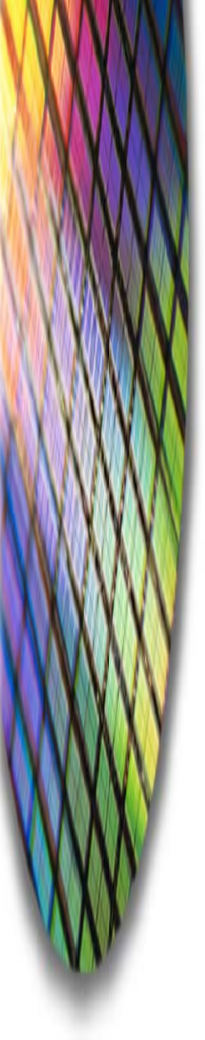


A3MAP: Architecture-Aware Analytic Mapping for Networks-on-Chip

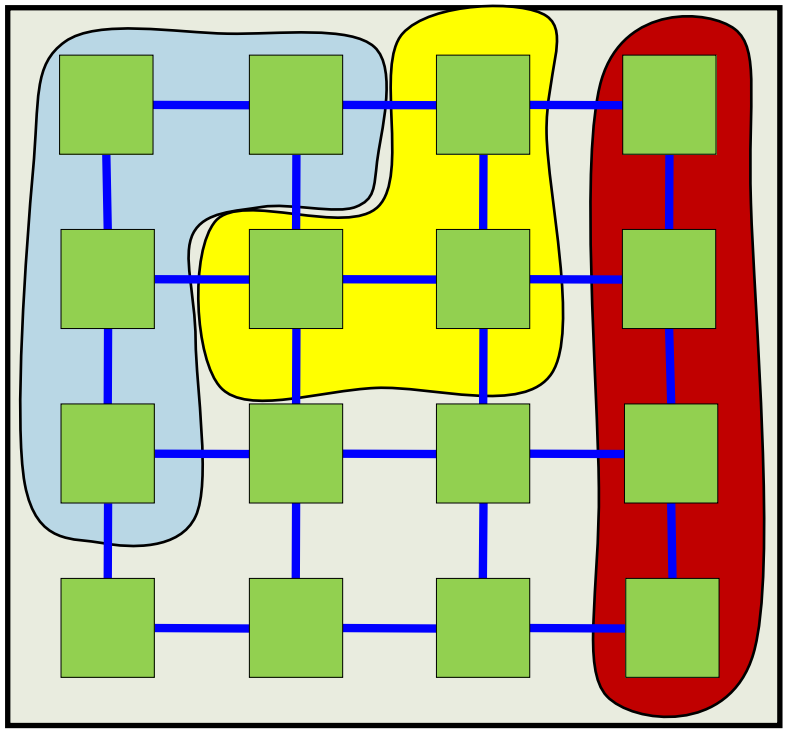
Wooyoung Jang and David Z. Pan*

Dept. of Electrical and Computer Engineering
University of Texas at Austin

Outline

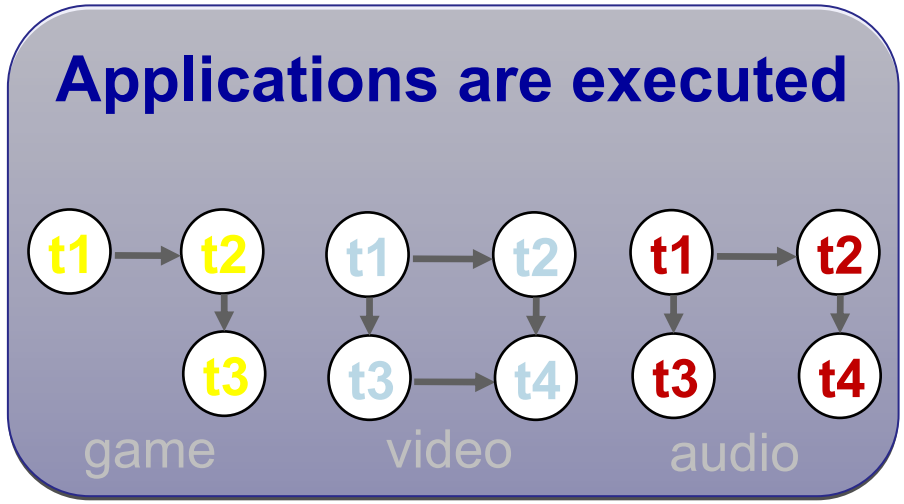
- Introduction – Task mapping in NoC
- A3MAP formulation
- A3MAP algorithms
  - A3MAP-SR (Successive Relaxation)
  - A3MAP-GA (Genetic Algorithm)
- Experimental results
- Conclusion

Task Mapping (Scheduling) in NoC

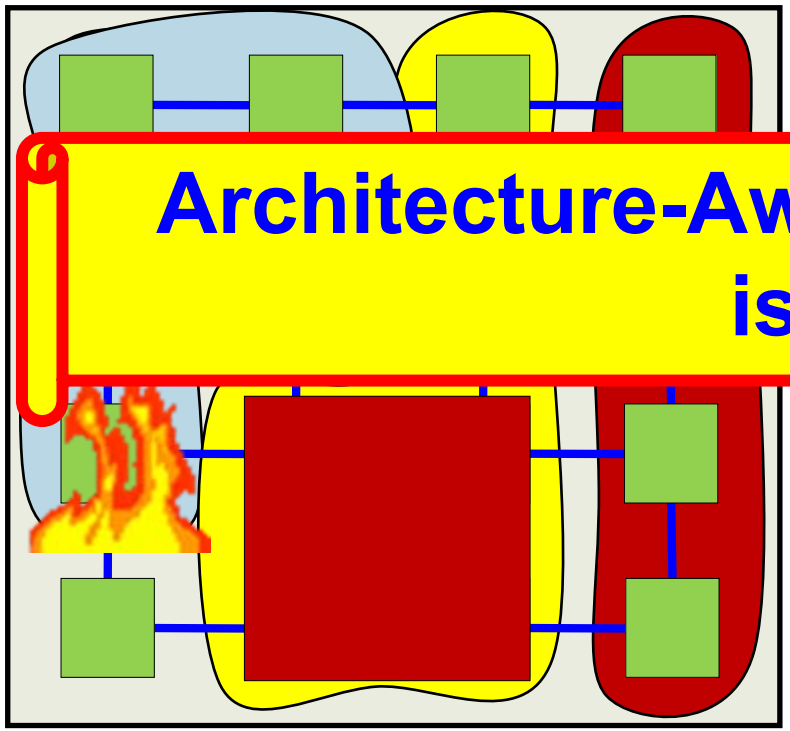


- Multi-processor System-on-Chip
  - **include homogeneous core**
    - Intel duo-, quad- and octal core

- Multiple applications are mapped on MPSoC with **regular mesh**



Task Mapping (Scheduling) in NoC



- Multi-processor System-on-Chip
  - include homogeneous core
    - Intel duo-, quad- and octal core
  - **include heterogeneous core**
    - TI OMAP
    - Philips Nexperia

Architecture-Aware Analytic Mapping is needed

- x Size of core is different
- x Some cores or links are faulty or degraded

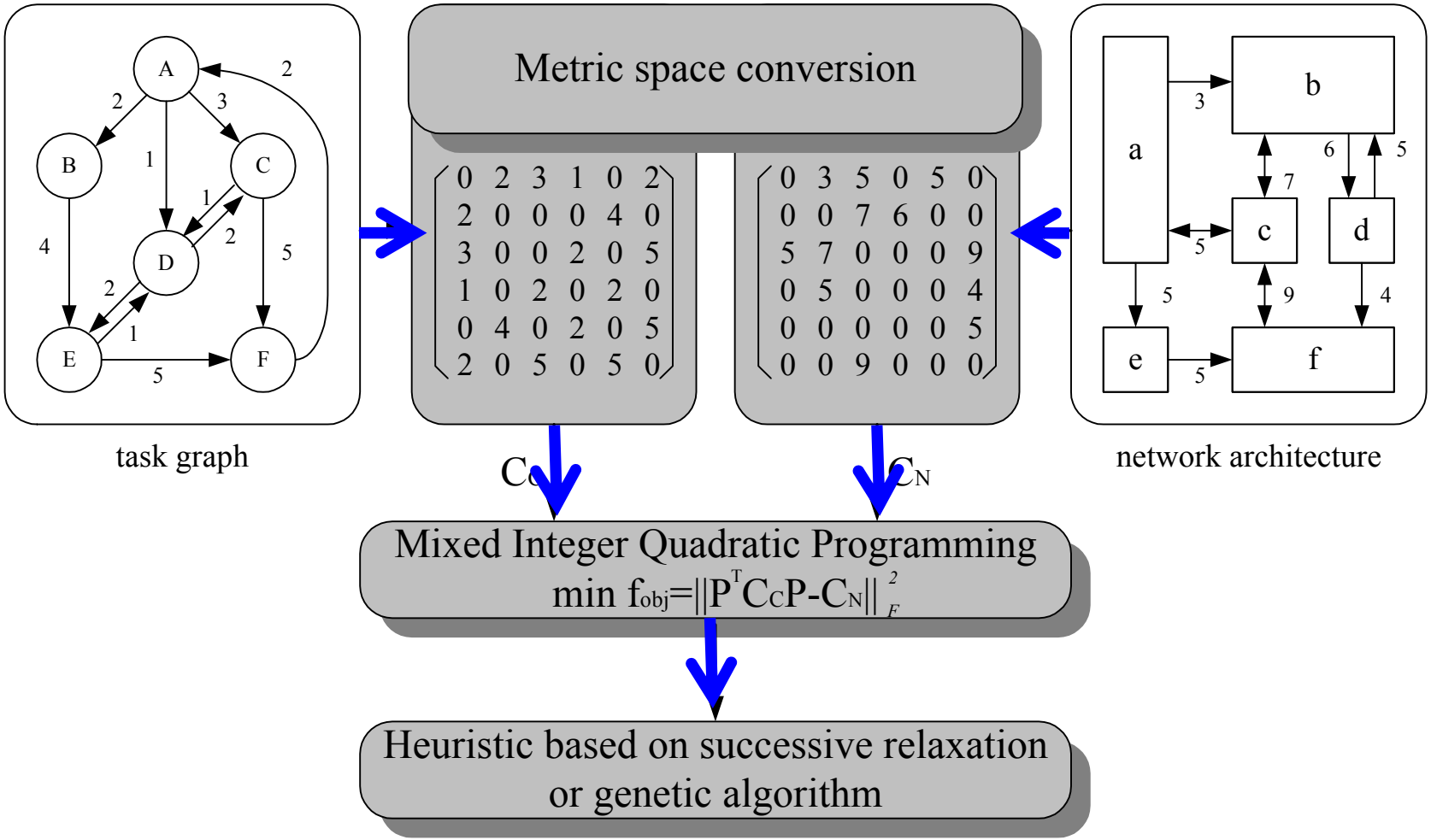
Related works

- Homogeneous core
  - J. Hu et al., ASPDAC'03: branch and bound algorithm
  - S. Murali and G. D. Micheli, DATE'04: a fast algorithm
  - D. Shin and J. Kim, CODES+ISSS'04: genetic algorithm
  - C. L. Chou and R. Marculescu, CODES+ISSS'07: run-time application mapping with multiple voltage levels
  - E.H.-M. Sha et al., TPDS'04: run-time mapping for small task graphs
  - E. Carvalho et al.'07: run-time heuristics but non-scalable
  - P.-C. Chang et al., DAC'08: dynamic voltage scaling and ant colony optimization algorithm
  - M. A. A. Faruque et al., DAC'08: distributed run-time application mapping
  - …

x **Not consider different link bandwidth, link direction and wirelengh of link**

A3MAP Methodology

- Overview of A3MAP



A3MAP: Metric Space Conversion

- Compose interconnection matrix
  - Graph with n vertex, G(V,E) where $v_i \in V$
  - $e_{i,j} \in E$ represents communication between v_i to v_j
  - $vol(e_{i,j})$ represents communication volume in a task graph
  - $bw(e_{i,j})$ represents bandwidth requirement in a network

$$
\begin{array}{c} \\ v_A \\ v_B \\ v_C \\ v_D \\ v_E \\ v_F \end{array}
\begin{array}{c}
\begin{array}{cccccc} v_A & v_B & v_C & v_D & v_E & v_F \end{array} \\
\begin{pmatrix}
0 & bw(e_{AB}) & bw(e_{AC}) & bw(e_{AD}) & bw(e_{AE}) & bw(e_{AF}) \\
bw(e_{BA}) & 0 & bw(e_{BC}) & bw(e_{BD}) & bw(e_{BE}) & bw(e_{BF}) \\
bw(e_{CA}) & bw(e_{CB}) & 0 & bw(e_{CD}) & bw(e_{CE}) & bw(e_{CF}) \\
bw(e_{DA}) & bw(e_{DB}) & bw(e_{DC}) & 0 & bw(e_{DE}) & bw(e_{DF}) \\
bw(e_{EA}) & bw(e_{EB}) & bw(e_{EC}) & bw(e_{ED}) & 0 & bw(e_{EF}) \\
bw(e_{FA}) & bw(e_{FB}) & bw(e_{FC}) & bw(e_{FD}) & bw(e_{FE}) & 0
\end{pmatrix}
\end{array}
$$

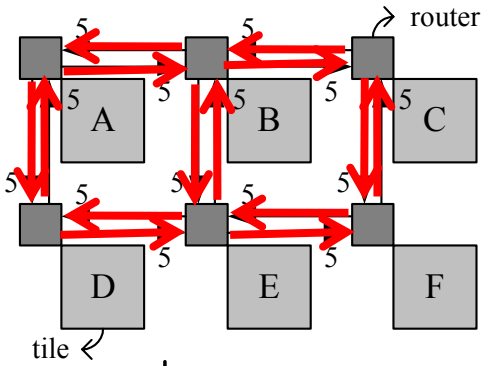
(a) Interconnection matrix, C_N

$$
\begin{array}{c} \\ v_A \\ v_B \\ v_C \\ v_D \\ v_E \\ v_F \end{array}
\begin{array}{c}
\begin{array}{cccccc} v_A & v_B & v_C & v_D & v_E & v_F \end{array} \\
\begin{pmatrix}
0 & vol(e_{AB}) & vol(e_{AC}) & vol(e_{AD}) & vol(e_{AE}) & vol(e_{AF}) \\
vol(e_{BA}) & 0 & vol(e_{BC}) & vol(e_{BD}) & vol(e_{BE}) & vol(e_{BF}) \\
vol(e_{CA}) & vol(e_{CB}) & 0 & vol(e_{CD}) & vol(e_{CE}) & vol(e_{CF}) \\
vol(e_{DA}) & vol(e_{DB}) & vol(e_{DC}) & 0 & vol(e_{DE}) & vol(e_{DF}) \\
vol(e_{EA}) & vol(e_{EB}) & vol(e_{EC}) & vol(e_{ED}) & 0 & vol(e_{EF}) \\
vol(e_{FA}) & vol(e_{FB}) & vol(e_{FC}) & vol(e_{FD}) & vol(e_{FE}) & 0
\end{pmatrix}
\end{array}
$$

(b) Interconnection matrix, C_C

A3MAP: Metric Space Conversion

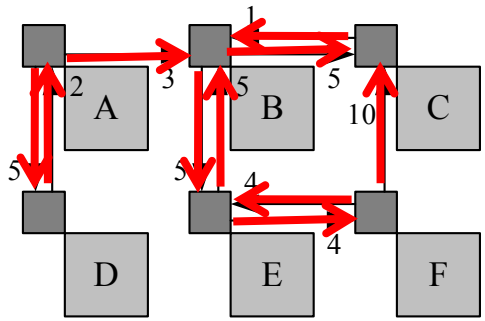
- Convert graph into Interconnection matrix



Metric space conversion

	v_A	v_B	v_C	v_D	v_E	v_F
v_A	0	5	0	5	0	0
v_B	5	0	5	0	5	0
v_C	0	5	0	0	0	5
v_D	5	0	0	0	5	0
v_E	0	5	0	5	0	5
v_F	0	0	5	0	5	0

(a) regular mesh network



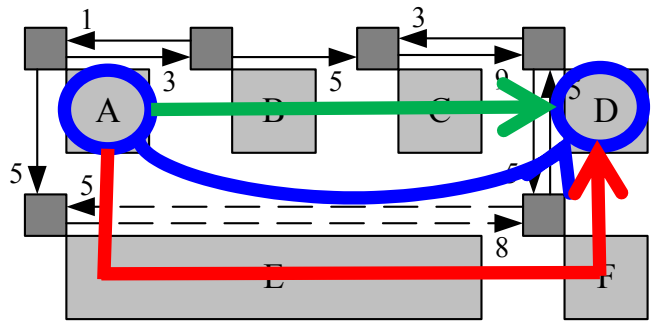
Metric space conversion

	v_A	v_B	v_C	v_D	v_E	v_F
v_A	0	3	0	5	0	0
v_B	0	0	5	0	5	0
v_C	0	1	0	0	0	0
v_D	2	0	0	0	0	0
v_E	0	5	0	0	0	4
v_F	0	0	10	0	4	0

(b) irregular mesh network

A3MAP: Metric Space Conversion

- Convert graph into Interconnection matrix



Metric space conversion

	v_A	v_B	v_C	v_D	v_E	v_F
v_A	0	3	0	0	5	0
v_B	1	0	5	0	0	0
v_C	0	0	0	9	0	0
v_D	0	0	3	0	0	5
v_E	0	0	0	0	0	$\alpha 8$
v_F	0	0	0	5	$\alpha 5$	0

(c) custom network

- Traffic A→D
- Two paths with same packet hop count
  - Path 1: A →B →C →D
  - Path 2: A →E →F →D
- Wirelength passed
  - Path 1: 3 units
  - Path 2: 5 units
- $E_{link}=E_{drivers}+E_{repeaters}$
- α (<1)=$E_{short\ link}/E_{long\ link}$

A3MAP: TASK Mapping Concept

- Most task graphs are different from mapped network graph
- Distortion of task graph, $P^T C_C P$ is not evitable in the network, where P is permutation matrix
- P(i,j)=1 means task j is mapped to tile i
- Minimization of difference between distortion of task graph and network graph

A3MAP: Problem Formulation

- Formulation of Objective

$$\min f_{obj} = \left\| P^T C_C P - C_N \right\|_F^2 = \left\| C_C P - P C_N \right\|_F^2$$

Find P to minimize f_{obj}

Subject to linear and integer constraints

$$\sum_{j=1}^{n} P(i,j) = 1, \forall i = 1,2,...,n$$

$$\sum_{i=1}^{n} P(i,j) = 1, \forall j = 1,2,...,n$$

$$P(i,j) \in \{0,1\}$$

- ✓ Convex quadratic objective function
- ✓ Solution space is discrete (i.e., non-convex)
- ✓ Mixed Integer Quadratic Programming (NP-hard)

A3MAP Algorithms

- A3MAP-SR (Successive Relaxation)
  - Run-time task mapping algorithm for MIQP
  - Relaxing $P(i,j) \in \{0,1\}$ to $0 \le P(i,j) \le 1$ ($\rightarrow$ QP)
  - Using QP solver
  - Guiding P(i,j) to 1 or 0

- A3MAP-GA (Genetic Algorithm)
  - Design-time task mapping algorithm for MIQP
  - Efficient random searching algorithm
  - Cycle crossover
  - Mutation operation

A3MAP- Successive Relaxation

- Relax $P(i,j) \in \{0,1\}$ to $0 \le P(i,j) \le 1$
- i_Threshold (i_T) = n(V) = 6
- QP solver → P =

$$\begin{pmatrix} 0.1 & 0.2 & 0.2 & 0.3 & 0.1 & 0.1 \\ 0.0 & 0.1 & 0.5 & 0.1 & 0.2 & 0.1 \\ 0.4 & 0.2 & 0.0 & 0.3 & 0.1 & 0.0 \\ 0.1 & 0.0 & 0.1 & 0.1 & 0.1 & 0.6 \\ 0.3 & 0.2 & 0.1 & 0.0 & 0.4 & 0.0 \\ 0.1 & 0.3 & 0.1 & 0.2 & 0.1 & 0.2 \end{pmatrix}$$

- Find i_{max}, j_{max} for max{P(i,j)}
  - if max{P(i_{max},j_{max})} > 1/i_T
    - P(i_{max},j_{max}) = 1
    - P(i,j_{max}) = 0, i = 1,2,..,n
    - P(i_{max},j) = 0, j = 1,2,..,n

A3MAP- Successive Relaxation

- Relax $P(i,j) \in \{0,1\}$ to $0 \le P(i,j) \le 1$
- i_Threshold (i_T) = n(V) = 5
- QP solver
- Find i_{max}, j_{max} for max{P(i,j)}
  - if max{P(i_{max},j_{max})} > 1/i_T
    - P(i_{max},j_{max}) = 1
    - P(i,j_{max}) = 0, i = 1,2,..,n
    - P(i_{max},j) = 0, j = 1,2,..,n
    - i_T = i_T – 1
  - end if

P =

0.1	0.2	0.2	0.3	0.1	0.0
0.0	0.1	0.5	0.1	0.2	0.0
0.4	0.2	0.0	0.3	0.1	0.0
0.0	0.0	0.0	0.0	0.0	1.0
0.3	0.2	0.1	0.0	0.4	0.0
0.1	0.3	0.1	0.2	0.1	0.0

A3MAP- Successive Relaxation

- Relax $P(i,j) \in \{0,1\}$ to $0 \le P(i,j) \le 1$
- i_Threshold (i_T) = n(V) = 4
- QP solver
- Find i_{max}, j_{max} for max{P(i,j)}
  - if max{P(i_{max},j_{max})} > 1/i_T
    - P(i_{max},j_{max}) = 1
    - P(i,j_{max}) = 0, i = 1,2,..,n
    - P(i_{max},j) = 0, j = 1,2,..,n
    - i_T = i_T – 1
  - end if

$$P = \begin{pmatrix} 0.1 & 0.2 & 0.0 & 0.3 & 0.1 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.4 & 0.2 & 0.0 & 0.3 & 0.1 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \\ 0.3 & 0.2 & 0.0 & 0.0 & 0.4 & 0.0 \\ 0.1 & 0.3 & 0.0 & 0.2 & 0.1 & 0.0 \end{pmatrix}$$

A3MAP- Successive Relaxation

- Relax $P(i,j) \in \{0,1\}$ to $0 \le P(i,j) \le 1$
- i_Threshold (i_T) = n(V) = 3
- QP solver
- Find i_{max}, j_{max} for max{P(i,j)}
  - if max{P(i_{max},j_{max})} > 1/i_T
    - P(i_{max},j_{max}) = 1
    - P(i,j_{max}) = 0, i = 1,2,..,n
    - P(i_{max},j) = 0, j = 1,2,..,n
    - i_T = i_T – 1
  - end if

$$P = \begin{pmatrix} 0.0 & 0.2 & 0.0 & 0.3 & 0.1 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \\ 0.0 & 0.2 & 0.0 & 0.0 & 0.4 & 0.0 \\ 0.0 & 0.3 & 0.0 & 0.2 & 0.1 & 0.0 \end{pmatrix}$$

A3MAP- Successive Relaxation

- Relax $P(i,j) \in \{0,1\}$ to $0 \le P(i,j) \le 1$
- i_Threshold (i_T) = n(V) = 2
- QP solver
- Find i_{max}, j_{max} for max{P(i,j)}
  - if max{P(i_{max},j_{max})} > 1/i_T
    - P(i_{max},j_{max}) = 1
    - P(i,j_{max}) = 0, i = 1,2,..,n
    - P(i_{max},j) = 0, j = 1,2,..,n
    - i_T = i_T – 1
  - end if

$$P = \begin{pmatrix} 0.0 & 0.2 & 0.0 & 0.3 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.3 & 0.0 & 0.2 & 0.0 & 0.0 \end{pmatrix}$$

A3MAP- Successive Relaxation

- Relax $P(i,j) \in \{0,1\}$ to $0 \le P(i,j) \le 1$
- i_Threshold (i_T) = n(V) = 2
- QP solver → P =

$$\begin{pmatrix} 0.0 & 0.7 & 0.0 & 0.3 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.3 & 0.0 & 0.7 & 0.0 & 0.0 \end{pmatrix}$$

- Find i_{max}, j_{max} for max{P(i,j)}
  - if max{P(i_{max},j_{max})} > 1/i_T
    - P(i_{max},j_{max}) = 1
    - P(i,j_{max}) = 0, i = 1,2,..,n
    - P(i_{max},j) = 0, j = 1,2,..,n
    - i_T = i_T – 1
  - end if

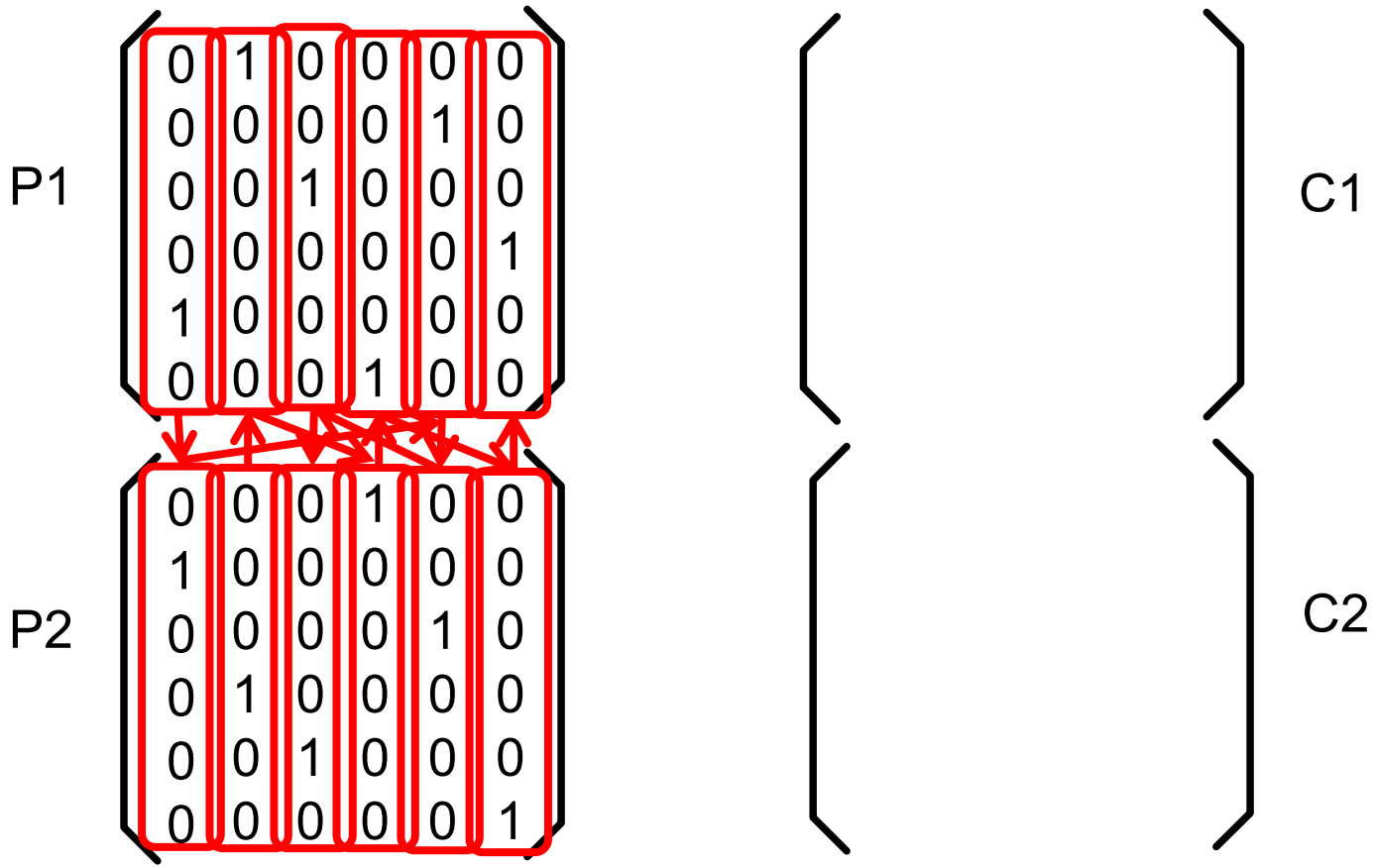
A3MAP- Successive Relaxation

- Relax $P(i,j) \in \{0,1\}$ to $0 \le P(i,j) \le 1$
- i_Threshold (i_T) = n(V) = 1
- QP solver
- Find i_{max}, j_{max} for max{P(i,j)}
  - if max{P(i_{max},j_{max})} > 1/i_T
    - P(i_{max},j_{max}) = 1
    - P(i,j_{max}) = 0, i = 1,2,..,n
    - P(i_{max},j) = 0, j = 1,2,..,n
    - i_T = i_T – 1
  - end if

$$P = \begin{pmatrix} 0.0 & 0.1 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 1.0 & 0.0 & 0.0 & 0.0 \\ 0.1 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1.0 & 0.0 & 0.0 \end{pmatrix}$$

A3MAP-Genetic Algorithm

- Individual parent, P1 with arbitrary gene
- Individual parent, P2 with arbitrary gene
- (C1, C2) = Cycle crossover (P1, P2)



A3MAP-Genetic Algorithm

- Individual parent, P1 with arbitrary gene
- Individual parent, P2 with arbitrary gene
- (C1, C2) = Cycle crossover (P1, P2)
- Mutation (C1)
  - Keep mutation if f_{obj} reduce
- Mutation (C2)
  - Keep mutation if f_{obj} reduce
- P1 = minimum f_{obj} between C1 and C2

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \quad C1$$

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{pmatrix} \quad C2$$

Experimental Results

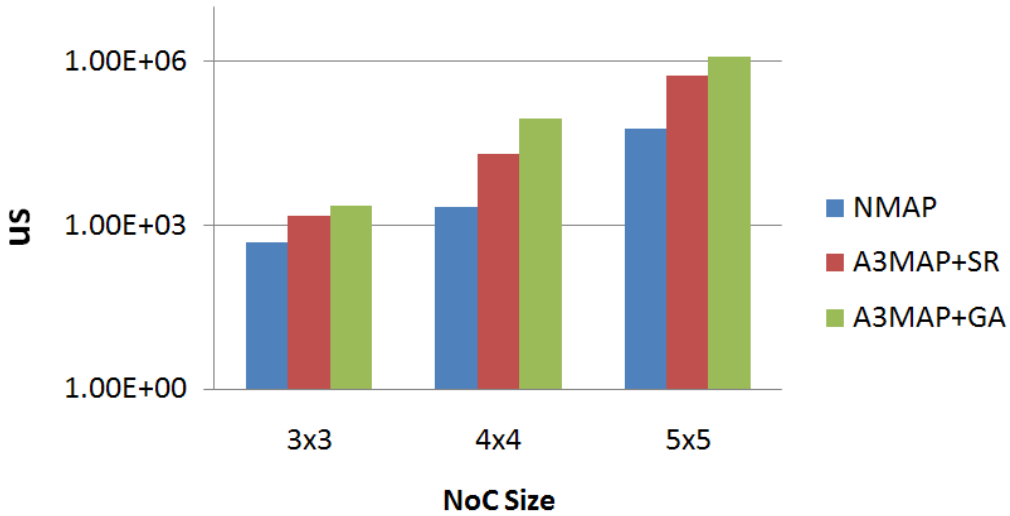
- A3MAP-SR implemented in CPLEX11.2
- A3MAP-GA implemented in C++
- Compared to NMAP [S. Murali, DATE'04]
- Benchmark
  - MPEG-4 video object plane decoder (mapped on 4x4)
  - E3S suites (mapped on 3x3 to 5x5)
- Intel 2.4GHz CoreDuo and 8GB RAM

Experimental Results – Regular Mesh

- Industrial benchmark for packet h comparison

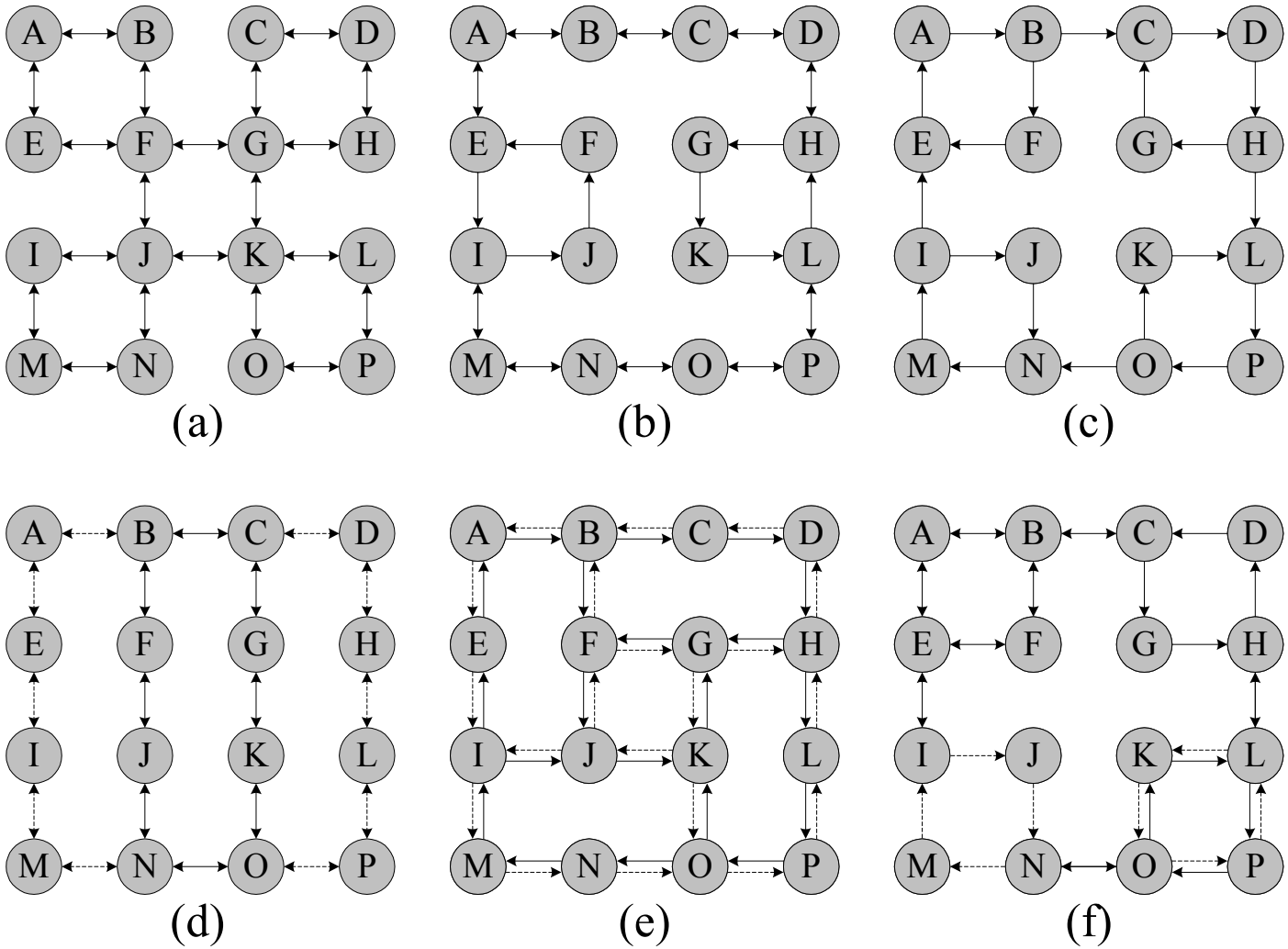
Application	NMAP	A3MAP-SR	imp. (%)	A3MPA-GA	imp. (%)
consumer	50	50	0	49	2
VOPD	4309	4265	1.0	4141	3.9
AI	187	151	19.3	147	21.4
telecomm	127	115	9.4	102	19.7

- For random benchmark **5.7%** and **8.8%** fewer traffic
- Runtime



Experimental Results – Irregular Mesh

- Irregular mesh architectures experimented



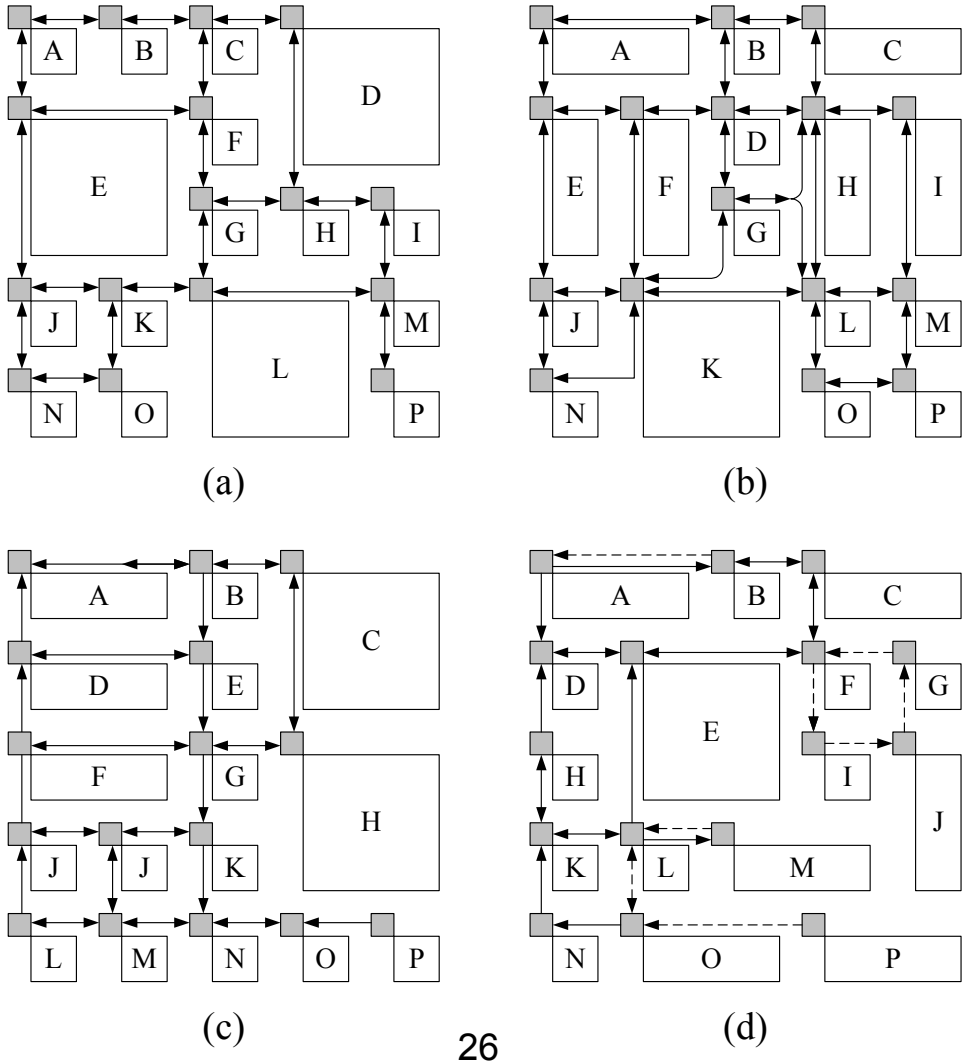
Experimental Results – Irregular Mesh

- Packet hop count comparison on MPEG-4 VOPD
  - Improvement of A3MAP-SR and -GA is 1% and 3.9%, respectively, in regular mesh network.

network	NMAP	A3MAP-SR	imp. (%)	A3MAP-GA	imp. (%)
(a)	4869	4839	0.6	4237	13
(b)	5699	4619	19	4457	21.8
(c)	7810	7317	6.3	4619	40.9
(d)	4923	4301	12.6	4295	12.8
(e)	5706	4199	26.4	4183	26.7
(f)	8103	4844	40.2	4410	45.8
average	**6185**	**5187**	**16.1**	**4367**	**29.4**

Experimental Results – Custom NoC

- Custom architectures experimented



Experimental Results – Custom NoC

- Total packet hop count

network	NMAP	A3MAP-SR	imp. (%)	A3MAP-GA	imp. (%)
(a)	4488	4531	-1	4087	8.9
(b)	4264	4248	0.4	4199	1.5
(c)	6296	5867	6.8	5150	18.2
(d)	5524	4263	22.8	4263	22.8
average	**5143**	**4727**	**7.3**	**4425**	**12.9**

- Total wirelength passed by packets

network	NMAP	A3MAP-SR	imp. (%)	A3MAP-GA	imp. (%)
(a)	5879	5332	9.3	4543	22.7
(b)	5505	5049	9.4	4215	23.4
(c)	7835	7434	5.1	5613	28.4
(d)	9196	5170	43.8	5170	43.8
average	**7104**	**5746**	**16.6**	**4885**	**29.6**

Conclusion

- Architecture-aware task mapping based on analytical approach
- Applied on all kinds of networks such as regular mesh, irregular mesh and custom network
- Applied on faulty and degraded network or voltage-frequency island based NoC
- Applied on both run-time and design-time task mapping application