Ultra-Fast Variability-Aware Optimization of Mixed-Signal Designs using Bootstrapped Kriging

S. P. Mohanty[1], E. Kougianos[2], and V. P. Yanambaka[3]

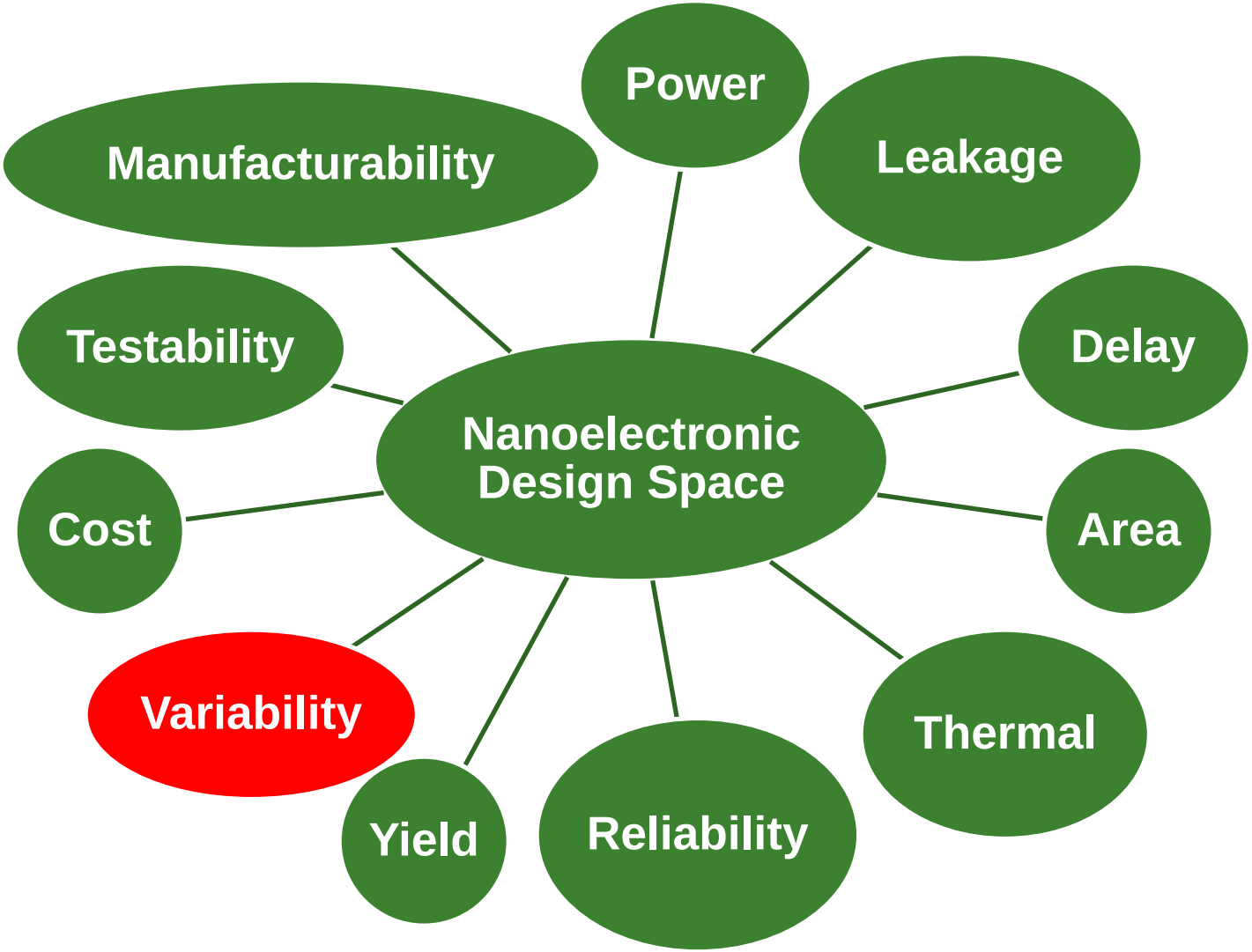
NanoSystem Design Laboratory (NSDL, http://nsdl.cse.unt.edu)
University of North Texas, Denton, TX 76203, USA.[1,2,3]

Email: saraju.mohanty@unt.edu[1], elias.kougianos@unt.edu[2] and VenkataPrasanthYanambaka@my.unt.edu[3]

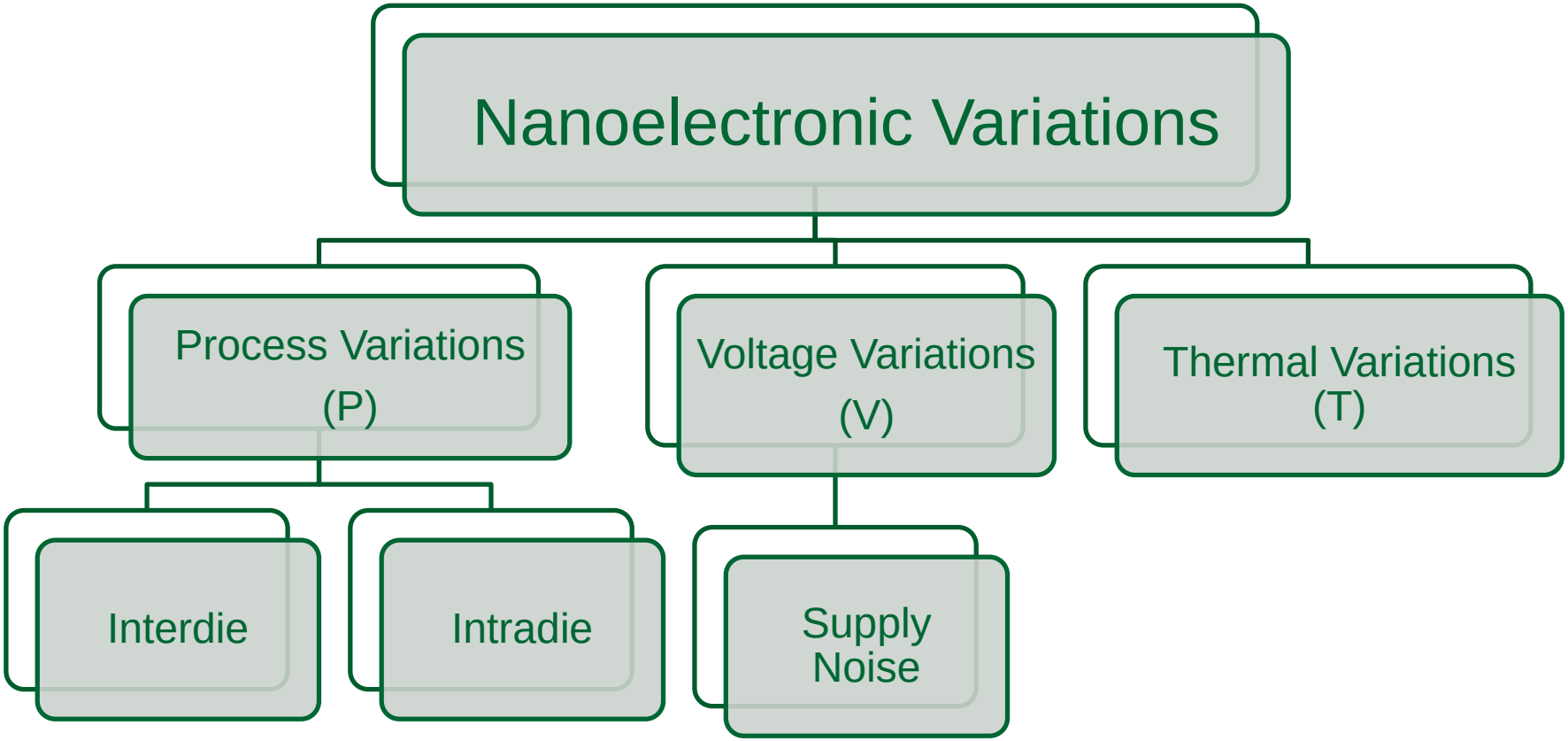
Outline of the talk

- Background and Motivation
- Novel Contributions
- Process Variation Aware Ultra-Fast Design Optimization Flow for Mixed Signal Circuits
- PSO Algorithm
- Conclusions

Nanoelectronics : Challenges



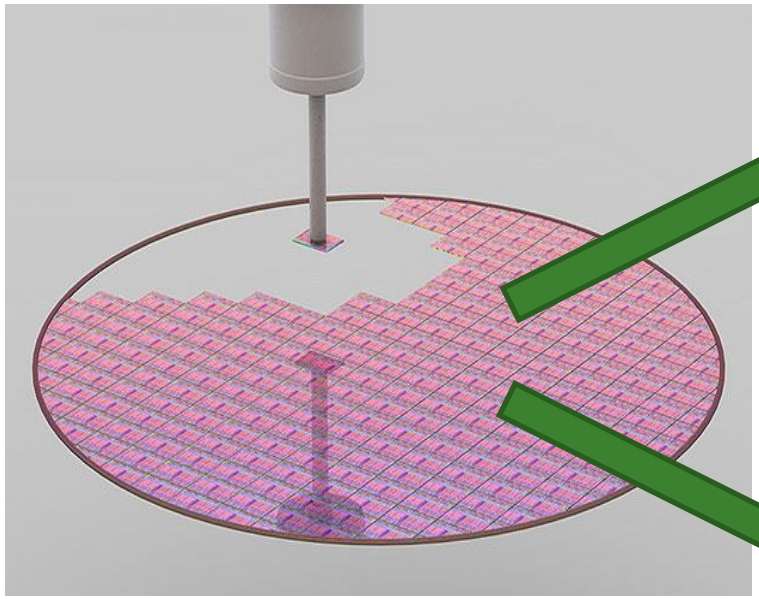
Nanoscale Variability : Overall



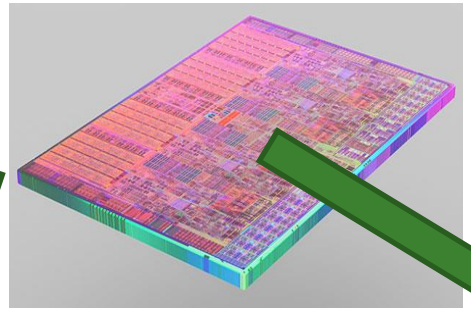
Nanoelectronics Variability ?

- Discrepancy between chip parameters --

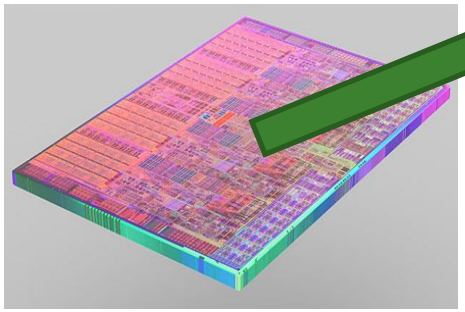
  Design Time versus Actual Post Fabrication

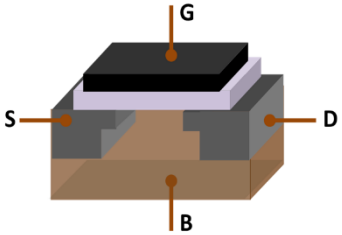


Same Design Fabricated

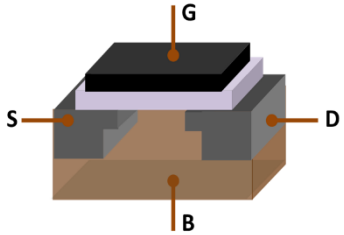


Each Chip has
Different Performance





Each Transistor
is Different

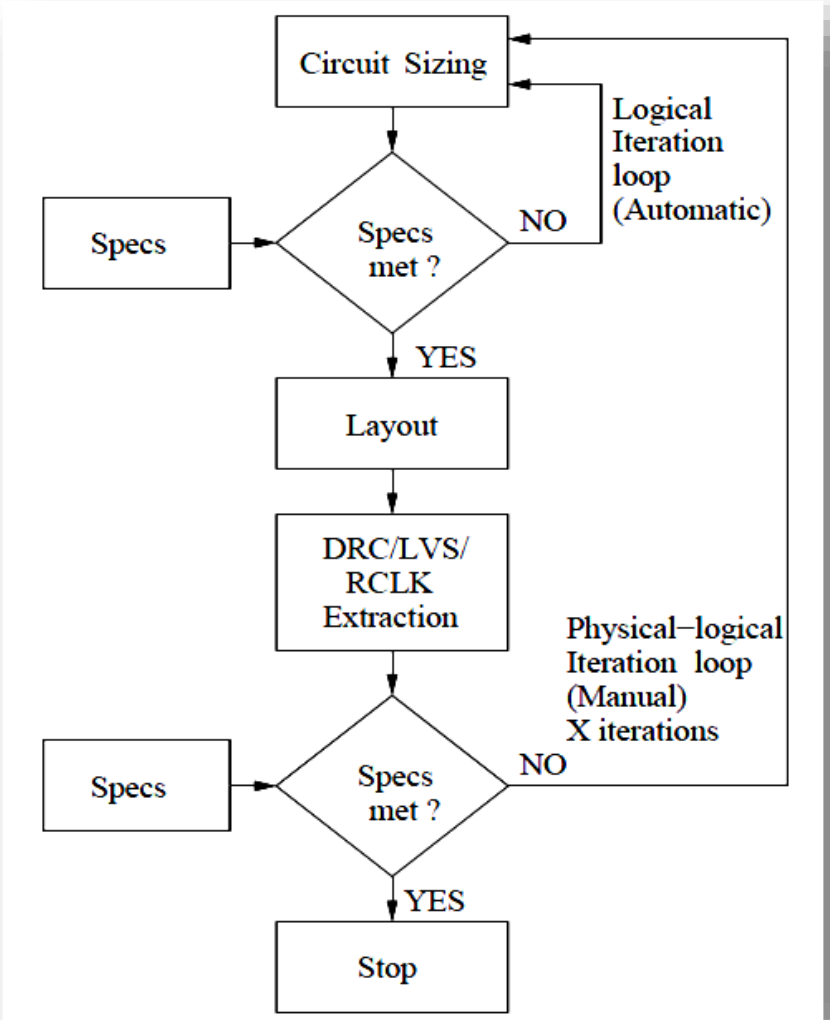


Source: http://apcmag.com/picture-gallery-how-a-chip-is-made.htm

Research Questions

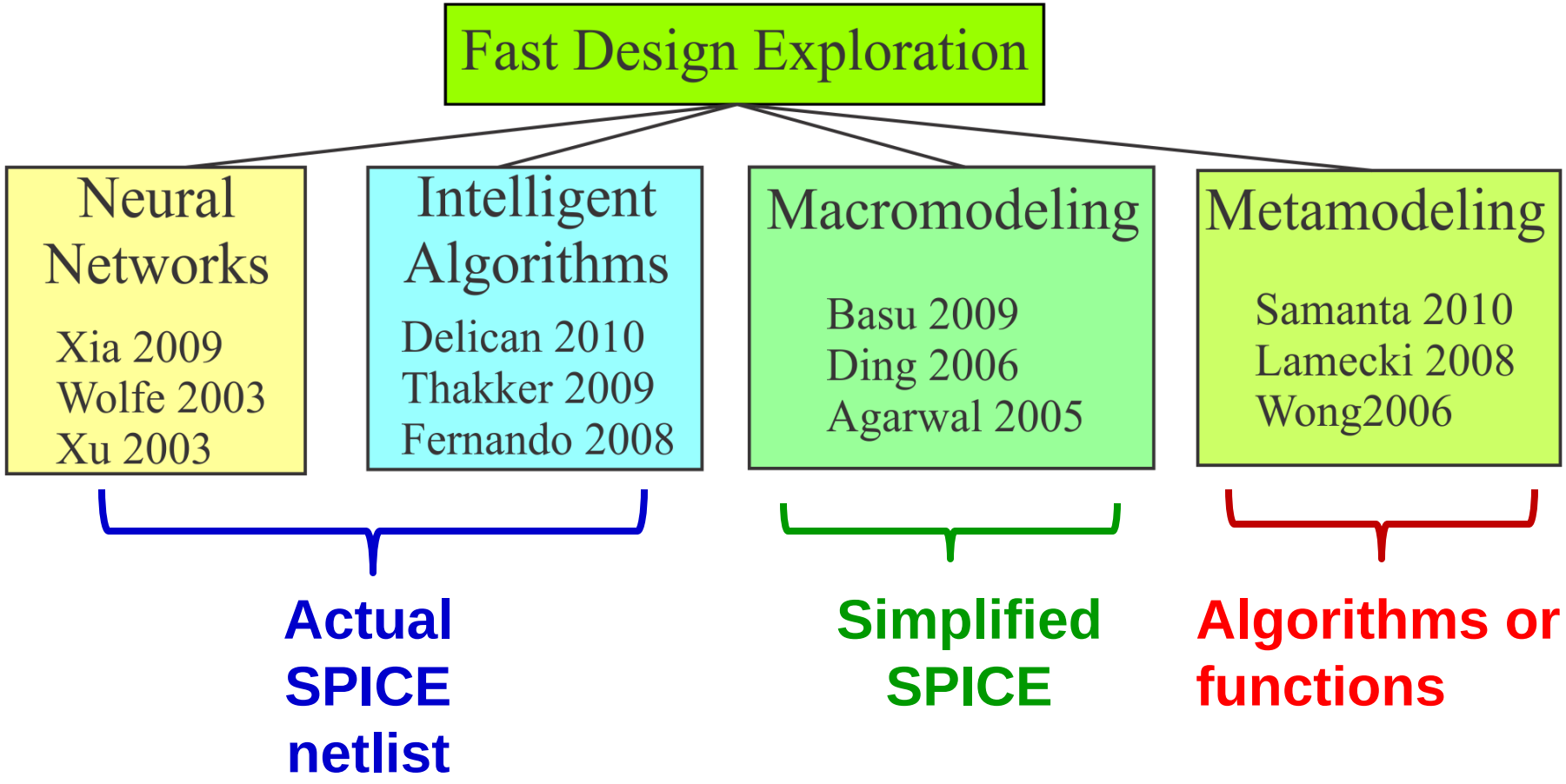
- How process variations effects can be captured in the design cycle?
- How process variation effects can be mitigated in the design cycle?
- How to achieve the above two with minimal design cycle consumed?

Standard Design Flow – Very Slow



- Standard design flow requires multiple manual iterations on the back-end layout to achieve parasitic closure between front-end circuit and back-end layout.
- Longer design cycle time.
- Error prone design.
- Higher non-recurrent cost.
- Difficult to handle nanoscale challenges.

Related Research: Fast Exploration



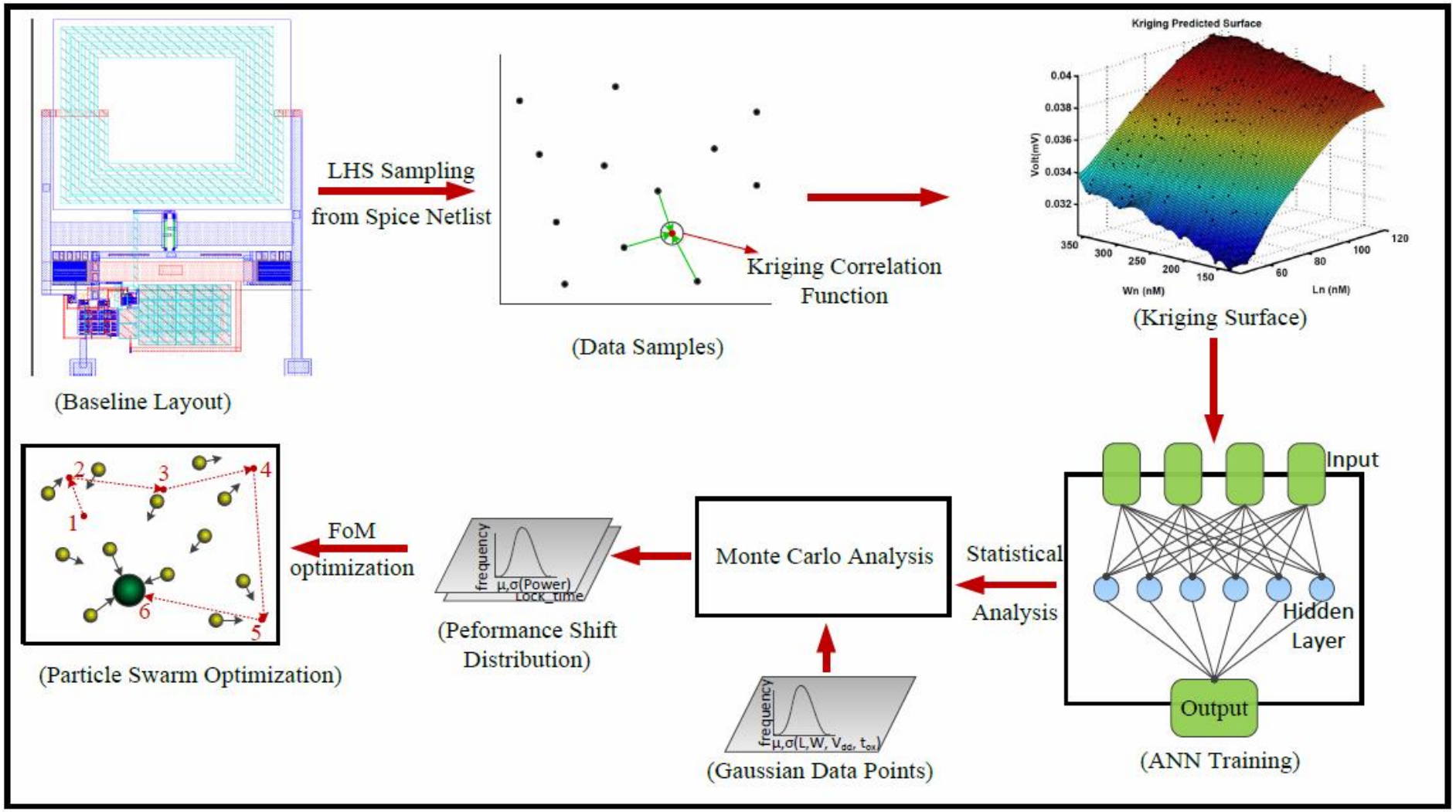
Related Research: Metamodels

- One of the most common and reliable method used is Polynomial Regression
- Non-Polynomial metamodels built from Neural Networks have been reported to surpass Polynomial metamodels
- Weight training process is critical in developing Neural Network models
- Kriging training for Neural Networks provides tradeoff between the accuracy of Kriging and scalability of NN models

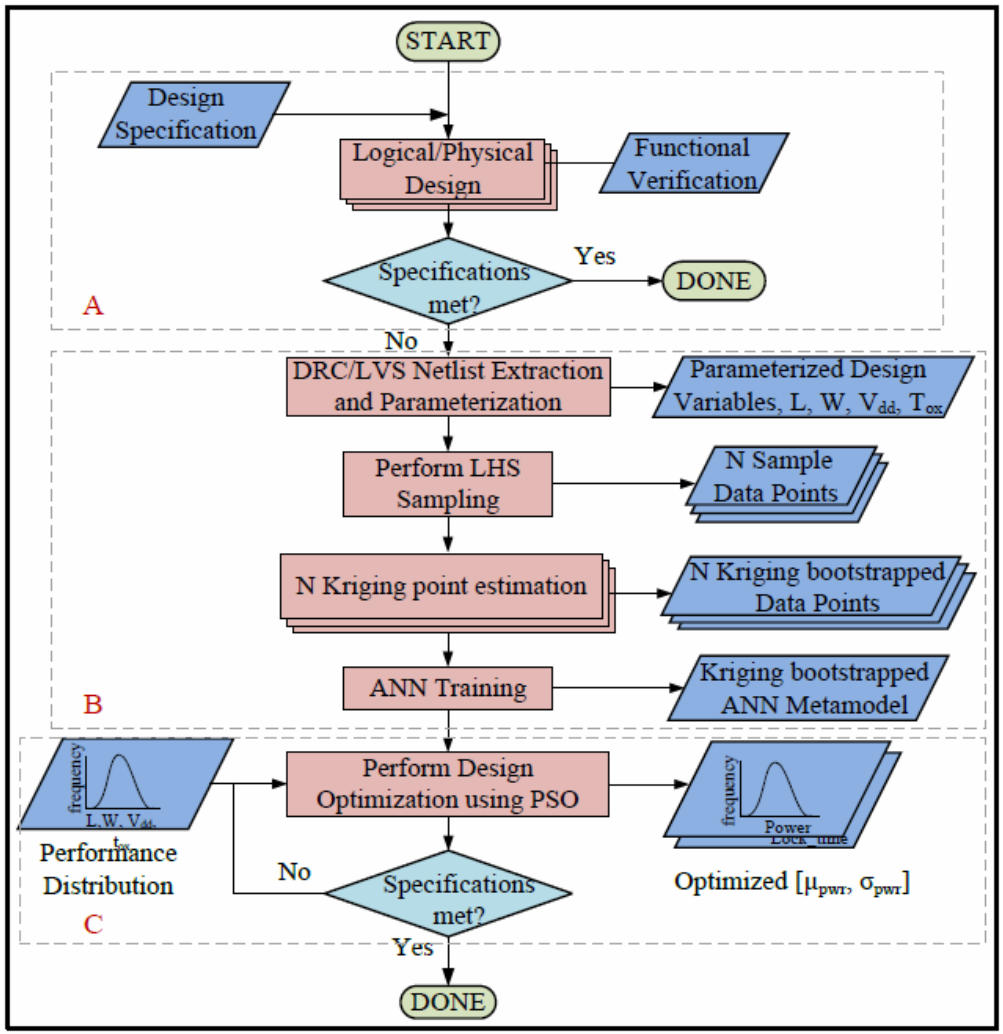
Novel Contributions of This Paper

- Fast and accurate physical design and optimization
- Adaption of PSO algorithm for nano-CMOS based process variation aware optimization
- Case study exploration using a 180nm CMOS based PLL design

Process Variation Aware Ultra-Fast Design Optimization Flow for Mixed-Signal Circuits

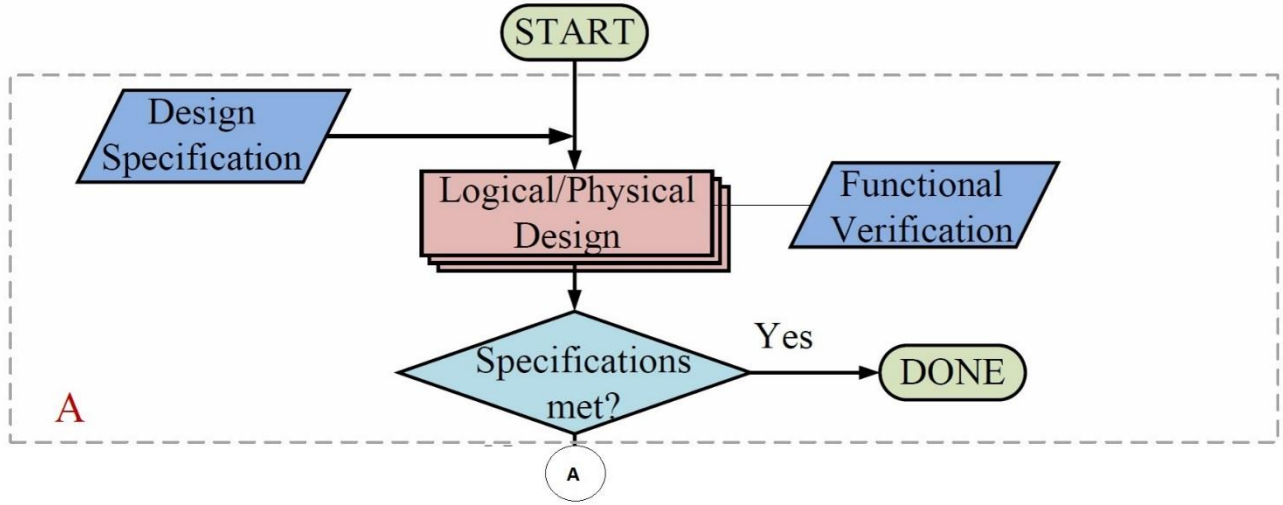


Proposed Design Optimization Flow



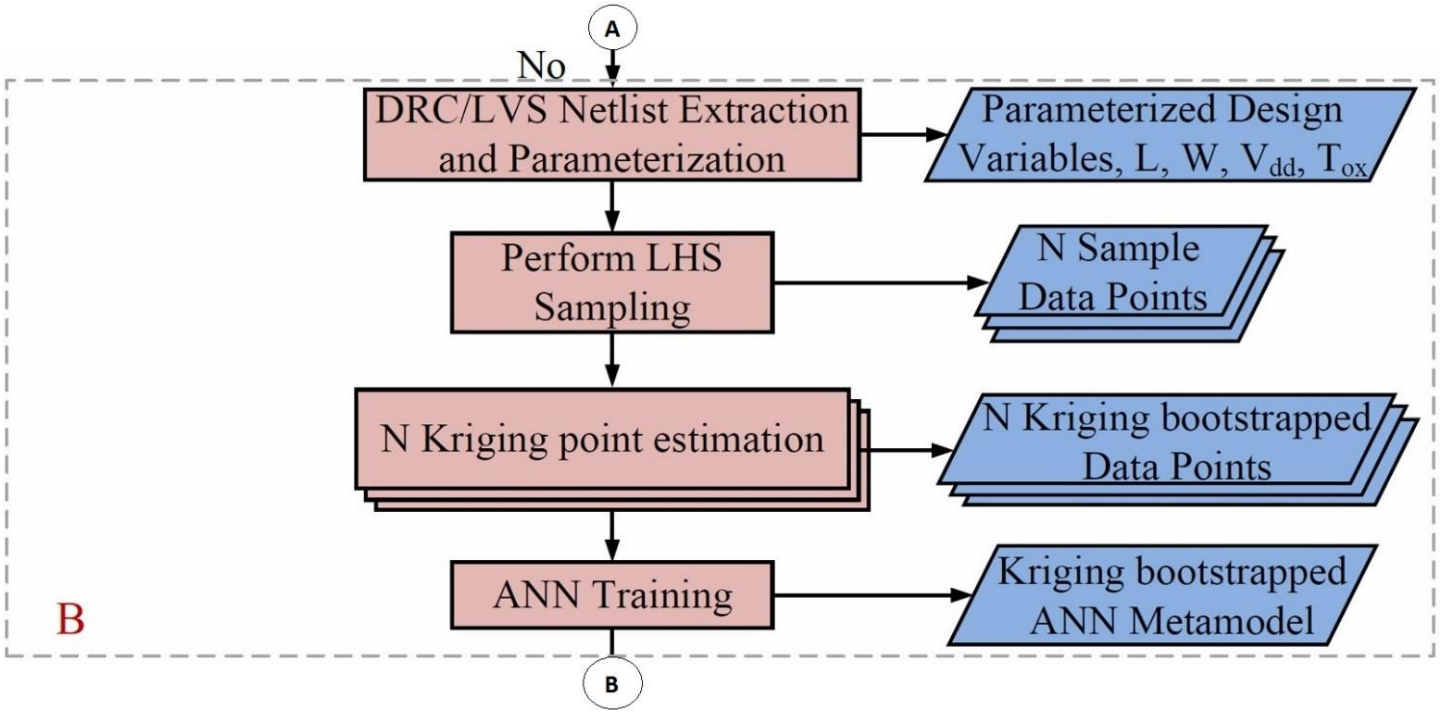
Proposed Design Optimization Flow Part - A

- First phase consists of baseline logical and physical design
- Baseline is simulated for functional verification
- This verification also serves to characterize circuit design objectives



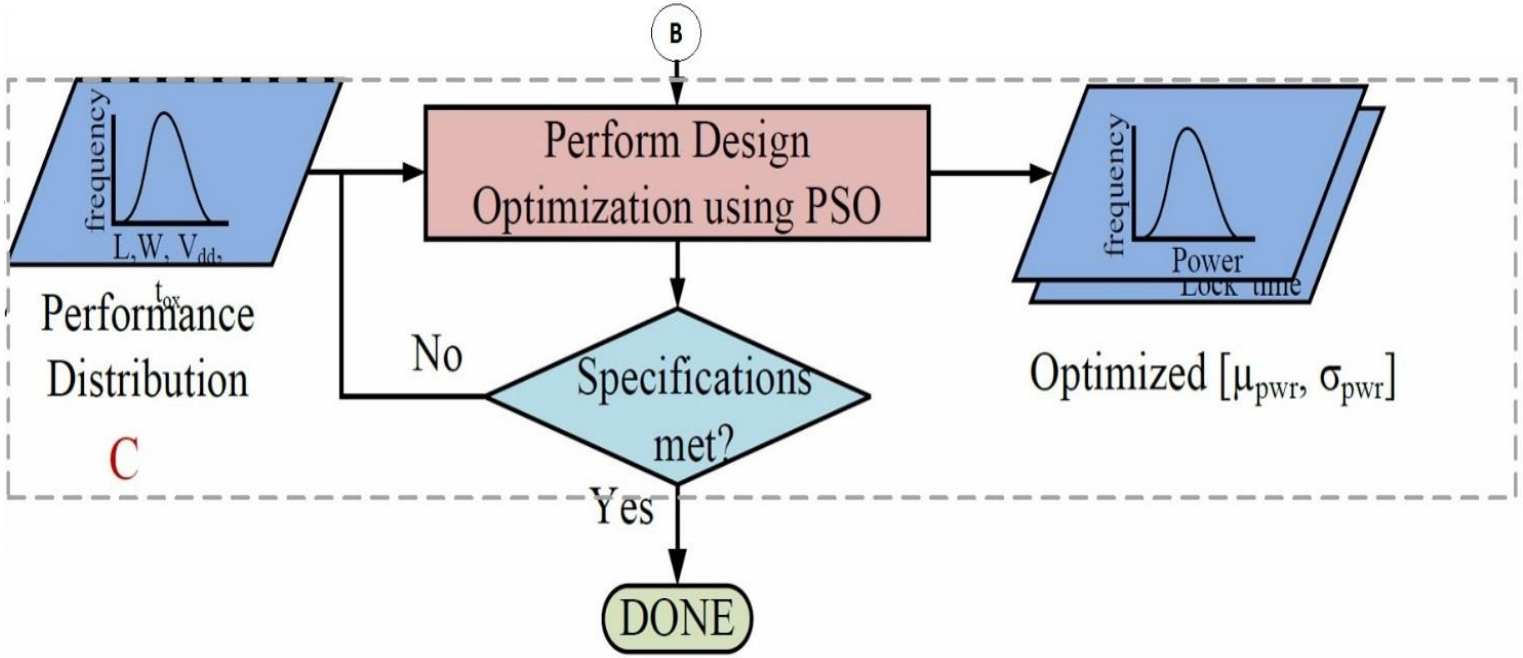
Proposed Design Optimization Flow Part - B

- This phase involves creation of process variation aware metamodel of the circuit design



Proposed Design Optimization Flow Part - C

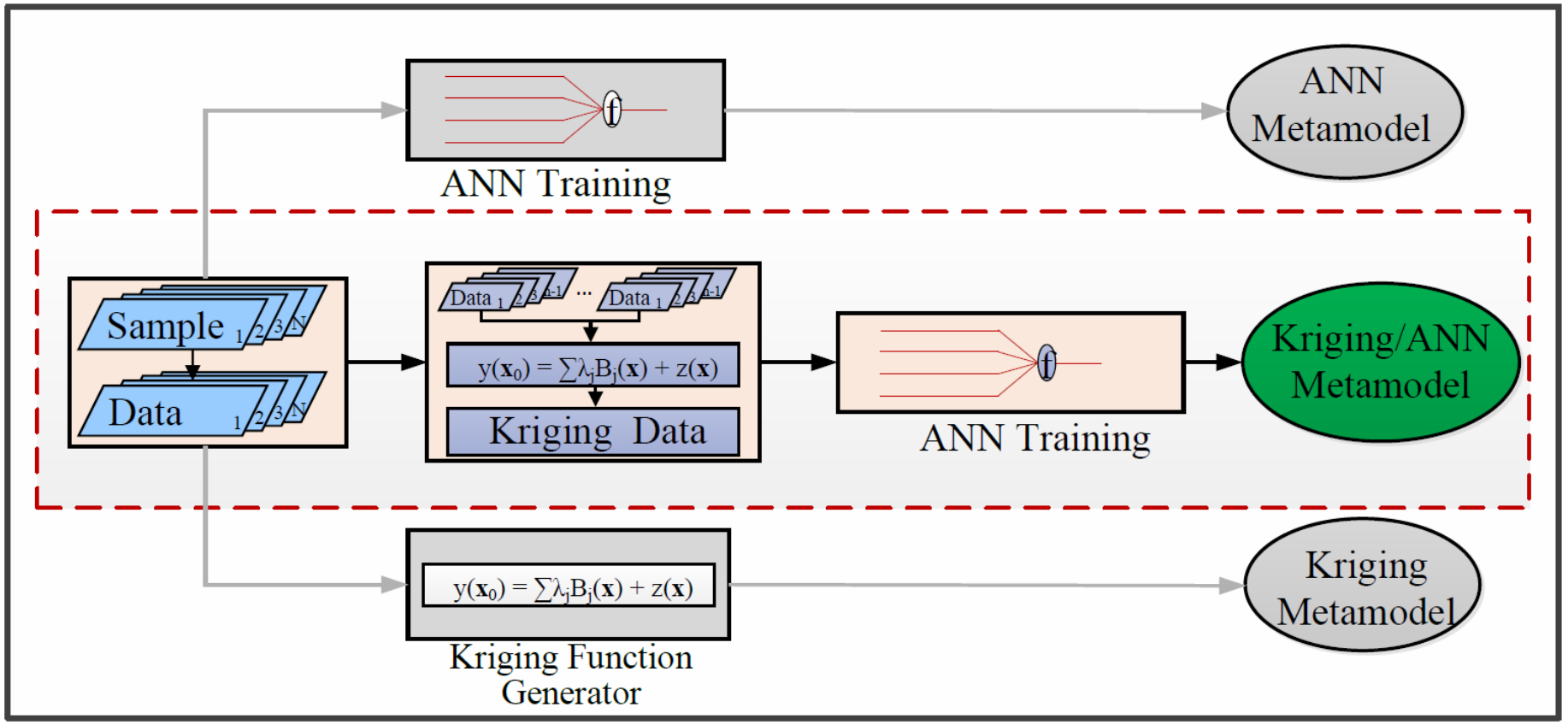
- This phase is process aware design optimization
- The optimization algorithm is used with the created metamodel and design objectives to optimize the design



Proposed Kriging based NN metamodel generation

- Kriging takes into account correlations between the input parameters in performance point prediction
- Hence it is very appealing and lends to high accuracy
- Use of Neural Networks can generate ultra-fast and accurate metamodels
- To ensure time efficiency and accuracy, we present a model that combines Kriging accuracy with NN speed

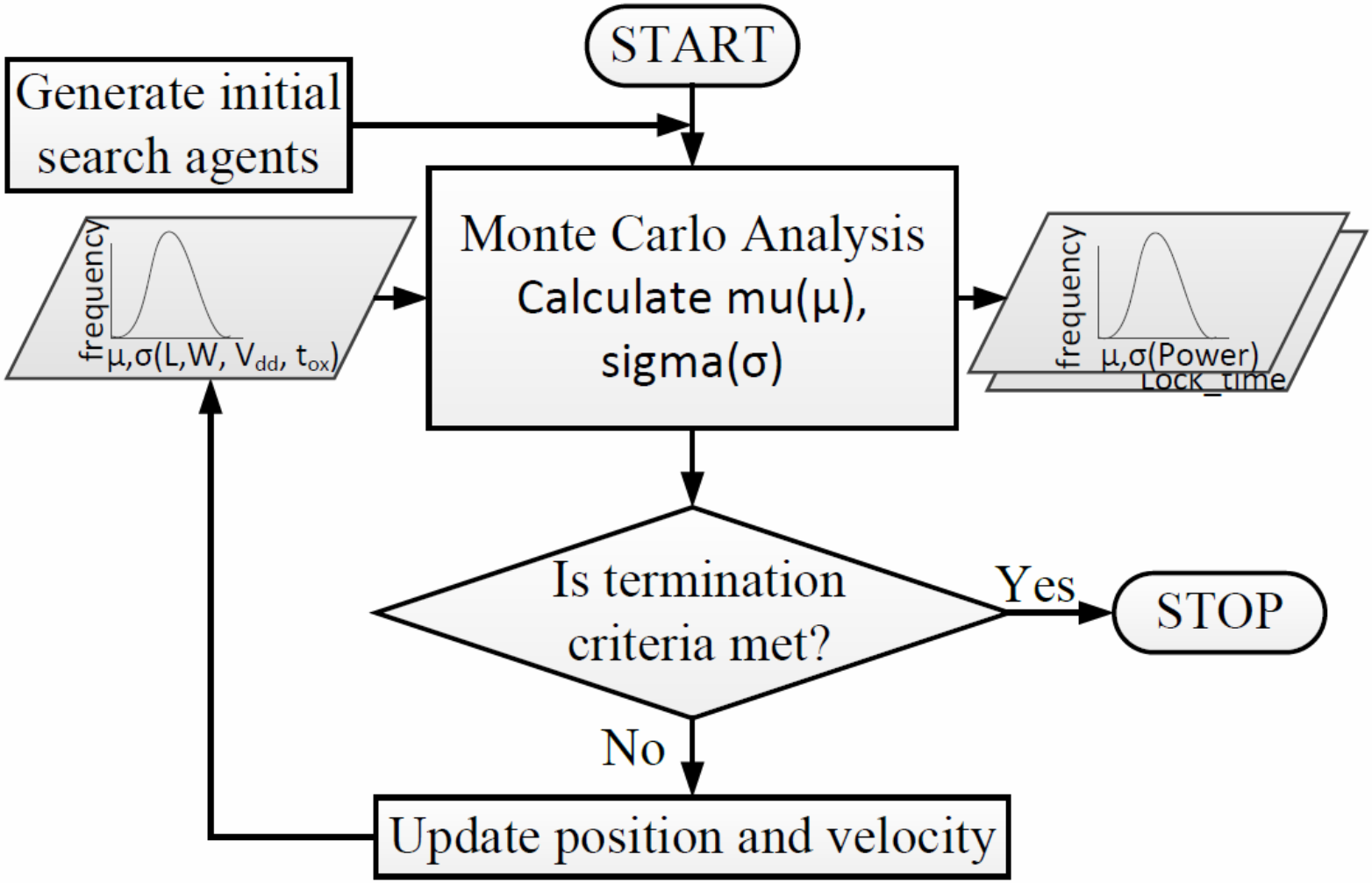
Proposed Kriging based NN metamodel generation flow



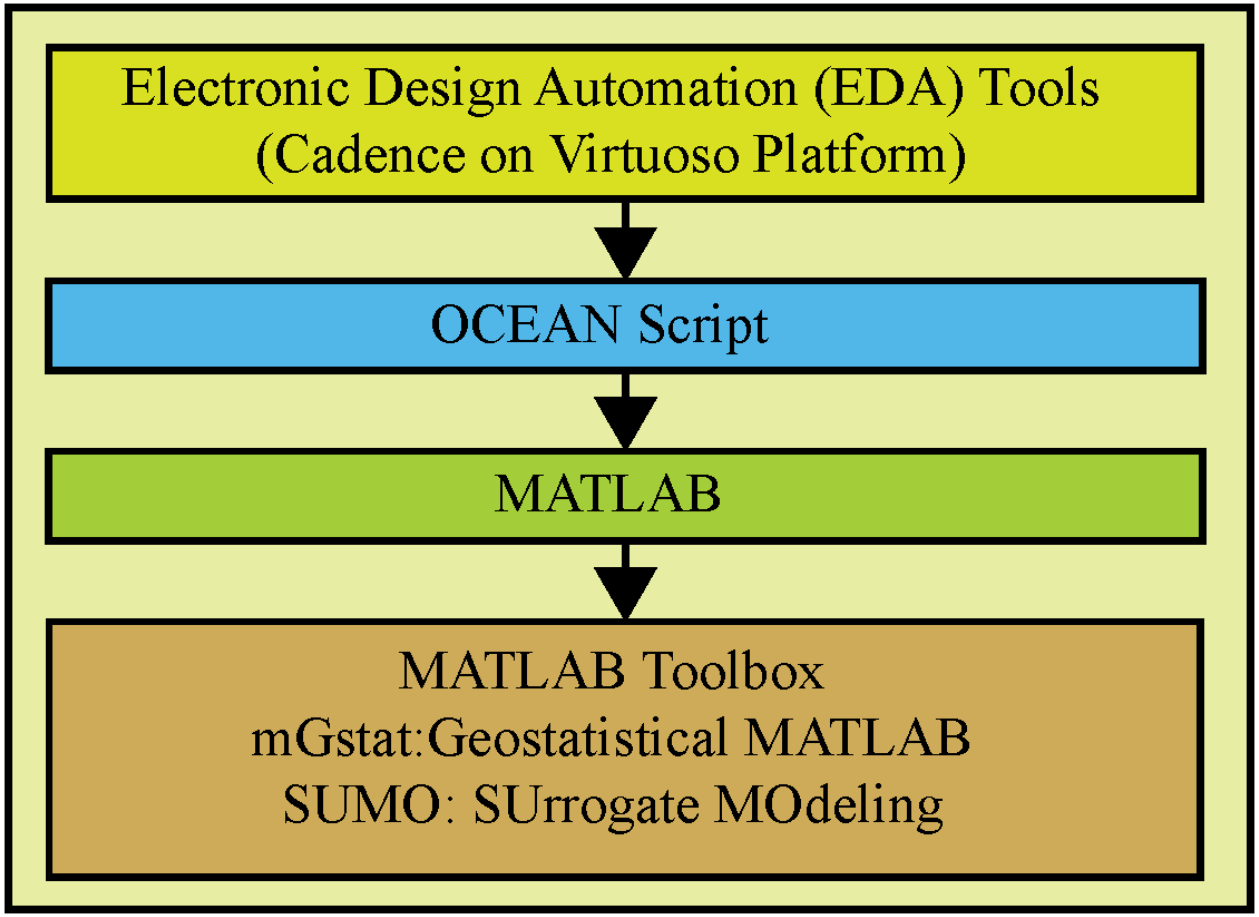
Particle Swarm Optimization Algorithm

- PSO is a type of evolutionary swarm intelligence algorithm for numerical optimization problems.
- The optimization problem implemented is to minimize power consumption of PLL circuit
- Process aware optimization involves minimizing the mean and standard deviation of optimal power consumption

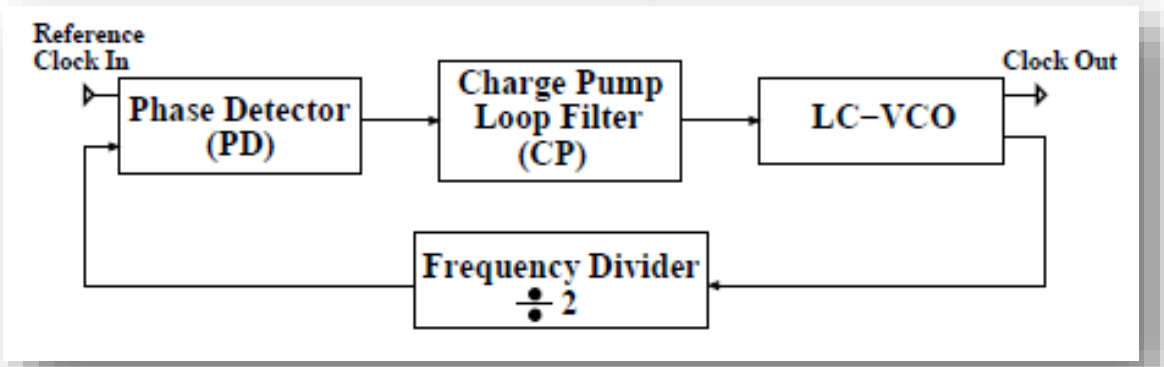
Flow Diagram for PSO Algorithm



Experimental Steps

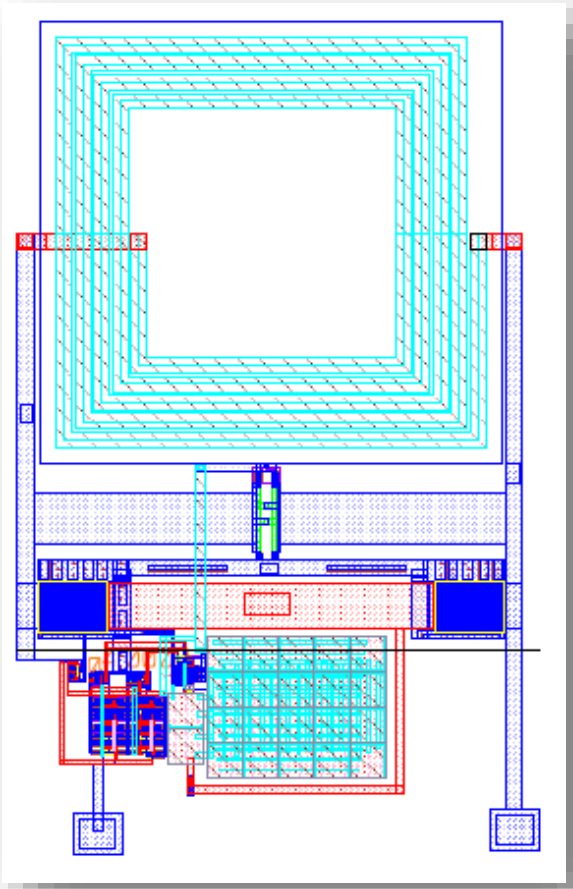


Case Study Circuit: 180nm PLL



Block diagram of a PLL.

- PLL circuit is characterized for frequency, power, vertical and horizontal jitter (for simple phase noise), and locking time.
- Metamodels are created for each FoM from the same sample set.
- 21 design parameters used.



PLL for 180nm.

Simulation Results

		SPICE Netlist	Kriging-ANN Metamodel			
			Before Optimization		After Optimization	
		Value	Value	Error (%)	Value	Error (%)
Power (P_{PLL})	Mean(μ)	2.48 mW	2.40 mW	3.33	2.35mW	2.08
	St.Dev.(σ)	0.42 mW	0.34 mW	19.05	0.39mW	7.14
Frequency (F_{PLL})	Mean(μ)	2.66 GHz	2.51 GHz	5.64	2.78GHz	4.51
	St.Dev.(σ)	10.95 MHz	41.93MHz	282.92	16.92MHz	54.52
Locking Time (Lck_{PLL})	Mean(μ)	5.51 μs	5.11 μs	7.26	5.21 μs	5.44
	St.Dev.(σ)	0.72 μs	0.44 μs	38.88	0.42 μs	41.67
Jitter (J_{PLL})	Mean(μ)	16.80 ns	14.69 ns	10.25	17.72ns	5.47
	St.Dev.(σ)	1.32 ps	4.50 ps	240.91	0.33ps	75

Conclusions

- Presented a statistical optimization design flow combining Kriging and Neural Network based metamodeling with PSO based algorithm
- The design technique was illustrated through a PLL circuit
- Research will expand for design optimization of various analog blocks in future


Thank you !!!