Application Mapping for Network-on-Chip

Vaughan Kitchen

September 26, 2019

Network-on-Chip

- Topologies
  - mesh
  - torus
  - octagon
  - hypercube
  - fat tree
  - butterfly
  - symmetric Clos
- Routing
  - Dimension-Ordered (DOR) / XY
  - Valiant Load-Balancing (VAL)
  - O1TURN
- **Application Mapping**
  - **NMAP**
  - **LMAP**
  - **PSMAP**
  - **ILP**

Application Example - Dining Philosophers

```
PAR
    Forks
    Room
    PAR i = [0 FOR 4]
        Philosopher( i )
```

```
PROC Philosopher(VALUE identity) =
    WHILE TRUE
        SEQ
            Think
            Enter[ identity ]! ANY
            PickUp[ identity ]! ANY
            PickUp[ identity +1 MOD 5]!ANY
            Eat
            PutDown[identity+1 MOD 5]!ANY
            PutDown[identity]! ANY
            Exit [ identity ]! ANY :
```

Application Example - Dining Philosophers Continued

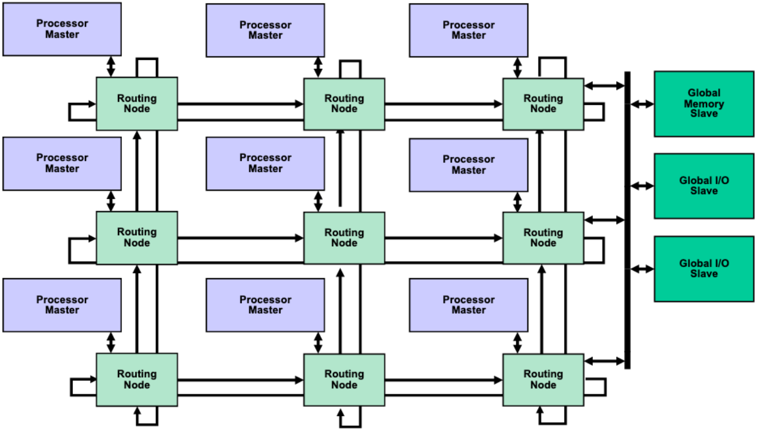
```
PROC Room =
    VAR
        Count :
    SEQ
        Count := 0
        WHILE TRUE
            ALT i = [0 FOR 4]
                Count<4 & Enter[i]?ANY
                    Count := Count + 1
                Exit[i]?ANY
                    Count := Count - 1 :
```

Application Example - Dining Philosophers Continued

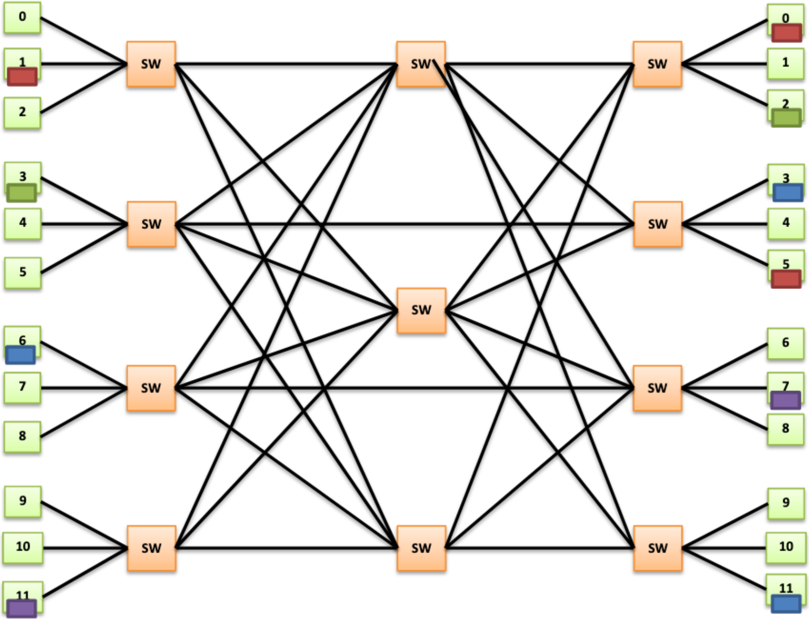
```
PROC Forks =
    VAR
        Free [4]  :
    WHILE TRUE
        ALT i = [0 FOR 4]
            Free [ i ]  & PickUp[i]?ANY
                Free [ i ]  := FALSE
            PutDown[i]?ANY
                Free [ i ]  := TRUE :
```

```
CHAN PickUp[4], PutDown[4], Enter[4], Exit [4]  :
```

Network-on-Chip Mesh Example



Network-on-Chip Clos Example



Application Graph Example

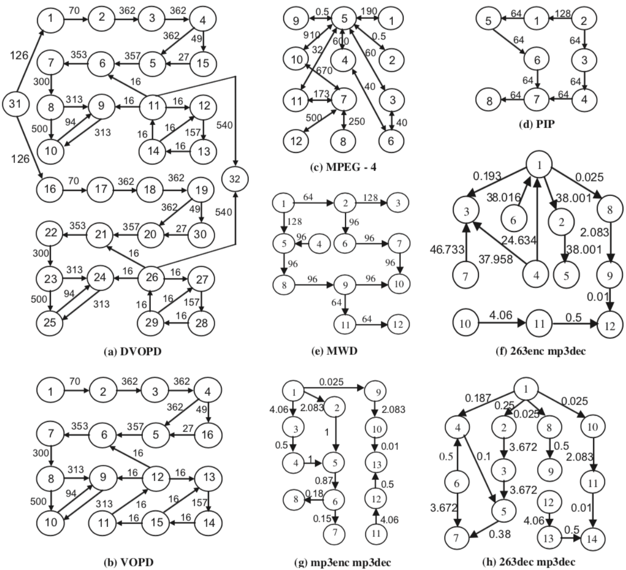
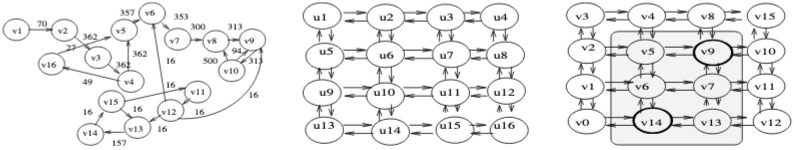


Fig. 12. Application core graphs with communication bandwidth (MB/s).

Application Graph Example Definitions

- VOPD - Video Object Plane Decoding
- DVOPD - Dual Video Object Plane Decoding
- MWD - Multi-Window Displayer
- PIP - Picture-in-Picture
- MP3 - MPEG-1 Audio Layer III
- MPEG-4 - Moving Pictures Expert Group

Application Graph Example



(a) Core graph

(b) NoC graph

(c) Mapping

Figure 2. Mapping of Core graph onto NoC graph

Problems

- Nodes can have more than 4 connections
- Nodes can have uneven chain lengths (wrap?)
- More application nodes than hardware nodes
- Connections between early and late nodes
- Unbalanced cores or links (big.LITTLE)
- Mapping is NP-hard
- **What does the best mapping mean?**

Solution

ad hoc topology

Actual solution

- On-line mapping has too much overhead
- Use a static mapping algorithm
  - NMAP - New Map?
    best performing at time of introduction
  - LMAP - Kernighan-Lin (K-L) partitioning Map?
    performs better than NMAP
  - PSMAP - Particle Swarm Map
  - ILP - Integer Linear Programming
    optimal solution
  - others
  - (most algorithms only available for mesh networks)

NMAP

1. initial mapping
2. minimum path computations
3. repeat 2. with vertices pair-wise swapping

NMAP - Continued

Initial mapping

1. the core with maximum communication placed onto maximally connected node
2. repeatedly add unmapped cores with maximal communication to mapped cores
3. place on node which minimizes cost with mapped cores (hop-count * bandwidth)

NMAP - Continued

Iterative improvement - minimum path

1. form quadrant graph with source node in center
2. destination node, and all nodes in shortest path will fall in one quadrant
3. compute Dijkstra's shortest path algorithm using nodes of this quadrant paying attention to bandwidth in path computation
4. repeatedly pairwise-swap and compute shortest path computations looking for a minima

LMAP

Initial mapping

1. add dummy nodes if number of nodes is not a power of 2
2. repeatedly perform K-L bi-partitioning (which calculates closeness of cores based on bandwidth)
3. partitioning forms a hierarchical grouping of cores
4. split the mesh in half alternating vertically and horizontally (bi-partitioning)
5. allocate the cores to the nodes following the splits (most connected cores towards the centre)

LMAP - Continued

Iterative improvement

1. for each partitioning level, partitioning is flipped and costs recomputed. the best cost is carried forward
2. partitioning runs from strongest connected nodes outwards to the full node set
3. once flipping completes dummy nodes are removed

PSMAP

1. performs Particle Swarm Optimization (PSO)
2. nodes are numbered. cores randomly assigned to nodes for each initial particle
3. local best (best the particle has seen), and global best saved each generation
4. second generation is evolved by randomly swapping cores
5. swap sequence considered as a series of swaps
6. swap sequence to personal best, or global best applied with random probability

ILP

- Integer linear programming
- Formulated as 0-1 ILP
- Used Xpress-MP to solve

Algorithm Comparison

Mapping algorithm	**VOPD**		**MPEG-4**		**PIP**	
	Comm. cost	CPU in s.	Comm. cost	CPU in s.	Comm. cost	CPU in s.
NMAP	4265.0	0.024	3672.0	0.016	640.0	0.010
LMAP	4189.0	0.040	4006.0	0.040	640.0	0.010
PSMAP	4119.0	0.260	3567.0	0.040	640.0	0.010
ILP	4119.0	4474.730	3567.0	21.530	640.0	1.280

References

- http://prog.vub.ac.be/~tjdhondt/ESL/CSP_to_OCCAM_files/Section%2013%20-%20CSP%20to%20OCCAM.pdf
- https://www.cs.otago.ac.nz/cosc402/lectures/lecture9.pdf
- Pradip Kumar Sahu and Santanu Chattopadhyay. 2012. A survey on application mapping strategies for Network-on-Chip design. http://dx.doi.org/10.1016/j.sysarc.2012.10.004
- Sulyman Tosun et al. 2009. An ILP formulation for application mapping onto Network-on-Chips. https://doi.org/10.1109/ICAICT.2009.5372524
- Pradip Kumar Sahu et al. 2011. Application Mapping onto Mesh Structured Network-on-Chip using Particle Swarm Optimization. https://doi.org/10.1109/ISVLSI.2011.21
- Srinivasan Murali and Giovanni De Micheli. 2004. Bandwidth-Constrained Mapping of Cores onto NoC Architectures. https://doi.org/10.1109/DATE.2004.1269002
- Pradip Kumar Sahu et al. 2010. A New Application Mapping Algorithm for Mesh based Network-on-Chip Design. https://doi.org/10.1109/INDCON.2010.5712700

Questions?