The Systolic Reconfigurable Mesh

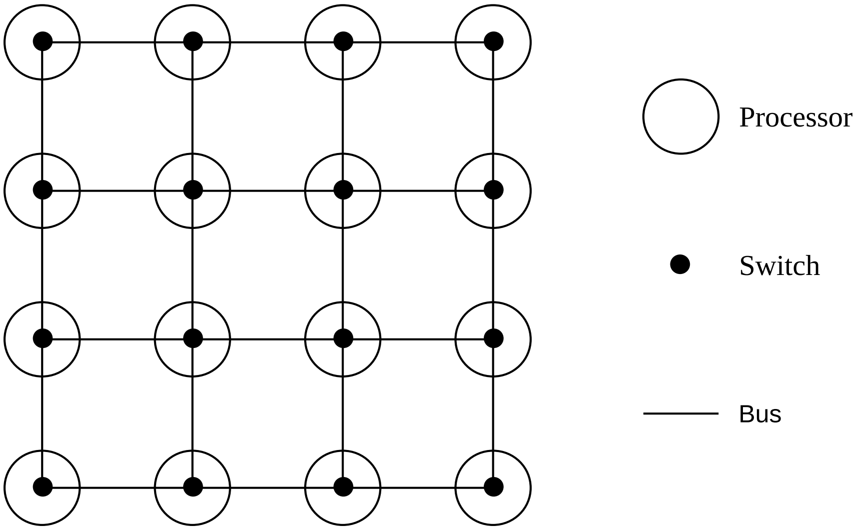
Mary M. Eshaghian

New Jersey Institute of Technology

Russ Miller

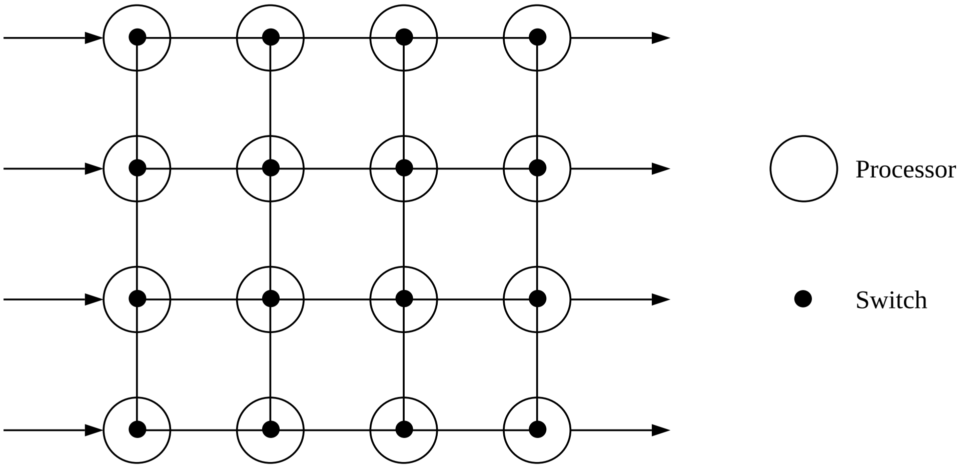
State University of New York at Buffalo

The Reconfigurable Mesh



- $n \times n$ array of processors
- Each processor has *dynamic* and *local* control over its switch setting
- Unit-delay broadcast
- Options for switch model
- Options for communication and processor computation model

The Systolic Reconfigurable Mesh



- Practical model
- Restricted domain
- Input from the left & Output to the right
- Phase 1: *Input and Preprocessing*
- Phase 2: *Static*
- Phase 3: *Output and Preprocessing*

Designing Algorithms for the SRM

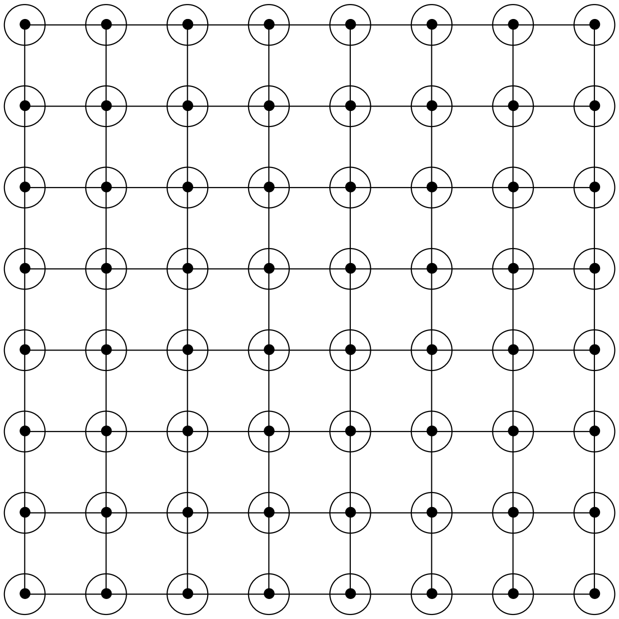
- Minimize constants
- Preserve the systolic nature of the process
- Eliminate the static stage
- Starting point:
  - Simulate mesh algorithms
  - Simulate reconfigurable mesh algorithms

Histogram

Input: all values in $1 \ldots n$

Output: result of i maintained in $SRM(i, n)$

1. Input the next column of data
2. Sort these n data items in $\Theta(1)$ time
3. Bus split in column and sum number of items of each value
4. Broadcast partial sums to diagonal
5. Broadcast from diagonal to proper row
6. Row broadcast and add to running sum

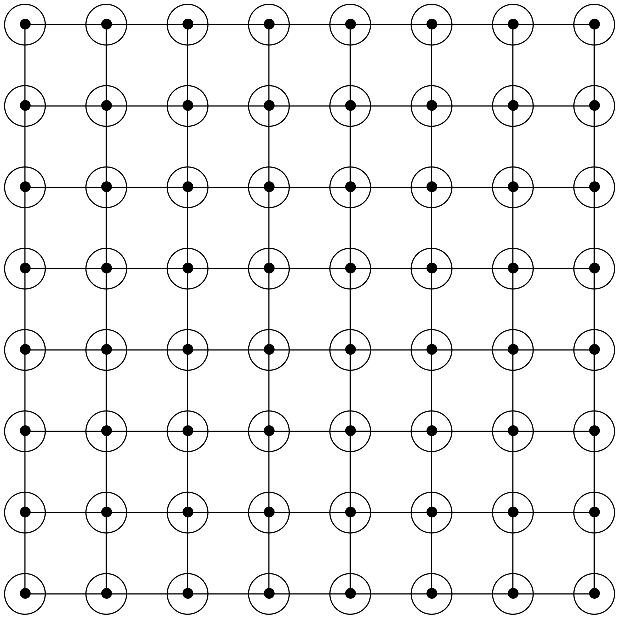


Convex Hull

Input: $n \times n$ binary image

Output: ‘marked’ image

1. Shift image to right and input next column
2. Bus split in column 1 to identify N and S pixels
3. Mark N and S as extreme points
4. Every marked pixel decides whether or not it is still an extreme point
5. Identify and record extreme points preceding N and S



Component Labeling

Input: $n \times n$ binary image

Output: labeled image

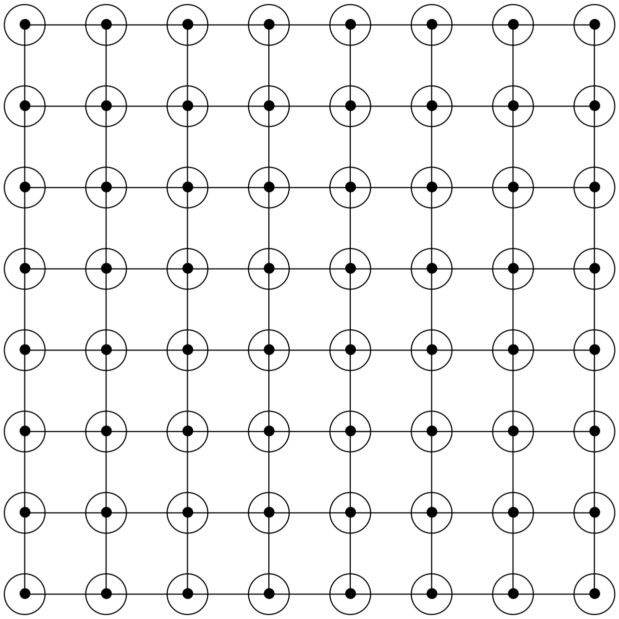
Labels: $< C_L, C_R, C_T >$

Flow: in from left, out to top

1. Input and Preprocessing:
   (a) Lockstep shift to right
   (b) Create connected subbus over every figure
   (c) Broadcast C_L
   (d) Bit-polling to resolve C_R
2. Output and Postprocessing:
   (a) Create connected subbus over every figure
   (b) If $C_T = 0$ then broadcast row label as C_T
   (c) Lockstep shift up

Input and Preprocessing step: $\Theta(\log n)$ time.

Output and Postprocessing step: $\Theta(1)$ time.



Summary of Results

Problem	Model	Static stage	Time complexity of a cycle	Total number of cycles
Histogram	Word	No	O(1) (includes sorting)	N+1
	Bit	No	O(log N)	N+1
Connex Hull	Word	No	O(1)	2N
	Bit	No	O(log N)	2N
Min/Max	Word	No	O(1)	N+1
	Bit	No	O(log N)	N+1
Labeling	Word	Yes	O(1)	3N
	Bit	No	O(log N)	2N
	Word	No	O(1) (includes sorting)	2N
(Restricted Image)	Word	No	O(1)	2N

Conclusion

1. Practical Model

2. Heterogeneous Computing

   (a) Image Understanding Architecture

   (b) Integrated Stand-Alone

3. Not General Purpose