2

Annual Report on Contract N00014-89-J-1528 in FY89

1. Summary

The tasks accomplished as of December 1, 1989 are:

A) Completion of a monograph entitled "*DECOMP: an Implementation of Dantzig-Wolfe Decomposition for Linear Programming*,"

B) Implementation of DECUBE, a linear programming decomposition code in Fortran on an Intel iPSC/2 Hypercube computer with 64 processors;

C.) Analysis and empirical study of computational strategies and the dynamics of information in parallel decomposition. Results were presented at the ORSA/TIMS Joint National Meeting in Vancouver, B.C., Canada in May 1989;

D) The multistage, multiproduct material requirements planning problem with capacity constraints. Results were presented at the ORSA/TIMS Joint National Meeting in New York, NY, in October 1989; AND

E) The dynamic traffic assignment problem. Initial results on the application of a nested decomposition algorithm using parallel computation were presented at the Joint USA-Italy Workshop on Urban Traffic Networks in Capri, Italy in June 1989. (KR)

A brief description of each task accomplished and its contribution to the project is given in the following sections.

2. Description of Tasks

A. Documentation of DECOMP

A large part of our computational work is based on DECOMP, a Fortran implementation of the Dantzig-Wolfe decomposition algorithm for block-angular linear programs. This code has evolved over a period of fifteen years into a robust and relatively portable experimental tool for large-scale mathematical programming. While it has been distributed by the PI to many researchers worldwide, comprehensive documentation was not available. This task resulted in a complete tutorial, user's guide and programmer's manual in the form of a 206-page monograph [6] published in the Springer-Verlag Lecture Notes Series in Economics and Mathematical Systems.

B. Design and Implementation of DECUBE:

The goal here is to implement Dantzig-Wolfe decomposition on an Intel iPSC/2 hypercube computer. The iPSC/2-d6 consists of 64 Intel 80386 processors connected in a six-dimensional hypercube topology. It is a superb example of the cost-effectiveness of multiprocessors and is becoming standard equipment in many advanced computing centers. The machine we used is at the Oak Ridge National Laboratory and access was provided by courtesy of Dr. Tom Dunigan and Dr. Mike Heath. The design and implementation of DECUBE drew heavily on our experience with DECOMPAR on the experimental multicomputer CRYSTAL at the University of Wisconsin at Madison. The DECOMPAR project [4] was also supported by ONR on a previous grant. Since the iPSC/2 provided a much more advanced and stable environment than CRYSTAL (which is now obsolete and out of commission), we were able to obtain significant performance measures and comparisons.

Our initial experience [3] demonstrated that DECUBE can outperform state-of-the-art commercial LP software (MPSX/370 V2) on mainframe supercomputers (IBM 3090) that cost 20 to 30 times more than the iPSC/2.

C) Computational Strategies and Dynamics of Information:

One of the major tasks under the current grant is to study the empirical performance of various computational strategies for decomposition. A typical DW-type algorithm using a multicomputer can be summarized as follows.

0. Node 0 controls Master Problem; each of the other nodes controls one or more Subproblems.
1. Node 0 solves Master Problem and sends prices to other nodes.
2. Non-zero nodes solve Subproblems and send proposals to Node 0.
3. Repeat 1 and 2 until termination.

For DW-type decomposition algorithms, internodal communication involves

a) prices form Node 0 to other nodes;
b) proposals from other nodes to Node 0.

The generation and transfer of such data can be done quite efficiently and constitute part of the design of DECUBE. The load balancing aspect is a lot more challanging and provides a rich source of research. As stated, Node 0 is idle while others are active and vice versa. Also, some non-zero node may be idle while others are still active. We propose several variants of the basic approach in order to improve on load balancing. These are known as information schemes.

I. *Basic Information Scheme (BIS)*

This is the standard strategy described above.

II. *Early Start Information Scheme (ESIS)*

Master is activated as soon as first proposal is generated.

III. *Early Termination Information Scheme (ETIS)*

Termination of the subproblems is synchronized

III. *Intermediate Prices Information Schemes (IPIS)*

All nodes are kept active if possible with prices and proposals communicated as soon as available.

As we progress from BIS through IPIS, we expect to gain efficiency in terms of load balancing. However, the convergence rate of the underlying algorithm may be adversely affected. This implies a trade-off for additional internodal communications. How these factors interact is of significant theoretical and empirical interest. It became apparent that truly instant feedback is impracticable in our computing environment because so much information is generated that the communication network gets jammed very quickly. Therefore, a variation on this theme is necessary to regulate the flow of information among the processors. With this design, we have learned that for most problems tested, a regulated scheme with maximal utilization of the master node performs best. Neither ETIS nor IPIS are promising. Further analysis are providing explanation of such behavior and allowing one to predict the effectiveness of various strategies based on the observed behavior of the subproblems. The technical details of this task is reported in T.C. Lee's doctoral dissertation [8]. The implications of the results are described in a technical report [5] entitled "Dynmics of Information in Distributed Decision Systems."

D. <u>The multistage, multiproduct material requirements planning problem with capacity constraints.</u>

We showed [7] that the basis in a class of linear programs arising from material requirements planning can be triangularized. This allows for efficient adaptation of the Simplex Method similar to those for network problems. It also suggests that for finite-loading (i.e. capacitated) MRP, a decomposition approach exploiting both subproblem structure and parallel processing can be effective for handling complex problems in multiproduct, multistage, multiperiod production systems. The initial results (reported in [3] and [9]) of using DECUBE on this class of problems are very promising. Examples with up to 30,000 constraints were solved under 15 minutes. A commercial LP code took over 50 minutes on a mainframe supercomputer costing 20 to 30 times more than the hypercube. We believe that this development will have substantial impact on the application of optimization to operations management.

E. The Dynamic Traffic Assignment Problem:

Previously, the PI has shown that a class of non-linear, non-convex dynamic network flow problems can be solved as a sequence of linear programs. The LP's have the staircase structure typical of time-phased problems. Initial experience of applying a nested decomposition algorithm using parallel computation was promising [2]. The PI, was invited to present these results at the Joint USA-Italy Workshop co-sponsored by the National Science Foundation on Urban Traffic Networks in Capri, Italy, in June 1989. This model will be further studied in the context of parallel computation and used as an important class of test problems throughout the project.

3. Technical Reports under the Current Contract

[1] Ho, J.K., "Nonprocedural implementation of mathematical programming algorithms," in R. Sharda (ed.), *Impact of Recent Advances in Computer Science on Operations Research*, North Holland, New York (1989) 226-237.

[2] J.K. Ho, "Solving the dynamic traffic assignment problem on a hypercube computer" (to appear in *TRANSPORTATION RESEARCH*)

[3] J.K. Ho and S.K. Gnanendran, "Distributed decompostion of block-angular linear programs on a hypercube computer" (submitted to *MATHEMATICAL PROGRAMMING*).

[4] Ho, J.K., T.C. Lee and R. Sundarraj, "Linear programming decomposition using parallel computers," *MATHEMATICAL PROGRAMMING* 42 (1988) 391-405.

[5] Ho, J.K., and T.C. Lee, "Dynamics of information in distributed decision systems" (submitted to *MANAGEMENT SCIENCE*).

[6] Ho, J.K. and R. Sundarraj, *DECOMP: an Implementation of Dantzig-Wolfe Decomposition for Linear Programming* , Springer-Verlag Lecture Notes Series on Economics and Mathematical Systems, No. 338, New York, 1989.

[7] Ho, J.K. and William McKenney, "Triangularity of the basis in linear programs for materials requirements planning" *OPERATIONS RESEARCH LETTERS* (1988) 273-278..

[8] Lee, Tak C., "Distributed optimization of linear programs using Dantzig-Wolfe decomposition", Doctoral dissertation, University of Tennessee, 1988.

[9] S.K. Gnanendran, "Resource-constrained material requirements planning using parallel linear programming", Doctoral dissertation, University of Tennessee, 1989.