

Chapter 4

Image Processing On Reconfigurable Meshes With Buses*

Jing-Fu Jenq
Tennessee State University

and

Sartaj Sahni
University of Florida

Abstract

In this chapter, we describe different reconfigurable mesh with buses architectures and show how several image processing problems can be solved efficiently on the weakest of these. The specific problems considered are: area and perimeter of components, shrinking and expanding, clustering, and template matching. In many cases, the resulting algorithms are faster than those for other parallel computer architectures.

1. Introduction

Recently, several researchers have proposed a modification to the well studied mesh architecture in which the interprocessor links are replaced by a reconfigurable bus. The resulting parallel computer architecture is called a reconfigurable mesh with bus (RMB). In all two dimensional $N \times N$ RMB computers, the N^2 processors are located at the N^2 grid points of an $N \times N$ grid (just as in a traditional mesh computer). However, the traditional linkage between mesh adjacent processors is absent. Instead, interprocessor communication takes place via a reconfigurable bus. The RMB family of architectures includes the RMESH, PARBUS, polymorphic torus, and the mesh reconfigurable network (MRN). The architectures have become popular because they are relatively easy to program and because many problems can be solved very efficiently on them. In fact, it is possible to solve some problems faster on an RMB computer than is theoretically possible on a PRAM computer (See for example: [10, 34, 35, 39]).

*This research was supported, in part, by the National Science Foundation under grant MIP-9103379.

In the RMESH [53, 26, 27, 28] version of an RMB (Fig. 1), the bus is comprised of $2N(N-1)$ segments that correspond to the $2N(N-1)$ interprocessor links in a traditional mesh. However, each of these segments has a switch on it. The switch on a bus segment may be set in the open or closed position by either one of the two processors at the ends of the segment. With the switch on each segment closed, all N^2 processors are attached to the same bus. As a result, if any one processor writes data to this bus, all remaining processors can read this data from the bus in the next cycle. I.e., it is possible to broadcast data in $O(1)$ time.

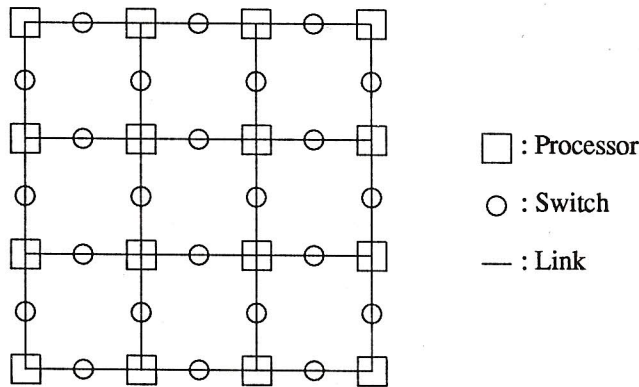


Fig. 1 A 4×4 RMESH

By opening the switches on all vertical bus segments and closing them on all horizontal ones, we form N independent buses with each one spanning the processors in a row of the mesh (Fig. 2(a)). Column buses that span all processors in the same column may be formed by closing all switches on column segments and opening all on row segments (Fig. 2(b)). As in other work dealing with the RMESH model, we assume that the time to broadcast data on a bus or subbus is $O(1)$. In the exclusive write model, only one processor can write data to a given (sub)bus at any time. In the concurrent write model several processors may simultaneously write to the same (sub)bus. Rules are provided to determine which of the several writers actually succeeds (e.g., arbitrary, maximum, exclusive or, etc.).

The PARBUS of [51, 52] is also a member of the RMB family. An $N \times N$ PARBUS (Fig. 3) is an $N \times N$ mesh in which the interprocessor links are bus segments and each processor has the ability to connect together arbitrary subsets of the four bus segments that connect to it. Bus segments that get so connected behave like a single bus. The bus segment interconnections at a processor are done by an internal four port switch. If the up to four bus segments at a processor are labeled N (North), E (East), W (West), and S (South), then this switch is able to realize any set, $A = \{A_1, A_2\}$, of connections where $A_i \subseteq \{N, E, W, S\}$, $1 \leq i \leq 2$ and the A_i 's are disjoint. For example $A = \{\{N, S\}, \{E, W\}\}$

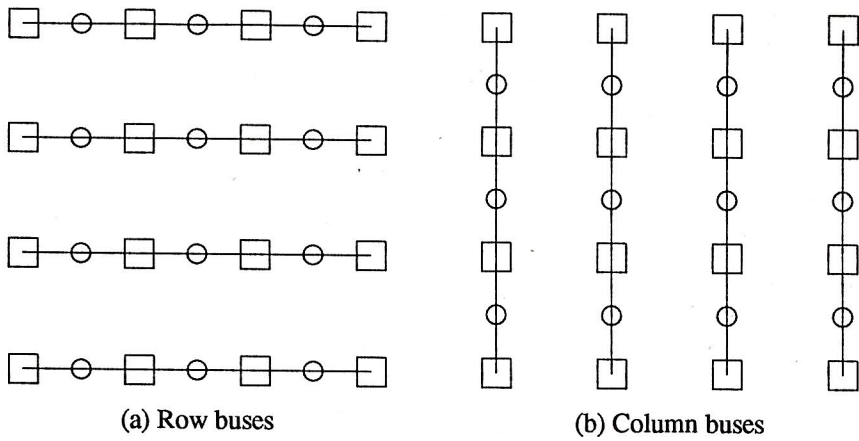
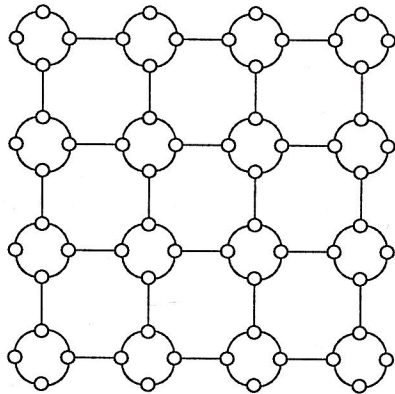


Fig. 2 Subbuses in an RMESH

results in connecting the North and South segments together and the East and West segments together. If this is done in each processor, then we get, simultaneously, disjoint row and column buses. If $A = \{[N,S], \phi\}$, then only column buses are formed. Similarly, when $A = \{[E,W], \phi\}$ only row buses are formed. If $A = \{[N,S,E,W], \phi\}$, then all four bus segments are connected. PARBUS algorithms for a variety of applications can be found in [29, 52, 23, 10, 11, 12, 13, 36, 37, 38, 49]. Observe that in an RMESH the realizable connections are of the form $A = \{A_1\}$, $A_1 \subseteq \{N,E,W,S\}$.

Fig. 3 A 4×4 PARBUS

The polymorphic torus and the mesh reconfigurable network (MRN) are

