Evaluating the Potential of Programmable Multiprocessor Cache Controllers

John B. Carter
Mike Hibler
Ravindra R. Kuramkote

UUCS-94-040

Department of Computer Science
University of Utah

Abstract

The next generation of scalable parallel systems (e.g., machines by KSR, Convex, and others) will have shared memory supported in hardware, unlike most current generation machines (e.g., offerings by Intel, nCube, and Thinking Machines). However, current shared memory architectures are constrained by the fact that their cache controllers are hardwired and inflexible, which limits the range of programs that can achieve scalable performance. This observation has led a number of researchers to propose building *programmable multiprocessor cache controllers* that can implement a variety of caching protocols, support multiple communication paradigms, or accept guidance from software. To evaluate the potential performance benefits of these designs, we have simulated five SPLASH benchmark programs on a virtual multiprocessor that supports five directory-based caching protocols. When we compared the off-line optimal performance of this design, wherein each cache line was maintained using the protocol that required the least communication, with the performance achieved when using a single protocol for all lines, we found that use of the "optimal" protocol reduced consistency traffic by 10-80%, with a mean improvement of 25-35%. Cache miss rates also dropped by up to 25%. Thus, the combination of programmable (or tunable) hardware and software able to exploit this added flexibility, e.g., via user pragmas or compiler analysis, could dramatically improve the performance of future shared memory multiprocessors.