MICRO-46, 9th December- 2013
Davis, California

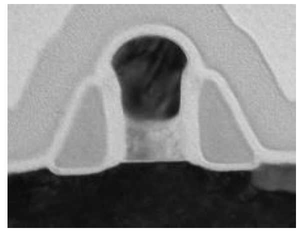
uDIREC: Unified Diagnosis and Reconfiguration for Frugal Bypass of NoC Faults

Ritesh Parikh and Valeria Bertacco

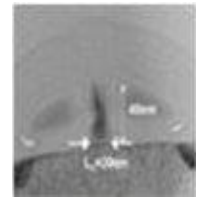
Electrical Engineering & Computer Science Department
The University of Michigan, Ann Arbor

Device Scaling and Processor Evolution

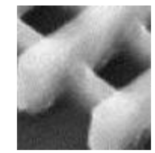
65nm



45nm



32nm



22nm

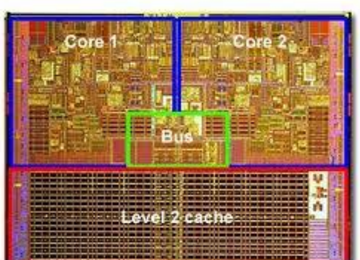
Shrinking transistor size

Transition to multicore

Waning reliability

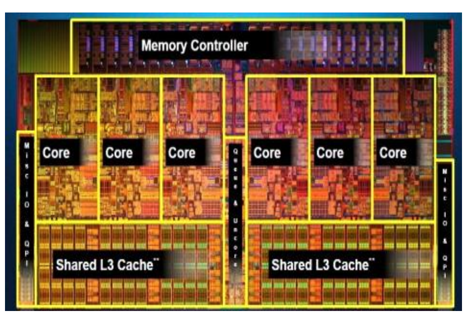
NoC based interconnect

Simple bus



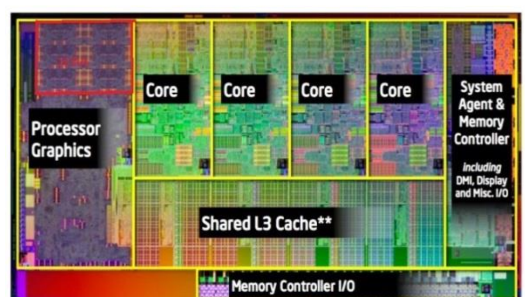
Core2Duo - 2007

Point-to-point bus



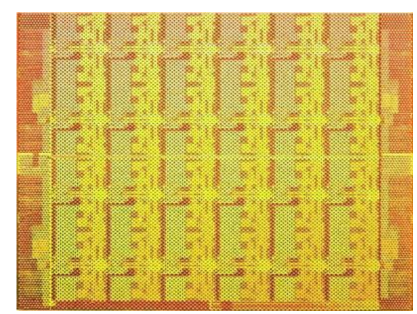
Intel Nehalem - 2008

High-speed ring network



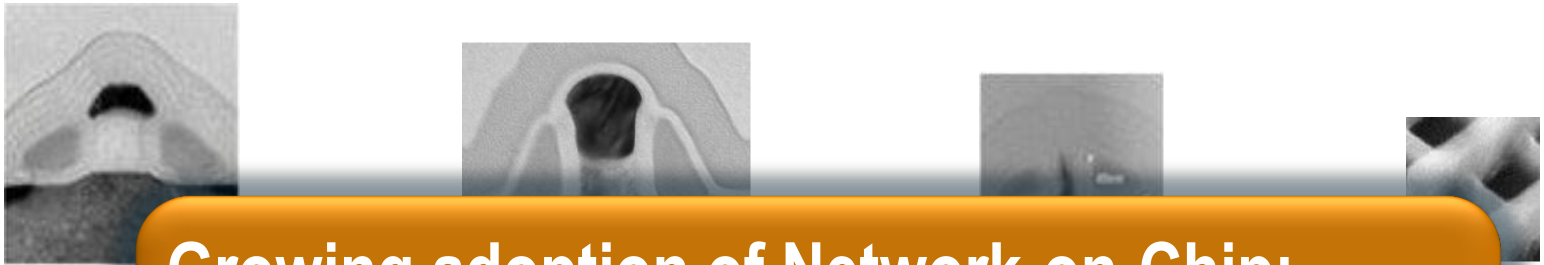
Intel Sandy Bridge - 2011

2D mesh network-on-chip



Intel SCC- 2009

Device Scaling and Processor Evolution



65nm

Growing adoption of Network-on-Chip:
- **Sole medium of on-chip communication**
- **Potential single-point of failure**

Shrinki

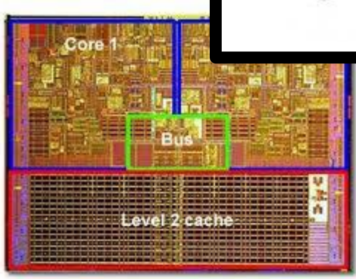
Transiti

NoC based interconnect

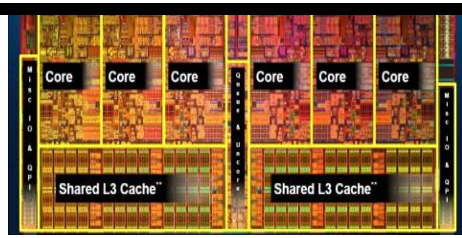
Intel Identifies Cougar Point Error, SATA Connection Single Point of Failure. January, 2011. Cost: $700 Million

Simple

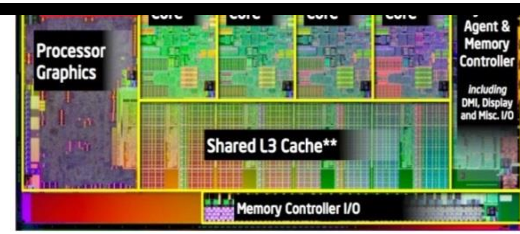
ork-on-chip



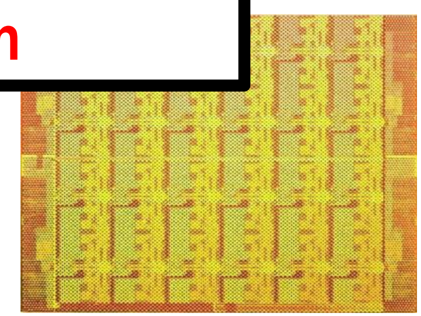
Core2Duo - 2007



Intel Nehalem - 2008

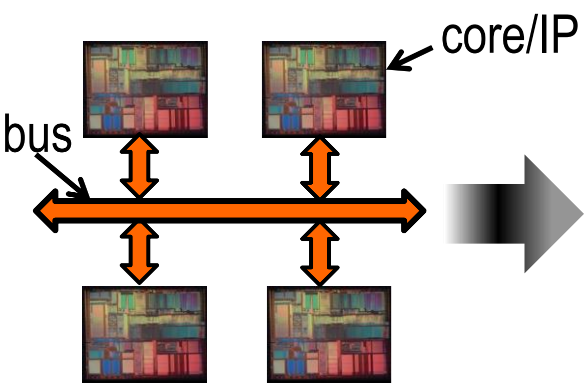


Intel Sandy Bridge - 2011

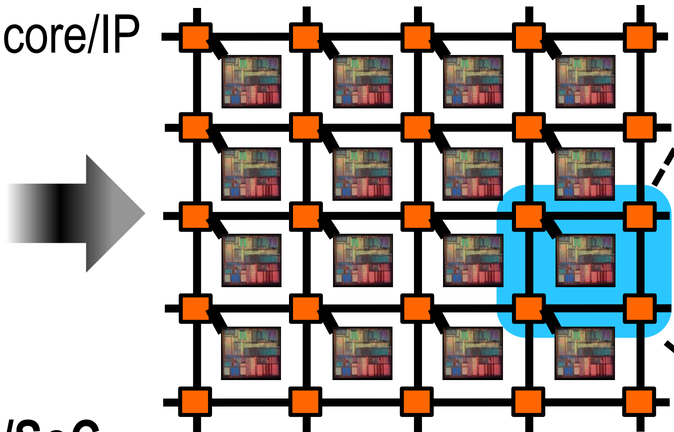


Intel SCC- 2009

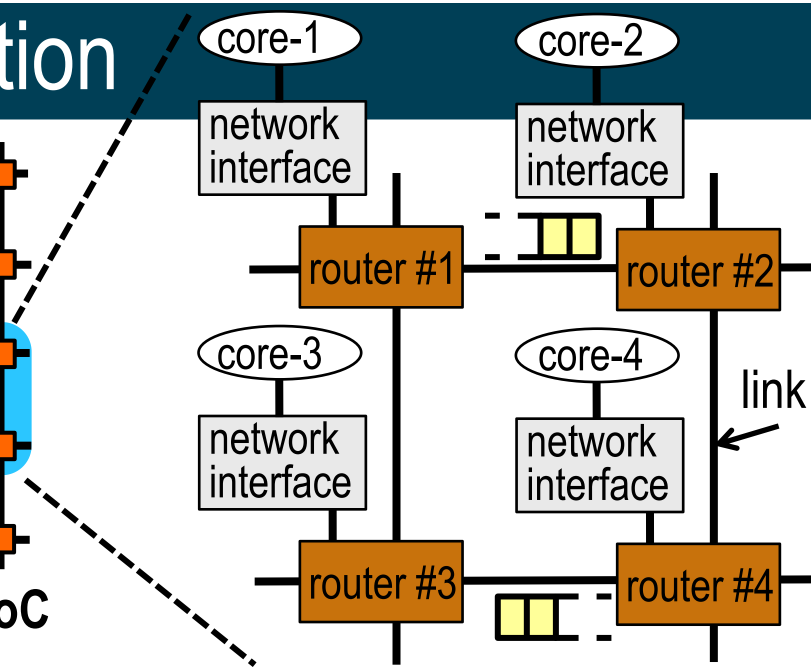
On-Chip Interconnect Evolution



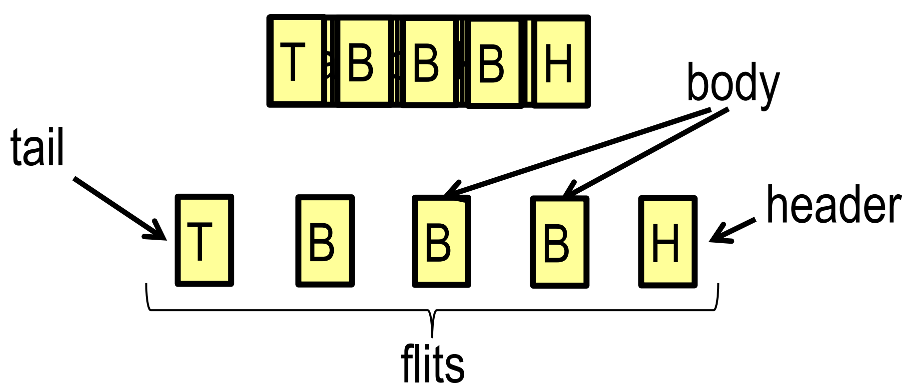
Bus-based multicore/SoC

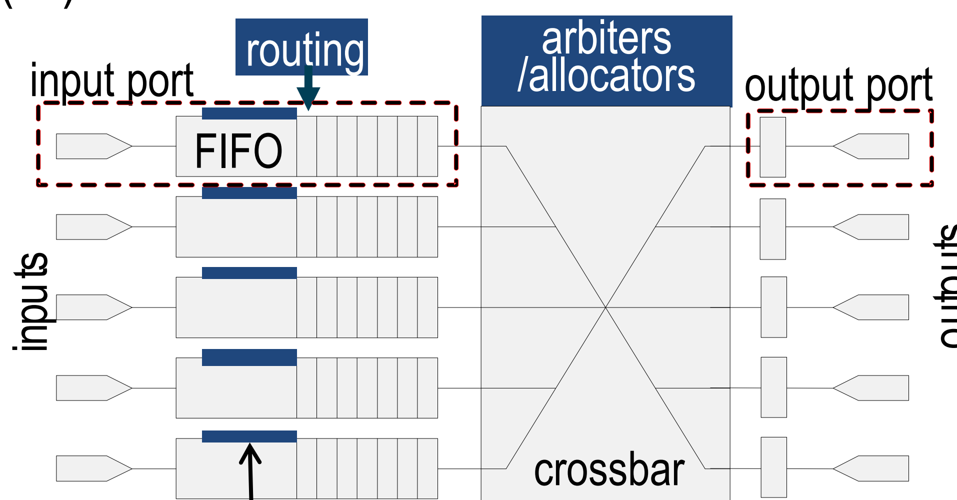


NoC-based multicore/SoC

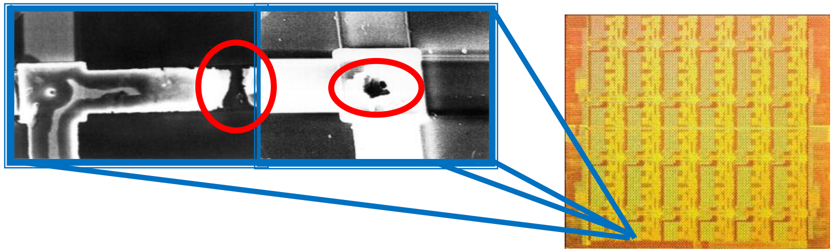


– Routers connected via links
– Cores connected via network interface (NI)





Permanent Faults in NoCs



— Permanent wear-out
— Device aging

NoCs: significant silicon footprint and heavy activity

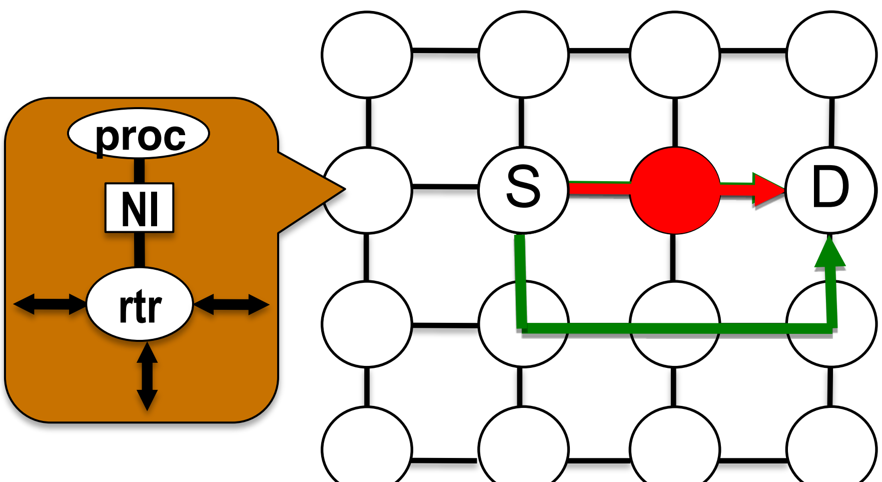
Detection → **Diagnosis** → **Reconfiguration** → **Recovery**

Detect if fault has occurred

Diagnose where fault has occurred

Reconfigure network to account for fault

Recover and resume normal operation



✗ cannot send on faulty path

✓ need to re-route around fault

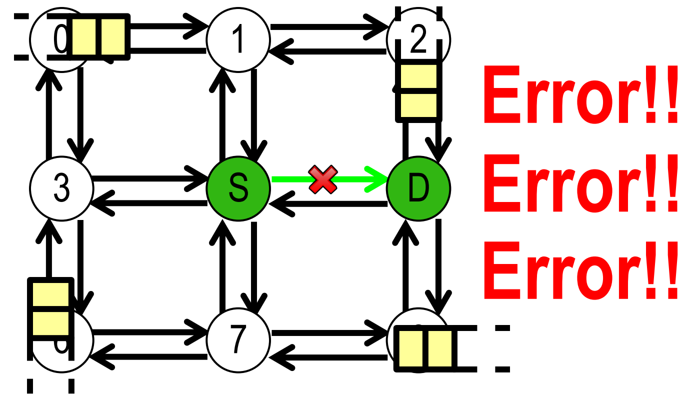
uDIREC solves two problems:

– **Fault diagnosis** at a fine resolution
– **Reconfiguration** to find new routes

Contributions

uDIREC (for **u**nified **DI**agnosis and **REC**onfiguration) incorporates:

- routing-aware scheme for diagnosis
- route-reconfiguration to circumvent faults



Fine-grain fault model for NoCs derived from:

- end-to-end diagnosis scheme
- frugal reconfiguration algorithm

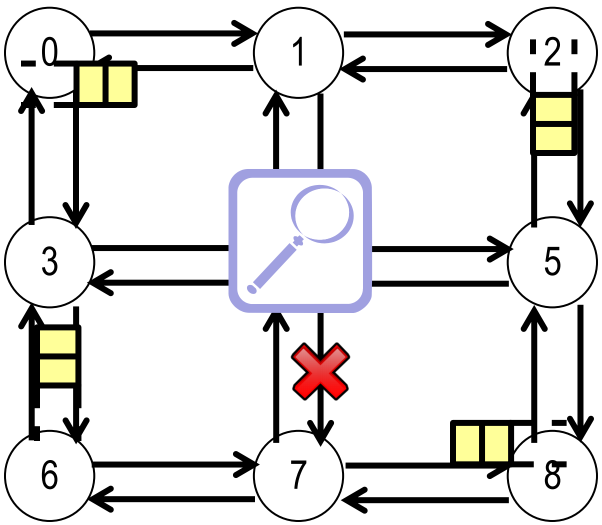
A deadlock-free routing algorithm for irregular networks with unidirectional links

Tightly integrated software implementation enables:

- low overhead
- no performance overhead during fault-free execution

Rest of This Talk

- ❑ **Prior Work**
- ❑ **Fine-Resolution Diagnosis**
- ❑ **Routing Algorithm**
- ❑ **Reconfiguration Algorithm**
- ❑ **Experimental Evaluation**

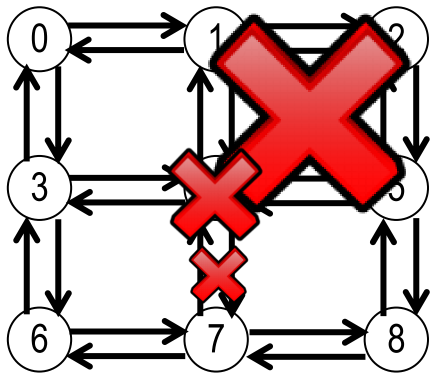


Prior Art

Permanent Fault Diagnosis

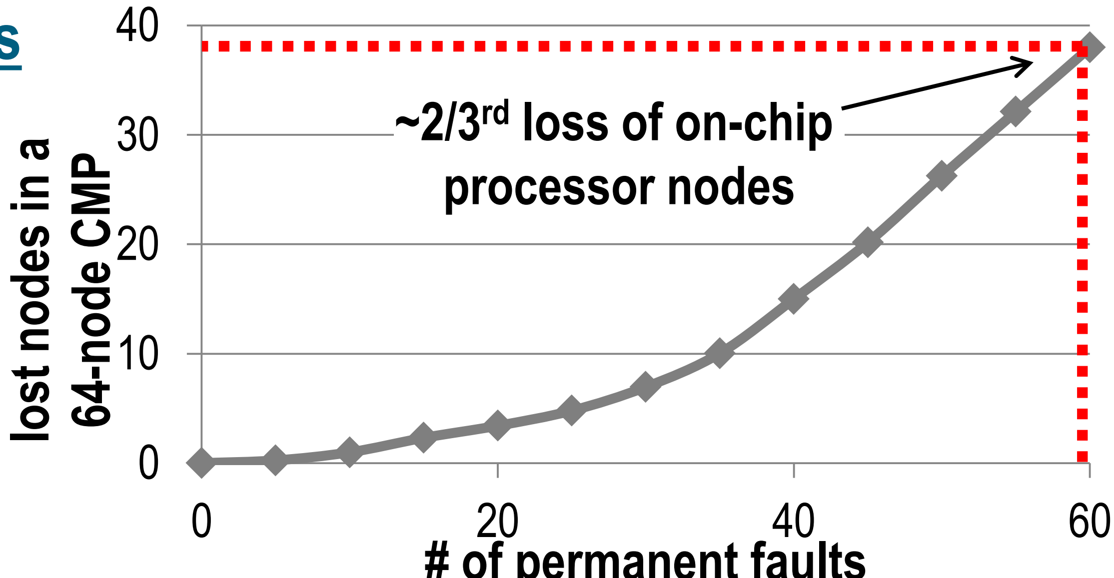
- Entire regions/routers [Puente'04]
- One or more bidirectional links [Fick'09]

Dedicated testing and high overhead



Reconfiguration around Faults

- Based on routing on irregular networks [Aisopos'11]
- Constrained by number and location of faults [Flich'07, Fukushima'09]



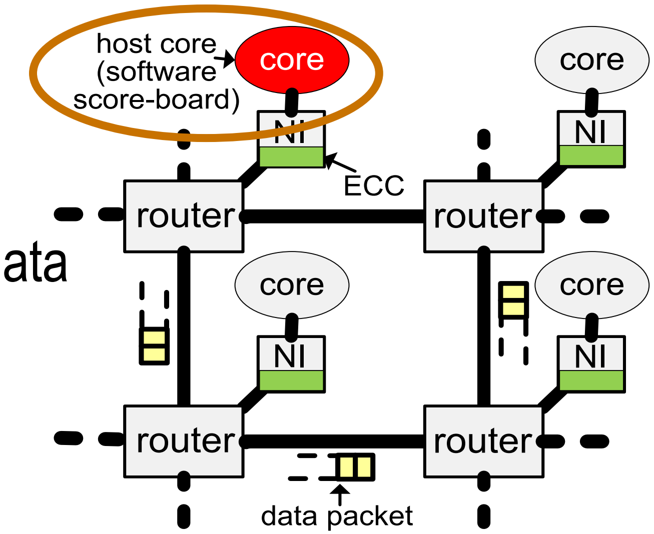
Detection and Diagnosis Mechanism

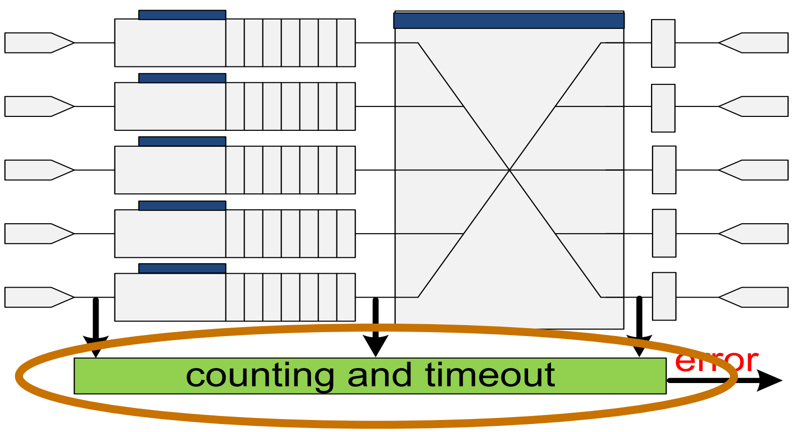
Inspired by Ghofrani et al., VTS'12

– Diagnose both datapath and control faults at low area overhead (< 3%)
– No runtime performance overhead when NoC is fault-free

Datapath faults

– End-to-end software based
– Scoreboard to collect symptoms of corrupted data





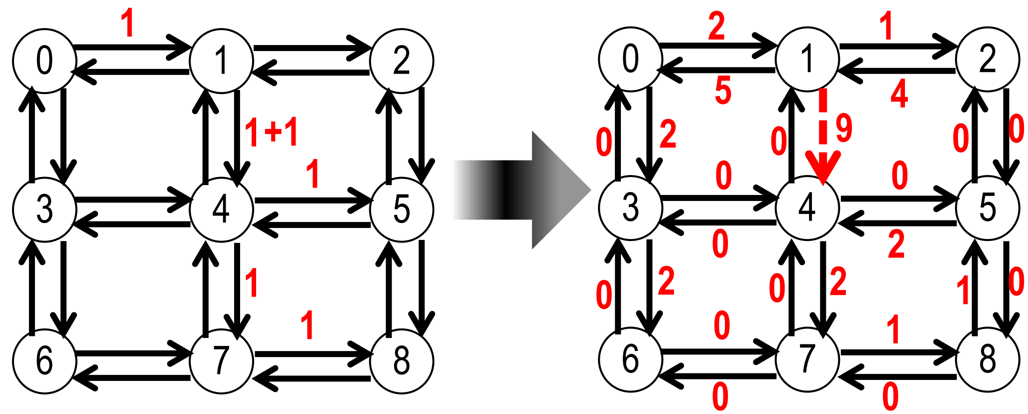
Control faults

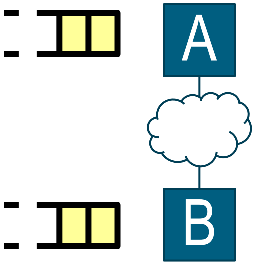
– Distributed hardware based
– Counting and timeout techniques

Fine-Resolution Fault Diagnosis

Packets are augmented with ECC [Shamshiri et al., ITC'11]

Erroneous transmission are reported to SW supervisor node





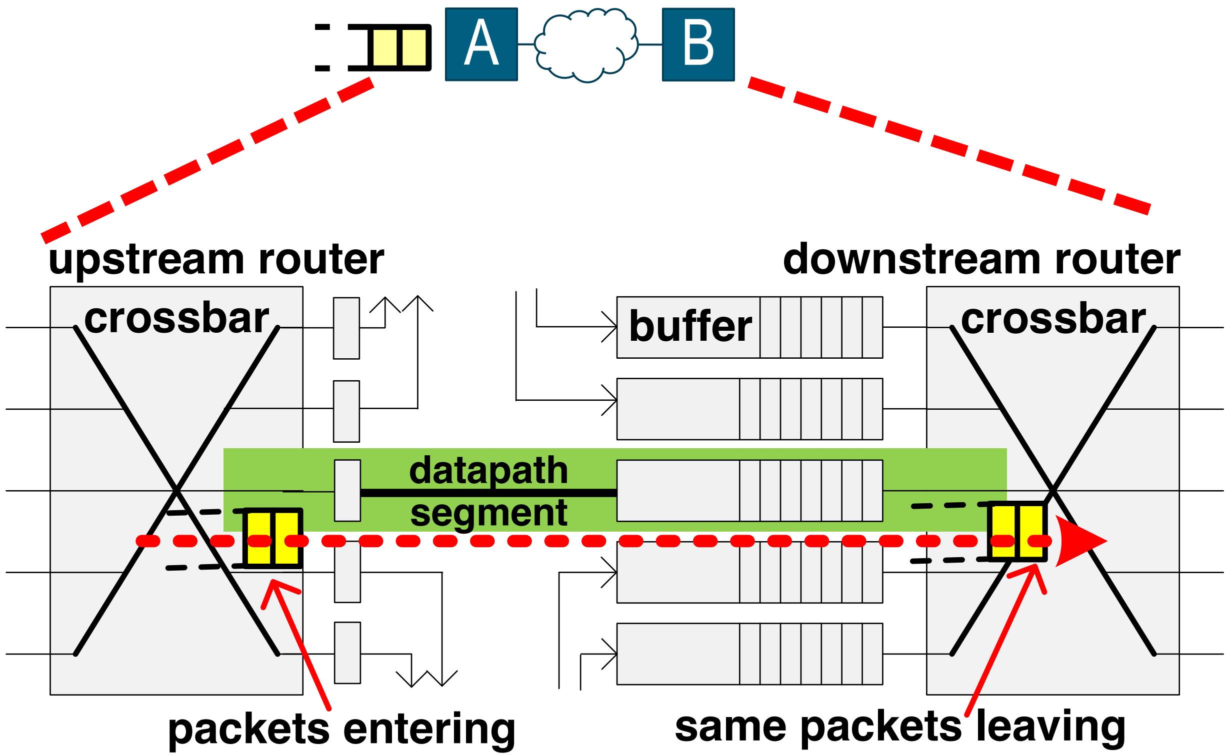
Differentiable from network ends

Undistinguishable from network ends

Finest granularity of:

- Routing reconfiguration
- End-to-end diagnosis

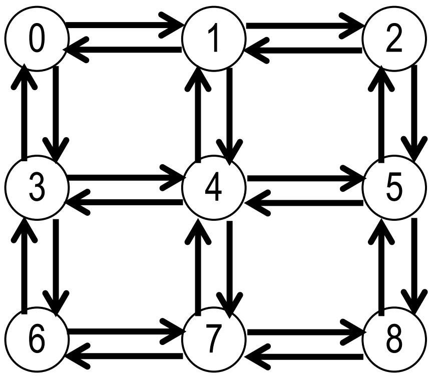
Fine-Grain Fault Model



Undistinguishable:
- O/P port
- Unidirectional link
- I/P port
- Crossbar contacts

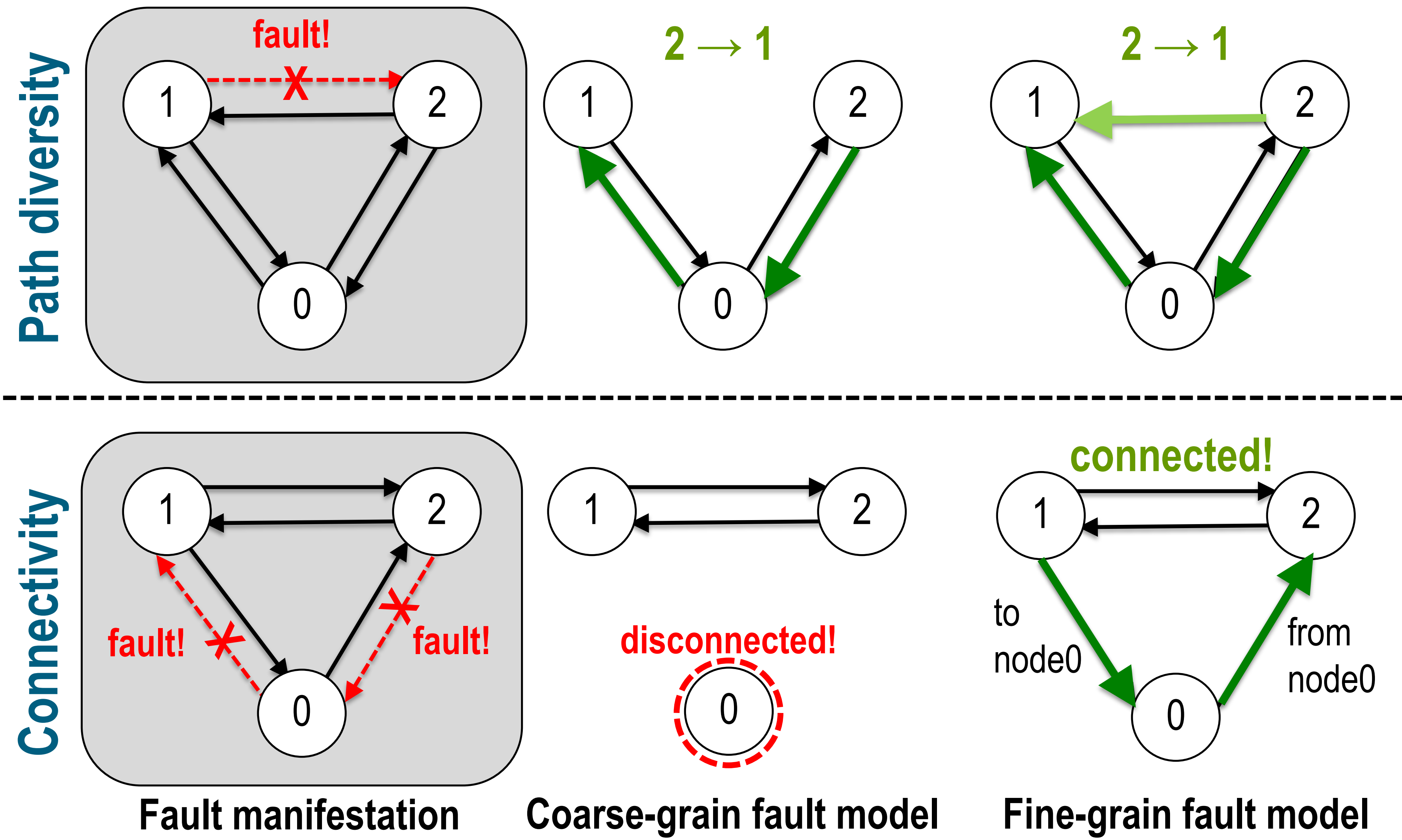
96% faults localized to a single datapath segment, or a unidirectional link

KEY OBSERVATION:
disable only ONE unidirectional link on most faults



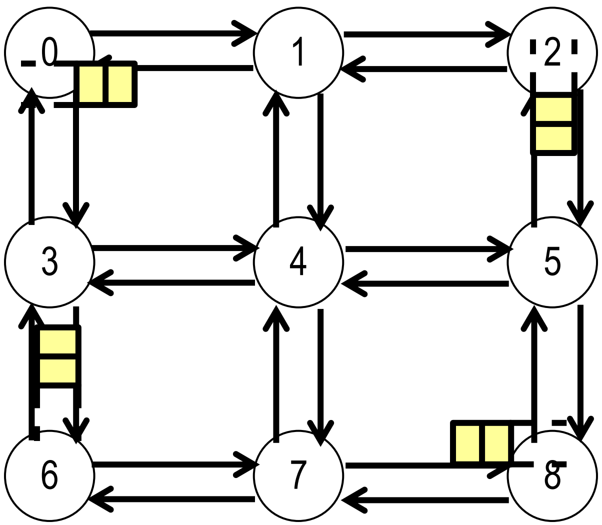
Error!!
Error!!
Error!!

Benefits of the Fine-Grain Fault Model



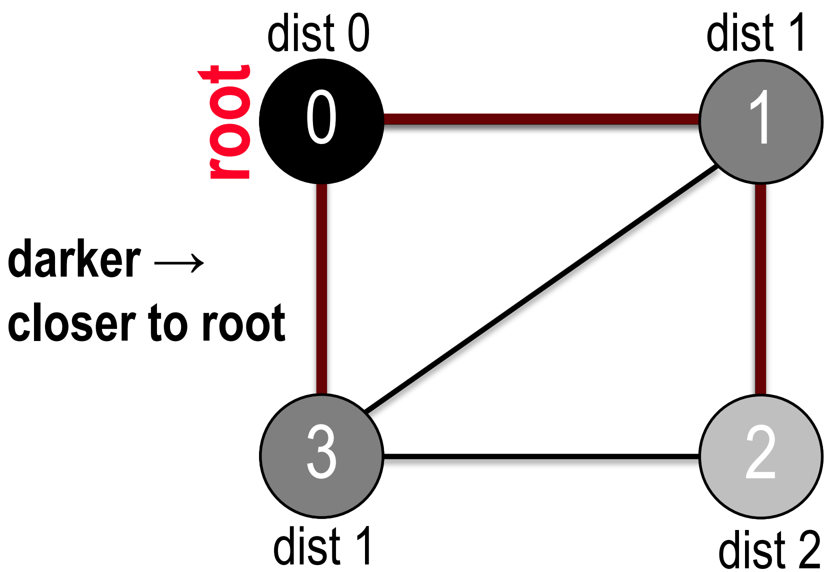
Rest of This Talk

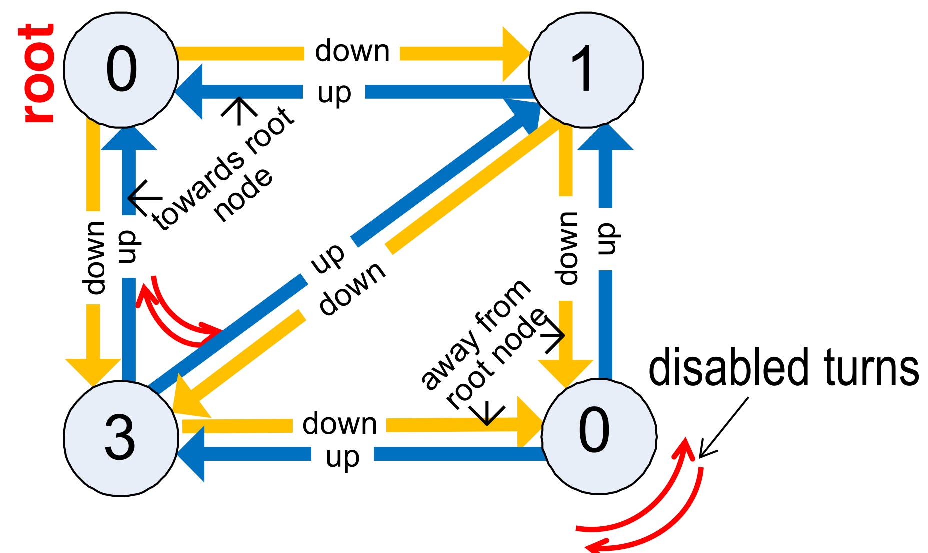
- ❑ Prior Work
- ❑ Fine-Resolution Diagnosis
- ❑ **Routing Algorithm**
- ❑ **Reconfiguration Algorithm**
- ❑ **Experimental Evaluation**



Routing in Irregular Networks

– Routing algorithm should disable paths that lead to deadlock
– Up*/down* routing disables **turns** to avoid deadlock
  1. Construct **spanning tree** rooted at a node (assumes bidirectional links)
  2. Mark links towards root: **up** (**down** otherwise)
  3. Disable all **down**→**up** turns
  4. Follow **up** links towards root and **down** links to destination

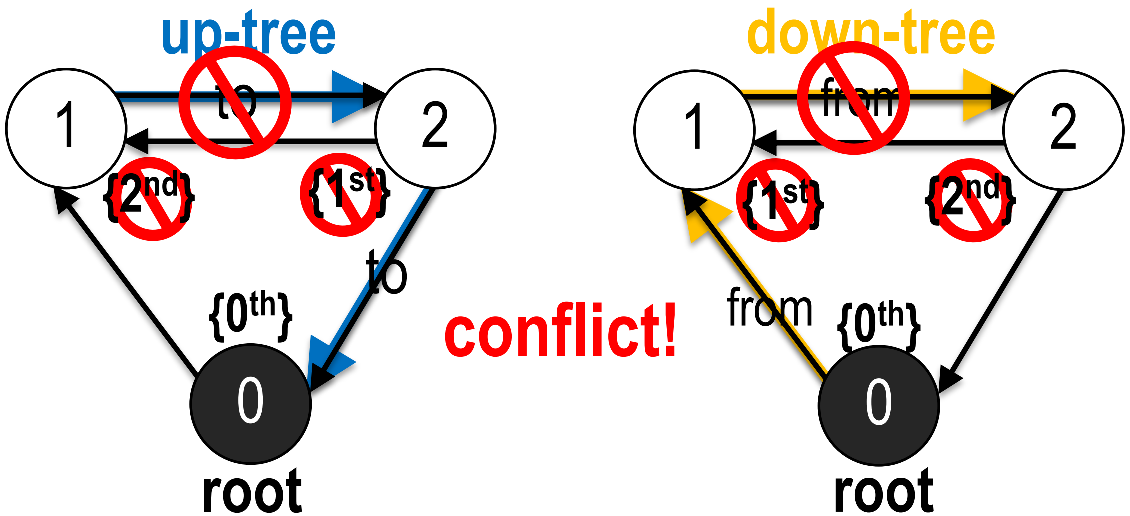




Up*/down* does not work with unidirectional links

Routing with Unidirectional Links

- Separate spanning trees for **up** (**up-tree**) and **down** (**down-tree**) links
  - **Up-tree**: unidirectional links towards root
  - **Down-tree**: unidirectional links away from root
  - **Consistent ordering/labeling**: no link marked both **up** and **down**
- As links are marked according to up*/down* principle (and no conflicts)
  - uDIREC routing is **deadlock free** (disable **down**→**up** turns)
  - Network is **connected** if both trees are complete



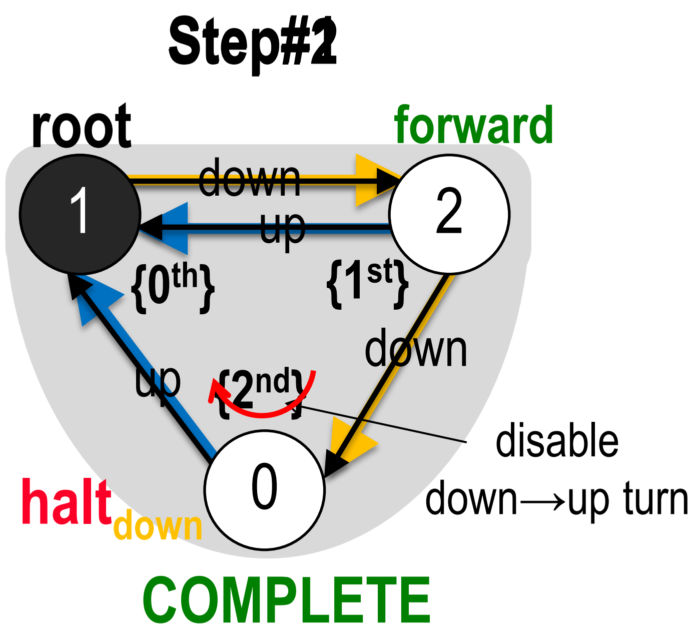
Growing Trees Concurrently

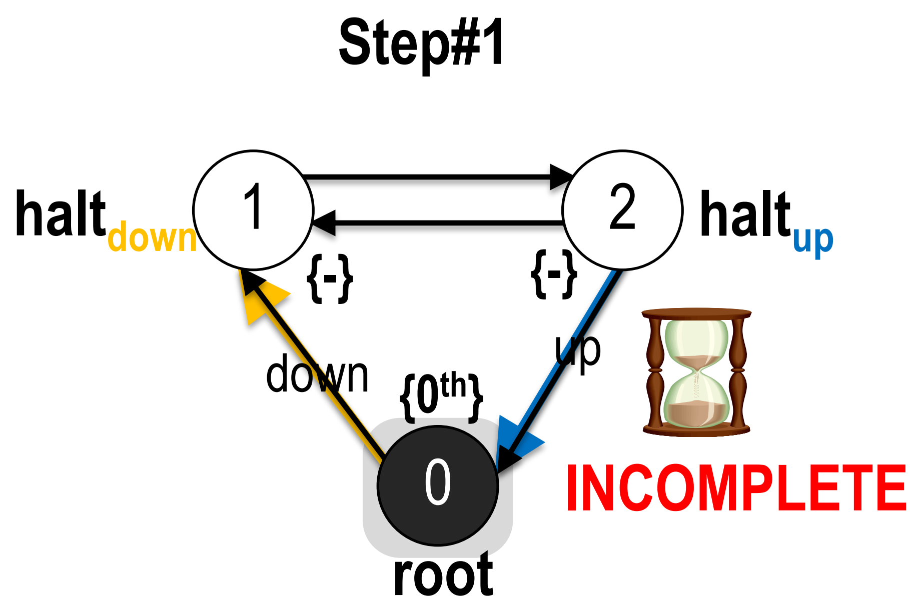
Up-tree and down-tree can be constructed:

- **Independently**: may lead to inconsistent marking
- **Concurrently**: consistent labeling ensured by construction

Grow tree beyond a node only if reachable by both up-tree and down-tree

Choice of root node affects connectivity





Reconfiguration Overview

Integrated with the software implementation of diagnosis scheme

– Suspension of network operation on fault detection

– Diagnosis of fault site via software scoreboard

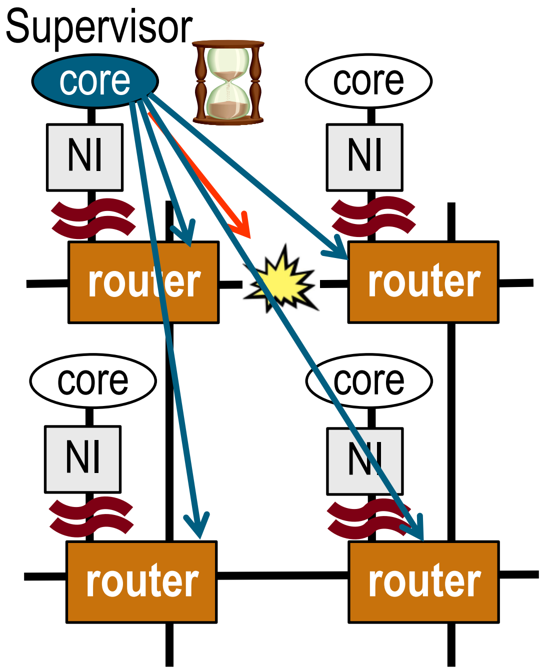
– Determination of surviving topology at supervisor

– Calculation of new routes in software

  - **Selection** of root that maximizes connectivity

– Distribution of new deadlock-free routes to routers

– Resumption of operation from uncorrupted state



Reconfiguration Algorithm

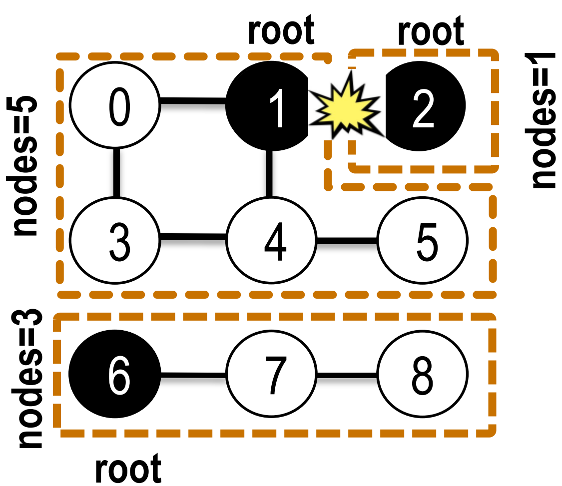
Root selection process

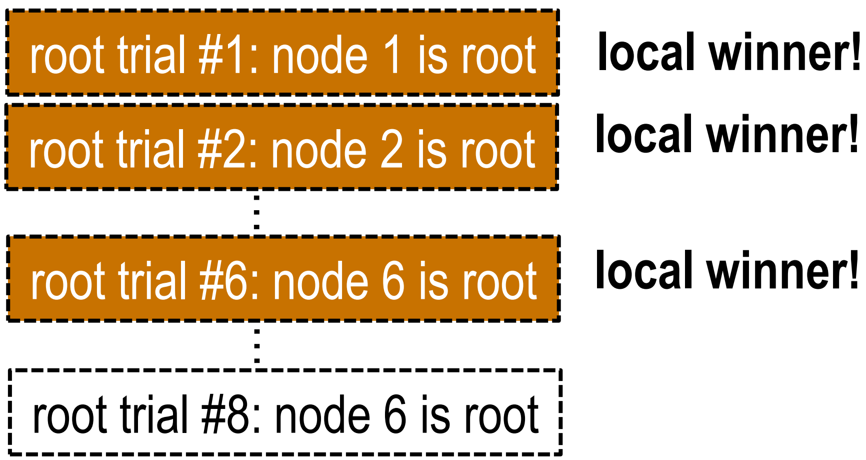
– Exhaustive search of the optimal root node

Optimality of the root node

– Based on number of connected nodes (in our experiments)

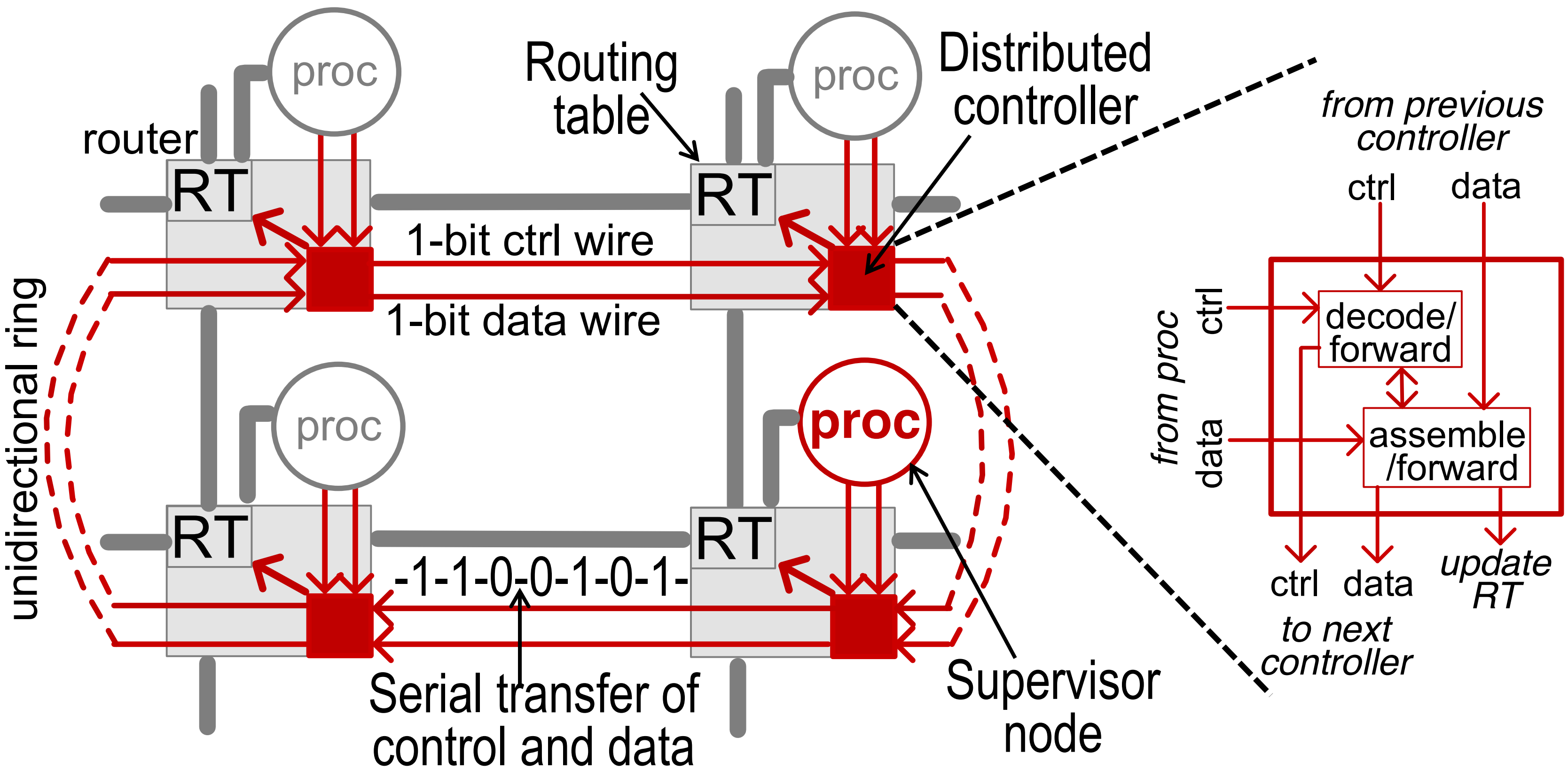
– Based on critical functionality within sub-network



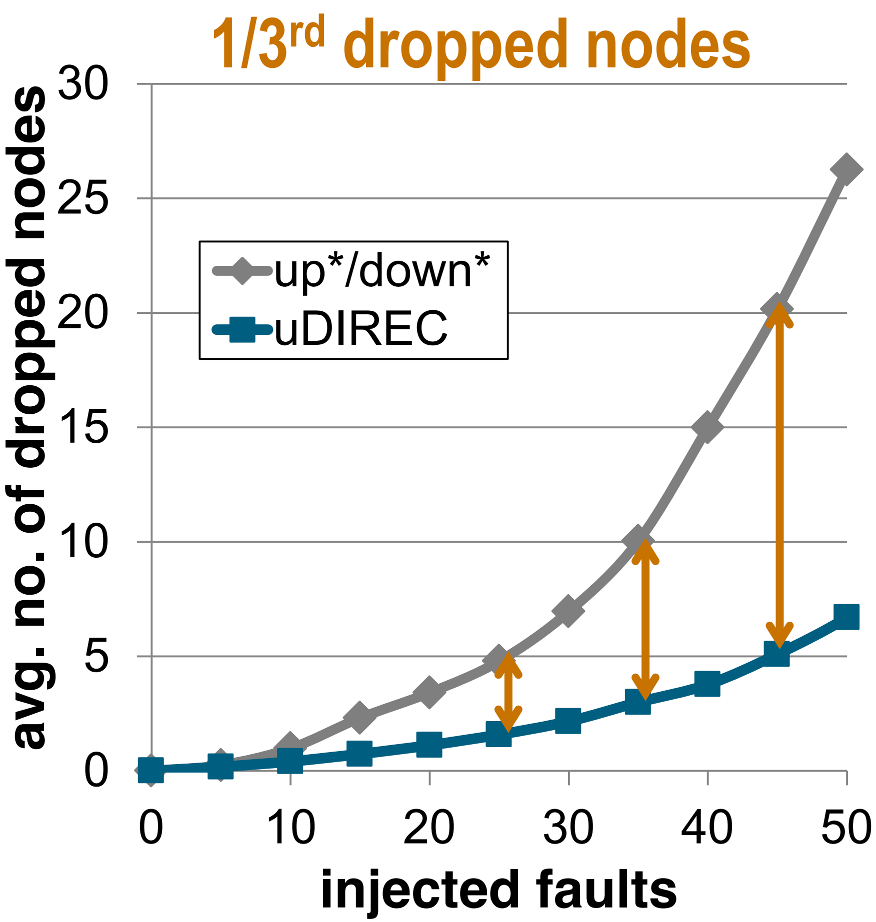


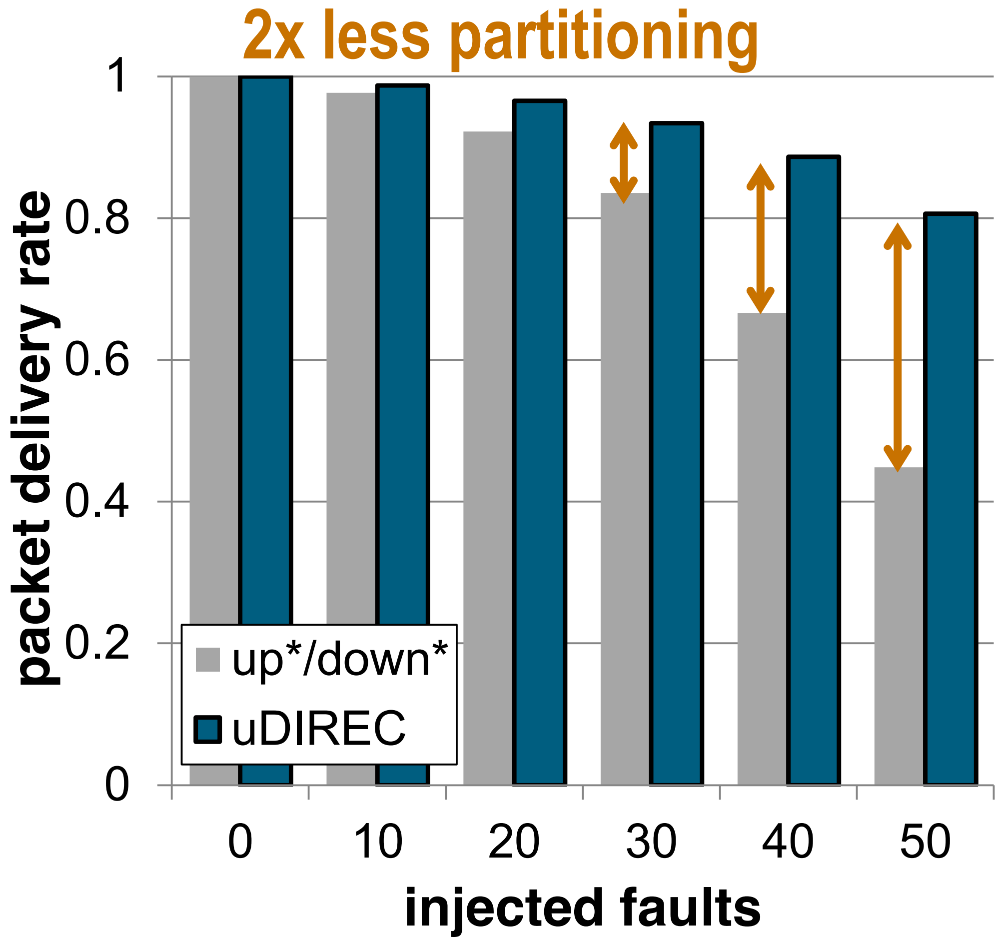
Reconfiguration Implementation

- Permanent faults are rare occurrences
- Routing table data = 4 (directions) * 64 (destinations) * 64 (RTs) **< 2 KB**
- Distribute using unidirectional ring of 1-control and 1-data wire



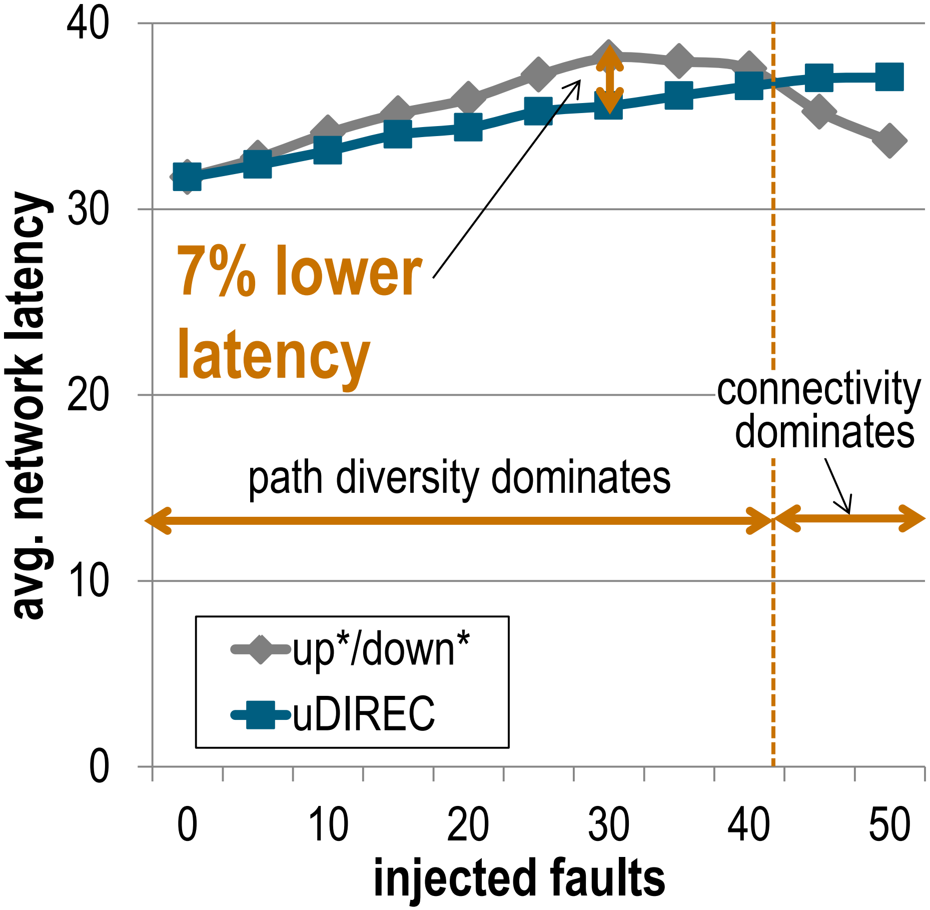
Reliability Results

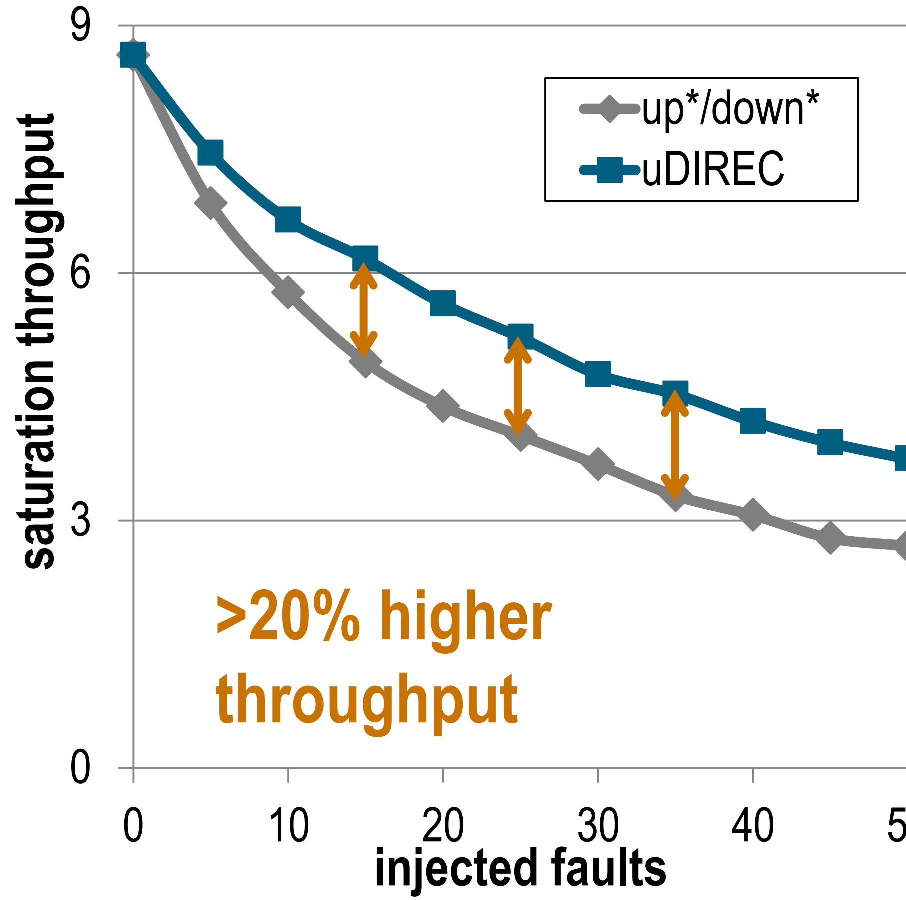




- As faults accumulate networks become disconnected
- uDIREC looses fewer nodes and partitions into fewer networks

Performance Results





- Initially latency degrades gracefully; at higher fault rates up*/down* drops much more nodes, hence lower latency
- uDIREC consistently delivers higher throughput

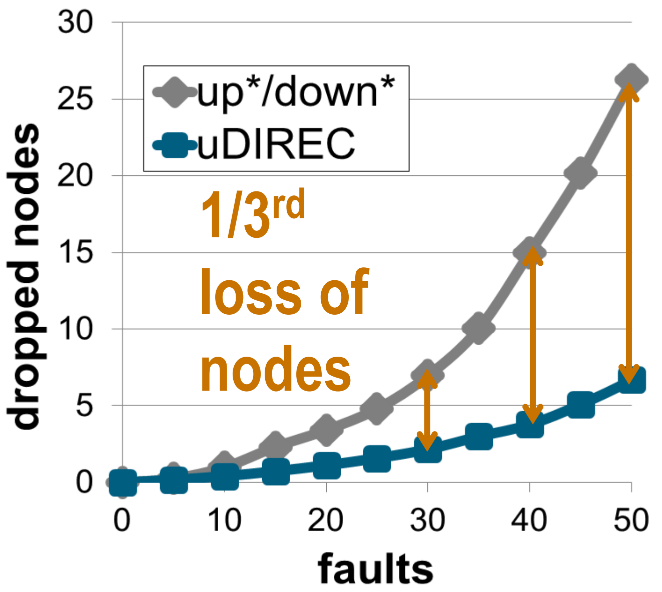
Conclusions

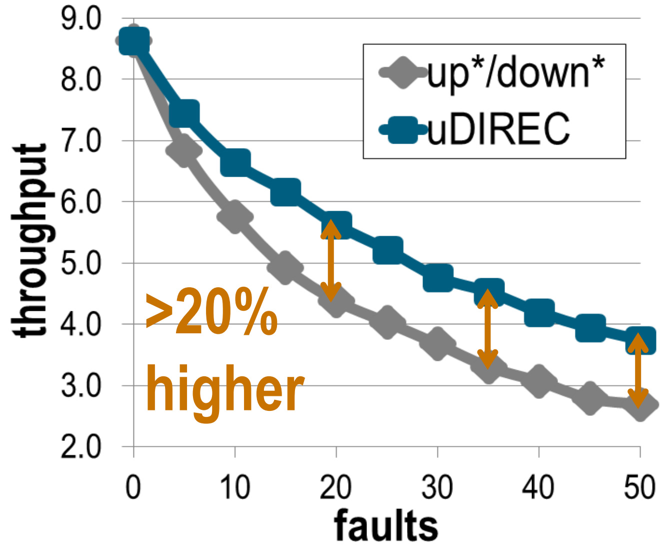
Proposed **uDIREC**: a unified diagnosis and reconfiguration solution

Fine-grained fault **diagnosis**, a novel **fault model**, a deadlock-free **routing** algorithm and a software-based **reconfiguration** algorithm

uDIREC drops only **1/3rd** nodes compared to state-of-the-art and delivers **>20%** higher throughput beyond 15 faults

uDIREC incurs **less than 1%** wiring overhead





Thank you! Questions?