Markov Network Based Equivalence Checking in Mixed-Signal Systems

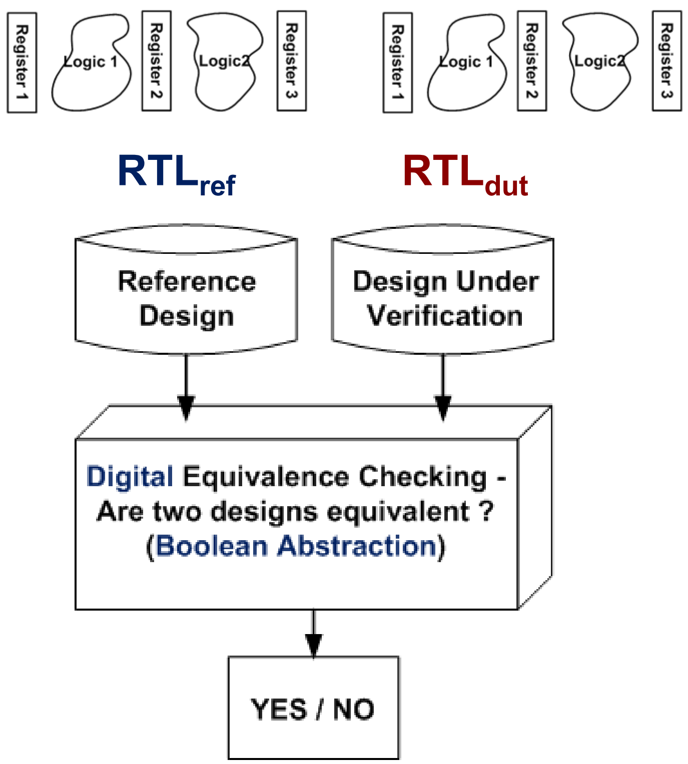
Sangho Youn and Jaeha Kim
Inter-university Semiconductor Research Center
Seoul National University, Seoul, Korea

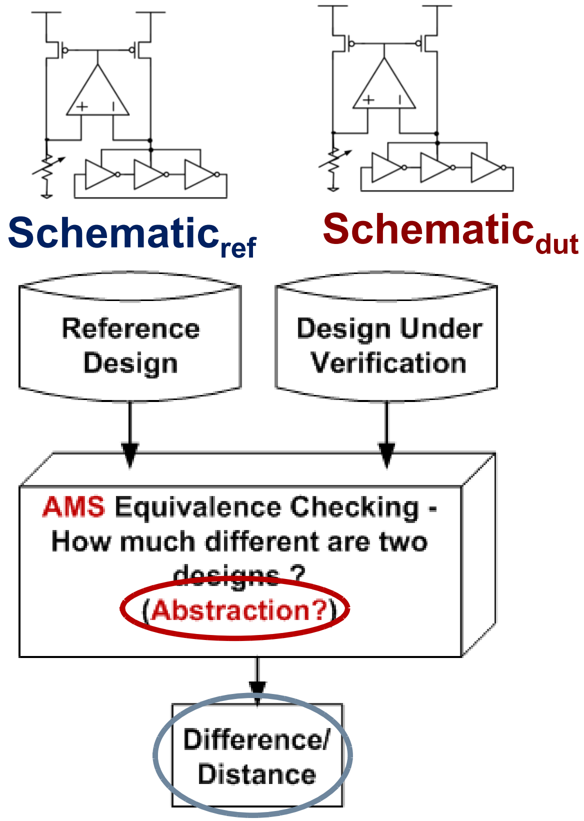
Outline

- Motivation
- Equivalence in AMS System
- Proposed Abstraction – Markov Network
- Proposed Equivalence Checking
- Experimental Results
- Conclusion

Motivation

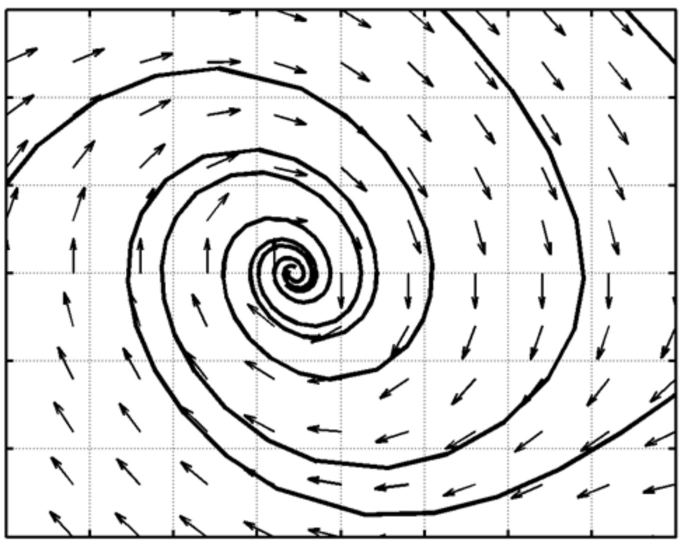
- Check equivalence between two designs described on the same abstraction level and possibly on different levels
- What abstraction/representation can we use for Analog?





State Space Representation

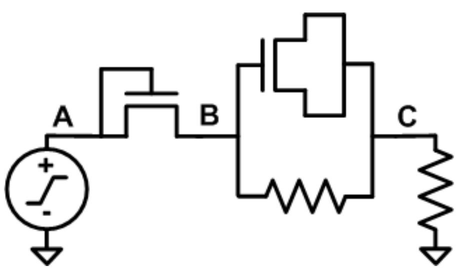
- Dynamic system
  - State space equation
    - Represents a circuit as an ordinary differential equation (ODE) set
    - Describes a circuit's transient behavior
    - Vector fields – *dX/dt = f(X,u,t)*



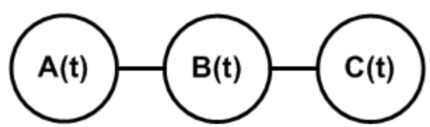
x` = y
*y` = -e*y-sin(x)*

Markov Chain Representation

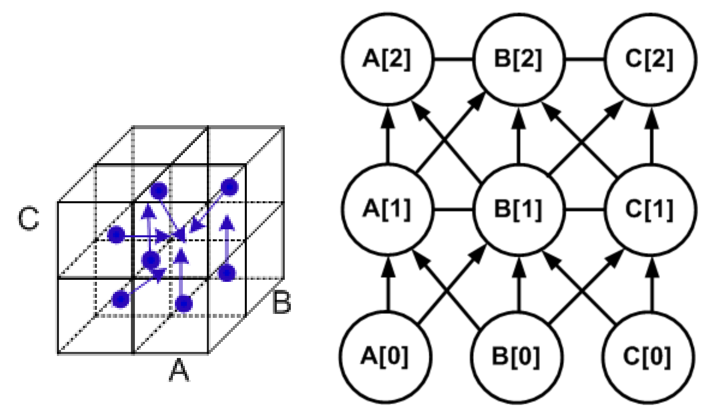
- Dynamic system and Markov Chain
  - With discretization of both time and state value, it becomes Markov Chain
    - Also it could easily reflect PVT variations
  - $P(X_{next}|X_{current})$ instead of $dX/dt = f(X,u,t)$



$g_A(A,B,A`,B`,t)=0$
$g_B(A,B,C,A`,B`,C`,t)=0$
$g_C(B,C,B`,C`,t)=0$



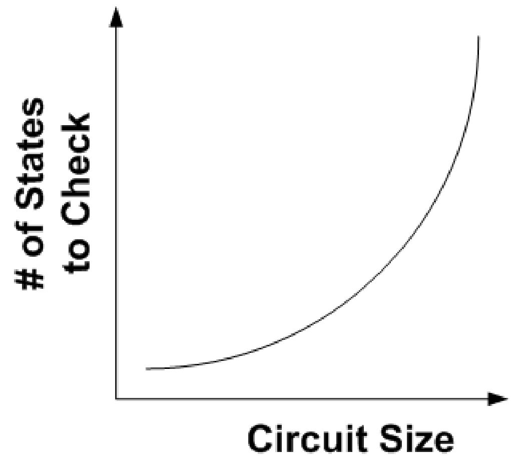
$A` = f_A(A,B,C,t)$
$B` = f_B(A,B,C,t)$
$C` = f_C(A,B,C,t)$



$P(A[n+1], B[n+1], C[n+1] \mid A[n], B[n], C[n])$

Equivalence in Analog System

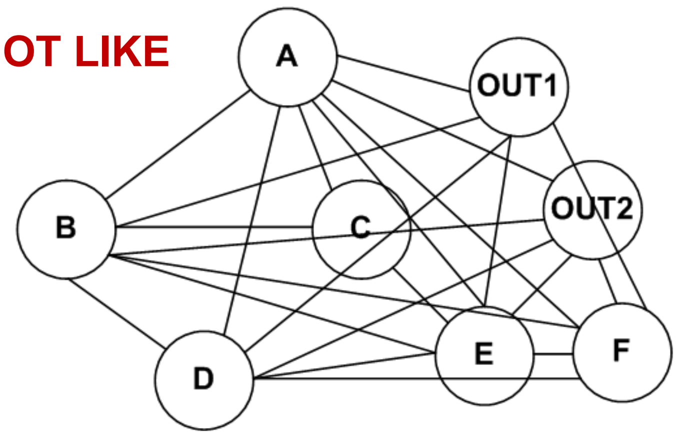
- When two systems' transition probabilities are same, two systems are equivalent
  - Compare $P_{ref}(X_{next}|X_{current})$ and $P_{dut}(X_{next}|X_{current})$
- However, as circuit size increases, the number of states to check grows exponentially !
  - A PLL with 300 nodes requires comparison of 300 dimensional transition probability



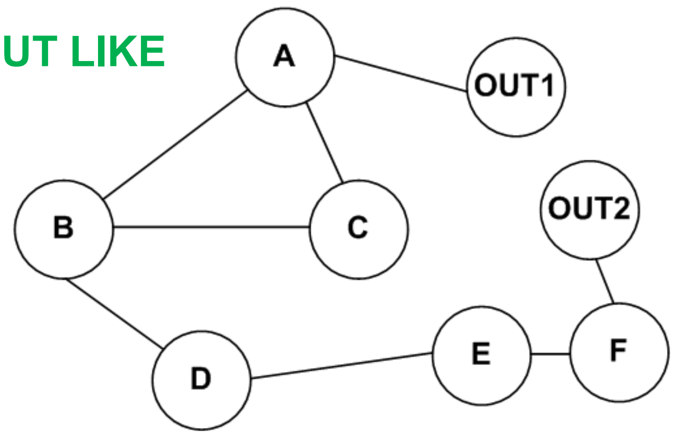
One Abstraction in Analog

- Circuit graph
  - Represents a circuit as a connectivity graph and check equivalence based on their graph structures
  - Layout versus schematic (LVS)
- Observation
  - Not all circuit nodes are directly connected
  - *Could we use this localization property to reduce complexity of comparing two high dimensional transition probabilities ?*

NOT LIKE

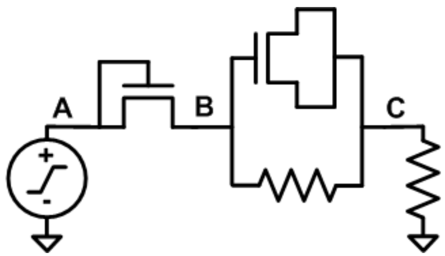


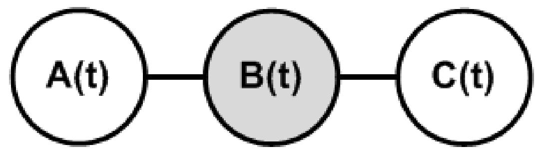
BUT LIKE



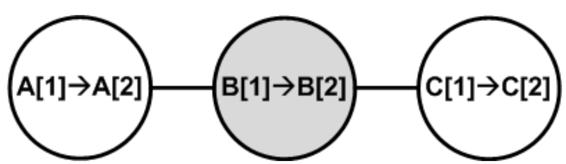
Localization Property and Conditional Independence among Circuit Nodal Responses

- Each nodal voltage/branch currents are unknowns or random variables



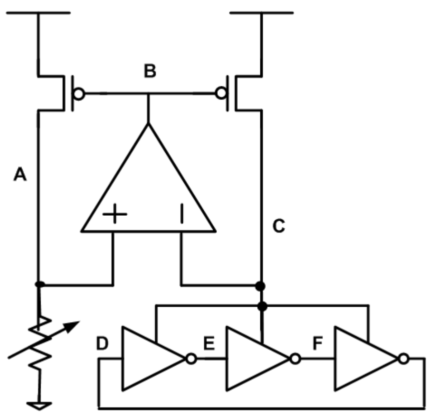


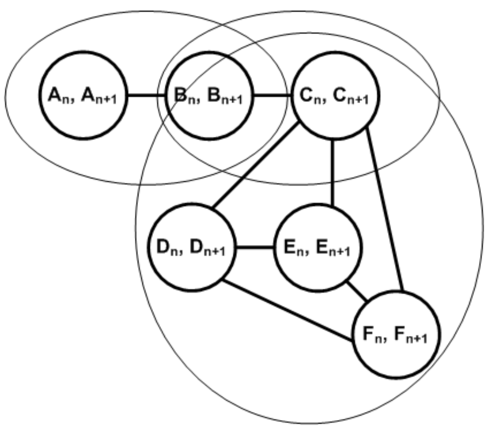
- A circuit is represented by a set of differential-algebraic equations that each equation is formed around these nodes
- With the given small time step, this set becomes algebraic equation set that could be separately solved when their boundary random variables are known – ***conditional independence*** !



Markov Network Representation

- Graphical model that includes all the conditional independences among circuit nodal responses
  - Each nodal voltage corresponds to a random variable
  - Construct an edge between two random variables when they directly interact
- Factorization property
  - Joint state distribution can be factorized into maximal cliques in the Markov network
  - Instead of comparing high dimensional transition probabilities, we can instead compare low dimensional factors of them !
  - $P(A_n, A_{n+1}, B_n, B_{n+1}, C_n, C_{n+1}, D_n, D_{n+1}, E_n, E_{n+1}, F_n, F_{n+1}) = K^* \Phi_1(A_n, A_{n+1}, B_n, B_{n+1})\ \Phi_2(B_n, B_{n+1}, C_n, C_{n+1})\ \Phi_3(C_n, C_{n+1}, D_n, D_{n+1}, E_n, E_{n+1}, F_n, F_{n+1})$



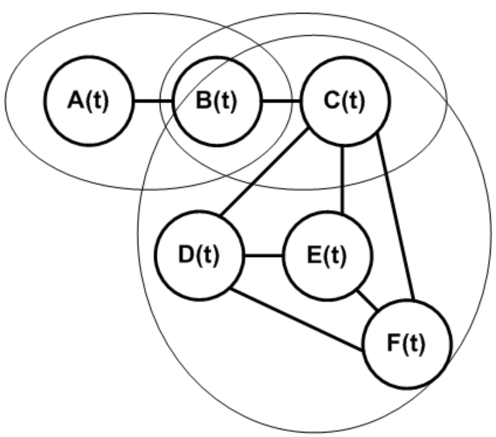


Learning Potentials

- Learning potentials is difficult task for a general graph
  - There could be many different potentials for a given Markov network due to normalization [Sandberg]

$$p(x) = \frac{1}{Z} \prod_{c \in \mathcal{C}} \psi_c(x_c)$$

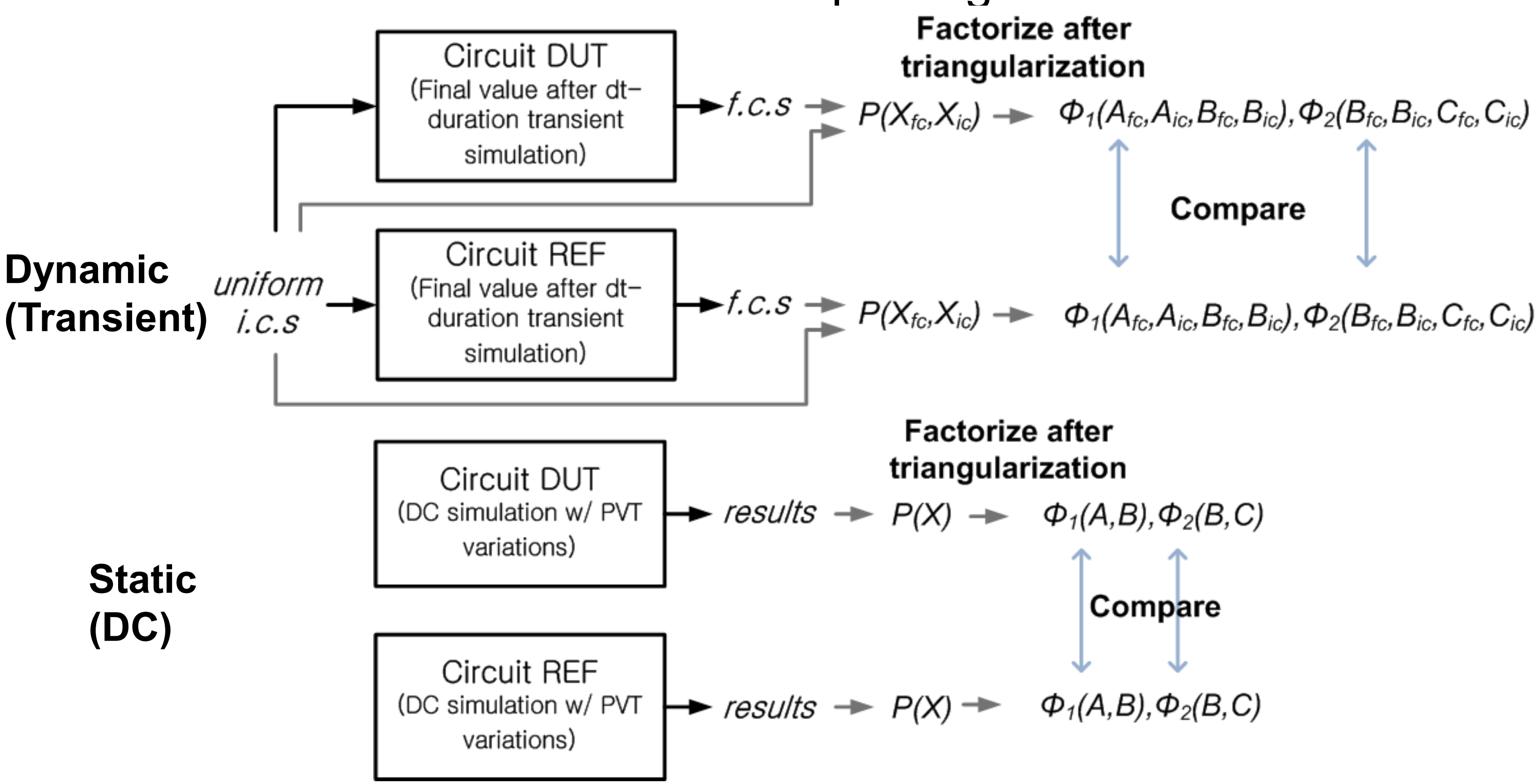
- However, for a decomposable graph, potentials can be estimated as a **marginal** of clique member nodes
  - Normalization factor is not needed to be estimated



P(A,B,C,D,E,F) =
[P(A,B)P(B,C)P(C,D,E,F)] /
[P(B)P(C)]

Markov Network Based Equivalence Checking

- Compare two systems' state distributions by comparing **factors** of two full joint distributions
  - Transition probabilities when inspecting dynamic behaviors
  - State distributions when inspecting static behaviors

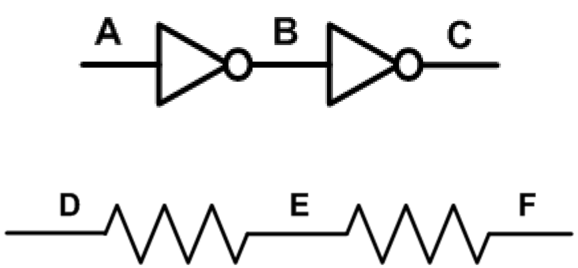


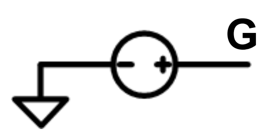
Network Reduction

- Not every node in a circuit is independent random variable
  - In other words, it can be expressed as a function of other variables or just a constant

➔ Not every node in the Markov network is independent random variable

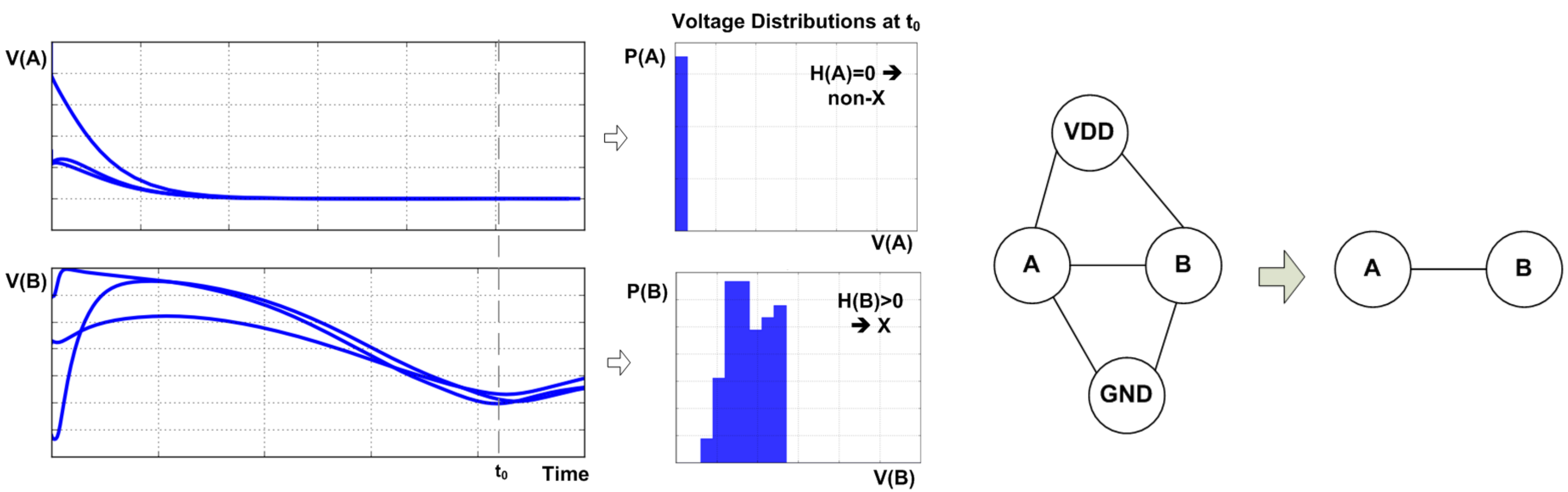
➔ Markov network can be reduced before transition probability comparison





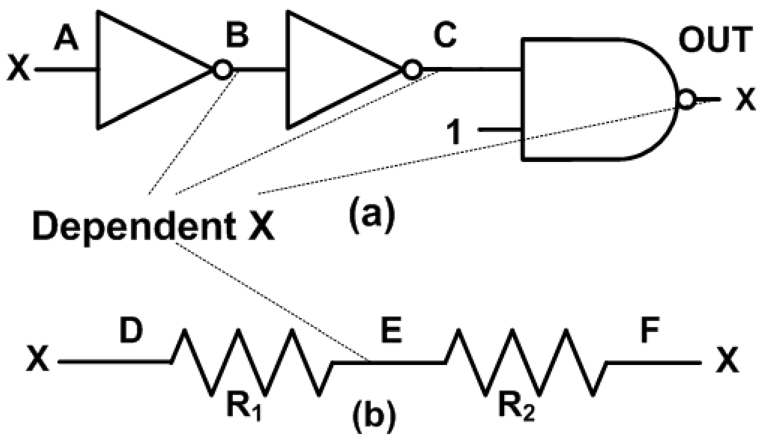
Entropy Based Network Reduction (1)

- Remove determinate nodes
  - Determinate node settles to a certain value regardless of its initial state and it is determined by the other nodes' values ➔ can be removed
  - Determinate node and Indeterminate node can distinguished by "X"
    - Settled / Unsettled ⇔ H(X)=0 / H(X)!=0 ⇔ non-X / X



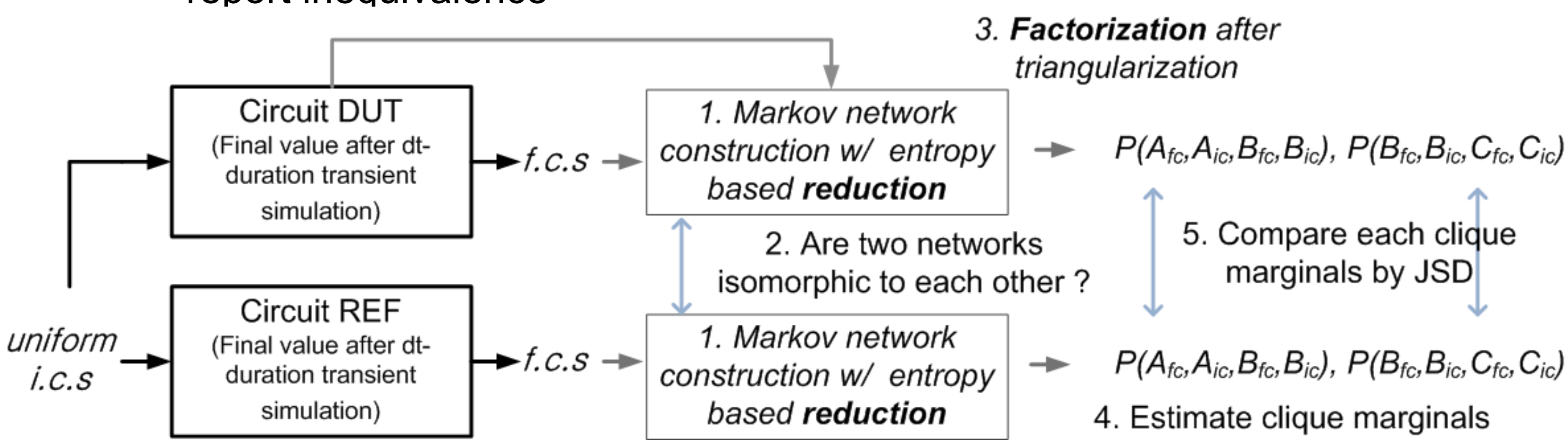
Entropy Based Network Reduction (2)

- Remove dependent nodes
  - If B=f(A), node B can be removed
  - Dependent Xs can be extracted by conditional entropy
    - H(B|A)=0 ➔ B depends on A
    - $H(Y|X_1,..,X_k)=0$ ➔ Y depends on $X_1,\ldots,X_k$



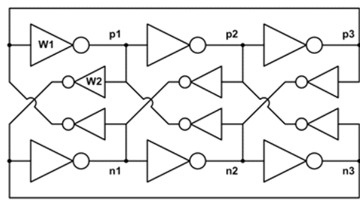
Proposed Markov Network Based Equivalence Check

1. Extract a connectivity graph from two netlists, run short simulations with uniform random initial conditions and reduce the graphs by the entropy based method (Network reduction)
2. Graph isomorphism test (like LVS)
   - If they are different, report inequivalence
3. Triangularize the graph (make graph decomposable)
4. Learn clique marginals of the MN representations of circuits
5. Compare two corresponding clique marginals with Jensen–Shannon divergence
   - $JSD(P \,||\, Q) = D(P \,||\, M)/2 + D(Q \,||\, M)/2,\ M = (P+Q)/2$
   - If any factor of two distributions are different more than the given threshold, report inequivalence

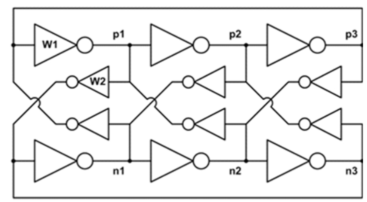


Possible Applications

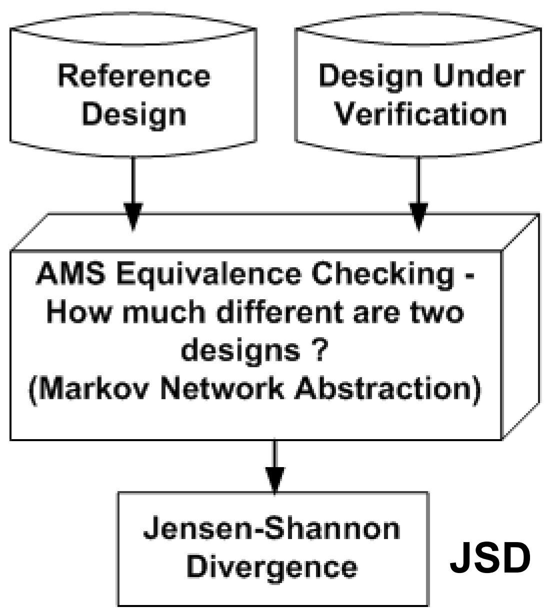
- EX1) two circuit netlists with different process technologies such as 90nm and 130nm technologies
- EX2) schematic netlist and layout extracted netlist with parastics



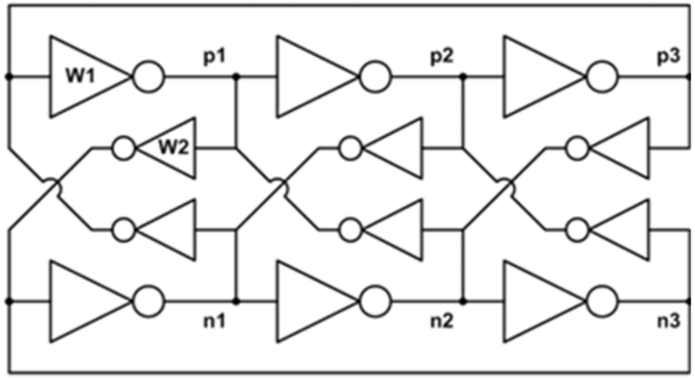
OSC in 130nm tech



OSC in 90 nm tech

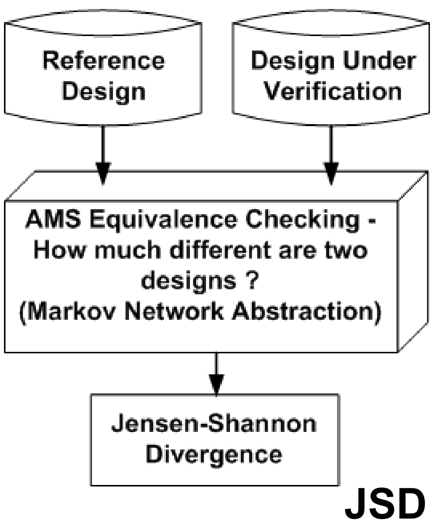


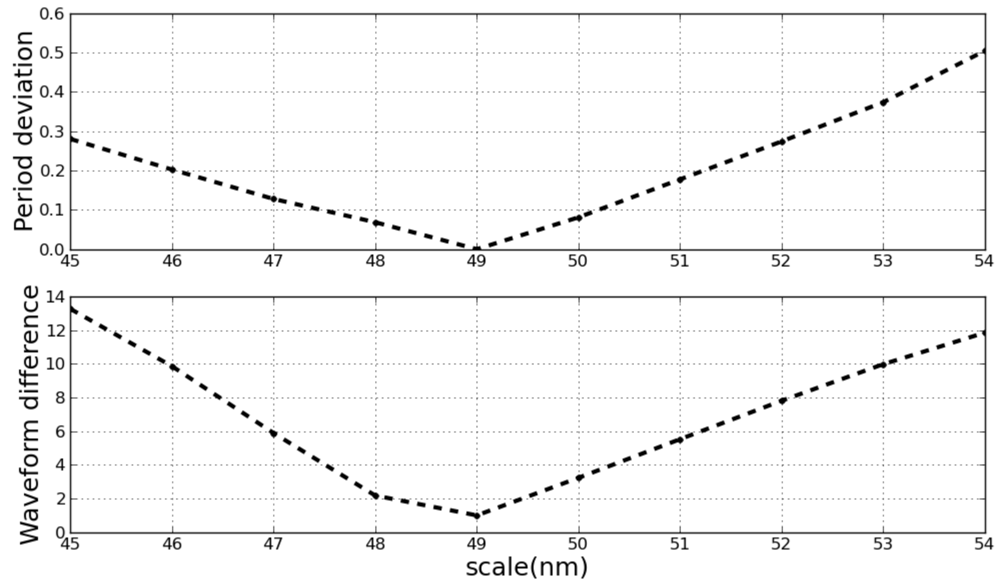
Coupled Ring Oscillator – Scaling from 0.13um to 90nm

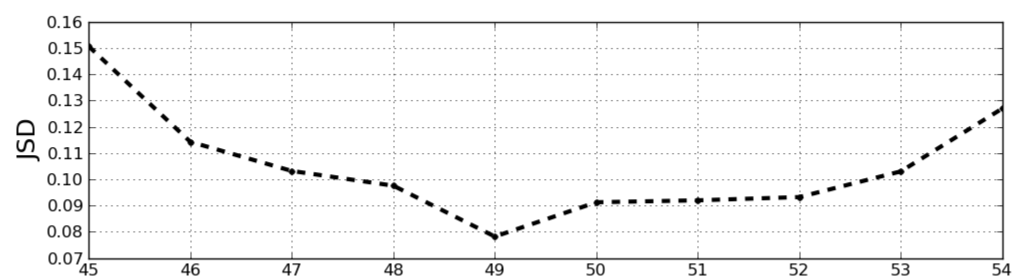


OSC in 130nm tech

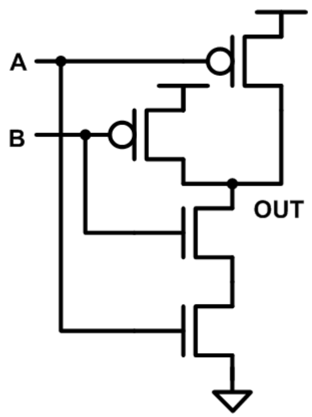
OSC in 90 nm tech

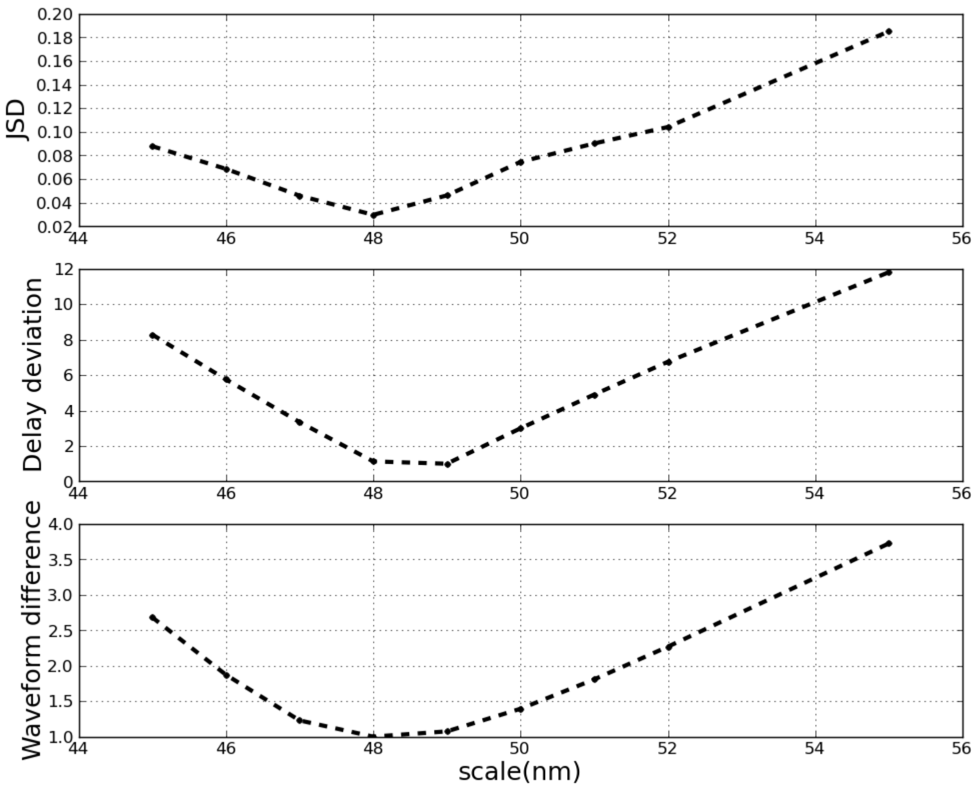




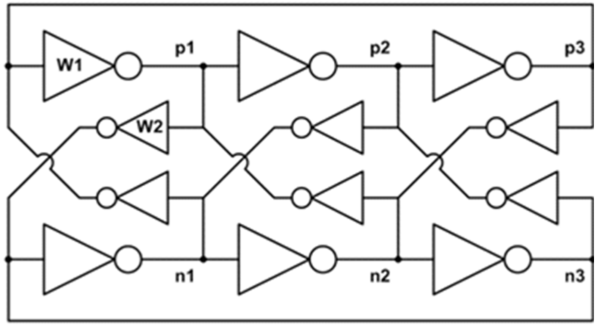


CMOS NAND Gate – Scaling from 0.13um to 90nm

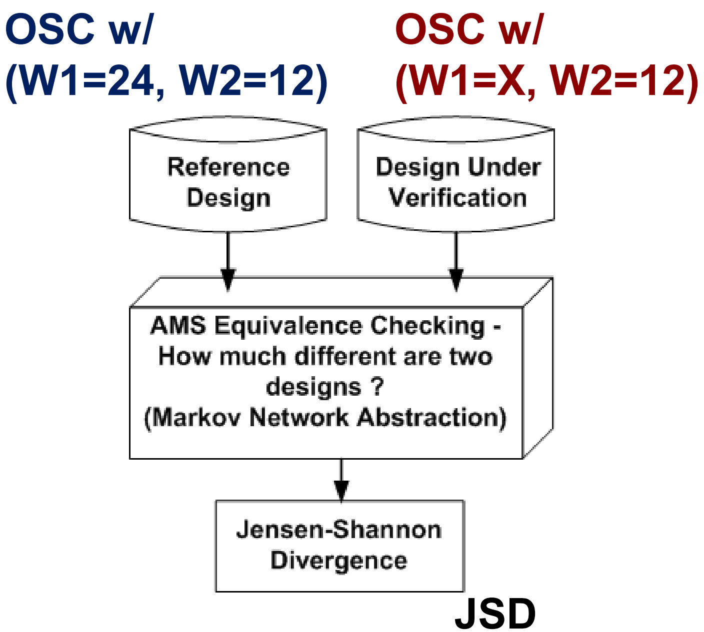


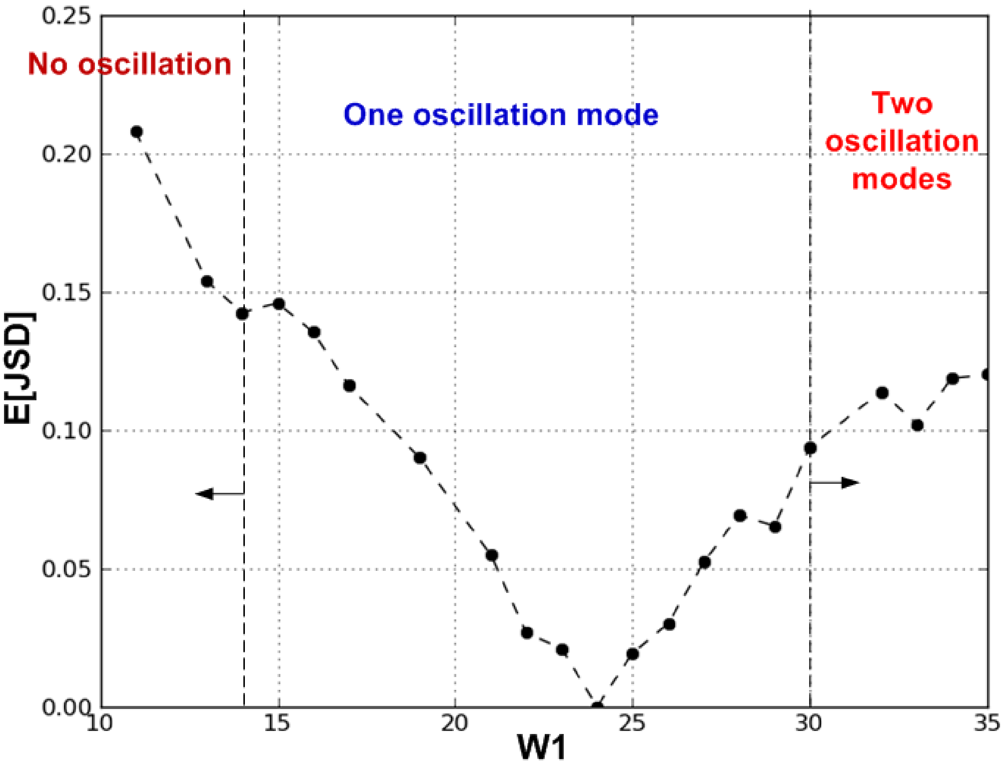


Coupled Ring Oscillator – Two Designs with Different W1:W2 Ratios



- W1 of reference design is 24
- W2 is fixed to 12





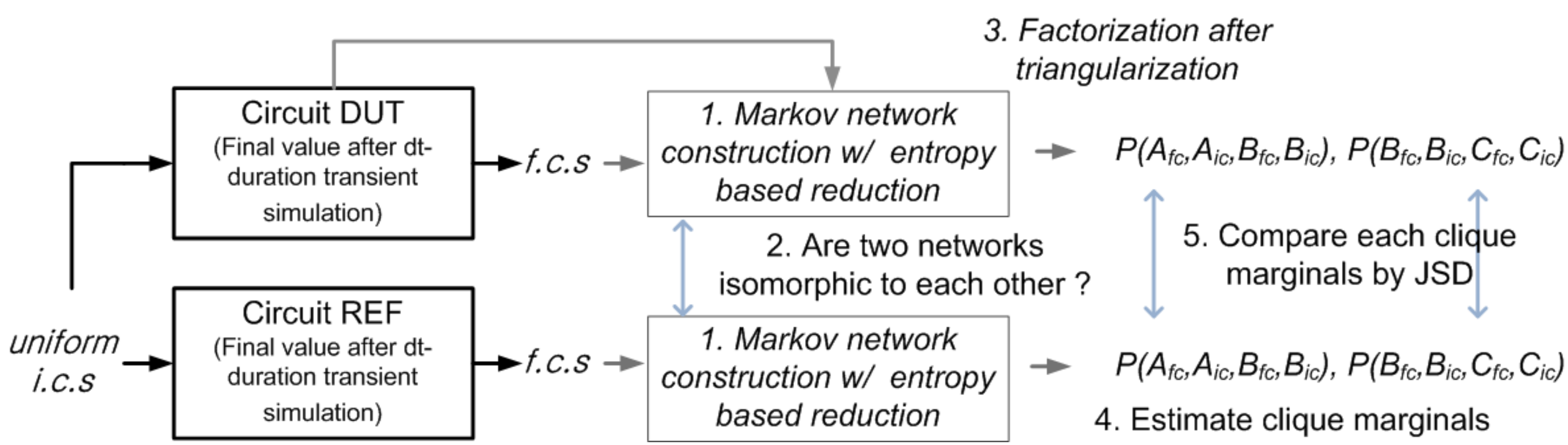
Complexity of the Proposed Equivalence Checking Algorithm

- Compared two designs with the same topology but with different sizes of transistors

	NAND	Oscillator	PLL
# of Circuit Nodes	4	6	374
# of Nodes after Reduction	4	6	225
# of Cliques in the MN	2	8	219
Maximum Clique Size in the MN	3	3	4
Average Clique Size in the MN	3	3	2.3
# of Cliques after Triangularization	2	2	192
Maximum Clique Size after Triangularization	3	5	5
Average Clique Size after Triangularization	3	5	2.9
# of Samples for 2 bit resolution for estimating potentials	64	1024	1024
Run Time of the Proposed Algorithm	13.1 (s)	148 (s)	1.2 (hour)

Conclusion

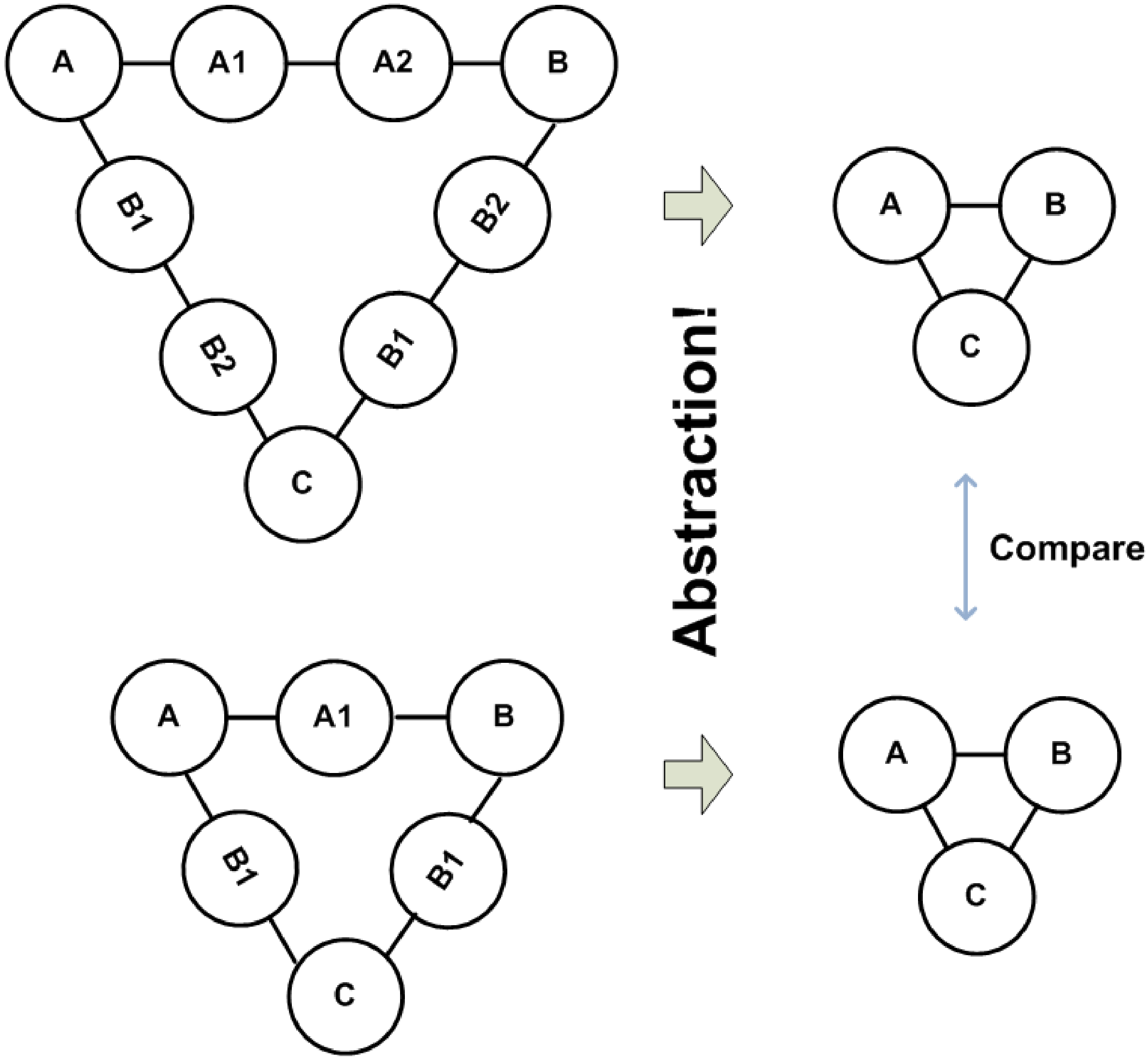
- A way to establish the equivalence between two analog/mixed-signal circuits based on their Markov network representations is proposed
  - Check equivalence between two circuits with the same topology
  - issues
    - How to set threshold?
    - How to relate JSD distance to a circuit's performance metrics



Additional Slides

- Follows

Possible Abstraction



Learning the Potentials

- The general recipe is:
  - (1) For every maximal clique C, set the clique potential to its empirical marginal
  - (2) For every intersection S between maximal cliques, associate an empirical marginal with that intersection and divide it into the potential of ONE of the cliques that form the intersection
- This will give ML estimates for decomposable Graphical Models
- In other words, for decomposable graph, we can acquire potential from empirical marginals for every maximal cliques and their interactions

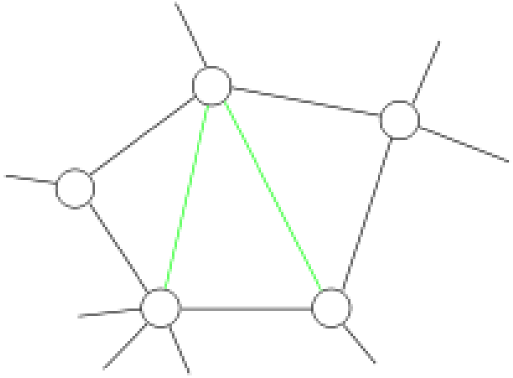
Source: Tibério Caetano, "Graphical Models - Lecture notes"

Comparison of Two Decomposable Markov Networks

- MN1==MN2 ⬄ every maximal clique C1 in MN1 and C2 in MN2 of the same maximal clique C and its intersection S, P1(C)==P2(C) and P1(S)==P2(S)
  - MN1 == MN2 ➔ P1(C)==P2(C) and P1(S)==P2(S) for every maximal cliques
  - P1(C)==P2(C) and P1(S)==P2(S) for every maximal cliques ➔ construct potentials according to the recipe in the previous slide and derive canonical potentials from the previously constructed potentials ➔ every of them is same ➔ MN1==MN2

Decomposable Graph

- Chordal graph
- A graph is chordal if each of its cycles of four or more nodes has a chord, which is an edge joining two nodes that are not adjacent in the cycle



http://en.wikipedia.org/wiki/Chordal_graph

Kullback–Leibler divergence

- A non-symmetric measure of the difference between two probability distributions P and Q

$$
\begin{aligned}
D_{\mathrm{KL}}(P\|Q) &= -\sum_x p(x)\log q(x) &+& \sum_x p(x)\log p(x) \\
&= H(P,Q) &-& H(P)
\end{aligned}
$$

http://en.wikipedia.org/wiki/KL_divergence

Jensen–Shannon divergence

- A popular method of measuring the similarity between two probability distributions. It is also known as information radius (IRad). or total divergence to the average. It is based on the Kullback–Leibler divergence, with the notable (and useful) difference that it is always a finite value.

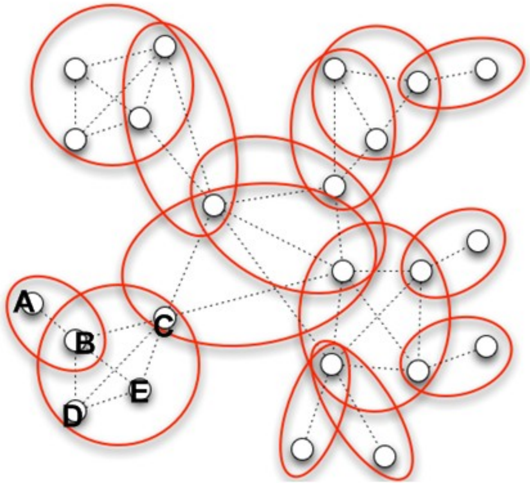
$$JSD(P \parallel Q) = \frac{1}{2}D(P \parallel M) + \frac{1}{2}D(Q \parallel M) \qquad M = \frac{1}{2}(P + Q)$$

$$0 \leq JSD(P \parallel Q) \leq 1$$

http://en.wikipedia.org/wiki/Jensen_Shannon_divergence

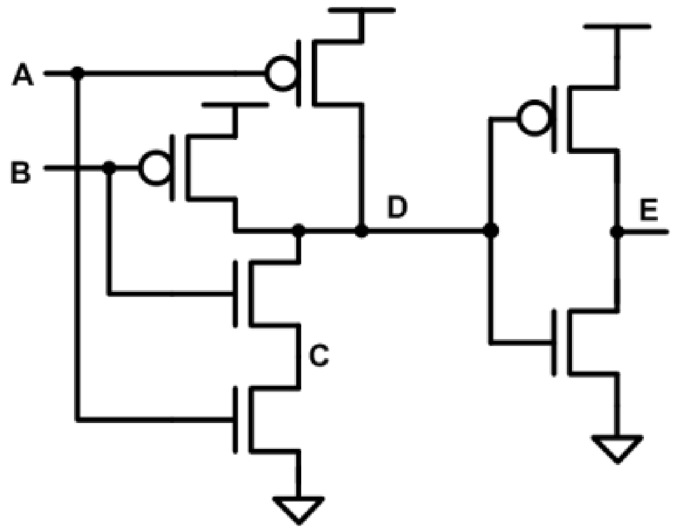
Markov Network

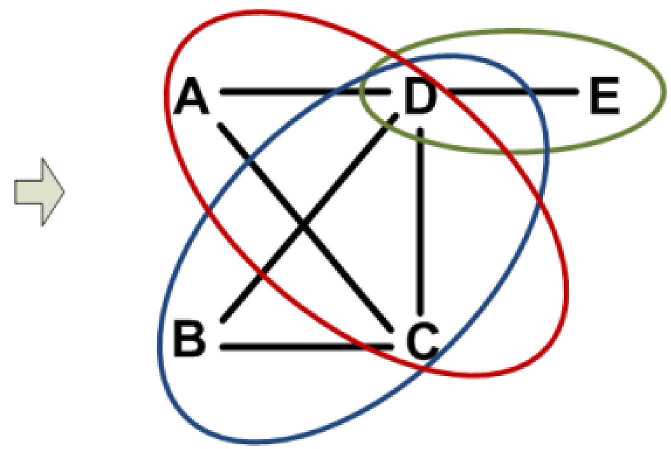
- Undirected graphical model of a set of random variables
- Edges indicate dependency relationships between nodes (r.v.s)
  - ***Conditional independence (CI)***
    - Ex) when B is observed, A and C are independent to each other
- **Factorization** property
  - The joint distribution of all variables can be decomposed to a set of potential functions among varaibles in maximal cliques (assuming strictly positive pdf)
  - P(A,B,C,D,E,…)
    $= \Phi_1 (A,B) * \Phi_2 (B,C,D,E) * \ldots$



Factorization

- Instead of a high dimensional joint distributions of all circuit node variables, it can be expressed as a set of decomposed low dimensional potentials of maximal cliques

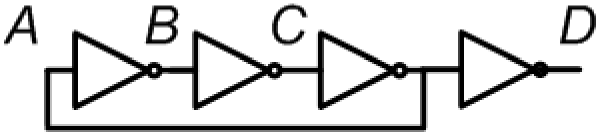


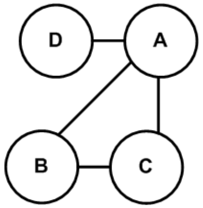


$$P(A,B,C,D,E) = \Phi_1 (A,C,D)\ \Phi_2(B,C,D)\ \Phi_3(D,E)/Z$$

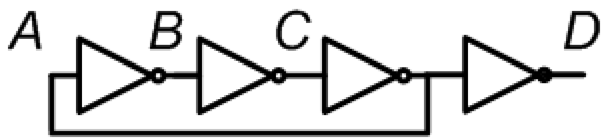
Factorization of Nodal Response Distribution

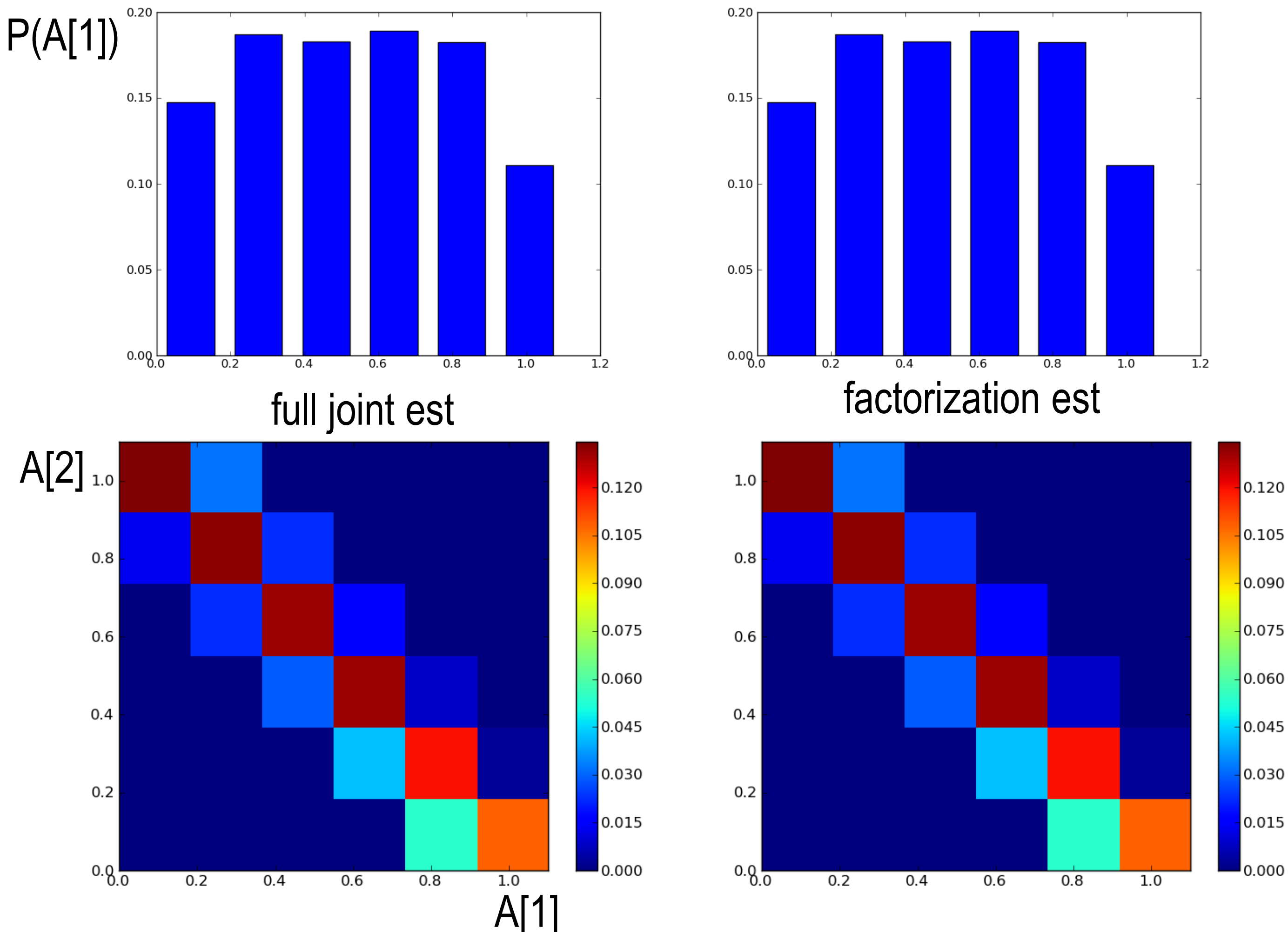
- P1 = P(A2,A1,B2,B1,C2,C1,D2,D1) is directly estimated by samples
- P2 = P(A2,A1,B2,B1,C2,C1)*P(A2,A1,D2,D1)/P(A2,A1)
- JSD(P1,P2) = 0.08671





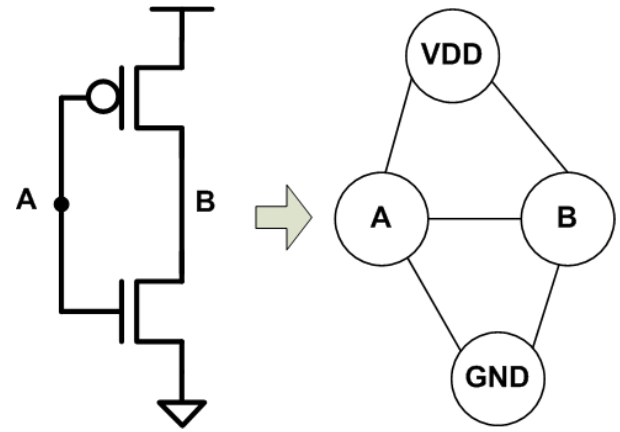
Factorization of Nodal Response Distribution

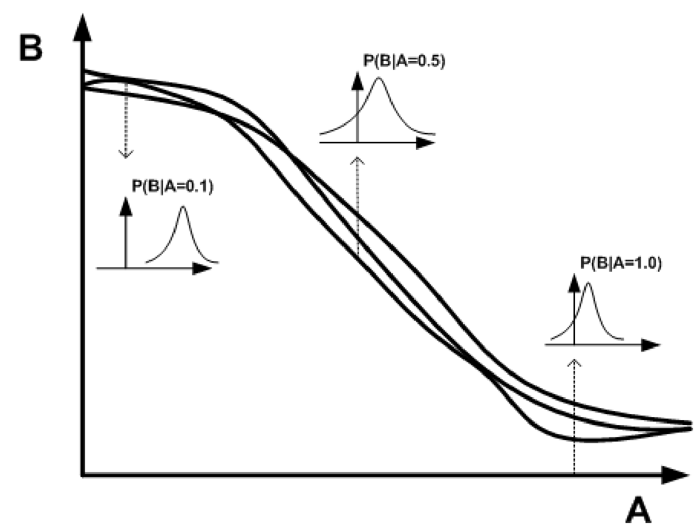




Markov Network Representation – CMOS Inverter Example

- Each circuit nodes in the circuit netlist ➔ Nodes in the Markov network
- Any connection via primitive instances such as resistor, capacitor or transistor ➔ an edge in the Markov network
- Relationships between circuit nodes via nodal equations ➔ probability relationships between nodes

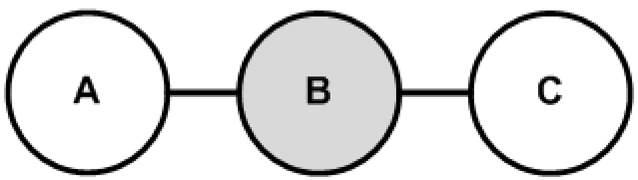




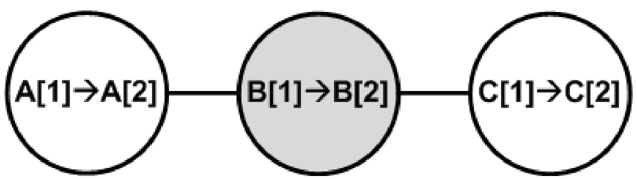
P(A,B,GND,VDD)=P(A,B)=P(B|A)P(A)

Markov Network Based Equivalence Checking

- Compare two systems' transition probabilities by comparing **factors** of two full joint distributions
  - If two full joints are same, any of its conditional distributions (i.e. transitional probabilities) are same as well
  - Static : $P_{ref}(A,B,C)$ and $P_{dut}(A,B,C)$
  - Dynamic : $P_{ref}(A_1,A_2,B_1,B_2,C_1,C_2)$ and $P_{dut}(A_1,A_2,B_1,B_2,C_1,C_2)$



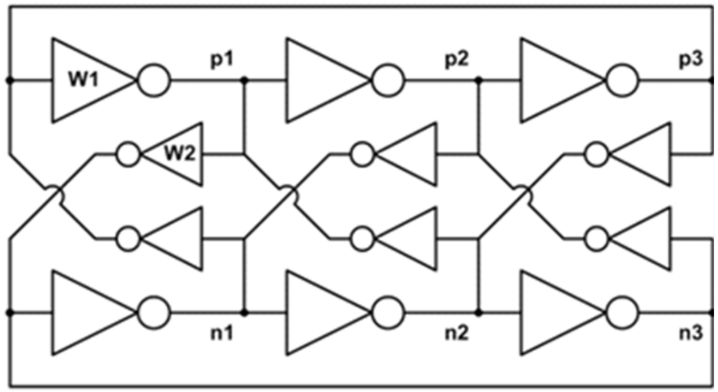
Static (DC)

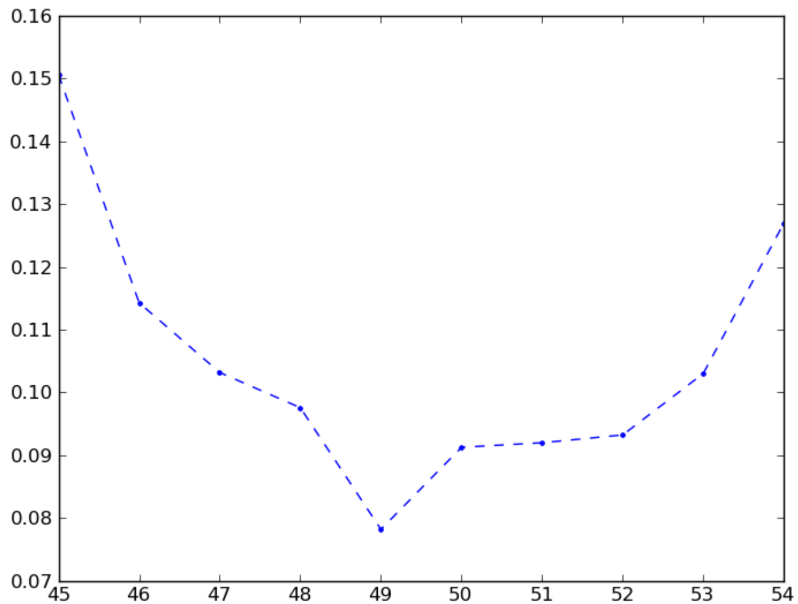


Dynamic (Transient)

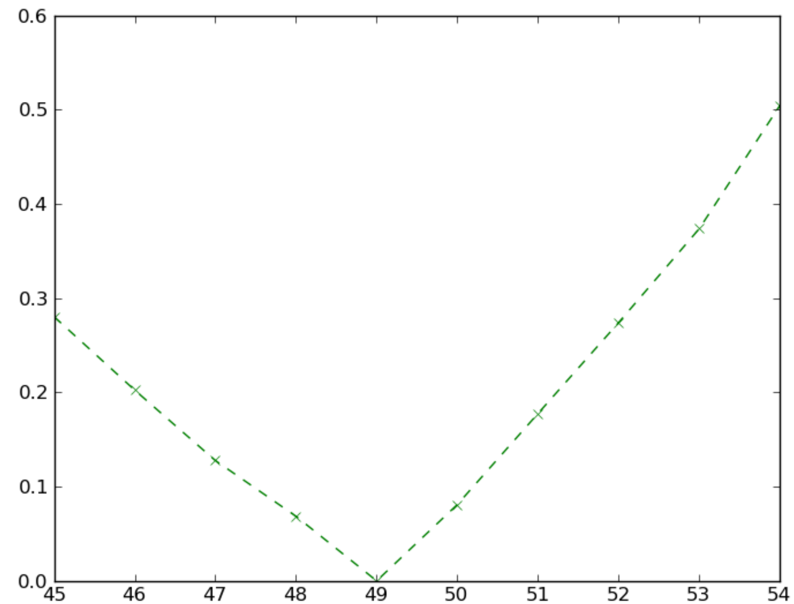
Ring Oscillator –
ptm130 vs ptm090 / scale

- ref – 013





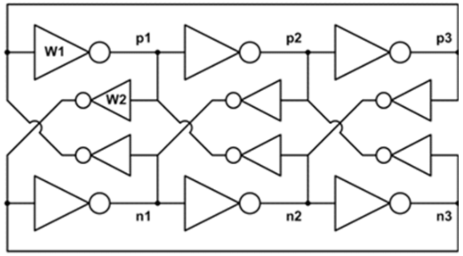
JSD



$|T_{ref}-T_{90}|/T_{ref}$

Ring Oscillator – W1

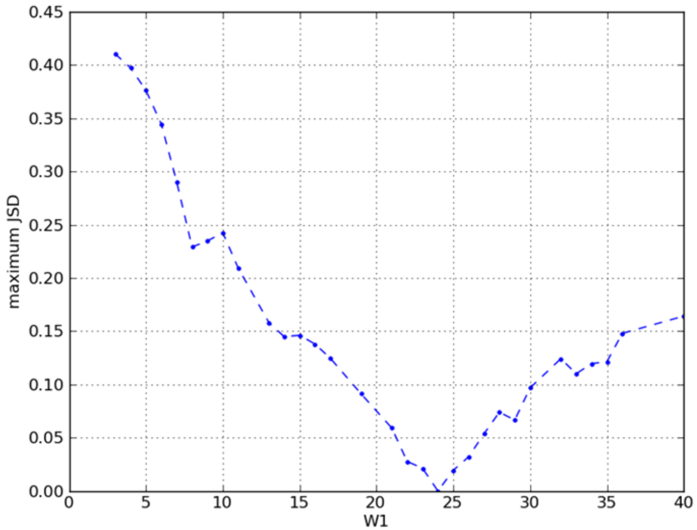
- w2=12



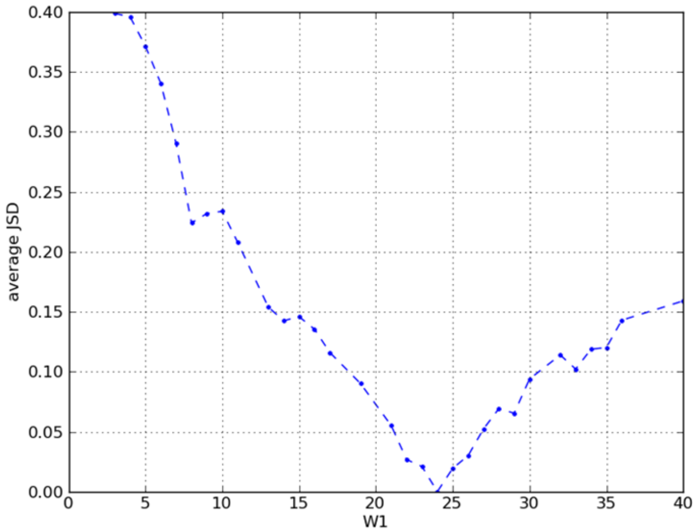
(a)

# of Circuit Nodes	6
# of Cliques in the MN	8
Maximum Clique Size in the MN	3
# of Cliques in the Junction Tree	2
Maximum Clique Size in the Junction Tree	5

(b)

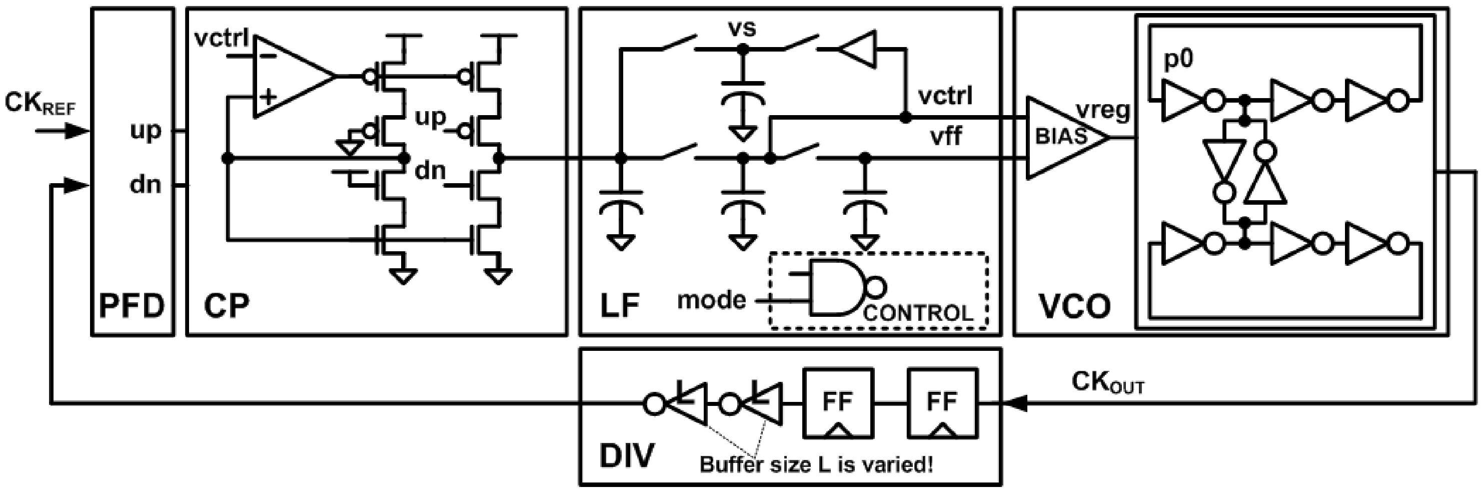


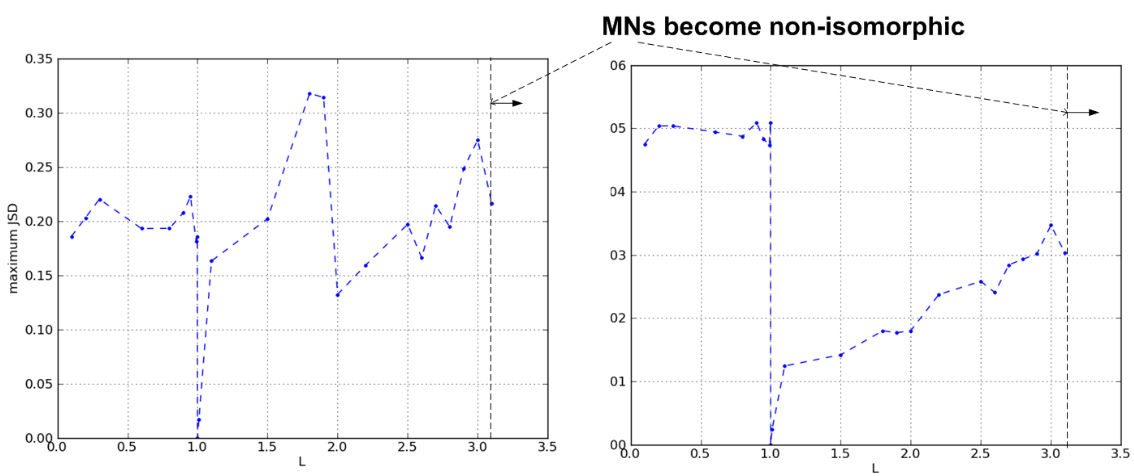
(a)



(b)

PLL – Divider Size





(a) (b)

# of Circuit Nodes	382
# of Nodes after Reduction	109
# of Cliques in the MN	119
Maximum Clique Size in the MN	4
# of Cliques in the Junction Tree	99
Maximum Clique Size in the Junction Tree	5