

# Design and Synthesis in the Presence of Faults

Rupak Majumdar

UCLA

[rupak@cs.ucla.edu](mailto:rupak@cs.ucla.edu)

Joint Work with Lei He and Yu Hu

# Some (Classically) Solved Problems

---

## 1. Discrete Controller Synthesis

Given an open system  $P$ , design a controller  $C$  so that  $P \parallel C$  satisfies specification

## 2. Logic Synthesis

Given the logical description of a circuit, find its optimal implementation

## 3. Compositional Reasoning

When can one component be substituted by another?

# The Boolean Abstraction

---

- Abstract away physical properties and reason about logic
- Well-developed theory of discrete systems (automata, logics, equivalences)

# Some (Classically) Solved Problems

---

## 1. Discrete Controller Synthesis

Theory of infinite games

## 2. Logic Synthesis

Boolean optimization problems (SAT)

## 3. Compositional Reasoning

Simulation, bisimulation, etc

# Rules Change

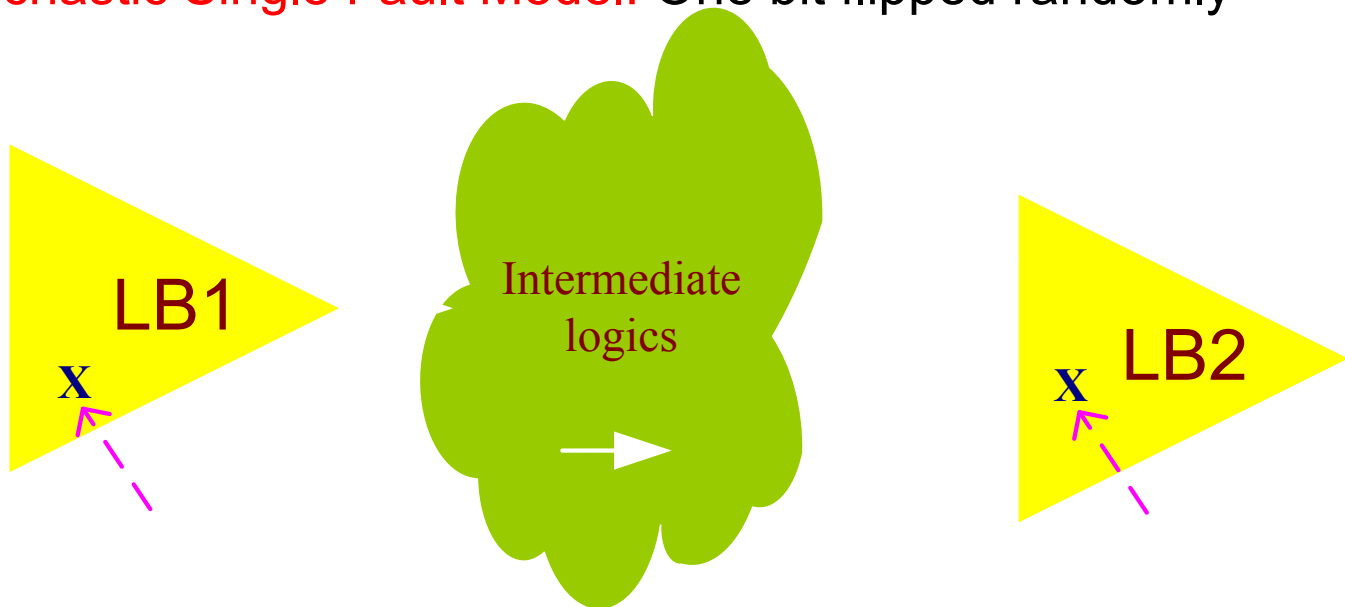
---

- What happens when we must solve these problems in the presence of faults?
- Why faults?
  - Process variations
  - Soft errors
  - Likely to get worse
- What changes?
  - The Boolean model not appropriate

# Modeling of Faults (FPGAs)

---

- Model faults in LUT configurations and faults in intermediate wires as random variables
  - Represent stochastic faults
  - **Stochastic Single Fault Model:** One bit flipped randomly



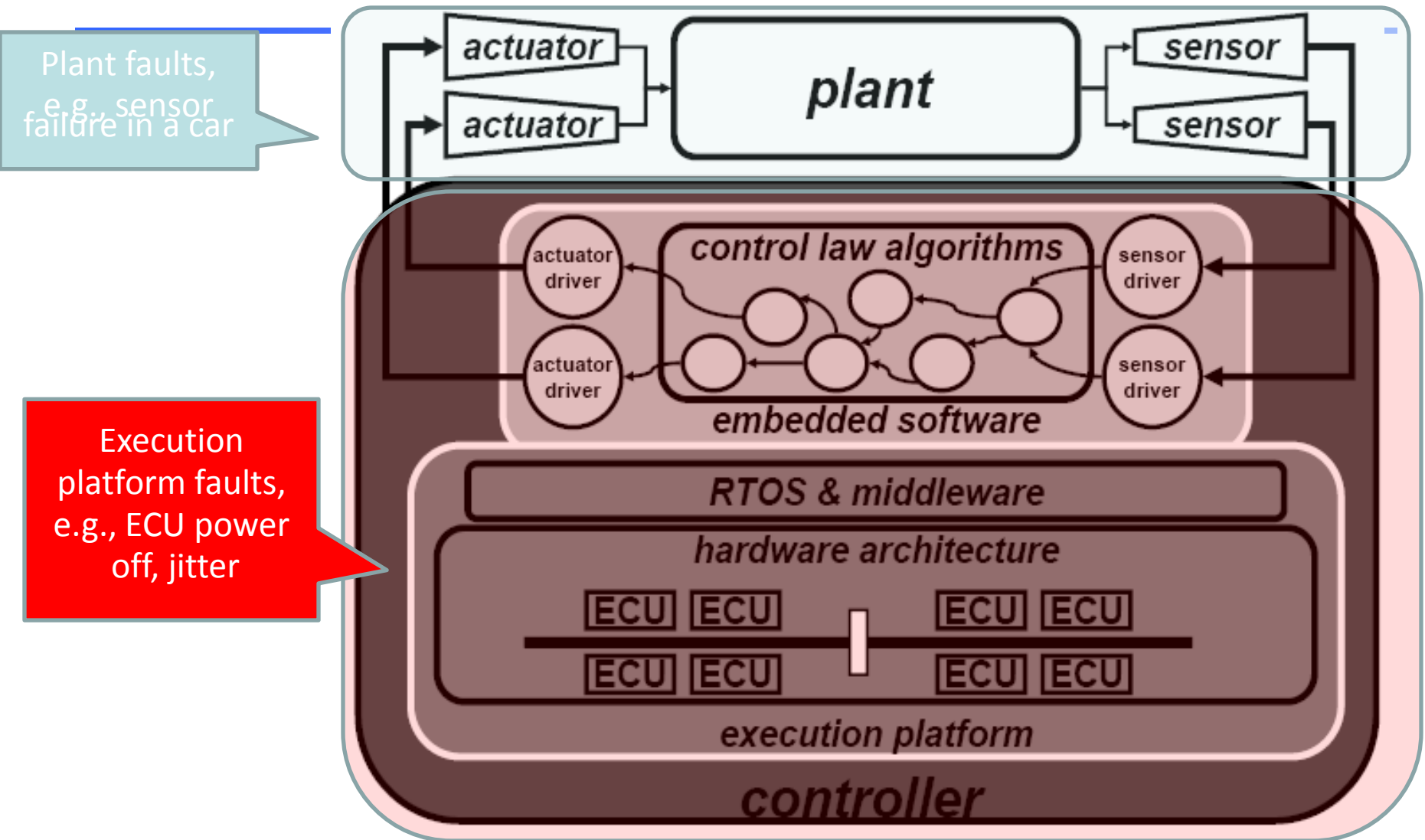
- Corollary of the model: Design must consider probabilistic effects

# “Standard” Solution

---

- For data path and memory, redundancy and Error Correction Coding
- For random logic, Triple Modular Redundancy
  - High overhead
  - TMR has over 5x-6x area/power overhead (for FPGA)
- What if this overhead is too high?
  - Opportunities for EDA ...
  - But need theoretical and engineering advances

# 1. Control Systems Implementation



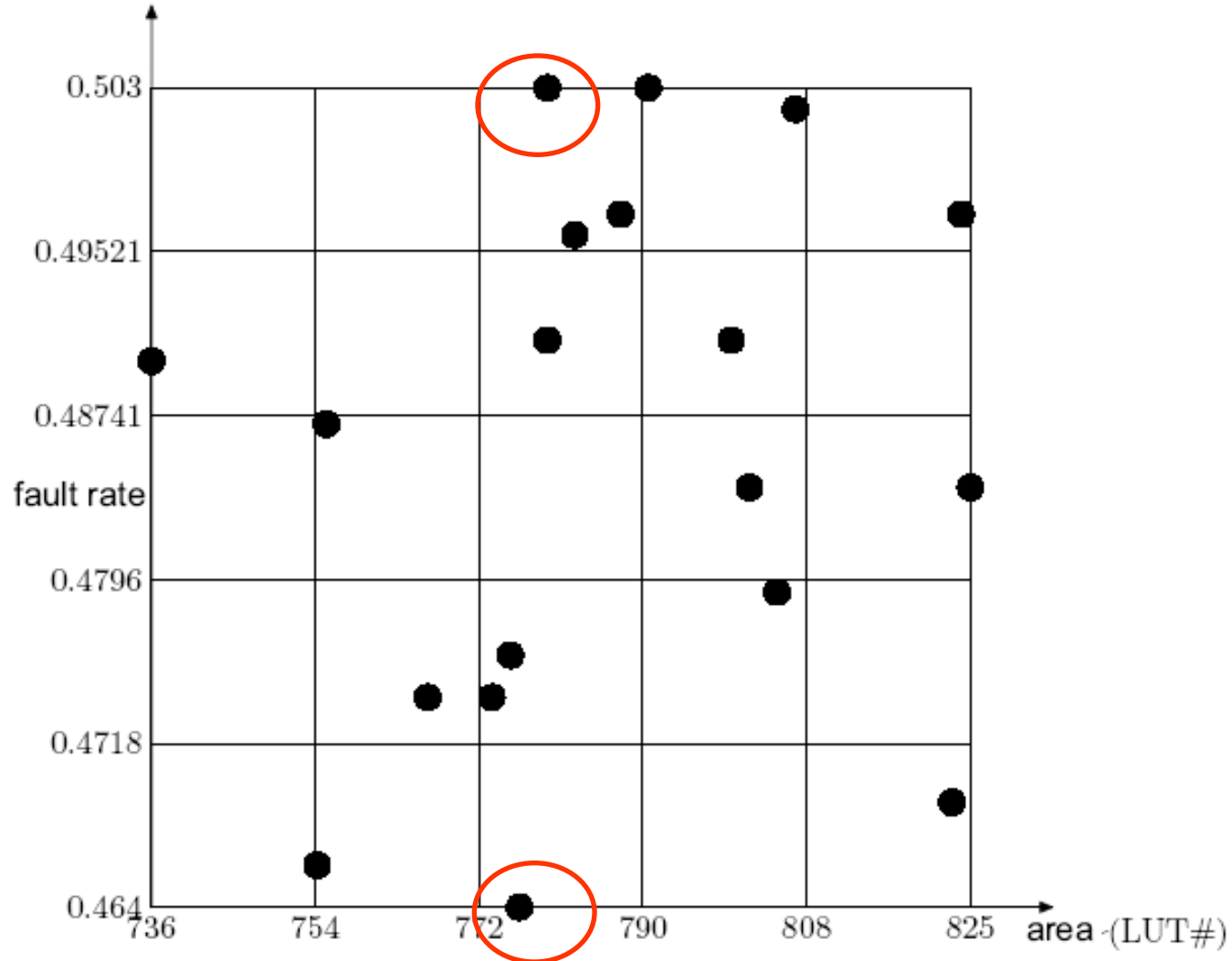


# Fault-Aware Controller Synthesis

---

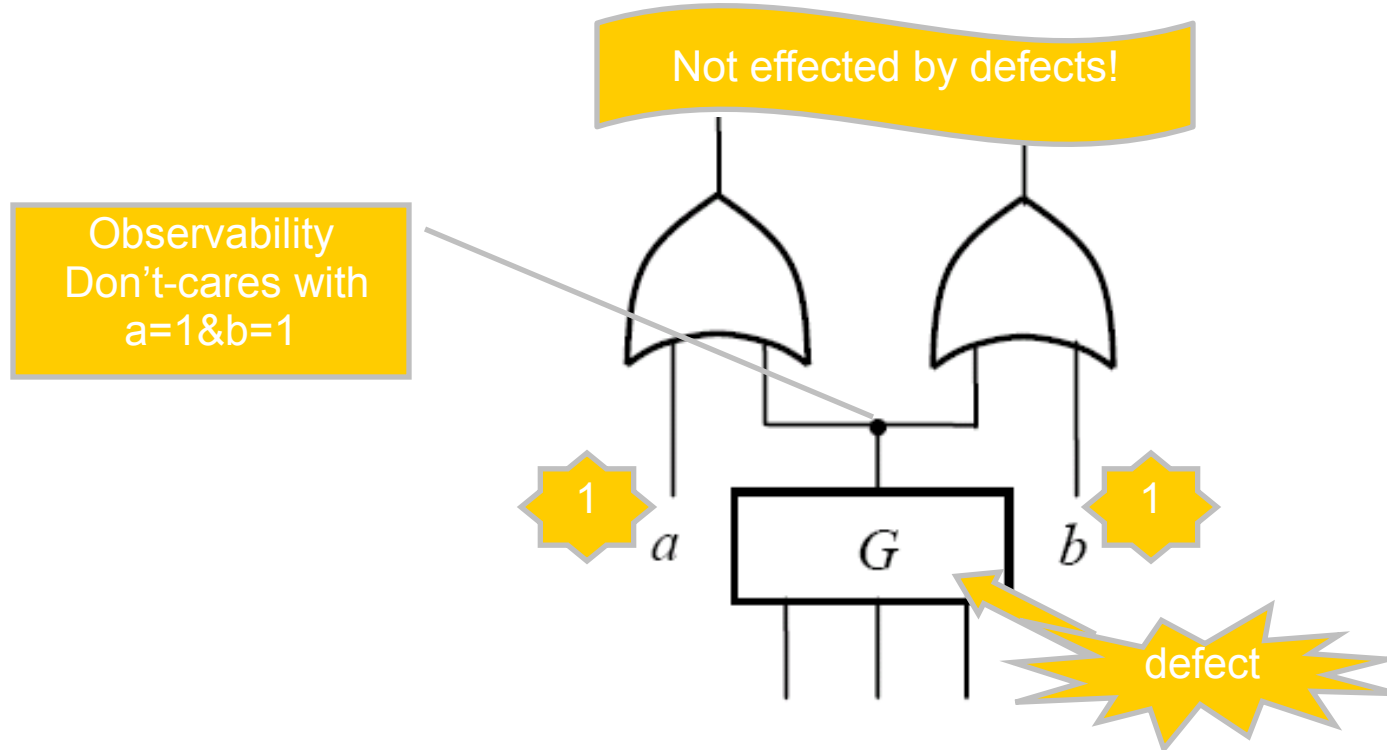
- Controller synthesis assumes deterministic, or worst case behavior
- Controller is given as a finite-state machine
  - Implemented as a sequential circuit
- Encoding problem: How to encode controller so that  $\Pr [C' \text{ winning}]$  is maximal, where  $C' = C$  with stochastic implementation error
  - Theory of stochastic synthesis (not yet practical)

## 2. Logic Synthesis for Reliability



# One Approach: Logic Masking for Reliability

- Defects are created equally but not propagated equally
- Logic don't-cares may mask the propagation of defects



[Markov et al07, HuZhengHeM.08, LeeHuHeM.09]

# Boolean Matching

---

- Boolean Matching
  - Inputs
    - PLB  $H$  and Boolean function  $F$
  - Outputs
    - Either that  $F$  cannot be implemented by PLB  $H$
    - Or the configuration of  $H$
- Reduction to SAT ( $\Sigma_2$ )

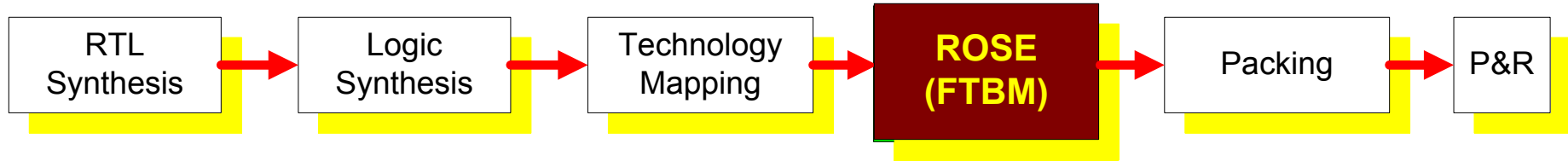
# Fault Tolerant Boolean Matching

---

- Fault Tolerant Boolean Matching
  - Inputs
    - PLB  $H$  and Boolean function  $F$
    - Fault rates for input bits and SRAM bits
  - Outputs
    - Either that  $F$  cannot be implemented by PLB  $H$
    - Or the configuration of  $H$  which minimizes the probability that the faults are observable in the output of the PLB under all input vectors
- Reduction to Stochastic SAT (E-MAJSAT)
  - Variables can be “randomly” quantified
  - Find solution that maximizes expected value
  - Needs more engineering work...

# ROSE: Robust Resynthesis

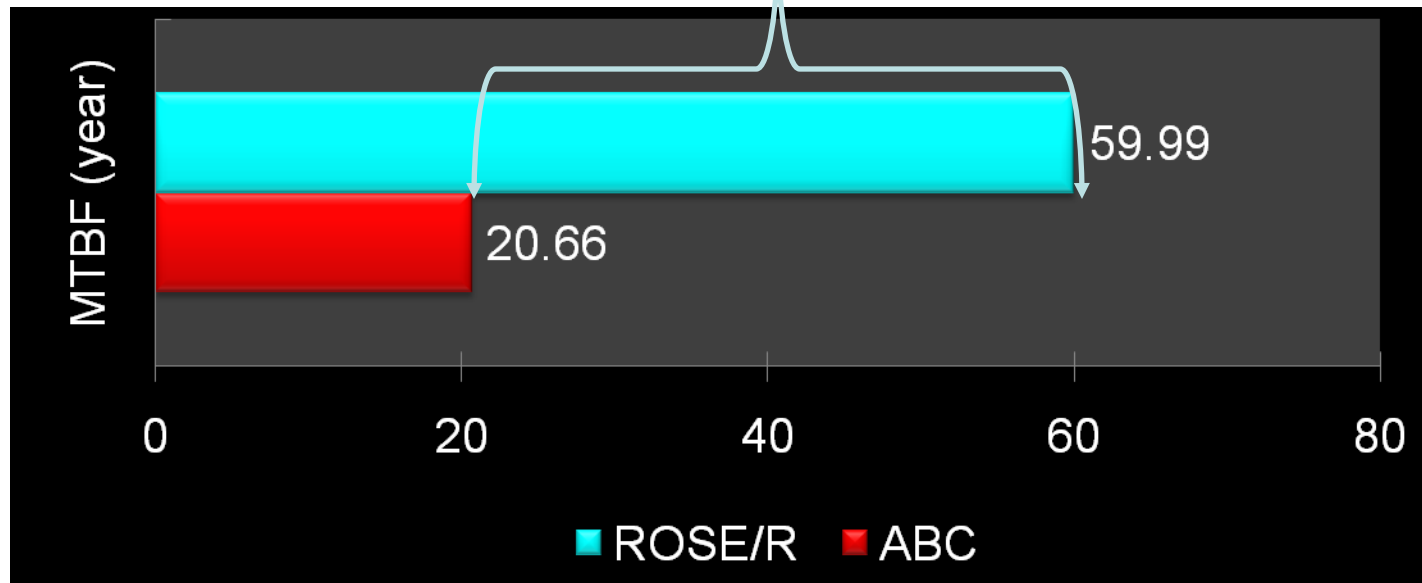
---



- Resynthesis based on FTBM:
  - Step 1: Find a Boolean matching solution
  - Step 2: Evaluation the stochastic fault rate of a solution
  - Can integrate with flow/physical design (without affecting design closure)

# Estimation of Mean Time To Failure

- Assume a chip with **330,000 LUTs**  
2X MTTF increase!



MTBF estimation [Mukherjee, HPCA, 2005]

- $MTBF = 10^9 / (24 \times 365) FIT_{total}$
- $FIT_{total} = 100 \times \text{Vulnerability Rate} \times \text{Intrinsic Error Rate}$
- Intrinsic Error Rate = Area x FIT rate, Vulnerability Rate = Mean of fault rate
  - FIT rate = 0.01 FIT/bit

# Only the Beginning...

---

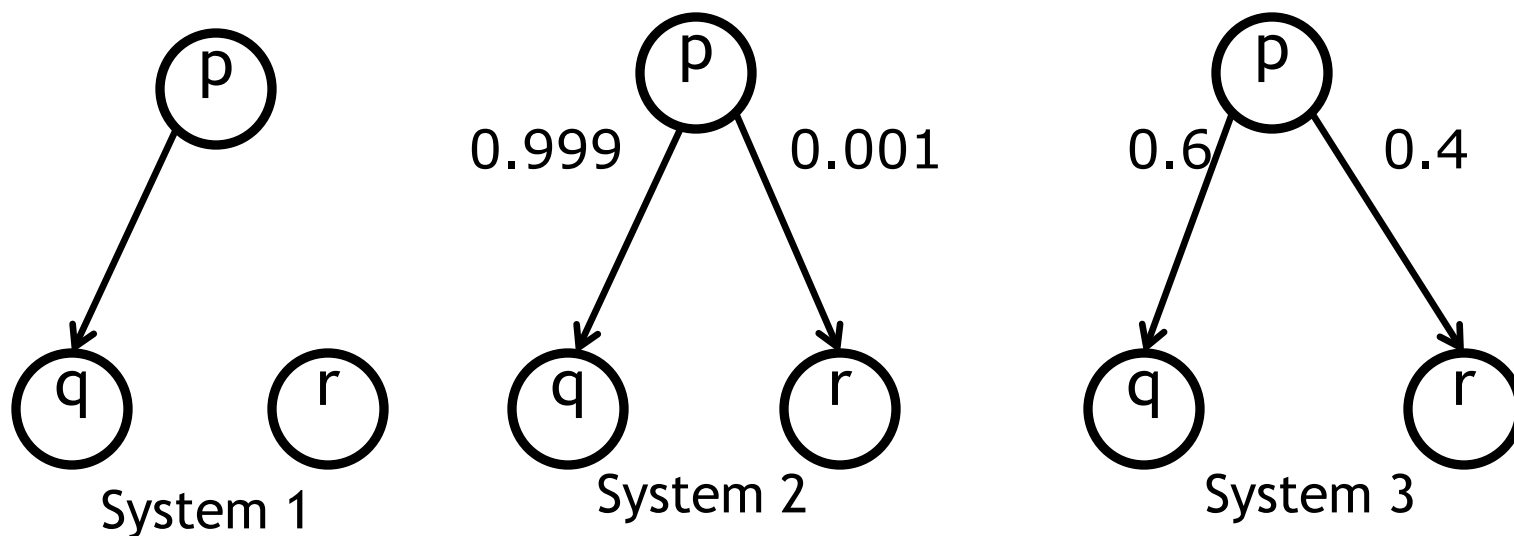
- Logic synthesis with reliability as an explicit objective
  - Multi-objective: maximize reliability under area/power/... budget constraints



# 3. Compositionality

---

- Problem: Classical behavioral equivalences are too precise



Not (probabilistic) bisimilar, but intuitively,  
System 2 is closer to System 1 than System 3

# Metrics on Systems

---

- Generalize behavioral equivalences to metrics

[Panangaden et al, WorrellOuaknine, deAlfaroM.RamanStoelinga07]

- Generalize logics to quantitative logics
- Robustness Theorem:

$$d(s,t) = \sup \{ \phi(s) - \phi(t) \mid \phi \text{ in } \mu \text{ calculus} \}$$

Open: Efficient algorithms for metrics?

# Conclusion

---

- EDA = Optimization of designs under constraints
- Emerging research opportunities for design under reliability constraints