ABSTRACT

Recent design trends in computer architectures have deviated from the single-core concept, where performance is achieved by parallelizing instruction execution or by executing a handful of threads, to the multi-core concept, where many simpler cores which co-exist on the same silicon surface execute applications in a true parallel manner. This new state-of-the-art architecture design paradigm was sparked by the need of harnessing the additional processing power offered by the increasing number of available on-chip transistors, in excess of a billion transistors on a single chip, available nowadays. In doing so, not only true and accelerated application execution can be achieved, but problems such as excessive power consumption ease in design and verification, the limitation of thermal envelopes, can be achieved as well. Hence multicore chips such as Chip Multi-Processors (CMPs) such as Intel's Single-chip Cloud Computer (SCC) and Multi-Processor Systems-on-Chips (MPSoCs) have already been realized.

Unfortunately, however, CMOS technology down-scaling which has led to increasing transistor densities, has also made transistors operate unreliably, making them increasingly prone to breakdown and eventual failures. The same trend occurs in on-chip wires which interconnect tiles in a multi-processor such as a CMP. Physical effects such as electro-migration can break these wires apart, hence the on-chip interconnect which comes in the form a Network-on-Chip (NoC), a packet-based communication on-chip fabric, can become disconnected at various parts in its originally-designed topology. If no intelligent routing algorithm exists, which can bypass these hop-to-hop disconnections, in a 2D mesh, packets can no longer be delivered to their destinations, and as a result they may stall indefinitely in their router buffers, causing a protocol-induced global deadlock to cover the entire topology. Hence a CMP can no longer operate, and becomes completely unusable.

Even a single link failure may disable the entire on-chip interconnect system with the use of a routing algorithm which is oblivious to link faults. To alleviate this critical problem, in this Thesis we propose an adaptive fault-tolerant routing algorithm, called Metis[1], that can deliver packets to their destinations in a highly-disconnected on-chip network environment. Metis works

[1] Due to the fact, that the proposed routing algorithm may handle any possible fault scenario by configuring a routing table, it was named Metis after the Greek Titan Metis who were the god of good counsel, advice, planning and wisdom.

in a recursive mode to discover paths that provide connectivity from any source-destination router pair in a 2D mesh NoC of any size. This connectivity information is stored in lookup tables found at every network router. It is able to create routing paths according to the current network state, i.e., the spatial distribution of faulty links in the network topology, which can take any form. Metis maintains full network communication connectivity as long as there is at least one routing path that connects two routers that request message exchange among them. This enables guaranteed message exchange between any pair of networks routers, albeit observed graceful performance degradation in terms of increasing latency and reduced throughput, as the number of faulty links increases.

In addition, Metis achieves load-balancing in network areas where no faulty links exist, utilizing 01TURN routing, combining the usage of two dimension-order routing algorithms simultaneously: XY and YX, where the former routes first along the X dimension and then along the Y dimension, while the latter executed the reverse routing order.

Metis was simulated under uniform random and transpose synthetic traffic patterns, using wormhole flow-control, in order to determine its performance and behavior, utilizing two spatial faulty link placement scenarios: (1) random, and (2) hotspot faulty link distributions. When compared against ARIADNE, an existing state-of-the-art fault-tolerant routing algorithm, Metis demonstrated up to 180.0 % and 266.67% improvement in throughput with a random faulty link placement, while it showed up to 220.0% and 137.5% increase in throughput with a hotspot faulty link placement, under uniform random and transpose traffic pattern usages, respectively. Metis was also tested using the Netrace benchmark suite demonstrating up to 38.71% improvement in network packet delivery latency when compared to ARIADNE. Finally, Metis was simulated under a lightly-loaded network to determine its basic routing delay under “no stress” conditions, while further experiments exhibit its superior throughput attainment just before network saturation.