Multi-toroidal Interconnects: Using Additional Communication Links to Improve Utilization of Parallel Computers

Yariv Aridor[1], Tamar Domany[1], Oleg Goldshmidt[1], Edi Shmueli[1], Jose Moreira[2], and Larry Stockmeier[3]

[1] IBM Haifa Research Labs, Haifa, Israel
{yariv,tamar,olegg,edi}@il.ibm.com
[2] IBM Watson Research Center, Yorktown, NY
jmoreira@us.ibm.com
[3] IBM Almaden Research Center
stock@acm.org

Abstract. Three-dimensional torus is a common topology of network interconnects of multicomputers due to its simplicity and high scalability. A parallel job submitted to a three-dimensional toroidal machine typically requires an isolated, contiguous, rectangular partition connected as a mesh or a torus. Such partitioning leads to fragmentation and thus reduces resource utilization of the machines. In particular, toroidal partitions often require allocation of additional communication links to close the torus. If the links are treated as dedicated resources (due to the partition isolation requirement) this may prevent allocation of other partitions that could, otherwise, use those links. Overall, on toroidal machines, the likelihood of successful allocation of a new partition decreases as the number of toroidal partitions increases.
This paper presents a novel "multi-toroidal" interconnect topology that is able to accommodate multiple adjacent meshed and toroidal partitions at the same time. We prove that this topology allows connecting every free partition of the machine as a torus without affecting existing partitions. We also show that for toroidal jobs this interconnect topology increases machine utilization by a factor of 2 to 4 (depending on the workload) compared with three-dimensional toroidal machines. This effect exists for different scheduling policies. The BlueGene/L supercomputer being developed by IBM Research is an example of a multi-toroidal interconnect architecture.

1 Introduction

Tightly coupled multicomputers provide a natural way to build large-scale parallel systems. A tightly coupled multicomputer consists of a collection of *nodes*. Each node has one or several CPUs, memory, and network connections. Such systems are intended to run massively parallel computational jobs. A typical job requires a set of nodes, called a *partition*, connected in a particular fashion, e.g.

D. Feitelson, L. Rudolph, and U. Schwiegelshohn (Eds.): JSSPP 2004, LNCS 3277, pp. 144–159, 2005.

a three-dimensional rectangular block wired as mesh or torus. The job management system has to allocate a partition to each job and to schedule the waiting jobs optimally in order to maximize the machine utilization and to reduce the jobs' response times.

The high performance network that connects the nodes is designed with the jobs' topology requirements in mind. A frequently used interconnect topology is a three-dimensional mesh or torus where every node is connected to six neighbors, two in each dimension (a torus differs from a mesh in that the six edges are connected in a wrap-around fashion). This interconnect topology is simple, scalable (the number of links grows linearly with the machine size), and fits many types of real-world computations. Examples of three-dimensional toroidal parallel systems are the Cray T3D and T3E machines [2,3,4].

Job partitions are usually allocated contiguously, i.e. the constituent nodes are geometrically adjacent. Contiguous allocation simplifies allocation algorithms and facilitates partition isolation, i.e. localization of the intra-job communications within the partition. The latter is required both for security reasons and/or to reduce message congestion on shared network links. Under the isolation requirement, the nodes that form a partition and the network links that connect them are dedicated resources used by at most one job at a time.

Efficient partition allocation is of critical importance to the system performance in terms of resource utilization and job response times. As shown in the next section, a toroidal partition often requires allocation of additional links to close the torus. If the links are treated as dedicated resources, this prevents allocation of other partitions that could, otherwise, use those links. Overall, on toroidal machines, the likelihood of successful allocation of a new partition decreases as the number of toroidal partitions increases. The problem is particularly acute when a first-come-first-served (FCFS) scheduling is used. Backfilling [7,8] is an improvement over FCFS, but we show below that the adverse effect of isolated toroidal partitions on utilization exists independently of the scheduling policy used.

In this paper we present a novel approach, hereafter called *multi-toroidal topology*, that augments the traditional toroidal interconnect with additional links to improve machine utilization while allocating isolated rectangular partitions connected as mesh or a torus. Unlike other existing solutions, such as full crossbar interconnects [5], the number of additional links is small, linear in the number of allocation units in the machine (see Section 2 for more details), and thus is inexpensive and highly scalable. A variant of multi-toroidal interconnect is implemented in the upcoming BlueGene/L supercomputer developed by IBM Research [16].

Multi-toroidal topology suggests a practical compromise between additional hardware complexity (due to additional communication links) and better system performance (in terms of utilization) gained by introducing it. Utilization is increased due to the ability to allocate multiple adjacent meshed and toroidal partitions thanks to the additional connectivity options and to the possibility