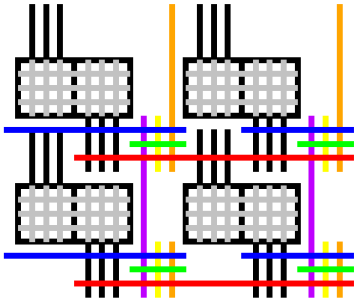


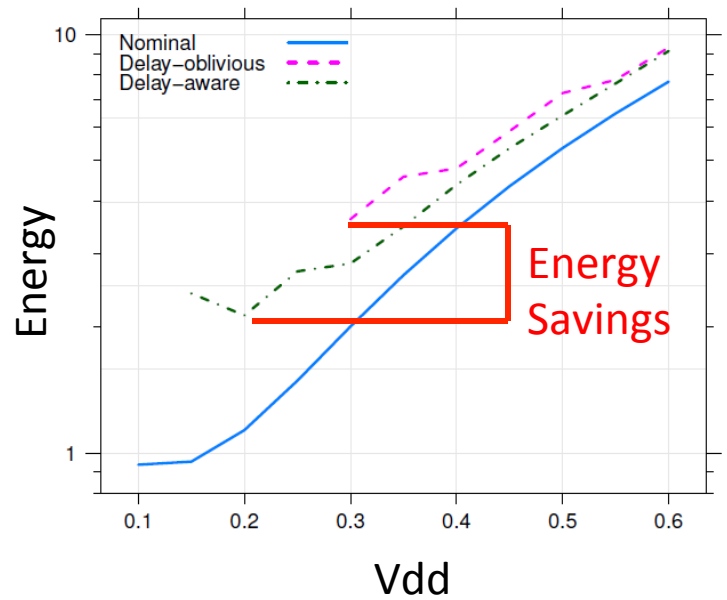
Limit Study of Energy & Delay Benefits of Component-Specific Routing

Nikil Mehta, Rafi Rubin, Andre DeHon

Penn Engineering

CALTECH

Overview

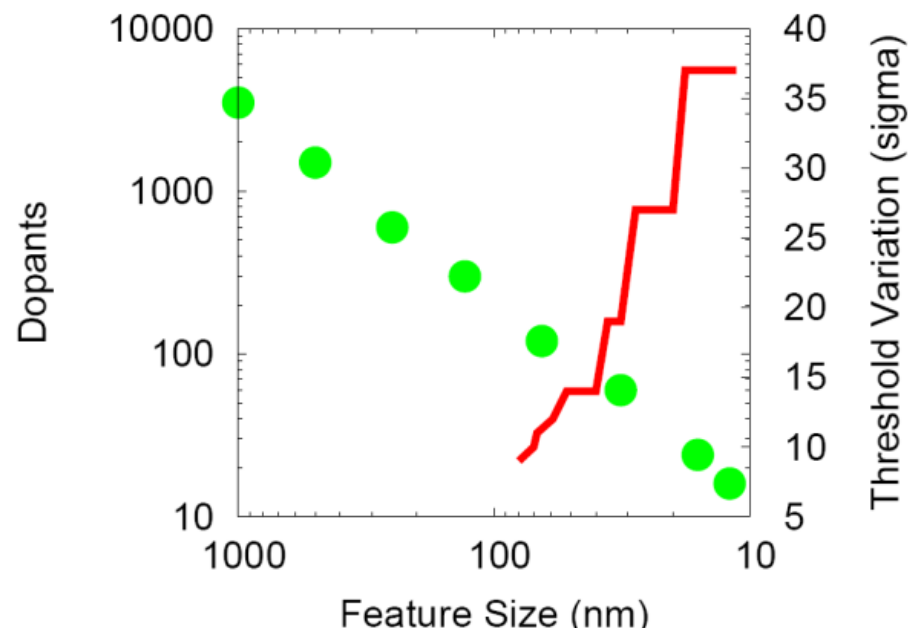


- Random variation challenge
  - Every transistor is different
  - Must margin delay & energy
- What if we created variation map per chip?
- If we *route per chip* using delay information:
  1. **Eliminate delay margins**
  2. **Reduce energy by 1.42-1.98x**
  3. **Extend *minimum energy* scaling by a technology generation**

Motivation

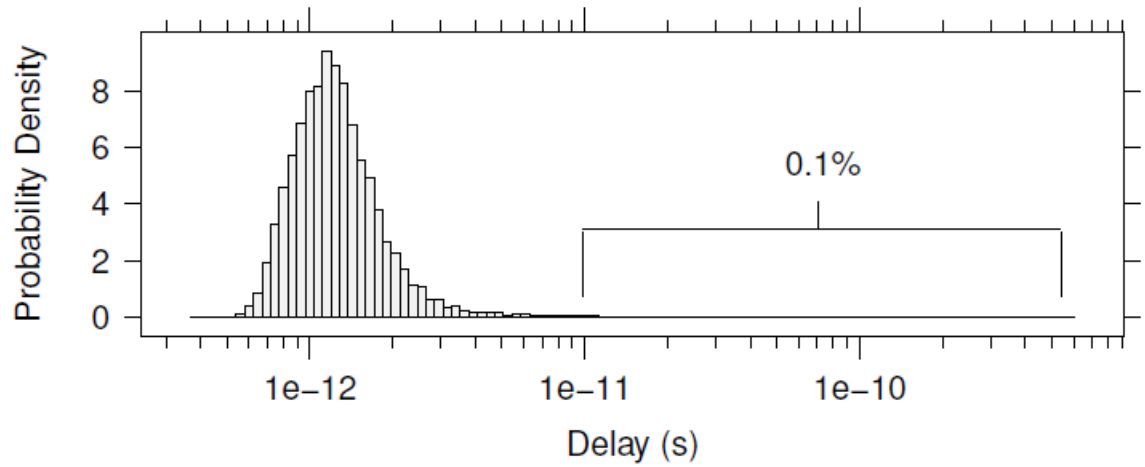
- Random V_{th} variation
  - Cannot know pre-fabrication
  - Getting worse
  - Fundamental, atomistic result of doping

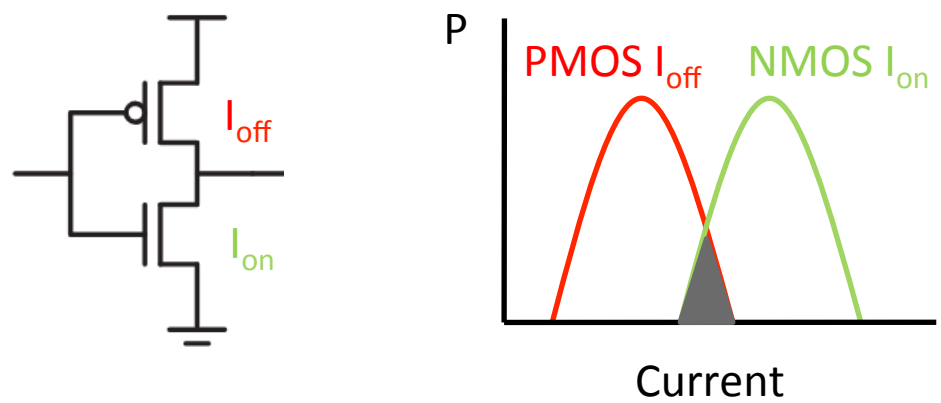
Source: ITRS 2010



Impact of V_{th} Variation

- Devices
  - Slow down
  - Leak
  - Fail
- Raise V_{dd} to avoid
  - V_{th} variation much worse at low V_{dd}
  - V_{dd} not scaling
- Delay/Energy margins

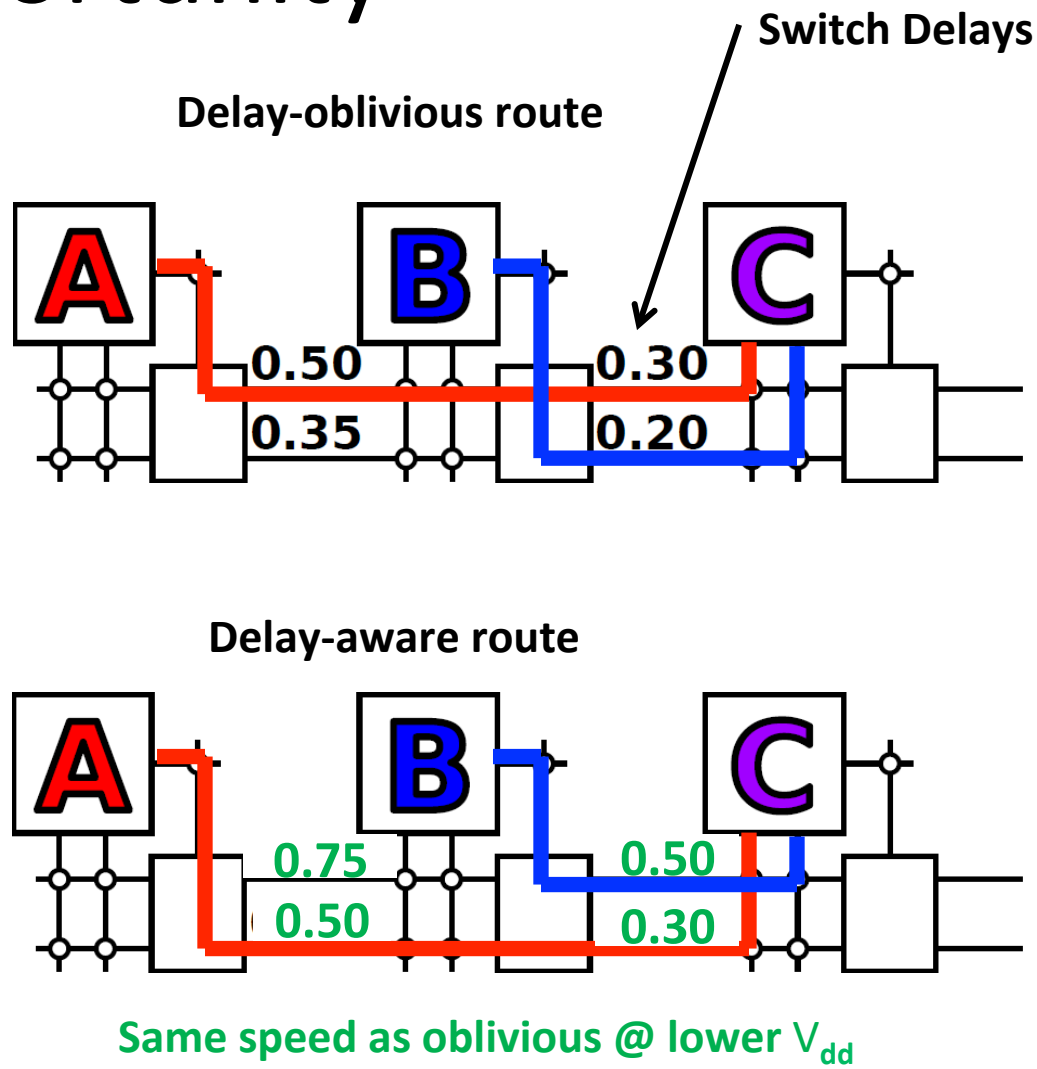




$I_{on} < I_{off}$

Opportunity

- FPGAs have reconfig!
- Measure variation then route
  - Avoid defects?
  - Faster?
  - Lower V_{dd}?
- **Limit Study**
  - Delay-*oblivious* vs Delay-*aware* routing



Challenges

- Obvious solution, but practical?
  - Disrupts FPGA model of one CAD run per *design*
    - Rubin, "Choose-Your-Own-Adventure Routing", FPGA 2009
  - How do we measure every device?
    - Sedcole, "Within-die Delay Variability in 90nm FPGAs" FPT 2006
  - How do we store this information?
  - **How do we use this information to map?**
  - **Is this worth the trouble?**

NOT This paper

This paper ("Limit Study")

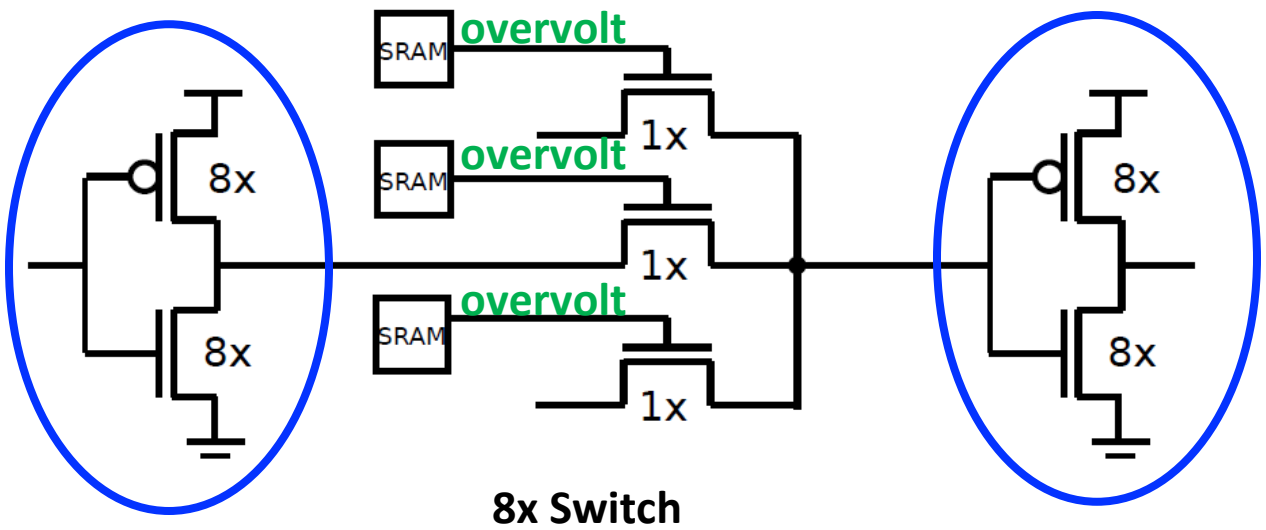
Methodology

- Delay & Energy model
  - PTM models (45nm-12nm)
  - ITRS variation (27% @22nm)
- Modify VPR 5.0
- Caveats
  - Only random V_{th} variation
  - Only interconnect
- More details in paper

- Experiment
  - Single placement
  - 50 random chips
  - Delay-oblivious
    - Route once, evaluate each chip
  - Delay-aware
    - Route each of 50 chips
  - 90% parametric delay/energy
    - 45th slowest/most energy chip

Sizing

- Critical in optimizing delay and energy
- Examined 1x-32x sized switches

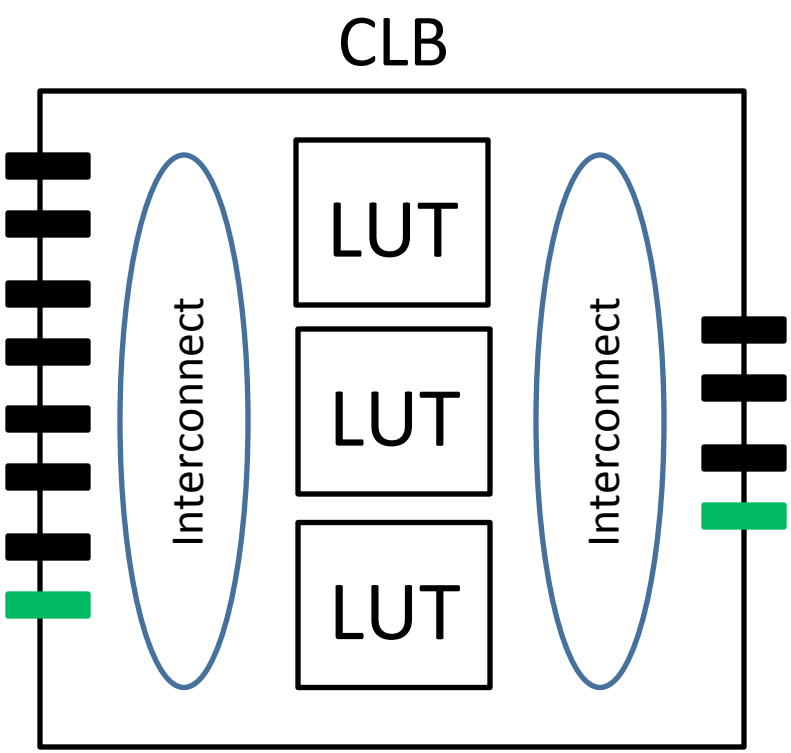


$\frac{3}{4}V_{th} \propto \sqrt{1/WL}$

$\frac{3}{4}V_{th}$ @ 22nm, 1x = 27%

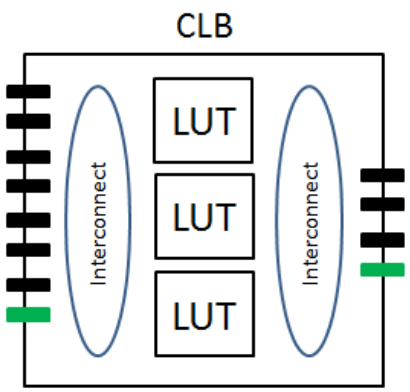
$\frac{3}{4}V_{th}$ @ 22nm, 8x ¼ 9%

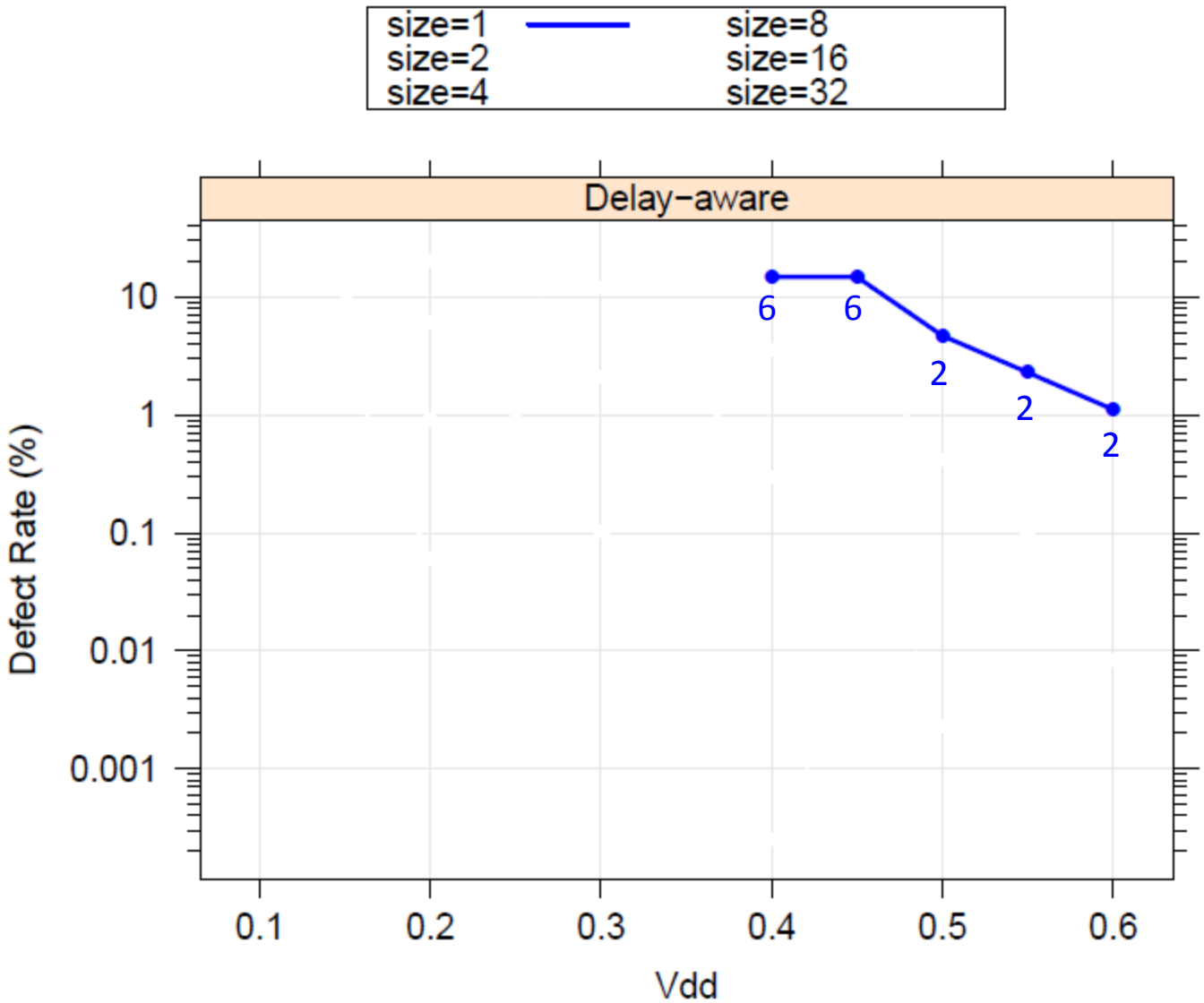
Sparing



- CLB I/O pins are bottleneck
  - Low V_{dd} defects
  - Cannot yield if I/O pins defective
- Add spare pins

Defect Rates

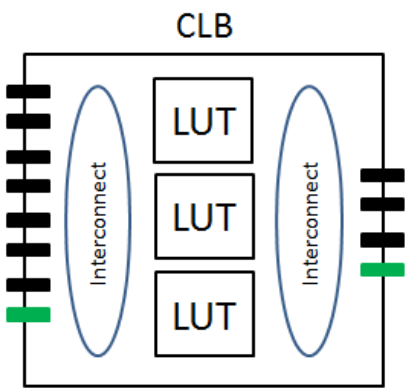


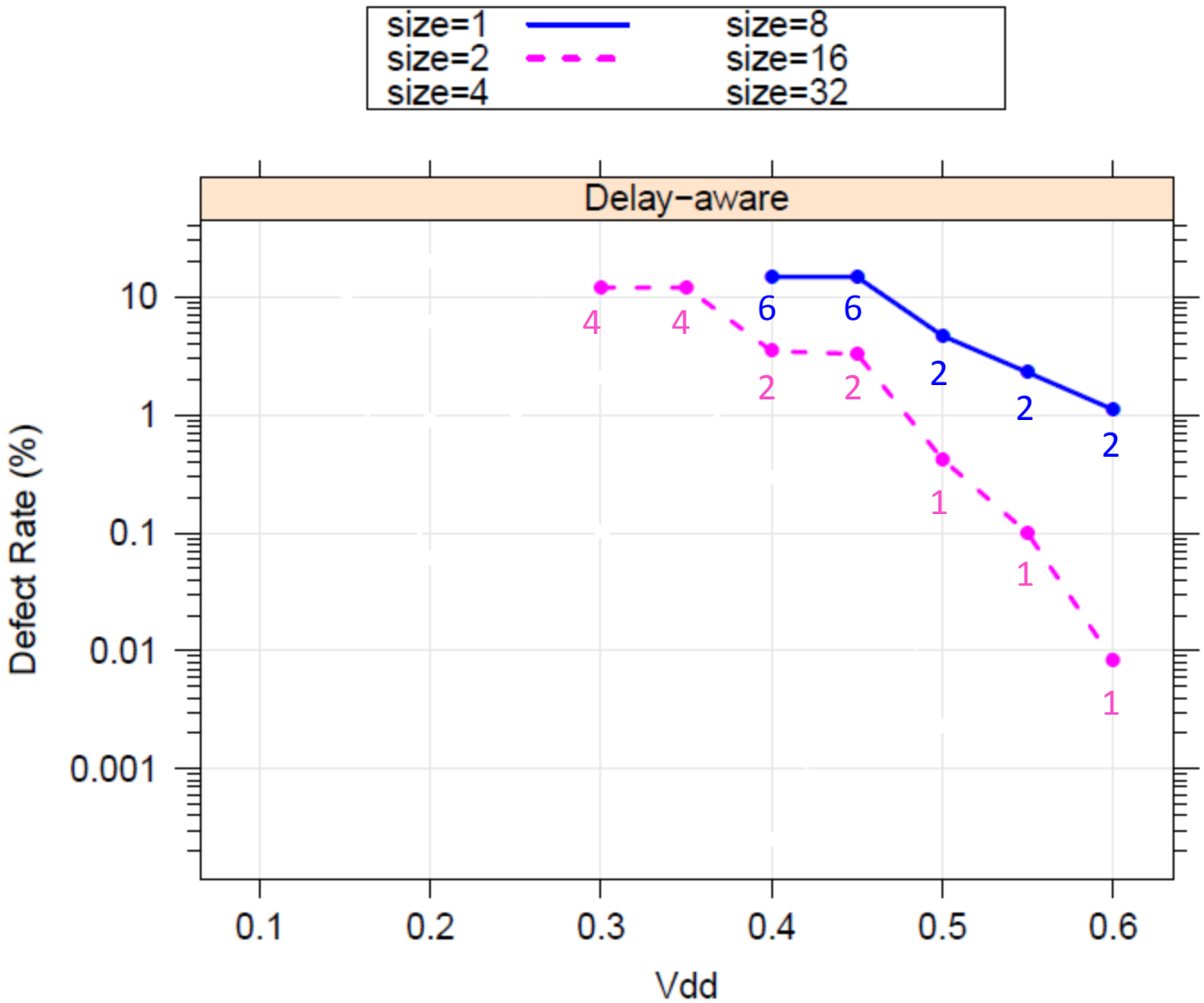


Lower V_{dd}
- More defects
- $E_{dynamic} / V_{dd}^2$

Tolerate defects
- Larger switches
- Spare IOs

Defect Rates





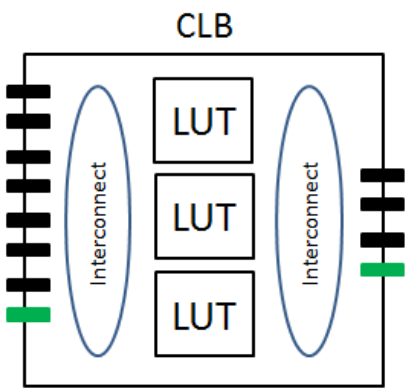
Lower V_{dd}

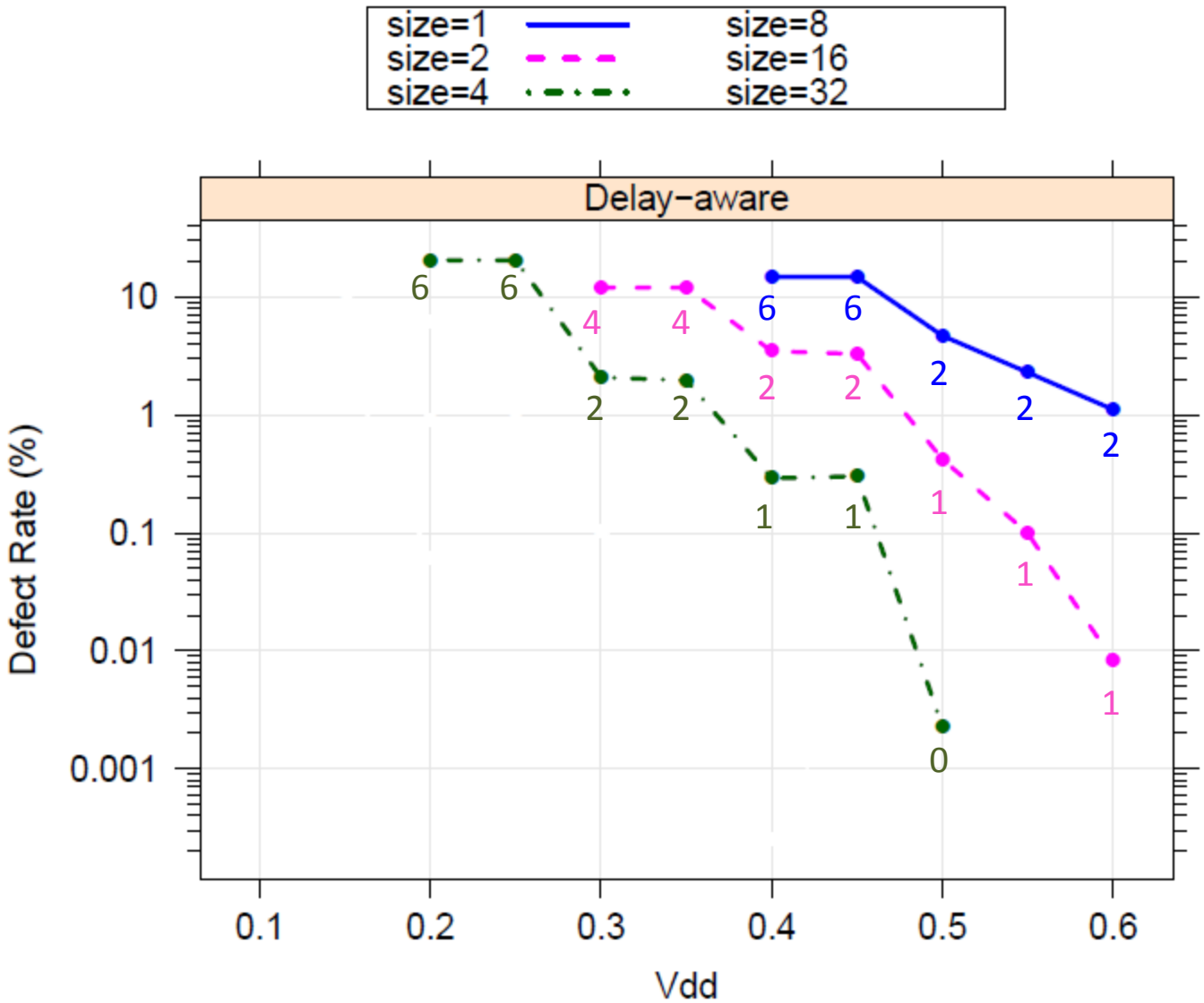
- More defects
- $E_{dynamic} / V_{dd}^2$

Tolerate defects

- Larger switches
- Spare IOs

Defect Rates

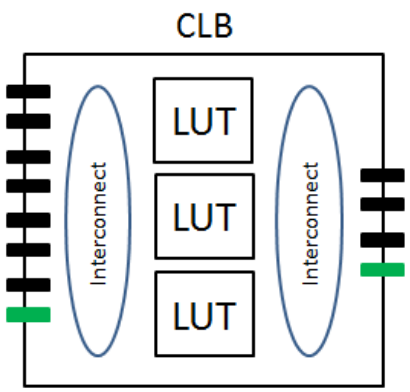


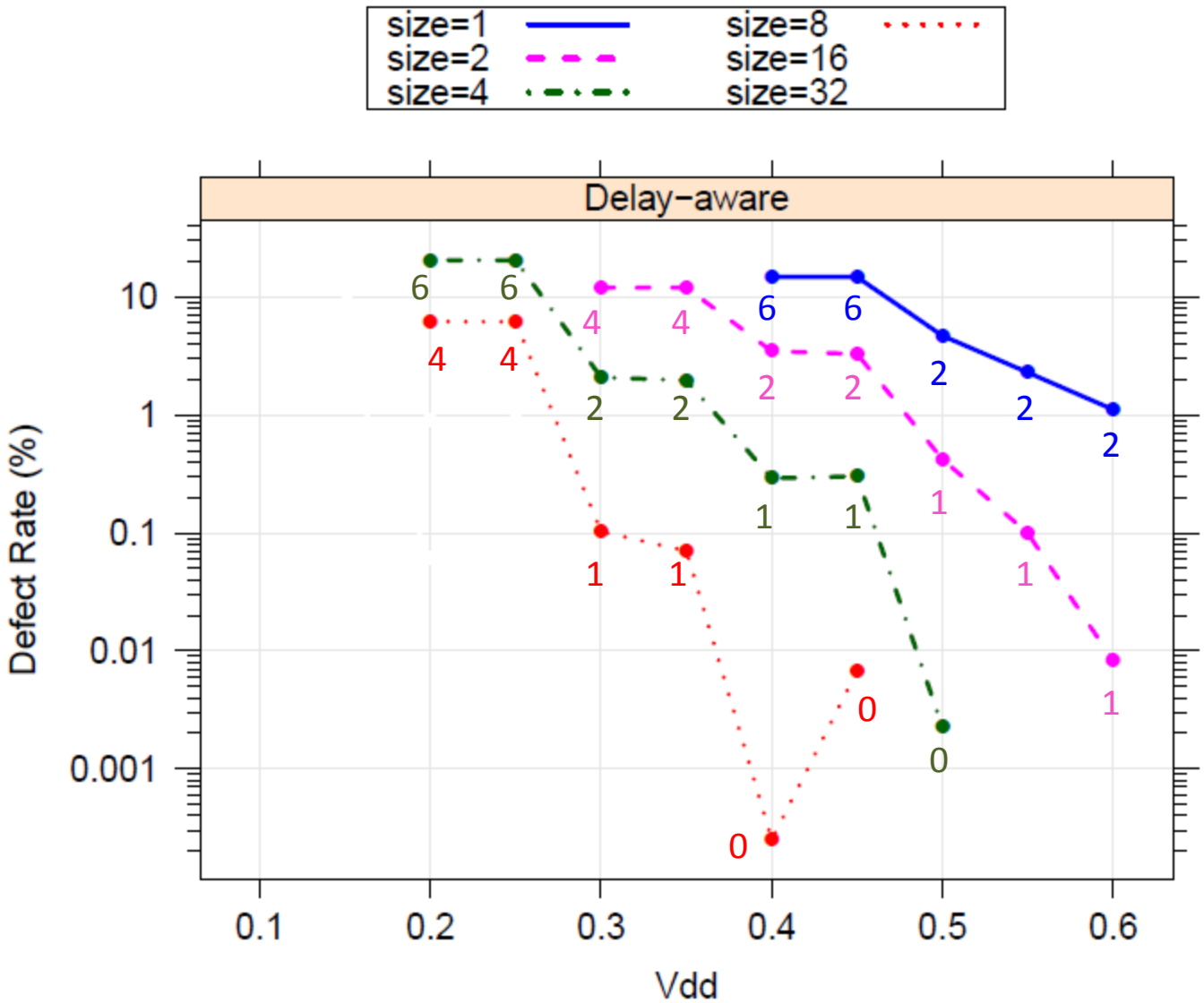


Lower V_{dd}
- More defects
- $E_{dynamic} / V_{dd}^2$

Tolerate defects
- Larger switches
- Spare IOs

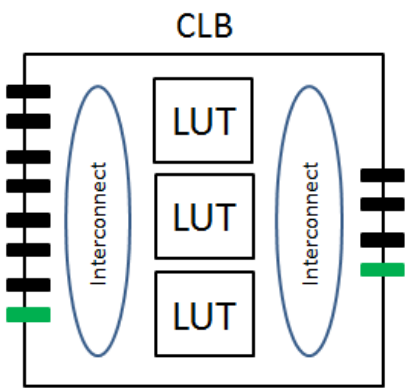
Defect Rates

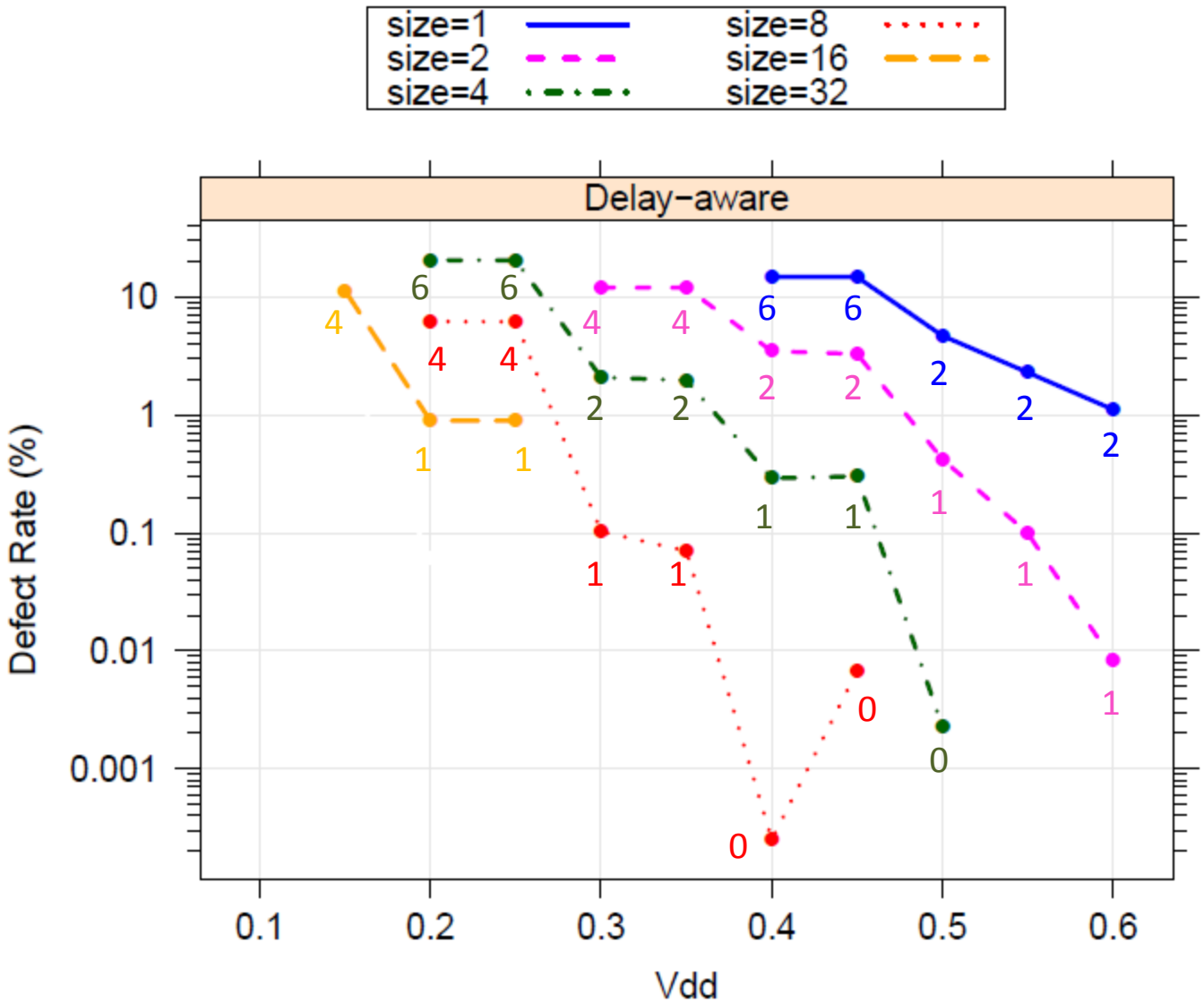




- Lower V_{dd}
  - More defects
  - $E_{dynamic} / V_{dd}^2$
- Tolerate defects
  - Larger switches
  - Spare IOs

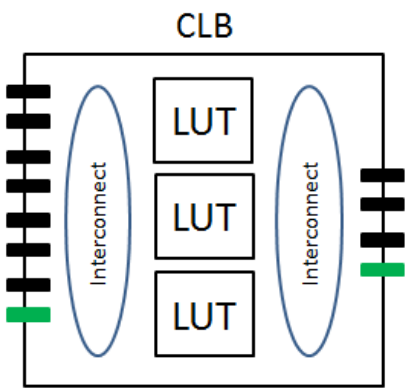
Defect Rates

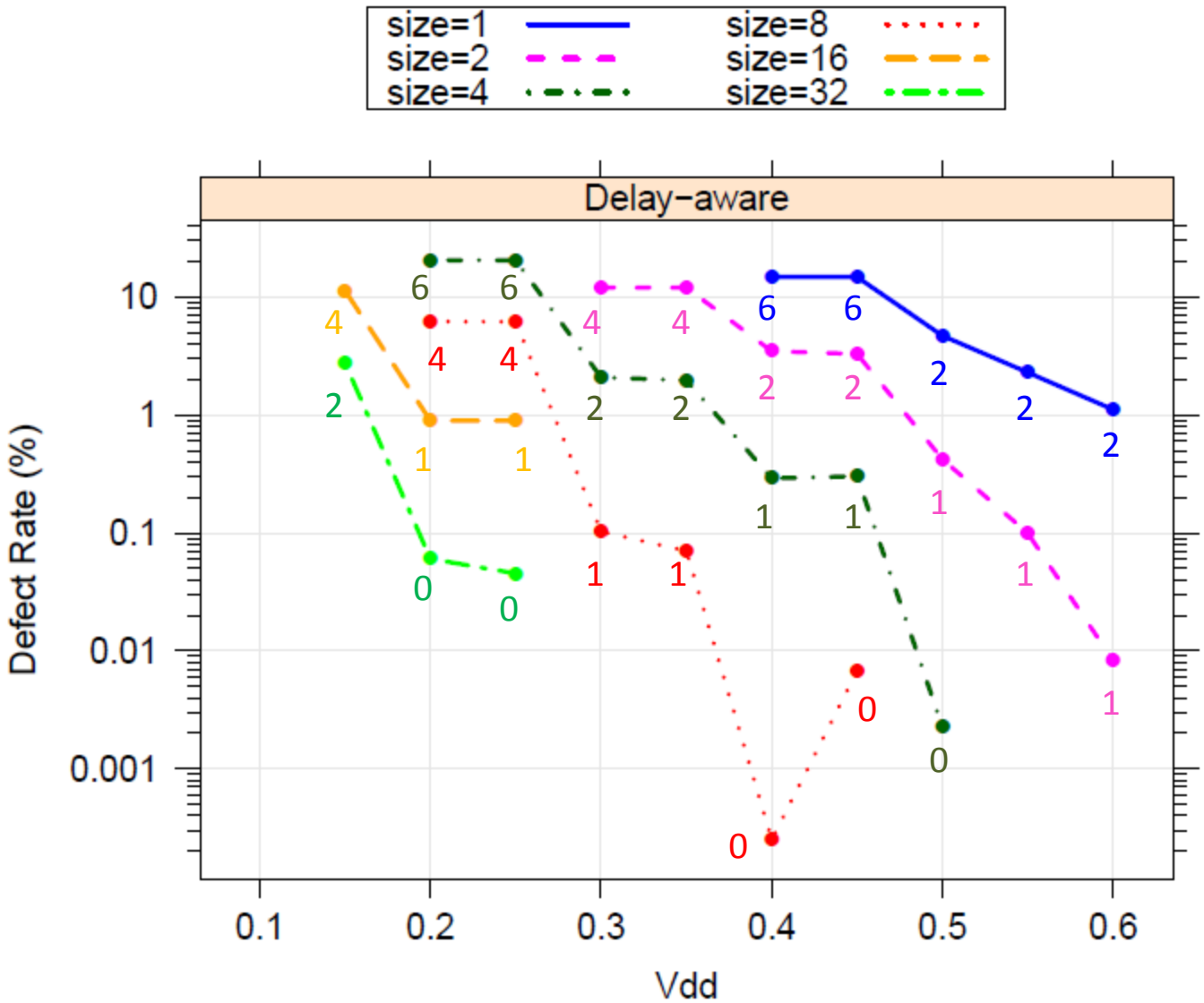




- Lower V_{dd}
  - More defects
  - $E_{dynamic} / V_{dd}^2$
- Tolerate defects
  - Larger switches
  - Spare IOs

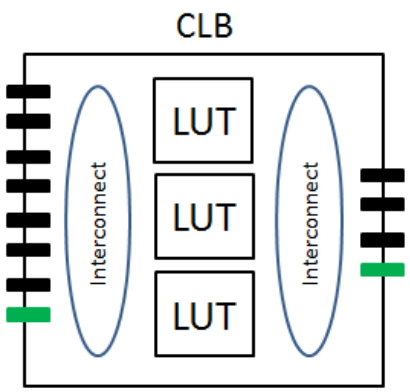
Defect Rates

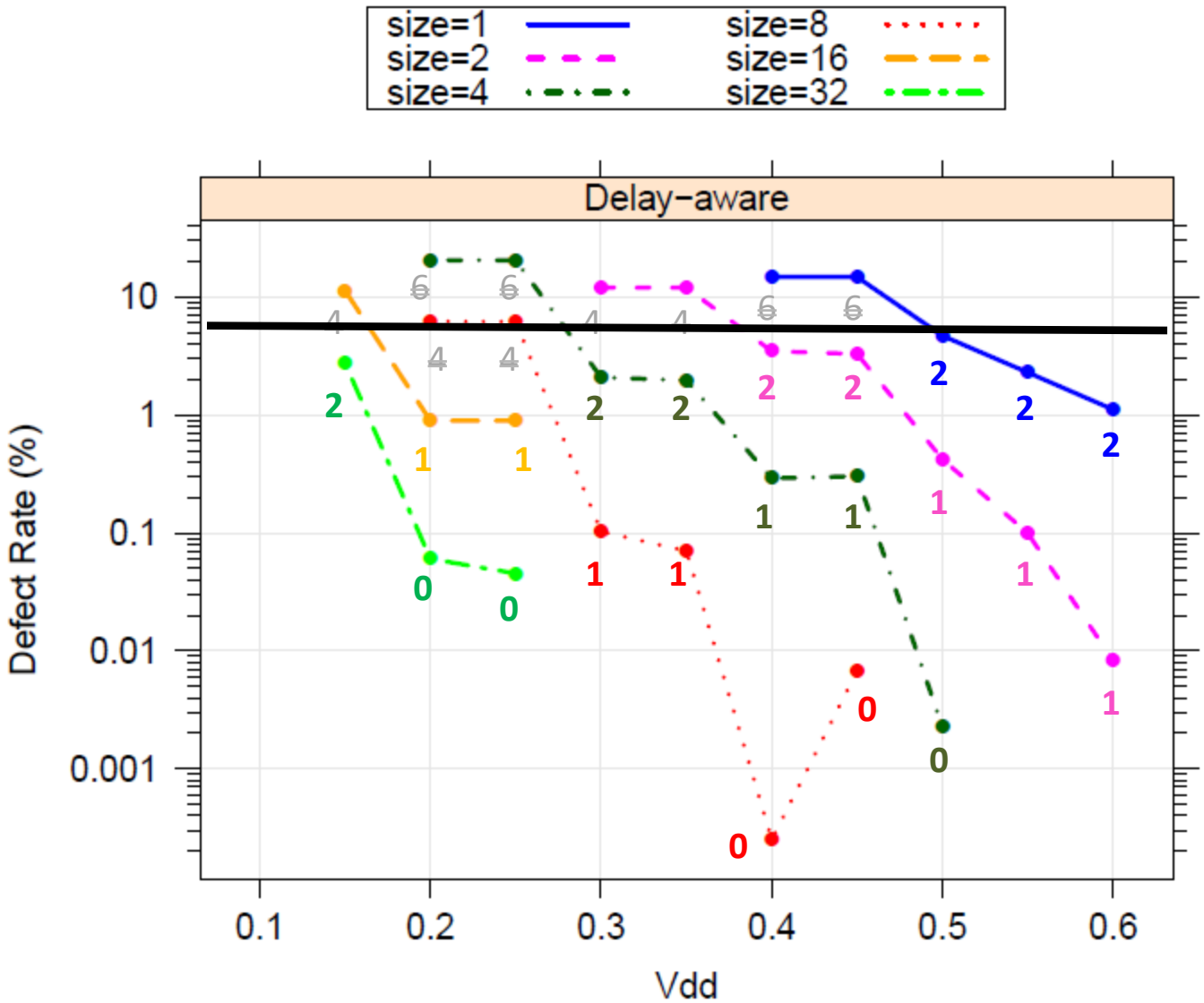




- Lower V_{dd}
  - More defects
  - $E_{dynamic} / V_{dd}^2$
- Tolerate defects
  - Larger switches
  - Spare IOs

Defect Rates

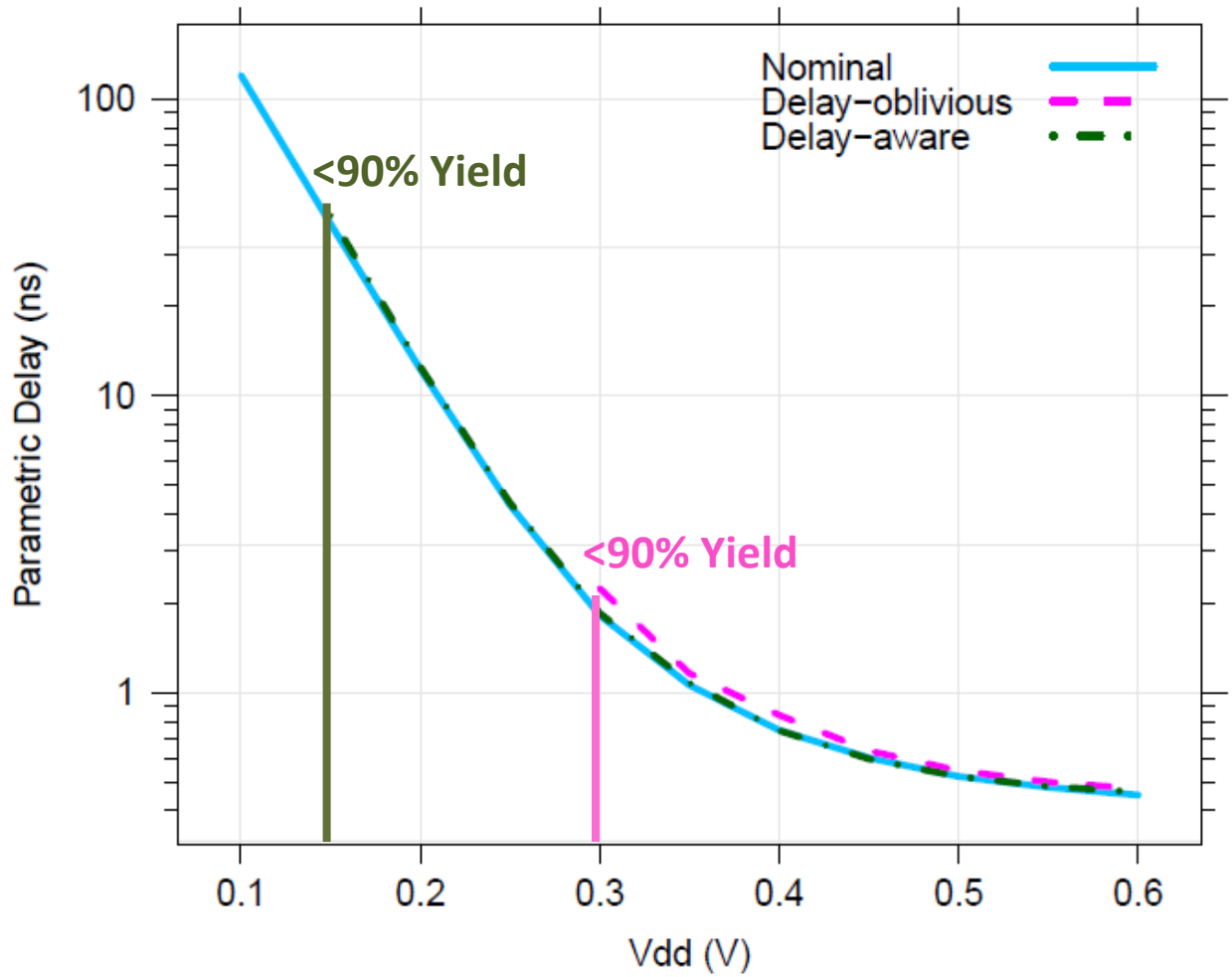




- Lower V_{dd}
  - More defects
  - $E_{dynamic} / V_{dd}^2$
- Tolerate defects
  - Larger switches
  - Spare IOs

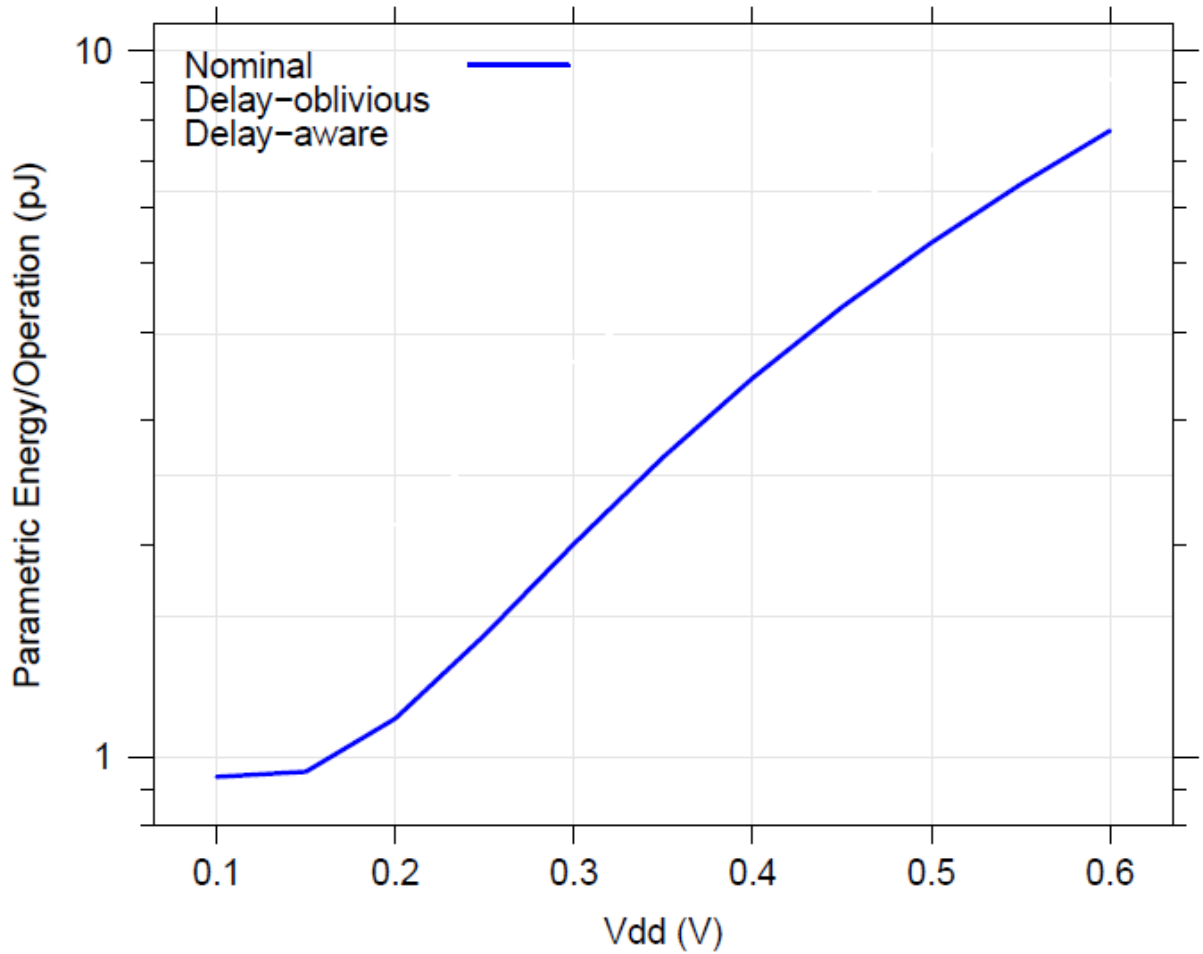
Delay vs V_{dd} (des, optimized sizing)

- Delay-aware routing eliminates delay margins



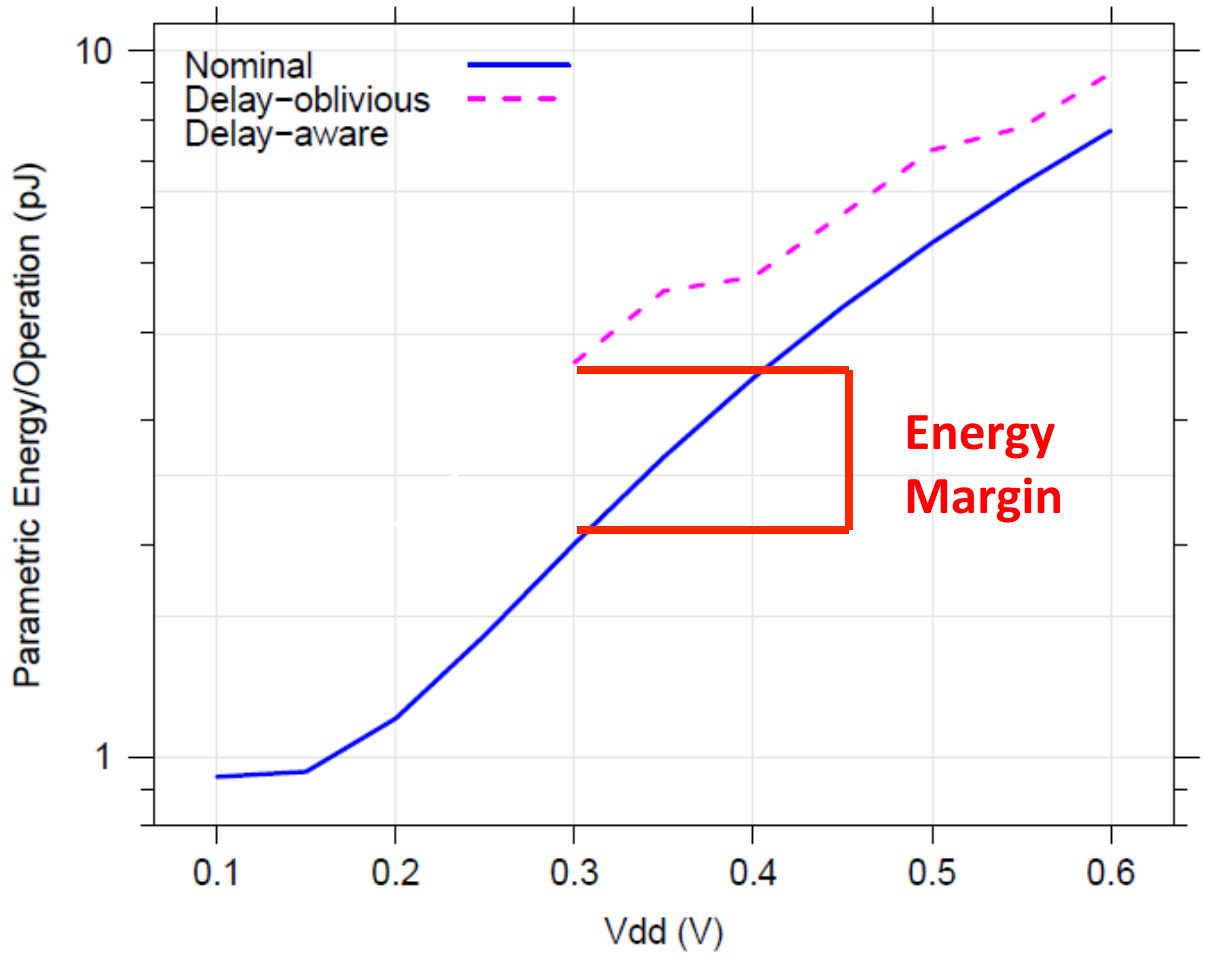
Energy vs V_{dd} (des, optimized sizing)

- Nominal uses minimum size



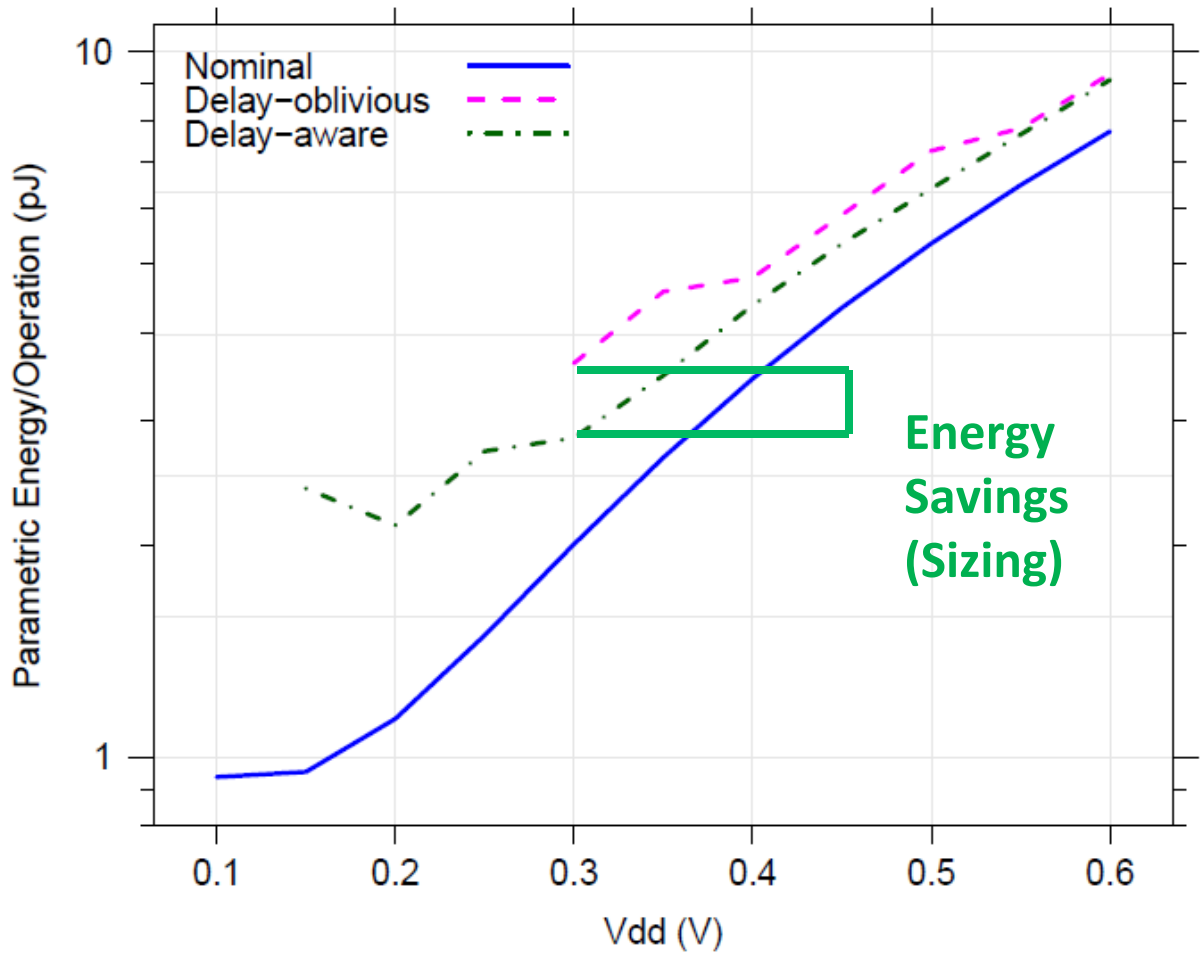
Energy vs V_{dd} (des, optimized sizing)

- Nominal uses minimum size



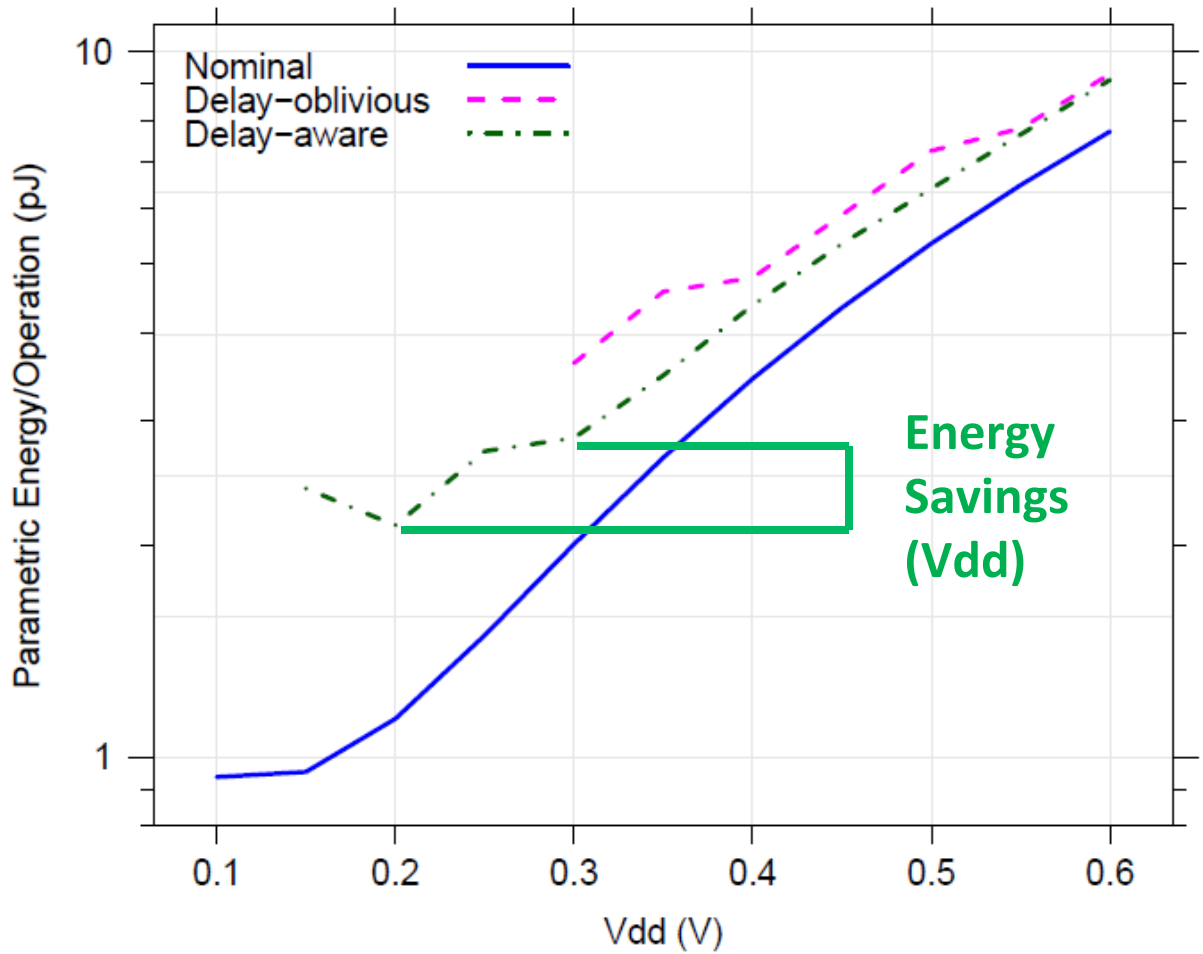
Energy vs V_{dd} (des, optimized sizing)

- Nominal uses minimum size
- Delay-aware routing reduces energy margins
  1. Smaller sizes



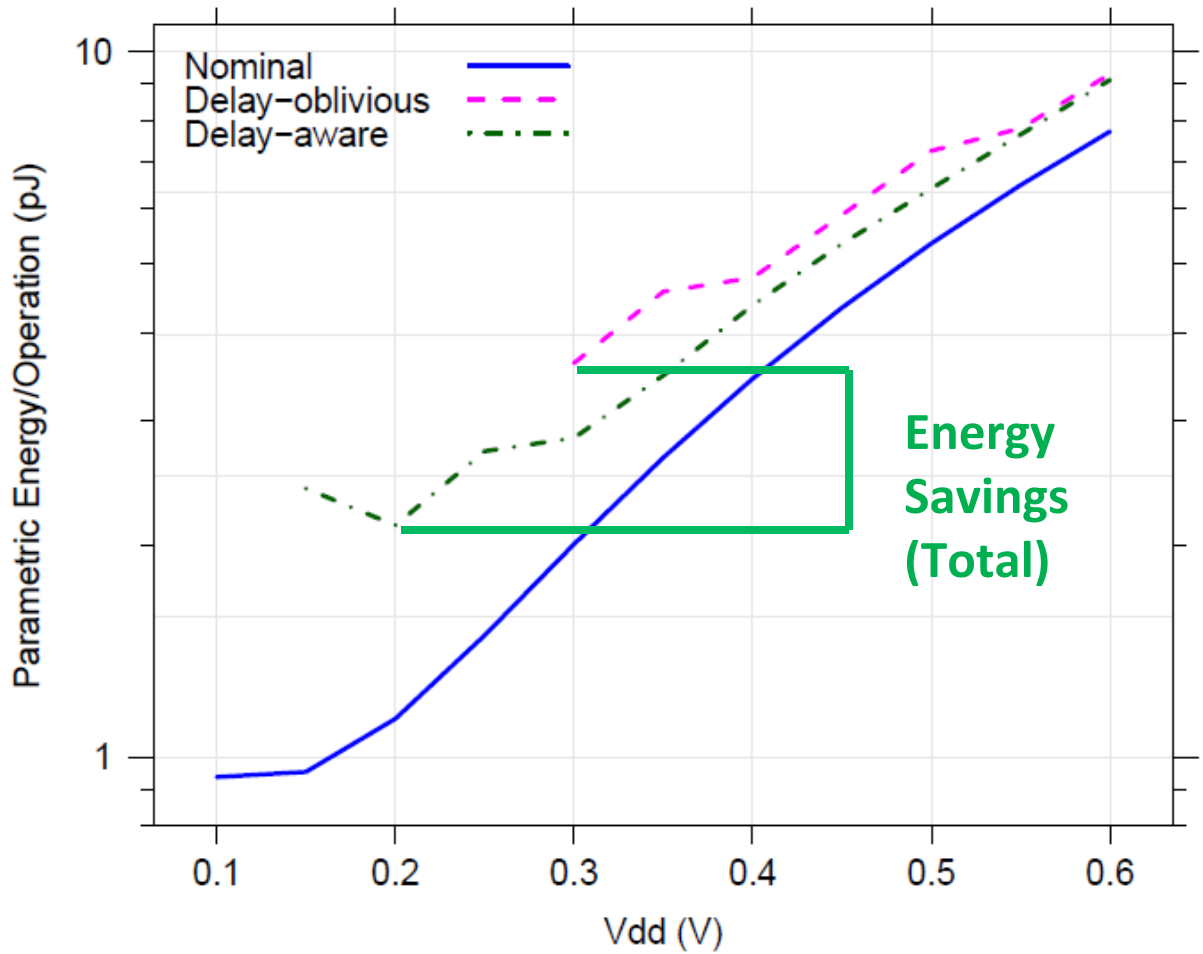
Energy vs V_{dd} (des, optimized sizing)

- Nominal uses minimum size
- Delay-aware routing reduces energy margins
  1. Smaller sizes
  2. Lower voltages



Energy vs V_{dd} (des, optimized sizing)

- Nominal uses minimum size
- Delay-aware routing reduces energy margins
  1. Smaller sizes
  2. Lower voltages
  3. Less Leakage



Results & Future Work

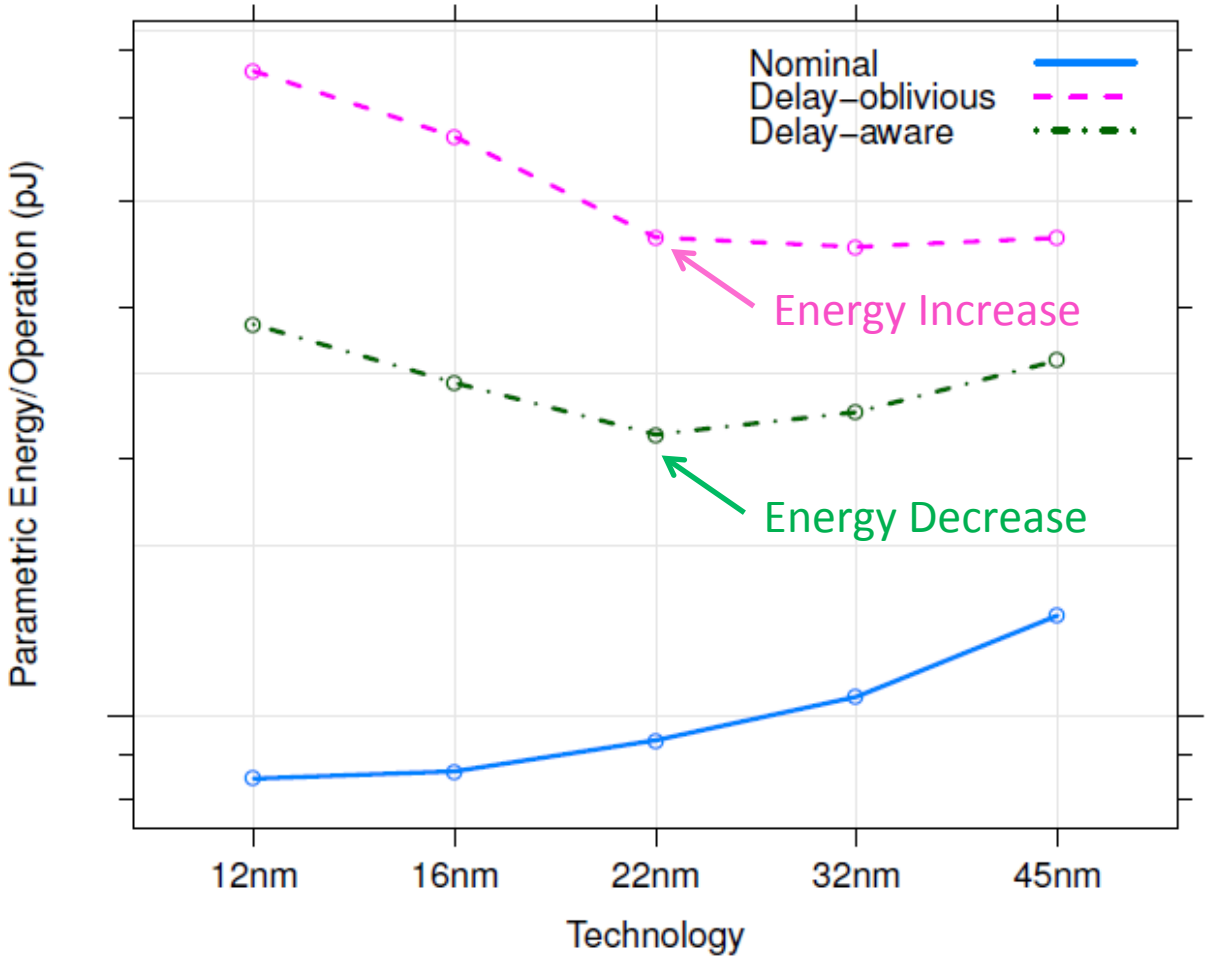
Design	LUT	C_{min}	Technology (nm)				
			45	32	22	16	12
alu4	1492	52	1.69	1.81	1.87	2.02	2.02
apex2	1876	76	1.56	1.73	1.82	1.96	2.04
apex4	1304	78	1.29	1.46	1.57	1.87	1.90
bigkey	1816	72	1.49	1.72	1.88	2.58	2.69
clma	7808	92	1.42	1.61	1.73	2.06	2.09
des	1504	78	1.39	1.55	1.27	1.93	1.97
diffeq	1280	76	1.30	1.41	1.65	2.09	2.12
dsip	1372	64	1.58	1.82	1.90	2.01	2.05
elliptic	2784	60	1.88	2.08	2.19	2.35	2.45
ex1010	4744	114	1.17	1.27	1.48	1.88	1.91
ex5p	1092	70	1.37	1.53	1.66	1.87	1.96
frisc	2892	82	1.74	2.00	2.09	2.21	2.25
misex3	1388	68	1.63	1.78	1.81	1.78	1.76
pdc	4616	96	1.46	1.60	1.63	1.60	1.59
s298	2020	60	1.31	1.47	1.62	1.84	1.79
s38417	6232	50	1.29	1.46	1.63	2.09	2.09
s38584.1	6064	60	1.35	1.61	1.68	1.91	1.90
seq	1724	78	1.62	1.74	1.85	1.98	2.00
spla	3784	86	1.04	1.18	1.30	1.47	1.43
tseng	972	48	1.39	1.57	1.73	2.28	2.48
Geomean (benefit)			1.42	1.60	1.69	1.98	1.98
Geomean (margins[1])			2.12	3.04	4.01	5.04	6.12

[1] margin = delay-oblivious/nominal

- 1.42-1.98x energy reduction across technologies
- Improvement?
  - Power gating
  - Selective sizing
  - Pipelining

Minimum Energy vs Technology

- Delay-oblivious scales to 32nm
- Delay-aware scales to 22nm



Conclusions

- Component-specific routing challenging but beneficial
- Is it worth the trouble?
  1. **Eliminate delay margins**
  2. **Reduce energy by 1.42-1.98x**
  3. **Extend *minimum energy* scaling by a technology generation**