A Fault-Tolerant Routing Strategy for Fibonacci-Class Cubes

Xinhua Zhang[1] and Peter K. K. Loh[2]

[1] Department of Computer Science
National University of Singapore, Singapore
[2] School of Computer Engineering
Nanyang Technological University, Singapore

Merits

- Applicable to all Fibonacci-class Cubes in a unified fashion, with only minimal modification of structural representation
- The maximum number of faulty components tolerable is the network's node availability: min(deg n) where n is a node
- For a n-dimensional Fibonacci-class Cube, each node of degree *deg* maintains and updates at most $n(deg + 2)$ bits' vector information
- Generates deadlock-free and livelock-free routes
- Can be implemented almost entirely with simple and practical routing hardware requiring minimal processor control

Road Map

- Introduction
- Generic approach to cycle-free routing (GACR)
- Fault-tolerant Fibonacci routing (FTFR)
- Experimental results
- Conclusion and future work

Road Map

- Introduction
- Generic approach to cycle-free routing (GACR)
- Fault-tolerant Fibonacci routing (FTFR)
- Experimental results
- Conclusion and future work

Introduction 1. Fibonacci-class cubes: *FC* definition

1. Fibonacci Cubes (FC_n) $f_n = f_{n-1} + f_{n-2}$

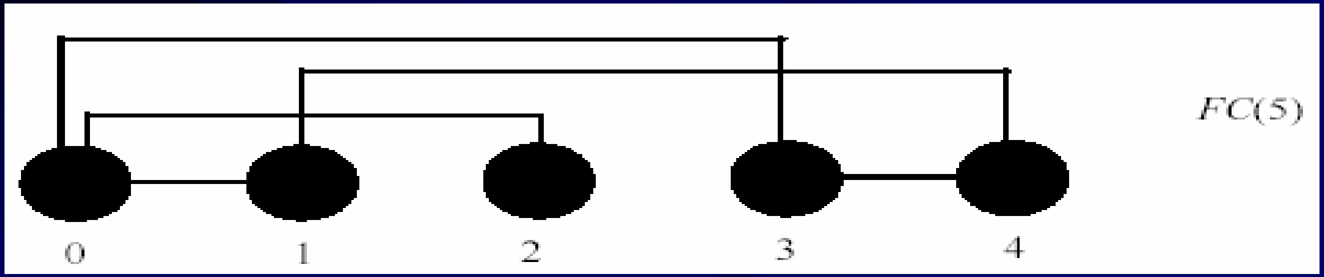
$f_0 = 0,\ f_1 = 1,\ f_2 = 1,\ f_3 = 2,\ f_4 = 3,\ f_5 = 5,\ f_6 = 8\ \ldots$

Natural number *i*	2nd order Fibonacci code
0	$(00000)_F$
1	$(00001)_F$
2	$(00010)_F$
3	$(00100)_F$
4	$(00101)_F$
5	$(01000)_F$
6	$(01001)_F$
7	$(01010)_F$
8	$(10000)_F$

Introduction 1. Fibonacci-class cubes: *FC* example

1. Fibonacci Cubes Example

Natural number *i*	2nd order Fibonacci code
0	000
1	001
2	010
3	100
4	101



Introduction 1. Fibonacci-class cubes: *FC* equivalent definition

1. Fibonacci Cubes: equivalent recursive definition

$$V_n = \left(0 \,\|\, V_{n-1}\right) \cup \left(10 \,\|\, V_{n-2}\right) \qquad n \geq 5$$

$$V_3 = \{1,\ 0\} \qquad V_4 = \{01,\ 00,\ 10\}$$

Edge: Hamming distance $= 1$

Introduction 1. Fibonacci-class cubes: *EFC* definition

2. Enhanced Fibonacci Cubes (EFC_n) $f_n = 2f_{n-2} + 2f_{n-4}$

$$V_n = (00 \| V_{n-2}) \cup (10 \| V_{n-2}) \cup (0100 \| V_{n-4}) \cup (0101 \| V_{n-4}) \qquad n \geq 7$$

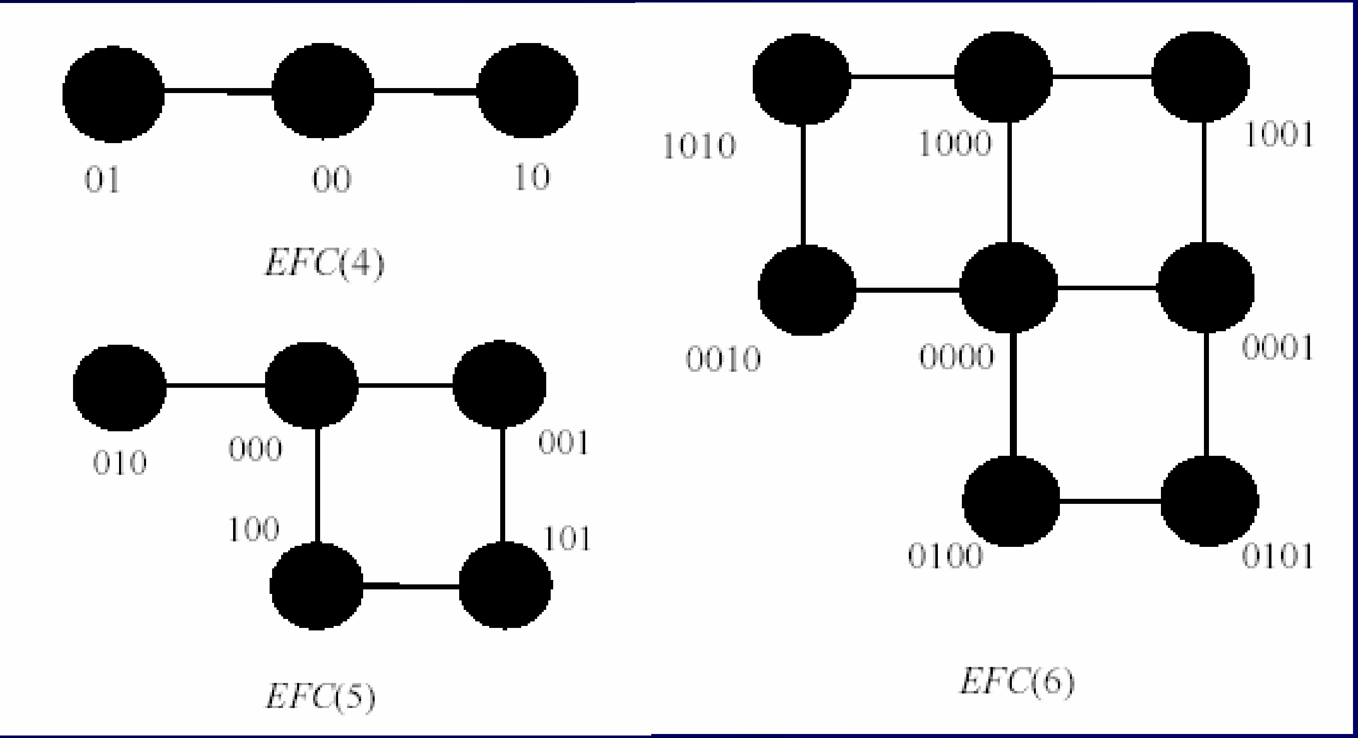
$V_3 = \{1, 0\}$ $\quad V_4 = \{01,\ 00,\ 10\}$ $\quad V_5 = \{001,\ 101,\ 100,\ 000,\ 010\}$

$V_6 = \{0001,\ 0101,\ 0100,\ 0000,\ 0010,\ 1010,\ 1000,\ 1001\}$

Edge: Hamming distance $= 1$

Introduction 1. Fibonacci-class cubes: *EFC* example

2. Enhanced Fibonacci Cubes Examples



Introduction 1. Fibonacci-class cubes: *XFC* definition

3. Extended Fibonacci Cubes $XFC_k(n)$

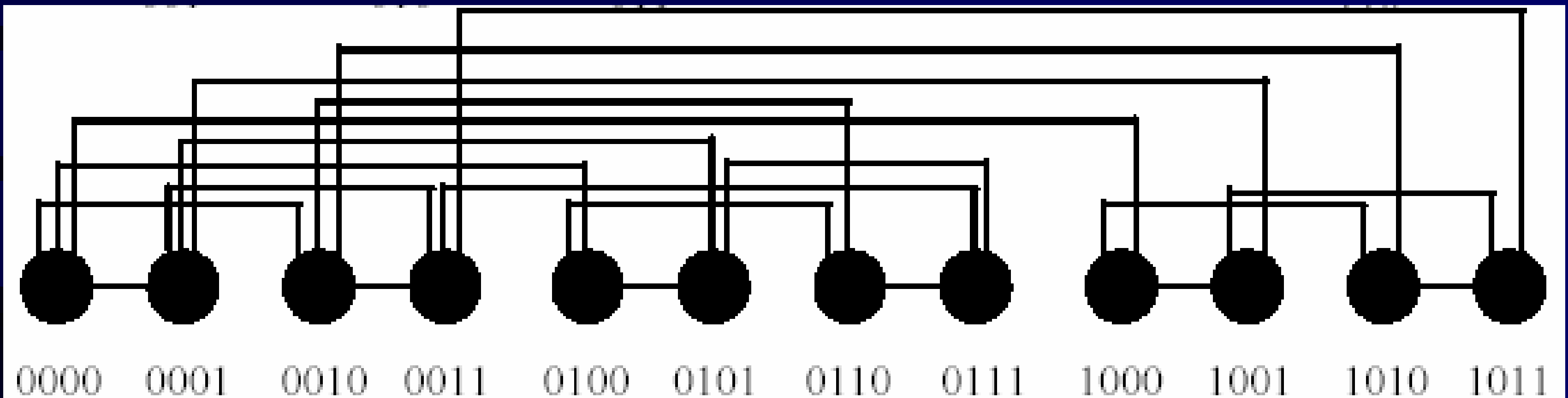
$$V_k(n) = \left(0 \,\|\, V_k(n-1)\right) \cup \left(10 \,\|\, V_k(n-2)\right) \quad n \geq k+4$$

$$V_k(k+2) = \{0,\ 1\}^k \qquad V_k(k+3) = \{0,\ 1\}^{k+1}$$

Edge: Hamming distance $= 1$

Introduction 1. Fibonacci-class cubes: *XFC* example

3. Extended Fibonacci Cubes $XFC_k(n)$



$XFC_2(6)$

Introduction 1. Fibonacci-class cubes: summary

In sum:

$$FC_n: \quad V_n = \left(0 \,\|\, V_{n-1}\right) \cup \left(10 \,\|\, V_{n-2}\right) \qquad n \geq 5$$

$$EFC_n: \quad V_n = \left(00 \,\|\, V_{n-2}\right) \cup \left(10 \,\|\, V_{n-2}\right) \cup \left(0100 \,\|\, V_{n-4}\right) \cup \left(0101 \,\|\, V_{n-4}\right) \qquad n \geq 7$$

$$XFC_k\left(n\right): \quad V_k(n) = \left(0 \,\|\, V_k(n-1)\right) \cup \left(10 \,\|\, V_k(n-2)\right) \qquad n \geq k+4$$

Edge: Hamming distance $= 1$

Introduction 2.
General Property

Proposition.

In a fault-free Fibonacci Cube, Enhanced Fibonacci Cube or Extended Fibonacci Cube:

there is always a preferred dimension available at the packet's present node before the destination is reached.

Implication: the use of a spare dimension can be boiled down to the encounter of faulty components (now or before).

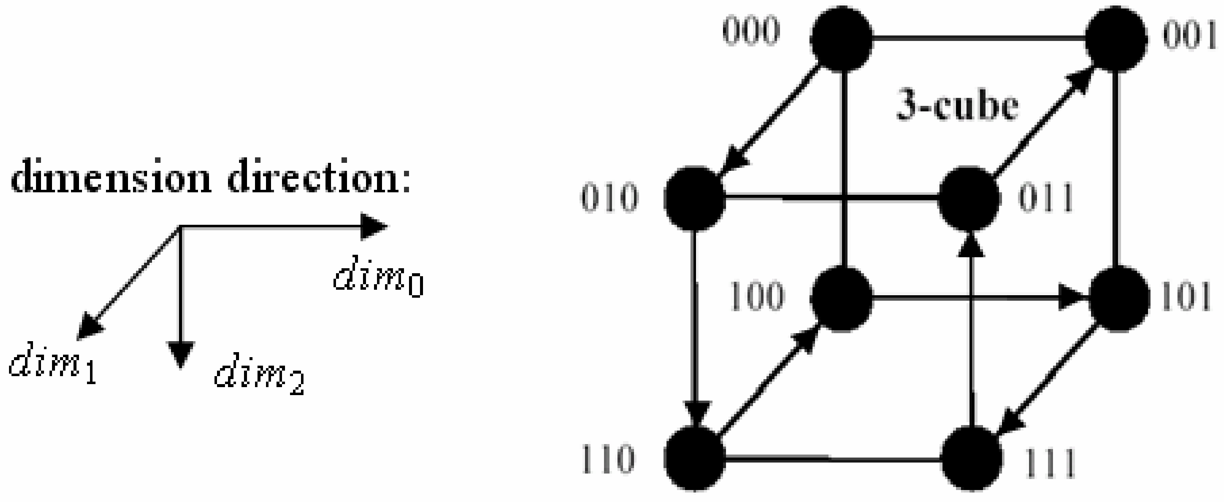
Road Map

- Introduction
- Generic approach to cycle-free routing (GACR)
- Fault-tolerant Fibonacci routing (FTFR)
- Experimental results
- Conclusion and future work

Generic approach to cycle-free routing (GACR)

- Purpose:
  1. avoid cycles in routing by checking the traversal history
  2. generality and efficiency

Generic approach to cycle-free routing: history vector



Basic GACR illustration

history: 1210121

Generic approach to cycle-free routing: cycle check

Equivalent condition for a route to contain cycle:

there exists a way of inserting '(' and ')' into the sequence such that each number in the parenthesis appears for an even number of times.

87586563243412 1 *a*

875865632434(121 2) ✗

875865(632434121 6) ✗

87586563243412 1 4 ✓

Generic approach to cycle-free routing: Cost

Cost:

Overhead length:

$$O(L_{max} \log n) = O(n \log n) \quad \text{if } O(L_{max}) = O(n)$$

Time complexity:

To check whether string s has a single 1: $O(1)$

To find all forbidden dimensions: $O(L_{cur}) = O(n)$

Road Map

- Introduction
- Generic approach to cycle-free routing (GACR)
- Fault-tolerant Fibonacci routing (FTFR)
- Experimental results
- Conclusion and future work

Fault-tolerant Fibonacci routing: Auxiliary vectors

- The main framework of the algorithm
- Auxiliary vectors
  - First filter out following dimensions
    - All the dimensions that are masked by GACR, including the incoming dimension
    - Dimensions which are faulty or non-existent by the definition of Fibonacci-class cubes (this makes the algorithm applicable to all Fibonacci-class cubes)
  - Setting a mask vector, M, with 0 for dimensions meeting either of the conditions above, and 1 otherwise (*adoptable*).

Fault-tolerant Fibonacci routing: Overview

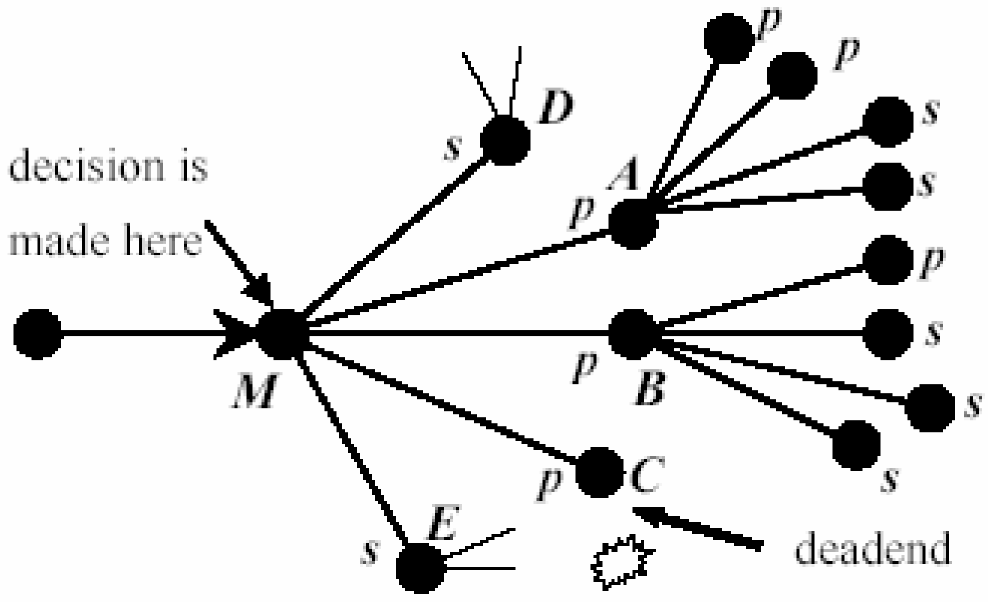


Illustration of *FTFR*

Fault-tolerant Fibonacci routing: Choosing from preferred dimensions

- If there are adoptable preferred dimensions
  - Look at neighbors on these dimensions
  - Pick the neighbor which has the largest number of preferred dimension (relative to the neighbor)
  - If tie, then pick the neighbor with the largest number of spare dimensions
  - If still tie, choose 0->1 dimension

Fault-tolerant Fibonacci routing: Choosing from spare dimensions

- If there is NO adoptable preferred dimension
  - Look at neighbors on spare dimensions
  - Pick the neighbor which has the largest number of preferred dimension
  - If tie, then pick the neighbor with the largest number of spare dimensions
  - If still tie, choose 1->1 dimension

Fault-tolerant Fibonacci routing: control of using spare dimension

- One caveat, control of using spare dimension
  - All dimensions can be used as a spare dimension for at most once
  - This is attained by using a mask vector DT:
    - Set DT to straight 1 at the start/source.
    - If one spare dimension is chosen to be used
      - Check if the corresponding bit in DT is 1
      - If 1, then OK. If 0, then forbid using it and try other dimensions.
      - After using the dimension, set the corresponding bit in DT to 0

Fault-tolerant Fibonacci routing: speed up

- two heuristics:
  - If the neighbor is the destination, then go to it.
  - If the neighbor is on dimension *d*, and the destination has a (imagined) link on dimension *d*, then add the network availability to the score

$$\underline{\#\,prefer \times n + \#\,spare} + node\_availability$$

Road Map

- Introduction
- Generic approach to cycle-free routing (GACR)
- Fault-tolerant Fibonacci routing (FTFR)
- Experimental results
- Conclusion and future work

Experimental Results

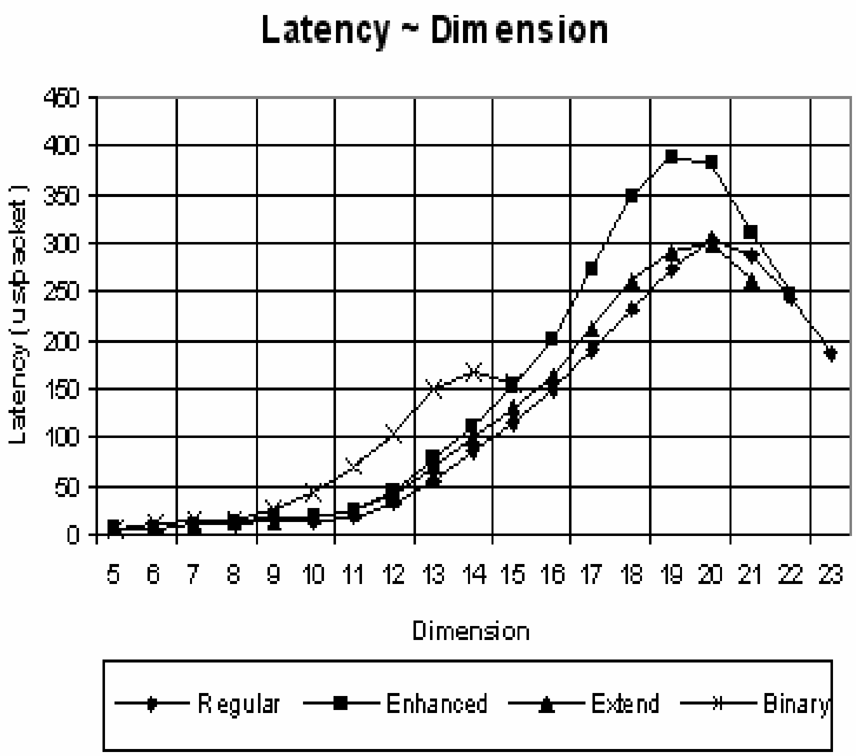
- Check false abortion
  - enumerated all possible locations of faulty components and (source, destination) pairs for three kinds of Fibonacci-class Cubes with dimensionality lower than 7. No false abortion occurs.
  - For higher dimensional cases, we can only randomly set faults and pick (source, destination) pairs. After one month's simulation on a 2.3 GHz CPU, still no false abortion occurs.

Experimental Results

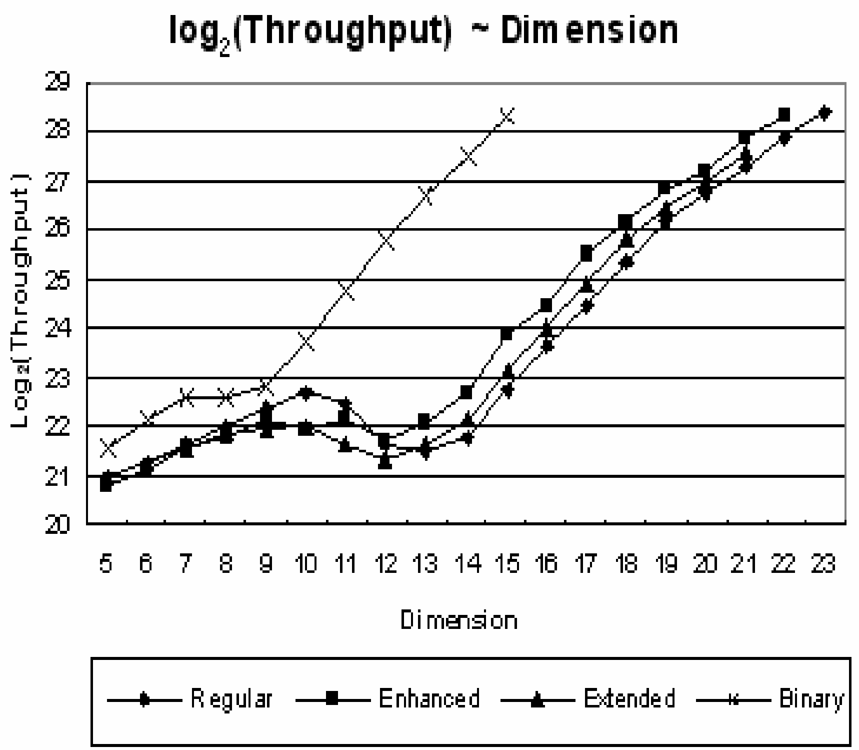
- Experimental settings
  - location of faults, source and destination are all randomly chosen by uniform distribution
  - a node is faulty when all of its incident links are faulty
  - fixed packet-sized messages
  - source and destination nodes must be non-faulty
  - eager readership is employed when packet service rate is faster than packet arrival rate

Experimental Results

- Comparison on various network sizes



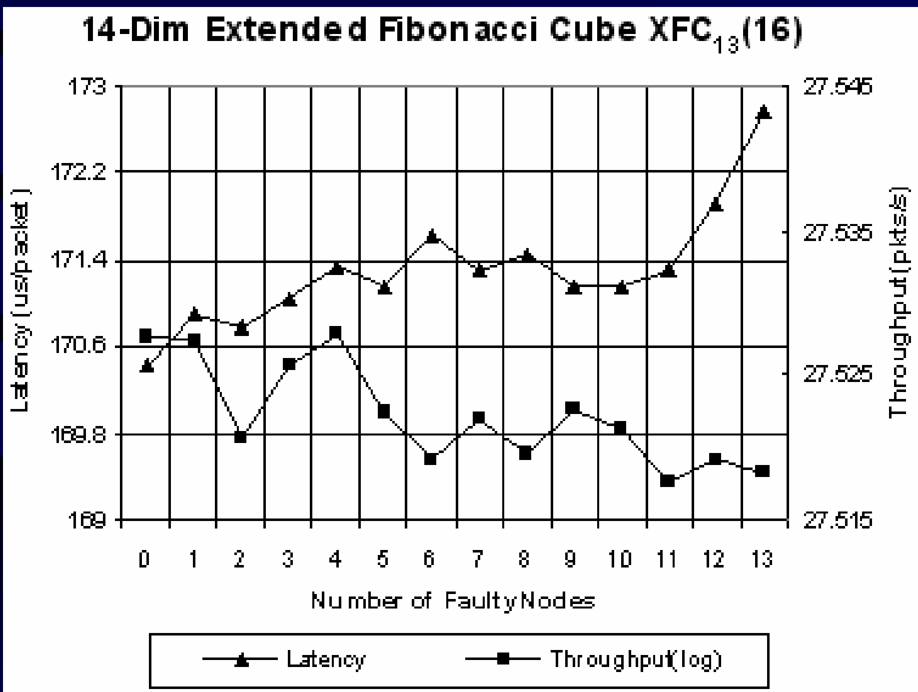
Latency of Fault-free Fibonacci-Class Cubes



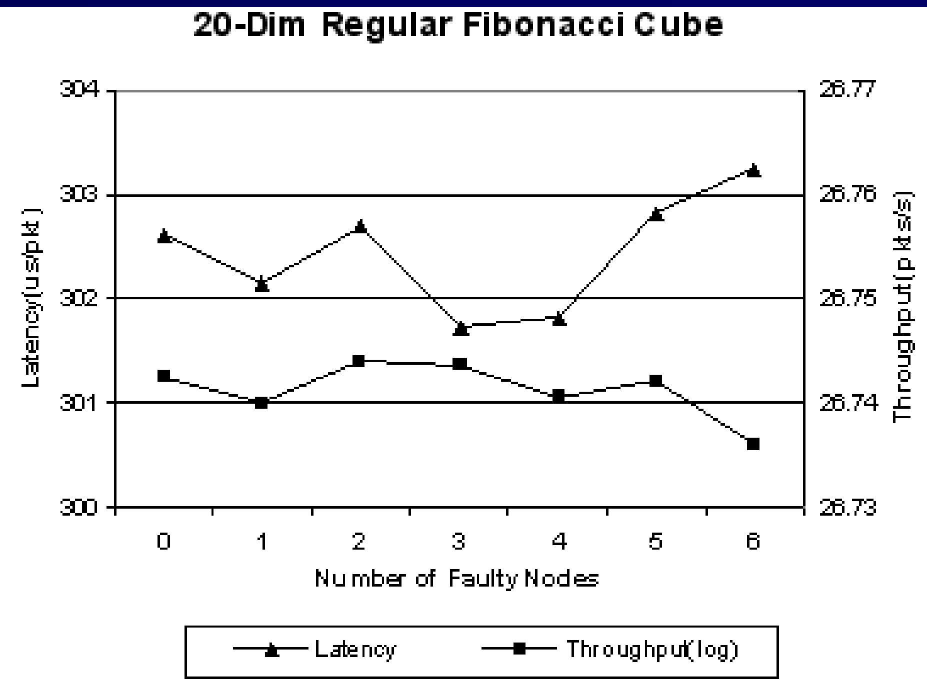
Throughput (logarithm) of Fault-free Fibonacci-class Cubes

Experimental Results

- Comparison on various numbers of faults



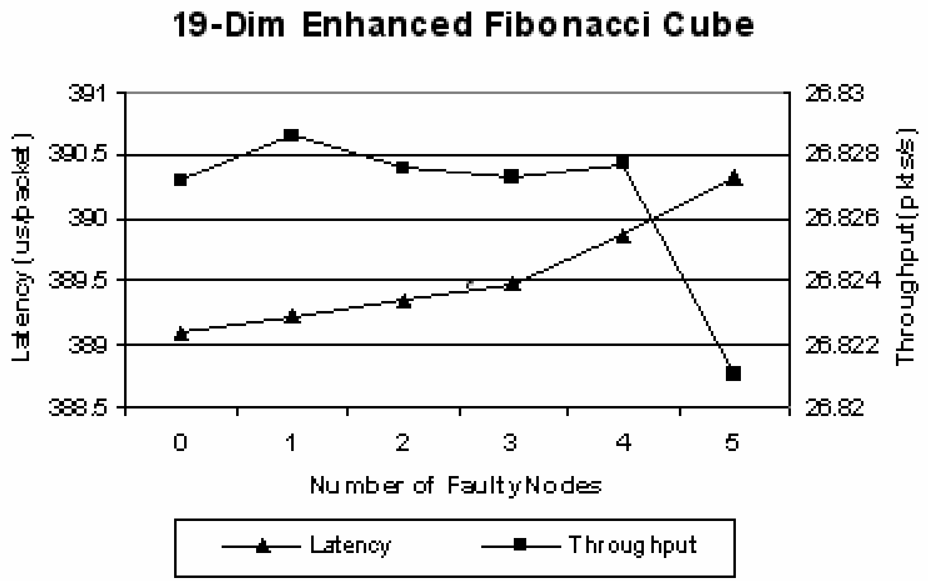
Latency and Throughput (logarithm)
of a faulty 14-dim Extended Fibonacci Cube



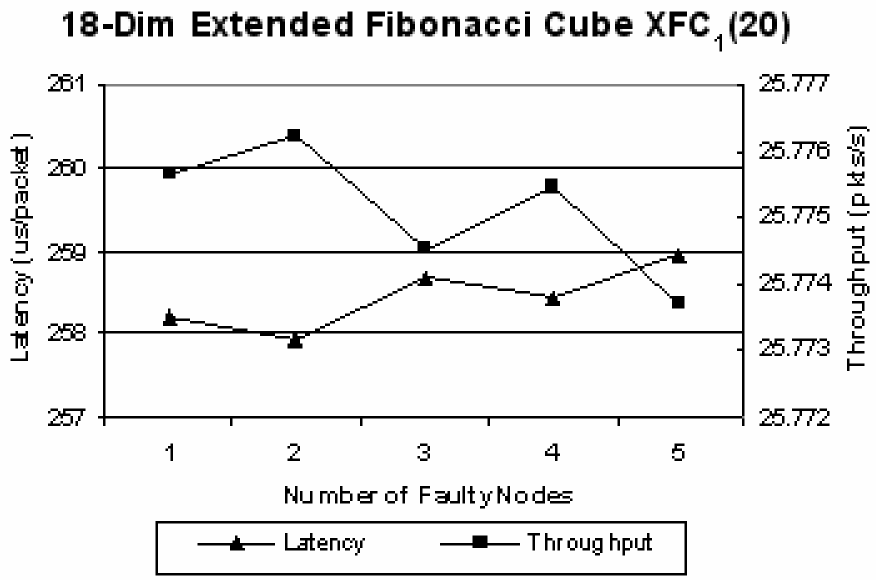
Latency and Throughput
(logarithm) of faulty 20-Dim FC

Experimental Results

- Comparison on various numbers of faults



Latency and Throughput (logarithm) of a faulty 19-Dim *EFC*



Latency and Throughput (logarithm) of a faulty 18-Dim *XFC*

Road Map

- Introduction
- Generic approach to cycle-free routing (GACR)
- Fault-tolerant Fibonacci routing (FTFR)
- Experimental results
- Conclusion and future work

Conclusion and future work

- Applicable to all Fibonacci-class cubes in a unified fashion.
- Although the Fibonacci-class cubes may be very sparsely connected, the algorithm can tolerate as many faulty components as the network node availability.
- The space and computation complexity as well as message overhead size are all moderate.
- Future: increase the number of faulty components tolerable, physical implementation.

Thank you !

Questions are welcomed.