Evaluating the XMT Parallel Programming Model

Dorit Naishlos, Joseph Nuzman, Chau-Wen Tseng and Uzi Vishkin
University of Maryland
e-mail: dorit@cs.umd.edu

Explicit-multithreading (XMT) is a parallel programming model designed for exploiting on-chip parallelism. Its features include a simple thread execution model and an efficient prefix-sum instruction for synchronizing shared data accesses. By taking advantage of low-overhead parallel threads and high on-chip memory bandwidth, the XMT model tries to reduce the burden on programmers by obviating the need for explicit task assignment and thread coarsening. This paper presents features of the XMT programming model, and evaluates their utility through experiments on a prototype XMT compiler and architecture simulator. We find the lack of explicit task assignment has slight ef-fects on performance for the XMT architecture. Despite low thread overhead, thread coarsening is still necessary to some extent, but can usually be automatically applied by the XMT compiler. The prefix-sum instruction provides more scalable synchronization than traditional locks, and the simple run-until-completion thread execution model (no busy-waits) does not impair perform-ance. Finally, the combination of features in XMT can encourage simpler par-allel algorithms that may be more efficient than more traditional complex ap-proaches.

Proceedings of the 15th International Parallel and Distributed Processing Symposium (IPDPS '01)
1530-2075/01 $10.00