

# A Low Level Analysis of Cellular Automata and Random Boolean Networks as a Computational Architecture

MS Thesis presentation by  
**Prateen Damera**

Advisor: Dr. Christof Teuscher

Electrical and Computer Engineering  
Portland State University

# Outline

- Motivation and Challenges
- Background
- My Contributions
- Methodology
  - Design
  - Evaluation
- Results
- Conclusion

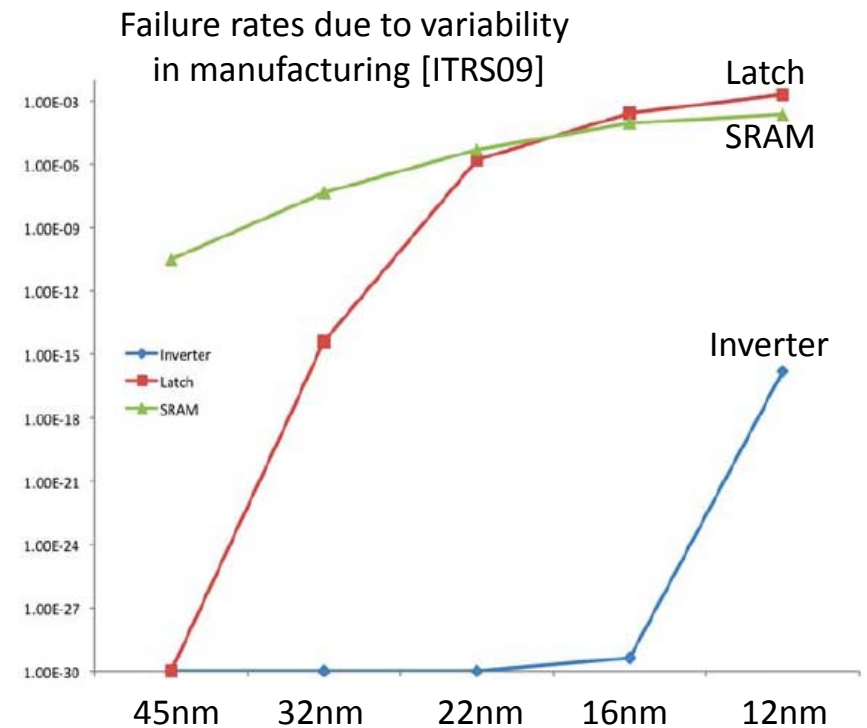
# Motivation (1)

- CMOS scaling physically limited by:

- High failure rates
- Variations in manufacturing process
- Quantum and thermal errors

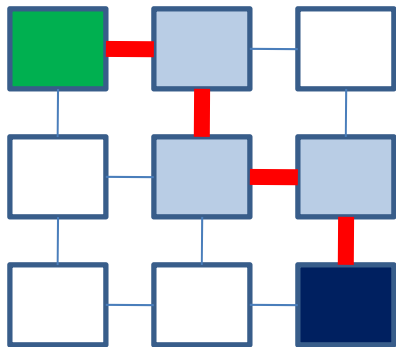
- **Self-assembled architectures**

- Embrace these limitations.
- Made up of a large number of small nodes connected together.
- Follow simple local rules to perform a global task, which can make them inherently robust and scalable.

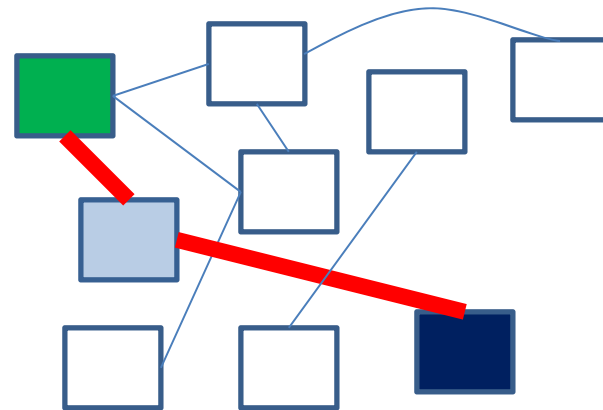


# Motivation (2)

- Regular arrangement of nodes with local connections result in better power dissipation distribution. These types of networks are called Cellular Automata [Zhirnov08].
- Information propagation is important for solving real world tasks. It is measured as the **average shortest path** between any two nodes.



Mesh network (local interconnects only)



Unstructured network (global interconnects)

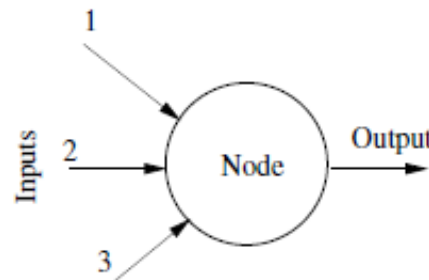
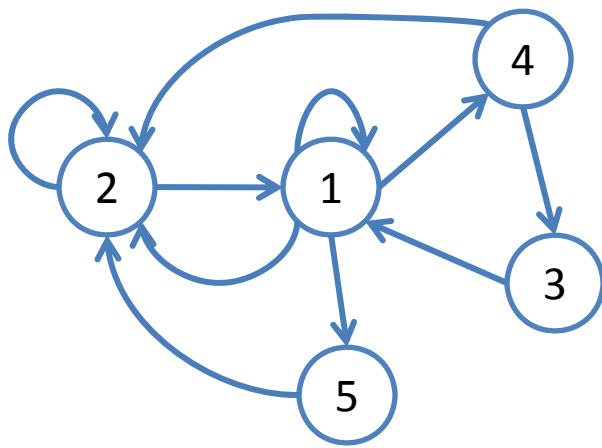
- An analysis of the power consumption and information propagation capabilities allow designers to make performance related decisions.

# Challenges

- Hardware-level model to understand scaling of area and power consumption with network parameters.
- Identification of the topology of the network to determine the local rules of each node.
- Configuration of the large number of nodes with the appropriate set of local rules.
- Determining network topologies that are capable of producing desired power and information propagation performance.

# Random Boolean Networks (RBN)

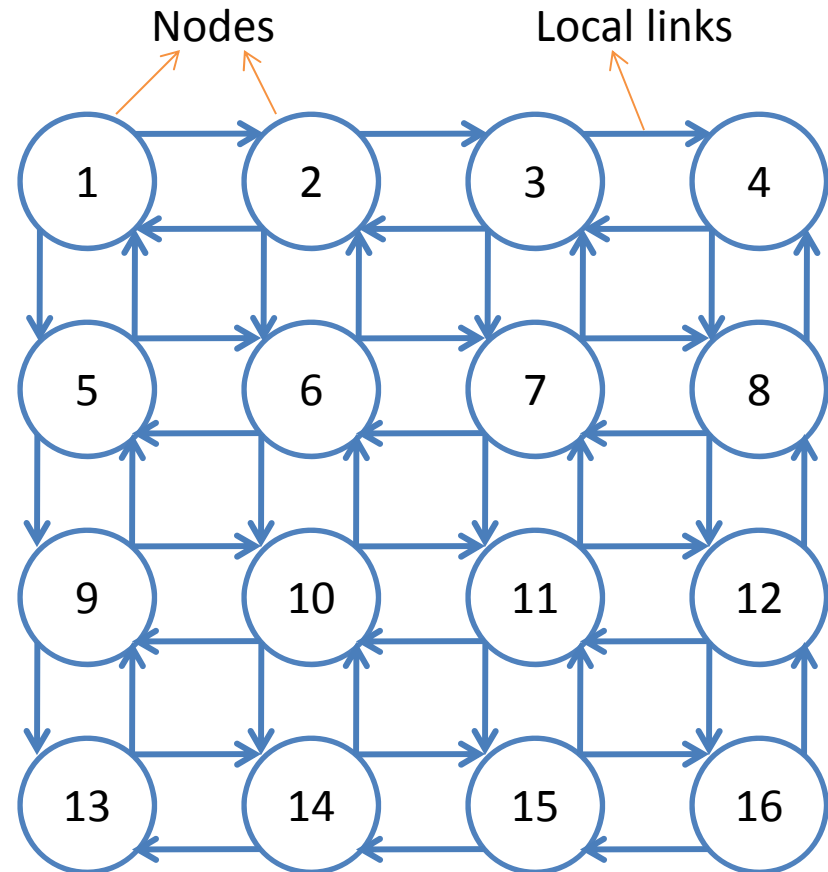
- Random arrangement of nodes, connected using randomly distributed links, and performing random functions [Kauffman93].
- Network parameters:
  - Number of nodes in network (N).
  - Average connectivity (k), also called average fan-in of the nodes.
- Output at current time step is a function of the combination of inputs at the previous time step.
- The networks are globally synchronized.



Input combination Inputs 3,2,1	Output
0,0,0	0
0,0,1	1
0,1,0	1
0,1,1	1
1,0,0	0
1,0,1	0
1,1,0	1
1,1,1	0

# Cellular Automata (CA)

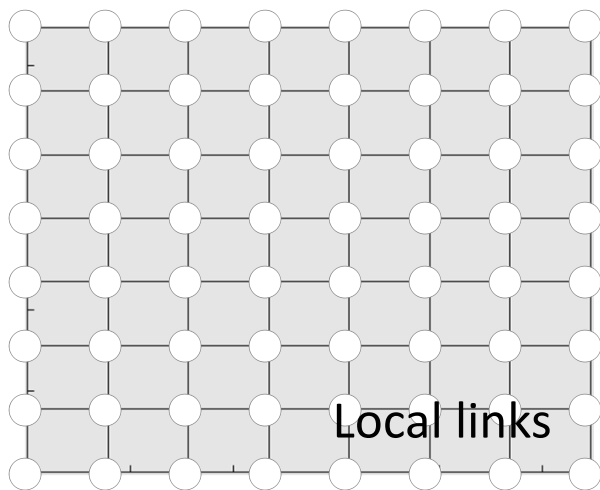
- Special case of random Boolean networks where the nodes are connected in the form of a regular mesh.
- Only local connections to immediate neighbors exist.
- Average shortest path between any two nodes of the network is longer than RBNs.
- Provides well distributed power consumption at the macro level.



# Small-World Networks

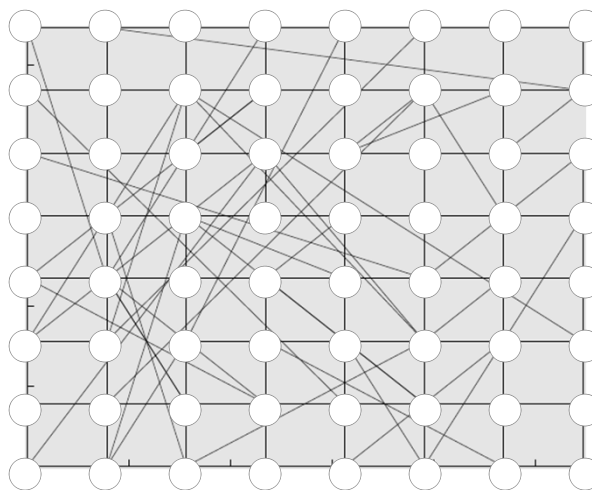
- A set of networks that lie between regular and random networks.
- Formed by the controlled re-wiring of a regular structure to create long range connections and reducing the average shortest path between nodes [Watts98, Kleinberg00, Petermann06].
- Parameters involved:
  - $R$  – Number of additional local links added to maintain connectivity.
  - $P$  – Probability of a link being chosen for re-wiring. Regular  $\leftrightarrow$  Irregular.
  - $\alpha$  – Factor for length distribution determined by power law function  $l^{-\alpha}$ . Local  $\leftrightarrow$  Global.

Initial mesh network



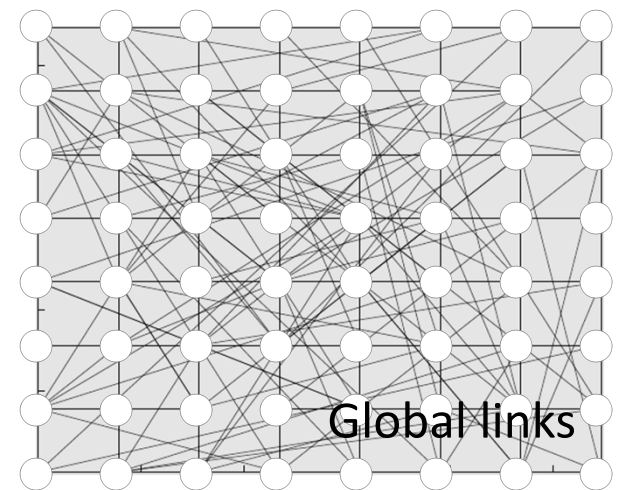
April 15, 2013

$R=100, p=50\%, \alpha=1$



MS Thesis Presentation - Prateen Damera

$R=100, p=50\%, \alpha=0$



8



# Goal of the Thesis

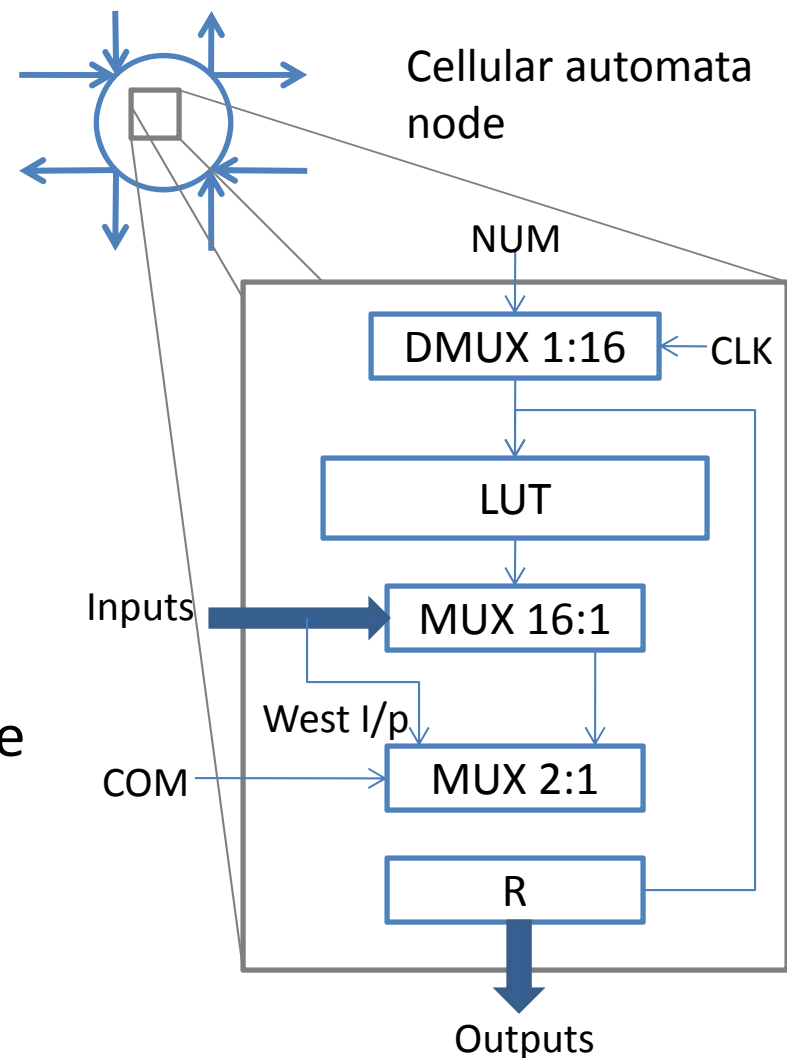
- Develop a framework to generate hardware models for CA, RBNs, and small-world networks of desired network parameters.
- Develop and implement generic algorithms to identify and configure these networks.
- Develop an evaluation framework to estimate area, power, and information propagation capabilities of the generated networks.
- Use the frameworks to obtain the network parameters that result in optimum performance for a desired application.

# My Contributions

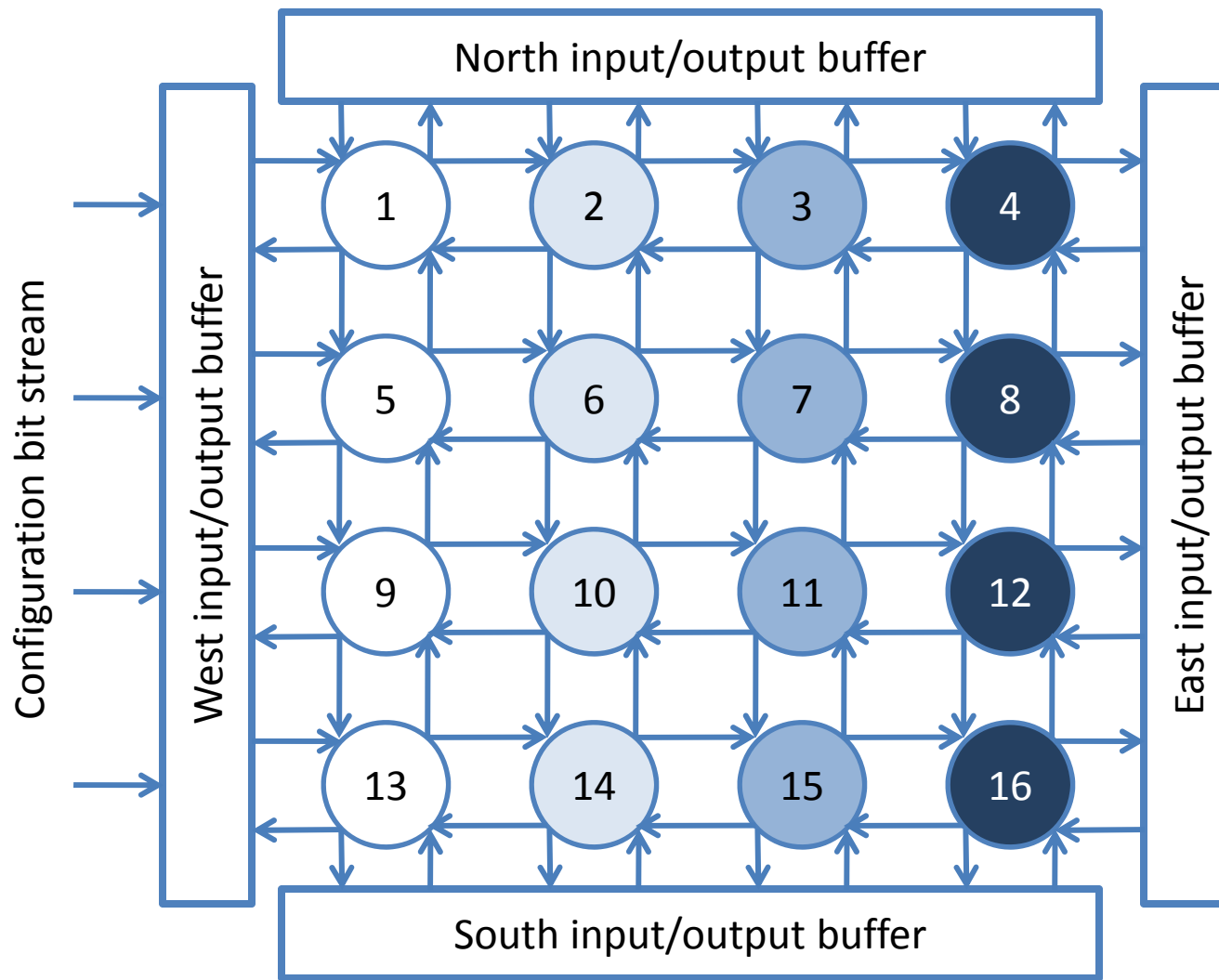
- Design of the identification and configuration algorithms for RBNs using a packet based mechanism.
- Implementation of the CA and RBN nodes in Verilog HDL
- Implementation of a MATLAB toolbox to generate HDL code for CA, RBNs, and small-world networks.
- Implementation of a framework to evaluate power, area, and information processing capabilities of the networks using Synopsys Design Compiler and MATLAB.
- Part of this work has been published in the IEEE Nano 2011 conference [Amarnath11].

# CA Node Implementation

- The CA node presented in [Dascalu00] is implemented for comparison with RBNs.
- A global clock is assumed for all nodes.
- Global signal 'com' determines if the node is in configuration or normal operation mode.
- Global signal 'num' determines the location in the LUT where the configuration bit from the left neighbor is to be placed.



# CA Configuration

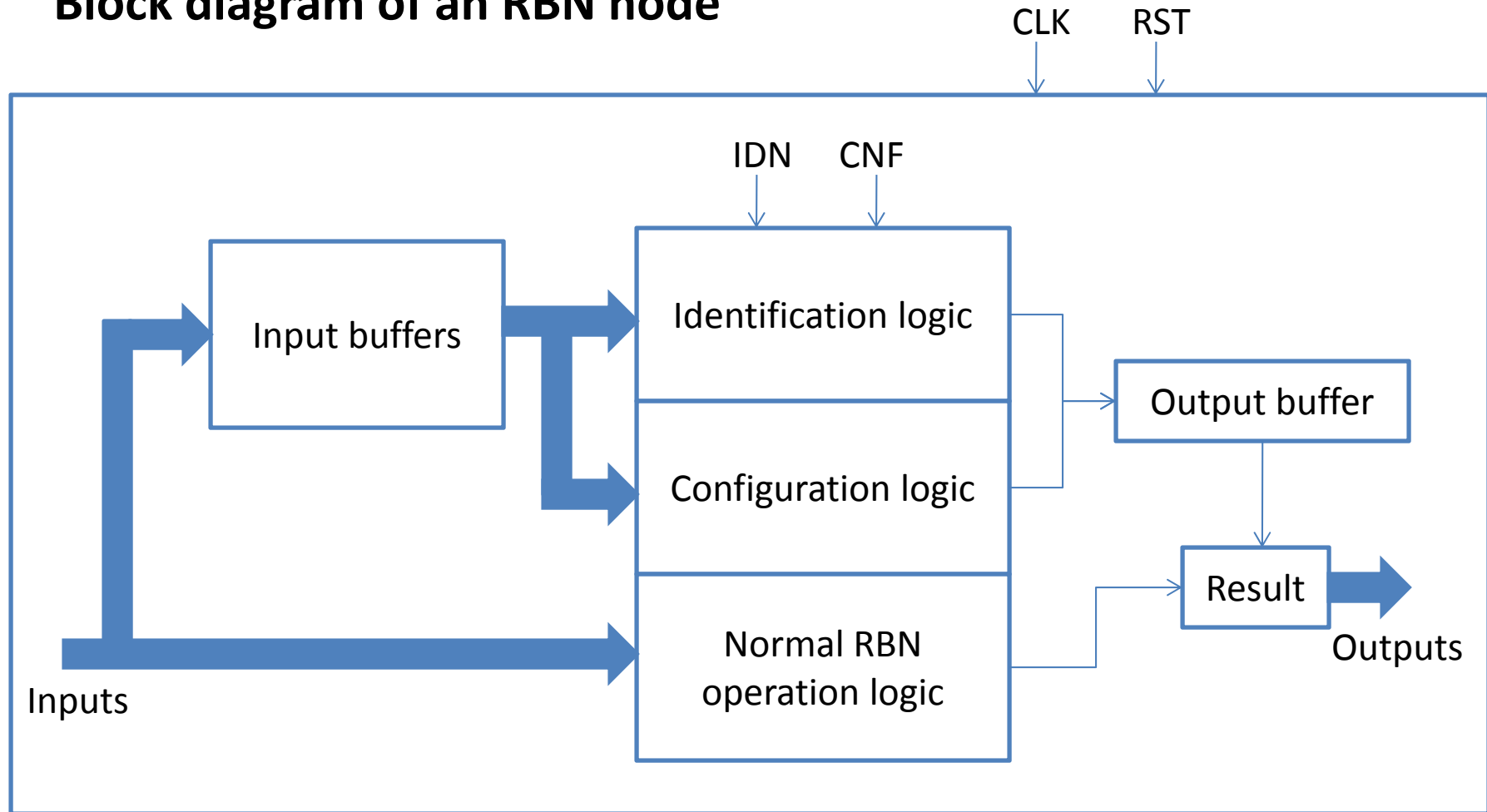


Configuration takes place from west to east. The nodes at the east end are configured first and the nodes at the west end are configured last.

Each node can be configured with a different set of rules.

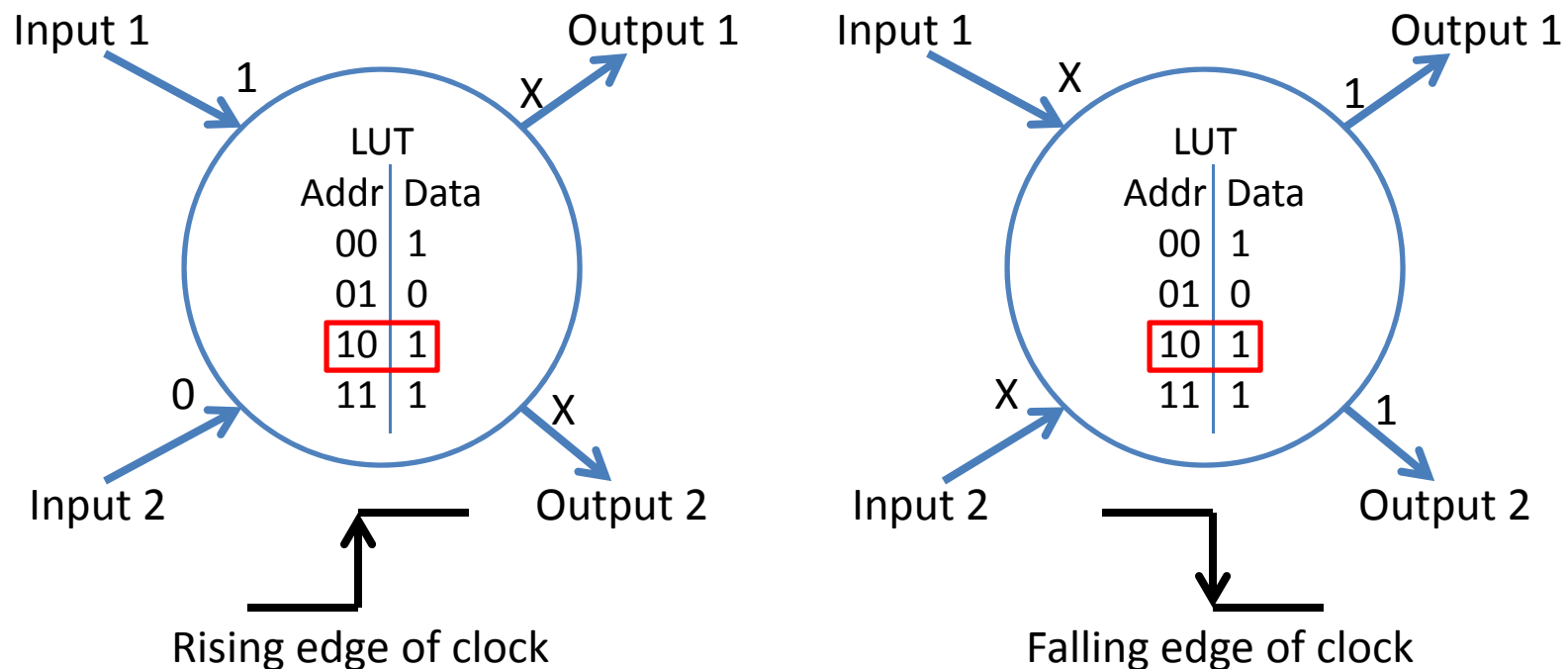
# RBN Node Implementation (1)

## Block diagram of an RBN node



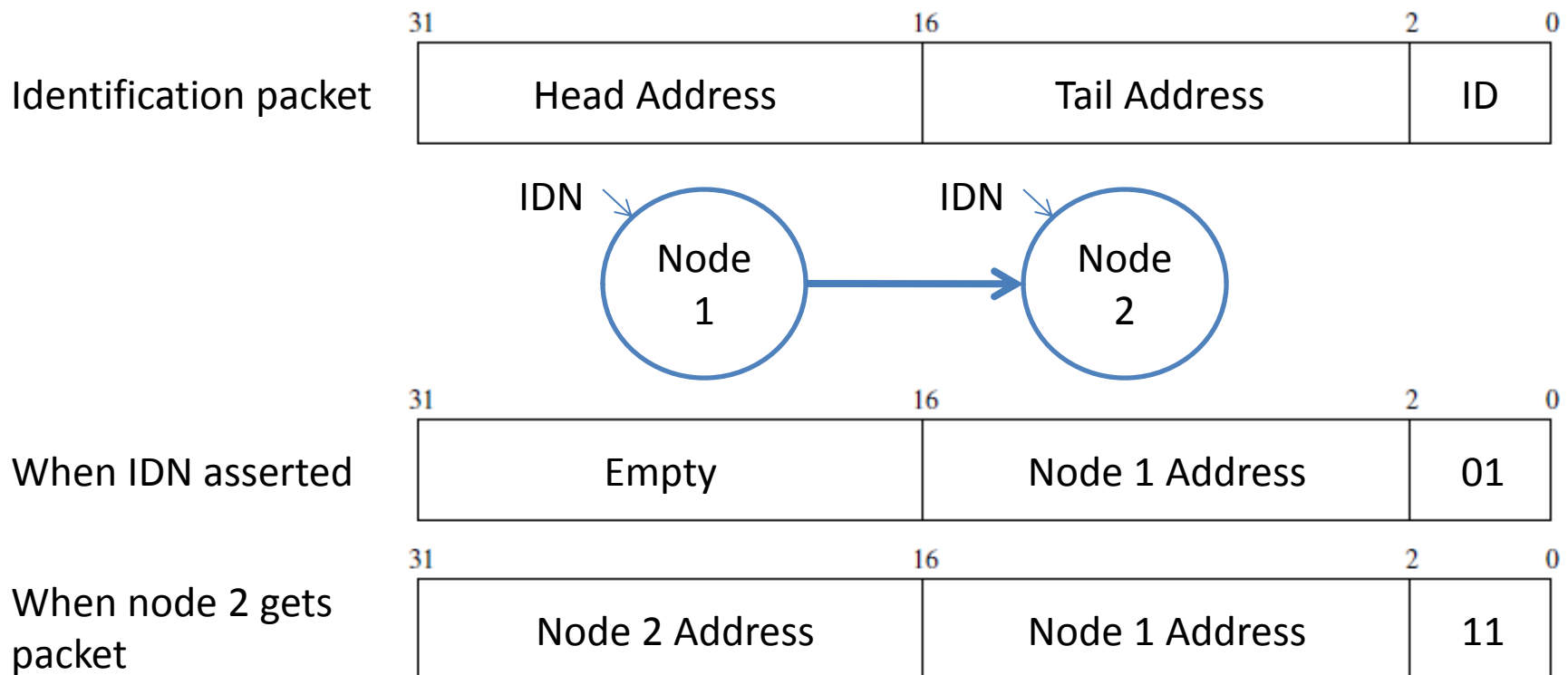
# RBN Node Implementation (2)

- **RBN node operation:**
  - A global clock is assumed for all nodes.
  - Global signals 'IDN' and 'CNF', when asserted, indicate identification and configuration modes respectively.



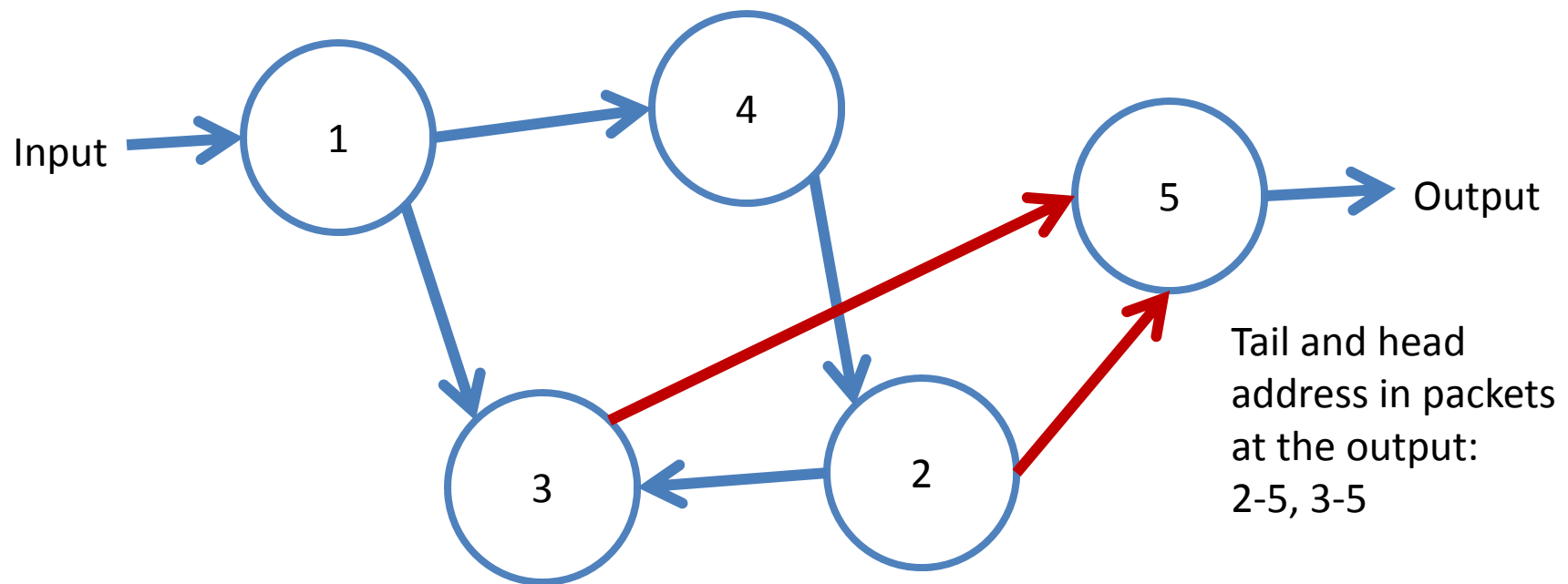
# RBN Identification (1)

- **Identification of the network** topology is required for an external compiler to analyze the network and determine the local rules for each node of the network.
- Each node is assumed to have a unique address.



# RBN Identification (2)

## Identification of an example network

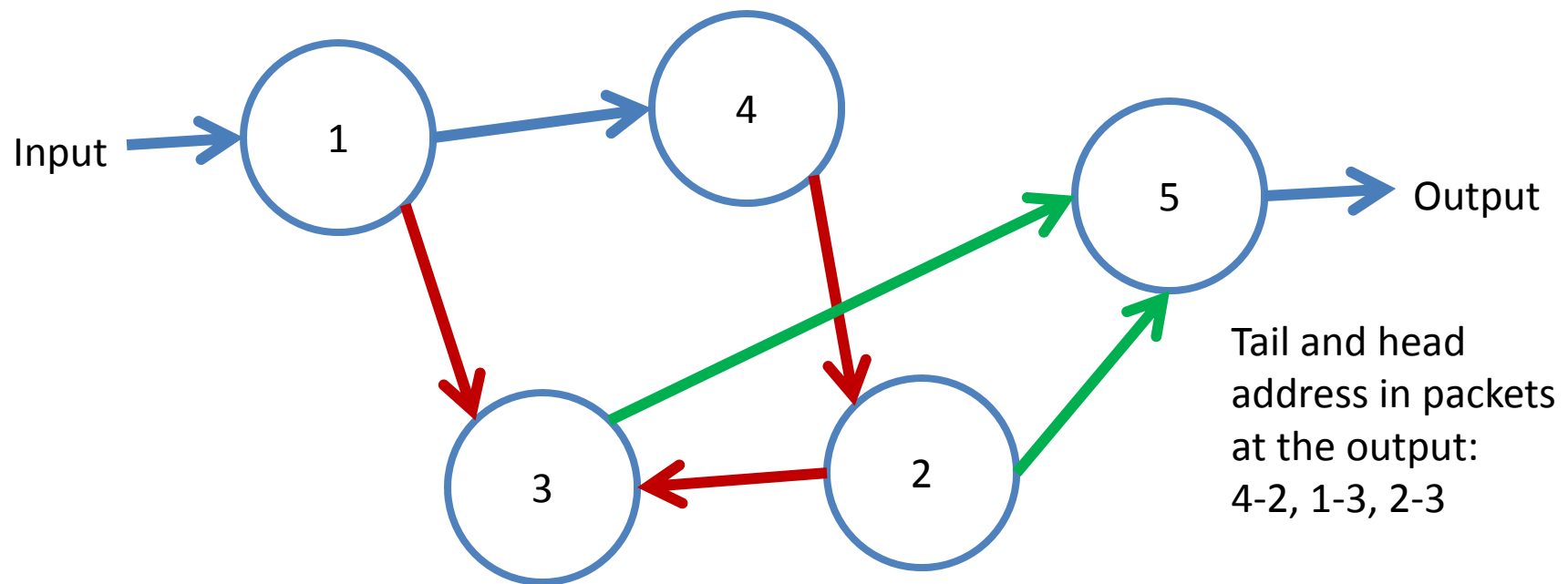


1 – Links that are directly connected to the input of the output node are seen at the output first



# RBN Identification (3)

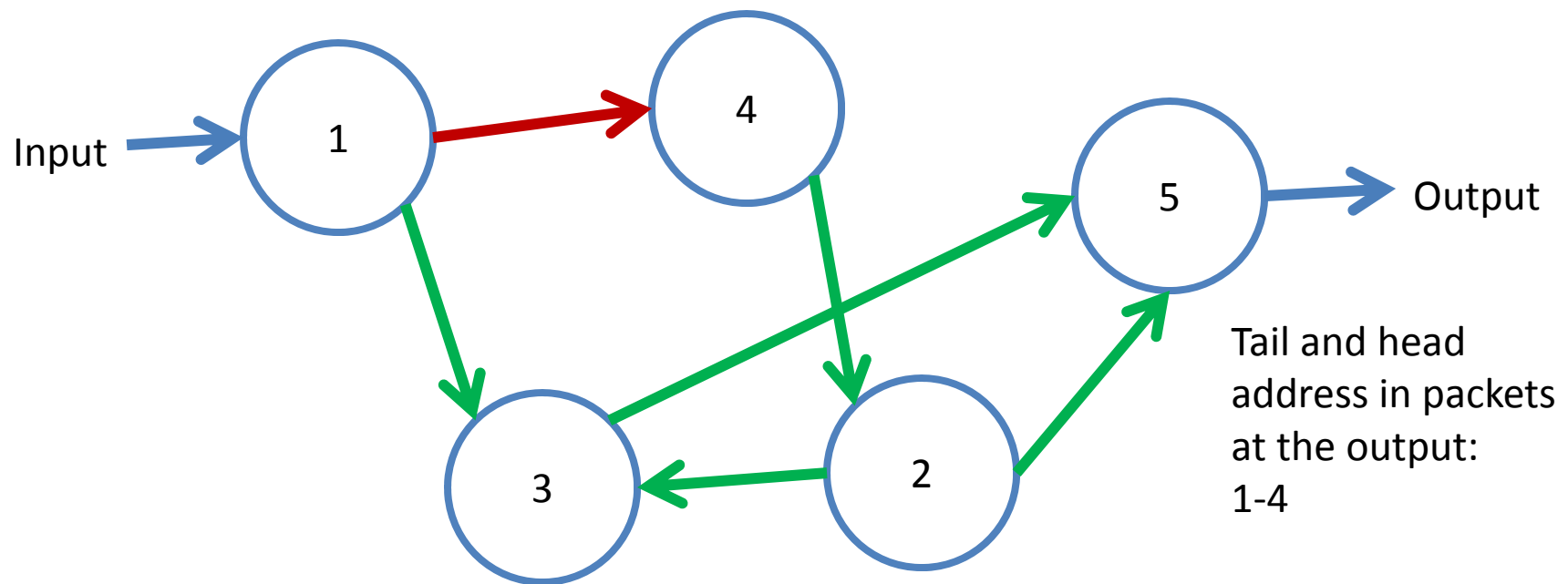
## Identification of an example network



2 – Links that are 1 node hop away from the output node are seen at the output

# RBN Identification (4)

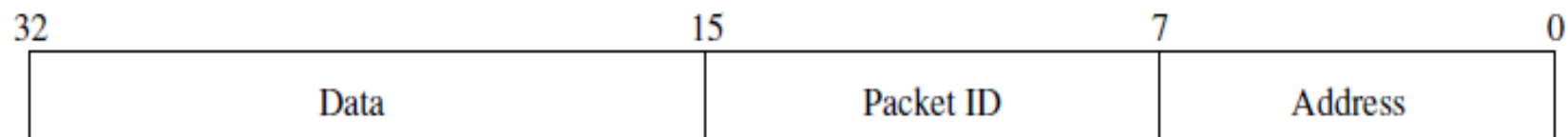
## Identification of an example network



3 – Links that are 2 node hops away from the output node are seen at the output.

# RBN Configuration (1)

- The local rules for the nodes of the network are required to be sent into the appropriate node. This process is called **network configuration**.
- Packet based mechanism is used for configuration.
- Each configuration packet is addressed to a unique node.
- The packet consists of the data that needs to be placed in the look-up-table of the node.
- The packet has information that specifies the starting address of the look-up-table where data is to be placed.

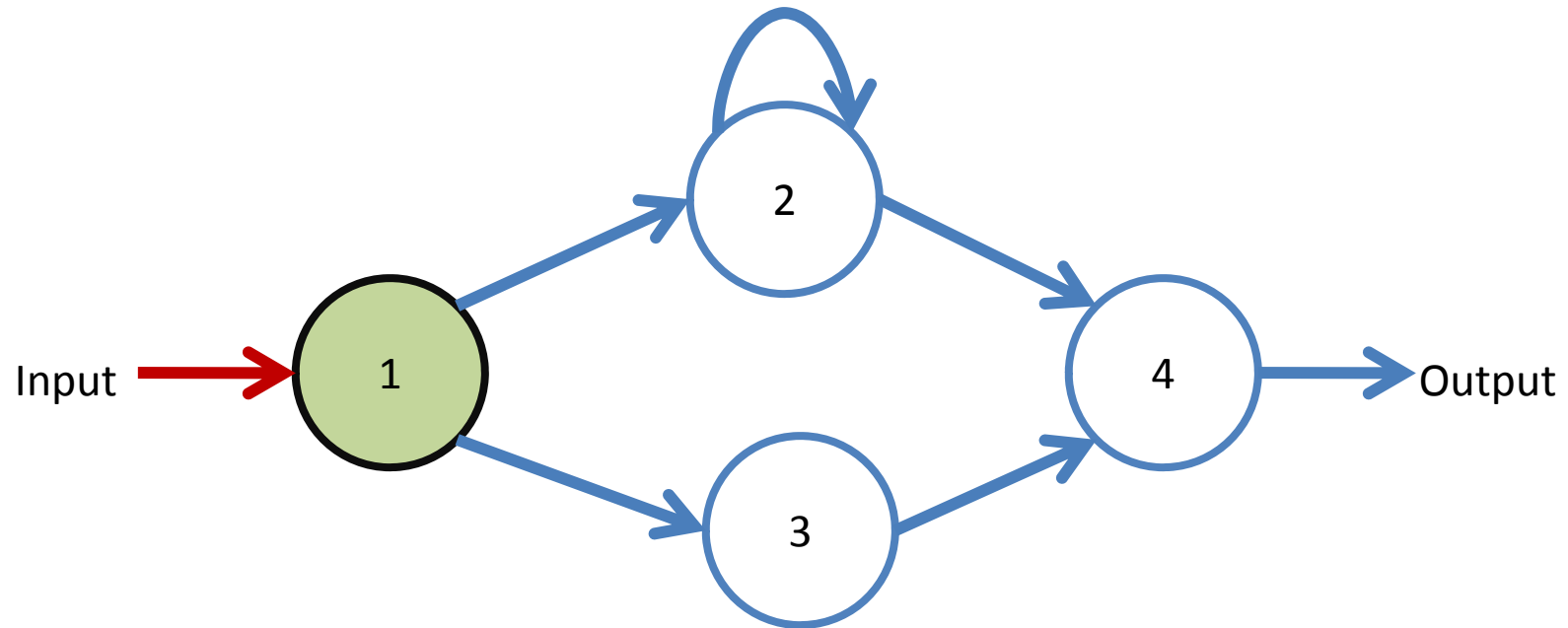


# RBN Configuration (2)

## Configuration of an example network

Configuration packet 1: 

0101010101010101		00000000		00000001
------------------	--	----------	--	----------

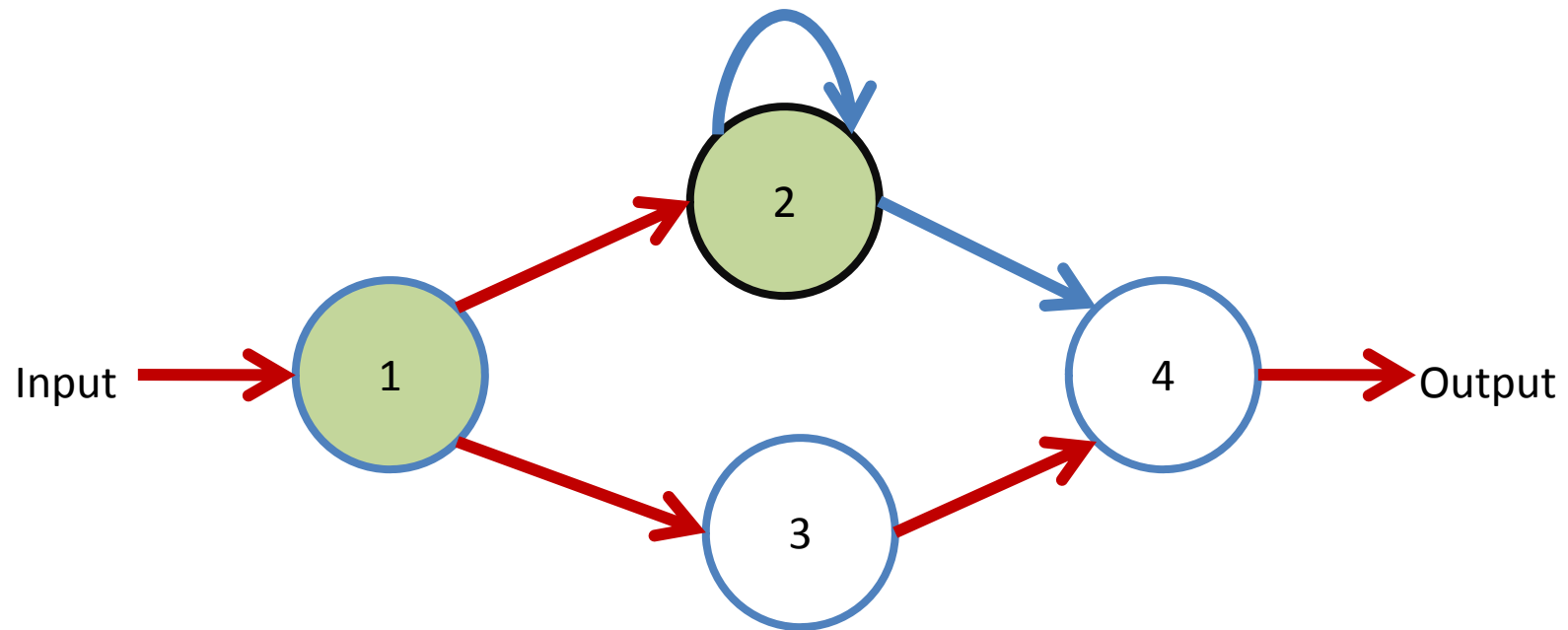


1 - Configuration packet addressed to node 1.

# RBN Configuration (3)

## Configuration of an example network

Configuration packet 2: 1111000011110000 | 00000000 | 00000010



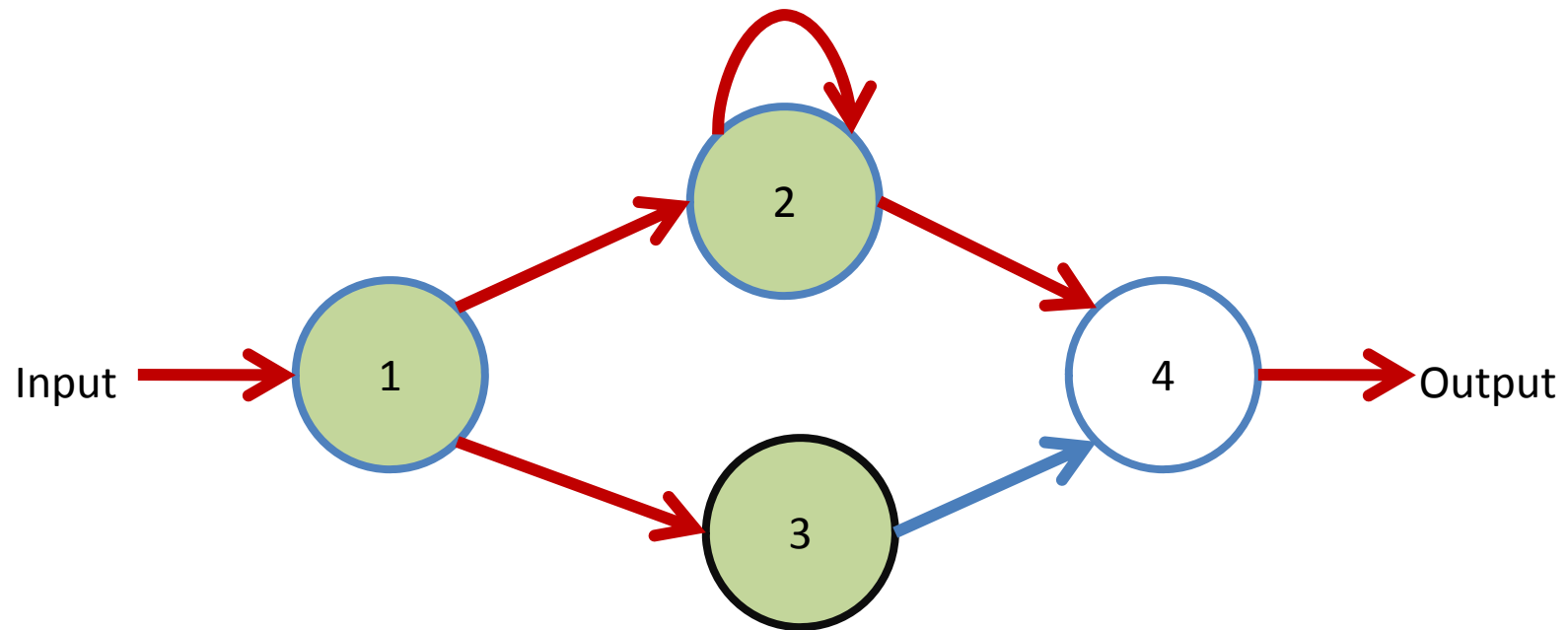
2 - Configuration packet addressed to node 2.

# RBN Configuration (4)

## Configuration of an example network

Configuration packet 3: 

0011001100110011		00000000		00000011
------------------	--	----------	--	----------



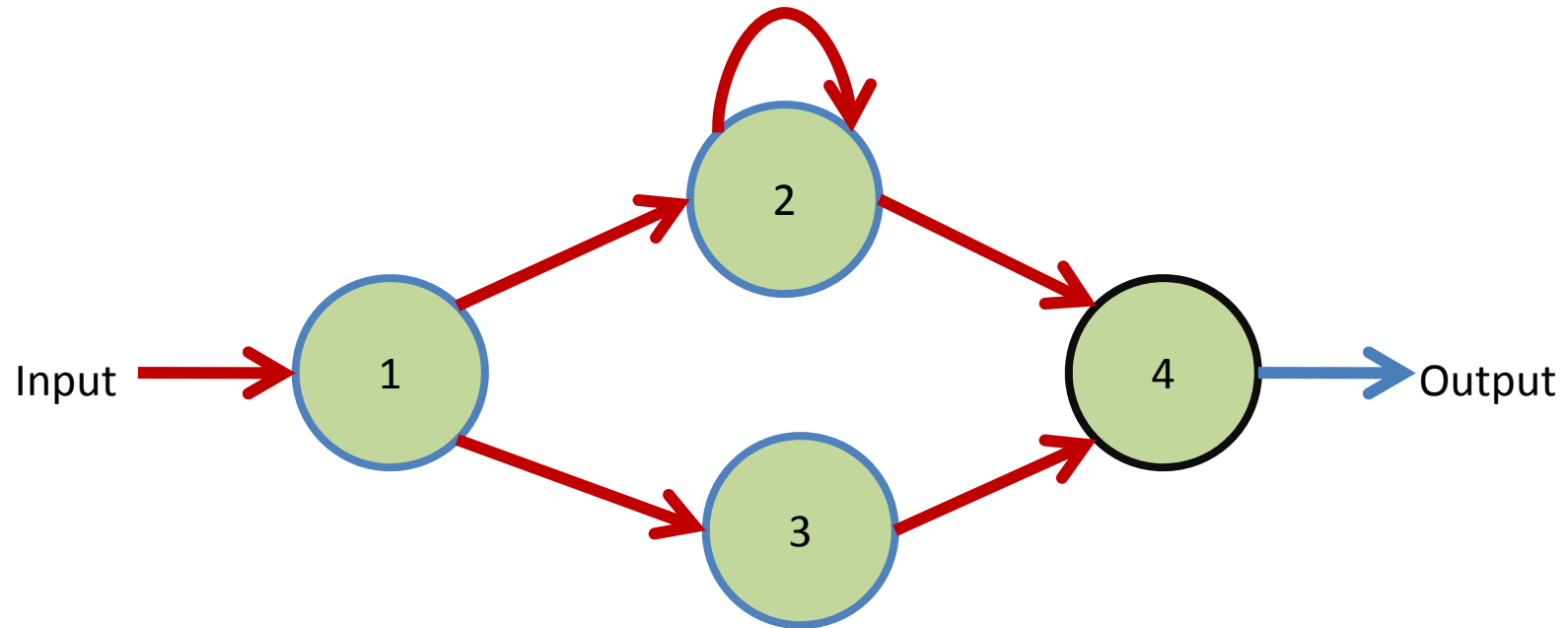
3 - Configuration packet addressed to node 3.

# RBN Configuration (5)

## Configuration of an example network

Configuration packet 4: 

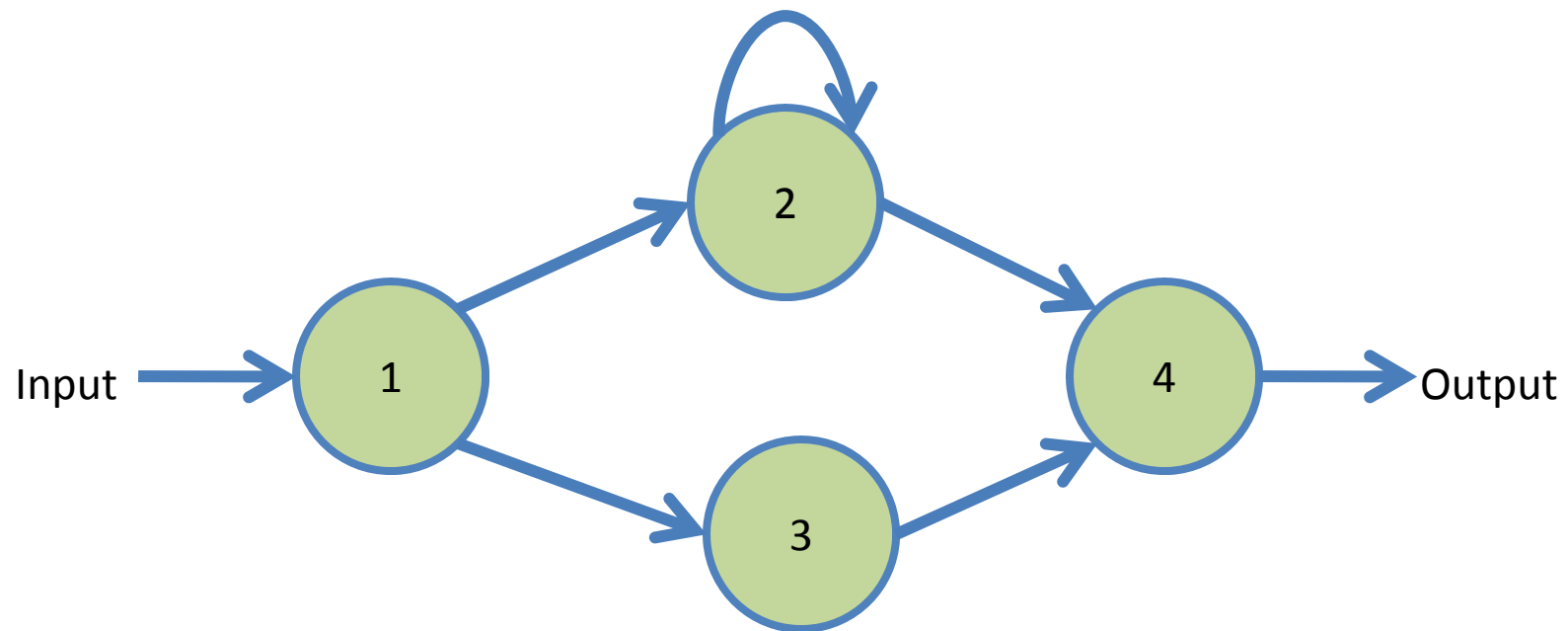
0001110001110011		00000000		00000100
------------------	--	----------	--	----------



4 - Configuration packet addressed to node 4.

# RBN Configuration (6)

## Configuration of an example network

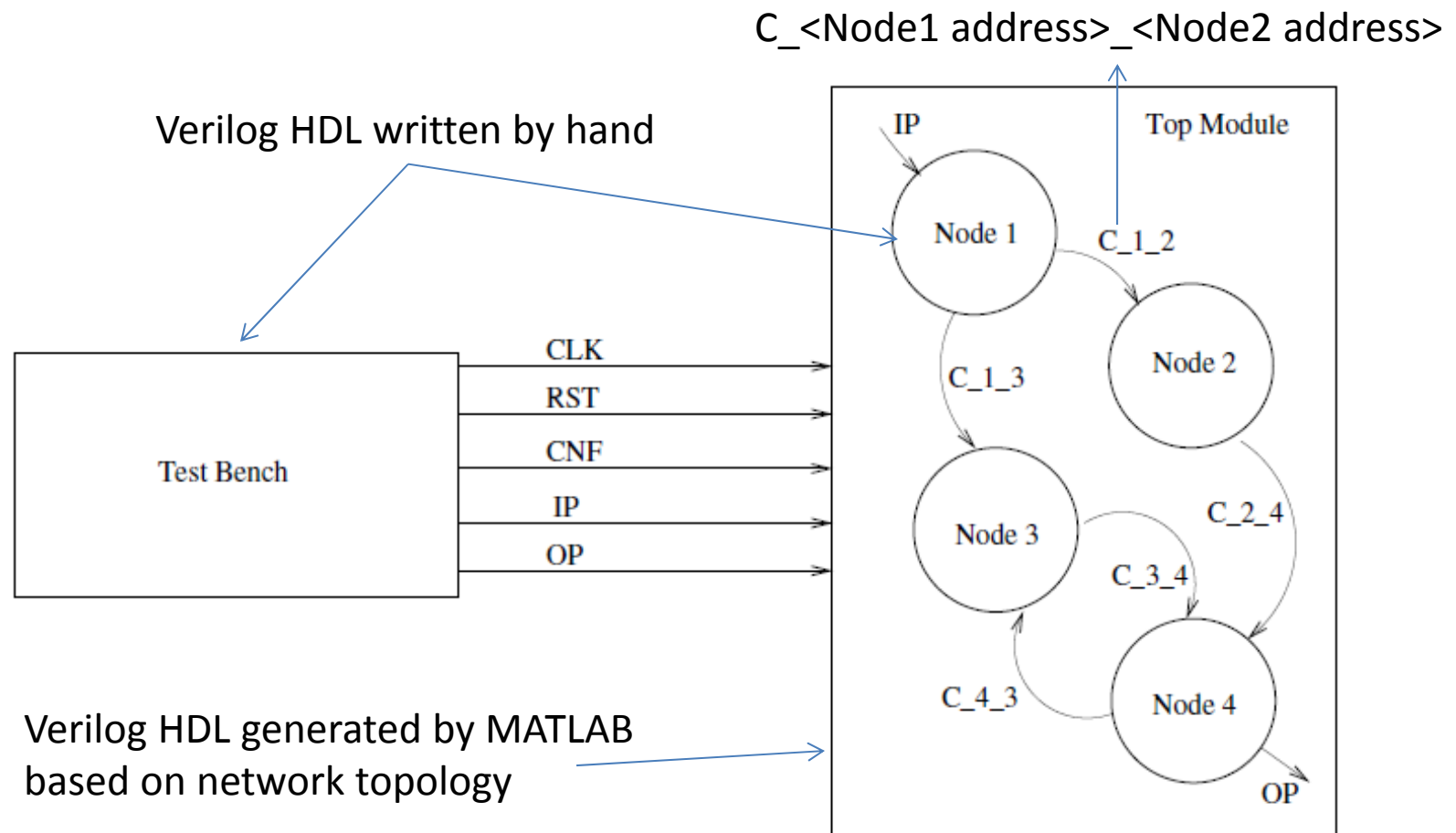


5 – All nodes configured with desired rules.



# HDL Code Generation

## Block diagram of the evaluation framework



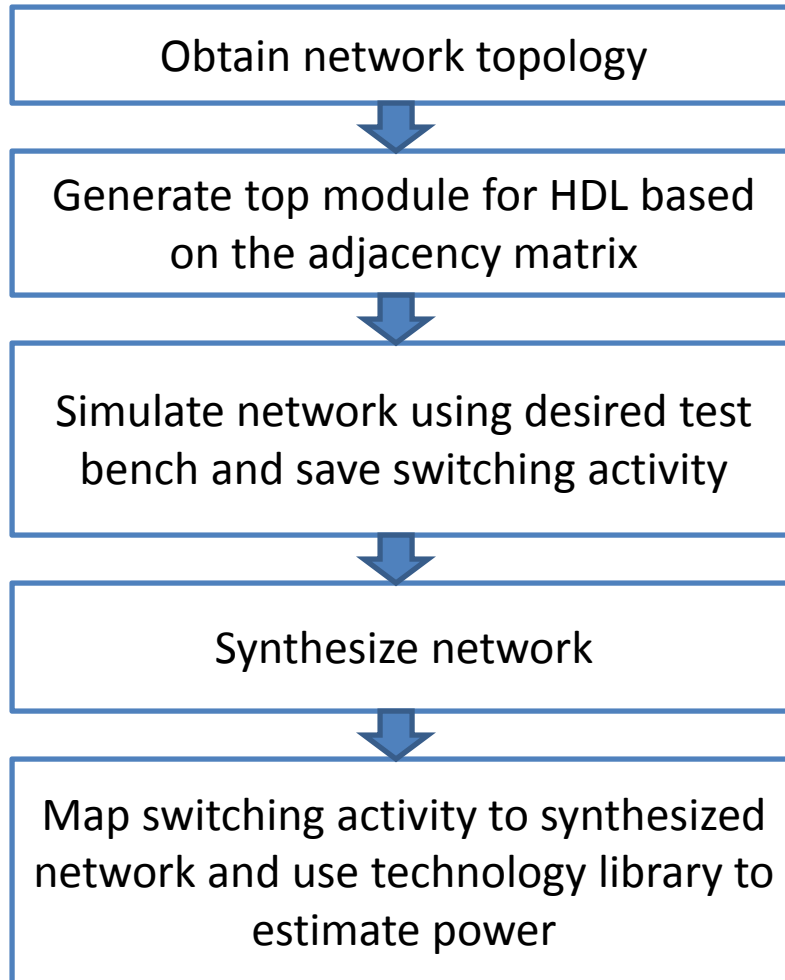
# Power Estimation (1)

- Power due to nodes is estimated by simulation in ModelSim and synthesis and power estimation using Synopsys Design Compiler [Synopsys10].
- Power due to interconnect is measured by measuring the total interconnect length and using the following equation [Snider07]:

$$\text{Dynamic power} = \frac{1}{2} * A * N * C * V_{dd}^2 * f$$

- Power consumption of the network during normal operation is done by synthesizing the entire network.
- Power consumption of the network for configuration and identification logic is approximated by estimating the power due to nodes with different fan-in and substituting the values for the nodes in the network.

# Power Estimation (2)



## Power estimation design flow

180nm technology library was used for this study.

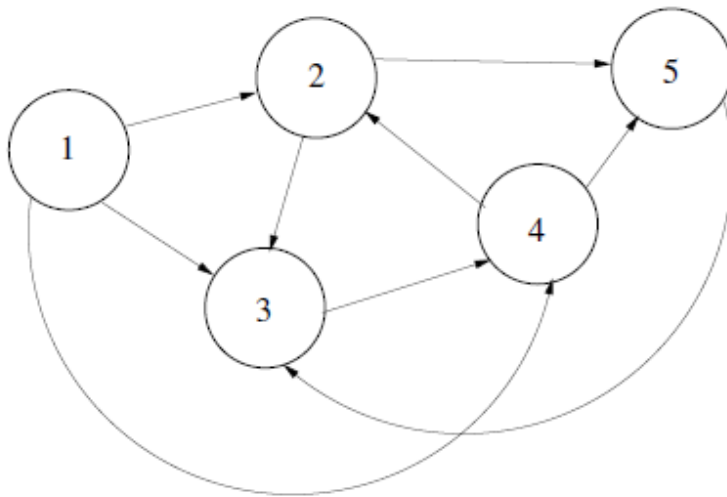
The power for the random networks was averaged over 50 networks.

# Area Estimation

- Area estimation is done by individually evaluating the area consumption of nodes with different fan-in and substituting the values for the nodes in the network.
- The HDL code for the node is synthesized area consumption for the desired CMOS technology library is estimated using design compiler.

# Density Task Performance Evaluation

- Density task is a commonly used **benchmark** problem to evaluate cellular automata [Mesot05].
- The task is successful if the state of each node (output) eventually is the state to which majority of the nodes are initialized. States are randomly assigned initially.
- Each node implements a simple local rule. Its state is the majority of inputs that the nodes received at that time step.

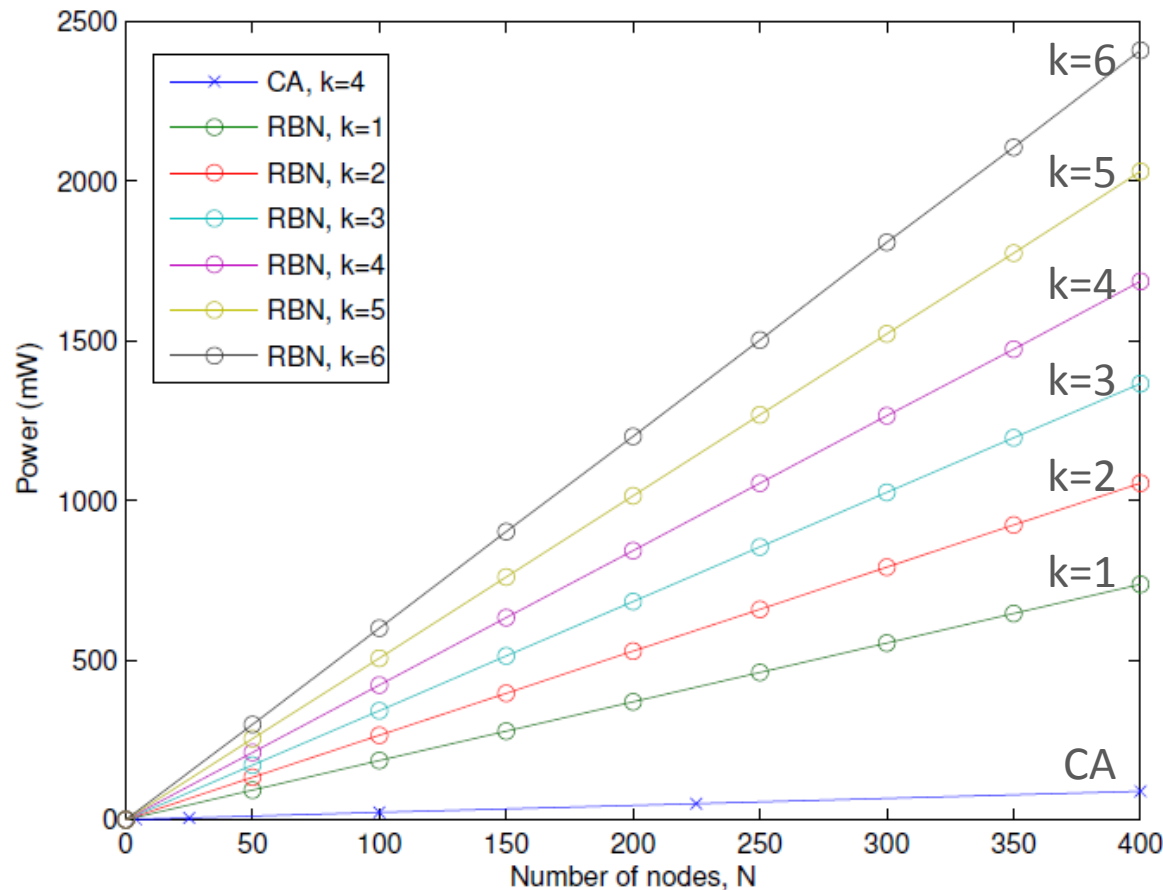


State transition diagram

Initial	0	0	0	1	1
	0	0	0	0	1
Final	0	0	0	0	0

# Power Consumption in CA and RBNs

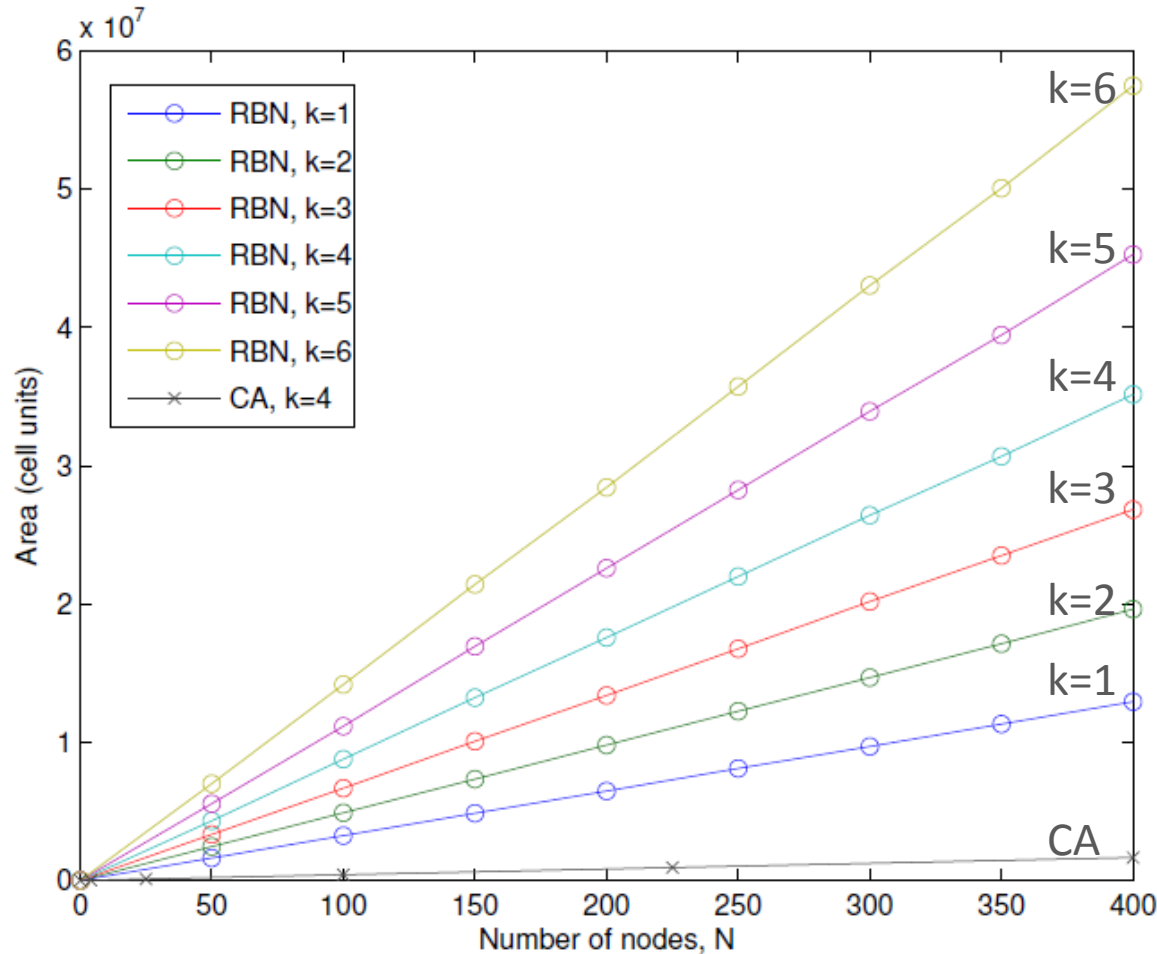
## Variation in power consumption with number of nodes in CA and RBN



- Linear increase in power consumption with increase in number of nodes.
- This can be attributed to the large amount of logic required for identification and configuration.
- With increasing  $k$ , the size of the input buffers and the look-up-tables increases causing the power consumption to increase.

# Area Consumption in CA and RBNs

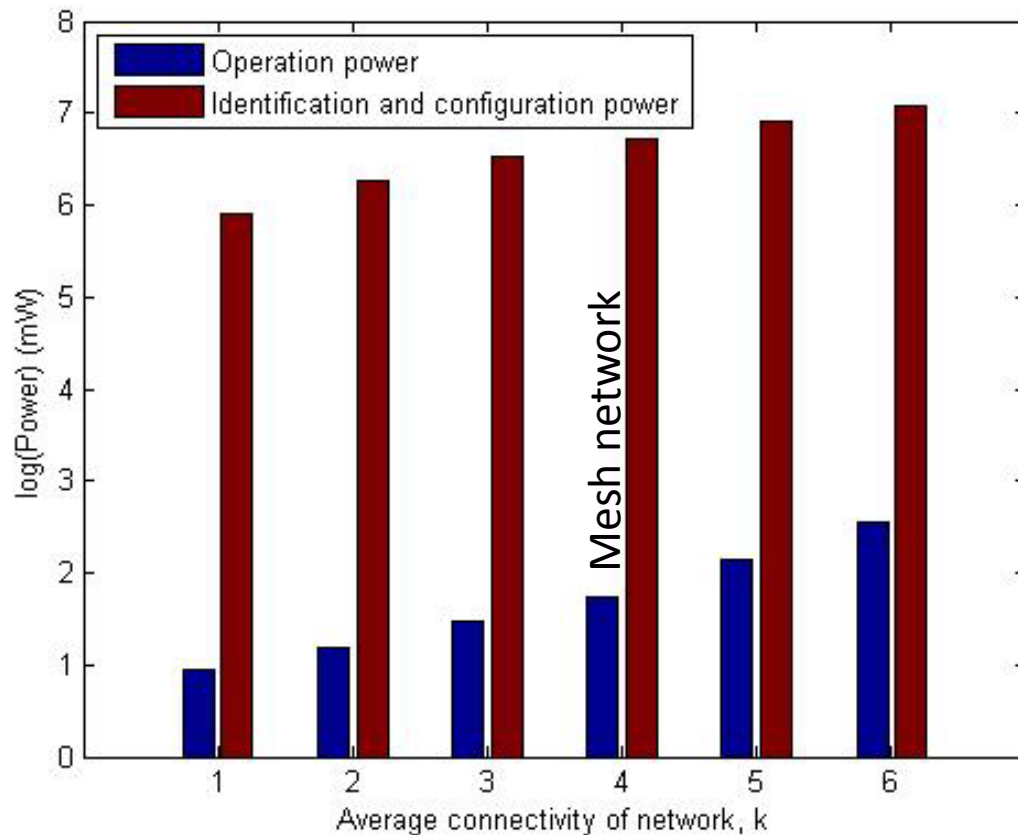
## Variation in area consumption with number of nodes in CA and RBN



- Linear increase in area consumption with increase in number of nodes.
- This is due to the large amount of memory required for the input buffers for RBNs.
- With increase in  $k$ , there is an increase in the number of input buffers and the size of the look-up-tables. Hence area consumption is more.

# Power Consumption in RBN Nodes

## Power consumption in RBNs due to identification, configuration, and normal operation

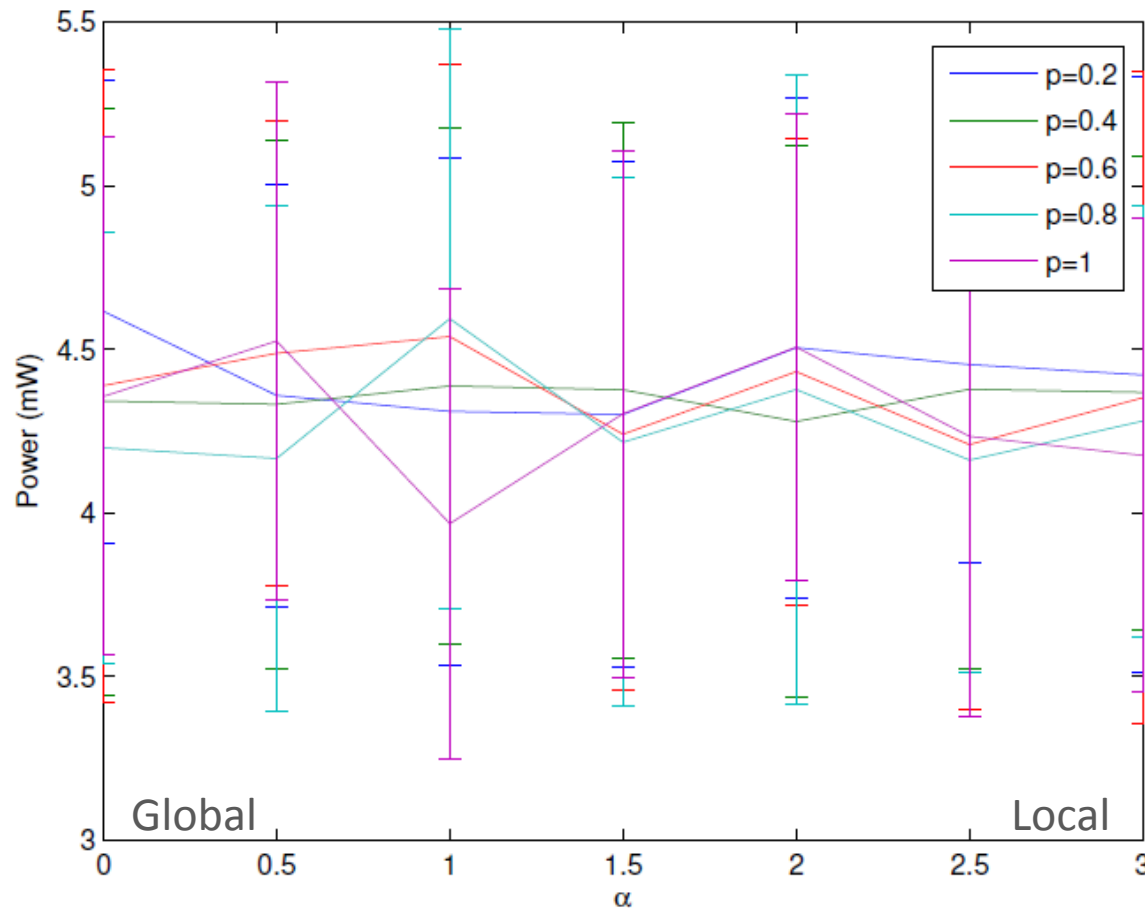


- Power due to configuration and identification is much higher compared to power due to normal operation because of the large amount of logic required to process packets of data.
- Identification and configuration logic does not need to be operational at all times.



# Power Consumption in Small-World Networks

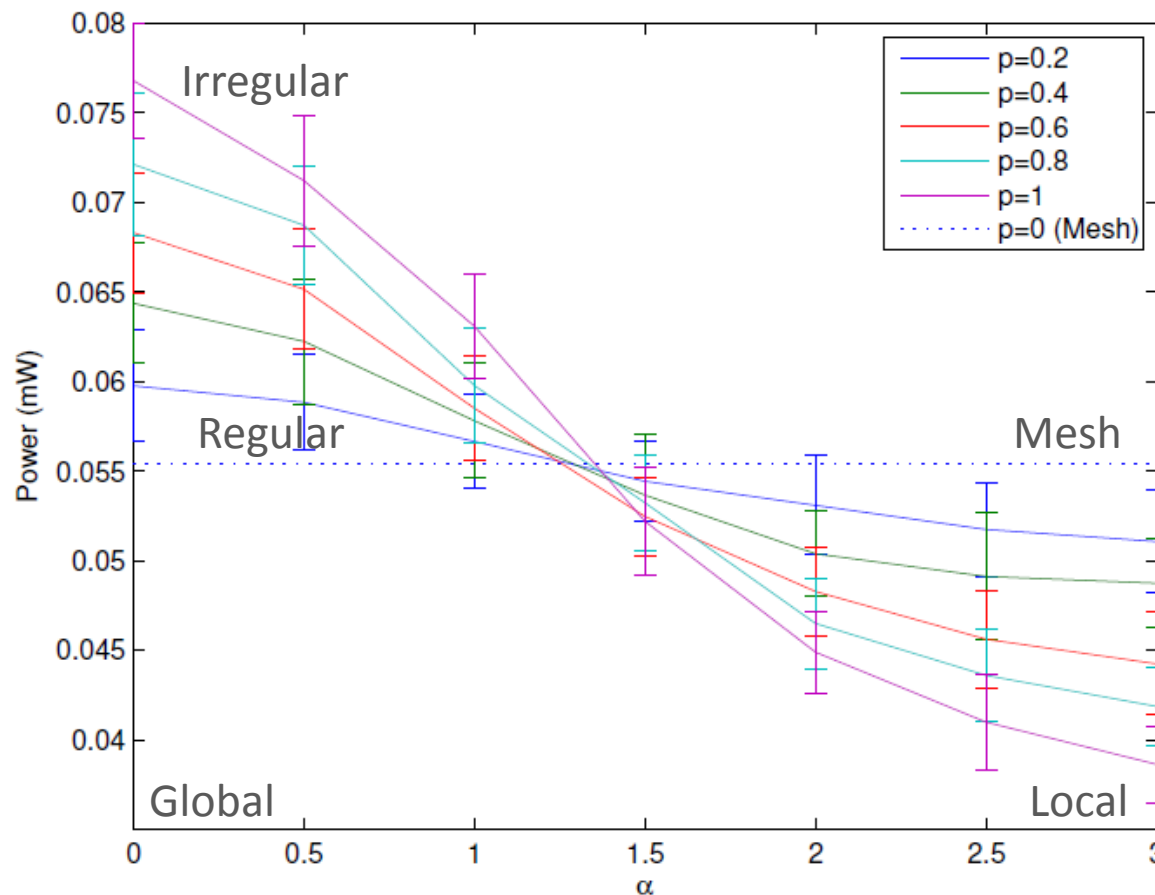
## Total power consumption in small-world networks



- Evaluation of small-world networks allow for the analysis of networks whose randomness is in between mesh and completely random.
- Node power dominates interconnect power in the nano-wire model used.
- Total power remains constant (3.9-4.7 mW) for small world networks since the number of nodes and connectivity remains approximately the same.

# Interconnect Power in Small-World Networks

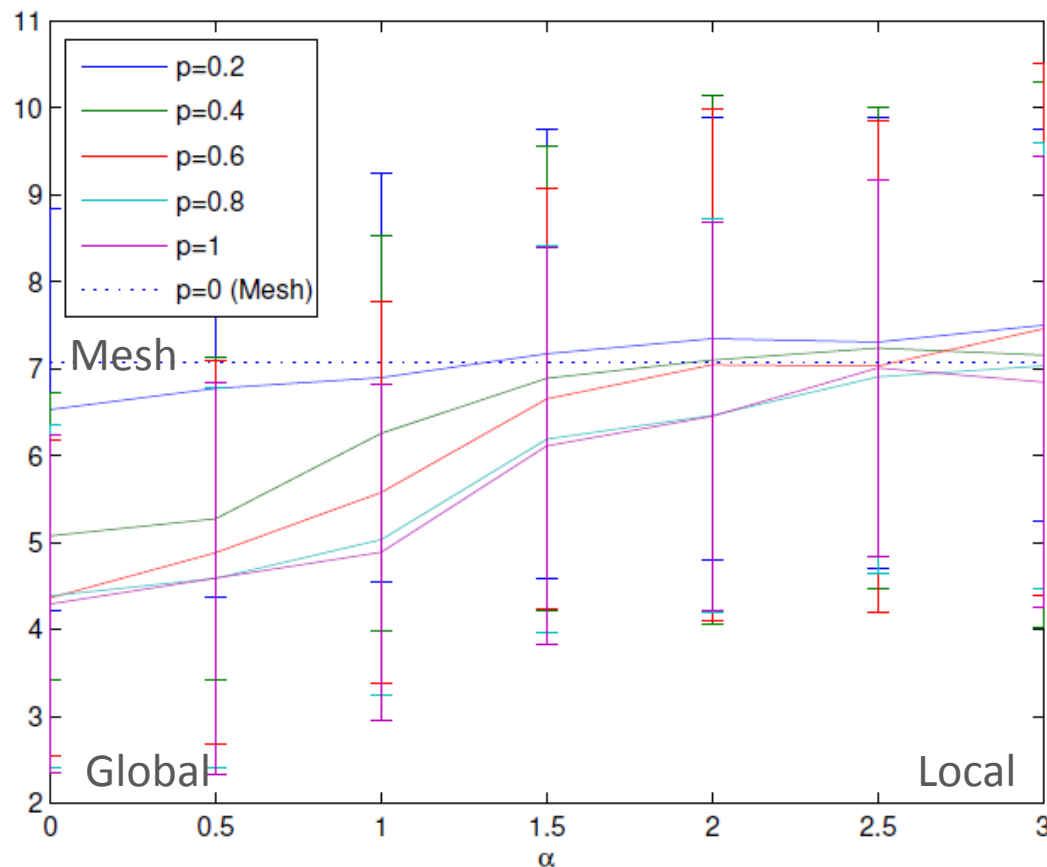
## Interconnect power consumption in small-world networks



- Locally connected networks consume lesser interconnect power compared to networks with longer interconnects.
- Mesh networks are used instead of CAs here for fair comparison.
- For smaller re-wiring probabilities, lesser number of long range links are inserted and hence the smaller variance in interconnect power.

# Density Task Performance

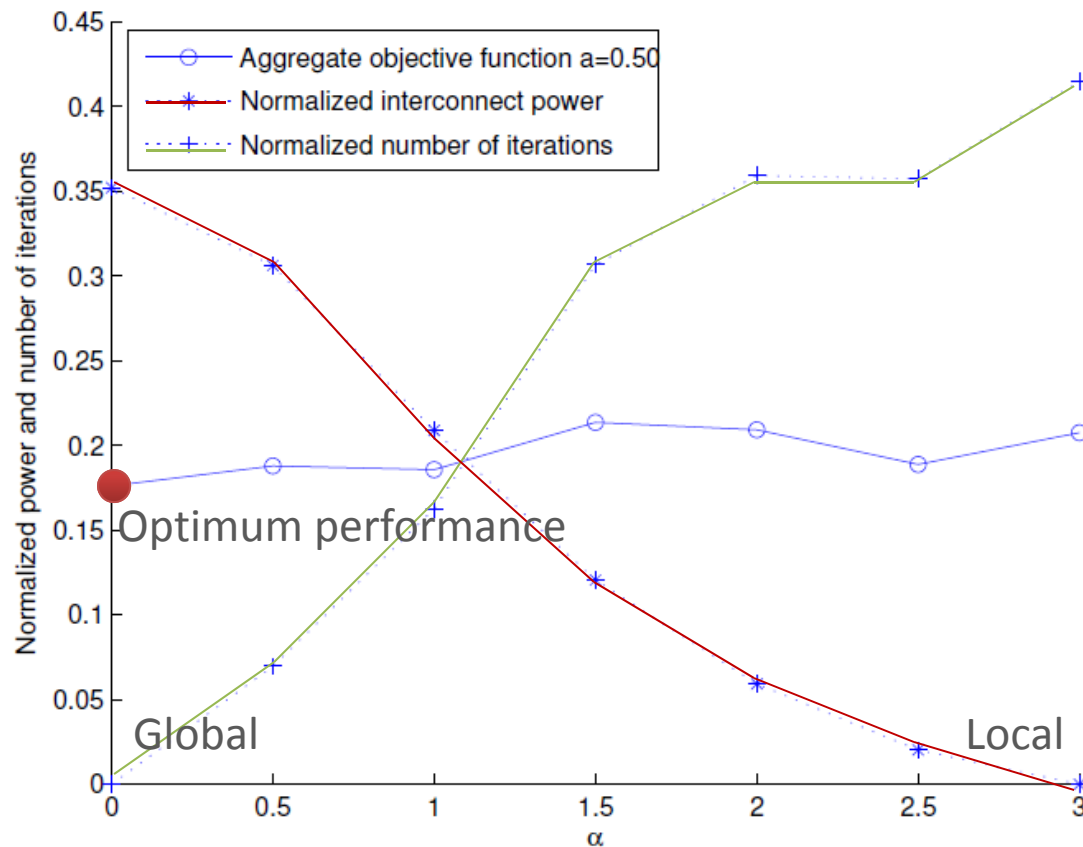
**Average number of iterations for small-world networks to successfully solve the density task**



- Locally connected networks take longer to solve the task while networks with more global links propagate information faster to solve the task quickly.

# Performance Comparison (1)

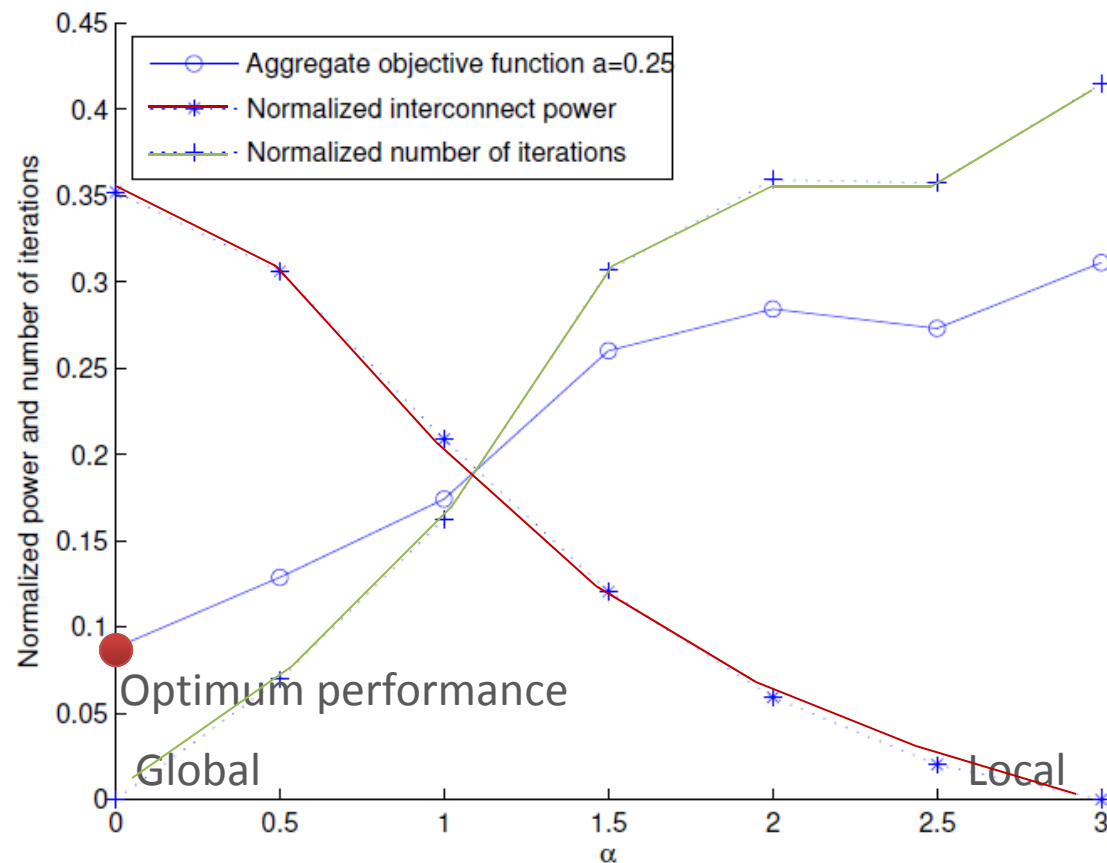
Performance comparison using aggregate objective function vs.  $\alpha$  for  $p=0.6$ ,  $N=64$ , and  $R=100$  with equal weights to interconnect power and density task performance



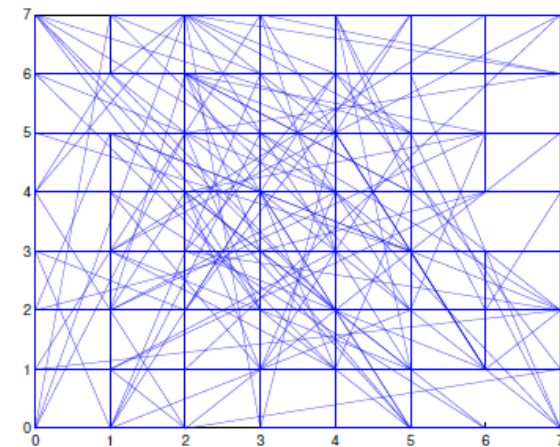
- The aggregate objective function is given as:  
 $(a)(\text{interconnect power}) + (1-a)(\text{number of iterations})$
- The interconnect power and density task performance in this case compensate each other.

# Performance Comparison (2)

**Performance comparison using aggregate objective function vs.  $\alpha$  for  $p=0.6$ ,  $N=64$ , and  $R=100$  with 75% weight towards density task performance**

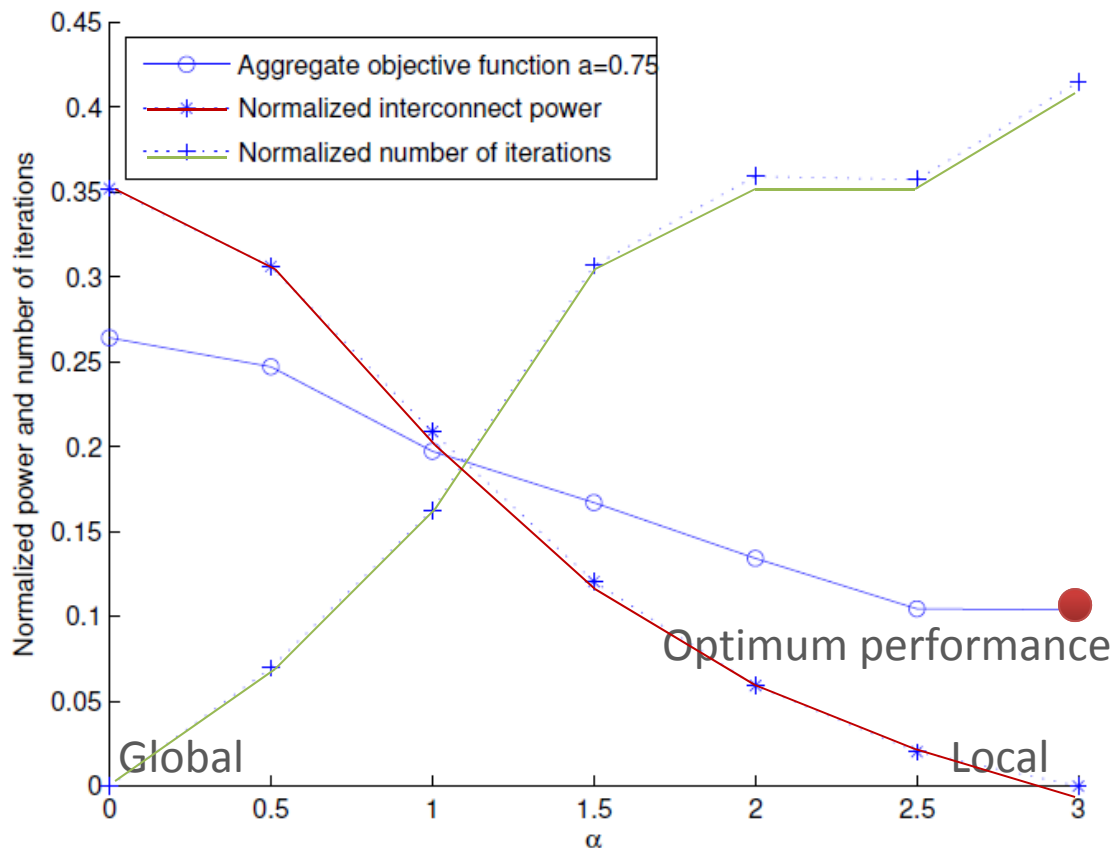


- The function is minimum at lower values of  $\alpha$  indicating networks with longer links are better at information propagation.

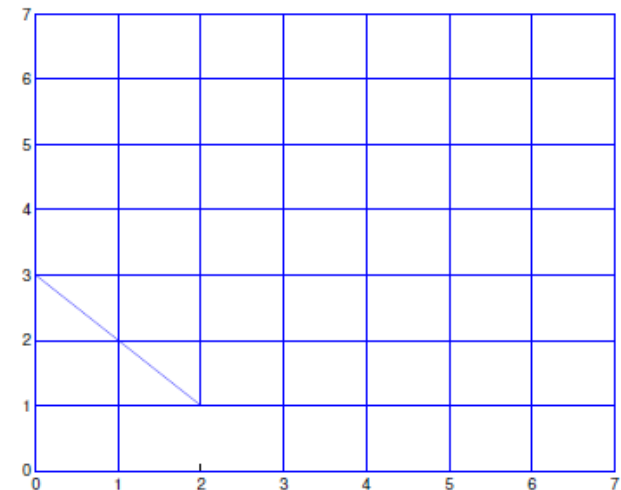


# Performance Comparison (3)

Performance comparison using aggregate objective function vs.  $\alpha$  for  $p=0.6$ ,  $N=64$ , and  $R=100$  with 75% weight towards better interconnect power

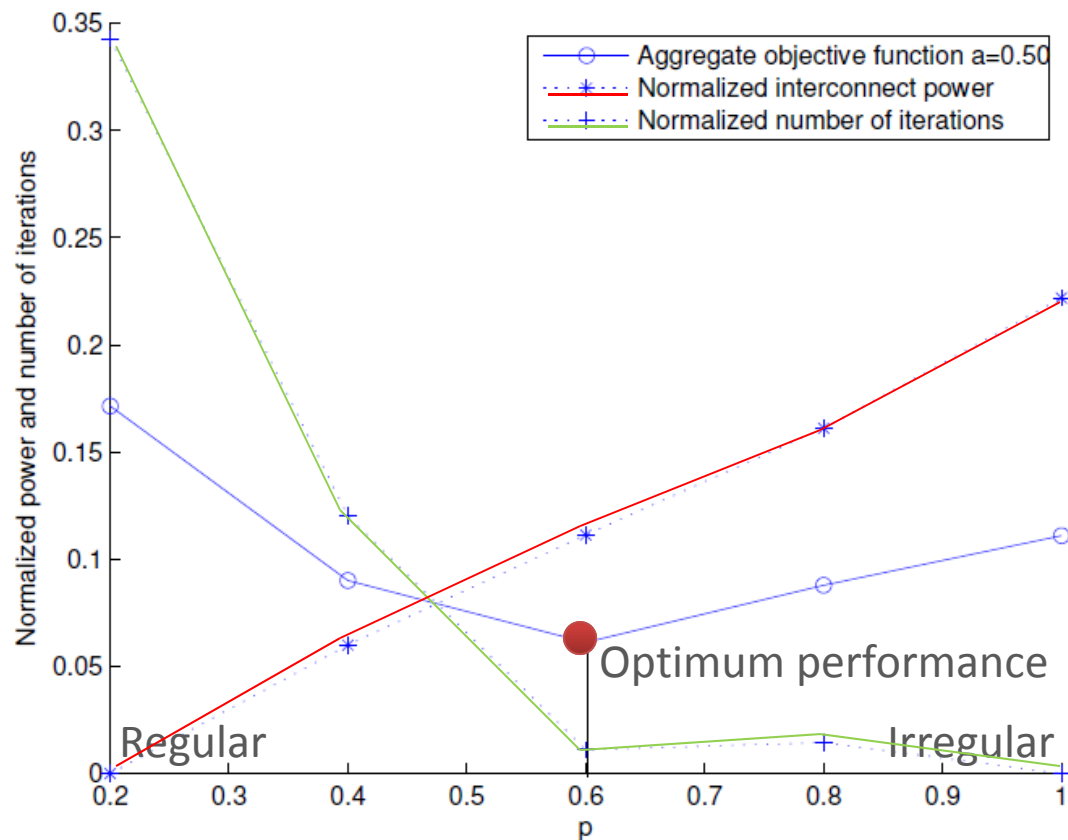


- The function is minimum at higher values of  $\alpha$  indicating networks with shorter links lead to better interconnect power.



# Performance Comparison (4)

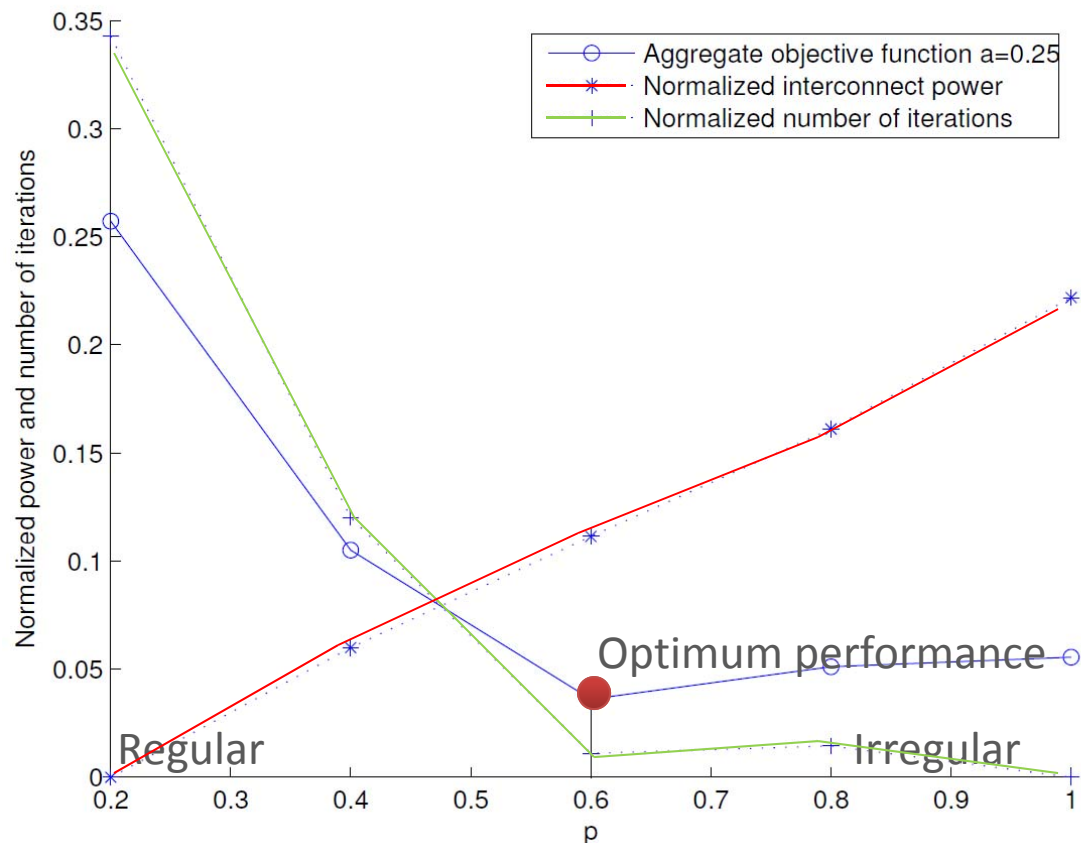
Performance comparison using aggregate objective function vs.  $p$  for  $\alpha=1$ ,  $N=64$ , and  $R=100$  with equal weights to interconnect power and density task performance.



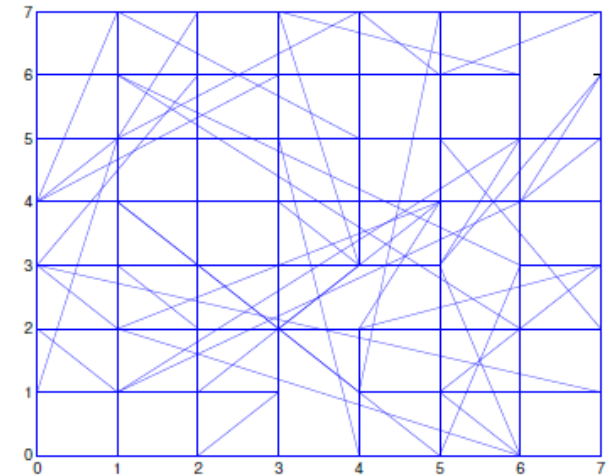
- The function is minimum at around  $p = 0.6$  indicating small world networks with more than half of its links connecting nodes that are farther apart provide best performance in terms of interconnect performance and information propagation.

# Performance Comparison (5)

Performance comparison using aggregate objective function vs.  $p$  for  $\alpha=1$ ,  $N=64$ , and  $R=100$  with 75% weight towards density task performance.



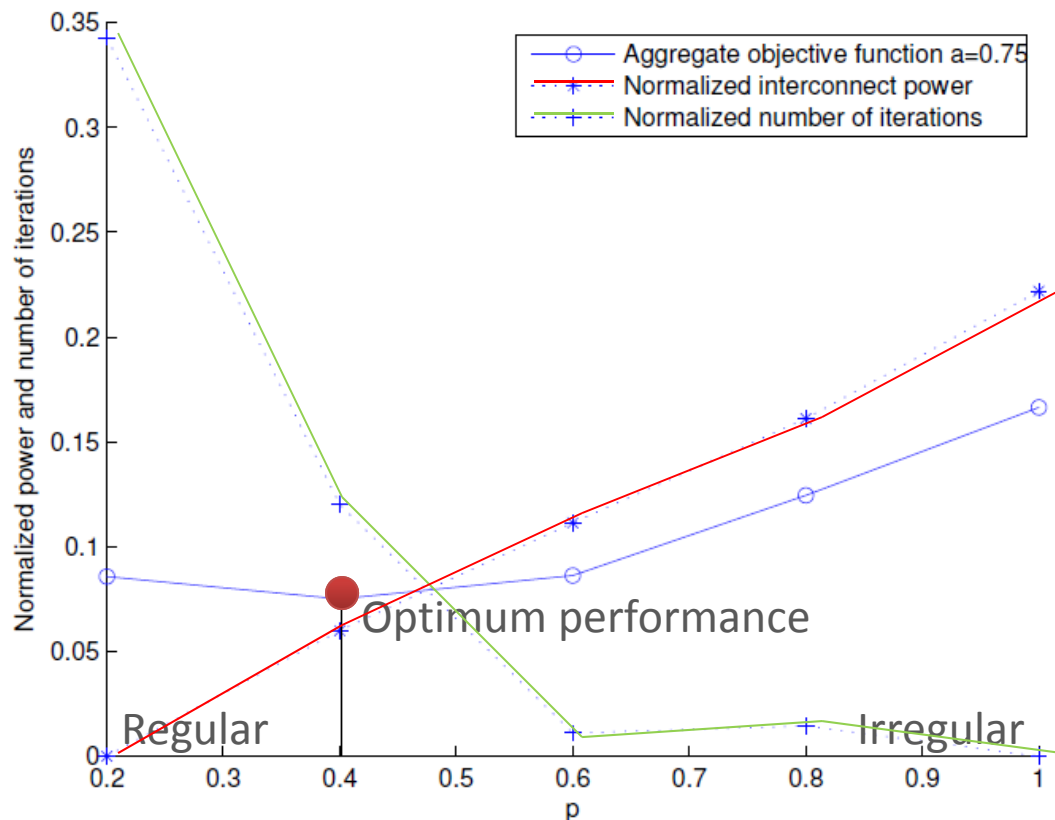
- The function is minimum at higher values of  $p$  indicating networks with higher number of re-wired links are better at information propagation.



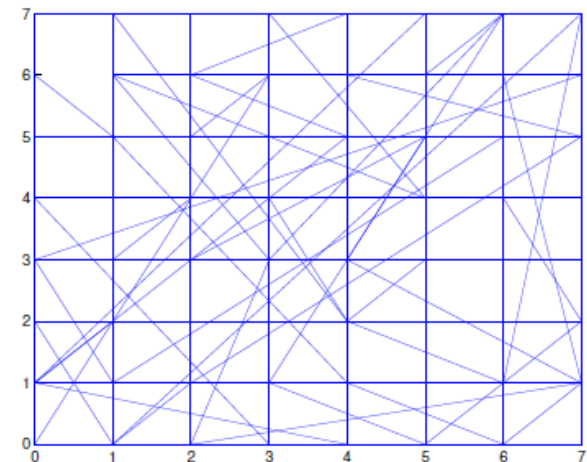


# Performance Comparison (6)

Performance comparison using aggregate objective function vs.  $p$  for  $\alpha=1$ ,  $N=64$ , and  $R=100$  with 75% weight towards better interconnect power



- The function is minimum at lower values of  $p$  indicating that networks with less number of re-wired links have lower interconnect power.



# Conclusion

- Frameworks to create and evaluate CA, RBNs, and small-world networks at the hardware-level have been developed.
- Methods to identify the topology of random networks and configure the nodes of the networks with desired set of rules have been implemented.
- An analysis of small world networks in terms of interconnect power and density task performance has shown that networks whose randomness lies between that of networks which have only local connections and networks that are completely random are suitable for implementation.
- A more complete analysis of the performance of the networks contradicts the study presented in [Zhirnov08] showing that regular structures do not provide optimum performance in terms of information propagation. Faster information propagation requires more long distance connections.

# Future Work

- The implementation of the identification and configuration mechanisms are not optimal and simplified for low power consumption. A better mechanism may result in better power performance.
- Compiler design that is capable of analyzing the network identification packets and generate configuration packets will allow for the evaluation of the networks for more complex applications.
- A comparison with current architectures can be achieved by running the same applications on both architectures.
- Asynchronous communication techniques are better suited for these architectures in order to avoid large clock distribution networks.

# References

- [ITRS09] International technology roadmap for semiconductors (ITRS). Technical report, Semiconductor Industry Association, 2011 update.
- [Amarnath11] A. Amarnath, P. Damera, A. Goudarzi, and C. Teuscher. Latency and power consumption in unstructured nanoscale Boolean networks. In *Proceedings of the 11<sup>th</sup> International Conference on Nanotechnology (IEEE Nano 2011)*, Cincinnati, OH, USA, 2011. IEEE. In press.
- [Dascalu00] M. Dascalu and E. Franti. Implementation of totalistic cellular automata. 1:273-276 vol.1, 2000.
- [Kauffman93] S. A. Kauffman. *The origins of order: self organization and selection in evolution*. Oxford University Press US, 1993.
- [Kleinberg00] J. M. Kleinberg. Navigation in a small world. *Nature*, 406(6798):845, 2000.
- [Mesot05] B. Mesot and C. Teuscher. Deducing local rules for solving global tasks with random Boolean networks. *Physica D: Nonlinear Phenomena*, 211(1-2):88-106, November 2005.
- [Petermann06] T. Petermann and P. De Los Rios. Physical realizability of small-world networks. *Physical Review E*, 73(2):026114, February 2006.
- [Snider07] G. S. Snider and R. S. Williams. Nano/CMOS architectures using a field programmable nanowire interconnect. *Nanotechnology*, 18(3):035204, January 2007.
- [Synopsys10] Synopsys. Design compiler user guide, June 2010.
- [Watts98] Collective dynamics of small-world networks. *Nature*, 393(6684):440-442, June 1998.
- [Zhirnov08] An assessment of integrated digital cellular automata architectures. *Computer*, 41(1):38-44, 2008.