DISTRIBUTED DATA STRUCTURES
FOR SCIENTIFIC COMPUTATION

L. R. Scott¹
J. M. Boyle²
and B. Bagheri³

ABSTRACT. We study language issues related to sharing variables in the context of programming non-shared memory multi-processors, such as the hypercube processors that are the subject of this conference. The language constructs developed are intended to support the technique of creating parallelism by distributing data structures, and operations on them, over several processors. We present two approaches to this problem and describe our implementation and experience with both. Computational results for both the NCUBE and Intel hypercubes are presented. Also described is an abstract framework that underlies the two approaches; this framework holds the promise of allowing automated subdivision of scientific computing problems.

1. INTRODUCTION. The most obvious way to utilize a parallel computer is to identify tasks that can be done independently of each other and assign them separately to individual processors. This approach works well for some applications, such as ray tracing in computer graphics, but in many others the structure of the problem does not immediately offer many such opportunities. When the possibilities of task independence have been exhausted, one way to “create” parallelism is to divide a problem’s data structures, and the operations performed on them, and distribute them among the processors. Such an approach works well, e.g., for solving many problems in scientific computation, and its implementation on shared memory machines has been discussed before (see [3,4,8]). The main requirement is synchronization of individual processors (or processes), cf. Dongarra and Sorensen [5] and references therein, and techniques for doing this exist in some languages, e.g., in Ada.

In this paper, we focus on language issues related to automating this approach to programming parallel computers when the target is a parallel computer that does NOT have a shared memory, such as the hypercube-architecture machines that are the subject of this conference. The main new language constructs introduced here give the appearance

¹Departments of Computer Science and of Mathematics, Pennsylvania State University, University Park, PA 16802
²Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439
³Department of Mathematics, University of Michigan, Ann Arbor, MI 48109