehicle for parallel programming for TRAC while the CSL system is being completed.

The Computation Structures Language compiler has been nearly completed and is now in test.

We have also initiated work on a data flow programming language which defines inputs and outputs at the functional level rather than at the instruction level. This work is in a preliminary stage.

1.4 Parallel Architectures And Hardware Support

There were two major efforts in this area during the year. The first was the initial attempt to decompose the switch node for the banyan network into a chip set. A design which decomposes the 90 chip set into six chips, two control chips which are design for CMOS implementation and 4 high performance chips (ECL) which are from the original set. The latter chips implement the data movement functions.

The second major effort was to define the synchronous packet moving capabilities in the TRAC network in detail and demonstrate that a unified functionality which will support both the data realignment and data movement requirements of SIMD architectures and the input/output transmission required for data flow architecture can be realized on TRAC's banyan network.

2.0 FUTURE RESEARCH

2.1 Parallel Structuring Of Computations

The first priority for the new year will be to push the parallel formulation of the GAMTEB Monte Carlo code into execution and to measure its behavior.

The second major thrust will be to look at parallel structuring of general discrete event simulation. We will attempt first to develop a formulation of discrete event simulation which lends itself to parallelism by the specification of independence of events and secondly to apply this technique to some non-trivial application such as a war-gaming problem.

Work will continue on the numerical algorithm front on both parallel structuring of dense matrices on TRAC and the further development of communication requirements for general discretizations of partial differential equations.

We will begin a simulation study of the MSB radix formulations of parallel sorting to determine the buffering requirements in order to make radix based parallel sorting competitive with or superior to comparison based sorting in terms of data movement.

2.2 Basic Software Systems

The first task during the year will be to implement the processor resident monitor which virtualizes processor structures. The algorithms for the processor resident monitor have been included in the implementation design and coding will begin in or about February.

The other major implementation effort will be to carry through to implementation design the function level data flow system and to attempt to use it as a vehicle for algorithm formulation.

When the CSL compiler is completed we will begin programming of examples in CSL including translation of the Monte Carlo code from the parallel Pascal formulation to a CSL formulation.

We will also pick up during the year the job resource analysis problem which was set aside after Doug DeGroot completed his degree [DEG81].

2.3 Parallel Architectures

The major problem in TRAC has been the lack of reliability in the hardware. Major effort will be expended during the year on the establishment of fault tolerant techniques for the switch network. We will continue to work upon the development of chip sets for the banyan network node.

An effort will be mounted upon the design of "expanding" banyans. An "expanding" banyan network is constructed from nodes with unequal fan-out and spread and has the maximum number of nodes at the middle level of the multilevel switch rather than at the apex or base of the switch network. The interest in "expanding" banyans has arisen from the fact that our problem formulation studies have indicated that switchable memory will be needed for efficient formulation of many problems. The expanding banyans will greatly add to the capability of the current banyan networks for implementing switchable memory.

References

1. [BRO82] Browne, J.C. and Lipovski, G.J., "Reconfigurable Network Architectured Computer Systems: An Environment for Parallel Computing", (to appear in the Proc. of the International Workshop on High-Level Computer Architecture, Fort Lauderdale, FL, Dec. 1982).

2. [GAJ82] Gajwani, A., "Structuring the GAMTEB Monte Carlo Code for TRAC", preliminary TRAC report, September 1982.

3. [DEG82] DeGroot, R.D., "Mapping Computation Structures onto SW-Banyan Networks", Ph.D. Dissertation, Department of Computer Sciences, The University of Texas at Austin, December 1981.

4. [KAP81] Kapur, R.N. and Browne, J.C., "Block Tridiagonal System Solution on Reconfigurable Array Computers", Proc. 10th International Conf. on Parallel Processing, Traverse City, Michigan, August 1981.

5. [KAP82] Kapur, R.N. and Browne, J.C., "Block Tridiagonal System Solution on Reconfigurable Array Computers (Revised)", submitted to SIAM Journal on Scientific and Statistical Computing.

6. [VAR82] Varman, P., "Order Preserving Hashing", Department of Computer Sciences, The University of Texas at Austin, July 1982.

END

FILMED

3-84

DTIC