Improved On-Chip Analytical Power and Area Modeling

Andrew B. Kahng
Bill Lin
Kambiz Samadi
(http://vlsicad.ucsd.edu)

University of California, San Diego

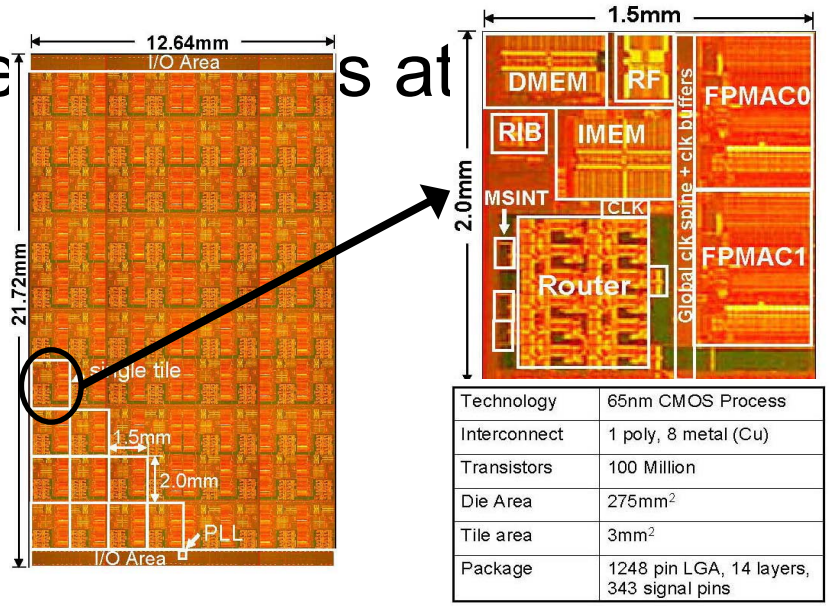
January 20, 2010

Outline

- Motivation
- Implementation Flow and Scope of Study
- Modeling Methodology
  - Modeling Problem
  - Power Modeling
  - Area Modeling
- Experimental Results and Discussion
- Conclusions

Motivation

- NoCs needed to interconnect many-core chips
- Performance was the primary concern
- Power efficiency is now critical
  - 28% of total power in Intel 80-core Teraflops chip is due to interconnection networks (routers + links);
  - → Need rapid power estimation to trade off alternative architectures
- Rapid power-are s at ctural level



Technology	65nm CMOS Process
Interconnect	1 poly, 8 metal (Cu)
Transistors	100 Million
Die Area	275mm^2
Tile area	3mm^2
Package	1248 pin LGA, 14 layers, 343 signal pins

Related Work

- Real-chip power measurements (Isci et al. ’03)
- RTL-level NoC power estimations (A. Banerjee et al. ’07 and N. Banerjee et al. ’04)
  - Simulation time is slow
  - Requires detailed RTL
- Architectural-level power estimation
  - Interconnection network (Patel et al. ’97); model is not instantiated with architectural parameters → not suitable to explore tradeoffs in router microarchitecture
  - Uniprocessor power modeling (Wattch: Brooks et al. ’00 and SimplePower: Ye et al. ’00)
- ORION models
  - Recently enhanced (i.e., ORION 2.0)
  - Early-stage design space exploration

Gap #1: Models Tied to µArchitecture / Implementation

- Developed from a mix of template architectures / circuit implementations (cf. ORION 2.0)
  - Not accurate within an architecture-specific CAD flow
  - Useful for early-stage estimations (e.g., complementary to our approach)
- Power and area estimations via parametric regression (Meloni et al. '07)
  - Regression process assumes certain functional forms → depends on the underlying architecture / circuit implementation
  - Does not consider implementation parameters (e.g., aspect ratio, etc.)

Reduced accuracy → not suitable for efficient design space exploration

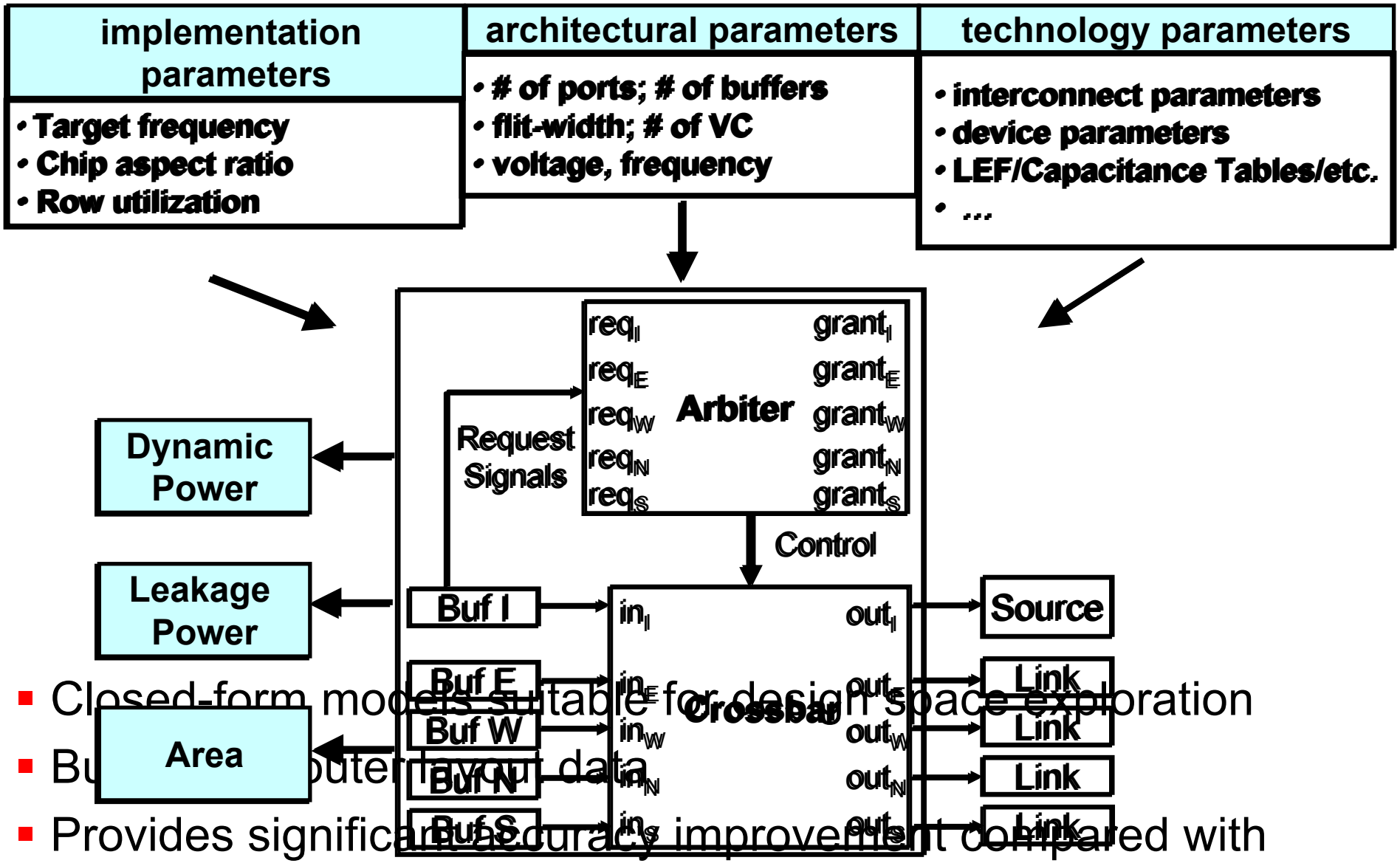
Gap #2: Models Overlook μArchitecture Details

- Parametric cycle-accurate traffic driven power models, without consideration of microarchitectural parameters (cf. NOCEE)
- Power model with limited dependency on microarchitectural parameters; derived from synthesis results

Reduced applicability to energy design space explorations

- **Goal:** Develop a modeling framework that: (1) is architecture-independent, (2) considers all the relevant microarchitectural details

Improved NoC Router Power-Area Models

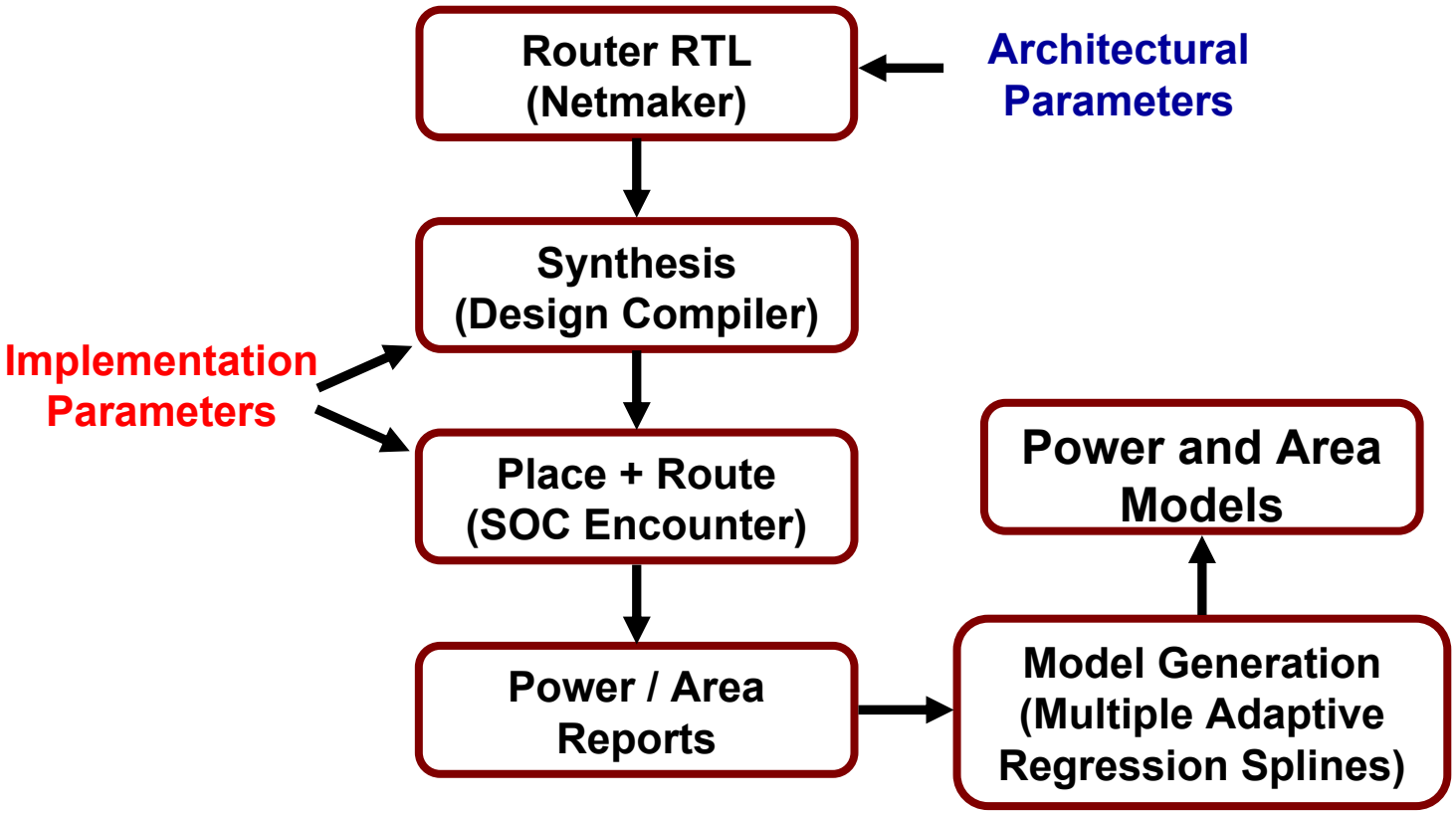


- Closed-form models suitable for design space exploration
- Built-in router layout data
- Provides significant accuracy improvement compared with existing models (e.g., ORION 2.0)

Outline

- ✓Motivation
- Implementation Flow and Scope of Study
- Modeling Methodology
  - Modeling Problem
  - Power Modeling
  - Area Modeling
- Experimental Results and Discussion
- Conclusions

Implementation Flow and Tools



- RTL generation from architecture
- Timing-driven synthesis, place and route flow
- Use range of **architectural** and **implementation** parameters to capture design space
- Nonparametric regression modeling

Scope of Study

- *Netmaker (Cambridge)* → fully synthesizable router RTL codes
- Libraries: TSMC (1) 130G, (2) 90G, and (3) 65GP
- Tool Chain: Synopsys Design Compiler (DC), Cadence SOC Encounter (SOCE), Salford MARS 3.0

- Experimental axes:
  - Technology nodes: {130nm, 90nm, 65nm}
  - Implementation parameters:
    - f_{clk} = target clock frequency
    - *ar* = aspect ratio
    - *util* = row utilization
  - Architectural parameters:
    - *fw* = flit-width
    - n_{vc} = number of virtual channels
    - n_{port} = number of input/output ports
    - l_{buf} = buffer length (#flit buffers / VC)

Outline

- ✓Motivation
- ✓Implementation Flow and Scope of Study
- Modeling Methodology
  - Modeling Problem
  - Power Modeling
  - Area Modeling
- Experimental Results and Discussion
- Conclusions

Modeling Problem

- Problem: Accurately predict *y* given vector of parameters $\vec{x}$
- Difficulties: (1) which variables *x* to use, and (2) how different variables combine to generate *y*

$$y = f(\vec{x}) + noise$$

- Parametric regression: requires a functional form
- Nonparametric regression: learns about the best model from the data itself

  → For our purpose, allows decoupling of underlying architecture / implementation from modeling effort

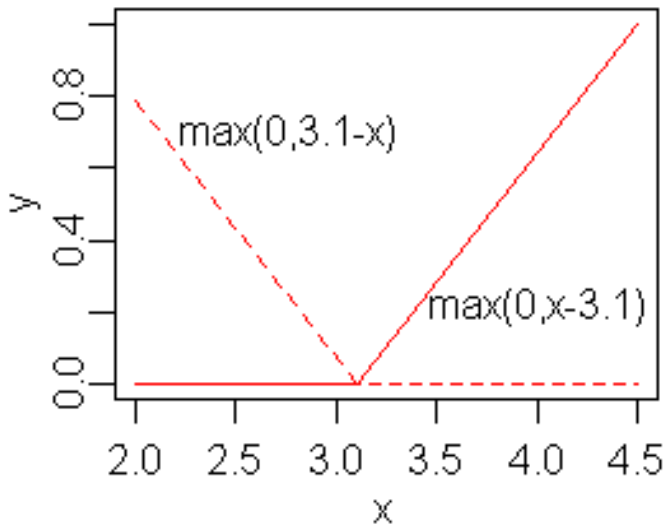
- Our approach: Use nonparametric regression to model power and area of an on-chip router

Multivariate Adaptive Regression Splines (MARS)

- MARS is a nonparametric regression technique
- MARS builds models of form:

$$\hat{f}(\vec{x}) = c_0 + \sum_{i=1}^{k} c_i B_i(\vec{x})$$

- Each basis function $B_i(x)$ can be:
  - a constant
  - a "hinge" function max(0, c-x) or max(0, x-c)
  - a product of two or more hinge functions



- Two modeling steps:
  - (1) forward pass: obtains model with defined maximum number of terms
  - (2) backward pass: improves generality by avoiding an overfit model

Power and Area Modeling

- We model power dependence on microarchitecture and implementation parameters
  - $P_{dynamic} = 0.5 \times \alpha \times c_{switching} \times V^2 \times f_{clk}$
  - $P_{leakage} = i_{leak} \times V$
- Our modeling task:
  - Model dependence of $(P_{dynamic} / \alpha \times V^2 \times f_{clk})$ on microarchitectural and implementation parameters
  - Model dependence of $(P_{leakage} / V)$ on microarchitectural and implementation parameters
- Similarly, we model area dependence on microarchitecture and implementation parameters
  - Area is the sum of standard cell area

Example MARS Output Models (1)

Dynamic power model of a router in 65nm technology

$B_1 = \max(0, n_{port} - 5)$; $B_2 = \max(0, 5 - n_{port})$; …
$B_{34} = \max(0, f_{clk} - 200) \times B_1$; $B_{35} = \max(0, 200 - f_{clk})\ B_1$;

$$\mathbf{P_{dynamic} = 0.5 \times \alpha \times (0.83 + 0.64 \times B_1 - 0.31 \times B_2 + 0.16 \times B_3 \ldots - 0.003 \times B_{33} + 0.003 \times B_{34} - 0.003 \times B_{35}) \times V^2}$$

Leakage power model of a router in 65nm technology

$B_1 = \max(0, n_{port} - 5)$; $B_2 = \max(0, 5 - n_{port})$; …
$B_{34} = \max(0, n_{vc} - 3) \times B_{27}$; $B_{35} = \max(0, 3 - n_{vc}) \times B_{27}$;

$$\mathbf{P_{leakage} = (0.13 + 0.04 \times B_1 - 0.04 \times B_2 + 0.01 \times B_3 \ldots - 6.59E\text{-}5 \times B_{34} - 5.53E\text{-}5 \times B_{35}) \times V}$$

Example MARS Output Models (2)

Area model of a router in 65nm technology

$B_1 = \max(0, n_{port} - 5)$; $B_2 = \max(0, 5 - n_{port})$; …
$B_{34} = \max(0, 24 - fw) \times B_{14}$; $B_{35} = \max(0, f_{clk} - 100) \times B_{15}$;

$Area = 0.02 + 0.01 \times B_1 - 0.004 \times B_2 + 0.003 \times B_3$ … $- 4.59E\text{-}6 \times B_{34} - 1.23E\text{-}7 \times B_{35}$

Total wirelength model of a router in 65nm technology (NEW)

$B_1 = \max(0, n_{port} - 5)$; $B_2 = \max(0, 5 - n_{port})$; …
$B_{33} = \max(0, 1 - ar) \times B_{26}$; $B_{34} = \max(0, util - 0.7) \times B_8$;

$WL_{total} = 112269 + 64952.4 \times B_1 - 31881.3 \times B_2$ … $+ 157.639 \times B_{33} - 321.06 \times B_{34}$

- Closed-form expressions with respect to architectural and implementation parameters
- Suitable to drive early-stage architecture-level design exploration

Outline

- ✓ Motivation
- ✓ Implementation Flow and Scope of Study
- ✓ Modeling Methodology
  - ✓ Modeling Problem
  - ✓ Power Modeling
  - ✓ Area Modeling
- Experimental Results and Discussion
- Conclusions

Model Validation

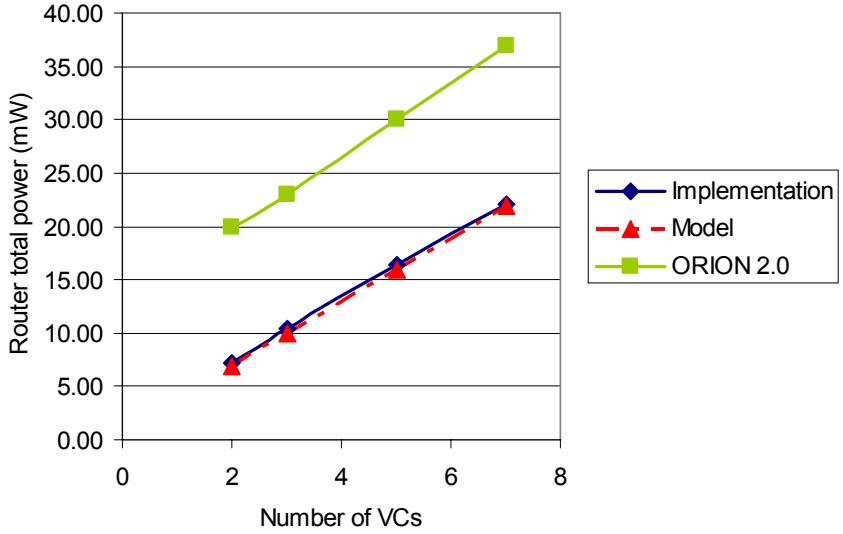
- We validate our models against layout data
- We compare our models against
  (1) parametric regression models and
  (2) ORION 2.0
- We show the importance of layout data in model generation → increased accuracy
- We show the sensitivity and stability of our models

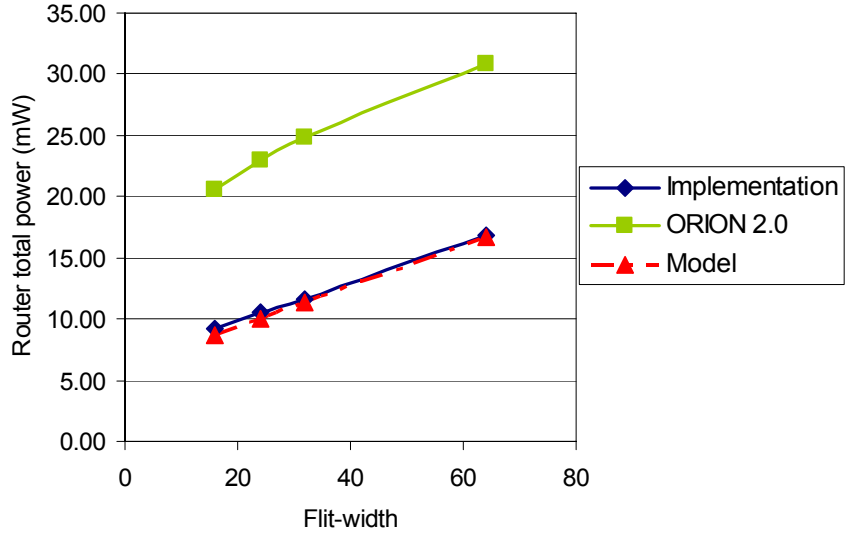
Comparison Models

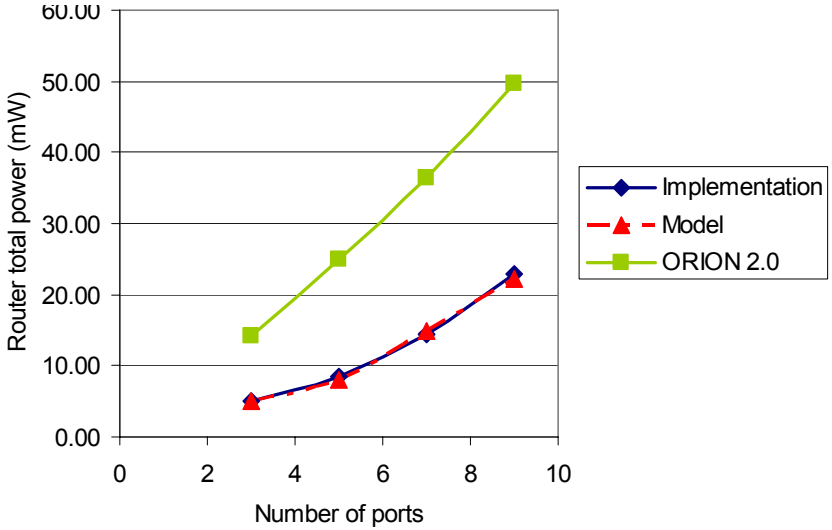
- Parametric Regression (PReg):
  - We assume baseline virtual channel (VC) with:
    - FIFO buffers implemented as flip-flop registers
      - → $c_{switching} \sim O(l_{buf} \times fw \times n_{port})$; $i_{leak} \sim O(l_{buf} \times fw \times n_{port} \times n_{vc})$
    - Multiplexer tree crossbar
      - → $c_{switching} \sim O(n^2_{port} \times fw)$; $i_{leak} \sim O(n^2_{port} \times fw)$
    - VC “selection” arbitration (cf. Kumar et al. ’07)
      - → $c_{switching} \sim O(n^2_{port})$; $i_{leak} \sim O(n^2_{port} \times n_{vc})$
  - → Requires modeler to have knowledge about the underlying architecture / circuit implementation
- ORION 2.0

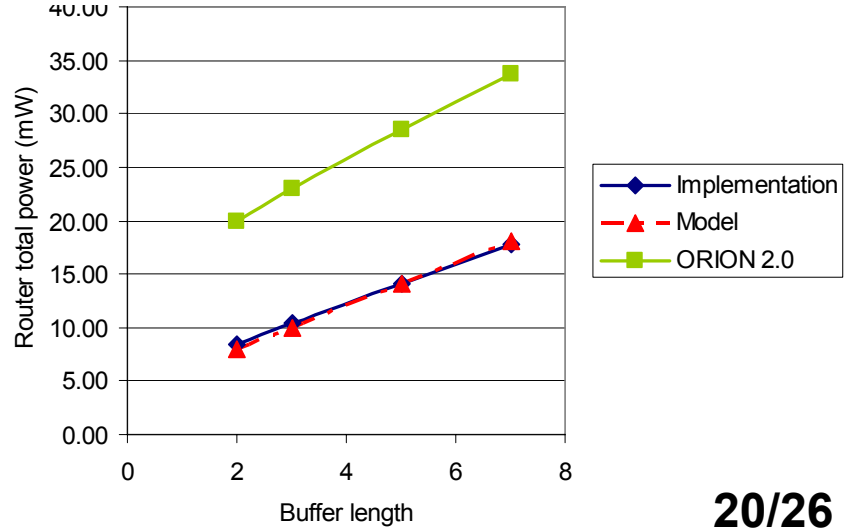
Comparison vs. ORION 2.0

- Comparison against ORION 2.0 w.r.t. microarchitectural parameters:
  - (1) #VC (n_{vc}), (2) flit-width (fw), (3) #port (n_{port}), and (4) buffer length (l_{buf})









Comparison vs. PReg. and ORION 2.0

Metric		Power Model			Area Model		
		New	PReg	ORION2.0	**New**	PReg	ORION2.0
min % error	130nm	**0.011**	7.659	9.526	**0.001**	29.88	10.121
	90nm	**0.008**	7.236	6.865	**0.002**	27.82	8.229
	65nm	**0.007**	6.921	7.73	**0.001**	29.12	9.111
max % error	130nm	**62.05**	96.51	103.2	**60.72**	107.8	104.118
	90nm	**60.04**	62.31	85.35	**60.15**	109.2	88.331
	65nm	**59.41**	108.4	81.81	**61.84**	111.3	86.228
avg % error	130nm	**6.012**	23.46	41.33	**5.961**	26.33	38.117
	90nm	**5.654**	25.11	30.22	**5.019**	27.11	32.566
	65nm	**5.817**	24.43	32.78	**5.411**	26.23	33.298

- Power estimation error reductions
  - PReg: avg error 76.2% (24.4% → 5.8%), max error 45.2% (108.4% → 59.4%)
  - ORION 2.0: avg error 82.3% (32.8% → 5.8%), max error 27.4% (81.8% → 59.4%)
- Area estimation error reductions
  - PReg: avg error 79.4% (26.2% → %5.4), max error 45.5% (111.3% → 61.8%)
  - ORION 2.0: avg error 83.8% (33.3% → 5.4%), max error 28.3% (86.2% → 61.8%)

Variable Importance

- We use MARS to identify relative variable importance
- Dominant parameter post-synthesis: n_{vc}
- Dominant parameter post-layout: n_{port}
  - Shows impact of missing layout information at post-synthesis stage
- Example: multiplexer crossbar power is due to (1) multiplexers and (2) interconnection grid between input / output ports
  - Post-synthesis model is oblivious to (2)

Parameter	Variable Importance (%)	
	Post-Synthesis	Post-Layout
n_{port}	92.98	100
n_{vc}	100	95.44
l_{buf}	88.41	73.99
fw	67.03	64.81

Model Sensitivity and Stability

- **Sensitivity** to size of training data
  - (1) s_{tr} = 1/2, (2) s_{tr} = 1/3, (3) s_{tr} = 1/5, (4) s_{tr} = 1/10, and (5) s_{tr} = 64
  - For (1)-(4) we train models using a fraction s_{tr} of the available data points, and validate them on the rest of the data points
  - For (5) we use 64 (out of 256) data points to train the model, and validate it across all 256 available data points

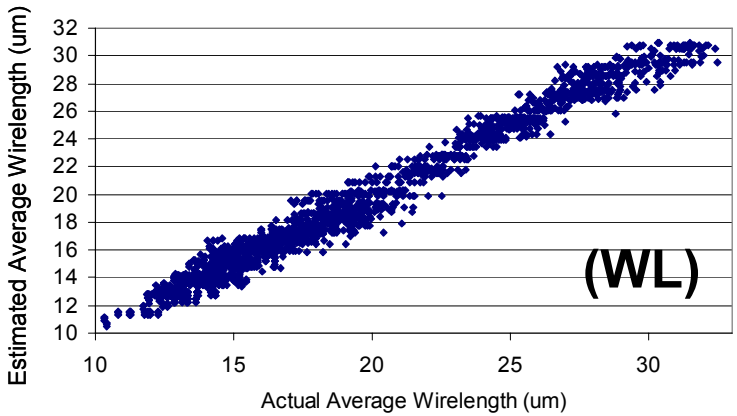
Metric	Power Model				
	s_{tr} = 1/2	s_{tr} = 1/3	s_{tr} = 1/5	s_{tr} = 1/10	s_{tr} = 64
min % error	0.006	0.006	0.007	0.01	0.006
max % error	12.415	49.226	81.11	109.224	77.32
avg % error	1.662	4.012	7.997	27.177	21.23

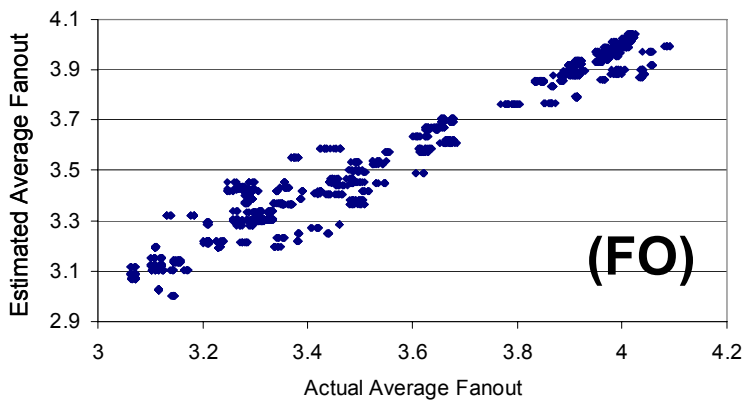
- **Stability** w.r.t. random choice of training data

Metric	Power Model				
	EXP 1	EXP 2	EXP 3	EXP 4	EXP 5
max % error	12.415	13.126	13.911	12.013	11.932
avg % error	1.662	1.412	1.214	1.077	1.103

Extensibility of Approach

- Have used same methodology to develop models for interconnect wirelength (WL) and fanout (FO)
- Wirelength model
  - On average, within 3.4% of layout data
  - 91% reduction of avg error vs. existing models (cf. Christie et al. ’00)
- Fanout model
  - On average, within 0.8% of the layout data
  - 96% reduction of avg error vs. existing models (cf. Zarkesh-Ha et al. ’00)





Outline

- ✓ Motivation
- ✓ Implementation Flow and Scope of Study
- ✓ Modeling Methodology
  - ✓ Modeling Problem
  - ✓ Power Modeling
  - ✓ Area Modeling
- ✓ Experimental Results and Discussion
- Conclusions

Conclusions and Future Directions

- Generally applicable modeling methodology that can leverage architectural parameters and RTL-to-layout implementation
- Achieved accurate power and area models for on-chip router
- Improvement over parametric regression models
  - Power: 76.2% (45.2%) reduction of average (maximum) error
  - Area: 79.4% (44.5%) reduction of average (maximum) error
- Improvement over ORION 2.0
  - Power: 82.3% (27.4%) reduction of average (maximum) error
  - Area: 83.8% (28.3%) reduction of average (maximum) error

- Ongoing work
  - Maximum f_{clk} modeling w.r.t. architectural and implementation parameters
  - Other architectural building blocks (DSP cores, DesignWare library, …)
  - Power, performance and cost estimators for 3-D design space exploration
  - Accurate trace-driven NoC power estimation models