Machine Learning for Next Generation EDA

Paul Franzon

Cirrus Logic Distinguished Professor
Director of Graduate Programs
Department of Electrical and Computer Engineering
NC State University
919.515.7351, paulf@ncsu.edu

Outline

▷ **Introduction**

▷ **Applying Machine Learning to EDA**

  ◆ IP Reuse

  ◆ Physical Design

  ◆ Replacing Design Rules

▷ **Machine Learning Acceleration**

▷ **Conclusions**

Machine Learning

▷ **Learning from Data**
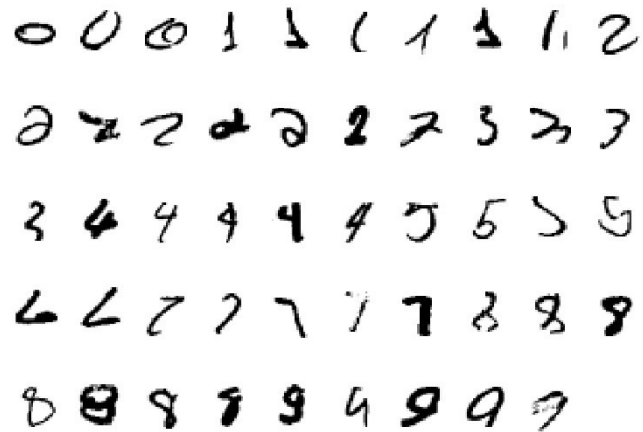
Massive Data

Train or Fit a Model
e.g. Neural Network

Use model to
recognize previously
unseen examples

… Types of Learning

▷ **Off-line learning**

◆ Optimized neural network against a fixed training set using off-line optimization

◆ Usually labeled data

▷ **Incremental Learning**

◆ Modify learning using inline data

◆ Usually labeled data

▷ **On-line (In-line) learning**

◆ Learn entirely using data in the field

◆ Alternate learning and inference cycles

◆ Sometimes unlabeled dta

4

Questions being addressed @ NCSU

▷ **What does machine learning mean to hardware designers?**

A. **A. Building computation engines that specialize in machine learning.**
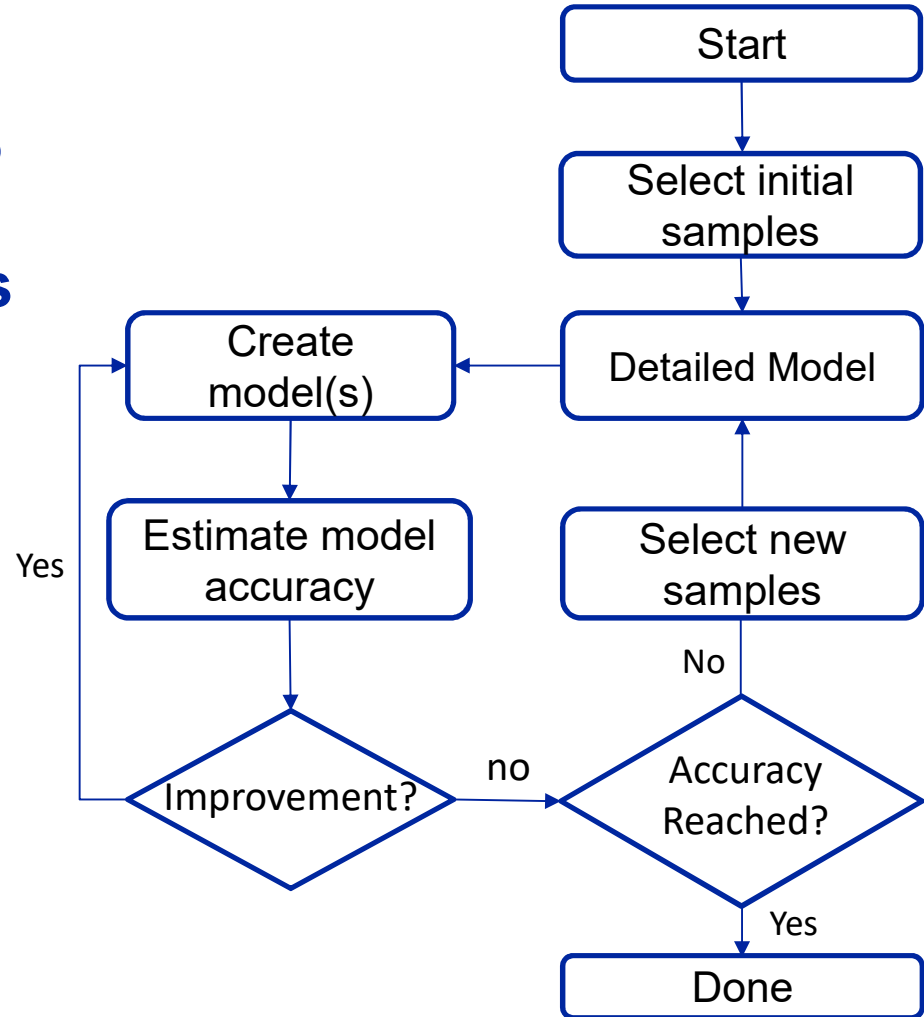
B. **Applying machine learning to EDA.**

Surrogate Modeling

▷ **"Train" a global model that is fast to evaluate from multiple evaluations of a detailed model that is slow to evaluate**

Surrogate Model
↓

$$x \rightarrow f(x) \rightarrow y$$

Start

↓

Select initial samples

↓

Create model(s) ← Detailed Model

↓ ↑

Estimate model accuracy | Select new samples

↓ No ↑

Improvement? — no → Accuracy Reached?

Yes

Yes ↓

Done

Surrogate Modeling

Basic idea: *Accurately* approximating the black-box design with *limited* number of samples.

Objective:

$$\min \epsilon = f(x) - \hat{f}(x)$$

ϵ : Model Error
$f(x)$: Design Response
$\hat{f}(x)$: Model Prediction

Input Parameter Space

A

B

C

D

Simulation /Measurement
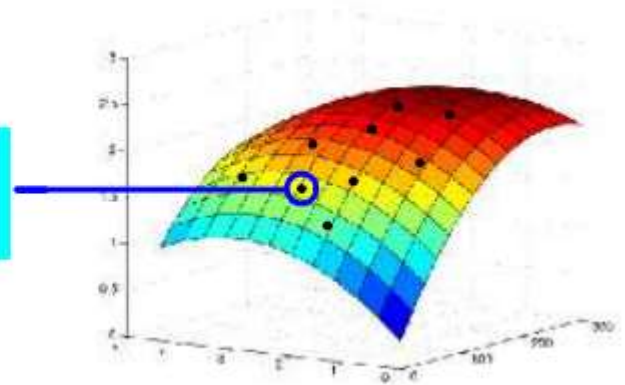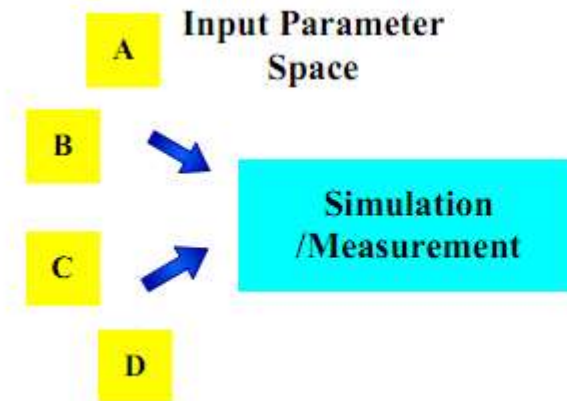
Accuracy: $RRSE = \sqrt{\dfrac{\sum_{i=1}^{n}(y_i - \overline{y_i})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$

Advantages:

- Modeling accuracy and efficiency
- Fast to execute – mathematical expressions vs. systematic simulation
- Various modeling techniques exists for choices: Kriging, Radial-based functions, neural networks, etc.
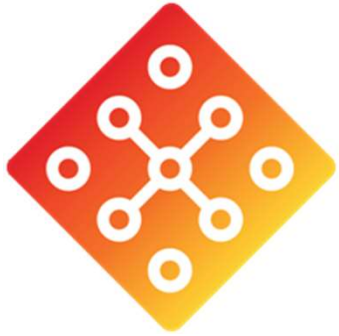
SUMO

Outline

▷ **Introduction**

▷ **Applying Machine Learning to EDA**

  ◆ IP Reuse

  ◆ Physical Design

  ◆ Improving DFM design closure*

  ◆ Accurate modeling for high speed IO*

▷ **Conclusions**

*Wont be presented today

CAEML

CENTER FOR ADVANCED ELECTRONICS
THROUGH MACHINE LEARNING

**Applying Machine Learning to
Electronic Design**

Principal Investigators
Elyse Rosenbaum, Illinois (Center Director)
Paul Franzon, NCSU (Site Director)
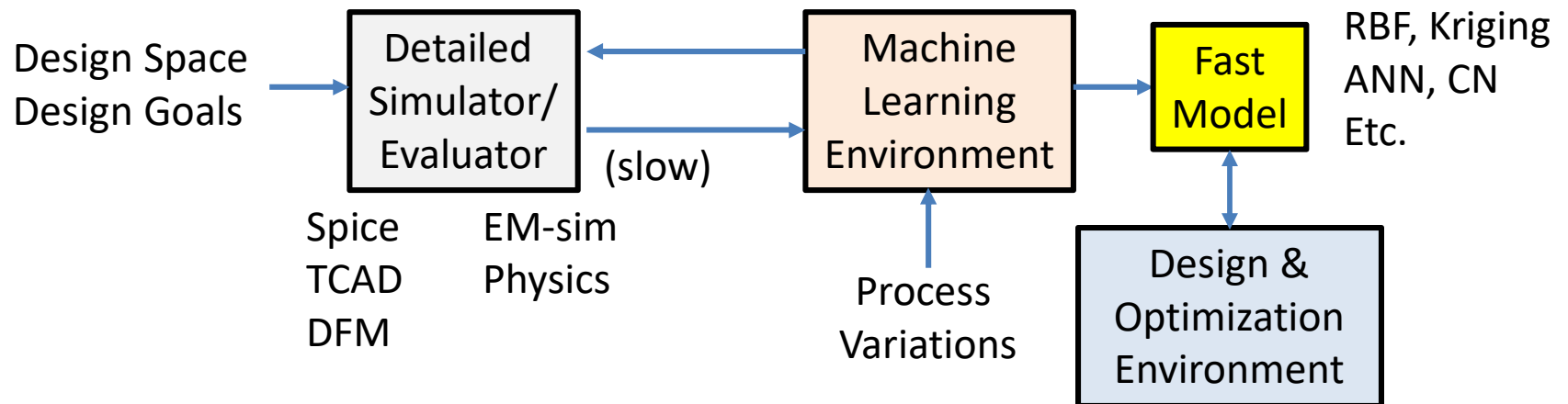Madhavan Swaminathan, Georgia Tech (Site Director)

Vision

This is NOT our vision

Vision

To enable fast, accurate design and verification of microelectronic circuits and systems by creating and applying machine learning algorithms to derive models used for electronic design automation

   – These models can also be used to obscure IP

ML in EDA Progression

1st Generation: Big data models for improving design productivity through machine learning

2nd Generation: "Little data" models for improving design productivity through machine intelligence

3rd Generation: Models and methods to flatten the design and verification hierarchy

CAEML

Center for Advanced Electronics Through Machine Learning

Joint NSF/industry funded center

Industry cost: $50,000 per year

Benefits: Rights to all IP; Early access to students; Mentor/guide/select projects

INTELLECTUAL PROPERTY REUSE THROUGH MACHINE LEARNING

Weiyi Qi, Bowen Li, Yang Yi, Brian Floyd, Paul Franzon
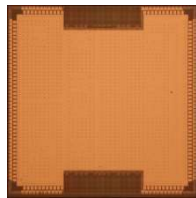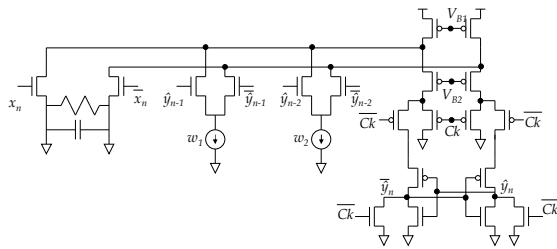
North Carolina State University

CAEML

CENTER FOR ADVANCED ELECTRONICS THROUGH MACHINE LEARNING

Problem Statement

- Port analog and custom digital IP from one technology to another, e.g.

Foundry X
32 nm
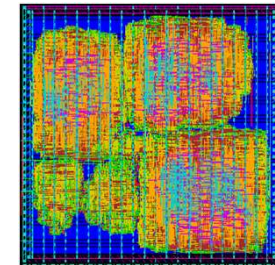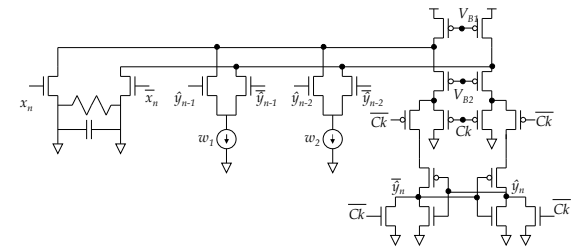
Foundry Y
28 nm

Solution Alternatives

Approach	Pros	Cons
Optimization using Spice Model	Direct, accurate	Very long latency
Optimization using design equations	Quick	Inaccurate
Expert design system	Works well	Requires expertize to be captured for each individual design
Optimization using Surrogate Model	Quick, accurate	Requires SM to be fitted offline
Bayesian optimization	Accurate, fewest overall simulations, Can start with SM	Longer latency than using SM only

Bayesian Optimization

- We propose to use a Bayesian optimization technique for efficient design optimization:

  Let f denote the statistical model and $\boldsymbol{D}$ the samples; we have:

  $$P(f|\boldsymbol{D}) = \frac{P(\boldsymbol{D}|f)P(f)}{P(\boldsymbol{D})} \propto P(\boldsymbol{D}|f)P(f)$$

  Posterior $\qquad\qquad\qquad$ Likelihood Prior Model

- Bayesian optimization flow:
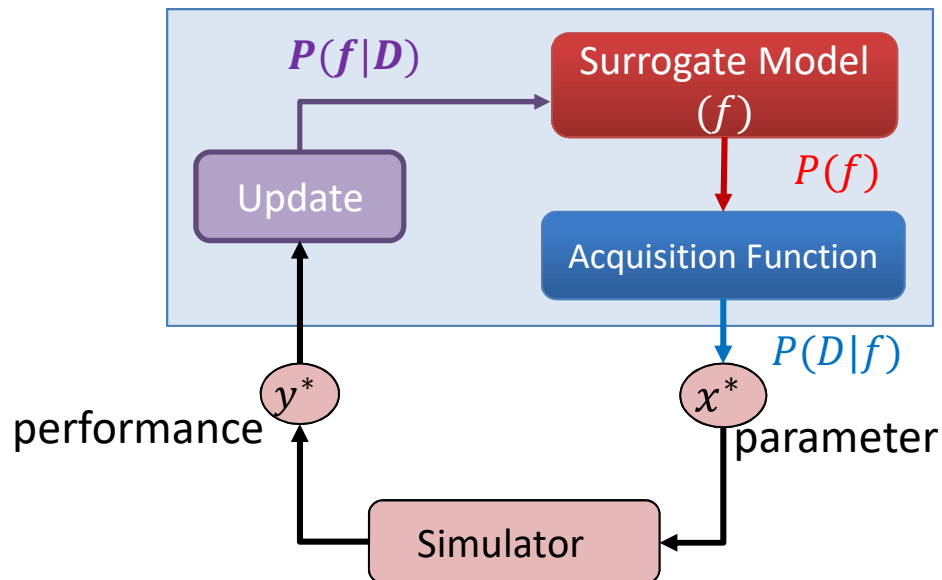
Two key components:

(1) Statistical surrogate model:
- Gaussian Process (GP) models or Student-T Process (TP) models
- Fit existing data and predict performance expectation and uncertainty; prior models are updated with newly acquired sample to form posterior models

(2) Acquisition function:
- Determining next best sample to simulate

Bayesian Optimization: Picking Next Point to Simulate

➢ Probability of Improvement (PI) calculates how probable it is that simulating a new point will improve f(x) at that point
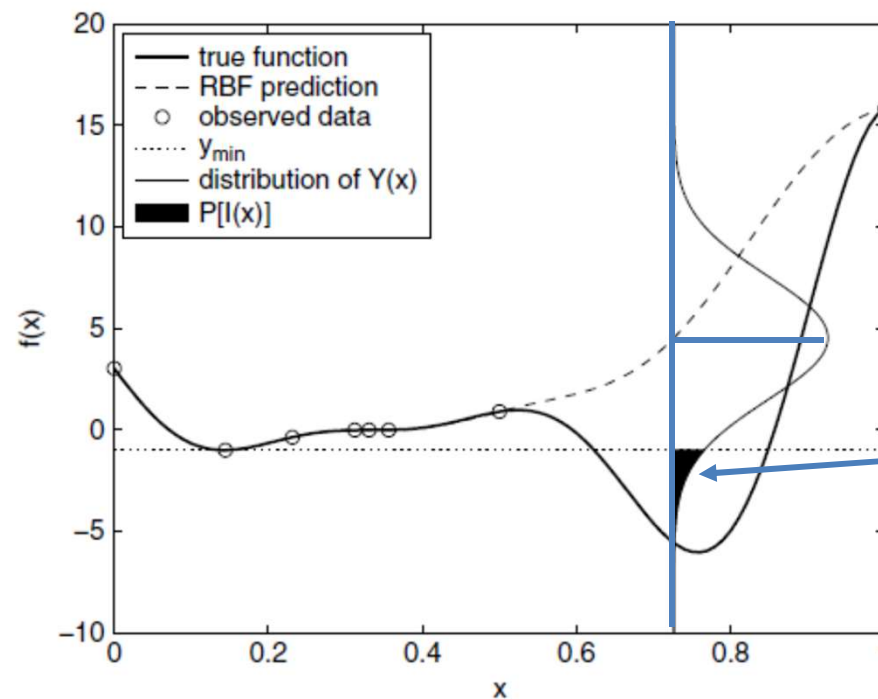
PI = area in black

Plot from: Forrester, Alexander, Andras Sobester, and Andy Keane. *Engineering design via surrogate modelling: a practical guide*. John Wiley & Sons, 2008.

Circuit Blocks to be Studied

- 77-GHz vehicular radar blocks:
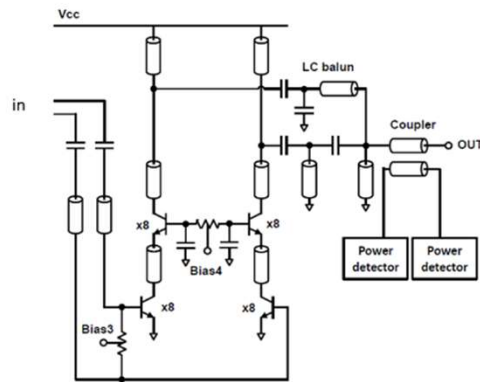
(a) 80-GHz power amplifier
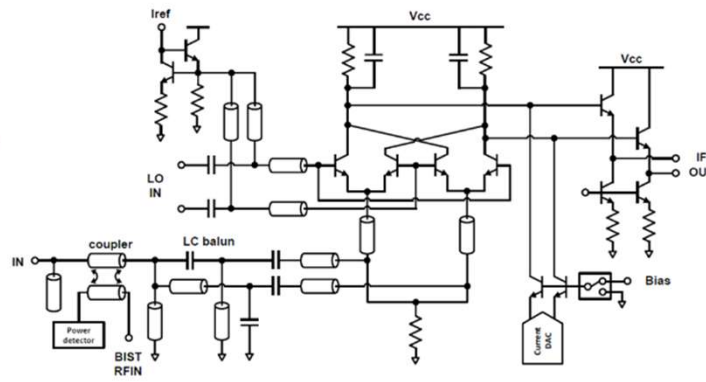
(b) 80-GHz downconversion mixer
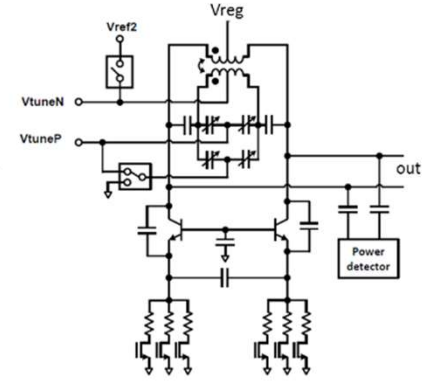
(c) 20-GHz voltage-controlled oscillator

- SerDes:

Equalizer Driver

Equalizer Receiver

DFE

Channel

clk

CDR

N:2 Mux   2:1 Mux
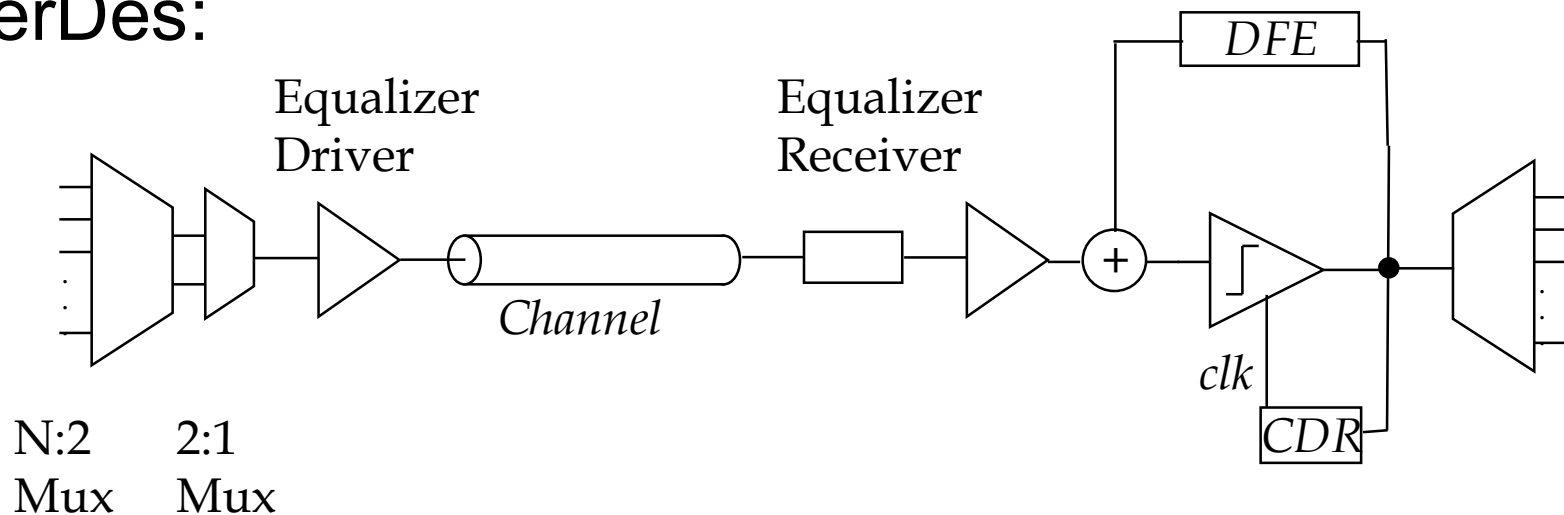
Preliminary Results:
Porting 77-GHz Balun from 8XP to 9HP

- 77-GHz PA uses balun at output. Conceptually simple, but surprisingly complicated to optimize through ML.

- Bayesian optimization subroutine first applied in existing technology (8XP).

- Then we reuse the balun in 9HP process with the same approach; the final optimized design will be used for tape-out
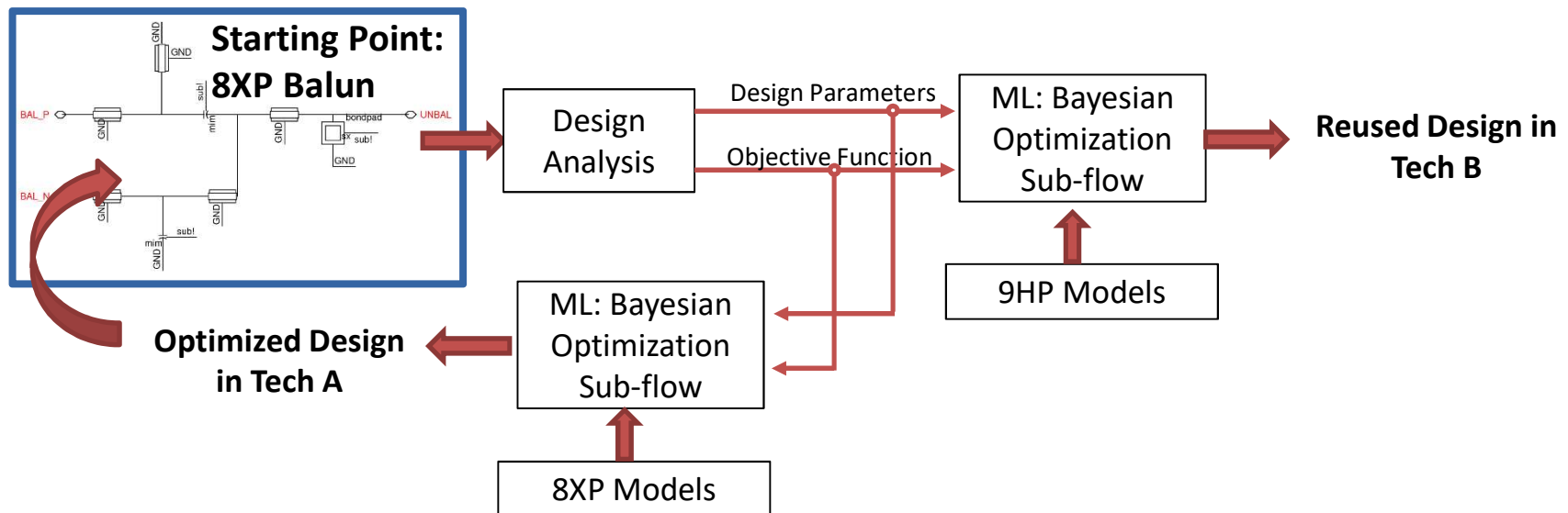
Step 1: Defining Range and Requirements for Balun

An LC balun is a commonly used passive balun in microwave IC that converts a signal into a pair of out-of-phase signals, or vice versa, while suppressing the common mode on the balanced port output.

Design Parameter	Range
I0_l	[30u, 1000u]
I0_w	[2u, 100u]
I0_s	[3u, 20u]
I1_l	[30u, 1000u]
I1_w	[2u, 100u]
I1_s	[3u, 20u]
C0	[20f, 200f]
C1	[20f, 200f]

P1: Balanced + 50 Ohm

P2: Balanced - 50 Ohm

P3: Unbalanced port 50 Ohm

S-Parameters	Requirements
$|S_{33}|$	N.A. (< -10 pref.)
$|S_{22} - S_{11}|$	< 0.1
$|\phi(S_{22} - S_{11})|$	< 15
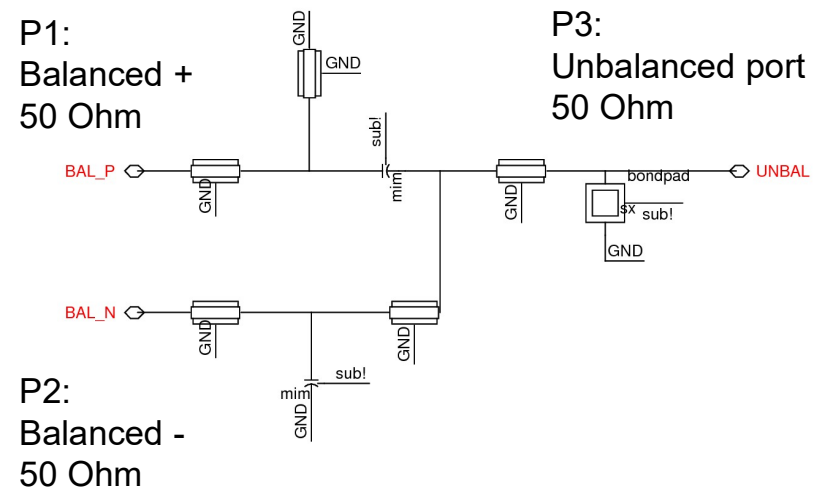
S-Parameters	Requirements
$dB\ Loss(S_{23}, S_{13})$	> -5
$|S_{23} - S_{13}|$	< 0.1
$|\phi(S_{23} - S_{13}) - 180|$	< 15

Step 2a: Design Analysis: Input Parameter Screening

- Design parameter **<u>screening</u>**

    - Not all design parameters are of equal importance
    - Large number of parameters will induce the curse of dimensionality

- The modified Morris' screening algorithm (Campolongo, 2007) uses one-factor-at-a-time (OFAT) sample scheme that depends *linearly* on the number of design parameters; suitable for complex design analysis.
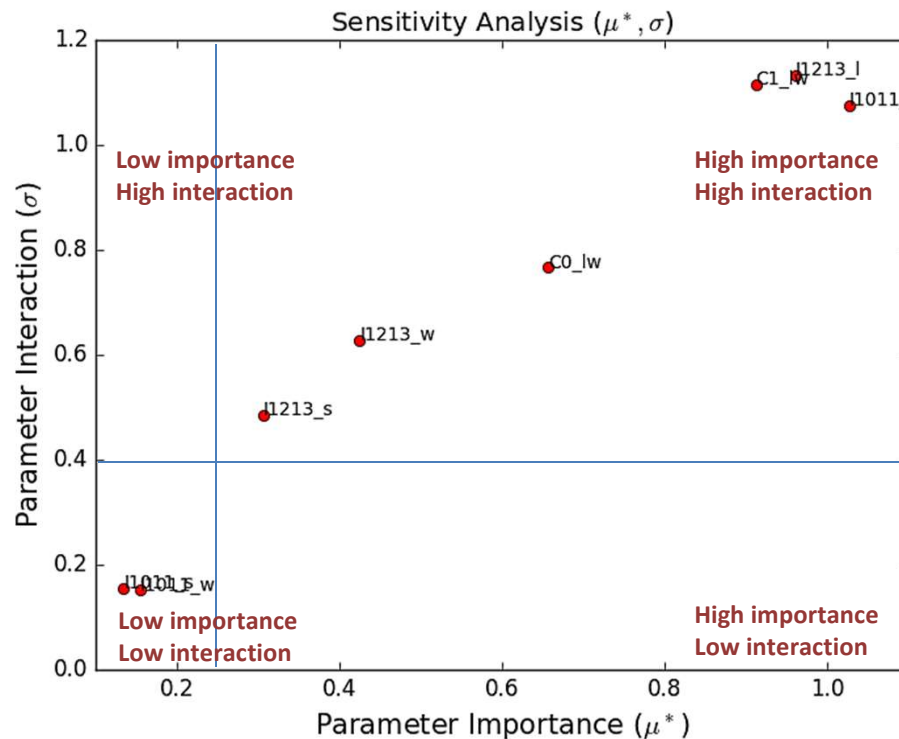
Result:

- *Trimmed two parameters that have low importance*

Step 2b: Design Space Exploration

- Design objective analysis:
  - Design objective analysis prevents over- or underestimated design goals
  - Designer can also learn about the design tradeoffs, either graphically or numerically, by examining the correlation coefficient table.
  - Designer can also find the upper- and lower-bounds for each design objective, and map them to [0,1] for multi-objective optimization
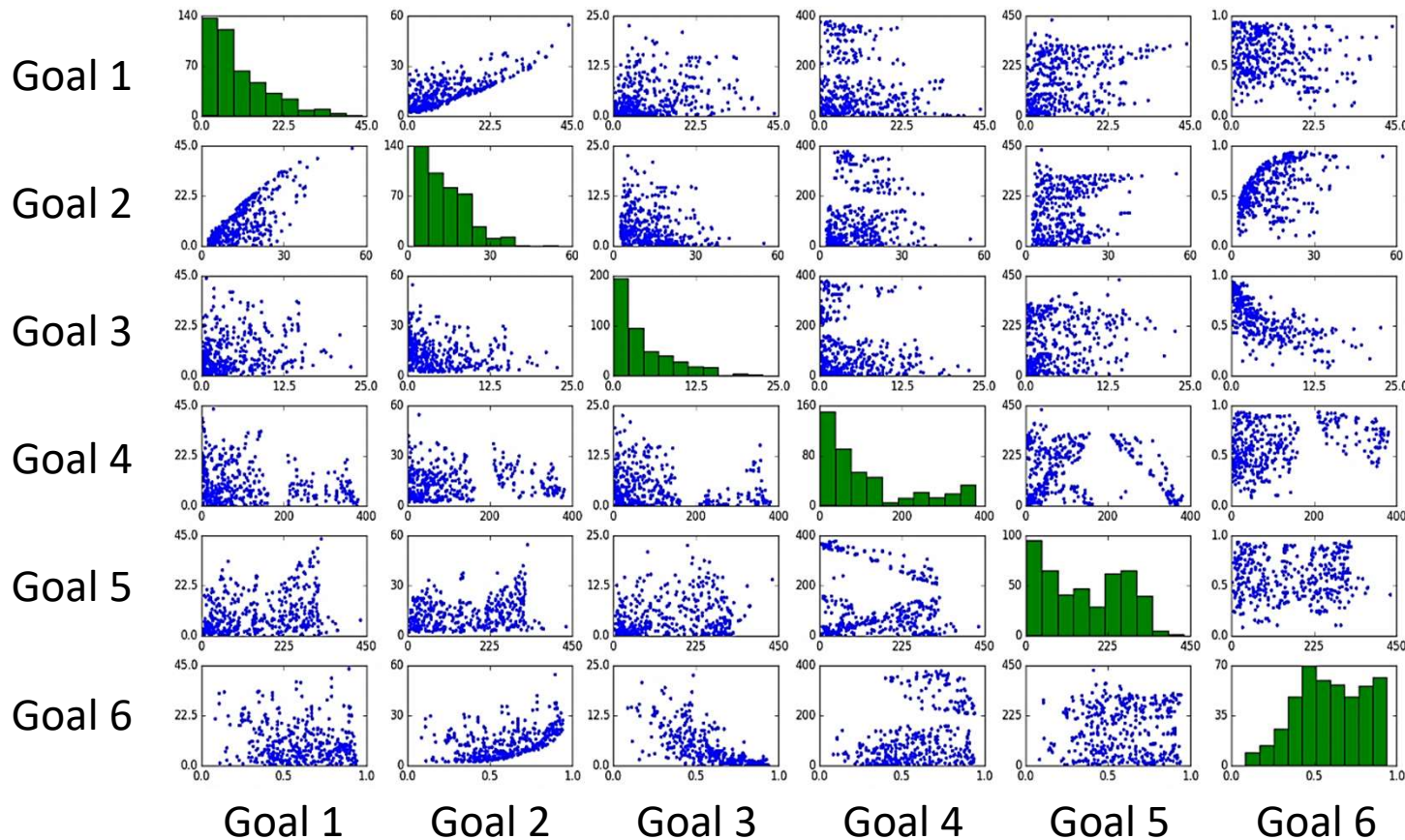
Result:

- *Feedback to user on trade-offs, correlations, and conflicts.*

Step 3: Optimization in Existing Technology

Three approaches for optimizing the balun design are compared:

(1) Genetic programming: A representation of evolutionary programming algorithms that are widely used for analog design synthesis/reuse

(2) Bayesian optimization: Use Gaussian process surrogate model

(3) Bayesian optimization: Use Student T process surrogate model

Balun Design Optimization

Bayesian Opt. is
>3x more efficient
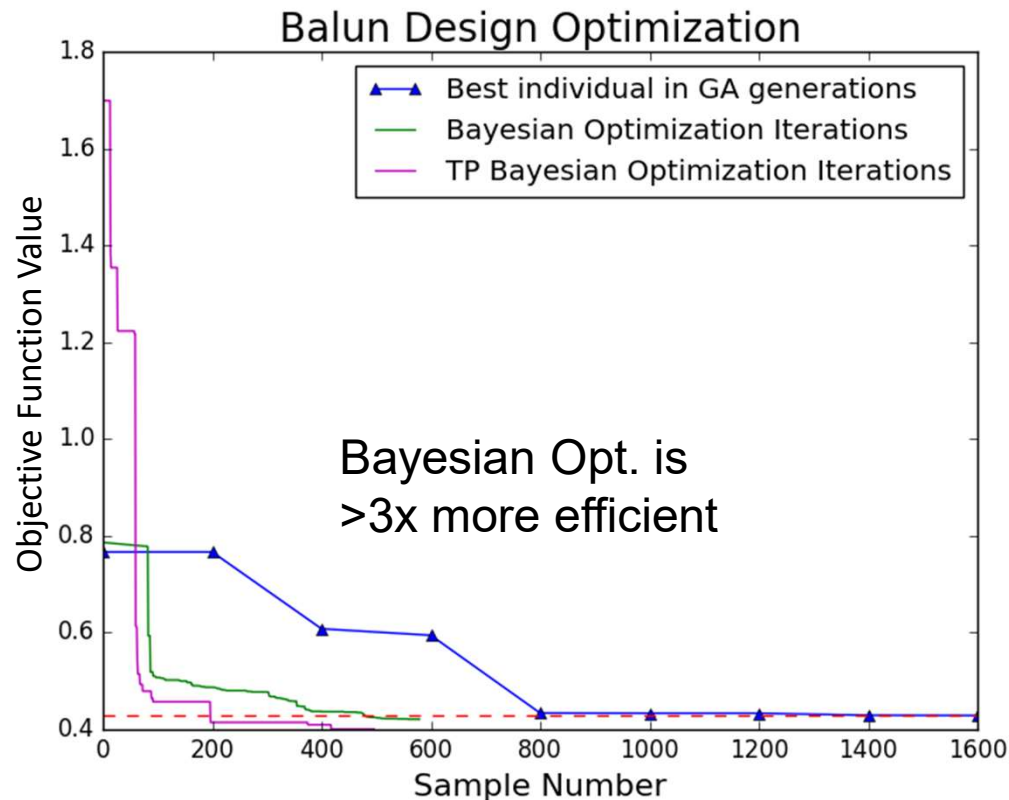
Step 3: Balun Optimization Result in Existing Technology

Human Design
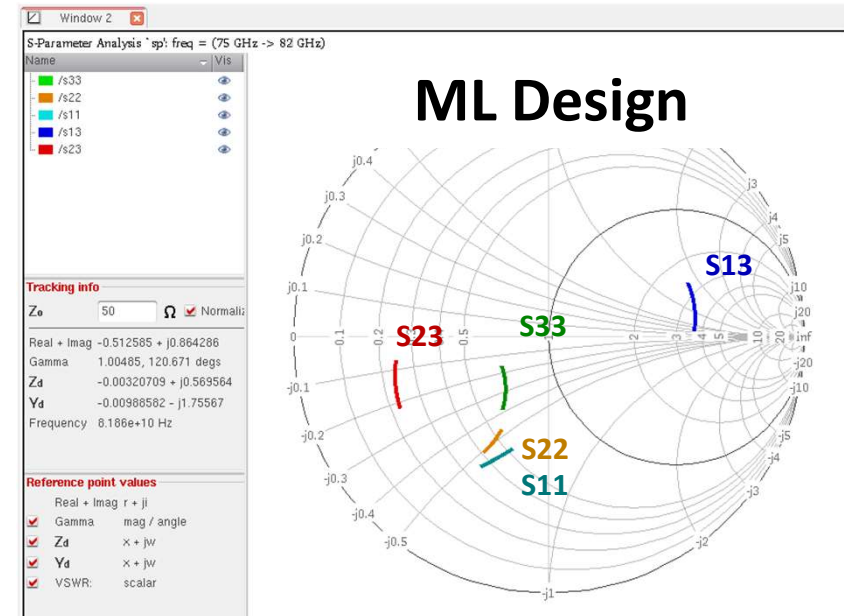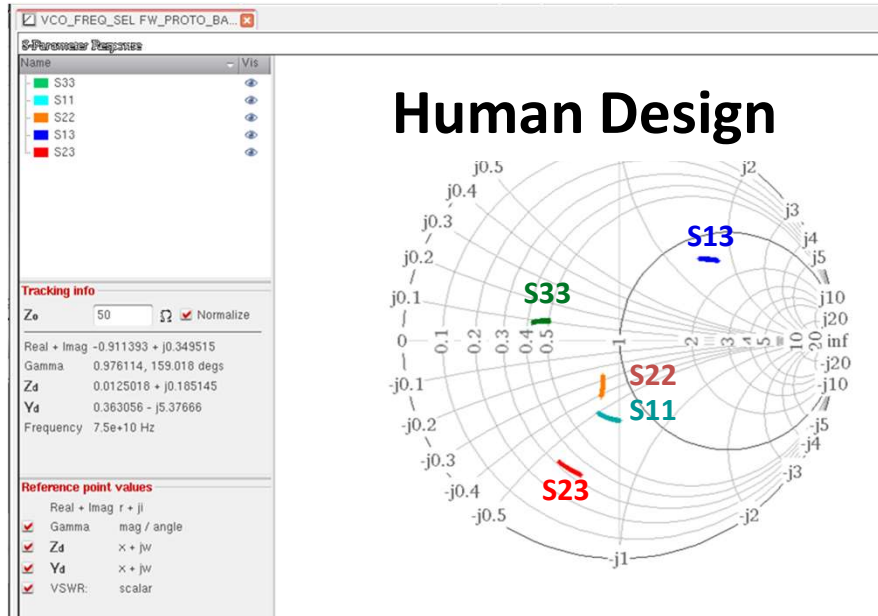
ML Design

Metric	Target	Human Result	ML Result
$|S_{33}|$	< -10	-7.7	**-10.5**
$|S_{22} - S_{11}|$	< 0.1	0.22	**0.06**
$|\phi(S_{22} - S_{11})|$	< 15	25.5	**9**
$|S_{23} - S_{13}|$	< 0.1	0.31	**0.09**
$|\phi(S_{23} - S_{13}) - 180|$	< 15	28.7	**4.5**
$dB\ Loss(S_{23}, S_{13})$	> -5	-9.4	**-4.8**

Step 4: Porting to New Technology (9HP)

Now we migrate the passive LC balun design into IBM 9HP node with **three key components kept consistent**, which makes IP migration a *push-button* process:

(1) Design IP topology

(2) Algorithm settings:

–   **Surrogate model type & acquisition function**

(3) Design objective functions:

–   **Objective scalarization weights**

Balun Design Reuse in 9HP

Design Reuse in 9HP
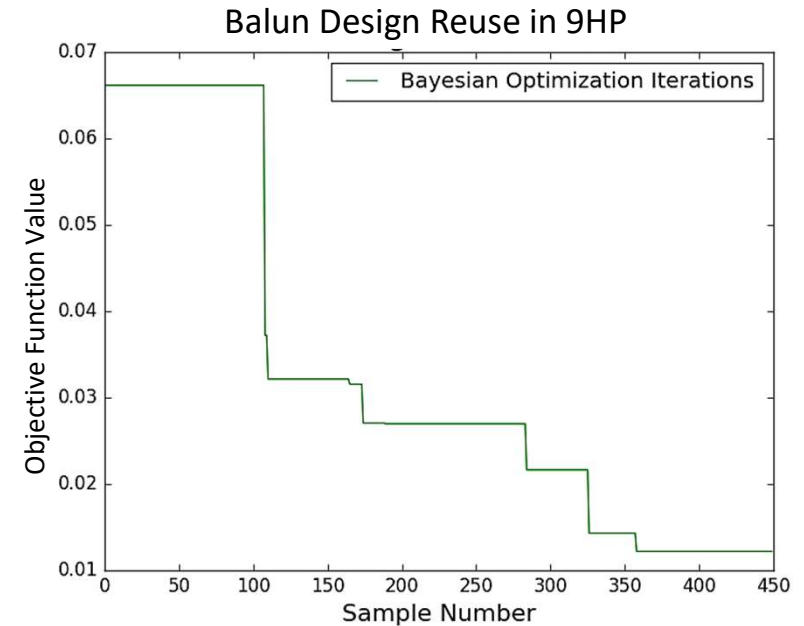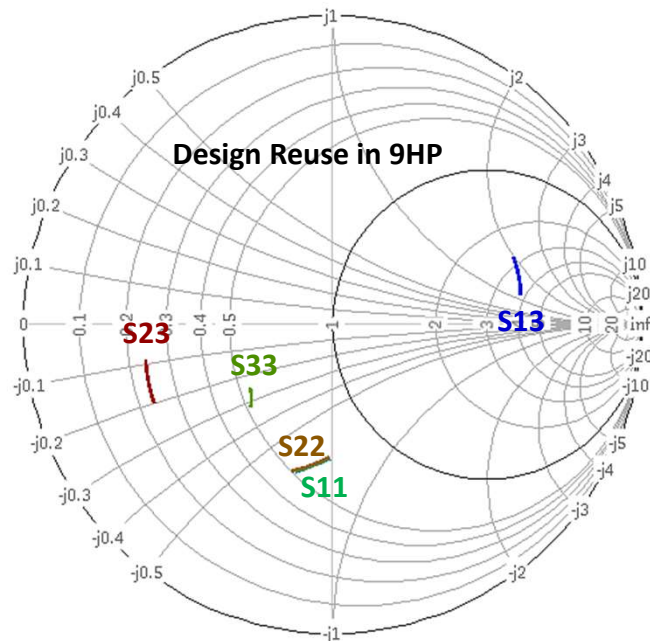
Metric	Target	Result in 8XP	Result in 9HP
$\|S_{33}\|$	N.A. (< -10 pref.)	-10.5	**-9.7**
$\|S_{22} - S_{11}\|$	< 0.1	0.06	**0.01**
$\|\phi(S_{22} - S_{11})\|$	< 15	9	**1.5**
$\|S_{23} - S_{13}\|$	< 0.1	0.09	**0.04**
$\|\phi(S_{23} - S_{13}) - 180\|$	< 15	4.5	**3.4**
$dB\ Loss(S_{23}, S_{13})$	> -5	-4.8	**-2.4**

NC STATE

MACHINE LEARNING IN PHYSICAL DESIGN

Bowen Li, Weiyi Qi,

Billy Huggins,

W. Rhett Davis,

Paul Franzon

ECE Department

North Carolina State University
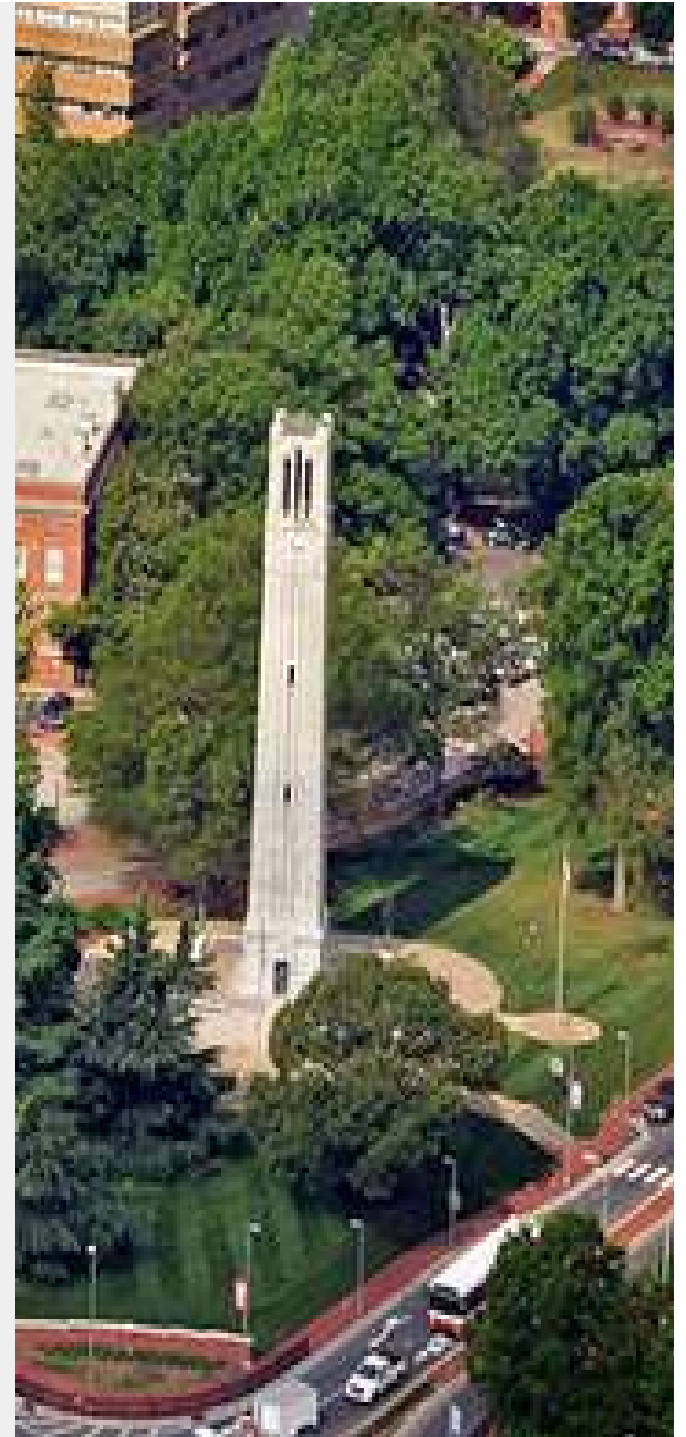
Physical Design
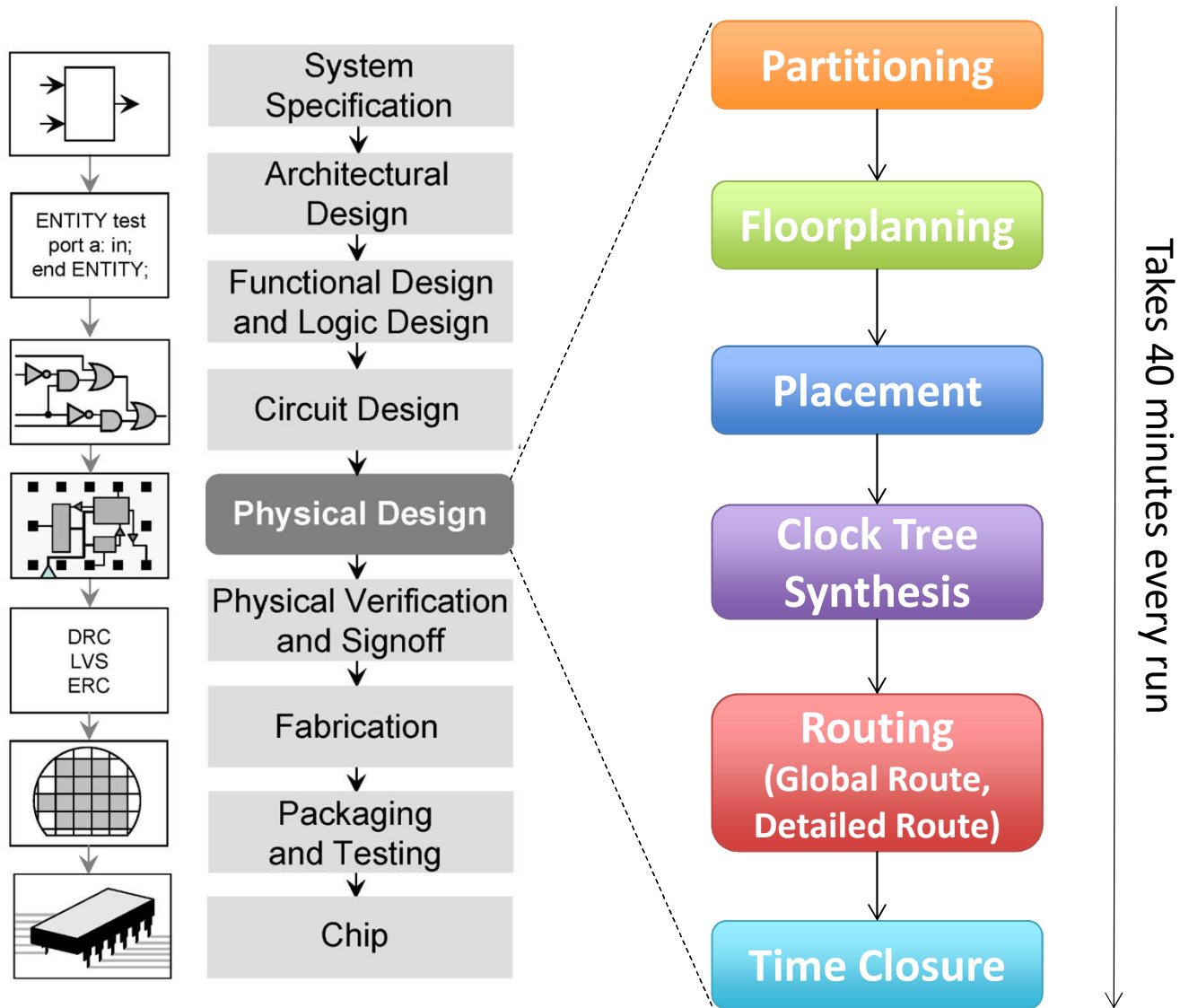
Source: Wikimedia Commons

System Specification → Architectural Design → Functional Design and Logic Design → Circuit Design → **Physical Design** → Physical Verification and Signoff → Fabrication → Packaging and Testing → Chip

Partitioning → Floorplanning → Placement → Clock Tree Synthesis → Routing (Global Route, Detailed Route) → Time Closure

Takes 40 minutes every run

Routing

Problem Statement:
How to set up control knobs to achieve specific desired outcomes

Input Knob	Meaning		Output	Units
Clock Target	Clock frequency		Power	W
Num_layer	Number of routing layers		Area	Sq.mm.
Init_density_ratio	% cell area		Setup Slack	ps
skew	Clock skew		Hold slack	ps
Sink_max_tran	Clock tree leaf trans time		Congestion	% density
Buf_max_tran	Clock tree buffer tr time		DRC error count	count

Initial Experiment

Cortex SOC:

Gate count: 18k gates

Net count: 18k nets

Target clock: 10 ns

Design Goal:

 Minimize area while meeting timing and being DRC clean.

Technology:

NCSU 45 PDK
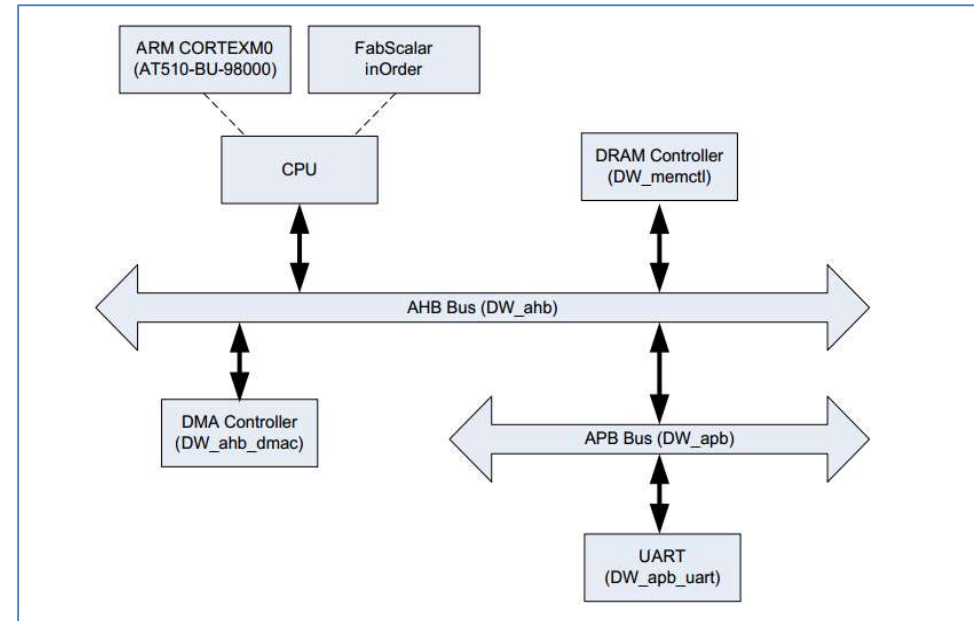
Building a Surrogate Model

Model building:

- Each routing run takes 40 minutes
- Total of ~50 runs needed to complete model
- Total time: Overnight
- Kriging Model

Models fitted:

- Congestion
- Setup slack
- Hold slack

Plot of Hold_slack using KrigingModel

Hold time

Hold_slack

SetMaxTran
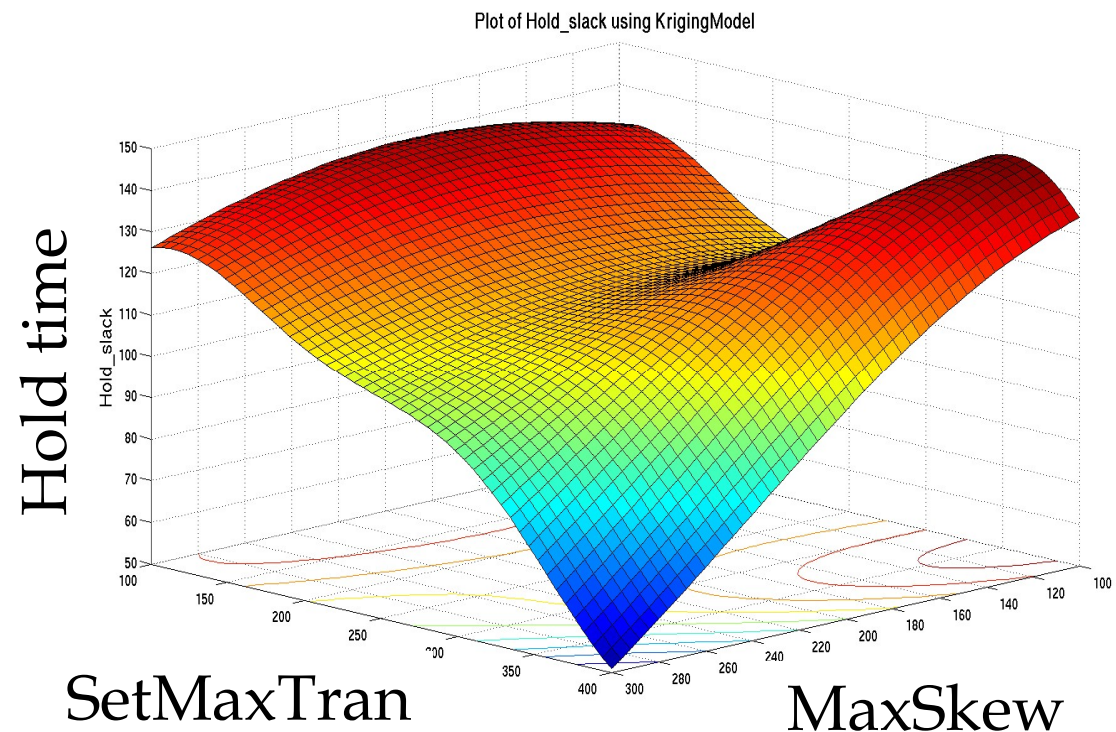
MaxSkew

Physical design results

➤ Design Iterations after model lookup

Iter.	CLKper	Den.	Layer	Max Skew	Sink Max Tran	Cong.	Viol	Hold slack	Setup Slack	Comments
1	10	0.6	8	300	400	0.28H /1.51V	105	-61.3ps	6.46ns	Over-congested; Hold time violated
2	10	0.5	8	300	400	0.03H /0.39V	6	-48.7ps	6.55ns	Over-congested; Hold time violated
3	10	0.45	8	300	400	0.02H /0.11V	0	2.4ps	6.48ns	No DRC errors; hold fixed; hold margin is low
4	10	0.45	8	200	300	0.02H /0.17V	0	10.5ps	6.37ns	Final Design
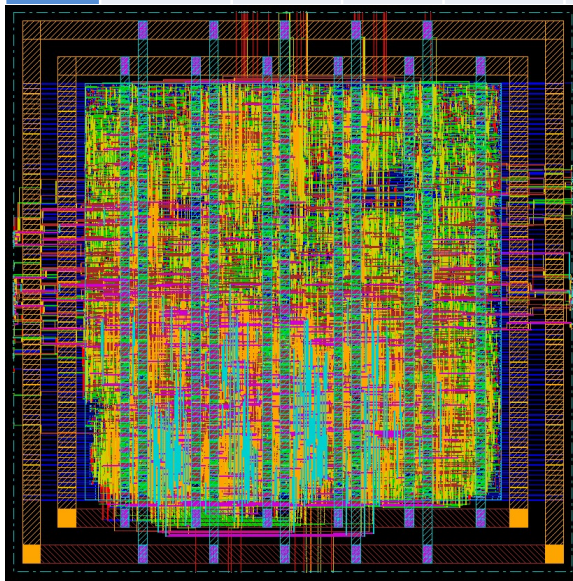
❑ Surrogate model provides guidance for design & optimization
❑ Able to achieve an optimal design with 4 iterations
❑ Human designer took 20 iterations

# of Standard Cells		39990
Area (μm^2)	Core	98109.284 (313.224*313.224)
	Chip	54363.008 (233.158*233.158)
Cell Density		55.4 %

Building a more accurate model

1. Data Selection

Six Physical Design Features		Ten Surrogate Models for each outputs	Ten Results after Placement & Global Route	
	Clock_target			hold_slack_trial
	Num_layer			Worst Negative Slack
	Init_density			Total Negative Slack
	ratio			violating_path
	SinkMaxTran			x_neg_5_8
	BufMaxTran			x_neg_1_4
				x_pos_0_10
				x_pos_11_20
				power_trial
				area_trial

Machine Learning Methods	Four Results after Detailed Route	
		Power
		Area
		Hold Slack
		Number of DRC Violations

Surrogate Modeling for GR in Physical Design
Surrogate Model Builders

- Artificial Neural Networks (ann)

- Kriging

- Radial Basis Function (rbf)

- Kriging genetic

- …

Surrogate Model Fitting Alternatives

3. Surrogate Modeling for GR in Physical Design
Model Accuracy: Root Relative Squared Error

Six Physical Design Features	Clock_target
	Num_layer
	Init_density
	ratio
	SinkMaxTran
	BufMaxTran

→ Ten Surrogate Models for each outputs →

Ten Results after Global Route

| hold_slack_trial |
| Worst Negative Slack |
| Total Negative Slack |
| violating_path |
| x_neg_5_8 |
| x_neg_1_4 |
| x_pos_0_10 |
| x_pos_11_20 |
| power_trial |
| area_trial |
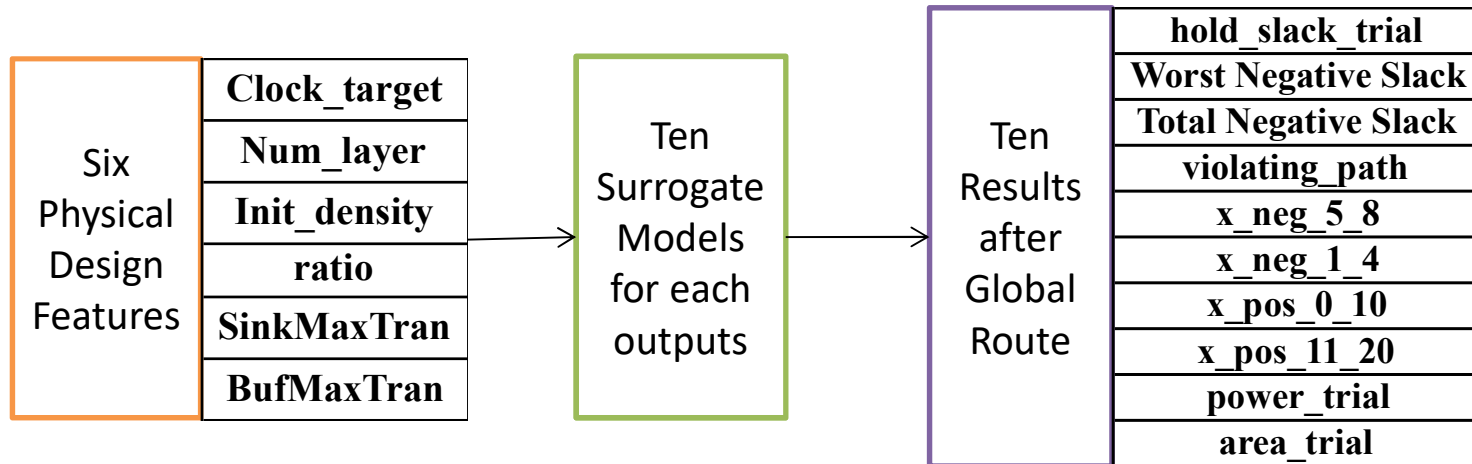
The Root Relative Squared Error (RRSE) is relative to squared error compared to a simple predictor (the average of values).

$$RRSE = \sqrt{RSE} = \sqrt{\sum \left(f(\overline{x}_i) - y_i \right)^2 / \sum \left(\bar{y}_i - y_i \right)^2}$$

RRSE is close to 0 → model is much better than a simple predictor
RRSE is close to or larger than 1 → model is worse than a simple predictor

RRSE < 0.5 is the target.

3. Surrogate Modeling for GR in Physical Design
Model Performances

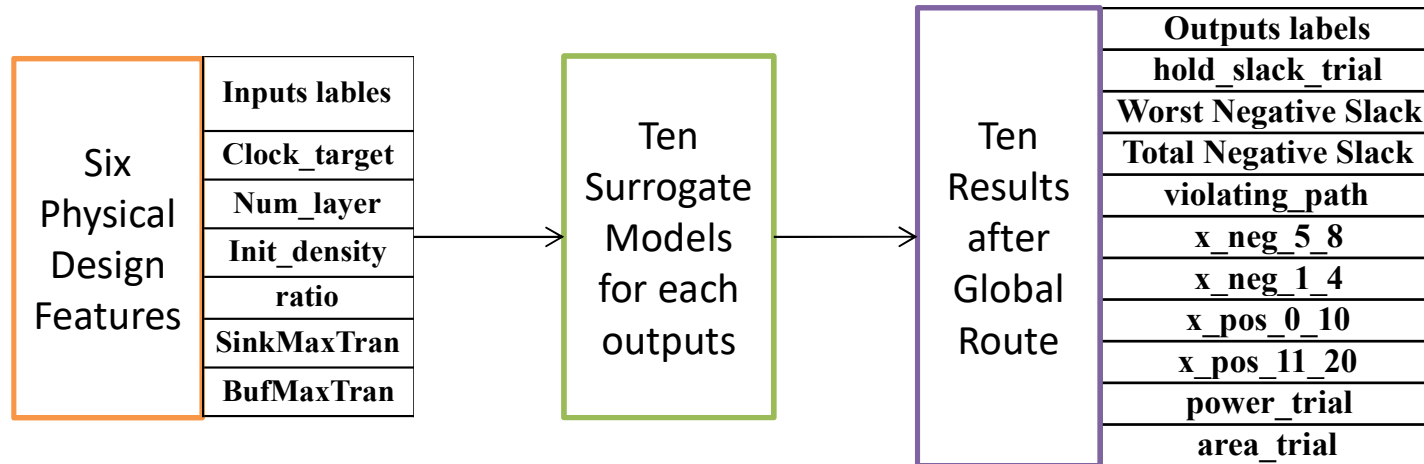|	area_trial	TNS	violating_path	WNS	x_neg_1_4	x_neg_5_8	x_pos_0_10	x_pos_11_20	hold_slack_trial	power_trial
anngenetic	**0.000**	**0.242**	0.383	0.076	**0.496**	**0.295**	**0.136**	**0.139**	1.001	**0.922**
ann	0.003	0.270	**0.364**	0.079	0.517	0.302	0.193	0.227	**1.000**	0.939
annfixed	0.003	0.267	0.411	0.090	0.536	0.302	0.201	0.181	1.002	0.948
rational	0.004	0.262	0.394	**0.072**	0.542	0.428	0.362	0.371	1.016	0.961
gpmlgenetic	0.000	0.310	0.392	0.090	0.521	0.439	0.366	0.376	1.000	0.929
kriginggenetic	0.001	0.311	0.459	0.096	0.591	0.382	0.269	0.275	1.101	0.942
lssvmgenetic	0.001	0.309	0.403	0.088	0.522	0.437	0.377	0.378	1.000	0.930
elm	0.000	0.322	0.400	0.093	0.523	0.448	0.373	0.377	1.000	0.936
kriging	0.000	0.338	0.461	0.176	0.612	0.405	0.297	0.307	1.003	1.003
gpmldirect	0.004	0.326	0.429	0.099	0.547	0.449	0.410	0.421	1.000	0.930
rbf	0.038	0.315	0.416	0.108	0.552	0.492	0.403	0.421	1.020	0.954
rbfgenetic	0.012	0.322	0.418	0.098	0.542	0.473	0.441	0.466	1.000	0.966
krigingpso	0.004	0.334	0.509	0.128	0.627	0.600	0.433	0.407	1.017	0.953
krigingoptim	0.028	0.391	0.511	0.126	0.812	0.651	0.340	0.688	1.040	1.130
krigingnsga	0.064	0.635	0.683	0.133	0.637	0.650	0.622	0.650	1.060	0.973
ipol	0.079	0.369	0.798	0.098	0.812	0.791	0.808	0.829	1.064	1.397

Sum of RRSE increases ↓

3. Surrogate Modeling for GR in Physical Design
GR Modeling Conclusion

	Inputs lables		Ten Surrogate Models for each outputs		Ten Results after Global Route	Outputs labels
Six Physical Design Features	Clock_target					hold_slack_trial
	Num_layer					Worst Negative Slack
	Init_density ratio					Total Negative Slack
	SinkMaxTran					violating_path
	BufMaxTran					x_neg_5_8
						x_neg_1_4
						x_pos_0_10
						x_pos_11_20
						power_trial
						area_trial

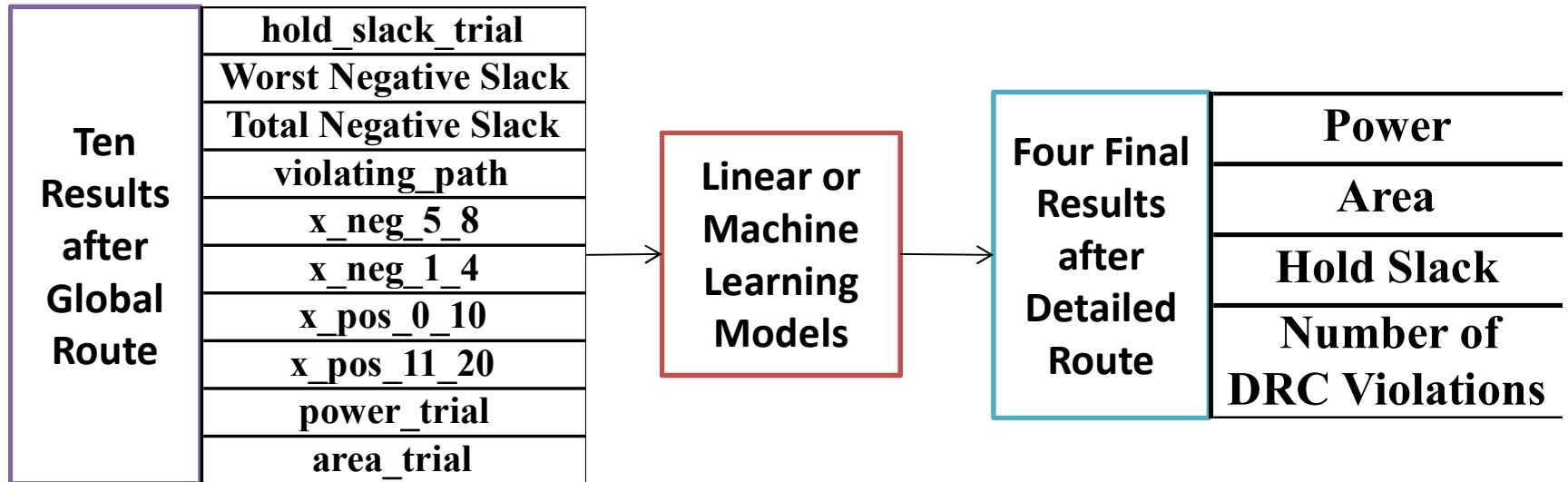**Global Results can be predicted correctly:**

- *Area, Total Negative Slack, number of violating paths, Worst Negative Slack, four groups of remaining tracks*

Best Model Builder:

- *Anngenetic*

4. Machine Learning for DR in Physical Design
DR Modeling Conclusion

Ten Results after Global Route		Linear or Machine Learning Models	Four Final Results after Detailed Route	
	hold_slack_trial			Power
	Worst Negative Slack			Area
	Total Negative Slack			Hold Slack
	violating_path			Number of DRC Violations
	x_neg_5_8			
	x_neg_1_4			
	x_pos_0_10			
	x_pos_11_20			
	power_trial			
	area_trial			

- Linear regression model for power and area

- Neural Networks model for hold slack

- Decision Tree models for hold slack and the number of DRC violations

Conclusions

- IP Reuse
  - Can result in more optimal analog designs than human designer
  - Can automate analog IP transfer between nodes
  - AND provide models for mixed signal verification

- Physical Design
  - Correct model choice permits problem to be modeled

Acknowledgements

▷ **Funding for cortical accelerators:**

◆ Google, DARPA

▷ **Students:**

◆ Joshua Schabel, Lee Baker, Sumon Dey, Weifu Li

▷ **Funding for CAEML**

◆ NSF + 11 member companies

◆ DARPA funded some of the background work (HEALICS)

▷ **NCSU Faculty:** Brian Floyd, Rhett Davis

▷ **NCSU Students:**

◆ Bowen Li, Weiyi Qi, Yi Wang, Billy Hutchins, Tsing Zhu