Optimal Data Intensive Flows for Network on Chip Mesh Networks (what is your opinion?)

Junwei Zhang*, Esther Arkin * and Thomas Robertazzi**,

*Dept. of Applied Mathematics and Statistics,

**Dept. of Electrical and Computer Engineering,

Stony Brook University,

Stony Brook NY 11794

Junwei.Zhang@stonybrook.edu

1.0 Background

Networks on chips (NOC) represent the smallest networks that have been implemented to date [refs including my recent book Introduction to Computer Networking] . A popular choice for the interconnection network on such networks on chips is the rectangular mesh. It is straightforward to implement and is a natural choice for a planar chip layout.

Data to be processed can be inserted into the chip at one or more so-called "injection points", that is node(s) in the mesh that forward the data to other nodes. Beyond NOCs, injecting data into a parallel processor's interconnection network has been done for some time, notably in IBM's Bluegene machines [ref].

In this paper it is sought to determine, for a given set of injection points how, optimally or near-optimally, to assign load to different processors in a known timed pattern so as to process a load of data in a minimal amount of time (i.e. minimize makespan). In this paper we succeed in presenting an optimal technique for single injection points in homogeneous meshes that involves no more complexity than linear equation solution. For multiple injection points we present algorithms that produce near optimal solutions using Voroni diagrams. The methodology presented here can be applied to a variety of switching/scheduling protocols besides those directly covered in this paper.

2.0 Approach

2.1 Divisible Load Theory

Crucial to our success in the single and multiple injection point cases, is the use of divisible load scheduling theory [refs]. Developed over the past few decades, it assumes load is a continuous variable that can be arbitrarily partitioned among processors and links in a network. Use is made of the divisible load scheduling's optimality principle [ref], which say makespan is minimized when one forces all processors to stop at the same time (intuitively otherwise one could transfer load from busy to idle processors to achieve a better solution). This leads to a series of chained linear flow and processing equations that can be solved by linear equation techniques, often yielding recursive and even closed form solutions for quantities such as makespan and speedup.

2.2 Voroni Diagrams

In the context of multiple injection point models, this paper represents the first use of Voroni diagrams [refs] in conjunction with divisible load scheduling for a significant applied problem. [Give a paragraph of background on Voroni diagrams}].

3.0 Single Injection Point Case

4.0 Multiple Injection Point Case

5.0 Significance of Work

6.0 Related Work (maybe) – mention Indian paper

7.0 Conclusion