

Efficient Fault-Tolerant Adaptive Routing under an Unconstrained Set of Node and Link Failures for Many-Core Systems-on-Chip

Michael Dimopoulos, Yi Gang, Mounir Benabdenbi, Lorena Anghel

Univ. Grenoble Alpes, TIMA, F-38031 Grenoble

CNRS, TIMA, F-38031 Grenoble

Grenoble, France

{firstname.lastname}@imag.fr

Abstract— An online fault tolerant routing algorithm for 2D Mesh Networks-on-Chip is presented in this work. It combines an adaptive routing algorithm with neighbor fault-awareness and a new traffic-balancing metric. To be able to cope with runtime permanent and temporary failures that may result in message corruption, message loss or deadlocks, the routing algorithm is enhanced with packet retransmission and a new message recovery scheme.

Simulation results, for various network sizes, different traffic patterns, under an unconstrained number of node and link faults, temporary and/or permanent, demonstrate the scalability and efficiency of the proposed algorithm to tolerate multiple failures likely encountered in deep submicron technologies. As the experiments have shown, the proposed algorithm maintains high reliability of more than 99.38% for a 2D mesh network of 16x16 and in the presence of 384 simultaneous link faults.

Keywords—*Networks on Chip; fault tolerant adaptive routing; congestion; permanent faults, transient faults; intermittent faults*

I. INTRODUCTION

We are in the many-core era. Chips like TILE-Gx72 [1] by Tilera, with 72 cores are highly complex systems on chip and need a fast and scalable infrastructure for supporting their communication demands. The Networks-on-chip (NoCs) [2] offer a promising scalable and modular solution to the increasing demands for higher speed and lower power.

As we move towards the late CMOS and beyond CMOS era the reliability of circuits is threatened [3] by the increased process-voltage and frequency variations. Under these hostile environments, the designs should be able to “adapt” and provide some functionality even in the presence of high failure rates.

The proposed solution, works by exploiting local information about the state of links and routers. It consists of an adaptive fault tolerant routing algorithm named CAFTA (Congestion Aware Fault Tolerant Algorithm) enhanced with neighbor fault-aware knowledge. In order to improve transmission latencies and to effectively guide the routing

decisions towards load-balanced network traffic, a new metric is introduced. In the presence of runtime errors, packet retransmission combined with novel message recovery mechanisms are utilized in order to provide fault tolerance under high failure rates. Extensive simulation experiments under various fault types (permanent, transient, intermittent) commonly encountered in advanced technology nodes demonstrate the effectiveness of the proposed fault tolerant scheme.

II. FAULT TOLERANT ROUTING WITH CAFTA

The basis for our router architecture is an input-buffered router with credit-based wormhole flow control. The routing algorithm is based on Variant B [4] with the following additions/extensions:

- Neighbor fault awareness: Each router knows the health status of its own links and of the links of its neighboring routers (routers at a distance one hop away).
- Packet-retransmission is used combined with a new recovery scheme.

A. Fault Model

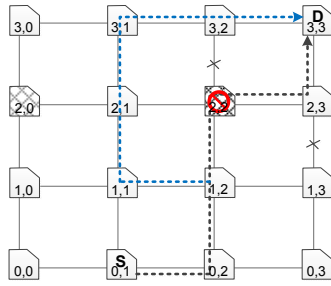
The links that connect the routers in a NoC are bi-directional (pair of two unidirectional links). Under the fault model we are using, a whole unidirectional link could be faulty. Thus, for a whole (bi-directional) link to fail, faults on both of its constituents unidirectional links should be present. Also, a node is considered faulty when all of its links are faulty. Faults can be permanent, transient, and intermittent.

In our fault tolerant scheme we make the following assumptions:

- The messages are protected by ECC. In case of error, flit retransmission occurs from the upstream router.
- Software / hardware Self-checking is used to detect faults at processing nodes, routers, links [3].

B. Packet recovery

During circuit operation, a runtime fault (permanent, transient, intermittent) may cause the corruption of the message, or lead to message deadlocks and eventually message loss. A k -retry-retransmission mechanism is employed at the router and for the output port as selected by the route computation stage (RC). This remedies cases where runtime faults, temporary in nature, lead to errors which are uncorrectable by the ECC mechanism (e.g. multi-bit error). For situations where the message propagation path towards the destination is blocked due to a permanent link or node failure, and after k unsuccessful retries (for our studies we have set $k=2$) the proposed adaptive fault-tolerant routing algorithm provides a solution by selecting another output port to route the message. Last, for the case of worm split, where an intermediate link/node in the path set by the header flit fails, the message (payload) cannot progress towards the destination. For such cases, a special mechanism is adopted here for message recovery termed *Split Message Recovery* or *SMR*. The aforementioned schemes will be better understood with the following example.



Node (2,0) is faulty, faulty links: $\{(1,3) \leftrightarrow (2,3)\}$, $\{(2,2) \leftrightarrow (3,2)\}$
A runtime failure appears in node (2,2)

Figure 1. Example of message recovery in the case of a worm split.

Let us consider the situation appearing in Fig. 1 where a message is routed from S (0,1) to D (3,3). The header flit has reached the destination D and the payload is guided to D via the path (0,1),(0,2),(1,2),(2,2),(2,3),(3,3). A run-time failure at node (2,2) splits the worm. As a result, the rest of the message cannot reach the destination D and eventually the transmission will fail (for a permanent failure). To resolve from such situation, a pseudo tail flit is created in router (2,3) and a timer is set to kI cycles (kI a user defined number). At the same time, in router (1,2) a second retransmission is attempted. If the output link is still unusable, a pseudo header flit is created in router (1,2) and is send towards the destination through an alternative path selected by the adaptive routing algorithm: $\{(0,1),(0,2),(1,2),(1,1),(2,1),(3,1),(3,2),(3,3)\}$. Furthermore, after kI cycles, the pseudo tail flit from router (2,3) is send to the destination (3,3) thus releasing the occupied resources.

III. EXPERIMENTAL SETUP AND RESULTS

CAFTA has been implemented in C++ on top of IRIS [5] a cycle-accurate network simulator and its performance has been studied under various 2D-mesh networks sizes, with different

traffic patterns and under various fault injection cases.

The fault patterns considered are permanent, transient and intermittent faults. More specifically, we have mixed type of fault injections. Five different fault injection ratios (fir) are investigated (see Table I) and for each fir the chosen network size was simulated for 5000 cycles. In the simulations, a warm-up period of 2000 cycles has been utilized.

In Figure 2 it is presented the percentage of packets delivered with respect to the failure rate (faulty link model) for three different NoC sizes and uniform random traffic. Let us consider for example the case of a 16x16 mesh NoC (see Fig. 2). Overall, CAFTA maintains high levels of reliability and succeeds in delivering more than 99.9% of the transmitted messages when the percentage of faulty links is up to 10%; i.e. 96 simultaneous link faults. The success delivery rate reduces to 99.38% when the percentage of faulty links increases to 40%; i.e. 384 simultaneous link faults.

REFERENCES

- [1] http://www.tilera.com/about_tilera/press-releases/
- [2] J. Duato, S. Yalamanchili, L. Ni, "Interconnection networks: an engineering approach", Morgan Kaufmann Publishers, 2003.
- [3] M. Nicolaidis, L. Anghel, N.-E. Zergainoh, Y. Zorian, T. Karnik, K. Bowman, J. Tschanz, Lu Shih-Lien, C. Tokunaga, A. Raychowdhury, M. Khellah, J. Kulkarni, V. De, D. Avresky, "Design for test and reliability in ultimate CMOS," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2012, pp. 677-682, March 2012.
- [4] M. Dimopoulos, Y. Gang, M. Benabdenbi, L. Anghel, N. Zergainoh, M. Nicolaidis, "Fault-tolerant adaptive routing under permanent and temporary failures for many-core systems-on-chip", IEEE 19th Int. On-Line Testing Symp. (IOLTS' 2013), Chania, Greece, pp. 7-12, 2013.
- [5] Manifold. <http://manifold.gatech.edu>.

TABLE I. SIMULATED NETWORK CONFIGURATION

Topology (Mesh) size	10x10, 12x12, 16x16
Traffic pattern	Uniform random
Flit size (in bits)	128
Arbitration	Round-robin
Router latency (cycles)	4
VCs/port	4/2 per VN (2 VNs in total)
Buffer size / packet size (in flits)	8/10
Fault injection ratios (faults/link/cycle)	0, 0.027, 0.053, 0.078, 0.10
Total simulation Time (cycles)	7000
Warm-up time (cycles)	2000
Iterations/fault injection pattern	1000

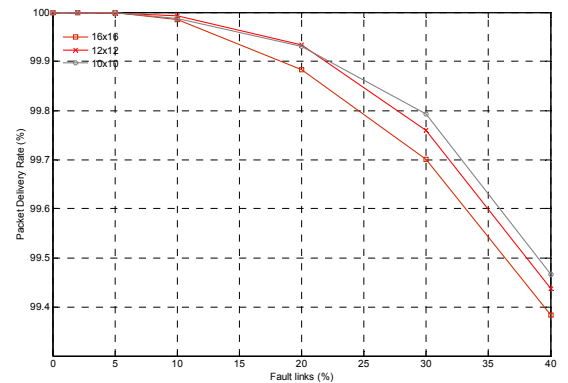


Figure 2. Packet delivery rate for a 2D mesh and uniform random traffic.