

## Proposed Project Topic: Multi-Node Programming

### Group Members:

Henry Fu (hwfu)

Yeow Cheng Ong (ycong)

Harn Hua Ng (harnhua)

### Overview

In this project, methods for mapping stream programs over multiple stream processing nodes are developed and evaluated. Specifically, these methods are used to partition data and/or instructions across the nodes, and communicate data/state information to coordinate the processors. The example chosen for this project is that of IP Packet Routing.

### Metric

Execution time of a single-Stream Processor configuration is compared against that of a multi-node configuration.

### Setup

Simulation is done with the idebug simulator using the existing Imagine StreamC and KernelC development tools. The definition of a node is shown in Figure 1 below:

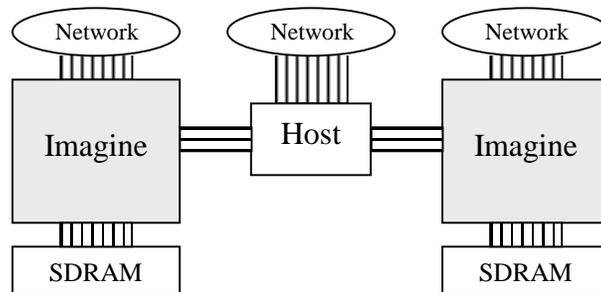
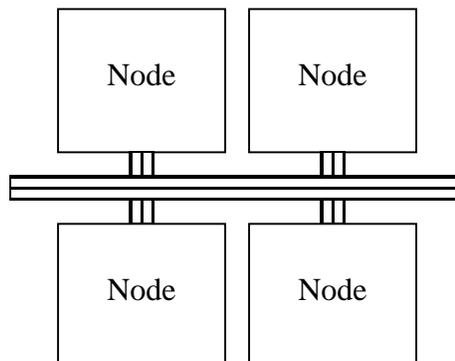


Figure 1: A Node (1 Host Processor + 2 Imagine Stream Processors)

Four of these nodes are linked to form a basic multi-node configuration block, as shown in Figure 2 below:



Basic Multi-Node Configuration

## Experiment

Based on the functionality of IP Routing, addressing and error checking information are extracted from each packet, compared against a table of existing values, and re-routed to an appropriate destination address.

The three main steps are:

- Error Checking based on CRC checksum
- Table lookup – longest prefix matching against table of values stored in memory
- Next Hop Address assignment and insertion into packets

Data stream in this example is represented by the packet traffic.

The same application is run on a single Imagine processor configuration and on several multi-node configurations, and the execution times will be recorded. Let  $N$  be the number of nodes used in a multi-node configuration, and  $S$  be the speedup in execution time, as compared to that on a single Imagine processor configuration.

## Example of Method for Load Balancing

3 nodes (3 hosts and 6 Imagine processors) are used to perform the table lookup, while (1 host and 2 Imagine processors) is used for error checking and assignment of the next hop address.

- Lookup table is split into 3, each given to a node. (*data distribution*)
- Packet traffic is split into 3 streams in round-robin fashion, and each stream is then distributed to each node. (*data distribution*)

There are in total 6 lookups at a time, since one node can perform 2 lookups. After each lookup, the Imagine processor has to pass the longest match result, along with the current packet to the neighboring processor of another node to continue the longest match search.

- After passing through 3 Imagine processors of 3 different nodes, the longest match is found and the result is sent to the last node for error checking and next hop address changing. (*instruction distribution*)

## Tentative Schedule

5/14 Set up multi-node configuration in simulation environment.

Update of progress in class.

Brook assignment due.

5/21 Meet with TAs or Prof. Dally for progress update and evaluation of mapping methods.

5/23 Functional IP Routing application in idebug for single-processor and multi-node configurations.

From this point onwards, run simulations for different values of  $N$ . Results are analyzed and methods re-evaluated.

6/4 Present results in write-up and oral presentation.