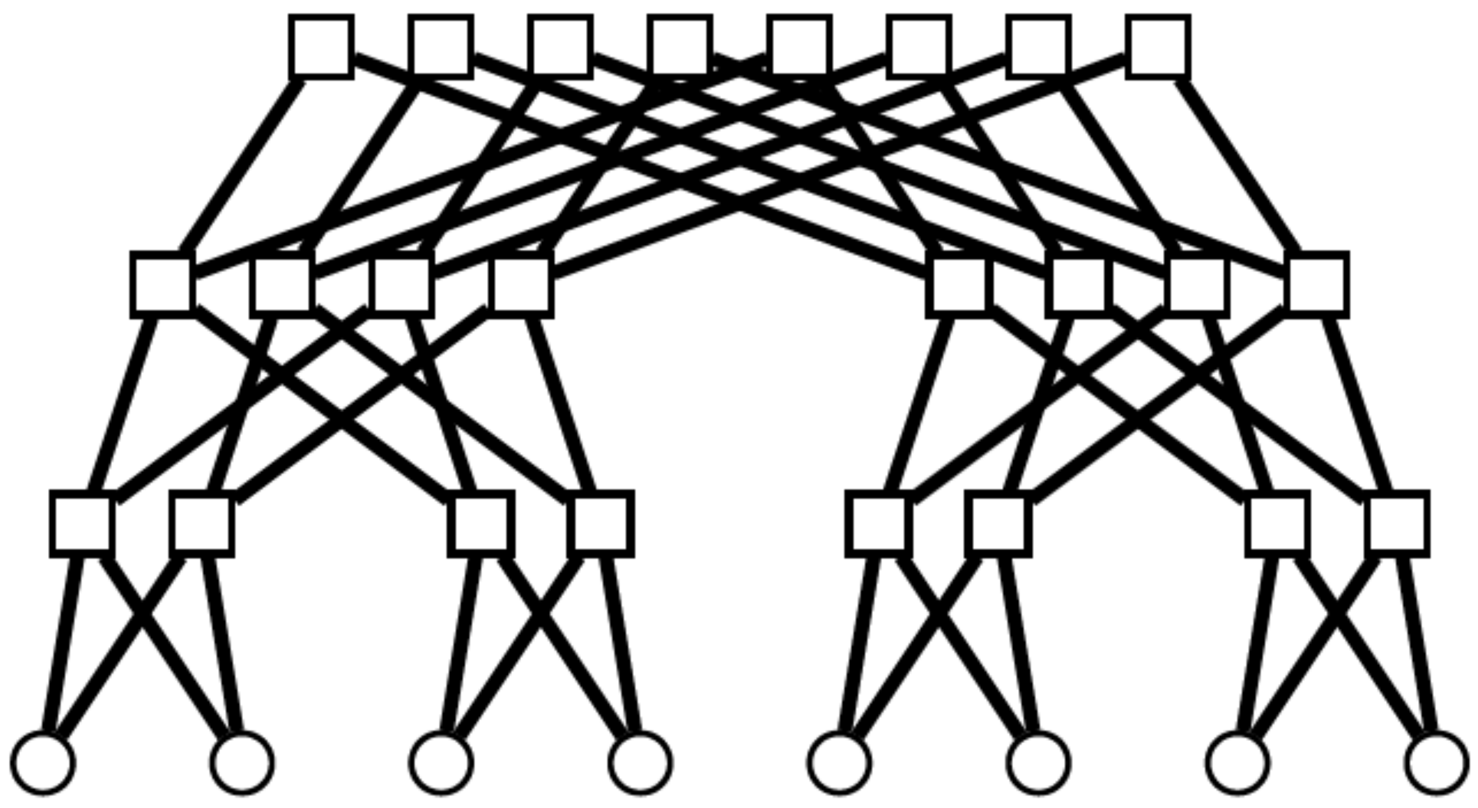
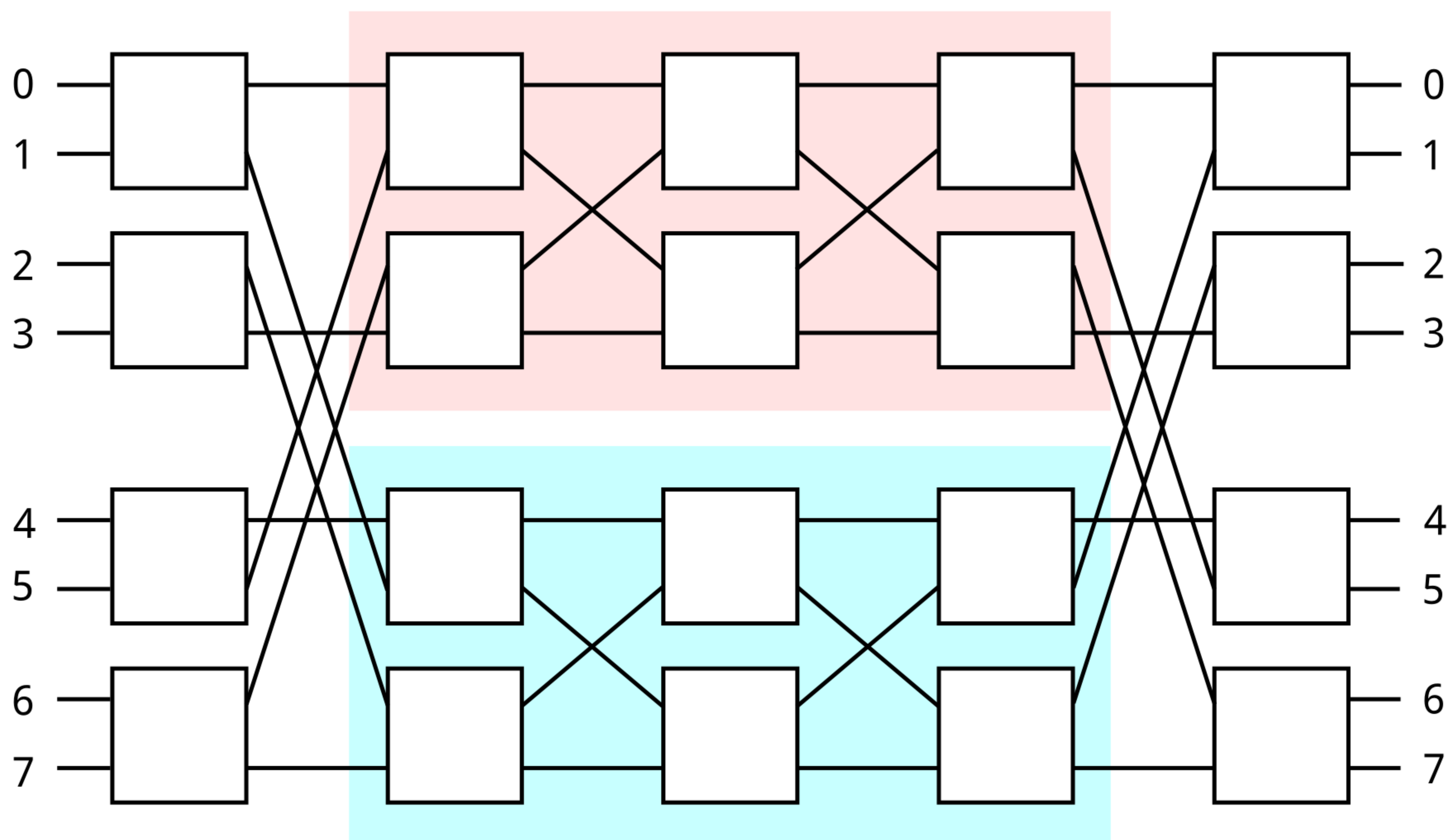
Reducing the Number of Data Collisions in the Interconnection Network of a General-Purpose Parallel Architecture

Benjamin Tovar Matthews
Supervised by David May

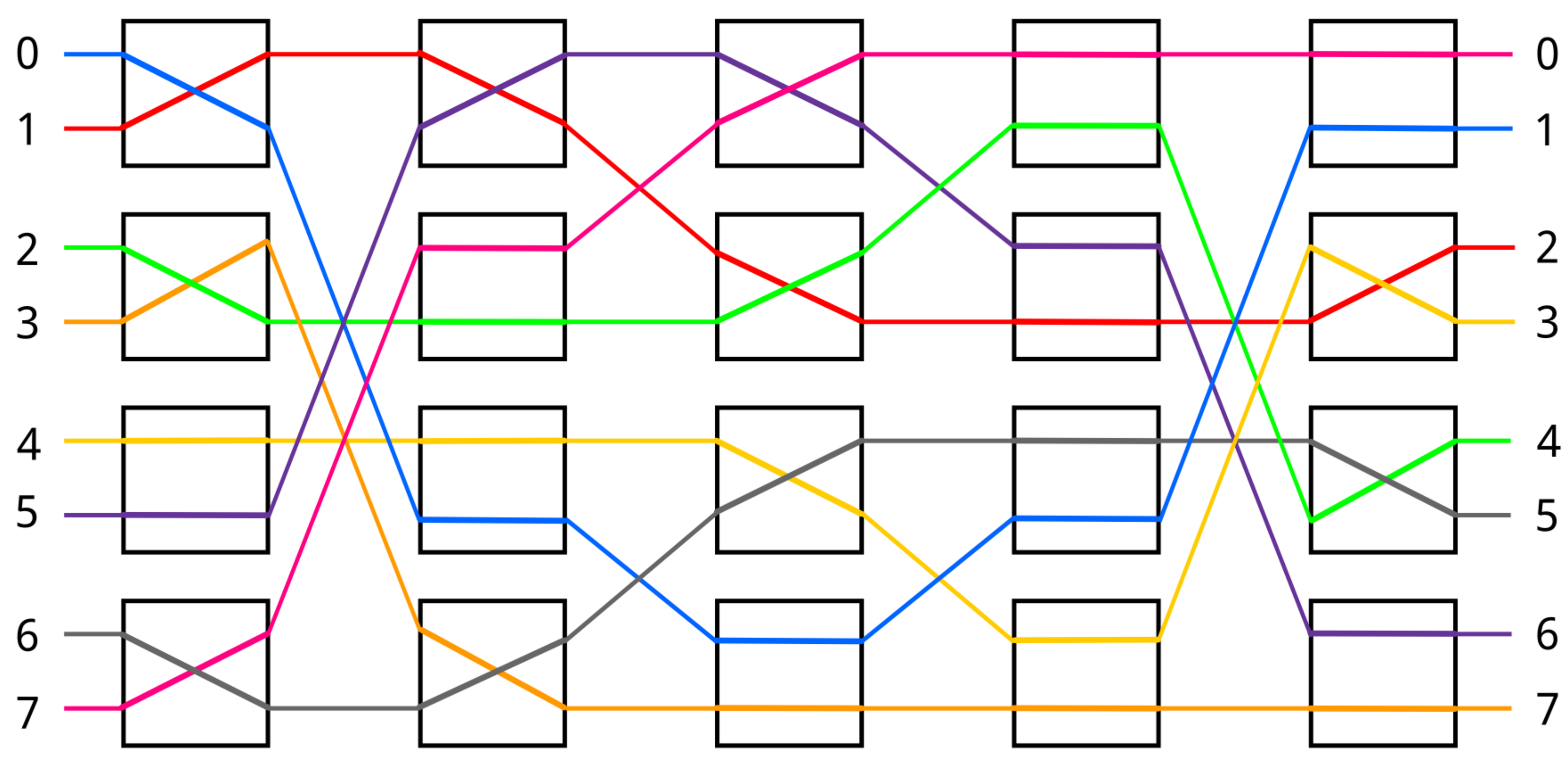
In general-purpose parallel architectures, multiple processors are connected together by a network of interconnects. These connected processors can then send packets of data across the network to other processors in the network that might require the data. During this communication, if two packets of data require use of the same interconnect at the same time, a collision in the network occurs. In this situation, one of the packets has to wait for the interconnect to become vacant again before it can continue its transmission across the network. This increases the network latency of the packet, and reduces the efficiency of the communication. My project aims to find a solution to this reduced efficiency caused by data collisions in the interconnection network.





I have created an emulation of a general-purpose parallel architecture, including multiple processors, the interconnection network and the routing algorithm. The emulation takes a parallel program as input, distributes the relevant code sections in the program to each processor, then executes the program in emulated parallel. The output of the emulation includes the total number of packets sent across the network, the emulated run time of the program, and the total number of collisions that occurred in the network.

The number of collisions in the network is reduced by my developed routing algorithm. This algorithm uses the characteristics of the Beneš topology used in the network to calculate collision-free routes for full and partial permutations of communication. It also aims to reduce the path length for each packet and distribute the packets evenly across the network, reducing the possibility of collisions between packets belonging to different permutations. This leads to fewer collisions occurring in the network, and increased communication efficiency.



Department of Computer Science

University of BRISTOL