Abstract

The Bulk Synchronous Parallel (BSP) programming model is attractive for parallel programs that proceed in “phases”: it is simpler to program than more general message-passing models such as MPI, and facilitates performance forecasting. While restricting overlap of communication and computation, it lends itself to efficient communication through aggregation and scheduling, thereby preventing unnecessary endpoint contention. InfiniBand is a standard high-speed, low-latency interconnect for clusters. This paper explores the possibility of efficient BSP implementation on computer clusters. We present an architecture and prototype implementation of BSP in an InfiniBand-connected PC-based cluster, utilizing the unique capabilities of InfiniBand for efficient implementation of key BSP functions such as barrier synchronization. This enables demanding applications to achieve good speedups. Our commodity BSP-IB cluster forms a well-balanced parallel machine, outperforming custom multiprocessor BSP machines (e.g., IBM SP and Origin 2000) in every respect.

1 Introduction

In parallel programs, it is desirable to permit computations to proceed as flexibly and aggressively as possible while maintaining correctness. Consequently, message-passing models like MPI [1] that permit a processor to send results to another processor as soon as they become available appear very attractive. Unfortunately, however, this flexibility can only be exploited if the programmer takes upon himself much of the responsibility for avoiding incoherence. (The communication libraries offer mechanisms that assist in so doing, but the programmer is charged with deciding when reduced synchronization may be allowed.) Also, many parallel programs are structured in phases, be it because of the very nature of the algorithms being programmed or as a means for the programmer to manage complexity. In each phase, a processor computes intermediate results over a subset of the data, and subsequently exchanges results with some subset of the processors. In phase-based programs, the