



Optimizing Dynamic Mapping Techniques for On-Line NoC Test

Qiong Wu

University of Electronic Science and Technology of China

Outline

- ❖ Background
- ❖ Problem definition
- ❖ Testing-aware mapping algorithm(TAMA)
- ❖ Evaluation
- ❖ Conclusion

Background

Network on Chip

a modular and scalable communication architecture

internal defects
aging process
wear-out

tiny defects

Intermittent faults

occur repeatedly in fixed positions while having a certain randomness on the time of occurrence

increasing test frequency

idle times

Mapping algorithms

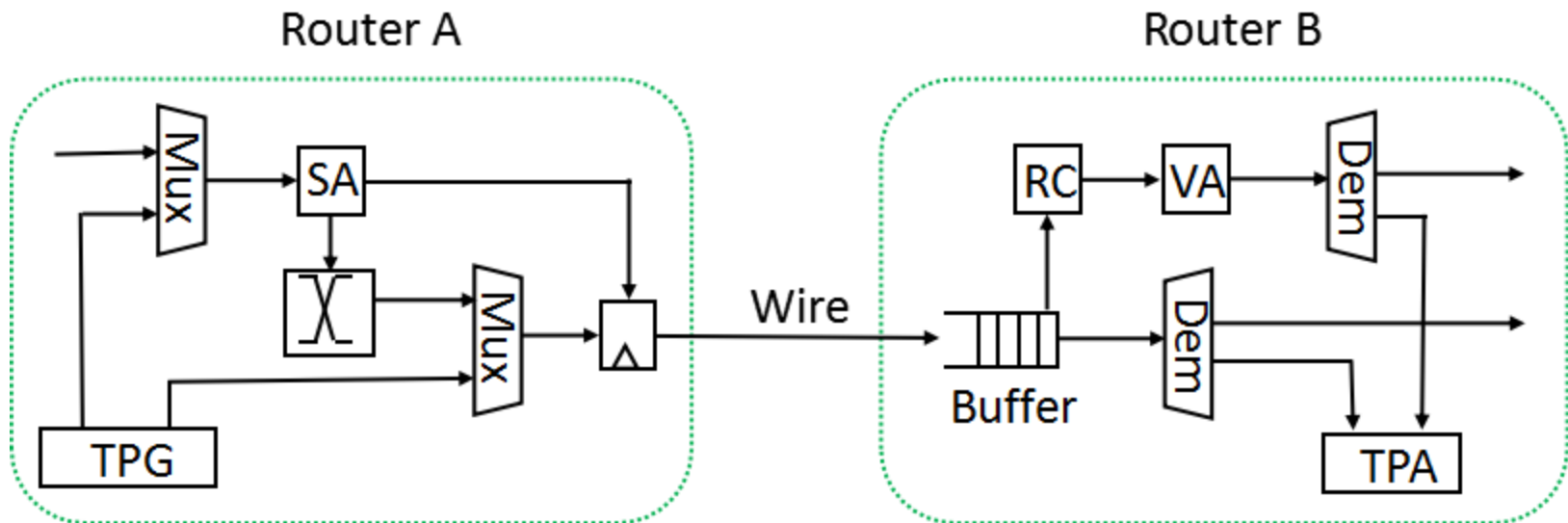
decide the **location** of application tasks

underlying routing algorithm and the mapping strategy distinguish **idle paths**

Outline

- ❖ Background
- ❖ Problem definition
- ❖ Testing-aware mapping algorithm(TAMA)
- ❖ Evaluation
- ❖ Conclusion

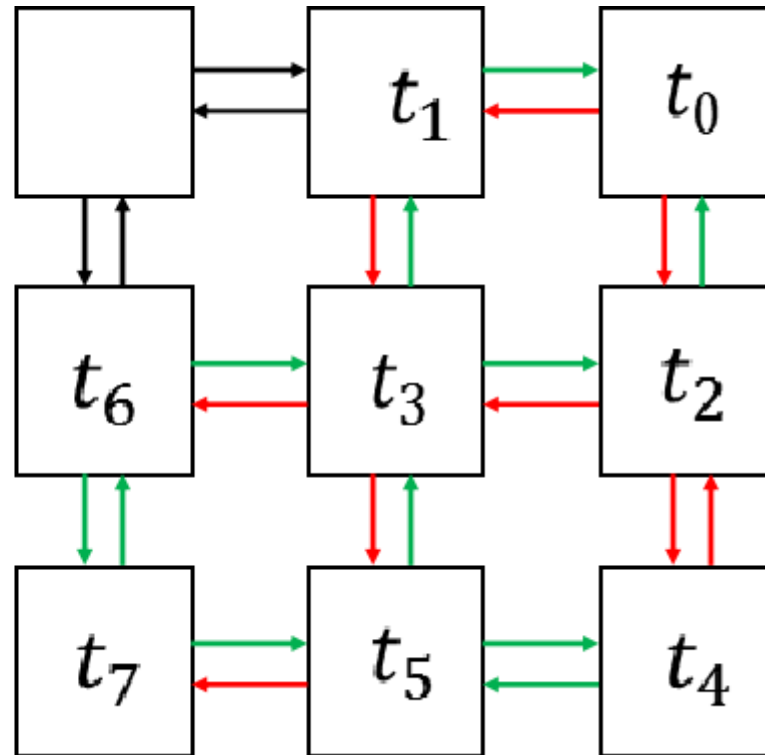
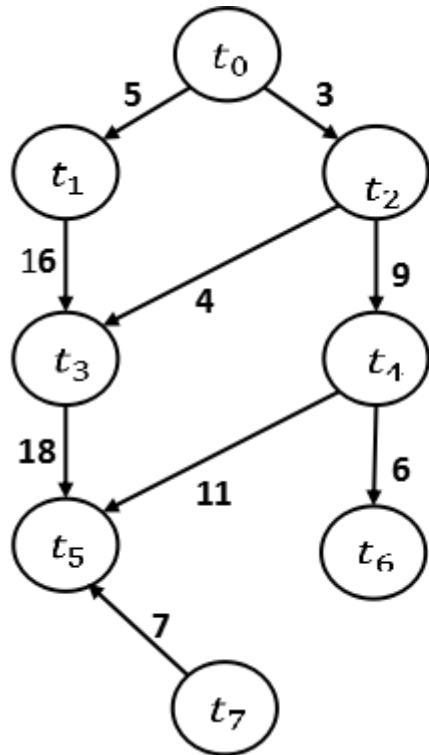
BIST Structure



A link/path test:

testing all the components between the TPG unit of one router to the TPA unit of the next router

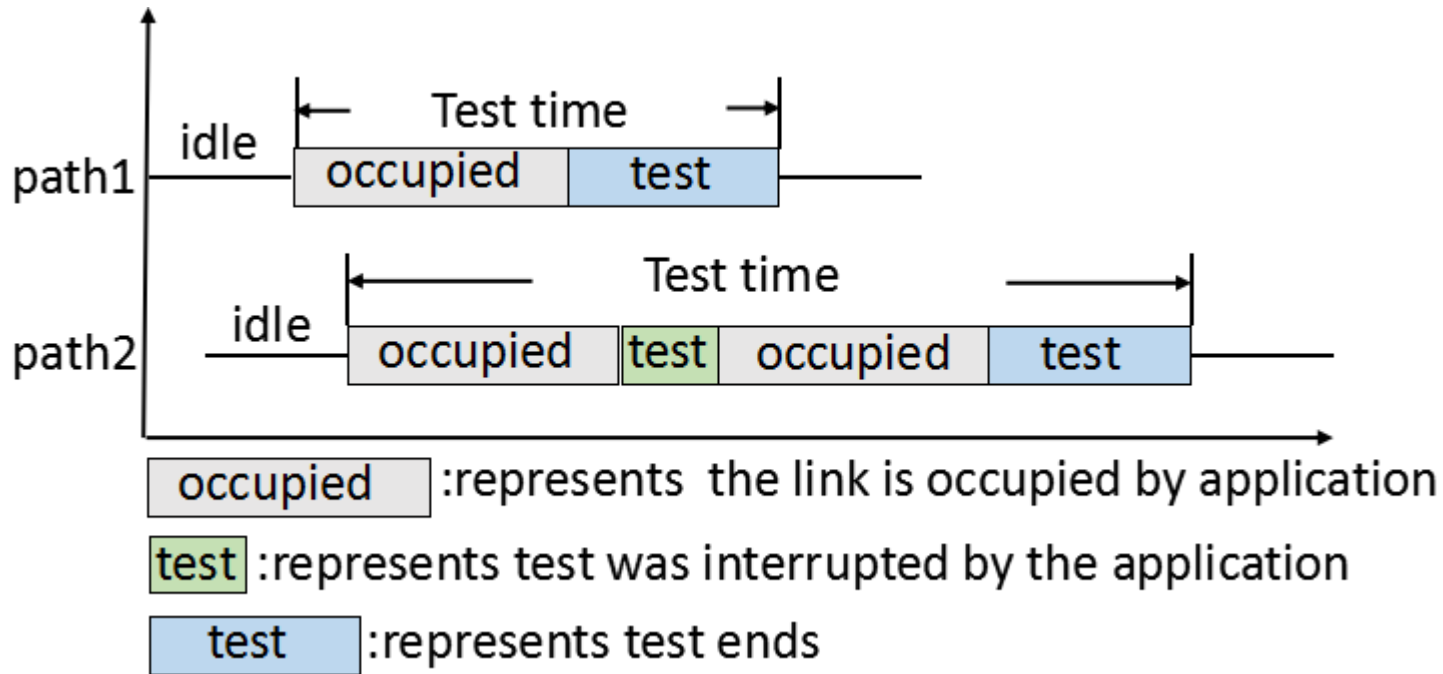
Testing the Paths



An application with 8 tasks and 9 edges running on NoC. We use the underlying XY routing to show the result of free (green) and occupied (red) paths.

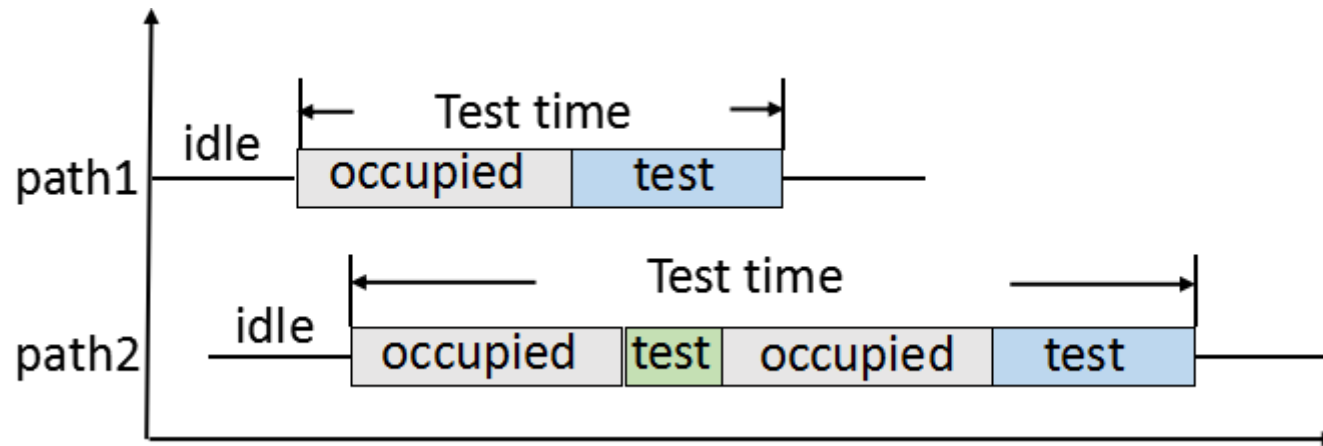
On-the-field test strategy

The path test time under two conditions.



The test start time; Identifying and testing the free paths; Handling the conflicting situation of test and mapping

Reliability evaluation metrics



Define a reliability evaluation: **average test time**
Lower average test time $\xrightarrow{\text{means}}$ higher test frequency
intermittent faults $\xleftarrow{\text{detect}}$

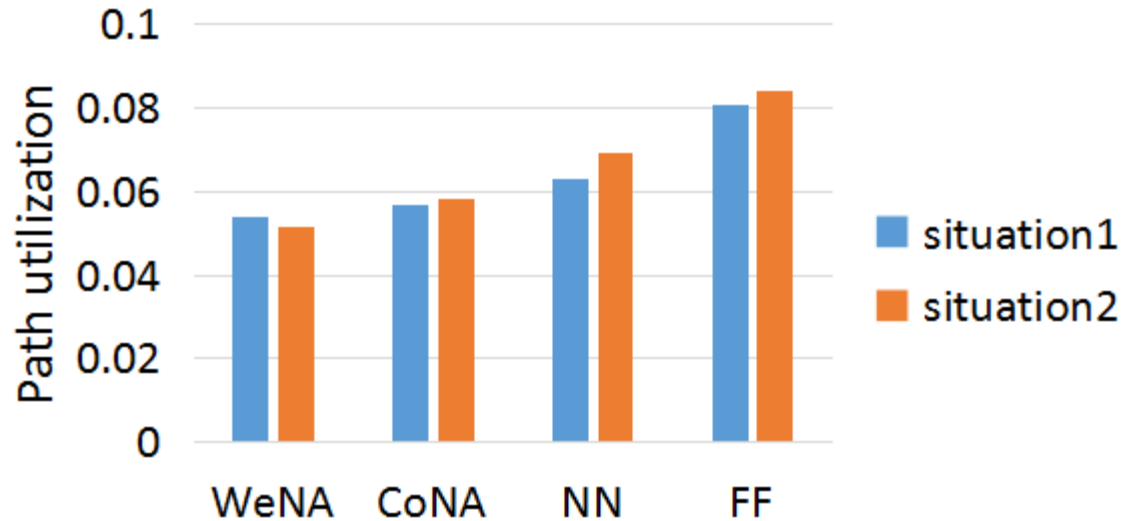
Outline

- ❖ Background
- ❖ Problem definition
- ❖ Testing-aware mapping algorithm(TAMA)
- ❖ Evaluation
- ❖ Conclusion

Testing-aware mapping algorithm(TAMA)

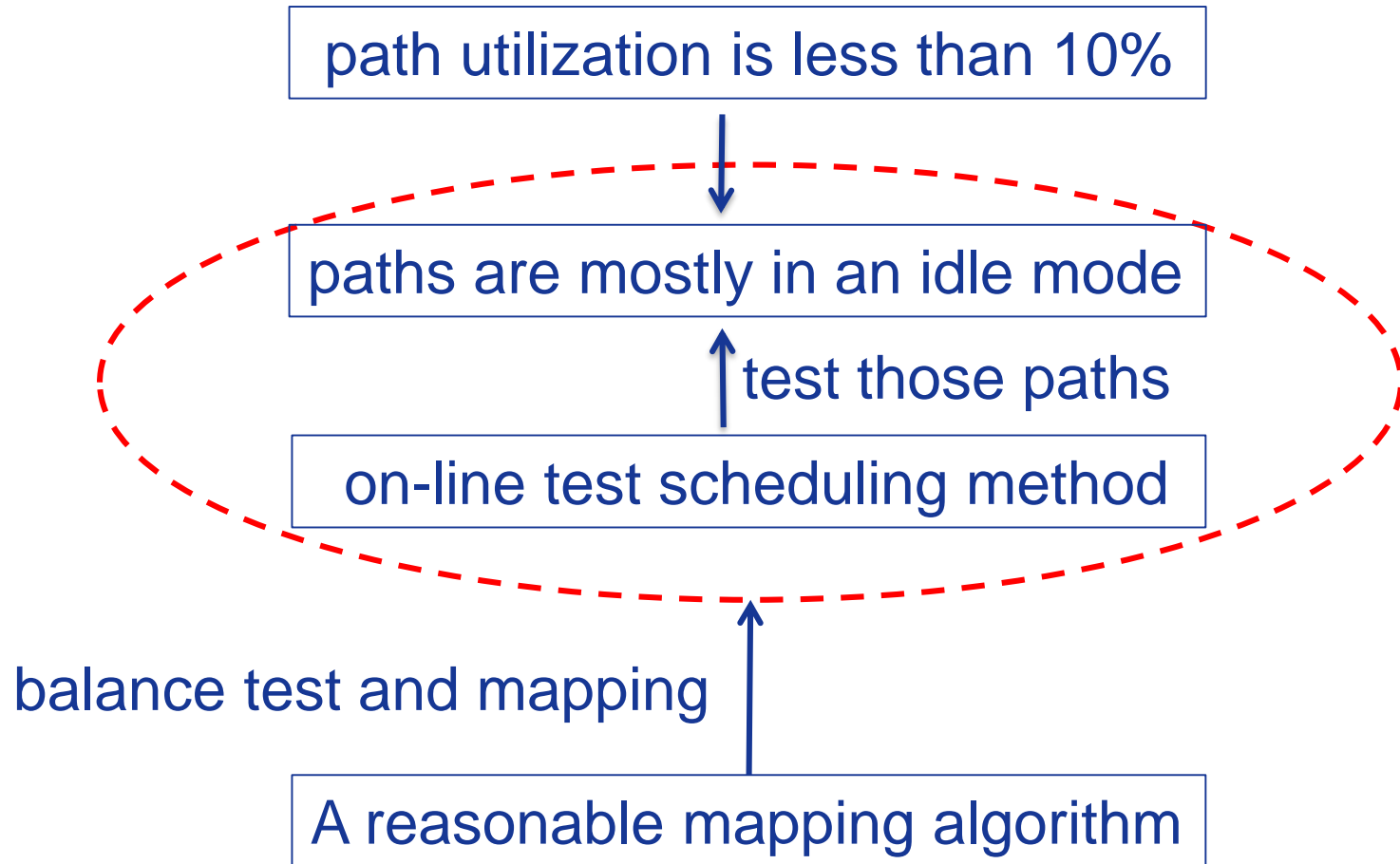
The link utilization is defined:

$$L_{utilization} = \frac{\sum_{t=1}^{T_s} N_c(t)}{N_l * T_s}$$



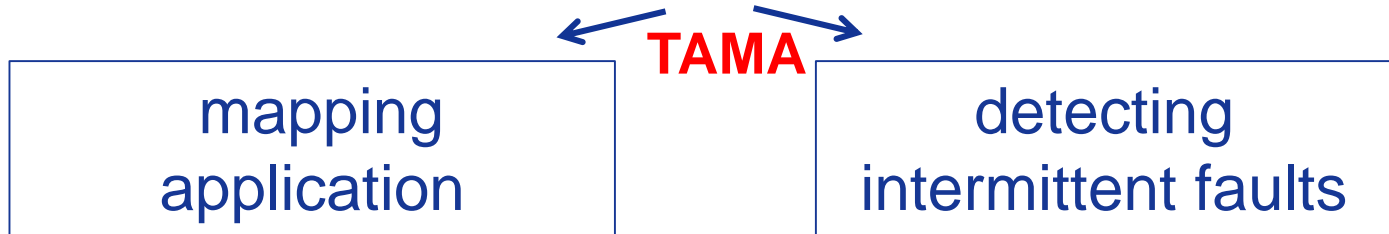
N_c : the number of utilized paths at each simulation cycle;
 N_l : the number of paths in the network;
 T_s : the total simulation cycles.

Testing-aware mapping algorithm(TAMA)



Testing-aware mapping algorithm(TAMA)

there is a quest for a testing-aware mapping algorithm



A. Sorting tasks

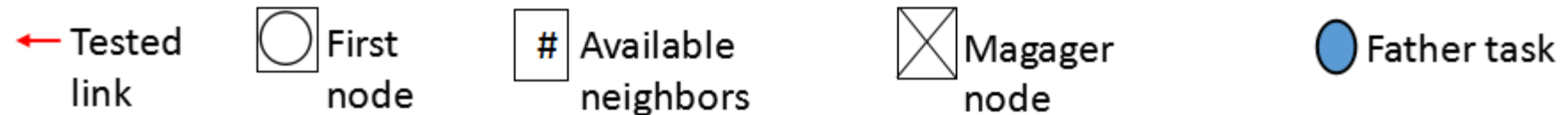
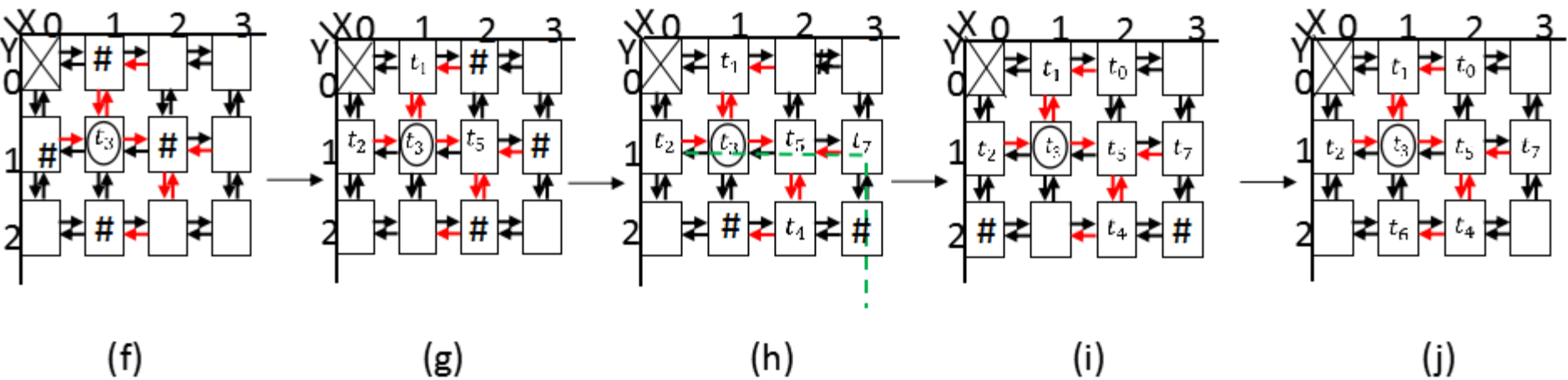
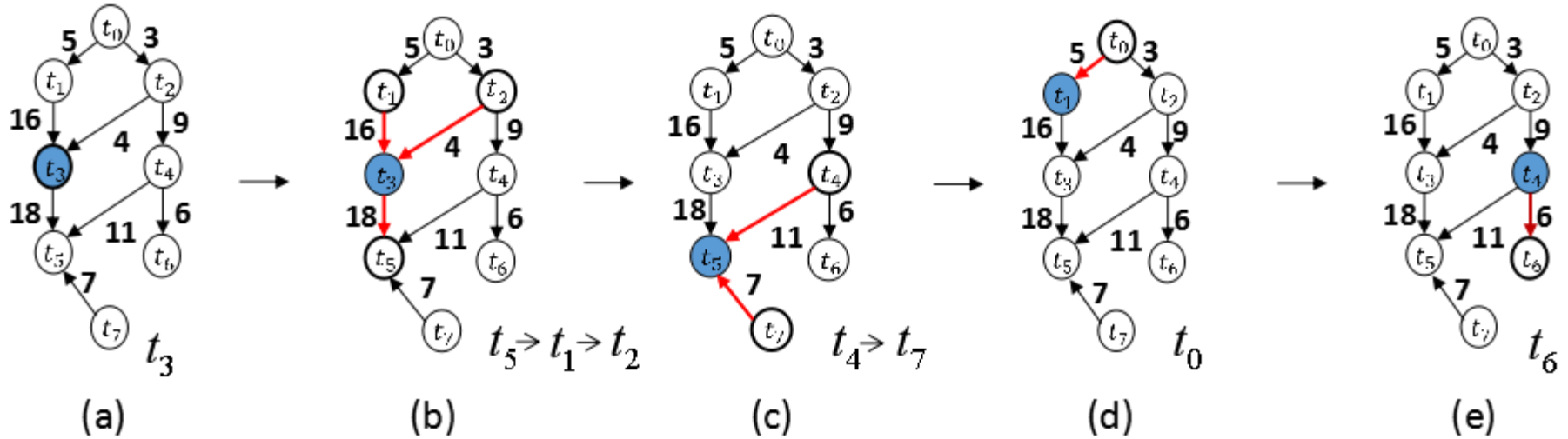
1: first task

the largest number of edges and communication volume

2: remaining tasks

the breadth-first traversal technique

Testing-aware mapping algorithm(TAMA)



Testing-aware mapping algorithm(TAMA)

B. Mapping tasks
to nodes

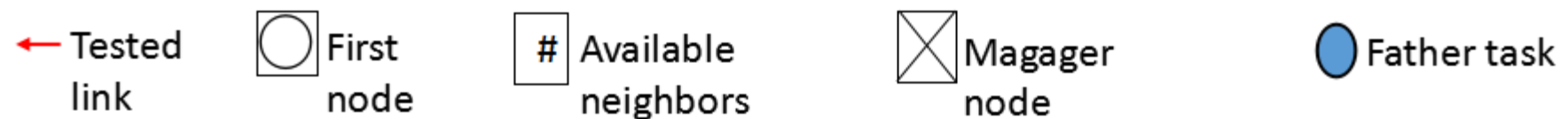
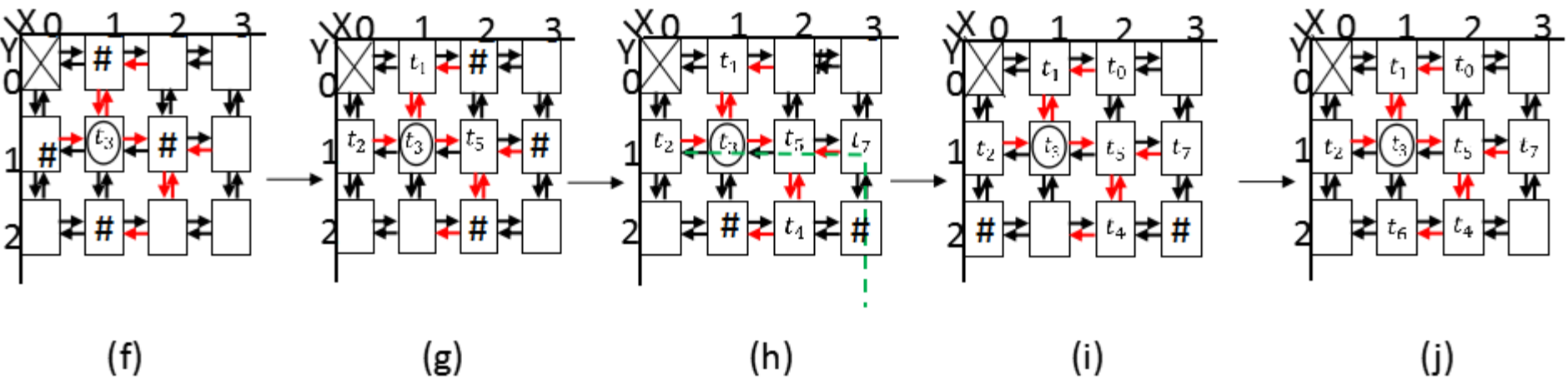
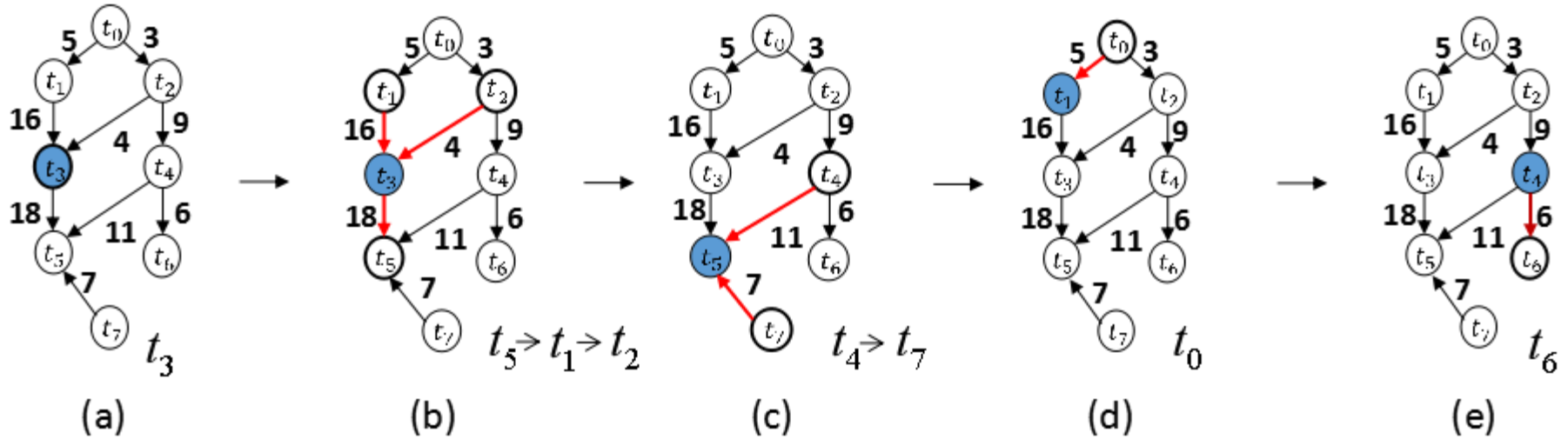
1. first node

2. remaining
nodes

a. maximum number of free neighbors
b. maximum number of tested links
c. closer to the manager

a. closest to the father task node
b. maximum number of tested links
c. data flow direction

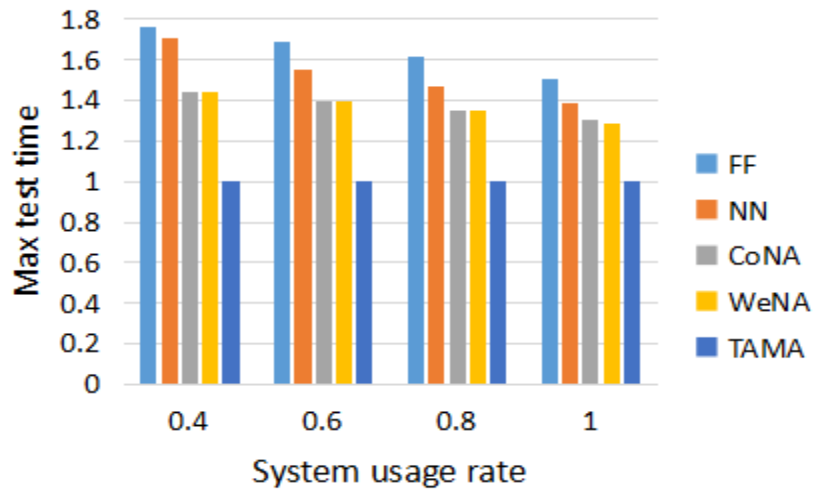
Testing-aware mapping algorithm(TAMA)



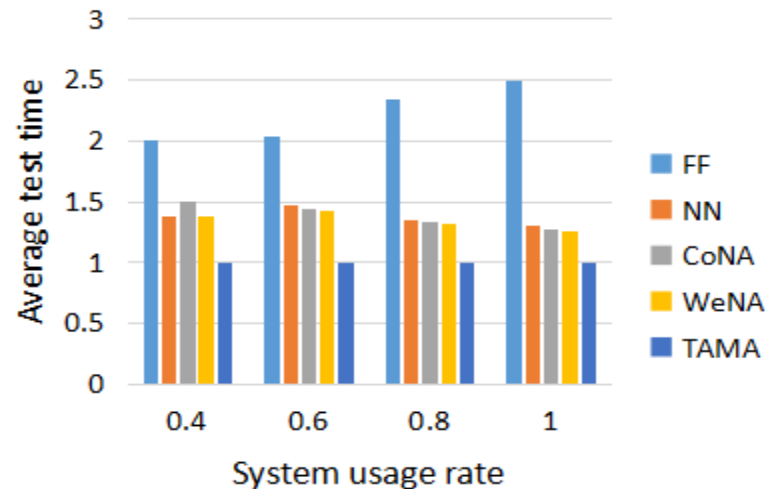
Outline

- ❖ Background
- ❖ Problem definition
- ❖ Testing-aware mapping algorithm(TAMA)
- ❖ Evaluation
- ❖ Conclusion

Maximum Test Time and Average Test Time



(a)



(b)

Obviously decreases the maximum test time and average test time.
The benefit of TAMA is more significant in lower usage rates

↑
Testing paths at idle times

Interrupted Test Rate

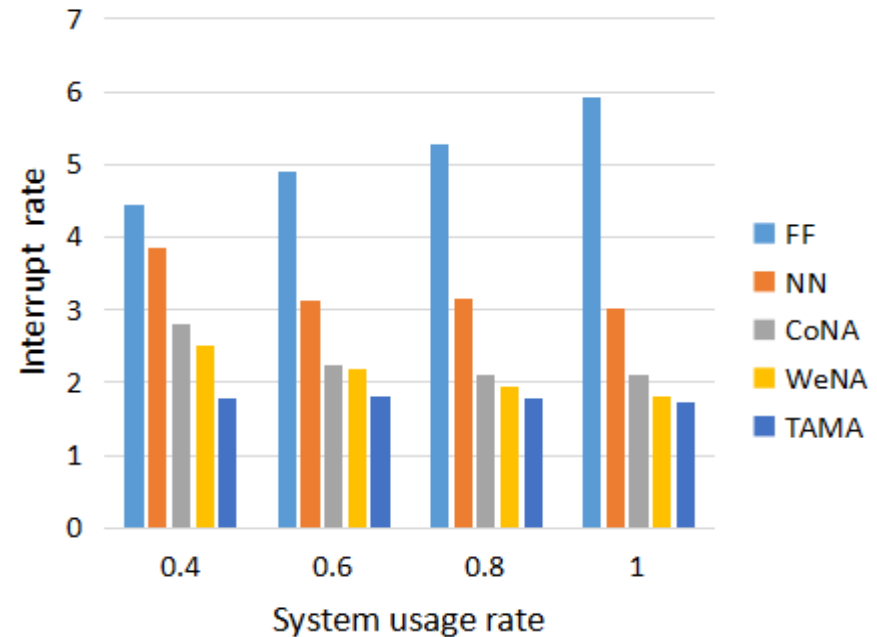
Interrupted test rate defined as:

$$T_{\text{interrupted}} = \frac{T_c + T_{ic}}{T_c}$$

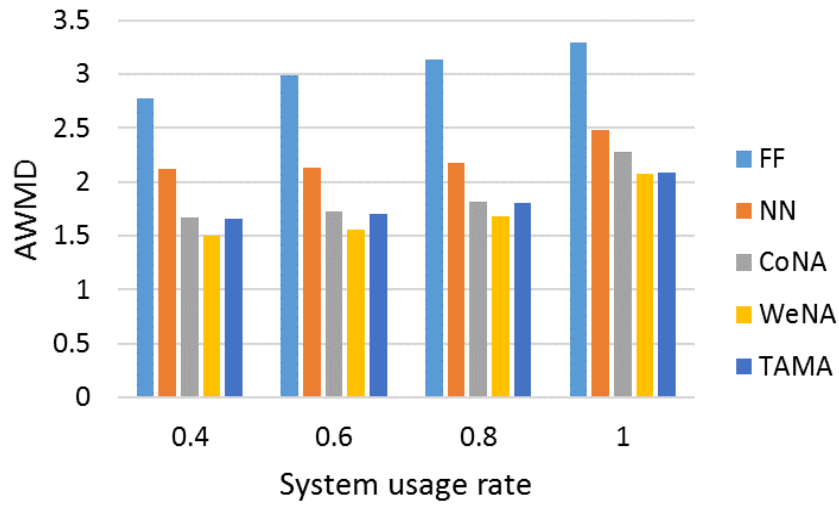
- The interrupted test rate is lowest in TAMA
- The fact that TAMA identifies the idle paths

T_c : the total number of test path during the application mapping

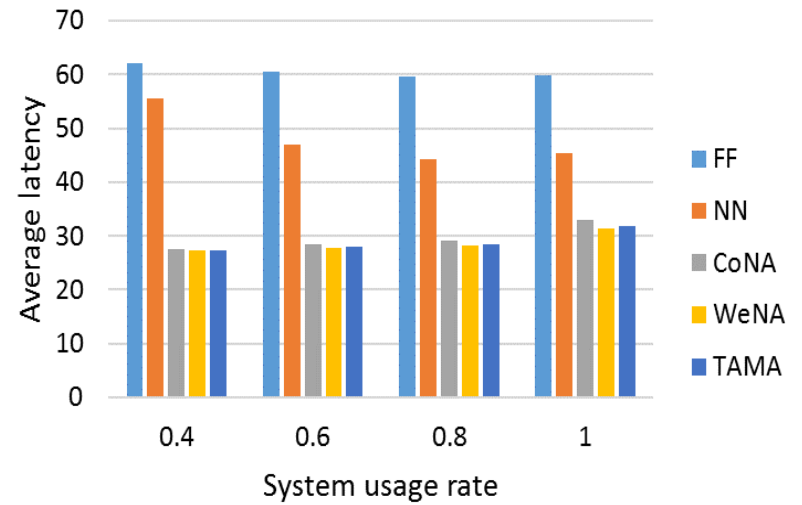
T_{ic} : the number of test that are interrupted due to the request of mapping



AWMD and Average Latency



(a)



(b)

The AWMA of TAMA is always lower than the other methods.
The average latency follows the same trend.

Outline

- ❖ Background
- ❖ Problem definition
- ❖ Testing-aware mapping algorithm(TAMA)
- ❖ Evaluation
- ❖ Conclusion

Conclusion

- ✓ A mapping algorithm to map application and detect intermittent faults over the NoC platform
- ✓ Experimental results showed that our mapping algorithm leads to improvements on the maximum test time , average test time and interrupt rate at no comprise of AWMD and average latency

Thank You!

ESYSim Tool QQ Group

