NARCO: Neighbor Aware Turn Model-Based Fault Tolerant Routing for NoCs

Yong Zou, *Student Member, IEEE*, and Sudeep Pasricha, *Member, IEEE*

***Abstract*—Network-on-chip (NoC) communication architectures are increasingly being used today to interconnect cores on-chip multiprocessor (CMP) based embedded systems. Permanent faults in NoCs due to fabrication challenges in sub-65 nm CMOS technologies and due to wearout underscore the need for fault tolerant design. In this letter, we propose a novel low-overhead neighbor aware, turn model based fault tolerant routing scheme (NARCO) for NoCs which combines threshold-based replication in network interfaces, a parameterizable region-based neighbor awareness in routers, and the odd–even and inverted odd–even turn models. The proposed scheme enables better packet arrival rate than state of the art, while enabling a tradeoff between communication reliability and energy overhead.**

***Index Terms*—Fault tolerant routing, networks-on-chip.**

I. INTRODUCTION

As CMOS technology aggressively scales below 65 nm and application complexity grows by leaps and bounds, next generation embedded processors are being driven to integrate multiple cores per chip. Reliability concerns in sub-65 nm nodes have in part contributed to the shift from traditional bus-based communication fabrics to network-on-chip (NoC) architectures that provide better predictability, scalability, performance, and utilization than buses [1]. The inherent redundancy in NoCs due to multiple paths between packet sources and sinks can greatly improve communication fault resiliency. To overcome permanent faults in NoC links and routers due to manufacturing defects, or after irreversible wearout damage due to electromigration in conductors, negative bias temperature instability (NBTI), or dielectric breakdown, fault tolerant routing schemes can ensure error free data packet delivery by using an alternate route that is free of faults. These schemes have been the focus of several research efforts over the last few years. In general, fault tolerant routing schemes can be broadly classified into three categories: stochastic, fully adaptive, and partially adaptive. *Stochastic* routing algorithms provide fault tolerance through data redundancy by probabilistically replicating packets multiple times and sending them over different routes [2]. Examples of such schemes include probabilistic gossip flooding scheme, directed flooding, and N-random walk [2], [3]. The major challenges with these approaches are their high-energy consumption, strong likelihood of deadlock and livelock, and poor performance even at low traffic congestion levels. *Fully adaptive* routing schemes make use of routing tables in every router or network interface (NI) to reflect the runtime state of the NoC and periodically update the tables when link or router failures occur to aid in adapting the flit path [4], [5]. However, these schemes have several drawbacks, including: 1) the need for frequent global fault and route information updates that can take thousands of cycles at runtime during which time the NoC is in an unstable state; 2) lack of scalability—table sizes increase rapidly with NoC size, increasing router (or NI) area, energy, and latency; and 3) strong possibility of deadlock, unless high overhead deadlock recovery mechanisms such as escape channels are used. *Partially adaptive* routing schemes enable a limited degree of adaptivity, placing various restrictions and requirements on routing around faulty nodes, primarily to avoid deadlocks. Turn model-based routing algorithms such as negative–first, north–last, and south–last [6]–[8] are examples of partially adaptive routing, where certain flit turns are forbidden to avoid deadlock. A recent work [9] combines the XY and YX schemes to achieve fault resilient transfers. However, the degree of adaptivity provided by these routing algorithms is highly unbalanced across the network, which in some cases results in poor performance. A few works have proposed using convex or rectangular fault regions with turn models [10], [11]. Such region-based fault tolerant routing schemes that make use of fault regions are generally too conservative (e.g., disabling fully functional routers to meet region shape requirements), have restrictions on the locations of the faults that can be bypassed, and also cannot adapt to runtime permanent faults.

In this letter, we propose a novel low-overhead neighbor aware turn model-based fault tolerant routing scheme (NARCO) for 2-D mesh-based NoCs. NARCO utilizes threshold-based replication which balances energy dissipation with the need to replicate packets to bypass permanent faults in routes. A parameterizable neighbor awareness in routers ensures a region-based awareness of faults that can improve routing decisions. Finally, a combination of odd–even (OE) and inverted odd–even (IOE)-based turn models ensures deadlock free and balanced packet delivery on routes that can pass around faults on the communication links. Our experimental results and comparison studies in Section III indicate that the proposed fault tolerant routing scheme outperforms existing turn model, stochastic random walk, and dual virtual channel-based routing schemes that have been proposed in literature.

II. NARCO: FAULT TOLERANT ROUTING OVERVIEW

In this section, we present an overview of the NARCO fault tolerant routing scheme for 2-D mesh NoCs. Section II-A

Manuscript received June 18, 2010; accepted July 25, 2010. Date of publication August 03, 2010; date of current version September 17, 2010. This manuscript was recommended for publication by D. Atienza.

The authors are with the Department of Electrical and Computer Engineering, Colorado State University, Fort Collins, CO 80523 USA (e-mail: yong.zou@colostate.edu; sudeep@colostate.edu).

Digital Object Identifier 10.1109/LES.2010.2063415

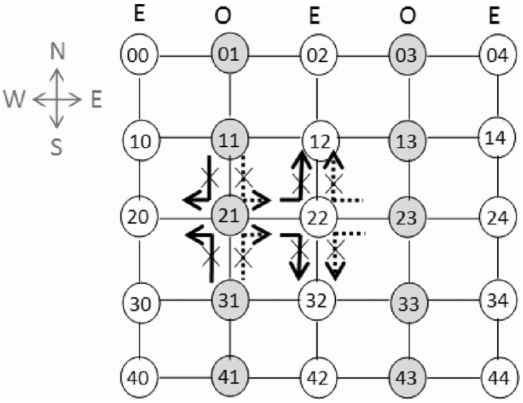


Fig. 1. OE and IOE prohibited turns based on column.

describes the OE turn and IOE turn models. Section II-B presents details of the implementation and operation of the proposed routing scheme that combines OE and IOE turn models. Section II-C describes how neighbor awareness can be used to make more informed routing decision.

A. OE and IOE Turn Models

The OE turn model for deadlock-free routing in 2-D meshes was introduced by Chiu [12]. Unlike traditional turn models that avoid deadlock by prohibiting certain turns, the OE turn model restricts the locations at which certain turns can occur to ensure that a circular wait does not occur. In OE turn model-based routing, columns in a 2-D mesh are alternately designated as odd (O) and even (E), as shown in Fig. 1 for a 5×5 2-D mesh. Then the following two main rules ensure deadlock free-routing in the OE turn model: *1) a packet is not allowed to take an EN or ES turn at any node located in an even column; and 2) a packet is not allowed to take a NW or SW turn at any node located in an odd column*. The IOE turn model can be understood by rotating the mesh by 180 degrees, while preserving the OE column designations and corresponding prohibited turns. Fig. 1 depicts restricted turns for the OE turn model with solid arrows and the restricted turns for the IOE turn model with dashed arrows.

B. Routing Algorithm Implementation Details

To ensure robustness against faults, redundancy is a necessary requirement, especially for environments with high fault rates and unpredictable fault distributions. As the level of redundancy is increased, system reliability improves as a general rule. However, redundancy also detrimentally impacts other design objectives such as energy and performance. Therefore, in practice it is important to limit redundancy to achieve a reasonable tradeoff between reliability, energy, and performance. Unlike directed and random probabilistic flooding algorithms that propagate multiple copies of a packet to achieve fault tolerant routing in NoCs, NARCO transmits only one redundant packet for each transmitted packet, and only if the fault rate is above a replication threshold δ. The original packet is sent using the OE turn model while the redundant packet is propagated using an IOE turn model scheme. This packet replication happens only at the source network interface. Two separate virtual channels (VCs), one dedicated to the OE packets and the other for the IOE packets ensure deadlock freedom. If the fault rate is low (i.e., below δ), replication is not utilized to save energy. The proposed routing algorithm prioritizes minimal paths that have higher probabilities of reaching the destination. Minimal paths and the replication threshold ensure low energy consumption under a diverse set of fault rates and distributions. The routers detect which of their adjacent (neighbor) links/nodes are faulty based on control signals. The proposed routing approach can be combined with error control coding (ECC) techniques for transient fault resiliency and optimizations such as router buffer reordering and router/NI backup paths to create a comprehensive fault tolerant NoC fabric for future CMP architectures.

Pseudocode: Turn Restrictions for OE Implementation

```
RESULT   check_neighbor(UI ip_dir, UI direct, ULL dest_id) {
   //ip_dir: direction where the packet comes from
   //direct: direction we want to check
   //cur_id: current node position
   //dest_id: destination node position.
   //dest_yco,dest_xco: y,x coordinates of destination node
   //cur_yco,cur_xco: y,x coordinates of current node
   //dif_yco=dest_yco-cur_yco
   //RESULT: LINK_FAULT, TURN_FAULT, BACK_TURN, OK
    if (check_linkfault(direct)==LINK_FAULT ){
            return LINK_FAULT;
    }
    if ((cur_yco%2==0)&&(ip_dir==E)&&(direct==N||direct==S)){) {
            return TURN_FAULT;
    }
    if ((cur_yco%2==1)&&(ip_dir==N||ip_dir==S)&&(direct==W)) {
            return TURN_FAULT;
    }
    if (direct==E) {
        if ((dest_xco!=cur_xco)&&(dif_yco==1)&&(dest_yco%2==0)) {
            return TURN_FAULT;
        }
        if (dest_yco<=cur_yco) {
            return TURN_FAULT;
        }
    }
    if ((dif_yco==1)&&(dest_yco%2==0)) {
        if ((dest_xco<cur_xco)&&(direct==S)) {
            return TURN_FAULT;
        } else if ((dest_xco>cur_xco)&&(direct==N)) {
            return TURN_FAULT;
        } else if ((dest_xco==cur_xco)&&(direct==N|| direct==S|| direct==W)) {
            return TURN_FAULT;
        }
    }
    if ((dest_yco<cur_yco)&&(cur_yco%2!=0)&&((direct==N)||(direct==S)))
            return TURN_FAULT;
    if (((borderW(cur_id))||(cur_yco%2!=0))&&(cur_yco==dest_yco)){
        if ((cur_xco<dest_xco)&&(direct==N)){
            return TURN_FAULT;
        } else if ((cur_xco>dest_xco)&&(direct==S)){
            return TURN_FAULT;
        }
    }
    if (ip_dir==direct){
            return BACK_TURN;
    }
            return OK ;
}
```

Fig. 2. SystemC pseudocode for OE turn restrictions.

Fig. 2 shows the SystemC pseudocode of the checks for the OE turn model in our router model to aid in detecting an invalid routing direction. The procedure is invoked in parallel for all directions in a router node. Note that the pseudocode for the IOE scheme is similar, and is not shown here for brevity. A high-level overview of the pseudocode is as follows. We first check whether the output port direction has a fault in its attached

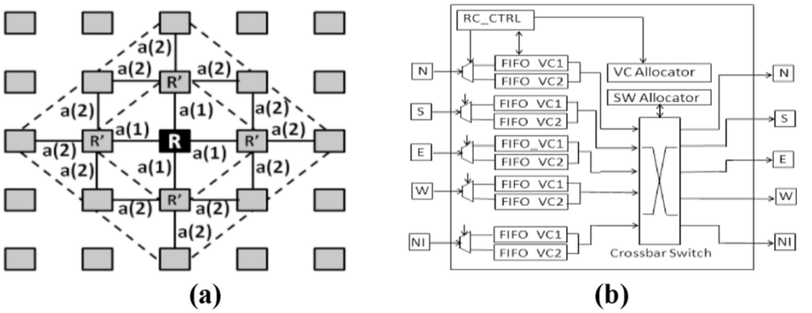


Fig. 3. (a) Neighborhood awareness and (b) router architecture.

adjacent link (Steps 2–4), in which case this is an invalid direction. Next, we check the restricted turn rules for the turn models as discussed in Section A (Steps 5–10) based on the router location (in an odd or even column), the input port of the packet, and its output port direction. If the packet is attempting a forbidden turn, then the direction is invalid. If the direction has no adjacent faults and does not violate the basic OE routing rules, we then check if the direction will lead to a turn rule violation downstream based on the location of the destination (Steps 11–36). Finally, we check for a back turn, which is not allowed (Steps 37–39). If all these checks pass, then the given direction is valid for packet transfer (Step 40).

After the turn restriction checks, it is possible for a packet to have multiple valid directions that it can follow from its current location. To ensure low-energy transfers, directions in minimal paths are always given higher priority in our scheme. If, however, two valid directions in minimal paths exist to the destination from the current router, we give higher priority to the direction that allows for greater path diversity so that the packet can still route around faults downstream. In NARCO, given two minimal paths to the destination, *we prioritize the North and South directions over the East and West directions*.

C. Neighbor Awareness and Router Architecture

The pseudocode for our implementation in the previous subsection assumed that each router knows the fault status of its adjacent links. To reduce the probability of encountering a fault downstream when forwarding a packet from a router, we incorporate neighbor awareness into our router. Fig. 3(a) shows the adjacent links (i.e., $\mathrm{adjacency} = 1$) for a router R with the links designated as $a(1)$. Neighbor awareness can be achieved with a small buffer in the router for the purpose of keeping the fault status in the router's neighborhood beyond its adjacent links. For instance, a buffer of size 16 bits is sufficient to keep a fault status of links with adjacency levels 1 $(a(1))$ and 2 $(a(2))$. The size of the neighborhood is customizable, but should not be too large as a large size entails a larger buffer size and a more complex fault status signaling network that can impact overall performance and energy consumption. The fault status buffer is initialized at startup and then periodically updated every 500 K cycles to account for runtime permanent faults that may (in practice infrequently) arise in the network.

Fig. 3(b) shows the architecture of our router with the fault tolerant routing and neighbor awareness incorporated in the route compute control (RC_CTRL) unit. The input FIFO buffers store up to 16 flits waiting to be forwarded to the next hop. There are two input FIFO buffers each dedicated to a VC, with one VC for the OE routed packets and the other for the IOE routed packets. To reduce energy consumption, if the fault rate is below the replication threshold δ, the RC_CTRL unit shuts down the IOE virtual channels in the router.

III. EXPERIMENTAL RESULTS

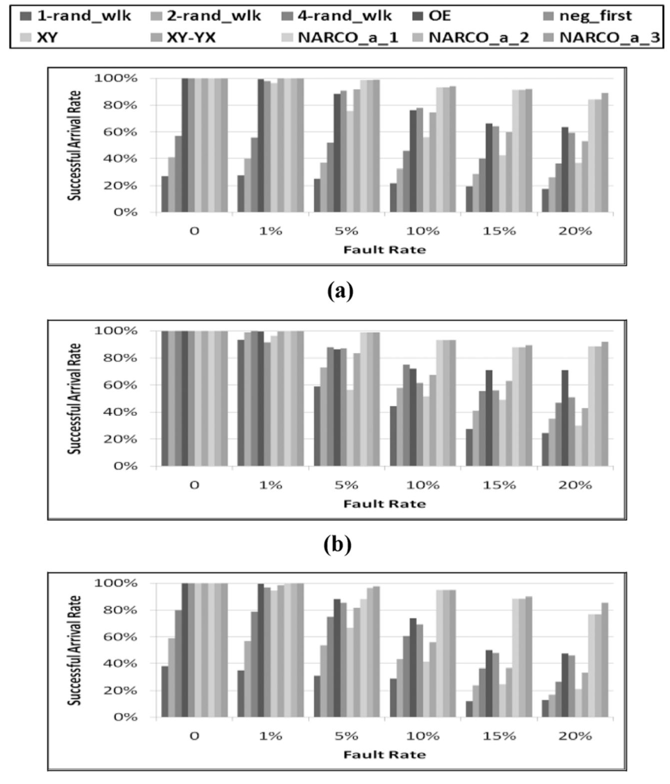
A. Experimental Setup

We extended the open source Nirgam NoC simulator [13] to support faults and different fault tolerant routing schemes, and incorporated dynamic and leakage power models for NoC components. For accurate power and delay analysis of the NARCO algorithm, we developed a gate level implementation of the algorithm and included delay and power results. A 2-D mesh of size 9×9 (81 cores) was considered in our analysis. The NoC fabric was clocked at a frequency of 1 GHz, with a CMOS implementation technology of 65 nm. Three different traffic models were analyzed: uniform random, transpose, and hotspot. Faults in the NoC were modeled as randomly distributed link-level hard failures that can represent design time or runtime failures. Ten different random fault distributions were generated and analyzed for each fault rate, and for each traffic type.

B. Comparison With Existing FT Routing Schemes

To demonstrate the effectiveness of the NARCO fault tolerant routing scheme for NoCs, we compared it to several existing fault tolerant routing schemes from literature. The following schemes were considered in the comparison study: 1) XY dimension order routing; 2) N-random walk for $\mathrm{N} = 1, 2, 4$ [3]; 3) negative first turn model [6]; 4) OE turn model [12]; 5) XY-YX which combines replication with the XY and YX routing schemes on two VCs [9]; and 6) our proposed NARCO scheme with neighborhood adjacency of 1, 2, and 3. A replication threshold value of $\delta = 6\%$ was chosen for the NARCO scheme, based on our extensive simulation studies that showed a good tradeoff between arrival rate and energy for $8\% \geq \delta \geq 2\%$. Results are shown for the three traffic types and a 20% injection rate (0.2 flits/node/cycle).

Fig. 4(a)–(c) compare the successful packet arrival rate for the various fault tolerant routing schemes under different fault rates. It can be seen that the configurations of our proposed fault tolerant routing scheme (NARCO_a_1, NARCO_a_2, NARCO_a_3) have a much higher successful packet arrival rate



(a)

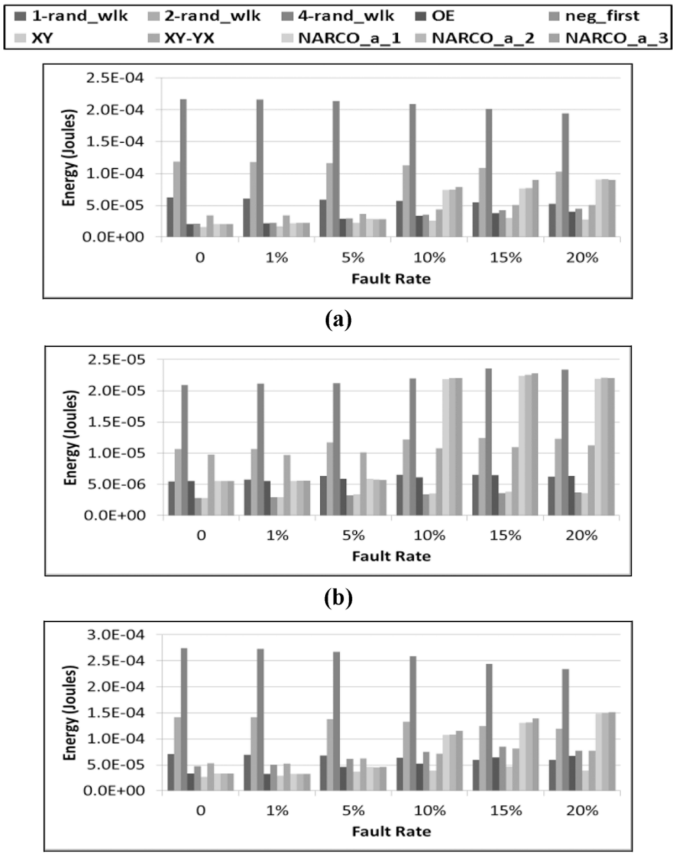
(b)

(c)

Fig. 4. Packet arrival rate for 9 × 9 NoC: (a) Uniform random traffic, (b) Hotspot traffic, and (c) Transpose traffic.

compared to the other schemes, especially for higher fault rates. In addition, having a higher neighbor awareness for high-fault rate environments ensures a better packet arrival rate.

Energy consumption for fault tolerant routing schemes is an important metric, especially in embedded systems often running on a fixed battery budget. Fig. 5(a)–(c) show the energy consumption of the NoC (in Joules) for the different routing schemes, under different fault rates. It can be seen that the N-random walk fault tolerant routing schemes have significantly higher energy consumption compared to the other schemes. This is due to the high level of replication with increasing values of N, as well as because of the non-minimal paths chosen to route the packets on. Single VC schemes (XY, negative first, OE) have a low-energy cost, but possess low fault tolerance (Fig. 4) making them not very viable for future CMPs. The XY-YX scheme makes use of two VCs, so it comes as no surprise that it has higher energy consumption than the single VC schemes. The extremely low-energy consumption of the NARCO configurations for low fault rates (comparable to the single VC schemes) is a strong motivation for the existence of the replication threshold parameter. For higher fault rates, the NARCO configurations have higher energy consumption than the XY-YX and single VC schemes, but this energy overhead ensures a significantly higher packet arrival rate compared to the other schemes, as can be seen from Fig. 4. Thus it can be seen that the proposed NARCO fault tolerant routing scheme not only provides a much higher fault tolerance compared to existing routing schemes, but also enables a tradeoff between reliability and energy consumption that can be tuned on a per-application basis by the designer by setting the replication threshold and neighbor awareness parameters.



(a)

(b)

(c)

Fig. 5. Communication energy for 9 × 9 NoC: (a) Uniform random traffic, (b) Hotspot traffic, and (c) Transpose traffic.

IV. CONCLUSION

In this letter, we propose a novel low-overhead NARCO for NoCs in embedded systems, which combines threshold-based replication, a parameterizable region based neighbor awareness in routers, and the OE and IOE turn models. The proposed scheme enables better packet arrival rate than state of the art, while enabling a tradeoff between communication reliability and energy overhead.

REFERENCES

[1] S. Pasricha and N. Dutt, *On-Chip Communication Architectures*. San Mateo, CA: Kauffman, Apr. 2008, ISBN 978-0-12-373892-9.

[2] H. Zhu, P. P. Pande, and C. Grecu, "Performance evaluation of adaptive routing algorithms for achieving fault tolerance in NoC fabrics," in *Proc. Int. Conf. Appl.-Specific Syst., Arch., Process. (ASAP)*, Seattle, WA, 2007, pp. 42–47.

[3] M. Pirretti, G. M. Link, R. R. Brooks, N. Vijaykrishnan, M. Kandemir, and M. J. Irwin, "Fault tolerant algorithms for network-on-chip interconnect," in *Proc. Annu. Symp. Very Large Scale Integr.*, Tampa, FL, 2004, pp. 46–51.

[4] Y. B. Kim and Y.-B. Kim, "Fault tolerant source routing for network-on-chip," in *Proc. Int. Symp. Defect Fault Tolerance VSLI Syst. (DFT)*, Rome, Italy, 2007, pp. 12–20.

[5] T. Schonwald, J. Zimmermann, O. Bringmann, and W. Rosenstiel, “Fully adaptive fault tolerant routing algorithm for network-on-chip architectures,” in *Proc. 10th Eur. Conf. Dig. Syst. Des.*, Luebeck, Germany, Aug. 2007, pp. 527–534.

[6] C. J. Glass and L. M. Ni, “The turn model for adaptive routing,” in *Proc. Annu. Symp. Comput. Arch. (ISCA)*, Gold Coast, Queensland, Australia, 1992, pp. 278–287.

[7] C. J. Glass and L. M. Ni, “Fault-tolerant wormhole routing in meshes without virtual channels,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 7, no. 6, pp. 620–635, Jun. 1996.

[8] C. M. Cunningham and D. R. Avresky, “Fault-tolerant adaptive routing for two-dimensional meshes,” in *Proc. IEEE. Symp. High-Perfom. Comput. Arch.*, Raleigh, NC, 1995, pp. 122–131.

[9] A. Patooghy and S. G. Miremadi, “XYX: A power & performance efficient fault-tolerant routing algorithm for network on chip,” in *Proc. Int. Conf. Parallel Process.*, Vienna, Austria, 2009, pp. 245–251.

[10] J. Wu, “A fault-tolerant and deadlock-free routing protocol in 2D meshes based on odd-even turn model,” *IEEE Trans. Comput.*, vol. 52, no. 9, pp. 1154–1169, Sep. 2003.

[11] A. Rezazadeh, M. Fathy, and A. Hassanzadeh, “If-cube3: An improved fault-tolerant routing algorithm to achieve less latency in NoCs,” in *Proc. IEEE Int. Adv. Comput. Conf. (IACC)*, Atlanta, GA, 2009, pp. 278–283.

[12] G.-M. Chiu, “The odd-even turn model for adaptive routing,” *IEEE Trans. Parallel Distrib. Syst.*, vol. 11, no. 7, pp. 729–738, Jul. 2000.

[13] *Nirgam Simulator*, [Online]. Available: http://nirgam.ecs.soton.ac.uk/