ISCAS 2025
IEEE INTERNATIONAL SYMPOSIUM ON CIRCUITS AND SYSTEMS
LONDON, UK

Emerging ML-AI Techniques for Analog EDA

Prof. Ioannis Savidis

ISCAS 2025 Tutorial
05-25-2025

Drexel UNIVERSITY

IDEA Design & Analysis Lab

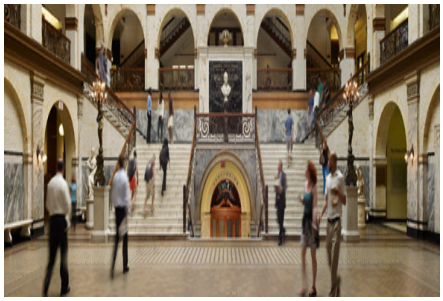
1

Drexel University – Electrical & Computer Engineering

2

Drexel University:

- **Founded** in 1891 by financier and philanthropist Anthony J. Drexel
- **Location:** four campuses: 3 in Philadelphia, 1 in New Jersey (Mt. Laurel)
- **Student Enrollment:** 15,346 undergraduates 8,859 graduate and professional students
- **Student Geographic Distribution:** Students come from 50 U.S. states and 130 foreign countries. Nearly 8% are international students



Drexel UNIVERSITY

2

DREXEL UNIVERSITY - IOANNIS SAVIDIS GROUP

Degrees: B.S.E., Duke University
M.S., University of Rochester
Ph.D., University of Rochester (2013)

Research Interests

Analysis, modeling, and design methodologies for high performance digital and mixed-signal integrated circuits; Emerging integrated circuit technologies; Electrical and thermal modeling and characterization, signal and power integrity, and power and clock delivery for 3-D IC technologies; hardware security (obfuscation and side-channel analysis); algorithms and methodologies for design automation including ML/AI based optimization; On-chip power management; Low-power circuit techniques; Algorithms and methodologies for secure IC design

LABORATORY & TEAM

- **Seven Ph.D. students**
  - Alec Aversa – Sequential digital circuit obfuscation
  - Saran Phatharodom – Digital obfuscation metrics
  - Jeff Wu – Application of ML/AI to analog/RF IC design
  - Ziyi Chen – Analog IP protection
  - Ashish Sharma – Heterogeneous circuit integration
  - Pratik Shrestha – Digital security and application of ML/AI to digital IC design
  - Nnaemeka Achebe – Application of ML to RF design
  - Amit Varde – EDA foundational modeling, ML based analog design
- **2,000 square feet of dedicated research space**
- **Access to leading CAD software packages:** Cadence (Virtuoso, Encounter, assure), Synopsys (Primetime, Hspice, Taurus), and Siemens Mentor Graphics (Calibre)

Drexel UNIVERSITY

3

ICE Research: Heterogeneous & Reconfigurable Integration for Edge Compute

Heterogeneous Chiplet and 3-D Integration

- 3-D/VLSI design methodologies for power and clock network design
- Multi-plane power noise modeling
  - Methodologies to mitigate cross-plane coupling
- Power management for multi-domain, multi-plane delivery
- Clock tree synthesis for heterogeneous device planes
- Multi-physics modeling of electro-thermal-mechanical characteristics of 3-D ICs
- Test vehicle design and characterization (three fabricated and tested ICs)

Power Management

-Near-threshold circuits (NTC) for low-power applications
  - Implement circuit families including CMOS and current mode logic in NTC
-Leakage reuse for multi-voltage domain systems
-Energy efficient heterogeneous DNN accelerators
-Power management for multi-domain delivery

ML/AI Circuit Design

- Applied algorithms for clock tree synthesis
- ML/AI algorithms for analog transistor sizing
  - Classification with adaptive labeling
- SMT based optimization of transistor sizing
- Graph based representation of circuit netlist
- EDA-Learn: framework for circuit data generation and ML modeling

Hardware Security & Trust

- Real time Trojan detection
- FPGA security and IP protection
  - Detection of hardware Trojans
  - Real time side-channel monitoring
- Attack prevention with trusted design
  - Metrics to quantify security
  - Algorithms and methodologies to obfuscate digital circuits
- Protection of analog circuits
  - Circuit redaction using field programmable analog arrays
  - Key-based parameter obfuscation
- Heterogeneous IC security (chiplet/3D)
  - Trojan detection
  - Side-channel analysis
  - Secure multi-plane communication

Objective:
Secure Heterogeneously Integrated Circuits and Systems for Edge Compute

4

Outline of Presentation

- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

6

Outline of Presentation

- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
- Analog Automation Platforms
- Conclusions

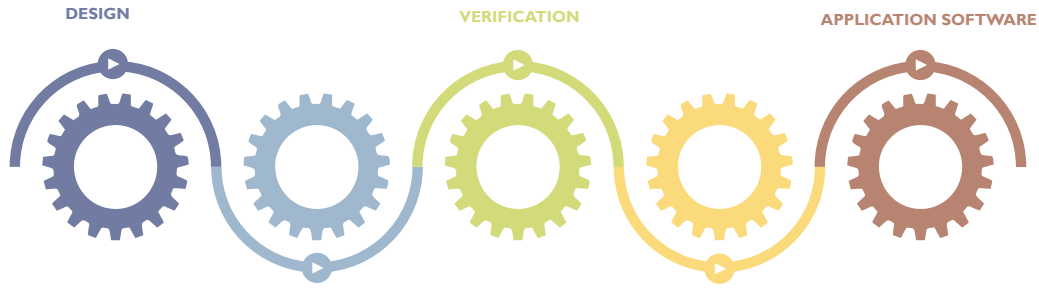
Drexel UNIVERSITY

7

Background: Electronic Design Automation (EDA)

8

- EDA: a system of software solutions for the design of integrated circuits
- A wide range of applications:
  - High-performance Computing
  - Autonomous vehicle
  - IoT
  - AI
  - …
- Primary Tools/Applications:

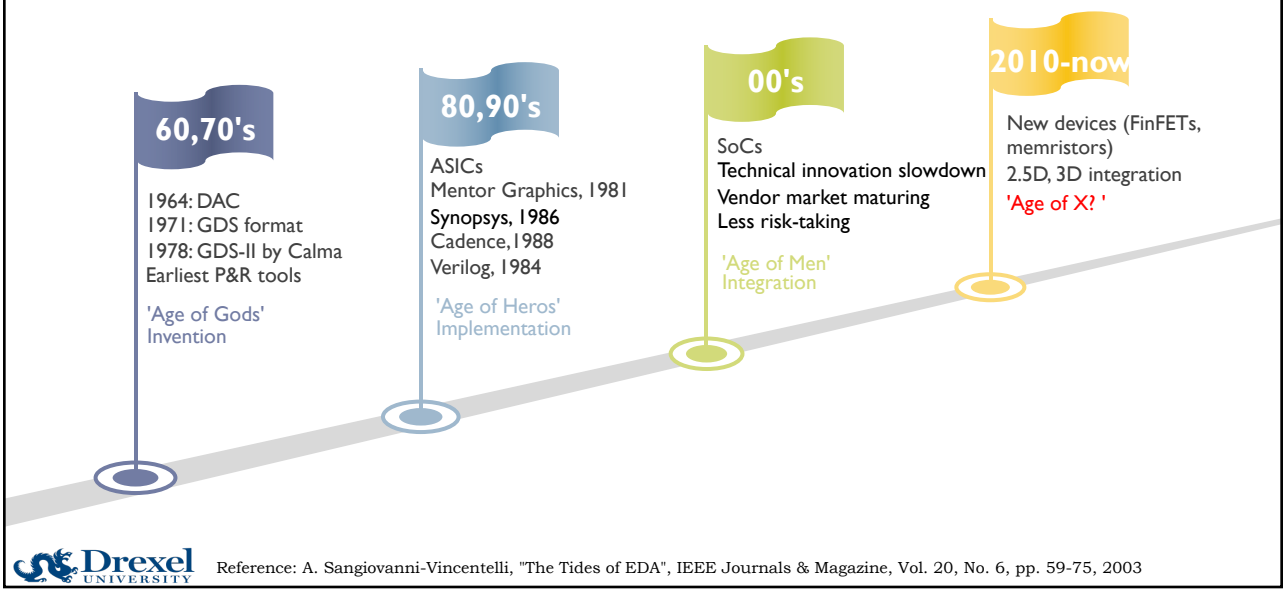


Drexel UNIVERSITY

8

A Remembrance of the Past: Timeline of (Digital) EDA Development

9



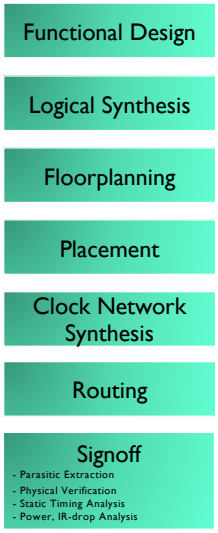
Drexel UNIVERSITY

Reference: A. Sangiovanni-Vincentelli, "The Tides of EDA", IEEE Journals & Magazine, Vol. 20, No. 6, pp. 59-75, 2003

9

Overview of Digital EDA

- Mature RTL-to-GDSII flow
  - Frontend: technology independent standardized design descriptions
    - Examples: VHDL, Verilog
  - Backend: physical implementation of circuits
    - Fabs provide libraries and simulation models for fab processes
- The high level of automation achieved in digital EDA results from abstraction
  - Programmability at high level
  - 'Divide and conquer' design strategy
  - Modular design for reusability



Analog IC Synthesis

- Analog circuits are highly customized with various functionalities

Amplifiers
- Opamp
- OTA

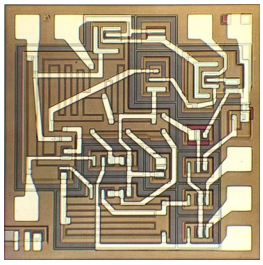
RF Transceivers
- LNA
- VCO
- Mixers
- PLL
- DLL
- PA

Data Converters
- ADC
- DAC

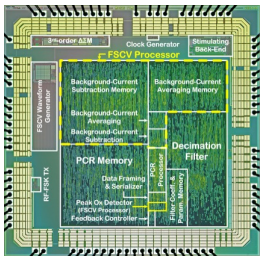
Filters

Reference Generators
- Crystal oscillator
- Bandgap references
- Biasing structures
- Clock generators and drivers

- Increased circuit complexity with next-generation target application domains
  - Bio-inspired computing
  - 6G communications
  - Autonomous vehicles (cars, UAVs, etc.)
  - …



µA709 operational amplifier by Fairchild, 1965

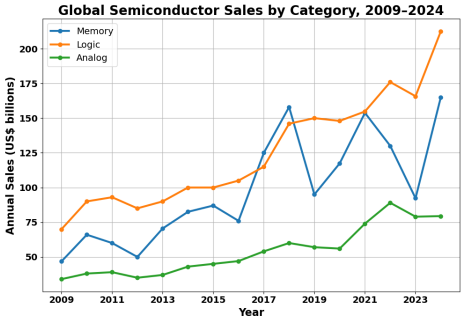


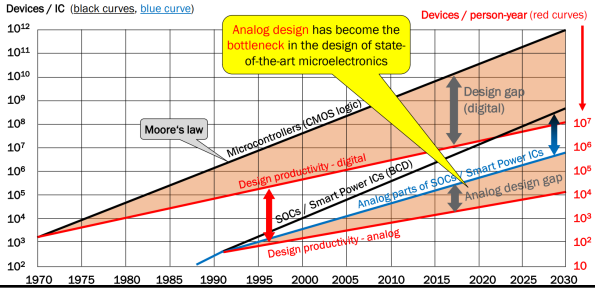
SoC for Sensing of Brain Neurochemistry, 2015

Reference: B. Bozorgzadeh, "Integrated Microsystems for High-Fidelity Sensing and Manipulation of Brain Neurochemistry", 2015

Background: Analog IC Design and Production

- Analog segment had the largest growth in sales during Covid (2020-2022)
  - Slows after Gen-AI wave compared with other segments (after 2023)
- Analog design productivity lags behind digital design by orders of magnitude
  - Design productivity = number of devices integrated on a chip / required design effort in person-years
    - Results in long time to market
    - Significant design effort and experience required





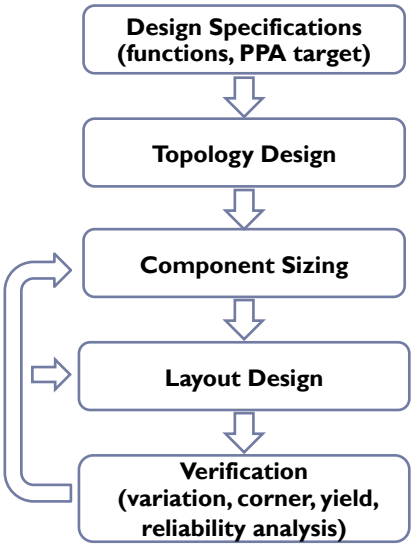
Reference: 1. Semiconductor Industry Association
2. J. Scheible, "Optimized is Not Always Optimal," Proceedings of the International Symposium on Physical Design, pp. 151–158, 2022

Drexel UNIVERSITY

12

Typical Analog Synthesis Flow

- Top-down tasks
  - Specification formulation
  - Topology selection
  - Component sizing
    - Tuning the sizes of the devices in an analog circuit to meet design specifications
- Bottom-up tasks of layout design
  - Placement
    - Device grouping based on matching and symmetry
    - Determine coordinates of pins and devices in layout
  - Routing
    - Optimize routing under design constraints, e.g., area, design rules, matching
- Additional tasks:
  - Interconnect impedance prediction
    - Reduce the gap between schematic simulation results and post-layout results
  - Compensation for effects of PVT variations and reliability issues

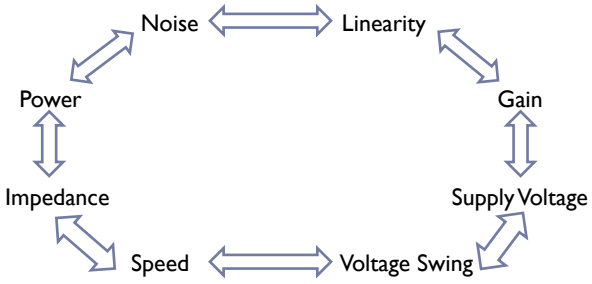


Drexel UNIVERSITY

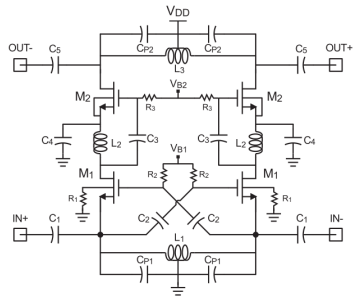
13

Unique Challenges to Automate Analog Design

- Analog circuits are highly customized with various functionalities
  - Design considerations beyond PPA
- Complex and non-linear circuit behaviors resulting from physics
  - Design, performance, and process parameters all interrelated



Typical considerations for op-amp design



$$F = 1 + \frac{R_l}{Rs} + \frac{R_g}{R_s} + \gamma X g_d Rs \left(\frac{\omega_0}{\omega_T}\right)^2$$

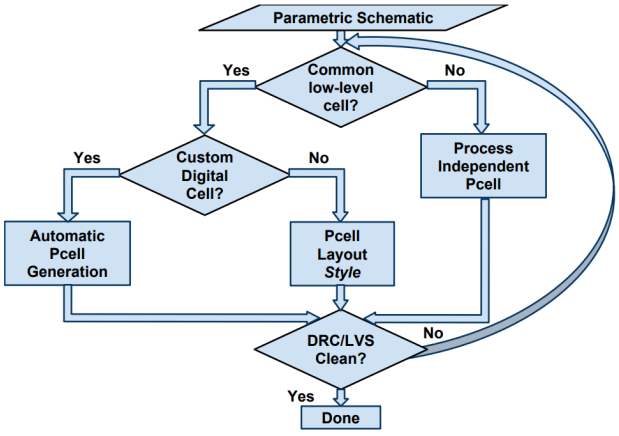
Noise factor of a low-noise amplifier

Reference: B. Razavi, "Design of Analog CMOS Integrated Circuits", McGraw-Hill, 2001

14

Does Digital Idea of Abstraction Apply to Analog EDA?

- Abstraction levels from low to high: device level ⇨ sub-block level ⇨ system level
- Past attempt: template-based 'standard' cells of analog blocks (amplifiers, comparators..)
  - Berkeley Analog Generator
  - BAG follows pre-defined design procedures
  - Synopsys' Pycells as technology-independent parameterized layout cells
  - Limitation:
    - Procedural design processes are inflexible
      - Require laborious setup and input by designers
    - Solutions are suboptimal

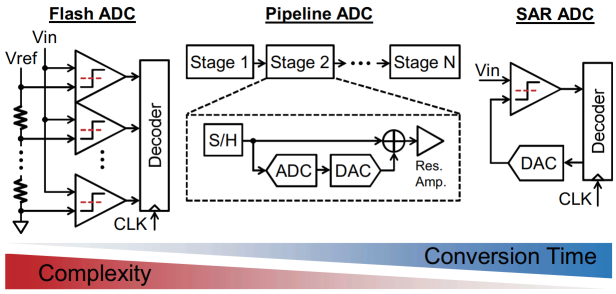


Reference: J. Crossley, et. al., "BAG: A Designer-oriented Integrated Framework For The Development Of AMS Circuit Generators," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp. 74–81, Nov. 2013

15

Digitize AMS Circuits?

- Digitally-assisted AMS design
  - Examples
    - Calibration of nonlinearity of ADCs
    - Digital predistortion and noise shaping for DACs
- Mostly-digital architectures
  - Examples:
    - Data-converter-based transmitter and receiver
    - DPLL
    - DLDO
- Digital-like AMS operations
  - 5 GS/s time-to-digital (TDC) based on inverters and flip-flops only
  - Filters, amplifiers implemented with digital standard cells
- Advantage: auto synthesized with digital EDA flows
- Limitation: does not apply to analog and RF circuits with high performance requirements
- Instead of standardizing analog blocks or making analog circuits digital, develop automated design toolboxes that allow customization

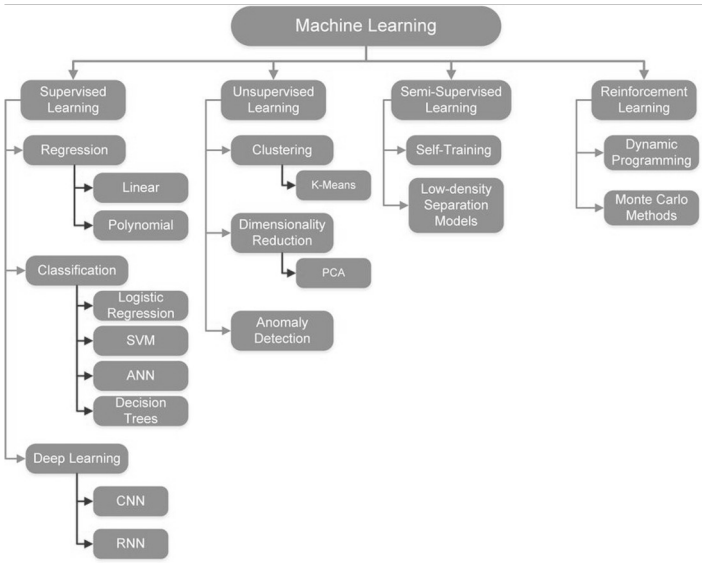


Reference: S. Su, et al., "Analog/Mixed-Signal Circuit Synthesis Enabled by the Advancements of Circuit Architectures and Machine Learning Algorithms," Proceedings of the Asia and South Pacific Design Automation Conference, pp.100–107, 2022

16

Introduction of Machine Learning Algorithms

- Machine learning: train models to learn from data
- Leverage information from data to improve performance on prediction and generation tasks
- Diverse algorithm choices

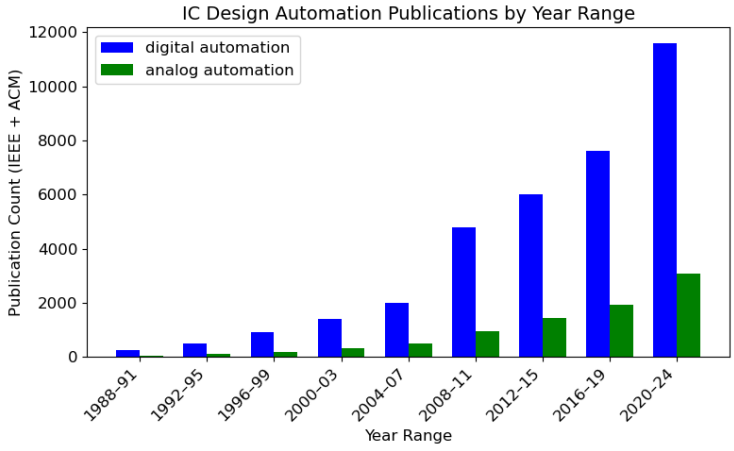


Reference: A. Moubayed, M. Injadat, A. B. Nassif, H. Lutfiyya and A. Shami, "E-Learning: Challenges and Research Opportunities Using Machine Learning & Data Analytics," IEEE Access, Vol. 6, No. 1, pp. 39117-39138, 2018

17

Comparison of IEEE Publication Count for Digital and Analog Automation

- "Explosion of digital automation": 2008-2012, 2018-Now
- Interests in analog design automation steadily rise



Drexel UNIVERSITY

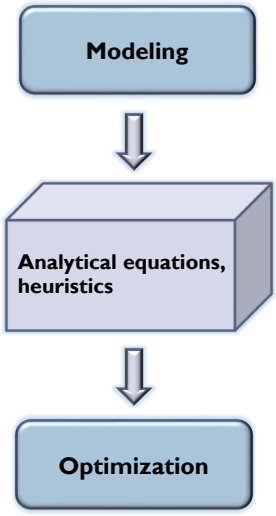
18

Recap: Heuristic Approaches for Circuit Design and Synthesis

- Circuit design is primarily performed in two steps:
  - **Modeling** with analytical equations and heuristics
    - Design variables: what we can control
    - Design targets: what we want

$$F = 1 + \frac{R_l}{Rs} + \frac{R_g}{R_s} + \gamma X g_d Rs \left(\frac{\omega_0}{\omega_T}\right)^2$$

  - **Optimization**
    - Tune design parameters manually
      - "Art": depends on designer insight
      - Limitation: local tuning = sub-optimal
    - Formulate optimization problems
      - Objective of optimization: FoM
      - Constraints set by design specifications and process
      - Limitation: analytical functions different from true design space due to circuit non-linearity, simplifications of models
        - Deviate from simulation results

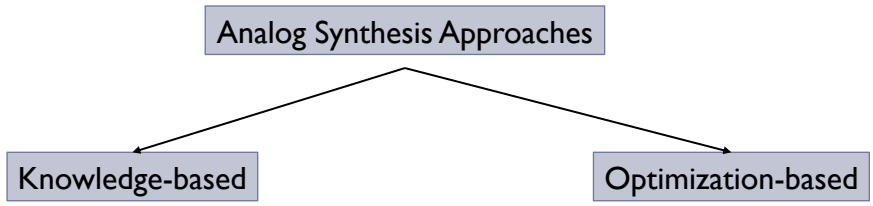


Drexel UNIVERSITY

19

Overview of Heuristic Approaches for Analog Synthesis

- Design with analytical equations or rules set by human designers
- Top-down hierarchical design flow



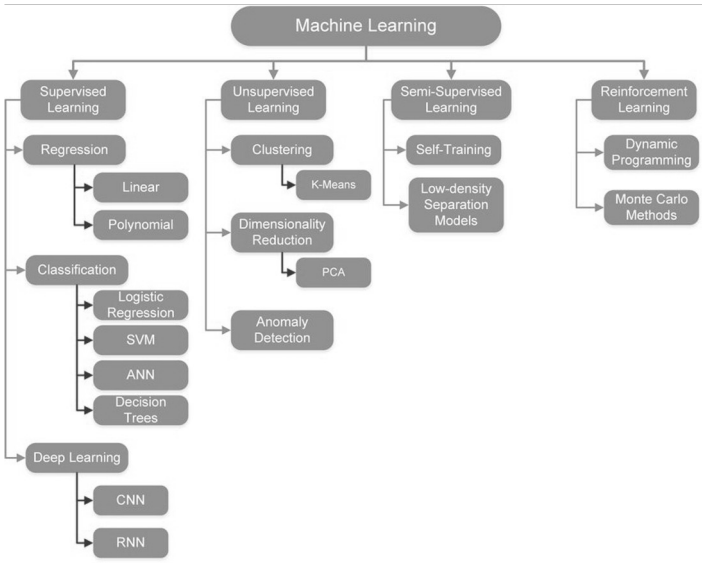
- Procedural generation of design
  - Rule-based
  - Template-based
- Optimization algorithms handle designer-provided specifications

Outline of Presentation

- Background Introduction
- Machine Learning Techniques for Analog EDA
  - Background on machine learning
  - Statistical learning algorithms
  - Neural-network-based learning algorithms
  - Transfer learning
- Optimization Techniques for Analog EDA
- Circuit Applications
- Analog Automation Platforms
- Conclusions

Potential Solution for Analog EDA: Machine Learning

- Machine learning: train models to learn from data
- Leverage information from data to improve performance on prediction and generation tasks
- Diverse algorithm choices

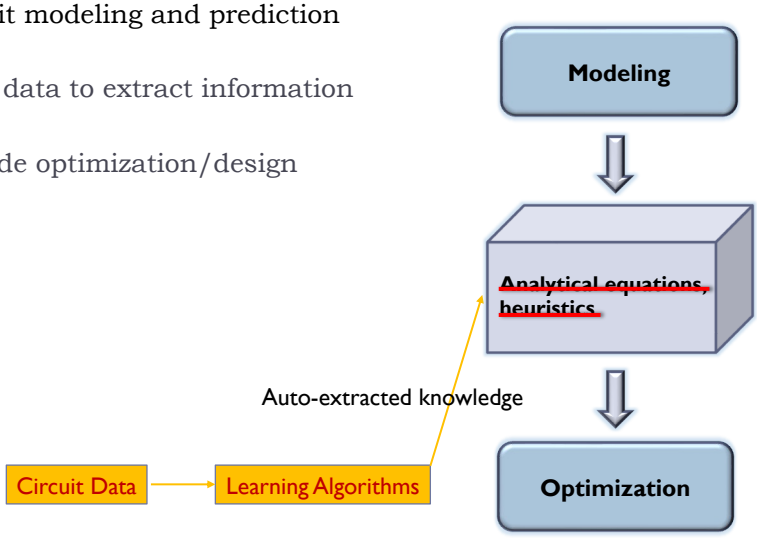


A. Moubayed, M. Injadat, A. B. Nassif, H. Lutfiyya and A. Shami, "E-Learning: Challenges and Research Opportunities Using Machine Learning & Data Analytics," IEEE Access, Vol. 6, No. 1, pp. 39117-39138, 2018

22

Machine Learning for EDA

- Approach of applying ML for circuit modeling and prediction
  - Train learning models on circuit data to extract information
  - Use extracted information to guide optimization/design



23

Machine Learning for Analog EDA

- Unique advantages brought by ML for EDA problems
  - Design space exploration
    - Extract patterns of circuit parameters/characteristics from circuit data
  - Predict performance metrics or physical characteristics of a circuit through the design stages
    - Early warning/skip doomed runs
      - Simulation time
  - Guide optimization for design generation
- ML for analog EDA in two key steps:

Learning
ML algorithms for circuit modeling

Optimization
Optimization algorithms for circuit generation

Drexel UNIVERSITY

24

Biggest Challenge of Applying ML to Analog EDA: Data

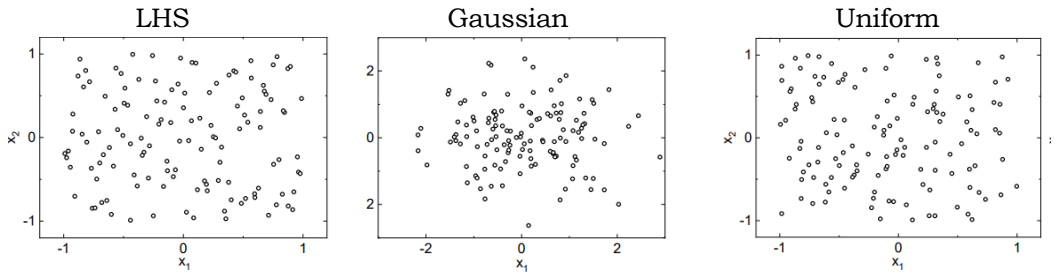
- Primary sources for circuit data:
  - Random sampling from the design space (design of experiments) with circuit simulators
    - Challenge:
      - Numerical circuit solvers are computationally costly
        - One execution run of system-level simulation may take hours or days
  - Expert designs generated by human designers or automation tools
    - Challenge:
      - Design IPs are often proprietary
      - Lack of benchmark circuits and standardized data format for analog EDA
- Vision/hope: open repository with production ready designs that are encrypted and secured but allow for ML design research by the community

Drexel UNIVERSITY

25

Sampling (Design of Experiments) for Circuit Data Generation

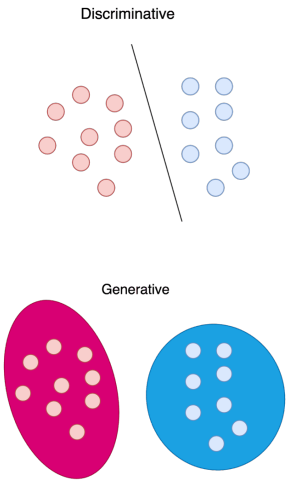
- Gaussian random sampling
- Uniform random sampling
- Latin hypercube sampling (LHS)
  - Divide design space into *M* equal intervals, sample from each interval
- Results show LHS performs best based on MSE and R-squared of models trained on moderate and large sample sizes



X. Shangguan, H. Ma, A. C. Cangellaris and X. Chen, "Effect of Sampling Method on the Regression Accuracy for a High-Speed Link Problem," Proceedings of the IEEE Conference on Electrical Performance of Electronic Packaging and Systems, pp.1-3, 2021

Data Generation for ML Model Training for Analog Circuits

- Two types of ML models for inputs x and label y
  - Discriminative models: learn mapping from x to y, i.e., $p(y|x)$
  - Generative: learn a distribution over data, i.e., $p(x, y)$
- Primary sources of circuit data
  - Random sampling from the design space (design of experiments)
    - Unbiased
    - Suitable for discriminative ML models
    - Inappropriate for generative models because it is equivalent to learning from 'randomness'
  - Past expert designs generated by human designers or prototype design automation tools
    - Biased
    - Suitable for generative ML models
    - Inappropriate for discriminative models because data is 'biased' towards good design



Source: theaisummer.com

Outline of Presentation

- Background Introduction
- Machine Learning Techniques for Analog EDA
  - Background on machine learning
  - Statistical learning algorithms
  - Neural-network-based learning algorithms
  - Transfer learning
- Optimization Techniques for Analog EDA
- Circuit Applications
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

28

Machine Learning for Analog EDA

- Unique advantages brought by ML for EDA problems
  - Design space exploration
    - Extract patterns of circuit parameters/characteristics from circuit data
  - Predict performance metrics or physical characteristics of a circuit through the design stages
    - Early warning/skip doomed runs
      - Simulation time
  - Guide optimization for design generation
- ML for analog EDA in two key steps:

Learning
ML algorithms for circuit modeling

Optimization
Optimization algorithms for circuit generation

Drexel UNIVERSITY

29

Statistical Learning Algorithms

- Statistical algorithms are predecessors of learning algorithms
- Algorithms widely utilized in EDA for decades:
  - Linear regression models
  - Gaussian process models
  - K-nearest neighbors
  - Support vector machines
  - Tree-based models

Drexel UNIVERSITY

30

Statistical Learning Algorithms

- Statistical algorithms are predecessors of learning algorithms
- Algorithms widely utilized in EDA for decades:
  - Linear regression models
  - Gaussian process models
  - K-nearest neighbors
  - Support vector machines
  - Tree-based models

Drexel UNIVERSITY

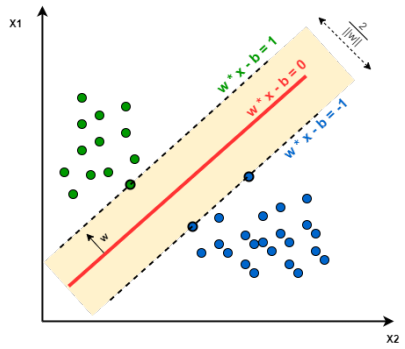
31

Linear Regression and Support Vector Machines

- For data D= { x, y }, a linear regression model is:

$$y_i = \sum \omega_j x_{ij} + N(0, \sigma^2)$$

- Support vector machine:
  - Decide a boundary with maximum distance from nearest points in each class



Source: baeldung.com

Statistical Learning Algorithms

- Statistical algorithms are predecessors of learning algorithms
- Algorithms widely utilized in EDA for decades:
  - Linear regression models
  - Gaussian process models
  - K-nearest neighbors
  - Support vector machines
  - Tree-based models

K-Nearest Neighbor Algorithm

- Classify based on distance between new data point and k nearest known data points
  - Distance metrics:
    - Euclidean distance
    - Manhattan distance
    - Hamming distance
    - ...
- Requires storage of all data
  - 'Lazy learning'
- Advantages:
  - Easy to implement and adapt
  - Few hyperparameters
- Disadvantages:
  - Does not scale well to large dataset with high dimensionality
    - Bottleneck in memory

Source: IBM.com

Statistical Learning Algortihms

- Statistical algorithms are predecessors of learning algorithms
- Algorithms widely utilized in EDA for decades:
  - Linear regression models
  - Gaussian process models
  - K-nearest neighbors
  - Support vector machines
  - Tree-based models

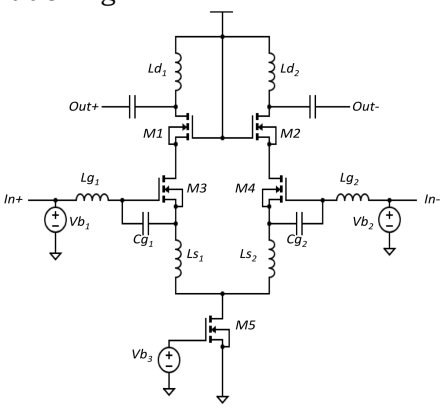
Tree-based Models

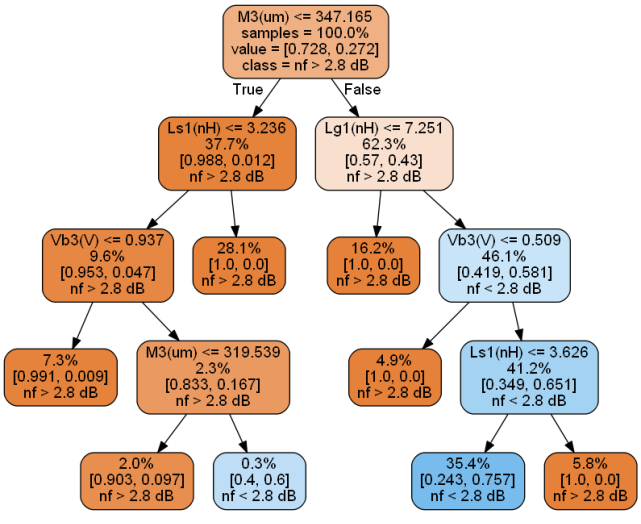
- Decision tree
  - Trained by maximizing information gain $\sum -p * \log_2 p_i$, where p is the probability of class i
- Ensemble of trees
  - Reduce overfitting resulting from a single tree
  - Random forest
    - For classification, return the class voted by most trees in ensemble
    - For regression, take the mean of predictions of all trees in ensemble
  - Gradient boost
    - Usually provides highest accuracy among tree-based models
- Advantages:
  - Data pre-processing not required
  - Fewer data required than neural networks
  - Interpretable tree structure provides additional design information
  - Allows ranking of feature importance

Drexel UNIVERSITY

Application 1: Decision Tree to Predict Noise Figure Based on Sizing

- Device sizes are generated with LHS
- Performance (NF) is evaluated with SPICE
- Tree structure shows design space partitioning



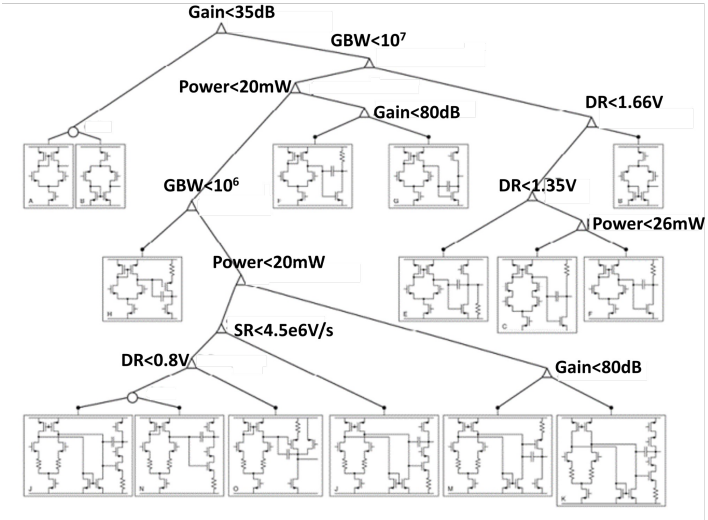


Drexel UNIVERSITY

Z. Wu and I. Savidis, "CALT: Classification with Adaptive Labeling Thresholds for Analog Circuit Sizing," Proceedings of the ACM/IEEE Workshop on Machine Learning for CAD, pp. 49–54, 2020

Application 2: Decision Tree to Map From Specifications To Topology

- Data is generated from MOJITO
  - MOJITO is an analog sizing platform that
    - Optimizes across thousands of analog circuit topologies
    - Returns a set of sized topologies with performance tradeoffs
- Trained tree model effectively provides topology selection rules for the target technology



[1] I. T. McConaghy, P. Palmers, G. Gielen and M. Steyaert, "Automated Extraction Of Expert Knowledge In Analog Topology Selection And Sizing," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp. 392-395,2008
[2] T. McConaghy, P. Palmers, G. Gielen and M. Steyaert, "Simultaneous Multi-Topology Multi-Objective Sizing Across Thousands of Analog Circuit Topologies," Proceedings of the ACM/IEEE Design Automation Conference, pp. 944-947, 2007

Drexel UNIVERSITY

38

Outline of Presentation

- Background Introduction
- Machine Learning Techniques for Analog EDA
  - Background on machine learning
  - Statistical learning algorithms
  - Neural-network-based learning algorithms
  - Transfer learning
- Optimization Techniques for Analog EDA
- Circuit Applications
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

39

Artificial Neural Networks (ANNs)

- Fundamental reason for success of AI in the recent decade:
  - Boost in algorithmic power: novel algorithms based on generalized neural network architecture
  - Boost in computing power (which again benefits from the development of IC, EDA)

- Deep neural networks
  - Depth is determined based on problem complexity and dataset size
  - Layers represent different levels of abstraction

Model Name	n_{params}	n_{layers}	d_{model}	n_{heads}	d_{head}	Batch Size	Learning Rate
GPT-3 Small	125M	12	768	12	64	0.5M	6.0×10^{-4}
GPT-3 Medium	350M	24	1024	16	64	0.5M	3.0×10^{-4}
GPT-3 Large	760M	24	1536	16	96	0.5M	2.5×10^{-4}
GPT-3 XL	1.3B	24	2048	24	128	1M	2.0×10^{-4}
GPT-3 2.7B	2.7B	32	2560	32	80	1M	1.6×10^{-4}
GPT-3 6.7B	6.7B	32	4096	32	128	2M	1.2×10^{-4}
GPT-3 13B	13.0B	40	5140	40	128	2M	1.0×10^{-4}
GPT-3 175B or "GPT-3"	175.0B	96	12288	96	128	3.2M	0.6×10^{-4}

Drexel UNIVERSITY

Reference: T, Brown, et. al, 'Language Models are Few-Shot Learners', Advances in Neural Information Processing Systems, Vol. 33, No. 1, pp.1877-1901, 2022

40

Learning Scenarios Powered by Variants of Neural Networks

Training scheme

- Supervised
- Semi-supervised
- Unsupervised
- Adversarial (GAN)
- Reinforcement
- Encoder-decoder

X

Data format

- Multi-layer perceptrons (feedforward neural networks)
  - Tabular data
- Convolutional neural networks
  - Image data
- Graph neural networks
  - Graph-structured data
- Recurrent neural networks
  - Time-series data

Combine any option from the left with any option from the right

Example: train a RNN in an adversarial approach

train a GNN in an unsupervised approach

Drexel UNIVERSITY

41

Learning Scenarios Powered by Variants of ANNs

Differentiate by training scheme

- Supervised
- Semi-supervised
- Unsupervised
- Adversarial (GAN)
- Reinforcement
- Encoder-decoder

X

Differentiate by data format

- Multi-layer perceptrons (feedforward neural networks)
- Convolutional neural networks
- Graph neural networks
- Recurrent neural networks

Apply specialized filters for the input format

Drexel UNIVERSITY

42

Learning Scenarios Powered by Variants of ANNs

Differentiate by training scheme

- Supervised
- Semi-supervised
- Unsupervised
- Adversarial (GAN)
- Reinforcement
- Encoder-decoder

X

Differentiate by data format

- Multi-layer perceptrons (feedforward neural networks)
- Convolutional neural networks
- Graph neural networks
- Recurrent neural networks

Drexel UNIVERSITY

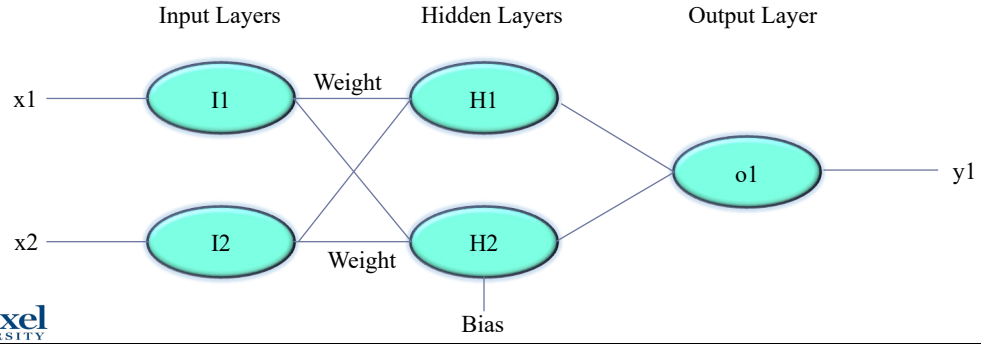
43

Multi-layer Perceptron (MLP) ANNs

- Each input neuron represents a piece of the data (image pixel, transistor feature…)
- Each neuron performs logistic regression:

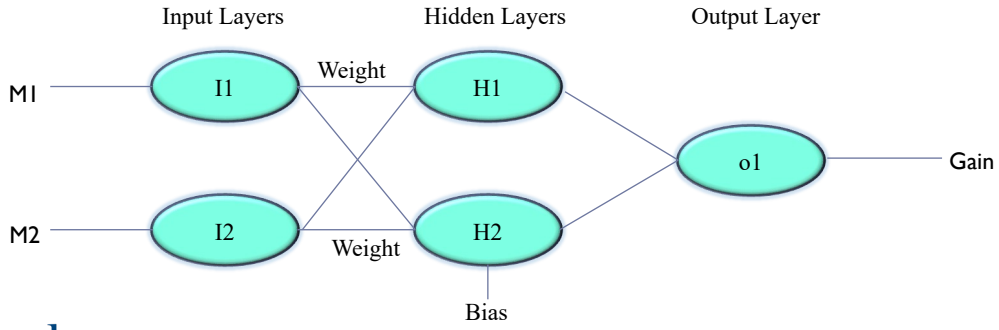
$$h_{w,b}(x) = f(\omega^T x + b)$$

- A neural network maps from input neurons to output labels
  - Works well for tabular data



Example: Problem Formulation For A Circuit Sizing Task

- Objective: predict AC gain of an amplifier based on sizes of two transistors
- Intuitive problem formulation:
  - Input features: two transistor sizes, M1, M2
  - Output label: real-valued SPICE evaluation of gain
  - After data generation, train a simple MLP to map from device sizes to gain values



Learning Scenarios Powered by Variants of ANNs

Differentiate by training scheme

- Supervised
- Semi-supervised
- Unsupervised
- Adversarial (GAN)
- Reinforcement
- Encoder-decoder

X

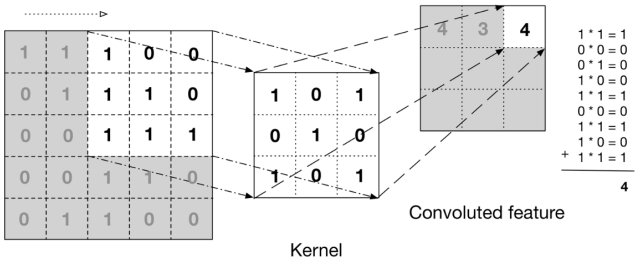
Differentiate by data format

- Multi-layer perceptrons (feedforward neural networks)
- Convolutional neural networks
- Graph neural networks
- Recurrent neural networks

Drexel UNIVERSITY

Image Data? Train Convolutional Neural Networks

- Circuit data in image format:
  - Physical design
    - Layout in GDSII
  - View of a gate netlist
  - Photos of fabricated PCBs
  - …
- Convolutional neural networks:
  - Each pixel is updated by the weighted sum of current pixel value and neighboring pixel values
  - Mathematically, apply convolution between kernel and pixel values



Source: analyticsvidhya.com

Drexel UNIVERSITY

Learning Scenarios Powered by Variants of ANNs

Differentiate by training scheme

- Supervised
- Semi-supervised
- Unsupervised
- Adversarial (GAN)
- Reinforcement
- Encoder-decoder

X

Differentiate by data format

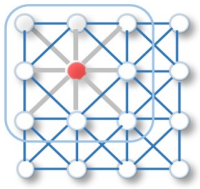
- Multi-layer perceptrons (feedforward neural networks)
- Convolutional neural networks
- Graph neural networks
- Recurrent neural networks

Drexel UNIVERSITY

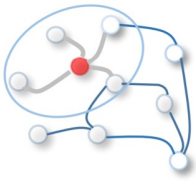
48

Graph Neural Networks

- Convolutional neural networks
  - Each pixel is a node
  - An image is a regular-shaped grid
  - Take a weighted average of node features (pixel values) along with neighboring node features



- Graph neural networks
  - Graph = (Vertices,Edges)
  - A graph is often irregular shaped
  - Objective is to combine features of the current node and neighboring node features
    - Similar to CNNs



GNNs are generalized versions of CNNs

Circuit applications: learn from numerical features of a circuit in a tabular approach, learn based on circuit graphs with numerical features associated with devices or entire circuits

Drexel UNIVERSITY

Reference: Wu, Zonghan et al. "A Comprehensive Survey on Graph Neural Networks," *IEEE Transactions on Neural Networks and Learning Systems*, Vol. 32, No. 1, pp. 4-24, 2021

49

Vanilla Graph Convolutional Neural Networks (GCN)

Three primary steps:

1) Generate embeddings h_i for each graph node i
   - Map original feature vectors to vectors of a fixed dimension via linear neural network layers

2) Aggregate embeddings of each node with embeddings of the neighboring nodes

$$h_i^{(l+1)} = \sigma\left(b^{(l)} + \sum_{j \in N(i)} \frac{1}{c_{ij}} W^{(l)} h_j^{(l)}\right)$$

New embedding of node i; Bias vector; Nonlinear activation function; Index set of direct neighboring nodes of node i; Normalization coefficient; Weight vector; Current embedding of node j

3) Send final embeddings to linear neural network layers for target predictions

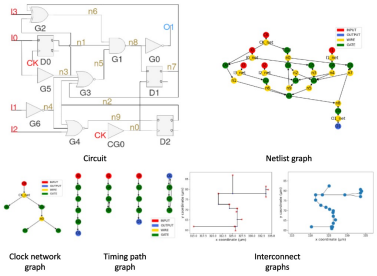
Node embedding layers ⇒ Aggregation layers ⇒ Linear layers for prediction

Drexel UNIVERSITY

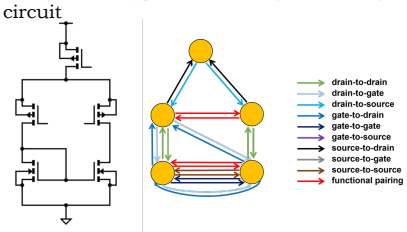
Heterogeneous Graphs to Represent Digital and Analog Circuits

- Heterogeneous node types to distinguish:
  - Analog circuits: transistors, passives…
  - Digital circuits: gates, pins…
  - An interconnect net can be either a graph node or a graph edge
    - When represented as an edge, equivalent to a hypergraph
- Heterogeneous edge types to distinguish:
  - Connections between different types of nodes
  - Signal flow
  - Connections between analog device terminals (drain, gate, source, bulk of transistors, capacitor terminals…)

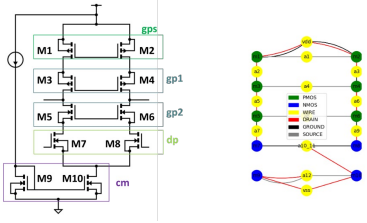
Example 1: heterogeneous node types for a digital circuit



Example 2: single node type with heterogeneous edge types for an analog circuit



Example 3: heterogeneous node types for an analog circuit

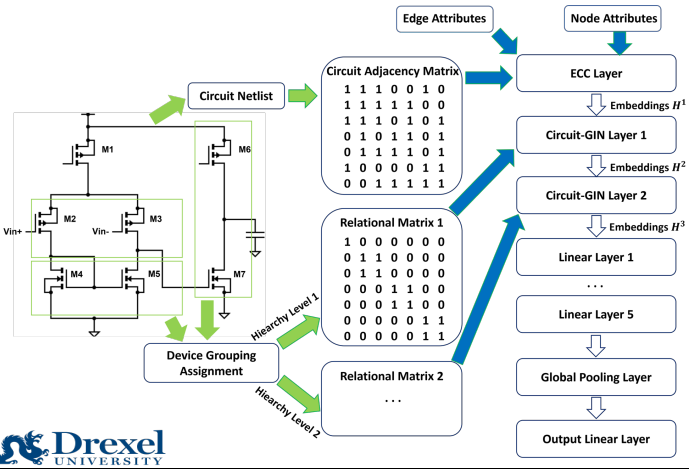


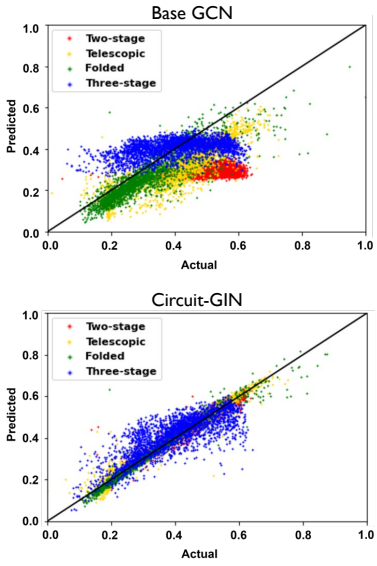
Drexel UNIVERSITY

H. Ren, G. F. Kokai, W. J. Turner and T. Ku. "ParaGraph: layout parasitics and device parameter prediction using graph neural networks," Proceedings of the ACM/EDAC/IEEE Design Automation Conference, pp. 1–6, 2020

Graph Convolution Based on Circuit Hierarchies

- Model consists of two ECC layers and a Circuit-GIN layer:
  - Circuit-GIN: GIN on a relational graph based on circuit hierarchy
- Circuit-GIN model outperforms base GCN model by distinguishing between four op-amp topology graphs by up to 16.7% in R-squared





Drexel UNIVERSITY

52

Choosing between CNN and GNN for Modeling of Analog Circuits

- Physical layout is usually better modeled with images
  - Geometric information included
    - Device coordinates
    - Pin coordinates
    - Instance orientations
    - ...
- Circuit topology is better represented with graphs
  - Graph isomorphism
  - Focus on topological connectivity information

Drexel UNIVERSITY

53

Learning Scenarios Powered by Variants of ANNs

Differentiate by training scheme

- Supervised
- Semi-supervised
- Unsupervised
- Adversial (GAN)
- Reinforcement
- Encoder-decoder

X

Differentiate by data format

- Multi-layer perceptrons (feedforward neural networks)
- Convolutional neural networks
- Graph neural networks
- Recurrent neural networks

Generative Adversial Networks

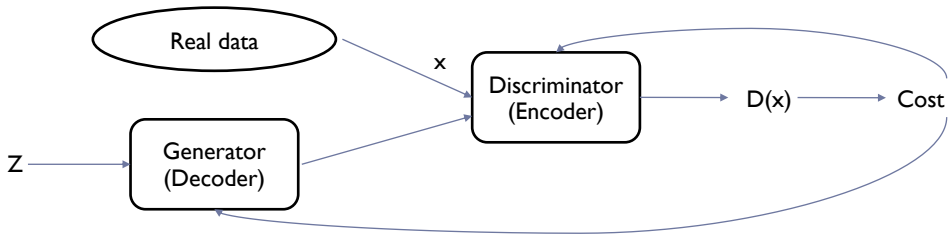
- Discriminator: $\max V(D) = \mathbb{E}_{x \sim p_{data}(x)}[logD(x)] + \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))]$

  recognize real data; recognize generated data

- Generator objective function: $\min V(G) = \mathbb{E}_{z \sim p_z(z)}[\log(1 - D(G(z)))]$

  optimize to fool discriminator

- Variants of GANs differ in objective function
- Suitable for prototype circuit design (layout) generation



I. Goodfellow, et al., "Generative Adversarial Nets", Proceedings of the International Conference on Neural Information Processing System, Vol. 2, No. 1, pp.,2672-2680, 2014

Practical Issues of Applying GAN

- Challenges in Training GANs
  - Mode collapse: limited varieties of samples are generated
  - Diminished gradient
  - Non-convergence
  - Overfitting **caused** by imbalance between the generator and discriminator
  - Highly sensitive to hyperparameters
- GAN for analog design generation
  - Difficult to be applied on tabular data
  - Suitable for image data
    - Placement images
    - Routing images

Drexel UNIVERSITY

56

Learning Scenarios Powered by Variants of ANNs

Differentiate by training scheme

- Supervised
- Semi-supervised
- Unsupervised
- Adversial (GAN)
- Reinforcement
- Encoder-decoder

X

Differentiate by data format

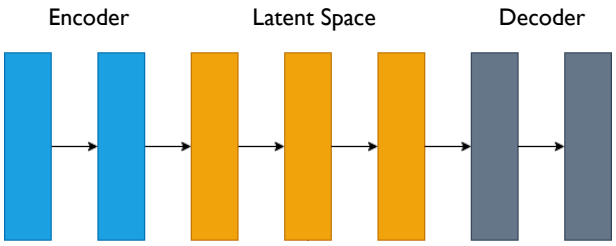
- Multi-layer perceptrons (feedforward neural networks)
- Convolutional neural networks
- Graph neural networks
- Recurrent neural networks

Drexel UNIVERSITY

57

Variational Auto-Encoders (VAEs)

- Generative algorithm to encode distribution of training data, then generate new data with similar distribution
- Encoder: map input to a low-dimensional latent space
  - Effectively dimensionality reduction
- Decoder: convert signal in latent space back to input space
- Difference from GAN:
  - GAN generator takes noise as input
    - Higher-quality generation
    - Harder to train
  - VAE takes signal from the low-dimensional latent space as input
    - Lower-quality generation
    - Easier to train



Drexel University

D. Kingma and M. Welling, "Auto-Encoding Variational Bayes", Proceedings of the International Conference on Learning Representations, 2014

58

Outline of Presentation

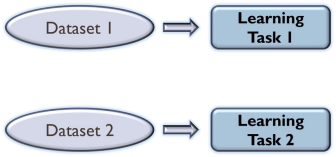
- Background Introduction
- Machine Learning Techniques for Analog EDA
  - Background on machine learning
  - Statistical learning algorithms
  - Neural-network-based learning algorithms
  - Transfer learning
- Optimization Techniques for Analog EDA
- Circuit Applications
- Analog Automation Platforms
- Conclusions

Drexel University

59

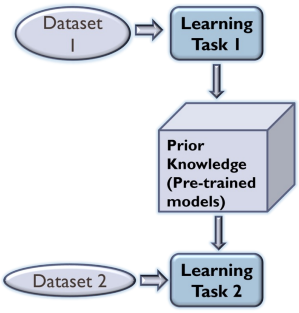
Transfer Learning to Improve Sample Efficiency

- Sample data in the search space of the given problem
  - Train independent machine learning models for each problem
- Challenge:
  - Circuit data is often proprietary
  - Acquiring sufficient data for each circuit task is costly or infeasible



Traditional supervised learning

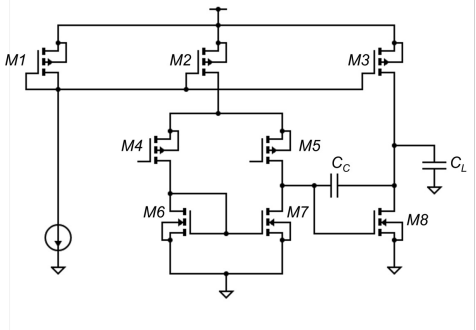
- Freeze a variable number of layers of the prior models
- Retrain with a smaller dataset in the target domain/node
- Benefits
  - Faster training
  - Improved model performance
  - Requirement of less data



Transfer learning

Transfer Performance Modeling across Technology Nodes for the Same Circuit

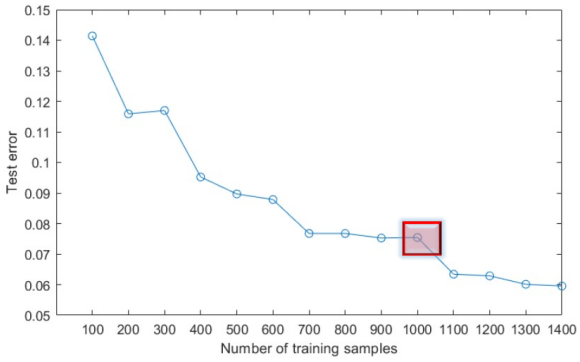
- Transfer learning is applied on models trained in 180nm for the performance modeling of an op-amp in 65nm
- Transfer learning significantly improves the sample efficiency for circuit performance modeling with simulation-based sizing data
  - Up to 50% improvement in MAE on test data

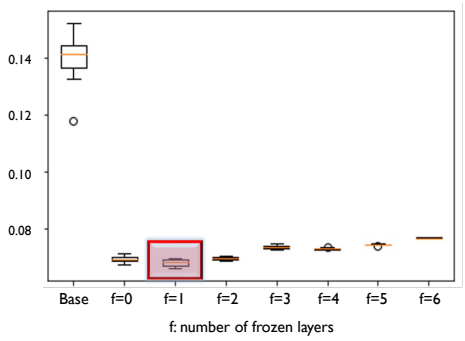


Z. Wu and I. Savidis, "Transfer Learning for Reuse of Analog Circuit Sizing Models Across Technology Nodes," Proceedings of the IEEE International Symposium on Circuits and Systems, pp. 1–5, 2022

Comparison of Sample Efficiency for Training of the Gain Predictor

- Standalone training requires 1000 training samples to achieve test error of 0.076
- Transfer learning requires 100 samples to achieve test error of 0.07





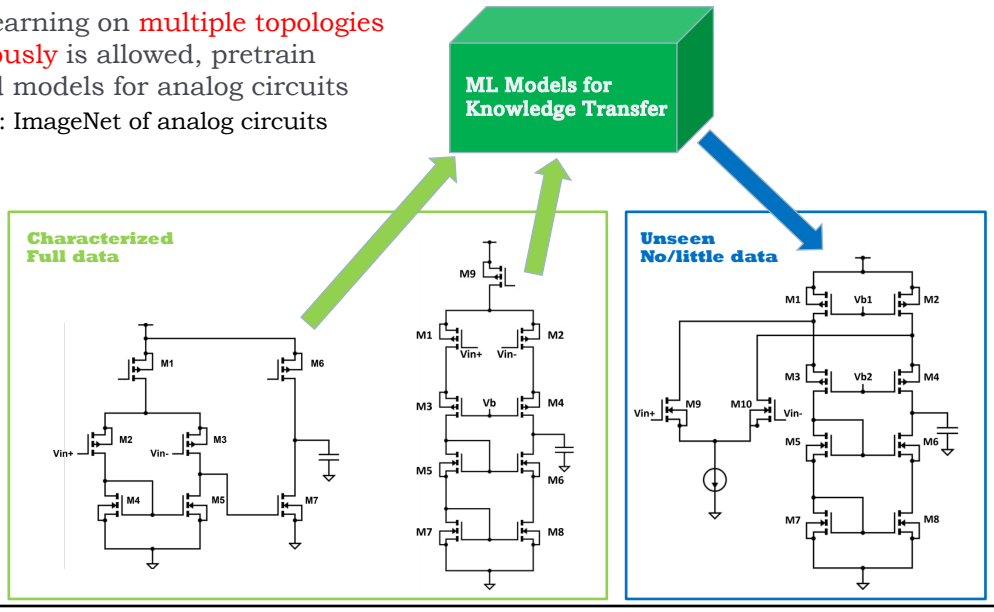
Drexel UNIVERSITY

Z. Wu and I. Savidis, "Transfer Learning for Reuse of Analog Circuit Sizing Models Across Technology Nodes," Proceedings of the IEEE International Symposium on Circuits and Systems, pp. 1–5, 2022

62

Transfer Performance Modeling across Analog Topologies

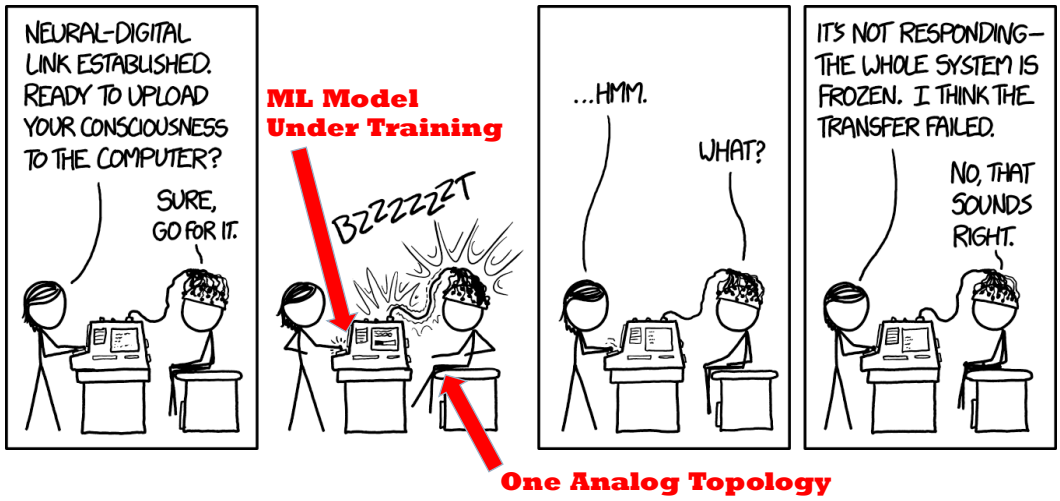
- Vision: If learning on multiple topologies simultaneously is allowed, pretrain generalized models for analog circuits
  - CircuitNet: ImageNet of analog circuits



Drexel UNIVERSITY

63

Is Transferring Predictions Across Analog Topologies Even Possible?



Drexel UNIVERSITY Source: xkcd.com

64

Challenges of Transferring Predictions Across Analog Topologies

- Each analog topology represents a unique mapping from the design space to performance space
  - Includes tradeoff considerations
- Traditional learning algorithms only work for a single topology if trained on device features
  - Different topologies ⇨ different number of devices ⇨ different feature dimensionality
- Therefore, models usually do not apply if circuit topology changes
  - Requires new data for a new circuit
- To transfer across topologies, first need models that learn topological information

Solution: Train Graph Neural Networks on Circuit Graphs

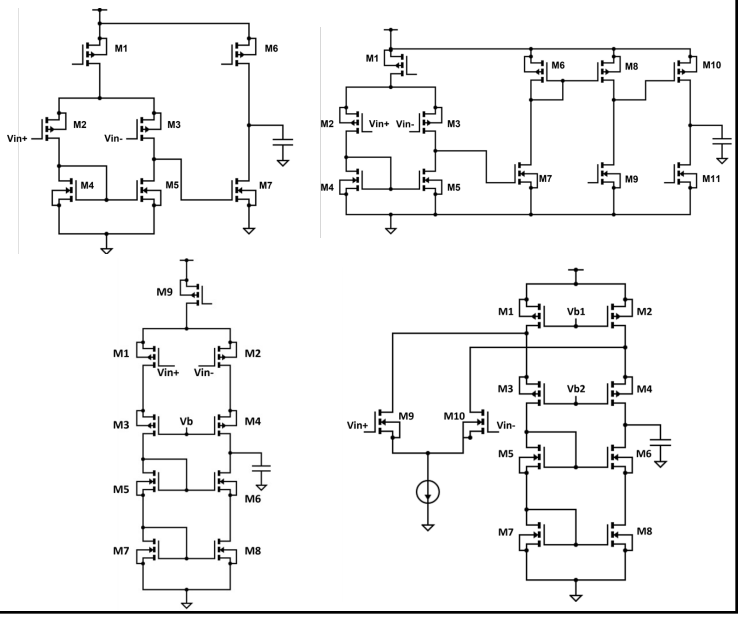
Drexel UNIVERSITY

65

Transfer Performance Models Across Four Op-amp Topologies

66

- Four op-amp topologies
  - Two-stage
  - Three-stage
  - Telescopic cascode
  - Folded cascode
- Device Features:
  - Transistor sizing
  - Transistor type



	Design Variables
Two-stage	W_1, W_2, W_4, W_6, W_7
Three-stage	$W_1, W_2, W_4, W_6, W_7, W_9, W_{10}, W_{11}$
Folded	W_1, W_3, W_5, W_7, W_9
Telescopic	W_1, W_3, W_5, W_7, W_9

Drexel UNIVERSITY

66

Transfer Performance Modeling by Applying GNN and Transfer Learning

67

- Train GNN for three amplifier topologies, test on the fourth topology
- Zero-shot learning: GNNs provide coarse estimates of the circuit performance
  - GNNs result in less test errors than the baseline ANNs for 14 of the 20 cases
- Few-shot learning: fine-tuned GNNs with transfer learning provide an average reduction of 70.6% in test error (RMSE) as compared to ANN models

	Two-stage as Test	Three-stage as Test	Folded as Test	Telescopic as Test
Power	-83% / -88%	+300% / -75%	+47% / -10%	+449% / -94%
Gain	-48% / -77%	+29% / -61%	-21% / -66%	-19% / -78%
Slew Rate	-62% / -74%	-80% / -82%	-97% / -99%	-98% / -99%
CMRR	-50% / -70%	-55% / -61%	-43% / -60%	-78% / -81%
PSRR	-15% / -67%	+59% / -39%	+33% / -60%	-67% / -71%

* Scenario 1 / Scenario 2

Drexel UNIVERSITY

Reference: Z. Wu and I. Savidis, "Transfer of Performance Models Across Analog Circuit Topologies with Graph Neural Networks," Proceedings of the ACM/IEEE Workshop on Machine Learning for CAD, pp. 159-165, 2022

67

Outline of Presentation

- Background introduction
- Machine learning techniques for analog EDA
- Optimization techniques for analog EDA
  - Gradient-based algorithms
  - Heuristic algorithms
  - Learning-based algorithms
  - Graph algorithms
- Circuit Applications
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

68

Machine Learning for Analog EDA

- Unique advantages brought by ML for EDA problems
  - Design space exploration
    - Extract patterns of circuit parameters/characteristics from circuit data
  - Predict performance metrics or physical characteristics of a circuit through the design stages
    - Early warning/skip doomed runs
      - Simulation time
  - Guide optimization for design generation
- ML for analog EDA in two key steps:

Learning
ML algorithms for circuit modeling

Optimization
Optimization algorithms for circuit generation

Drexel UNIVERSITY

69

Optimization Algorithms for EDA

- Gradient-based algorithms
- Heuristic algorithms
  - Greedy algorithms
  - Divide and conquer
  - Dynamic programming
  - Network flow algorithms
  - Linear/integer programming
  - Evolution-based algorithms
  - Simulated annealing
- Learning-based algorithms
  - Reinforcement learning
  - Surrogate-assisted optimization algorithms

- General flow of optimization process

Require: Objective function f
  $x^{(0)} \leftarrow$ random point in the domain of f
  for $i = 1, 2, \ldots$ **do**
    $\Delta x \leftarrow \pi(f, \{x^{(0)}, \ldots, x^{(i-1)}\})$
    if stopping condition is met **then**
      return $x^{(i-1)}$
    end if
    $x^{(i)} \leftarrow x^{(i-1)} + \Delta x$
  end for

Drexel UNIVERSITY

Reference: K. Li, J. Malik, "Learning to optimize," Proceedings of the International Conference on Learning Representations, 2017

Requirements on Optimization Algorithms for EDA

- Most optimization problems in physical design are NP-hard
- Optimization algorithms for EDA must have low time and space complexities, especially for physical design
  - When device count exceeds 100K, quadratic algorithms fail
  - Less of a problem for analog since device count is relatively small
- In physical design, key is to develop practical algorithms
  - Trade-off with optimality guarantee
- For an analog circuit, multi-objective optimization is required for multiple performance parameters
  - Trade-off between parameters (Pareto front) needed

Drexel UNIVERSITY

Reference: N.A.Sherwani, "Algorithms for VLSI Physical Design Automation", Springer, 1995

Outline of Presentation

- Background introduction
- Machine learning techniques for analog EDA
- Optimization techniques for analog EDA
  - Gradient-based algorithms
  - Heuristic algorithms
  - Learning-based algorithms
  - Graph algorithms
- Circuit Applications
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

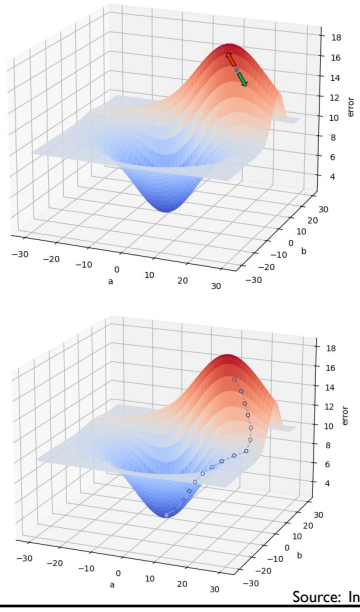
72

Gradient-based Optimization Algorithms

- At each step, search direction is defined by the gradient of the function evaluation
  - Intuitively, search along direction that reduces cost function at fastest rate
- Gradient descent
  - First-order search of local optimum

$$\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta_0, \theta_1)$$

- Advantages:
  - Theoretical guarantee of optimality
  - Fast execution for each iteration of search
- Limitation:
  - Requires explicit differentiable functions
  - Slow convergence and local minima for non-convex search space



Source: Interactivechaos.com

Drexel UNIVERSITY

73

Outline of Presentation

- Background introduction
- Machine learning techniques for analog EDA
- Optimization techniques for analog EDA
  - Gradient-based algorithms
  - Heuristic algorithms
  - Learning-based algorithms
  - Graph algorithms
- Circuit Applications
- Analog Automation Platforms
- Conclusions

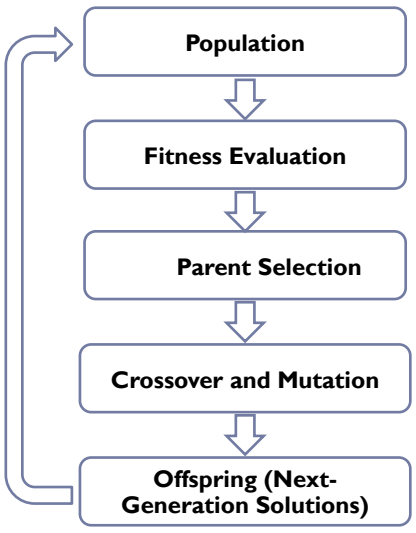
Drexel UNIVERSITY

74

Evolutionary Algorithms

- Characteristics of evolutionary algorithms
  - Population-Based
  - Fitness-Oriented
    - A fitness parameter to represent the quality of a solution
  - Variation-Driven
    - Crossover and mutation generates variants randomly
- Broad categories
  - Genetic algorithm
  - Particle swarm
  - Differential evolution
  - …

General flow of evolutionary algorithms



Drexel UNIVERSITY

75

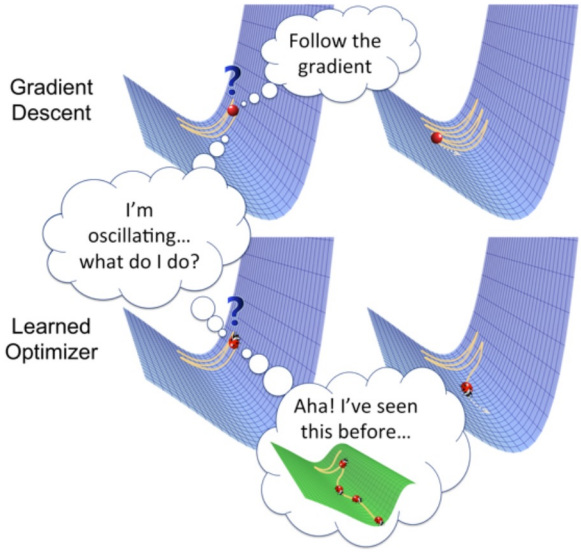
Simulated Annealing

- A global optimization technique by approximation
- Preferred when search space is discrete
- Parameter: a temperature value that keeps decreasing
  - temperature=initial temperature/(iteration+1)
- Steps:
  - Randomly initialize design variables and evaluate function value f_{old}
  - Sample again in the neighboring region and evaluate function value f_{new}
  - Action:
    - If function value improves, accept
    - If function value worsens, accept with probability $e^{-(f_{new}-f_{old})/temperature}$
- Advantage:
  - Reduces chance of getting stuck at local optimum since worse candidates are accepted with certain probability

Drexel UNIVERSITY

76

Reinforcement Learning for Optimization



Drexel UNIVERSITY

Reference: K. Li, J. Malik, "Learning to optimize," Proceedings of the International Conference on Learning Representations, 2017

77

Outline of Presentation

- Background introduction
- Machine learning techniques for analog EDA
- Optimization techniques for analog EDA
  - Gradient-based algorithms
  - Heuristic algorithms
  - Learning-based algorithms
  - Graph algorithms
- Circuit Applications
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

78

Reinforcement Learning (RL) for Optimization

- Define an agent to interact with the blackbox design space
  - Take action
  - Receive feedback
- Ingredients:
  - A state space: $\mathcal{S} \subseteq \mathbb{R}^D$
  - An action space: $\mathcal{A} \subseteq \mathbb{R}^d$
  - A cost function: $c : \mathcal{S} \rightarrow \mathbb{R}$
  - A time horizon: T
  - An initial state probability distribution: $p_i(s_0)$
  - A state transition probability distribution $p(s_{t+1} | s_t, a_t)$
- Objective: pick action (policy) $\pi(a_t | s_t, t)$ that maximizes expected cumulative rewards

Algorithm 1 General structure of optimization algorithms

Require: Objective function f
  $x^{(0)} \leftarrow$ random point in the domain of f
  for $i = 1, 2, \ldots$ **do**
    $\Delta x \leftarrow \phi(\{x^{(j)}, f(x^{(j)}), \nabla f(x^{(j)})\}_{j=0}^{i-1})$
    if stopping condition is met **then**
      return $x^{(i-1)}$
    end if
    $x^{(i)} \leftarrow x^{(i-1)} + \Delta x$
  end for

Gradient Descent	$\phi(\cdot) = -\gamma \nabla f(x^{(i-1)})$
Momentum	$\phi(\cdot) = -\gamma \left(\sum_{j=0}^{i-1} \alpha^{i-1-j} \nabla f(x^{(j)}) \right)$
Learned Algorithm	$\phi(\cdot) =$ Neural Net

Drexel UNIVERSITY

Reference: K. Li, J. Malik, "Learning to optimize," Proceedings of the International Conference on Learning Representations, 2017

79

RL Formulation for Optimization on Analog Circuits

- *Time horizon* = number of controllable design parameters in the circuit
- Action: decide on the value of one design parameter at each step
- After T steps, a set of candidate solution is generated
- Evaluate the solutions from a circuit solver (e.g., SPICE), calculate reward
- Maximize expected cumulative rewards = optimize target

  action space = feasible solution set for the design variables

Circuit applications:

- For placement optimization, move one instance each time
  - Instances: standard cells and macros for digital, devices or circuit blocks for analog
- For analog sizing, set figure of merit as optimization target
  - Action at each step is to determine the size of one transistor

Drexel UNIVERSITY

A. F. Budak, Z. Jiang, K. Zhu, A. Mirhoseini, A. Goldie and D. Z. Pan, "Reinforcement Learning for Electronic Design Automation: Case Studies and Perspectives," Proceedings of the Asia and South Pacific Design Automation Conference pp.500-505, 2022

80

Surrogate-assisted Blackbox Optimization

- **Two primary steps:**
  - **Step 1: Surrogate modeling**
    - Gaussian process models
    - Neural networks
    - SVMs
    - …
  - **Step 2: Active querying (adaptive sampling)**
    - Objective: sample points with maximum utility at minimum cost
      - Cost: number of points, simulation time
    - Methods:
      - Uncertainty sampling
        - Entropy
      - Information gain
        - Maximize KL divergence between posterior and prior
      - Query by committee
      - Response surface method

Drexel UNIVERSITY

81

Bayesian Optimization

- A type of surrogate-assisted blackbox optimization technique
- Gaussian process function as surrogate model
- Acquisition function: determine next point to query while balancing exploitation and exploration of the search space
  - Expected improvement (EI)
    - Maximize expected improvement over current best value

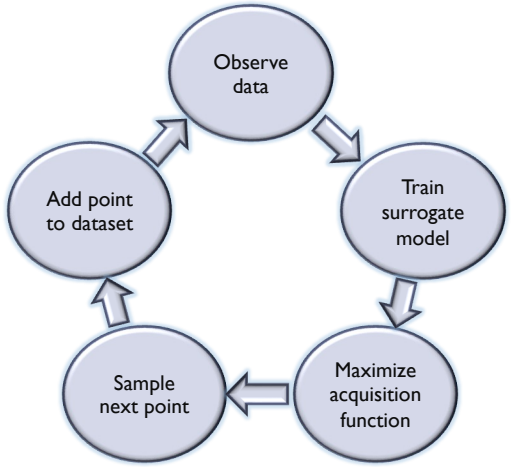
$$u(x) = \max(0, f' - f(x))$$

Current minimal value

  - Upper confidence bound (UCB)
    - Search areas with either best function value or largest uncertainty

$$\alpha(x;\lambda) = \mu(x) + \lambda\sigma(x)$$

Exploitation Exploration



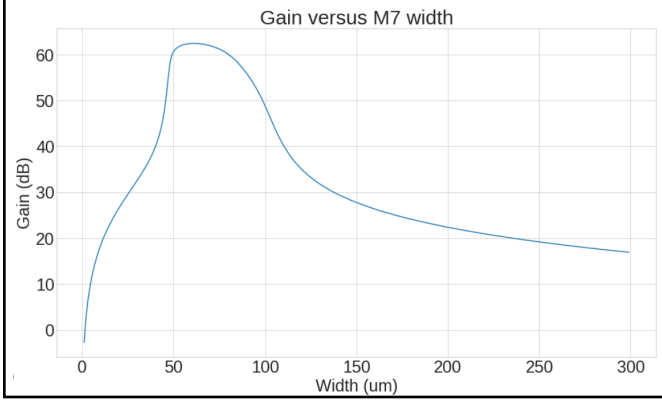
Drexel UNIVERSITY

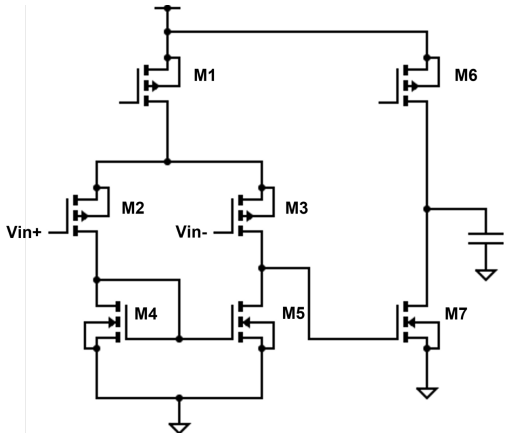
Reference: W. Lyu, et al., "An Efficient Bayesian Optimization Approach For Automated Optimization Of Analog Circuits," IEEE Transactions on Circuits and Systems, Vol. 65, No. 6, pp. 1954–1967, Jun. 2018

82

Example: Bayesian Optimization For the Sizing of An Amplifier

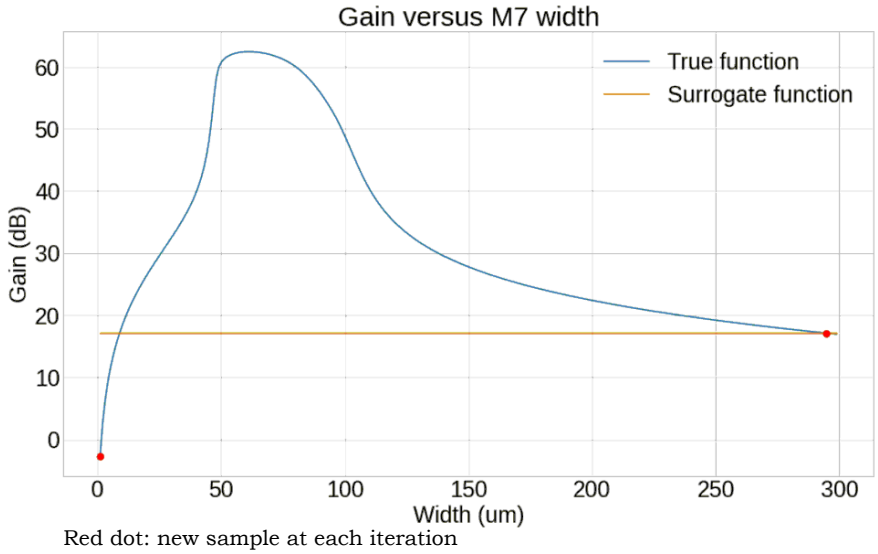
- Design parameter: W7 (transistor M7 width)
  - Assuming all other transistors are sized
- Performance parameter: AC gain
  - Specification: over 62dB
- As a reference, plot design space (gain vs W7) with a parametric sweep of W7:





83

Animation of Sampling and Surrogate Function Update with Bayesian Optimization



Red dot: new sample at each iteration

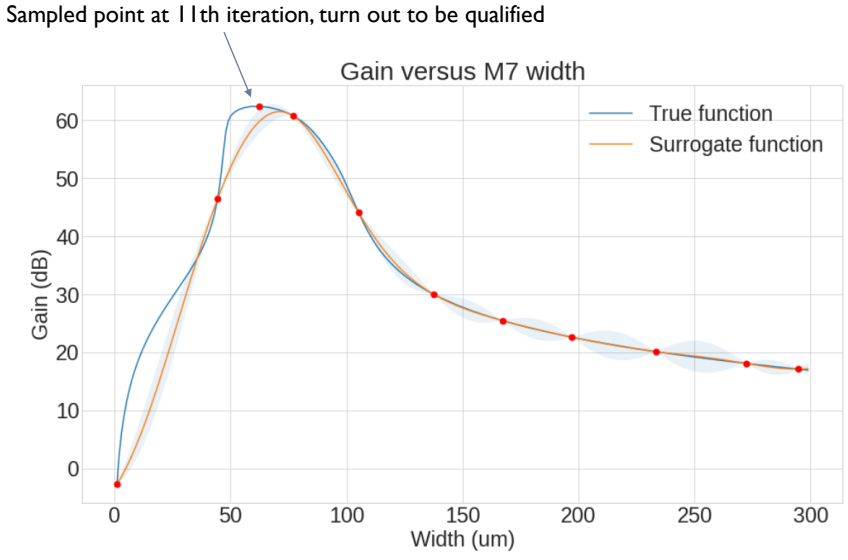
Grey region: uncertainty over the region between sampled points

Drexel UNIVERSITY

84

Animations of Sampling and Surrogate Function Update with Bayesian Optimization

- Qualified solution found to deliver gain of 62dB after 11 iterations



Drexel UNIVERSITY

85

Outline of Presentation

- Background introduction
- Machine learning techniques for analog EDA
- Optimization techniques for analog EDA
  - Gradient-based algorithms
  - Heuristic algorithms
  - Learning-based algorithms
  - Graph algorithms
- Circuit Applications
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

86

Graph-based Optimization Algorithms for EDA

- Algorithms specifically for graph-structured design space
- Types of graph applications:
  - Graph partitioning
  - Graph traversal (including shortest path search)
    - Depth-first search
      - Search in order of increasing depth before returning to root node
    - Breadth-first search
      - Search in order of distance from the source node
    - Best-first search
      - Search guided by cost criteria

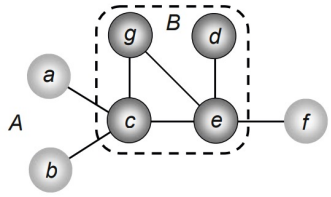
Drexel UNIVERSITY

Reference: A. B. Kahng, J. Lienig, I. L. Markov, and J. Hu, "VLSI Physical Design: From Graph Partitioning to Timing Closure," Springer, 2011

87

Graph Partitioning

- Objective of partitioning for digital physical design: minimize cut edges while balancing for partition sizes
  - Divide and conquer: place and route each partition separately before integration



- Objective of partitioning of analog circuits before layout: identify circuit hierarchies
  - Grouping utilized as constraints for placement and routing
    - Example: match for symmetry
  - Algorithms therefore differ from digital partitioning
    - Subgraph isomorphism

A. B. Kahng, J. Lienig, I. L. Markov, and J. Hu, "VLSI Physical Design: From Graph Partitioning to Timing Closure," Springer, 2011

Shortest Path Search

- Routing regions represented as a graph
- Dijkstra's algorithm (a.k.a, maze routing):
  - Find shortest paths between a given node and all other nodes
- A* algorithm:
  - Find shortest path between a given node and a target node
  - Given an initial and final cell on a square grid
    - g : cost of moving from the initial cell to a certain cell on grid
    - h : estimated cost of moving from the current cell to the final cell
      - Euclidean distance
      - Manhattan distance
    - f = g + h
    - Procedure: select and move to the smallest f-valued cell
  - Limitation: high space complexity as storage of all nodes in paths is required

A* search between bottom-left red dot to upper-right green dot

Source: Wikipedia

Reference: A. B. Kahng, J. Lienig, I. L. Markov, and J. Hu, "VLSI Physical Design: From Graph Partitioning to Timing Closure," Springer, 2011

Requirements on Optimization Algorithms for EDA

- Optimization algorithms for EDA must have low time and space complexities
  - Most optimization problems in physical design are NP-hard
  - When device count exceeds 100K, quadratic algorithms fail
    - Develop practical algorithms: trade-off with optimality guarantee
- Need multi-objective optimization that provides design trade-offs

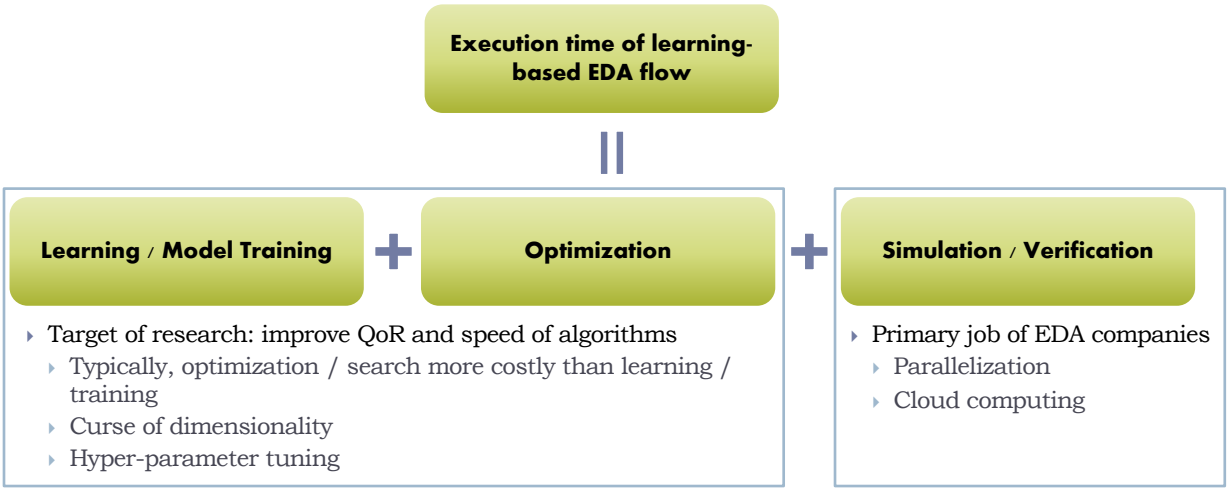
	Derivative-free?	Guarantee optimality?	Computational complexity	Allow prior knowledge	Tradeoff considerations
Gradient-based	No	Yes	Low	No	Yes
Evolution-based	Yes	No	Medium	No	Yes
Bayesian optimization	Yes	No	High	Yes	Yes
Reinforcement learning	Yes	No	High	Possibly	Not well formulated

Drexel UNIVERSITY

N.A.Sherwani, "Algorithms for VLSI Physical Design Automation," Springer, 1995

91

Execution time of Applying ML and Optimization for EDA Problems



Drexel UNIVERSITY

92

Outline of Presentation

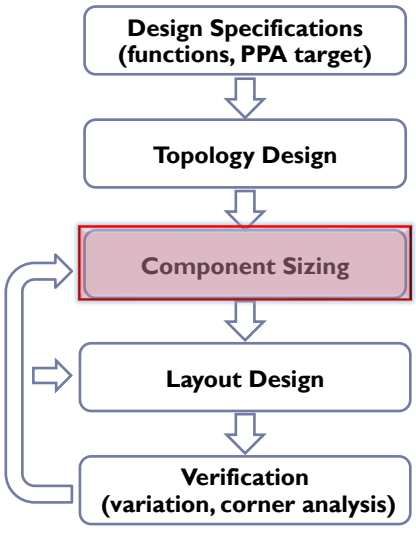
- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
  - Application 1: component sizing of analog ICs
  - Application 2: analog circuit topology generation
  - Application 3: constraint formulation for P&R
  - Application 4: placement
  - Application 5: routing
  - Application 6: device modeling
  - Application 7: RF matching network synthesis
  - Application 8: LLMs for analog circuit generation
  - Application 9: prediction of interconnect impedance
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

93

Case study: Component Sizing of Analog ICs

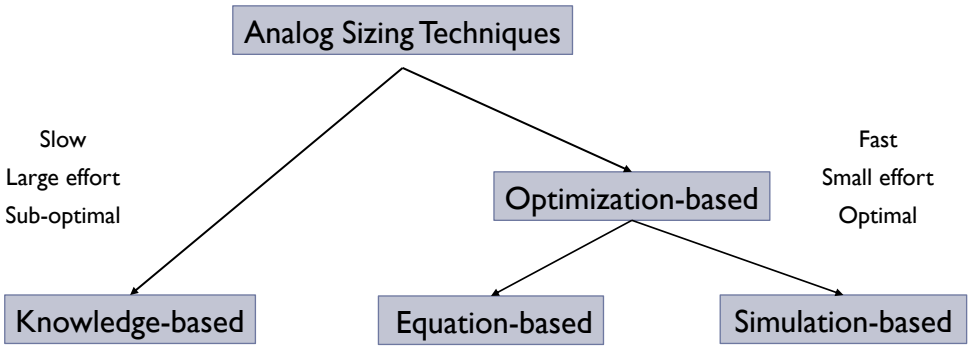
- Component sizing: tuning the sizes of the devices in an analog circuit to meet design specifications
  - Example: transistor width, length, number of fins and fingers; resistor, capacitor values…
  - A critical step in analog synthesis flow
- Utilize ML to:
  - Predict circuit performance based on design parameters
  - Predict layout-dependent effects and interconnect impedances to guide sizing optimization



Drexel UNIVERSITY

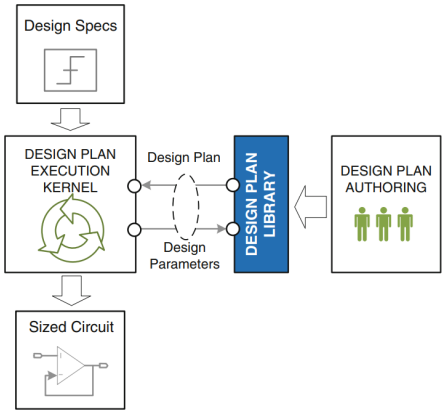
94

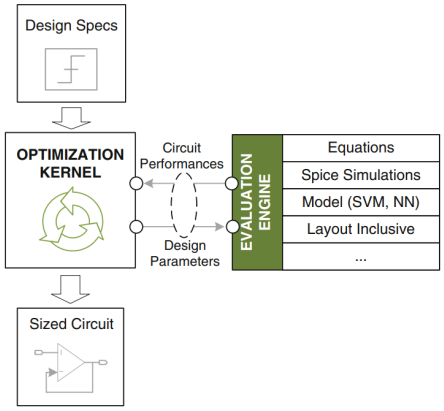
Classification of Analog Sizing Techniques in Literature



Classification of Analog Sizing Techniques in Literature

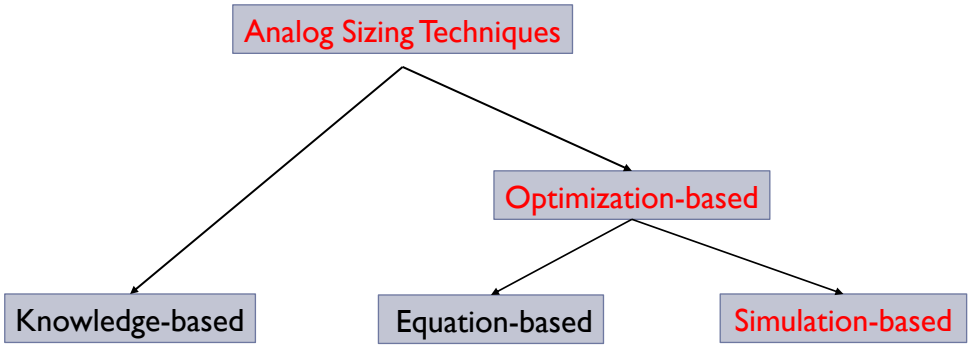
- Knowledge-based sizing
  - Pre-designed plans consisting of design equations and procedures
- Optimization-based sizing
  - Optimization problems formulated based on
    - Circuit equations
    - Simulation data





R. Lourenço, N. Lourenço, N. Horta, "AIDA-CMK: Multi-Algorithm Optimization Kernel Applied to Analog IC Sizing", Springer, 2015

Classification of Analog Sizing Techniques in Literature



Drexel UNIVERSITY

99

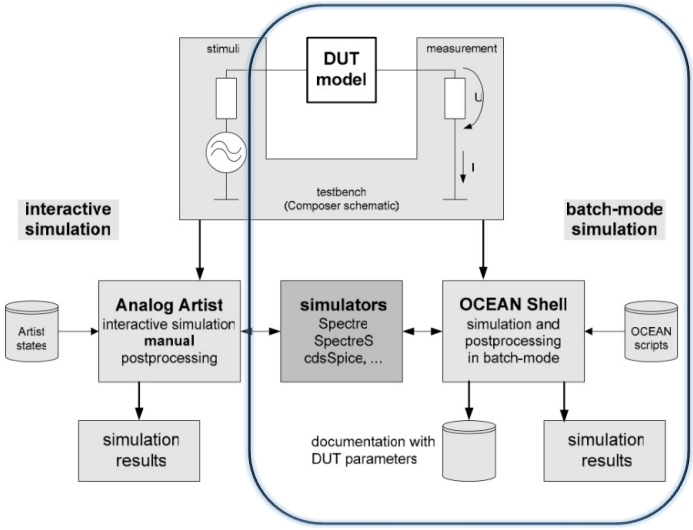
ML-driven Simulation-based Optimization for Analog Circuit Sizing

Treat the circuit as a blackbox

- Train surrogate models to map from the design space to the performance space
- Simulation-based sizing in two primary steps:
  Performance modeling + Optimization
- No design equations needed
- No discrepancy between theoretical and simulated results

Challenges:

- Circuit simulations with numerical solvers are expensive
  - Need to improve sample efficiency



Drexel UNIVERSITY

Reference: R. Frevert, et al., "Modeling and Simulation for RF System Design", Springer, pp.291, 2005

100

Variation-aware and Yield-aware Analog IC Sizing

- High-dimensional variation space
  - Process and environmental variations
- Trade-off between yield and performance
  - Optimizing for robustness sacrifices performance metrics such as gain or bandwidth
- Model second-order effects or mismatch behavior
- Statistical yield maximization

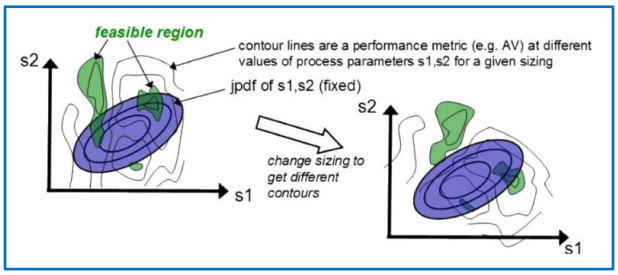
Drexel UNIVERSITY

101

Variation-Aware Analog Circuit Design: SANGRIA Framework

- Structural Homotopy
  - Lower layers includes simplified objectives
    - Nominal DC simulation
  - Higher layers includes tight objectives with full PVT corner simulations
  - Continuously injects random designs into lower layers to explore new regions
  - Helps escape local optima and promotes global exploration and refinement
- Response Surface Modeling
  - Builds adaptive surrogate models based on past simulation data
  - Selects new candidate designs
  - Scale linearly with sample size and input dimension
  - Uncertainty-aware multi-objective sampling

Design-centering view of yield optimization

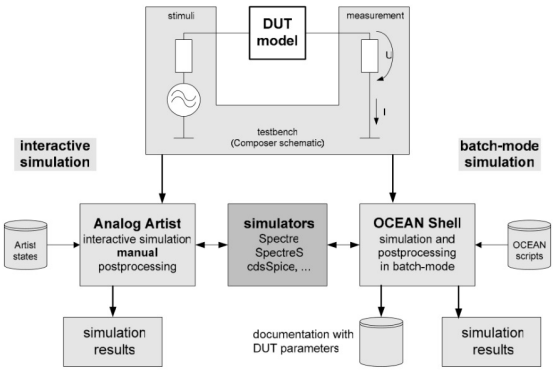


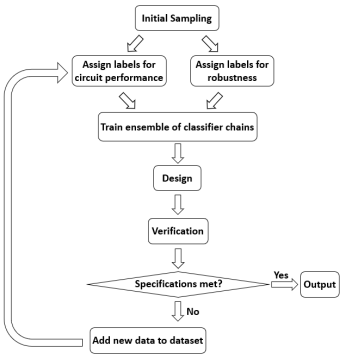
Drexel UNIVERSITY

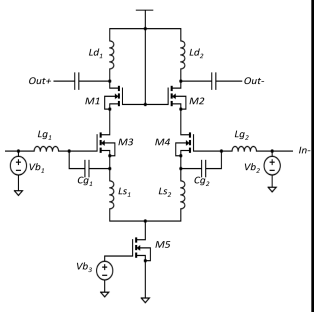
T. McConaghy and G. G. E. Gielen, "Globally Reliable Variation-Aware Sizing of Analog Integrated Circuits via Response Surfaces and Structural Homotopy," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 28, no. 11, pp. 1627-1640, Nov. 2009

105

VCALT: Variation-aware Classification with Adaptive Labeling Thresholds for Analog Sizing







$$x^* \in arg\max(p_1(x), ..., p_k(x), r_1(x), ..., r_k(x))$$

- 5 process corners and 3 temperature corners considered (TT, FF, SS, SF, FS) * (20°C, 80°C, 120°C)

Constraints

$$\begin{cases} 60\ nm \le transistor\ widths \le 900\ \mu m \\ 0.01\ nH \le inductor\ sizes \le 12\ nH \\ 30\ fF \le capacitor\ sizes \le 20\ pF \\ 0\ V \le biasing\ voltages \le 1.2\ V \end{cases}$$

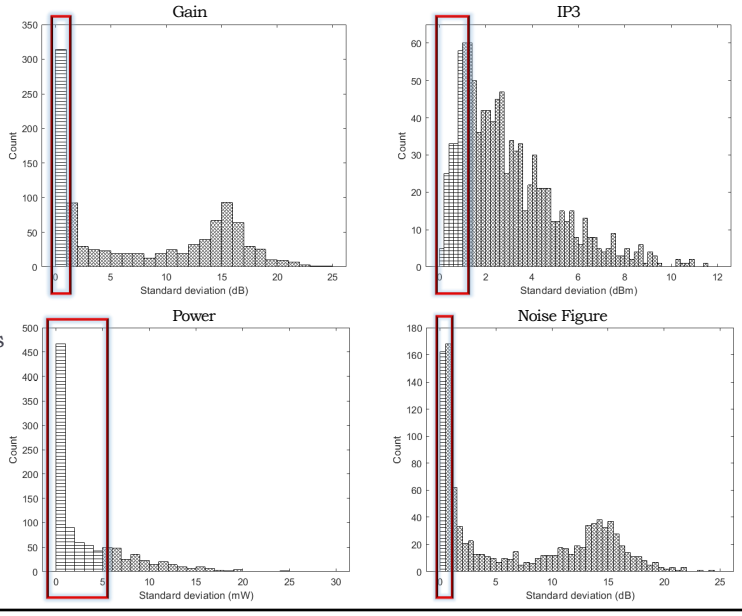
Specifications

$$\begin{cases} Gain \ge 10\ dB \\ NF \le 2.8\ dB \\ IP3 \ge -5\ dBm \\ Power \le 20\ mW \end{cases} \qquad \begin{cases} \sigma_{Gain} \le 1\ dB \\ \sigma_{NF} \le 0.5\ dB \\ \sigma_{IP3} \le 1\ dBm \\ \sigma_{Power} \le 5\ mW \end{cases}$$

[1] Z. Wu and I. Savidis, "Variation-aware Analog Circuit Sizing with Classifier Chains," Proceedings of the ACM/IEEE Workshop on Machine Learning for CAD, pp. 1–6, 2021
[2] R. Frevert, et al., "Modeling and Simulation for RF System Design", Springer, pp.291, 2005

Distribution of Standard Deviations of Performance Parameters across 15 Corners in the Initial Dataset

- Without accounting for robustness, design solutions have large fluctuations in circuit performance over different corners
- Classifiers to predict and select robust candidate solutions
  - Thresholds on the standard deviations are set by the designer



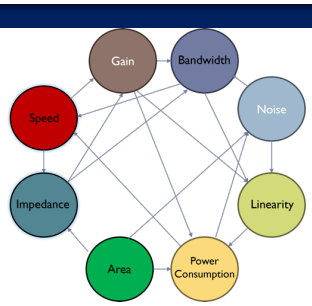
Outline of Presentation

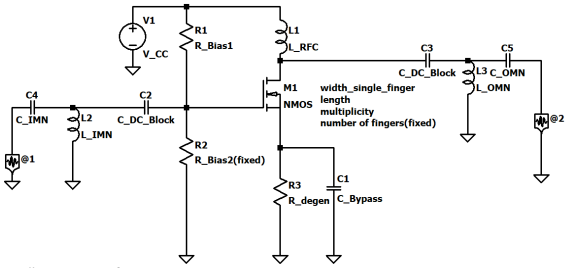
- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
  - Application 1: component sizing of analog ICs
  - Application 2: analog circuit topology generation
  - Application 3: constraint formulation for P&R
  - Application 4: placement
  - Application 5: routing
  - Application 6: device modeling
  - Application 7: RF matching network synthesis
  - Application 8: LLMs for analog circuit generation
  - Application 9: prediction of interconnect impedance
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

Challenges of Generating Analog/RF Topologies

- Objectives:
  - Generate the structural arrangement of circuit building blocks
  - Define device connectivity before sizing and layout
- Challenges faced:
  - Complex multi-objective tradeoffs with different performance parameters
  - Process variation and parasitics demand more robust topologies
  - Vast, expansive design space



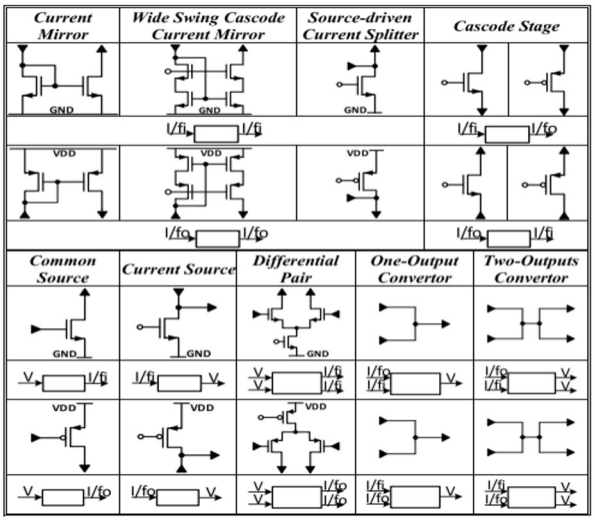


Drexel UNIVERSITY

Z. Zhao and L. Zhang, "Graph-grammar-based analog circuit topology synthesis," *Proceedings of the IEEE International Symposium on Circuits and Systems*, Sapporo, Japan, , pp. 1–5, May. 2019

An Automated Topology Synthesis Framework for Analog ICs

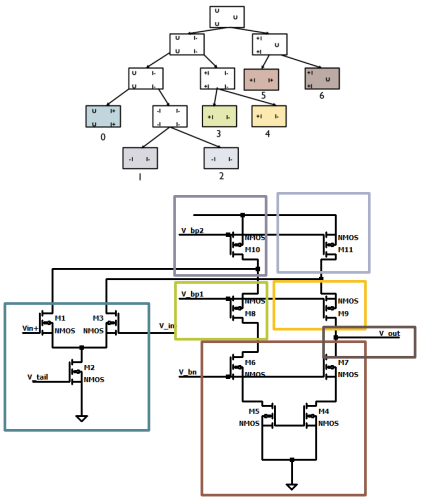
- Goals:
  - Task: Topology generation with GCTG from library, to meet performance metrics for unsized and sized circuits
  - Evaluation: SPICE-based verification for topologies passing un-sized stage



Z. Zhao and L. Zhang, "Graph-grammar-based analog circuit topology synthesis," *Proceedings of the IEEE International Symposium on Circuits and Systems*, Sapporo, Japan, , pp. 1–5, May. 2019

Graph Grammar for Circuit Topology Generation (GCTG)

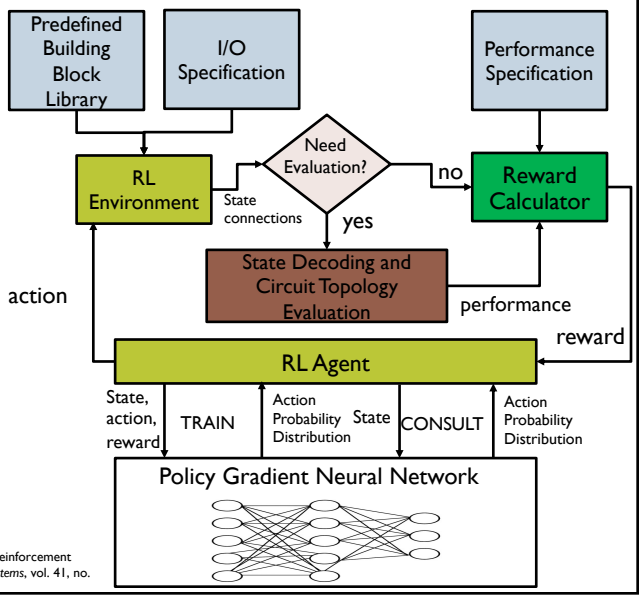
- Define a set of rules to systematically build or modify a graph structure
  - Used to control circuit signal path formation
- Each rule specifies a valid transformation:
  - Decomposition rule: Signals branch into multiple paths
  - Symmetry rule: Certain branches mirror each other
  - Termination Rule: Every signal path terminate into a functional block
  - Functional Assignment Rule: Assign specific functional roles
  - Port Compatibility Rule: Connections between blocks must respect signal type



Z. Zhao and L. Zhang, "Graph-grammar-based analog circuit topology synthesis," *Proceedings of the IEEE International Symposium on Circuits and Systems*, Sapporo, Japan, , pp. 1–5, May. 2019

Deep Reinforcement Learning for Analog Circuit Topology Generation

- Deep Reinforcement Learning (DRL) applies deep neural networks to approximate the policy or value function in reinforcement learning
- The policy is parameterized by the neural network
  - Optimized to maximize expected cumulative reward
- Training:
  - Reward signals are used to compute gradients and update network parameters using methods like policy gradients
- Purpose:
  - DRL replaces explicit action tables with continuous, generalizable models, allowing learning in environments too large for tabular RL



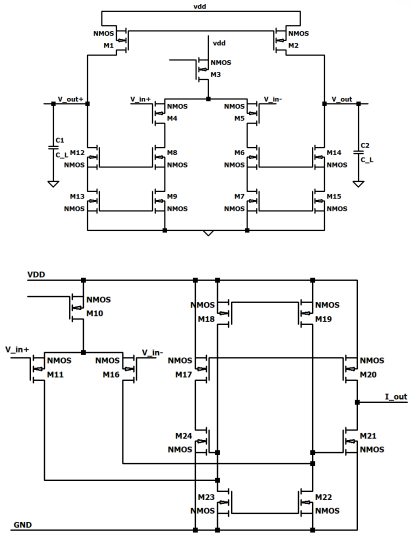
Drexel University

Z. Zhao and L. Zhang, "Analog integrated circuit topology synthesis with deep reinforcement learning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 12, pp. 5138–5151, Dec. 2022.

115

Analog Topology Synthesis With Deep Reinforcement Learning

- Attempting to synthesize operational amplifiers with the following specifications:
  - DC Gain ≥ 60 dB
  - Phase Margin (PM) ≥ 60°
  - Gain Margin (GM) ≥ 10 dB
  - Unity-Gain Bandwidth (UGB) ≥ 10 MHz
- ~8000 topologies generated
- 2150 valid after SPICE sizing
- Valid designs include subset of circuits with >100 dB gain & a subset with >1 GHz UGB
- Efficiency: 50–70% of topologies filtered before sizing using fast DC gain estimate
- 99% of filtered circuits (DC gain <30 dB) fail full sizing
- DRL learns to synthesize high-performance circuits with fewer simulations and better scalability than traditional methods.



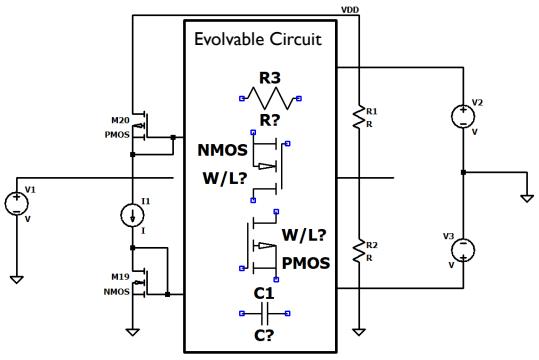
Drexel University

Z. Zhao and L. Zhang, "Analog integrated circuit topology synthesis with deep reinforcement learning," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 12, pp. 5138–5151, Dec. 2022.

116

Analog Flat-Level Circuit Synthesis With Genetic Algorithms

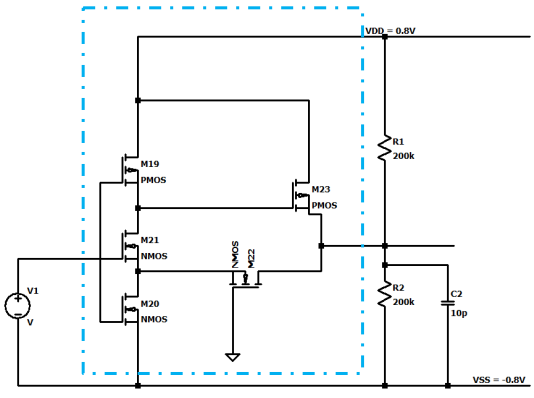
- Single-stage amplifier:
  - DC gain > 40 dB
  - THD < 1%,
- Two-stage amplifier:
  - DC gain > 40 dB,
  - GBW > 10 MHz,
  - THD < 1%,
- Power consumption minimized across all designs
- Evaluation: NGSPICE simulation, performance-driven fitness function



M. Campilho-Gomes, R. Tavares, and J. Goes, "Analog flat-level circuit synthesis with genetic algorithms," *IEEE Access*, vol. 12, pp. 115 532–115 545, Aug. 2024

Results of Analog Flat-Level Circuit Synthesis With Genetic Algorithms

- Single-Stage Amplifier:
  - DC gain: 42–48 dB
  - THD: 0.4–0.9%
  - Power: ~100–250 µW
- Two-Stage Amplifier:
  - DC gain: 45–53 dB
  - GBW: 10–15 MHz
  - THD: 0.5–0.8%
  - Power: ~120–300 µW
- Convergence:– 30–100 generations depending on circuit complexity
- Circuit Complexity:– 6–14 transistors per topology
- Power Reduction:– 20–30% improvement compared to random search baselines



M. Campilho-Gomes, R. Tavares, and J. Goes, "Analog flat-level circuit synthesis with genetic algorithms," *IEEE Access*, vol. 12, pp. 115 532–115 545, Aug. 2024

Comparison of Algorithms

Aspect	GCTG (Grammar Tree)	Deep RL	GA (Flat-Level Evolution)
Topology Growth	Tree structure expanded by recursive grammar rules	Incremental block addition via agent actions; topology grows step-by-step	Direct mutation and crossover of entire device-level circuits without hierarchy
Control Over Structure	Very high; tight constraint on search space	Medium — exploration space constrained by environment setup	Low —mutations and crossovers drive exploration
Feedback Mechanism	SPICE sim only for promising candidates	Agent only gets meaningful signal after full circuit build	Fitness function combining multiple objectives
Main Algorithmic Challenges	Managing tree size, avoiding redundant subtrees (isomorphism detection needed)	High variance in learning due to delayed sparse rewards; instability during training	Premature convergence to suboptimal topologies; structural bloat without good penalties
Best Suited For	Highly structured circuits with known flow	Discovering unconventional or creative topologies	Broad, flexible exploration when domain-specific constraints are weak

Drexel UNIVERSITY

119

Outline of Presentation

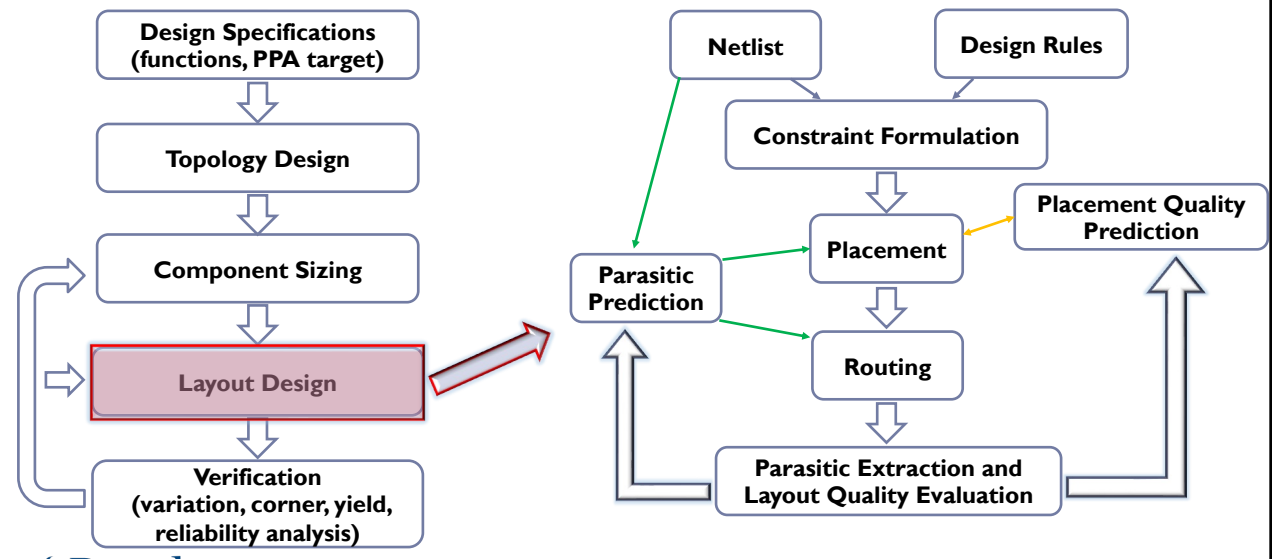
- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
  - Application 1: component sizing of analog ICs
  - Application 2: analog circuit topology generation
  - Application 3: constraint formulation for P&R
  - Application 4: placement
  - Application 5: routing
  - Application 6: device modeling
  - Application 7: RF matching network synthesis
  - Application 8: LLMs for analog circuit generation
  - Application 9: prediction of interconnect impedance
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

121

Breakdown of Analog Layout Automation

122



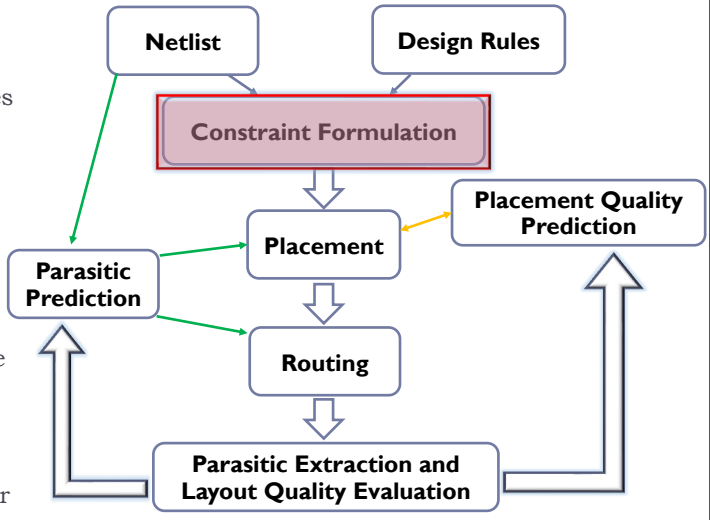
Drexel UNIVERSITY

122

Motivation to Perform Device Grouping and Circuit Hierarchy Recognition

123

- Human designers recognize analog subblocks
  - Account for device grouping and hierarchies when performing layout
  - Match devices for symmetry
- Heuristic techniques to recognize sub-circuits
  - Library-based: match with a library enumerating possible topologies
  - Rule-based: programmed rules to recognize sub-circuits
- Limitation of heuristic techniques:
  - Exhaustive enumeration of circuit blocks or recognition rules is often infeasible

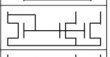
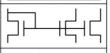


Drexel UNIVERSITY

123

Analog Circuit Hierarchies

- Four primary levels of an analog circuit
  - Individual device level
  - Sub-block level
    - Differential pairs
    - Current mirrors
    - …
  - Sub-circuit level
    - OTA
    - VCO
    - …
  - System level
    - RF transceiver
    - Data converter
- Heuristic techniques to recognize sub-circuits
  - Library-based: match with a library enumerating possible topologies
  - Rule-based: programmed rules to recognize sub-circuits

Primitive	Schematic	Primitive	Schematic
Resistor		Current mirror 1	
Capacitor		Current mirror 2	
Capacitor array		Voltage reference	
Switch		Level shifter	
Diode-connected load		Current mirror load	
Differential pair		Current mirror bank	
Cross-coupled pair 1		Level shifter bank	
Cross-coupled pair 2		Dummy 1	
Differential load		Dummy 2	
Cascode pair		Decap	

Reference: T. Dhar, et al., "ALIGN: A System for Automating Analog Layout," IEEE Design & Test, Vol. 38, No. 2, pp. 8-18, 2021

124

Recognizing Functional Groupings in Automated Analog Synthesis Flow

- Unlike system-level grouping, device-level grouping typically unavailable from a circuit netlist
  - Need algorithms for detection of device groupings
- Automated synthesis of analog ICs requires accurate and fast recognition of circuit hierarchies
  - Provide constraints on symmetry for device sizing, placement, and routing
  - Provide pathway towards machine reasoning and learning-based topology synthesis
    - Provide features for ML models for downstream circuit tasks

From a snippet of a system-level netlist, *can you tell which two devices are in which functional group?*

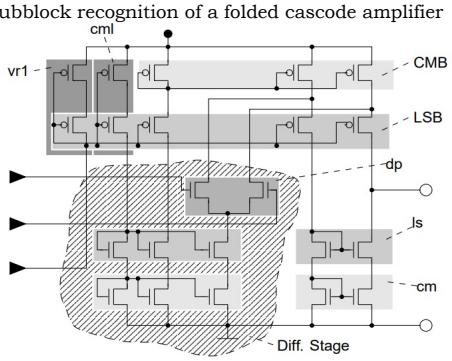
```
********************************************************************************
* Library Name: Receiver
* Cell Name:    VGA2_load
* View Name:    schematic
********************************************************************************

.SUBCKT VGA2_load Vbias Vcn Vcp Vinn Vinp Voutn Voutp Vtn Vtp gnd vdd
*.PININFO Vbias:I Vcn:I Vcp:I Vinn:I Vinp:I Vtn:I Vtp:I Voutn:O Voutp:O gnd:B
*.PININFO vdd:B
MM9 vdd Vcn Voutn Voutn nch l=300n w=600n m=1
MM11 vdd Vcn Voutp Voutp nch l=300n w=600n m=1
MM13 net023 Vbias gnd Vtn nch l=120.0n w=3u m=110
MM6 Voutn Vinp net023 net023 nch l=100n w=4u m=24
MM8 Voutp Vinn net023 net023 nch l=100n w=4u m=24
MM12 net028 net027 Voutp Voutp nch l=300n w=1u m=80
MM10 net029 net030 Voutn Voutn nch l=300n w=1u m=80
XC0 Voutn gnd gnd mimcap_um_sin_rf lt=70u wt=64.0u mimflag=3 m=1
XC1 Voutp gnd gnd mimcap_um_sin_rf lt=70u wt=64.0u mimflag=3 m=1
MM5 net030 Vcp vdd Vtp pch l=300n w=500n m=7
MM7 net027 Vcp vdd Vtp pch l=300n w=500n m=7
XR3 vdd net029 rppolyl l=13.6u w=2u m=1
XR4 vdd net028 rppolyl l=13.6u w=2u m=1
.ENDS

********************************************************************************
* Library Name: Receiver
* Cell Name:    VGA1_layout
* View Name:    schematic
********************************************************************************

.SUBCKT VGA1_layout Vbias Vcn Vcp Vinn Vinp Voutn Voutp Vtn Vtp gnd vdd
*.PININFO Vbias:I Vcn:I Vcp:I Vinn:I Vinp:I Vtn:I Vtp:I Voutn:O Voutp:O gnd:B
*.PININFO vdd:B
MM5 net34 Vcp vdd Vtp pch l=1u w=850.0n m=1
MM7 net33 Vcp vdd Vtp pch l=1u w=850.0n m=1
MM15 net22 Vbias gnd Vtn nch l=1u w=1u m=45
MM6 Voutn Vinp net22 net22 nch l=300n w=1u m=8
MM8 Voutp Vinn net22 net22 nch l=300n w=1u m=8
MM13 net22 Vbias gnd Vtn nch l=1u w=1u m=45
MM12 net31 net33 Voutp Voutp nch l=300n w=1u m=10
MM9 vdd Vcn Voutn Voutn nch l=300n w=300n m=1
MM11 vdd Vcn Voutp Voutp nch l=300n w=300n m=1
MM10 net32 net34 Voutn Voutn nch l=300n w=1u m=10
XR1 vdd net31 rppolys l=200.0000u w=400n m=1
XR0 vdd net32 rppolys l=200.0000u w=400n m=1
.ENDS

********************************************************************************
* Library Name: Receiver
* Cell Name:    charge_pump
* View Name:    schematic
********************************************************************************

.SUBCKT charge_pump_realmodel DOWN GND NDOWN NUP OUT UP VDD
```

125

Knowledge-based Constraint Formulation: Sizing Rules Method

- A library of analog building blocks implemented based on a hierarchical set of circuit components
  - Allows for subblock recognition and structural synthesis

Subblock recognition of a folded cascode amplifier

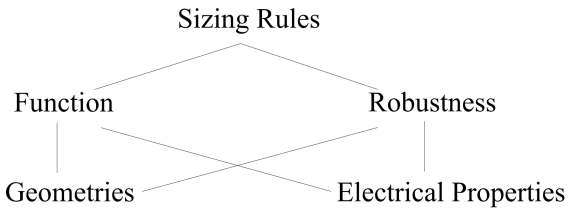


Function	Schematic NMOS (PMOS do.)	Hierarchical Level
Voltage-Controlled Resistor (vcres) Volt.-Contr. Current Source (vccs)		0
Simple Current Mirror (cm)		1
Level Shifter (ls)		
Voltage Reference 1 (vr1)		
Current Mirror Load (cml)		
Differential Pair (dp)		
Voltage Reference 2 (vr2)		
Flip-Flop (ff)		
Level Shifter Bank (LSB)		2
Current Mirror Bank (CMB)		
Cascode Current Mirror (CCM)		
4-Transistor Current Mirror (4TCM)		
Differential Stage (CM=cm/CCM/4TCM)	CM	3

H. Graeb, S. Zizala, J. Eckmueller, and K. Antreich, "The Sizing Rules Method For Analog Integrated Circuit Design," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp. 343–349, 2001

Sizing Rules Method

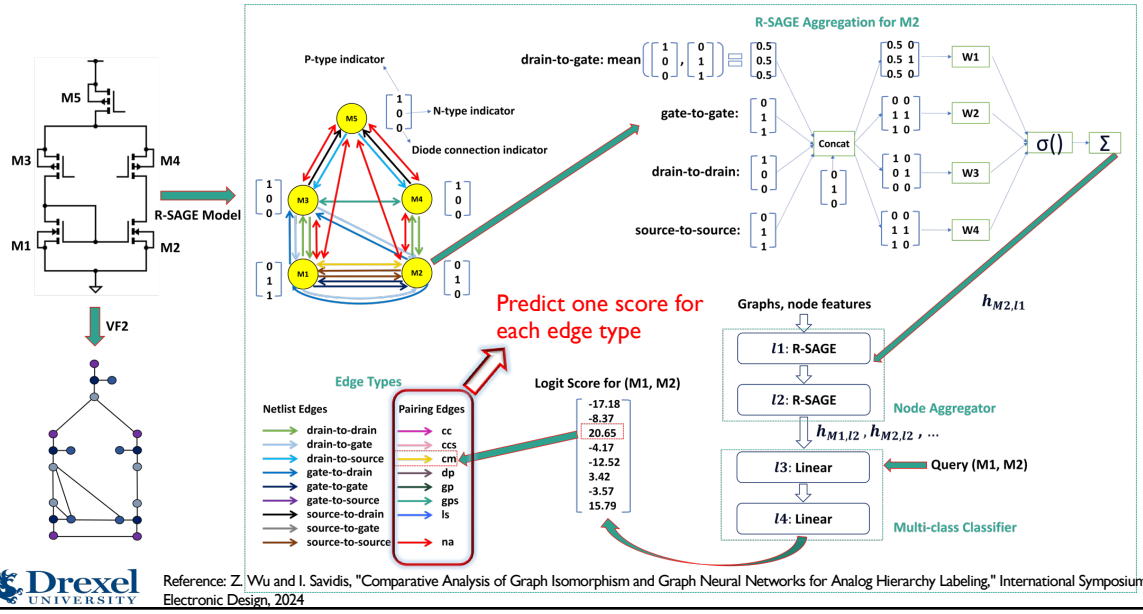
- Constraints on four types of circuit properties



- 30 generic constraints listed for the circuit subblocks
  - Examples of sizing rules:
    - Level 0: Constraints on transistor biasing
    - Level 2: Symmetry and matching constraints on current mirrors

H. Graeb, S. Zizala, J. Eckmueller, and K. Antreich, "The Sizing Rules Method For Analog Integrated Circuit Design," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp. 343–349, 2001

Labeling Analog Functional Groups with GNNs



Reference: Z. Wu and I. Savidis, "Comparative Analysis of Graph Isomorphism and Graph Neural Networks for Analog Hierarchy Labeling," International Symposium on Quality Electronic Design, 2024

129

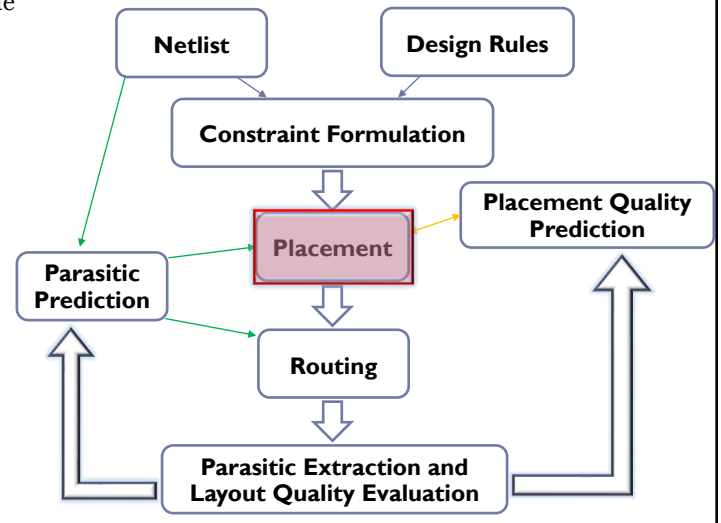
Outline of Presentation

- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
  - Application 1: component sizing of analog ICs
  - Application 2: analog circuit topology generation
  - Application 3: constraint formulation for P&R
  - Application 4: placement
  - Application 5: routing
  - Application 6: device modeling
  - Application 7: RF matching network synthesis
  - Application 8: LLMs for analog circuit generation
  - Application 9: prediction of interconnect impedance
- Analog Automation Platforms
- Conclusions

130

Analog Placement

- Analog placement: before routing, determine the locations of each instance
- Objective: minimize area and expected wirelength, mitigate layout effects on performance
- Heuristic approach:
  - Step 1: Represent a placement solution with a data structure
    - Examples:
      - O-tree
      - Segment tree
  - Step 2: Apply optimization algorithms to perturb the data structure in search of optimal solutions
    - Mixed integer linear programming
    - Nonlinear programming



Drexel University

131

Considerations for Analog Placement

- Matching Constraints
  - Symmetry for noise rejection and matching
  - Length-matching
  - Impedance-matching
- Parasitic Sensitivity
  - Interconnect impedance
    - IR drop
    - Signal coupling
  - Feedback from simulation or estimation
- Hierarchical and Modular Complexity
  - Hierarchical blocks with internal dependencies
  - block-level optimal floorplanning
- Process and Technology Constraints
  - Restrictive design rules in advanced nodes

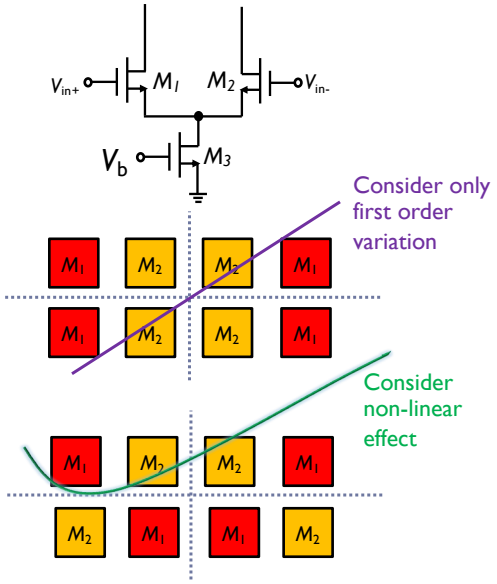
Drexel University

132

Placement Optimization Considering Nonlinear Spatial Variations

134

- Minimize nonlinear spatial variation effects in analog circuits
- Address limitations of common-centroid (CC) layout in mitigating higher-order spatial variations
- Initial placement
  - Sequential pattern ensuring no diffusion break
- Perturbation strategy
  - Swaps only allowed between units from different devices
  - Swap if:
    - Improves the objective
    - Probabilistically accepted via the Metropolis criterion
    - No increase in diffusion breaks

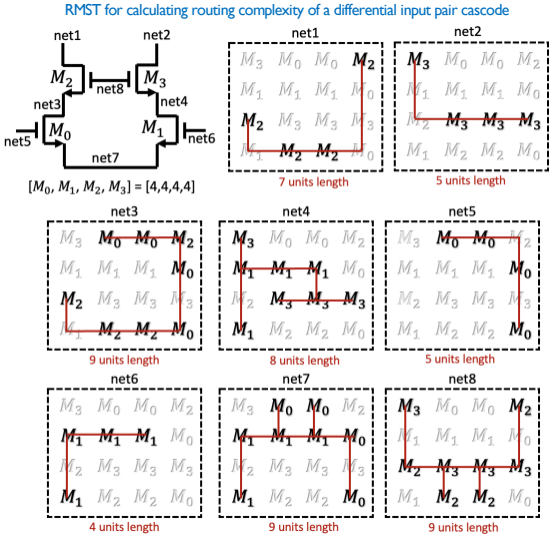


134

Placement Optimization Considering Nonlinear Spatial Variations

135

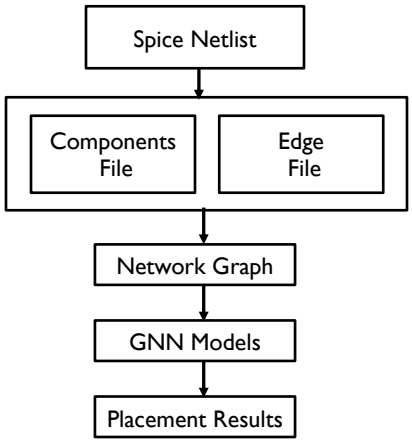
- Spatial Variation Mismatch
  - Approximated using a second-order Taylor series model
  - Difference between spatial variation responses of device pairs is minimized
- Routing Complexity
  - Estimated using Rectilinear Minimum Spanning Tree (RMST) on device connection graphs
  - RMST edge weights are summed across all nets
- Ensures no diffusion break
  - Enforcing valid source/drain continuity
  - Inserting dummy units if required
  - Scanning rows and correcting inconsistencies iteratively



135

Graph-Based Analog Placement

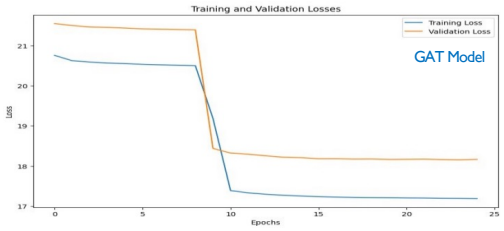
- Automate analog circuit placement using Graph Neural Networks (GNNs)
- Reduce component overlap and wirelength through learned spatial optimization
- Leverage GraphSAGE and Graph Attention Network (GAT) architectures to capture circuit connectivity and optimize placement

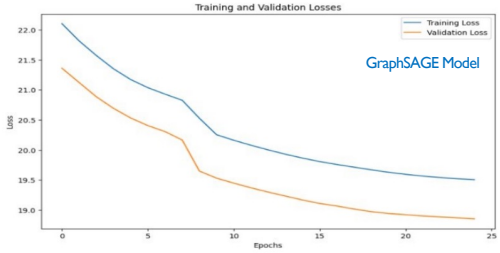


Yogitha H. K., Sinchana K., Shristi Biyani, and Sunita M. S. 2024. Optimization of Analog Circuit Placement: A Graph Neural Network Approach. In Proceedings of the 2024 10th International Conference on Computing and Artificial Intelligence (ICCAI '24). Association for Computing Machinery, New York, NY, USA, 399–403.

Graph-Based Analog Placement

- Netlist Parsing and Encoding
  - Component information
  - Net connections
- Graph Construction
  - Parsed data is transformed into graphs
  - Each node represents a component, and each edge represents a net connection
- GraphSAGE Model
  - Aggregates local neighborhood features to learn positional embeddings
- GAT
  - Applies attention weights to neighbors, capturing variable importance

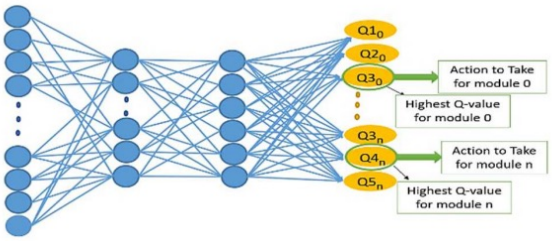




Yogitha H. K., Sinchana K., Shristi Biyani, and Sunita M. S. 2024. Optimization of Analog Circuit Placement: A Graph Neural Network Approach. In Proceedings of the 2024 10th International Conference on Computing and Artificial Intelligence (ICCAI '24). Association for Computing Machinery, New York, NY, USA, 399–403.

Placement for FinFET Technology using Reinforcement Learning

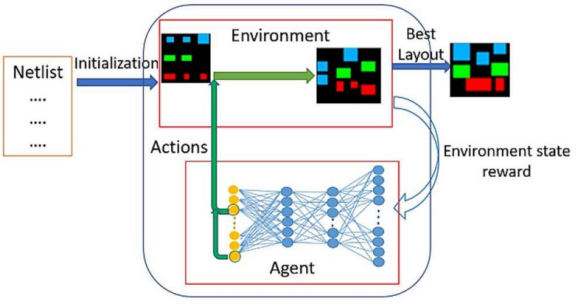
- Modeled as a Deep Q-Network
  - Input: Layout state (coordinates of modules)
  - Output: Q-values for 5 actions per module
  - Move: Up, Down, Left, Right, No move
- Trained using rewards based on placement quality
- Network layers
  - 2 Conv layers
  - 1 Max Pooling
  - 1 Fully Connected Layer



M. Ahmadi and L. Zhang, "Analog Layout Placement for FinFET Technology Using Reinforcement Learning," *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, Daegu, Korea, 2021, pp. 1-5

Placement for FinFET Technology using Reinforcement Learning

- Environment
  - Initializes modules across chip respecting FinFET rules
- Action
  - Executes module move if valid
  - No overlap or alignment maintained
  - Computes reward based on cost function
  - Updates Q-values for learning next action
- Reward Strategy
  - Valid actions that reduce cost below a are rewarded
  - Invalid actions (e.g., creating overlap or misalignment) are penalized
  - Encourages the agent to learn symmetry-aware, legal, compact layouts



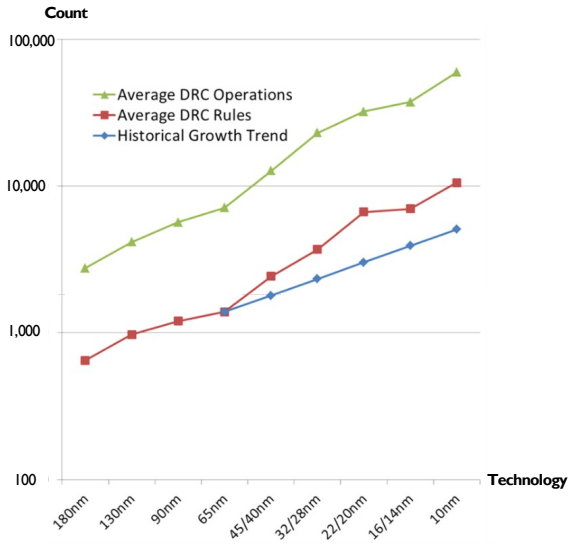
M. Ahmadi and L. Zhang, "Analog Layout Placement for FinFET Technology Using Reinforcement Learning," *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, Daegu, Korea, 2021, pp. 1-5

Outline of Presentation

- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
  - Application 1: component sizing of analog ICs
  - Application 2: analog circuit topology generation
  - Application 3: constraint formulation for P&R
  - Application 4: placement
  - Application 5: routing
  - Application 6: device modeling
  - Application 7: RF matching network synthesis
  - Application 8: LLMs for analog circuit generation
  - Application 9: prediction of interconnect impedance
- Analog Automation Platforms
- Conclusions

Analog Routing

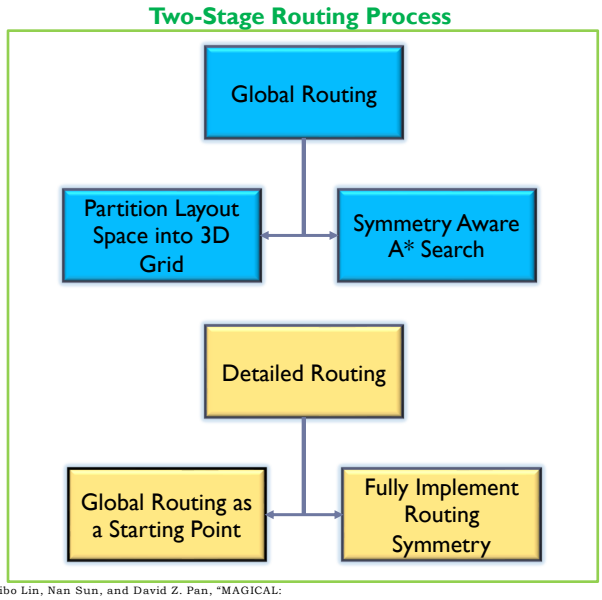
- Objective: minimize wirelength and area while optimizing circuit performance
- Primary considerations (constraints for optimization)
  - Design rules
    - Size rules (e.g., minimum width)
    - Separation rules
    - Overlap rules
  - Matching for symmetry
    - Example: common-centroid design of differential pairs
    - Impedance-matching
  - Effects of interconnect impedance (R,L,C) and coupling capacitance on performance parameters
    - Often not translating to geometric constraints directly
    - Example: minimizing wirelength does not necessarily result in optimal performance



Source Semiengineering.com

Two-Stage Routing Process: Global Routing and Detailed Routing

- Layout space partitioned into a 3D grid structure
  - Each grid cell represents a potential routing tile
- Routing graph consists of vertices (grid cells) and edges (possible paths)
  - Model routing availability and congestion.
- Symmetry-aware A* search algorithm to route each net
  - Enforce symmetry for matched signal nets Ensuring routing feasibility in congested areas through free space modeling

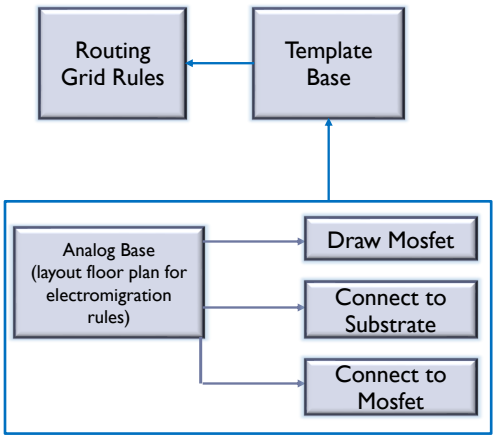


Hao Chen, Mingjie Liu, Biying Xu, Keren Zhu, Xiyuan Tang, Shaolan Li, Yibo Lin, Nan Sun, and David Z. Pan, "MAGICAL: An OpenSource Fully Automated Analog IC Layout System from Netlist to GDSII," IEEE Design Test, Vol. 38, No. 2, pp. 19–26, Apr. 2021.

142

Routing Grid Strategy for Process Portability

- Enforced routing grid system
  - Wire width and spacing are quantized
  - All wires on the same metal layer must travel in the same direction
  - Adjacent metal layers use orthogonal directions
- Design Rule Check
  - Adjusting the routing grid parameters
  - No need to redesign the layout
  - Portable across different process nodes
- Template Base is the fundamental class that all layout generators are based on
- Analog Base provides methods to create instances and connections

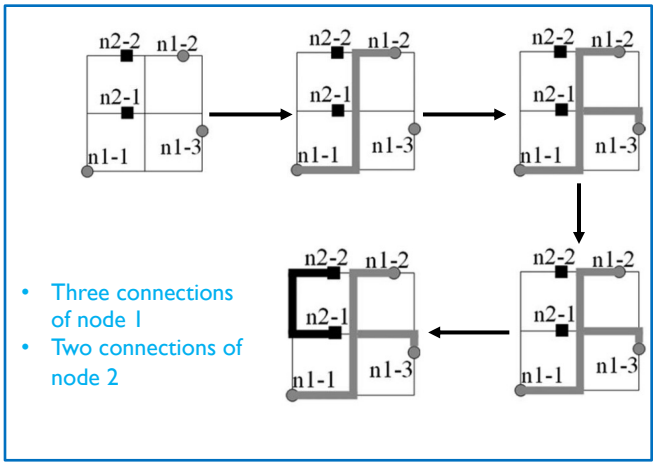


Eric Chang, Jaeduk Han, Woorham Bae, Zhongkai Wang, Nathan Narevsky, Borivoje NikoliC, and Elad Alon, "BAG2: A process-portable framework for generator-based AMS circuit design," pp. 1–8, May 2018.

143

Determines Logical Wire Paths across the Layout

- Channel intersection graph model
  - Layout region is abstracted to intersections between placed modules
  - Wires are projected to grid points for routing feasibility
- Iterative maze routing algorithm
  - Explores all routing options using a backtracking-based solver
  - Evaluates cost for each routing option
    - Wire length
    - Side-by-side length
  - Prioritizes connections that minimize performance degradation



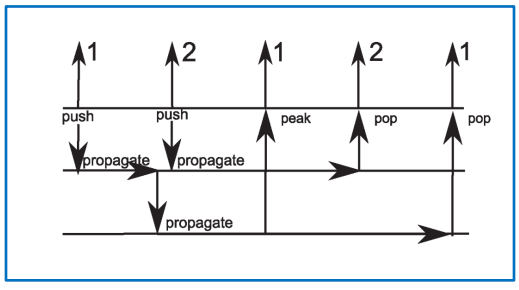
 Ender Yilmaz and GUnhan Dundar, "Analog Layout Generator for ¨ CMOS Circuits," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 28, No. 1, pp. 32–45, Dec. 2009.

144

Realizes the Physical Wires within Routing Channels

- Converts global routing paths into actual metal wires
  - Use routing channels between modules
  - Channels constructed using four basic operations
    - Push, Propagate, Peak, Pop
- Implements a switchbox-based routing scheme
  - Small 3×3 switchboxes inserted at transitions or junctions
  - Routing completed by filling spaces between switchboxes
- Algorithm explores all switchbox configurations
  - DRC compliance
  - Parasitic minimization

Routing plan generation on a simple one-sided routing example

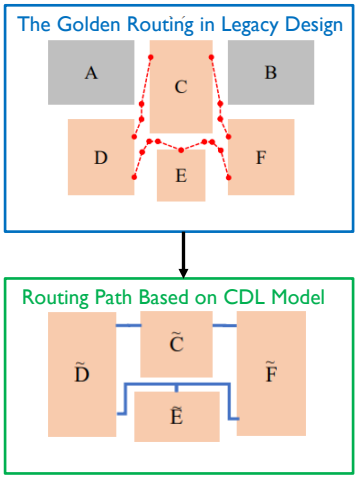


 Ender Yilmaz and GUnhan Dundar, "Analog Layout Generator for ¨ CMOS Circuits," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 28, No. 1, pp. 32–45, Dec. 2009.

145

Block-Level Routing Flow

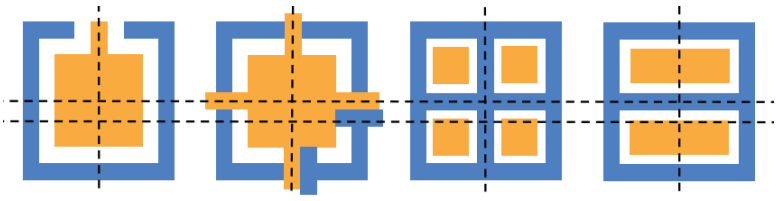
- Building Block Layout Migration
  - Extracts routing patterns from legacy building blocks
  - Reusable layout template library
  - Migrates templates to new circuits using building block matching
- Topology Preservation via Cartesian Detection Line (CDL)
  - Extract routing paths and preserve inter-block wire trajectories
  - Routing paths between blocks similar to legacy layout
  - Maintains routing direction, track usage, and block connections



M. P. -H. Lin *et al.*, "Achieving Analog Layout Integrity through Learning and Migration Invited Talk," *2020 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, San Diego, CA, USA, 2020, pp. 1-8.

Routing Strategies in Common-Centroid (CC) Layouts

- Preserve matching by maintaining symmetrical interconnect paths
- Cancels systematic process variation through balanced routing symmetry
- Minimizes mismatch due to parasitics and layout-dependent effects (LDEs)



Example of Common-Centroid Layout

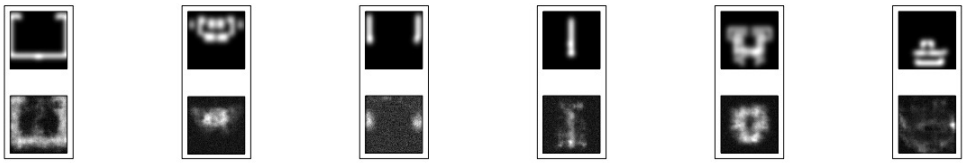
P. J. A. Harpe, et al., "A 26uW 8 bit 10 MS/s asynchronous SAR ADC for low energy radios," IEEE Journal of Solid-State Circuits (JSSC), vol. 46, no. 7, pp. 1585–1595, 2011.

GeniousRoute: ML-guided Routing

- Algorithm: Variational auto-encoders (VAEs) trained on layout images
  - Encoder: map input image to low-dimensional space
  - Decoder: generate routing guidance
  - Label: routing region of nets
- Routing prediction: for a given placement, VAE predicts the probability map that a wire is placed in a region
- Routing algorithm: A∗ search algorithm guided by the trained VAE model
- Limitation: GeniusRoute is trained on a dataset consisting of comparators and amplifiers without generalizing to other analog circuit types

ground-truth

inference



Drexel UNIVERSITY

K. Zhu. et al., "GeniusRoute: A New Analog Routing Paradigm Using Generative Neural Network Guidance," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design (ICCAD), pp. 1-8, 2019

149

Outline of Presentation

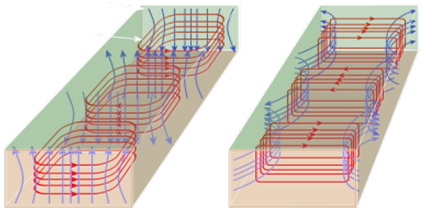
- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
  - Application 1: component sizing of analog ICs
  - Application 2: analog circuit topology generation
  - Application 3: constraint formulation for P&R
  - Application 4: placement
  - Application 5: routing
  - Application 6: device modeling
  - Application 7: RF matching network synthesis
  - Application 8: LLMs for analog circuit generation
  - Application 9: prediction of interconnect impedance
- Analog Automation Platforms
- Conclusions

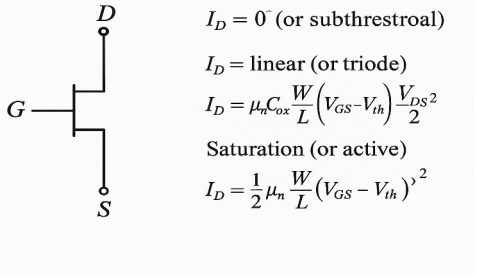
Drexel UNIVERSITY

150

What is Device and EM Modeling Synthesis

- Device modeling
  - Predicting how electronic properties of components behave (current, voltage, capacitance)
- EM synthesis
  - Designing structures where electromagnetic fields determine performance
- Challenge:
  - Models fail to generalize across device structures and conditions
  - Parameter extraction remains slow and requires expert tuning
  - Physically inaccurate or non-smooth model predictions under bias sweeps



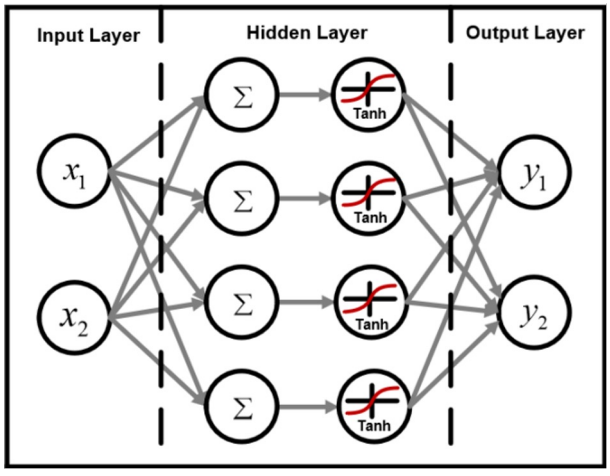


Drexel UNIVERSITY

151

Graph Based Compact Model

- The model is built as a graph of functional blocks, each representing a physical quantity
- Each block is computed by:
  - A compact model equation, or
  - A small MLP
- Structure:
  - Nodes = physical effects or quantities
  - Edges = physical dependencies
- Executed in sequential order, not learned via message passing
- Key benefit:
  - Structuring the model into modular blocks may help isolate distinct behaviors, which could improve interpretability by aligning the architecture with known physical dependencies.



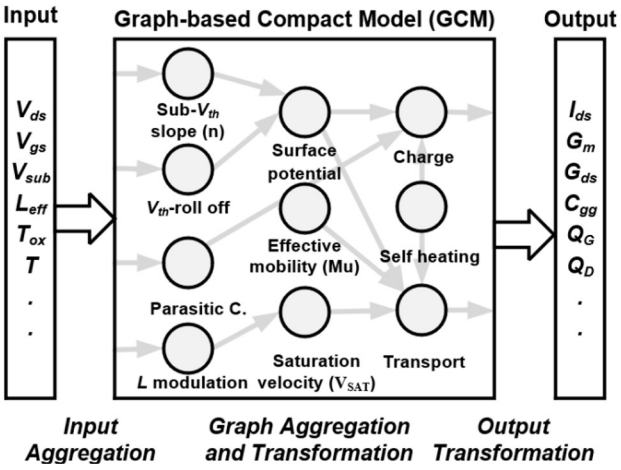
Drexel UNIVERSITY

Z. Yang, A. D. Gaidhane, K. Anderson, G. Workman, and Y. Cao, "Graph-based compact model (GCM) for efficient transistor parameter extraction: A machine learning approach on 12 nm finfets," *IEEE Transactions on Electron Devices*, vol. 71, no. 1, pp. 254–262, Nov. 2024.

152

Graph Based Compact Model

- This is not…
  - A traditional neural net:
    - GCM is not a single large black-box model
    - Each part of the device is modeled separately and explicitly
  - or a GNN either:
    - The "graph" describes model structure, not graph-learning (no message passing)
    - Each node computes independently based on inputs from upstream nodes
- Why this setup might help:
  - Modular structure may encourage the model to better learn localized behaviors
  - Separating computations may help reduce parameter entanglement

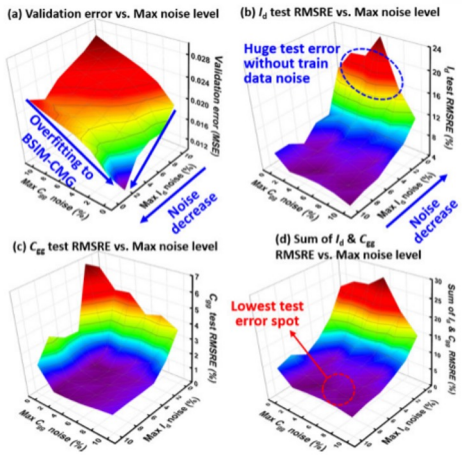


Z. Yang, A. D. Gaidhane, K. Anderson, G. Workman, and Y. Cao, "Graph-based compact model (GCM) for efficient transistor parameter extraction: A machine learning approach on 12 nm finfets," *IEEE Transactions on Electron Devices*, vol. 71, no. 1, pp. 254–262, Nov. 2024.

153

Multi-task Learning for BSIM Parameter Extraction of NSFETs

- Goal:
  - Multi-task model to predict I_D and C_{GG} across structural variations
- Setup:
  - Synthetic NSFET dataset with varied L_G, W_{NS}, T_{NS}
  - Inputs: Structural parameters + partial I_D and C_{GG} curves
  - Outputs: Full I_d–V_{GS}, I_d–V_{DS}, C_{GG}–V_{GS}
  - 9 unseen NSFETs for structural generalization test
- Training:
  - MTL with shared base layers and task-specific heads
  - Joint MSE loss for I_D and C_{GG}
  - Additive noise applied for robustness

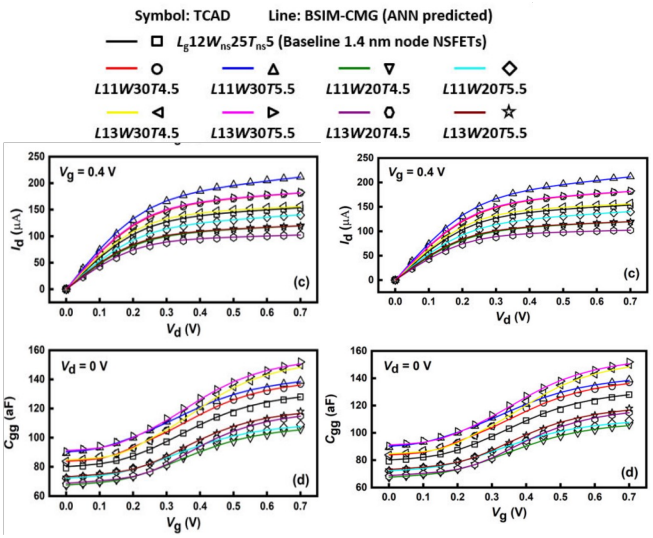


S. Lee, S. Eom, J. Jeong, J. Lee, S. Lee, H. Yun, Y. Ahn, and R.-H. Baek, "Multi-task learning for real-time bsim-cmg parameter extraction of nsfets with multiple structural variations," *IEEE Access*, vol. 12, pp. 184 619–184 630, Dec. 2024.

154

Results of Applying Multi-task Learning for BSIM Parameter Extraction

- Prediction Performance
  - Test RMSRE:
    - I_D = 4.26%
    - C_{GG} = 0.709%
    - Low error confirms MTL can simultaneously learn both outputs accurately
  - Bias range covered:
    - V_GS sweeps for I_D and C_{GG}
    - V_DS = 0.7 V for I_D,
    - V_DS = 0 V for C_{GG}
- Full electrical behavior reconstructed from partial input data
  - Generalization to 9 Unseen Structural variations tested:
    - L_G (11, 12, 13 nm)
    - T_{NS} (4.5, 5.0, 5.5 nm)
    - W_{NS} (20, 25, 30 nm)
  - Result:
    - No retraining required to handle geometry changes
    - RMSRE remains stable (~4–5% for I_D, ~0.7% for C_{GG})
    - Confirms robustness across device design space

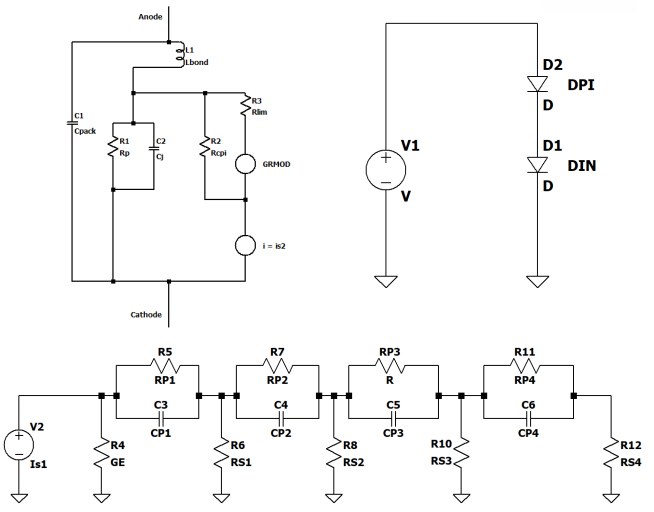


S. Lee, S. Eom, J. Jeong, J. Lee, S. Lee, H. Yun, Y. Ahn, and R.-H. Baek, "Multi-task learning for real-time bsim-cmg parameter extraction of nsfets with multiple structural variations," *IEEE Access*, vol. 12, pp. 184 619–184 630, Dec. 2024.

155

Experimental Setup and Training

- Goal
  - Reconstruct the internal model components of the PIN diode
- Setup:
  - PIN diode modeled as two junctions + 6th-order RC base network (GRMOD)
  - 200 simulations with parameter sweeps (τ: 2–50 ns, C_j: 1 pF–1 nF, R_p: 10 mΩ–1 Ω)
  - Input: voltage waveform under 10 A, 100 ns current pulse
  - Output: Output voltage and current
- Training:
  - Autoencoder with 5-layer encoder, 3-layer decoder, latent size = 8
  - Evaluated by waveform RMSE and parameter MAE

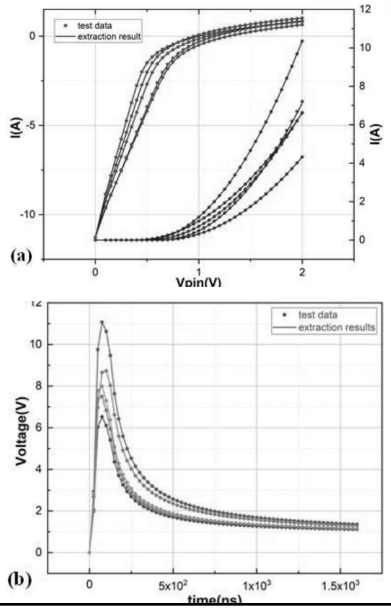


Y. Li, W. Dai, K. Geng, L. Zhang, R. Wang and R. Huang, "An Automatic Parameter Extraction Method Based on Autoencoder for PIN Diode Model," *2022 IEEE 16th International Conference on Solid-State & Integrated Circuit Technology (ICSICT)*, Nangjing, China, pp. 1-3, Oct. 2022

156

Results

- Normalized mean squared error (MSE):
  - Training set: 0.77%
  - Test set: 0.56%
- Accurate reproduction of key waveform features
- Evaluated across 200 parameter-swept SPICE simulations
- Valid under variation of τ, C_j, R_p, R_{mod}
- Confirms reliability of extracted parameters in modeling dynamic diode behavior

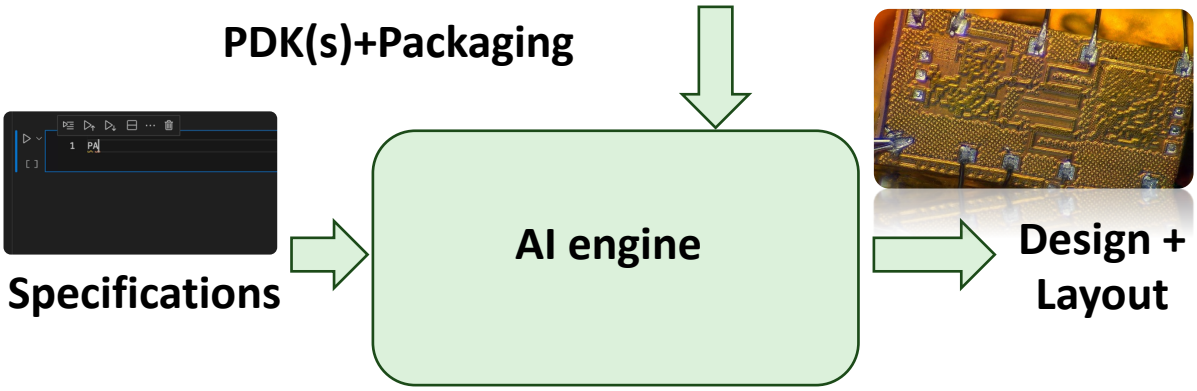


Y. Li, W. Dai, K. Geng, L. Zhang, R. Wang and R. Huang, "An Automatic Parameter Extraction Method Based on Autoencoder for PIN Diode Model," *2022 IEEE 16th International Conference on Solid-State & Integrated Circuit Technology (ICSICT)*, Nangjing, China, pp. 1-3, Oct. 2022

Outline of Presentation

- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
  - Application 1: component sizing of analog ICs
  - Application 2: analog circuit topology generation
  - Application 3: constraint formulation for P&R
  - Application 4: placement
  - Application 5: routing
  - Application 6: device modeling
  - Application 7: RF matching network synthesis
  - Application 8: LLMs for analog circuit generation
  - Application 9: prediction of interconnect impedance
- Analog Automation Platforms
- Conclusions

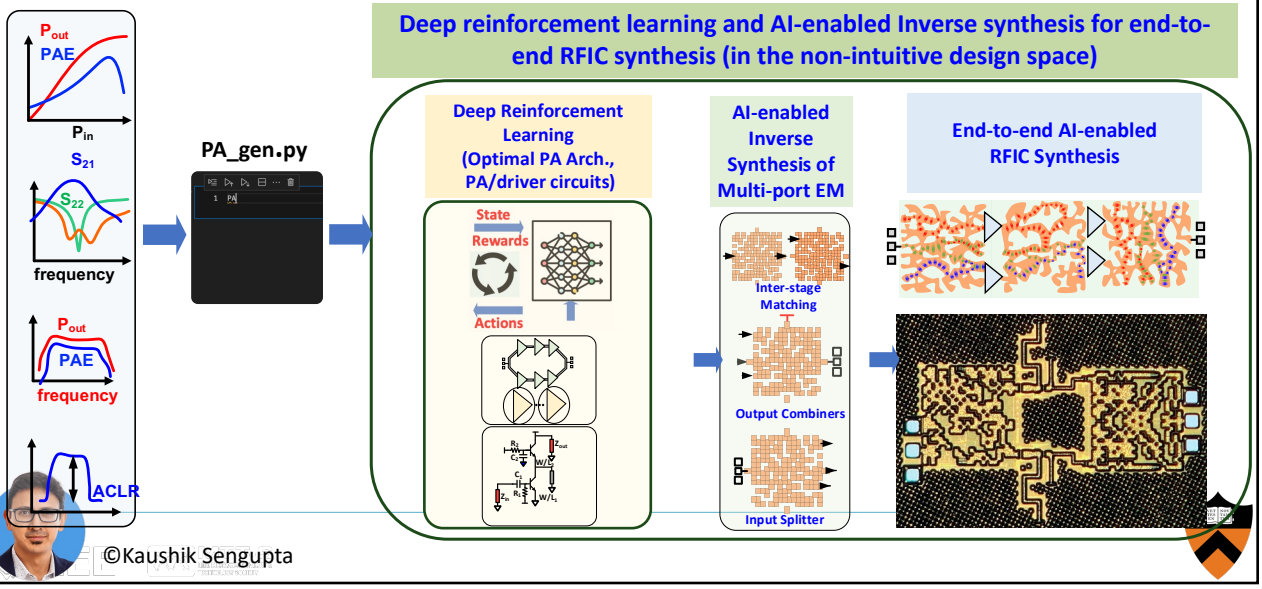
AI-enabled RFIC Design



- Rapid Designs (months/years to days/weeks).
- Invents new topologies beyond human intuition/knowledge.

160

RL+Inverse Design: End-to-end RFIC Design



166

US Chips and Science Act

Natcast

PRINCETON UNIVERSITY

USC cadence Qualcomm Texas Instruments

Drexel University KEYSIGHT TECHNOLOGIES NOKIA

Northeastern University GlobalFoundries ERICSSON

SKYWORKS

Raytheon An RTX Business EDGE Maury Microwave

©Kaushik Sengupta

ARTIFICIAL INTELLIGENCE DRIVEN RADIO FREQUENCY INTEGRATED CIRCUIT DESIGN ENABLEMENT (AIDRFIC) PROGRAM

This NSTC research and development program will address the domestic RFIC (Radio Frequency Integrated Circuit) industry and is focused on the adaptation of Artificial Intelligence (AI) and Machine Learning (ML) technology for use in RF design. Its objective is the successful demonstration of AI-based tools used to improve design productivity for RFICs thus lowering the risk of further investment in commercialization of the technology.

ADAPTING AI AND ML TECHNOLOGY FOR USE IN RF IC DESIGN

Radio Frequency Integrated Circuits (RFICs) are critical to many commercial systems provide wide-ranging functionality including low-noise and power amplification, phase shifting, mixing, switching, and filtering, to name a few. The emphasis of AIDRFIC is on the Circuit Design and Physical Design components of the flow for RF building blocks and corresponding System Verification of the higher-level system. AI-enabled tools for these components will improve the productivity of the design process, potentially allowing for greater focus on system-level partitioning and trade-offs.

170

Outline of Presentation

172

- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
  - Application 1: component sizing of analog ICs
  - Application 2: analog circuit topology generation
  - Application 3: constraint formulation for P&R
  - Application 4: placement
  - Application 5: routing
  - Application 6: device modeling
  - Application 7: RF matching network synthesis
  - Application 8: LLMs for analog circuit generation
  - Application 9: prediction of interconnect impedance
- Analog Automation Platforms
- Conclusions

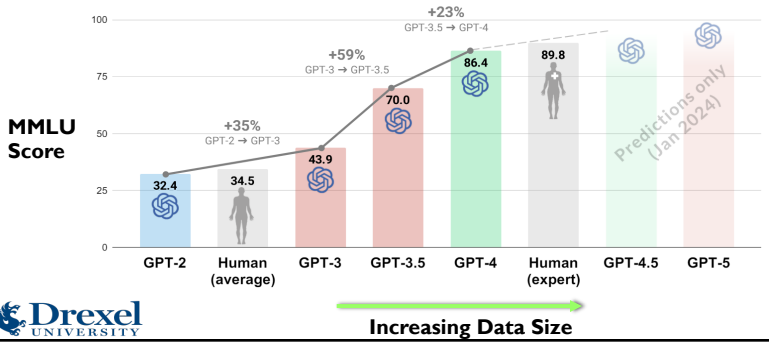
Drexel University

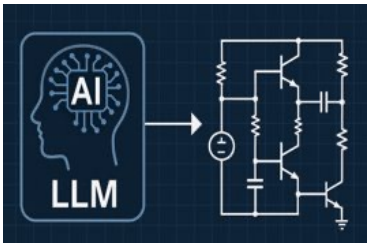
172

Large Language Models and Generative AI for EDA, the Future?

- LLMs are pre-trained models
  - More effective when fine-tuned on domain-specific data
- Works have been published in the past year in applying LLMs to EDA problems, but mostly for:
  - RTL and logic synthesis, HDL code debugging
  - Testbench generation
  - Scripting
- Success of LLMs is contingent on large amount of data
  - Similarly, to train a 'Circuit Net', the EDA community needs a large benchmark circuit dataset with standardized representation for both digital and analog

not for physical design yet

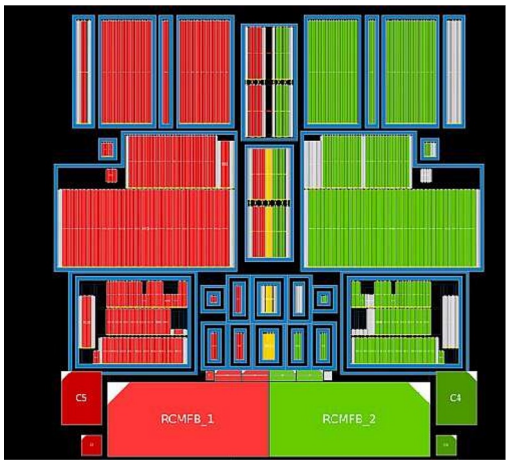




Source: Lifearchitect.ai

GLayout

- Goal:
  - Enable general-purpose analog layout generation from human language using a scalable, open-source LLM-based framework.
- Architecture
  - LLM Driven Prompt Interpretation
  - Strict Syntax Parsing and Compilation
  - Layout Generation Pipeline

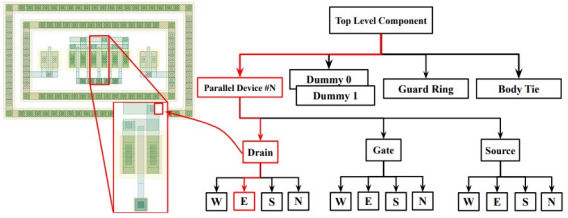


Source: planetanalog.com

Ali Hammoud, Chetanya Goyal, Sakib Pathen, Arlene Dai, Anhang Li, Gregory Kielian, and Mehdi Saligane, "Human Language to Analog Layout Using GLayout Layout Automation Framework," *ACM/IEEE 6th Symposium on Machine Learning for CAD (MLCAD)*, Salt Lake City (Snowbird), UT, USA, pp. 1-7 Sept 2024.

LLM-Driven Prompt Interpretation

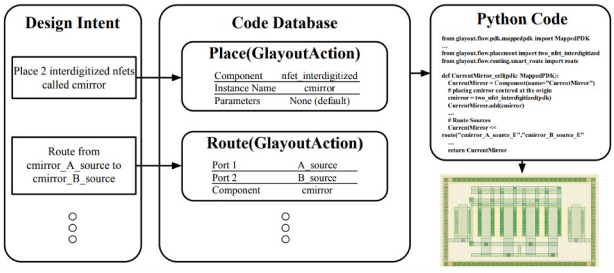
- LLM-Driven Prompt Interpretation
  - Natural language prompt from the user
  - LLM: Fine-tuned (3.8B–22B param) model interprets prompt using
    - Retrieval-Augmented Generation (RAG) and ICL(in context learning)
  - Output: Structured layout intent in strict syntax command format
- Domain-specific language defining layout operations:
  - create parameter, place, move, route, etc.
  - Parsing Engine:
    - Uses regex + CFG parser to extract command structure
    - Stores commands in a relational database
  - Compilation:
    - Translated via .get_code() method into Python templates
    - Combined into a hierarchical layout generator with full PDK support



Ali Hammoud, Chetanya Goyal, Sakib Pathen, Arlene Dai, Anhang Li, Gregory Kielian, and Mehdi Saligane, "Human Language to Analog Layout Using GLayout Layout Automation Framework," *ACM/IEEE 6th Symposium on Machine Learning for CAD (MLCAD)*, Salt Lake City (Snowbird), UT, USA, pp. 1-7 Sept 2024.

Layout Generation Pipeline

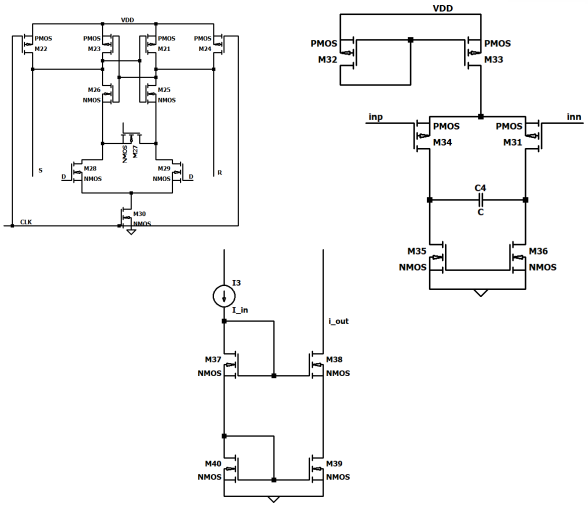
- Python Backend:
  - Uses GDSFactory + GLayout API to create GDSII layouts.
  - Supports advanced features like hierarchical Pcells, matched placement, routing macros.
- MappedPDK Abstraction:
  - Maps generic layer/rule identifiers to PDK-specific values.
  - Enables process portability without re-authoring layout code.



Ali Hammoud, Chetanya Goyal, Sakib Pathen, Arlene Dai, Anhang Li, Gregory Kielian, and Mehdi Saligane, "Human Language to Analog Layout Using GLayout Layout Automation Framework," *ACM/IEEE 6th Symposium on Machine Learning for CAD (MLCAD)*, Salt Lake City (Snowbird), UT, USA, pp. 1-7 Sept 2024.

Evaluation of the different models

- Task:
  - Generate strict syntax for unseen or re-styled analog circuits
- Models:
  - Phi-3 (3.8B), Mistral (7B), Codestral (22B)
  - Fine-tuned with QLoRA on <50 circuits
  - Trained for 2–4 epochs on a single A100 GPU
  - Loss on completions only (prevents overfitting to prompts)
- Scoring (per design):
  - 3 = Compiles + DRC + LVS
  - 2 = Compiles + DRC
  - 1 = Compiles only
  - 0 = Failure
- Evaluation Set:
  - 8 circuits, including 4-transistor layouts
  - Emphasis on generalization and robustness



Ali Hammoud, Chetanya Goyal, Sakib Pathen, Arlene Dai, Anhang Li, Gregory Kielian, and Mehdi Saligane, "Human Language to Analog Layout Using GLayout Layout Automation Framework," *ACM/IEEE 6th Symposium on Machine Learning for CAD (MLCAD)*, Salt Lake City (Snowbird), UT, USA, pp. 1-7 Sept 2024.

Results

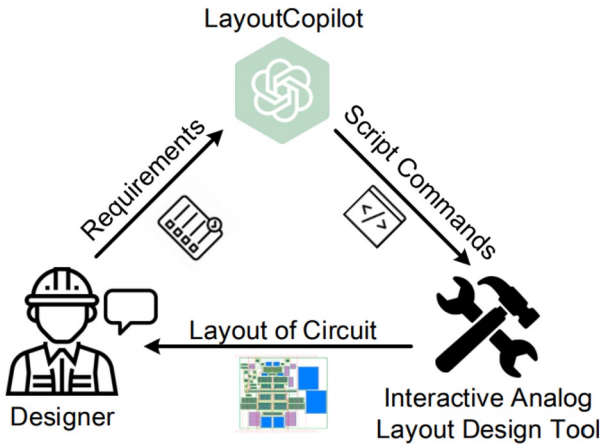
- Evaluation = code compiles + DRC + LVS
  - Task completion:
    - 3.8B → 10 tasks
    - 7B → 12 tasks
    - 22B → 18 tasks (70%)
- 22B model:
  - Strengths
    - 44% of tasks passed DRC + LVS
    - Valid outputs for up to 4-transistor layouts
    - Strong generalization to unseen topologies
  - Failure points:
    - Placement errors in large blocks• Overfitting after 1–2 epochs
    - LVS mismatches in dual-reference designsKey trend
    - Clear accuracy gain with larger model size
- Diminishing returns from 7B → 22B vs. 3.8B → 7B
- Largest performance gain seen from 3.8B → 7B
- 22B model produces working syntax within 1–2 prompts

Design	Difficulty	Assessment Goal	3.3B	7B	22B
P-type Diff Pair	easy	Different transistor type on known circuit	3	3	3
N-type Common Centroid Current Mirror	easy	Different placement technique on known circuit	1	3	3
Wilson Current Mirror	easy	New placement technique, new circuit	1	2	3
Mimcap Array	easy	New placement technique	2	3	3
Class AB Stage	medium	Different placement technique on known circuit	1	1	2
Integrator Stage	medium	Integrating known circuits, new circuit	0	1	1
Strong Arm Latch	medium	Integrating known circuits, new circuit	0	1	2
4 Stage Integrator	hard	Integrate a complex layout	0	0	1
ΔΣ Modulator	hard	Integrate a complex layout	0	0	0
			9	**14**	**18**

Ali Hammoud, Chetanya Goyal, Sakib Pathen, Arlene Dai, Anhang Li, Gregory Kielian, and Mehdi Saligane, "Human Language to Analog Layout Using GLayout Layout Automation Framework," *ACM/IEEE 6th Symposium on Machine Learning for CAD (MLCAD)*, Salt Lake City (Snowbird), UT, USA, pp. 1-7 Sept 2024.

LayoutCopilot

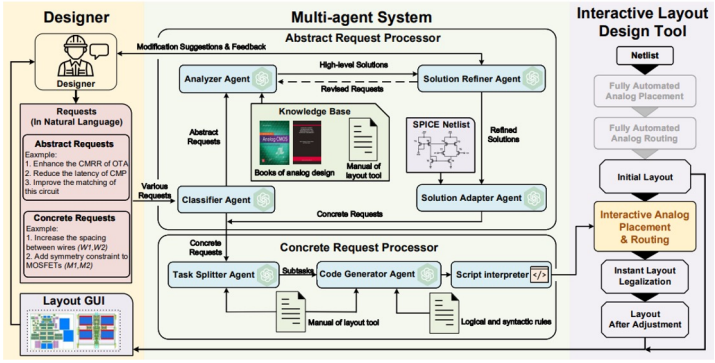
- Goal:
  - LLM-powered framework for interactive analog layout design.
- Motivation:
  - Fully automated tools lack flexibility for high-performance analog design.
  - Existing interactive tools still require scripting expertise.
- Core Idea:
  - Leverage LLMs and multi-agent collaboration to enable natural-language-driven layout editing, improving design efficiency.



Bingyang Liu, Haoyi Zhang, Xiaohan Gao, Zichen Kong, Xiyuan Tang, Yibo Lin, Runsheng Wang, and Ru Huang, "LayoutCopilot: An LLM-Powered Multi-Agent Collaborative Framework for Interactive Analog Layout Design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. early access, pp. 1–1, 2025

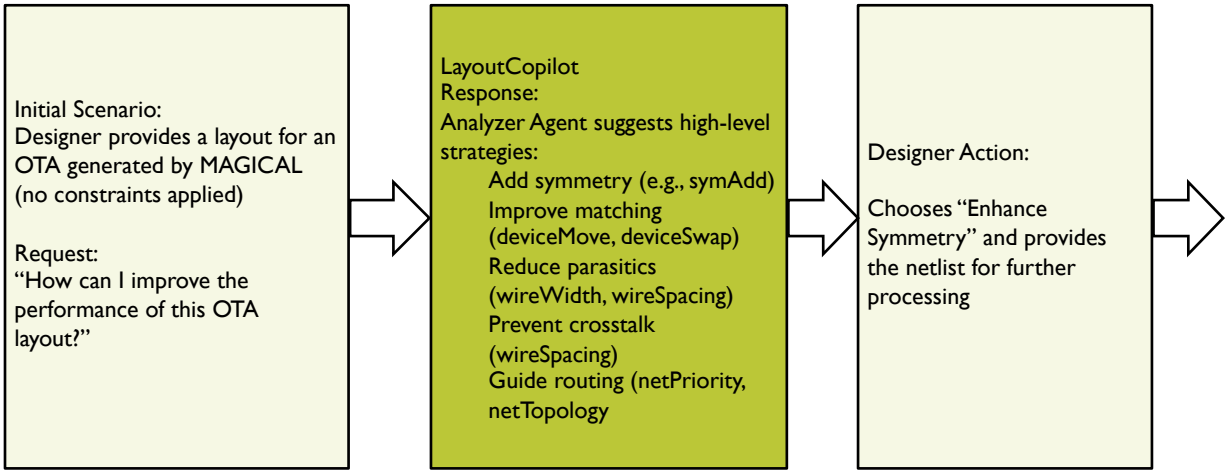
Architecture

- Abstract Request Processor(split into 4 internal agents)
  - Classifier Agent:
    - Categorizes requests as abstract vs. concrete
  - Analyzer Agent:
    - Retrieves relevant domain knowledge and proposes high-level strategies
  - Solution Refiner Agent:
    - Interfaces with the designer to refine or revise suggested actions
  - Solution Adapter Agent:
    - Converts strategies into concrete requests aligned with tool syntax
- Concrete Request Processor
  - Executes direct instructions (e.g., "add symmetry between M6, M7")
  - Decomposes tasks into subcommands
  - Ensures compliance with layout tool manuals
  - Delegates command generation to the Code Generator Agent
- Interactive Layout Editor
  - GUI + Command Interpreter
  - Maps commands to placement/routing operations
  - Updates layout in real time



Bingyang Liu, Haoyi Zhang, Xiaohan Gao, Zichen Kong, Xiyuan Tang, Yibo Lin, Runsheng Wang, and Ru Huang, "LayoutCopilot: An LLM-Powered Multi-Agent Collaborative Framework for Interactive Analog Layout Design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. early access, pp. 1–1, 2025

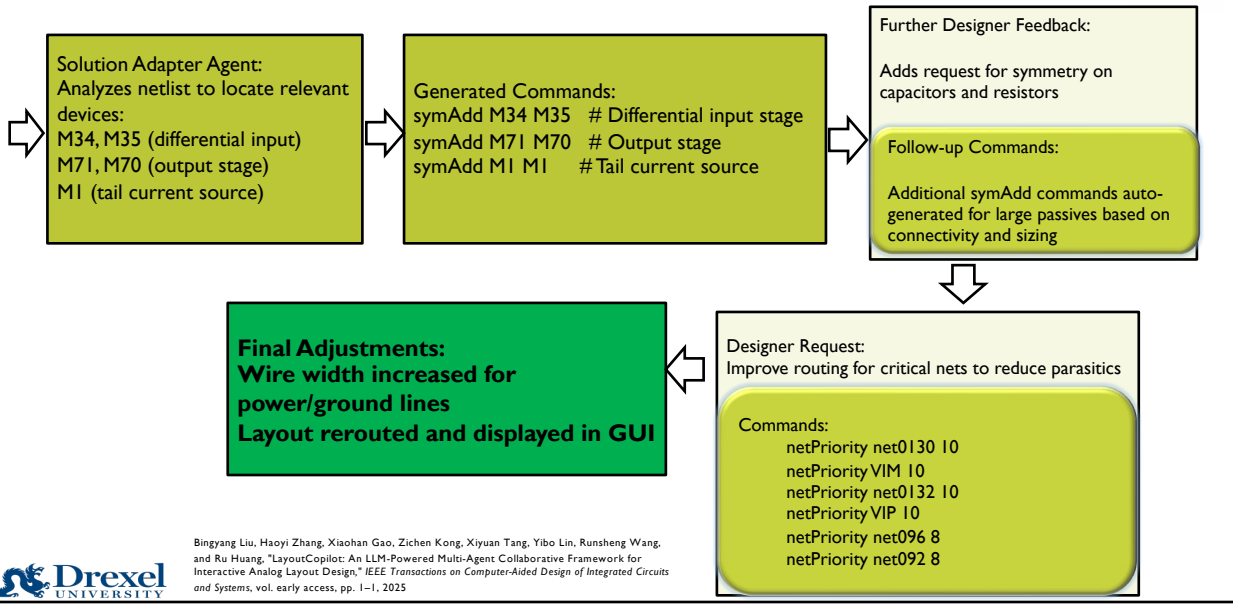
High-Level Interaction – OTA Optimization



Bingyang Liu, Haoyi Zhang, Xiaohan Gao, Zichen Kong, Xiyuan Tang, Yibo Lin, Runsheng Wang, and Ru Huang, "LayoutCopilot: An LLM-Powered Multi-Agent Collaborative Framework for Interactive Analog Layout Design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. early access, pp. 1–1, 2025

181

Concrete Request Formation and Execution



Bingyang Liu, Haoyi Zhang, Xiaohan Gao, Zichen Kong, Xiyuan Tang, Yibo Lin, Runsheng Wang, and Ru Huang, "LayoutCopilot: An LLM-Powered Multi-Agent Collaborative Framework for Interactive Analog Layout Design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. early access, pp. 1–1, 2025

182

Results and Comparisons

- Recovers OTA performance
  - Restores gain (38.26 dB vs. –8.75 dB)
  - Improves CMRR (58.7 dB vs. 27.3 dB).
  - Reduces noise (13.9 μV vs. 30.9 μV)

Benchmark		Schematic	MAGICAL w/o Constraints	LayoutCopilot
OTA	Gain(dB)	38.63	-8.75	**38.26**
	UGB(MHz)	6.85	-	**4.42**
	CMRR (dB)	-	27.3	**58.7**
	PM (degree)	70.98	-	**76.28**
COMP	CMP_Delay (ns)	3.3	6.3	**6.7**
	Noise (uV)	50.3	30.9	**13.9**
	RST_delay (ps)	89.8	165.8	**538.7**
	Power (uW)	19.9	32.0	**31.71**

Drexel UNIVERSITY

Bingyang Liu, Haoyi Zhang, Xiaohan Gao, Zichen Kong, Xiyuan Tang, Yibo Lin, Runsheng Wang, and Ru Huang, "LayoutCopilot: An LLM-Powered Multi-Agent Collaborative Framework for Interactive Analog Layout Design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. early access, pp. 1–1, 2025

183

Analog Circuit Layout with LLMs

Aspect	LayoutCopilot	GLayout
Model Used	GPT-style LLM (OpenAI GPT-4, API-based) within a multi-agent framework	Fine-tuned Flan-T5 (encoder-decoder)
Training	No LLM fine-tuning	Supervised fine-tuning (QLoRA, 8-bit) on <50 circuit examples
Usage	Agents interpret natural-language edits and generate actions	LLM translates prompt into strict syntax layout code
Inputs	User requests (e.g., "improve CMRR") with optional netlist or layout context	Natural language prompt (e.g., "create a differential pair layout")
Outputs	Sequences of layout commands(e.g., symAdd, route)	Parameterized Python functions via GDSFactory and GLayout API
LLM Role	Task-specific agents: classifier, analyzer, refiner, code generator	Monolithic generation: one model generates layout commands directly
System Design	Multi-agent with isolated tasks	Single model handling end-to-end generation

Drexel UNIVERSITY

184

Outline of Presentation

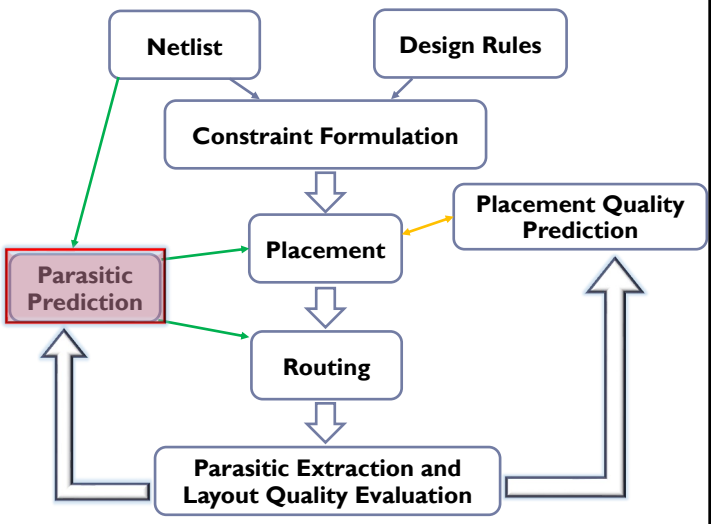
- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
  - Application 1: component sizing of analog ICs
  - Application 2: analog circuit topology generation
  - Application 3: constraint formulation for P&R
  - Application 4: placement
  - Application 5: routing
  - Application 6: device modeling
  - Application 7: RF matching network synthesis
  - Application 8: LLMs for analog circuit generation
  - Application 9: prediction of interconnect impedance
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

194

Case Study: Prediction of Interconnect Impedance

- Interconnect impedances have big impact on circuit performance
  - Results in gap between schematic and layout simulation results
- Most interconnect models are analytical
  - Few ML-based techniques
- Apply ML to predict parasitic impedances
  - Reduce error between pre-layout and post-layout simulation
    - Guide placement and routing
  - Reduce parasitic extraction simulations required
- Predict post-routing interconnect values at different stages of analog synthesis
  - At schematic level
  - At placement level



Drexel UNIVERSITY

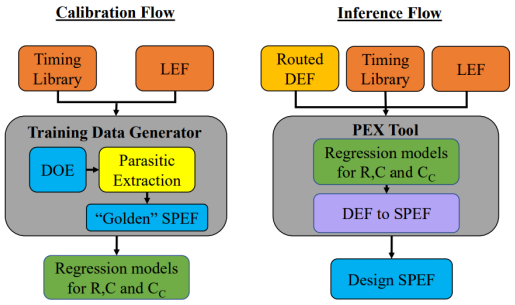
195

A Machine Learning Based Parasitic Extraction Tool

- Algorithm: regression
- Target circuit parameters for prediction: resistance, capacitance to ground, coupling, crossover, and crossunder capacitance of a net
- Training data is generated from Cadence Innovus with design of experiment (DOE)
  - Not exclusively for analog but provides physical modeling of interconnect capacitances
- Regression function is fixed
  - Only fitting regression parameters on data
  - Example: coupling capacitance expression

$$C_c = c/s + d \cdot l_{overlap} + e \cdot l_{overlap}/s + f$$

  - Inflexible to model interconnects at advanced technologies
- No results reported

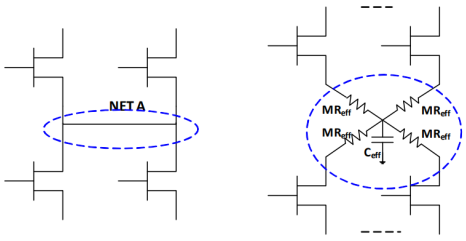


G. Pradipta, V. A. Chhabria, and S. S. Sapatnekar, "A Machine Learning Based Parasitic Extraction Tool," 2019

196

MLParest: ML-based Parasitic Estimation for Custom Circuit Design

- Algorithm: random forest
- Apply fixed interconnect star model



- Error between pre-layout and post-layout circuit simulation is reduced from 37% to 8% on benchmark analog circuits
- Limitation: inflexibility of the interconnect model
  - Only effective for timing prediction

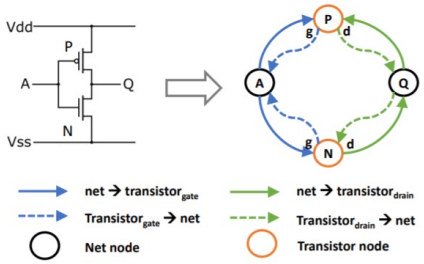
B.W. Shook. et.al, "MLParest: Machine Learning based Parasitic Estimation for Custom Circuit Design," Proceedings of the ACM/IEEE Design Automation Conference, pp. 1-6, 2020

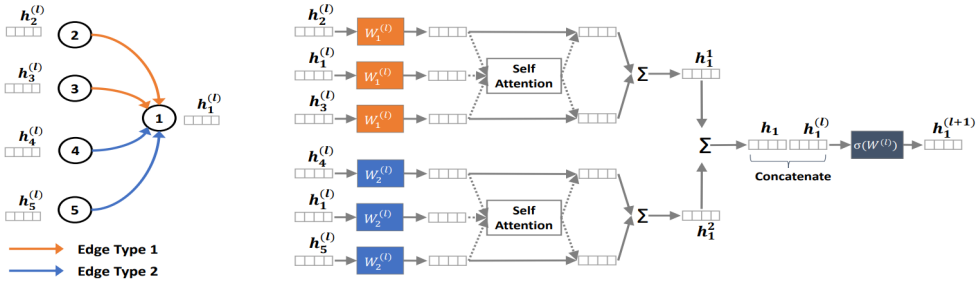
197

ParaGraph: Layout Parasitics and Device Parameter Prediction with GNNs

198

- Graph representation of a circuit
  - Heterogeneous graph: devices and nets both as graph nodes
- Multiple sub-models for different capacitance ranges
- Transistor features:
  - gate poly length
  - number of fingers
  - number of fins
  - multiplier





H. Ren, G. F. Kokai, W. J. Turner and T. Ku. "ParaGraph: Layout Parasitics And Device Parameter Prediction Using Graph Neural Networks", Proceedings of the ACM/IEEE Design Automation Conference, pp. 1–6, 2020

198

ParaGraph: Layout Parasitics and Device Parameter Prediction with GNNs

199

- Simulation errors between pre-layout predictions and post-layout on 67 circuit metrics in the testing circuits:

Error Range	Layout w/o parasitics	Designer's Estimation	Prediction w/ XGB	Prediction w/ ParaGraph
< 10%	4	6	17	44
10%-20%	0	17	14	10
20%-30%	5	18	4	8
30%-40%	35	2	7	4
40%-50%	14	6	9	1
> 50%	9	18	16	0
Mean	37.75%	>100%	32.14%	9.60%
Geometric Mean	29.01%	43.57%	15.46%	4.00%

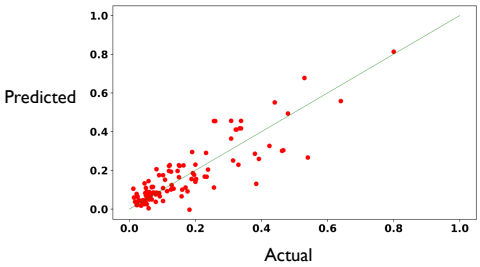
- GCN-based model achieves an average prediction R2 of 0.772 (110% better than XGBoost)
- Average simulation errors from over 100% with designer's estimation to less than 10%

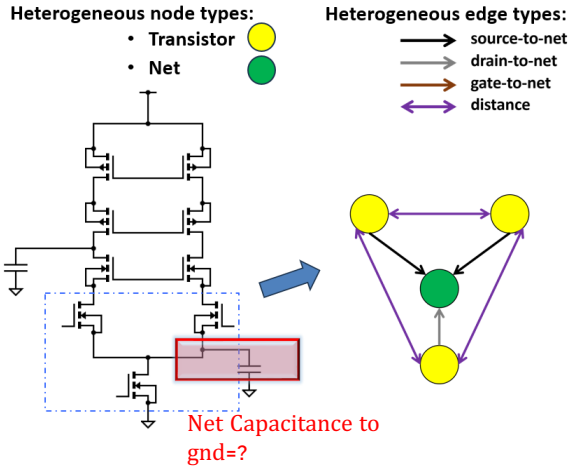
H. Ren, G. F. Kokai, W. J. Turner and T. Ku. "ParaGraph: Layout Parasitics And Device Parameter Prediction Using Graph Neural Networks", Proceedings of the ACM/IEEE Design Automation Conference, pp. 1–6, 2020

199

Analog/RF Parasitic Prediction Framework

- Apply edge-weighted GNNs for estimation of interconnect capacitance
  - Schematic level
  - Post-placement level
- Post-placement model leverages coordinates of placed devices
  - Euclidean distance between devices used as edge weights





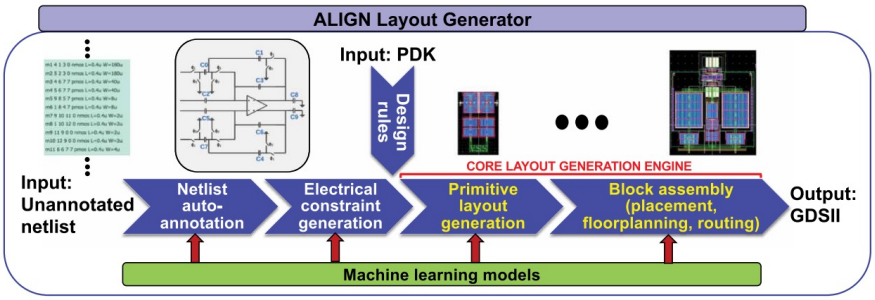
Z. Wu and I. Savidis, ": Edge-Weighted Graph Neural Networks for Post-Placement Interconnect Capacitance Estimation of Analog Circuits", Proceedings of the International Symposium on Circuits and Systems, pp. 1-5, May 2024.

Outline of Presentation

- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
- Analog Automation Platforms
- Conclusions

ALIGN: A System for Automating Analog Layout

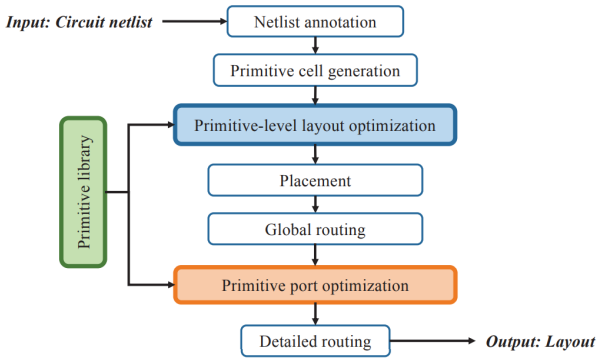
- Translation of a SPICE-level netlist into a physical layout
  - Input: sized netlist of the topology, specifications, PDK description
  - Output: GDSII of layout
  - Bottom-up approach with a mix of algorithmic techniques, template-driven design, and ML
- Highlight:
  - 24-hour turnaround with no human in the loop
  - Accounts for circuit hierarchies
  - Accounts for analog layout techniques (e.g., common centroid layout)



T. Dhar, et al., "ALIGN: A System for Automating Analog Layout," IEEE Design & Test, Vol. 38, No. 2, pp. 8-18, 2021

Synthesis Steps of ALIGN

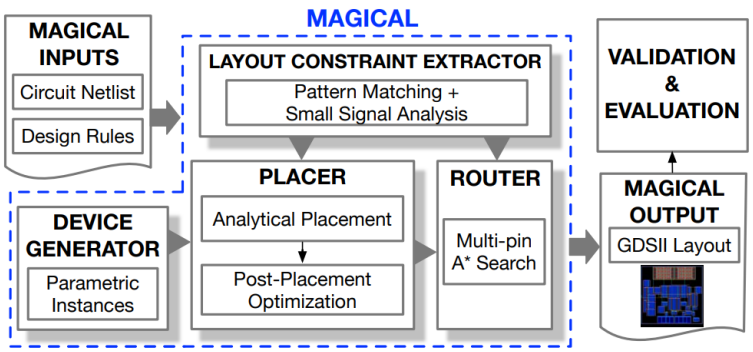
- Design Rule abstraction
  - Constraints from PDK
- Netlist Auto-annotation
  - Groups transistors and passives in the input netlist into building blocks and identifies geometric constraints on the layout of each block
- Electrical Constraint Generation
  - Performance constraints turned into layout constraints, such as the maximum allowable route length
- Parameterized layout generation of primitives
  - Parameters: transistor size, capacitor size, resistor size
- Block assembly
  - Place and route all blocks based on design hierarchy



T. Dhar, et al., "ALIGN: A System for Automating Analog Layout," IEEE Design & Test, Vol. 38, No. 2, pp. 8-18, 2021

MAGICAL: Machine Generated Analog IC Layout

- Netlist to GDSII synthesis flow for analog circuits
- Core placement algorithm:
  - Non-linear programming-based global placer
  - Linear programming-based legalizer
- Core routing algorithm:
  - Obstacle-aware path-finding algorithm searching the feasible routing paths
- Highlight: silicon-proven with TSMC 40nm 1GS/s delta-sigma ADC

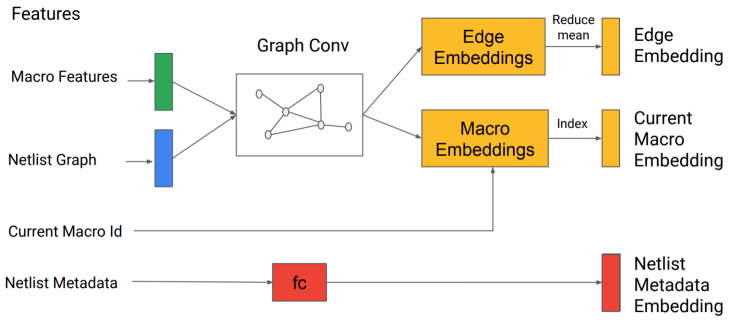


B. Xu, et al., "MAGICAL: Toward Fully Automated Analog IC Layout Leveraging Human and Machine Intelligence: Invited Paper," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp.1-8, 2019

204

Automated Cell Placement By Google

- Primarily for digital cell placement optimization with RL and GNN
  - Applications: design google accelerator chips (TPUs)
- RL for placing macros, heuristics to place standard cells
  - RL reward: expected wirelength (i.e., HPWL) and expected congestion
- Edge-based GNN operate on embeddings of placed partial graph and candidate node



A. Mirhoseini, et al., "A Graph Placement Methodology for Fast Chip Design", Nature, No. 594, pp. 207–212, 2021

205

Various Learning Scenarios Powered by Variants of ANNs

Differentiate by training scheme

- Supervised
- Semi-supervised
- Unsupervised
- Adversial (GAN)
- Reinforcement
- Encoder-decoder

X

Differentiate by data format

- Multi-layer perceptrons (feedforward neural networks)
- Convolutional neural networks
- Graph neural networks
- Recurrent neural networks

Can combine any option from the left with any option from the right

Drexel UNIVERSITY

206

Comparison of Analog Synthesis Platforms

- Even state-of-art analog synthesis platforms require pre-defined procedural rules

	Programming Language	Open-source?	Performs hierarchy recognition	Automated constraint generation	Technology dependency	Silicon-proven?
BAG	python	Yes	No	No	Independent	No
ALIGN	python, C	Yes	Yes	Yes	Requires compatible tech files	Yes
MAGICAL	Computation: C++, User interface and control: python	Yes	Yes	Yes	Requires compatible tech files	Yes

Drexel UNIVERSITY

207

Outline of Presentation

- Background Introduction
- Machine Learning Techniques for Analog EDA
- Optimization Techniques for Analog EDA
- Circuit Applications
- Analog Automation Platforms
- Conclusions

Drexel UNIVERSITY

208

Summary of AL/ML for EDA

- Benefits brought by ML for EDA:
  - Reduced simulations required, reduced turnaround time
  - Design space exploration
  - Prediction of parasitic impedances, reliability and variability
  - Guide optimization or direct generation of schematic and layout design
  - Migration and reuse of past designs
- Requirement on ML-based circuit models:
  - Sample efficiency
  - Generalization/transferability/reusability
- Current state: ML-assisted EDA flow for IC synthesis
  - ML is applied to refine/improve heuristics
    - Push optimality boundary in less time
  - Heuristics with procedural synthesis still dominate latest tools
- Future: ML-based EDA flow?
  - Direct generation of designs
  - Generative

Drexel UNIVERSITY

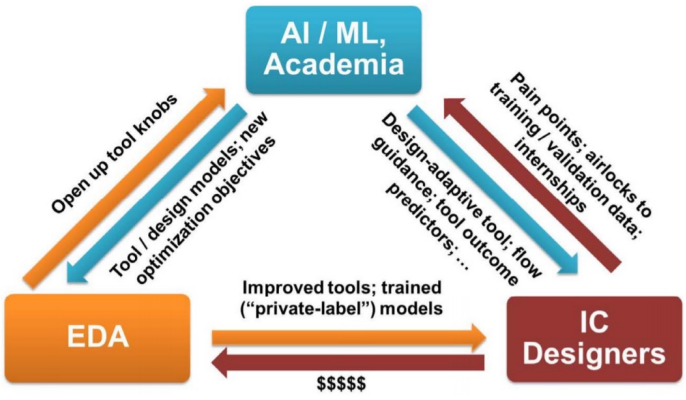
209

Future Directions

- Improve reliability, robustness, and interpretability of ML models for EDA
  - More mature and standardized flow of analog synthesis
- Strong AI for circuit synthesis
  - Current ML models identify correlation instead of causality among data
    - Learning causality requires expert knowledge
  - Ideal if AI understands human design thinking while discovering new rules for circuit synthesis
    - Need interpretability
- Meta-learning for EDA pipeline
  - Learning what to learn
    - Learn parameter values for base (pre-trained) models for circuit tasks
  - Learning which model to learn
    - Auto select the ML and optimization algorithms best suited for a given circuit task
  - Learning how to learn
    - Auto hyperparameter tuning of ML models and generation of pipeline for analog EDA
      - Parsing of standard circuit files (SPICE, DSPF, LEF, DEF…)

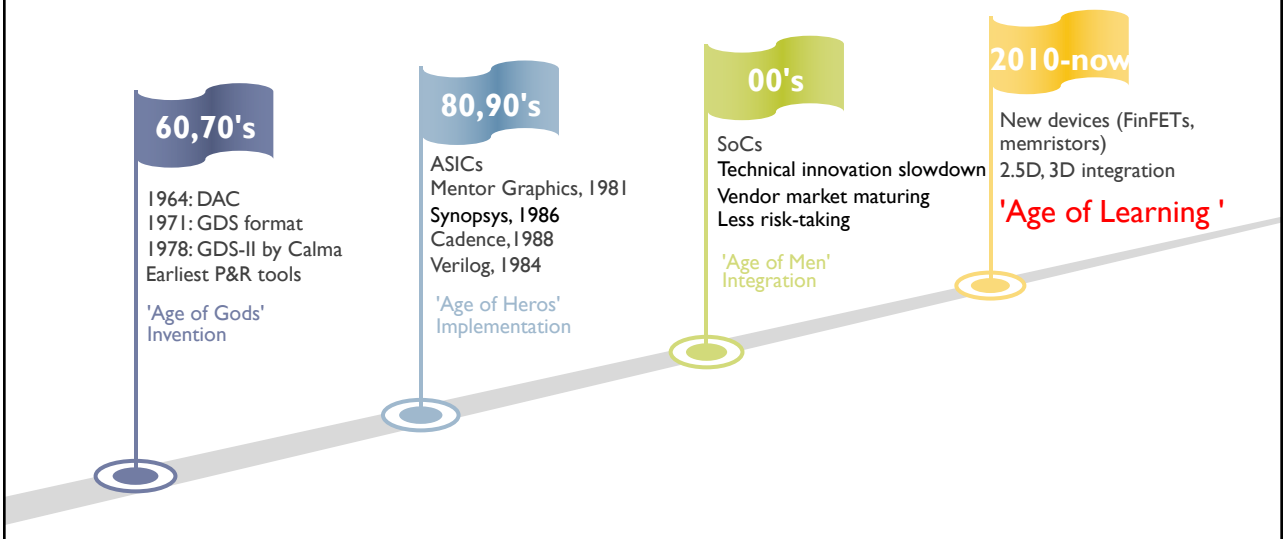
Conclusions

- EDA tools are like autonomous vehicles
  - Currently, driver control/attention is still required
- Analog EDA tools are like autonomous vehicles to be driven in more challenging environments
- Level of automation will keep rising
- More collaborations needed between circuit design, academia and industry



Reference: A. B. Kahng, "Machine Learning Applications in Physical Design: Recent Results and Directions", Proceedings of the International Symposium on Physical Design, pp. 68-73, 2018

A Remembrance of the Past: Timeline of (Digital) EDA Development



Drexel University

Reference: A. Sangiovanni-Vincentelli, "The Tides of EDA," IEEE Journals & Magazine, Vol. 20, No. 6, pp. 59-75, 2003

212

ICE Design & Analysis Lab

http://ice.ece.drexel.edu

Drexel University

213

References

- A. Sangiovanni-Vincentelli, "The Tides of EDA," IEEE Journals & Magazine, Vol. 20, No. 6, pp. 59-75, 2003
- J. Scheible, "Optimized is Not Always Optimal," Proceedings of the International Symposium on Physical Design, pp. 151–158, 2022
- B. Razavi, "Design of Analog CMOS Integrated Circuits," McGraw-Hill, 2001
- J. Crossley, et al., "BAG: A Designer-oriented Integrated Framework For The Development Of AMS Circuit Generators," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp. 74–81, Nov. 2013
- Q. Zhang et al., "An Open-Source And Autonomous Temperature Sensor Generator Verified With 64 Instances In SkyWater 130 nm For Comprehensive Design Space Exploration," IEEE Solid-State Circuits Letters, Vol. 5, No.1, pp. 174-177, 2022
- S. Su, et al., "Analog/Mixed-Signal Circuit Synthesis Enabled by the Advancements of Circuit Architectures and Machine Learning Algorithms," Proceedings of the Asia and South Pacific Design Automation Conference, pp.100–107, 2022
- A. Moubayed, M. Injadat, A. B. Nassif, H. Lutfiyya and A. Shami, "E-Learning: Challenges and Research Opportunities Using Machine Learning & Data Analytics," IEEE Access, Vol. 6, No.1, pp. 39117-39138, 2018
- X. Shangguan, H. Ma, A. C. Cangellaris and X. Chen, "Effect of Sampling Method on the Regression Accuracy for a High-Speed Link Problem," Proceedings of the IEEE Conference on Electrical Performance of Electronic Packaging and Systems, pp.1-3, 2021
- Z. Wu and I. Savidis, "CALT: Classification with Adaptive Labeling Thresholds for Analog Circuit Sizing," Proceedings of the ACM/IEEE Workshop on Machine Learning for CAD, pp. 49–54, 2020
- T. McConaghy, P. Palmers, G. Gielen and M. Steyaert, "Automated Extraction Of Expert Knowledge In Analog Topology Selection And Sizing," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp. 392-395,2008
- T. McConaghy, P. Palmers, G. Gielen and M. Steyaert, "Simultaneous Multi-Topology Multi-Objective Sizing Across Thousands of Analog Circuit Topologies," Proceedings of the ACM/IEEE Design Automation Conference, pp. 944-947, 2007
- T, Brown, et. al, "Language Models are Few-Shot Learners," Advances in Neural Information Processing Systems, Vol. 33, No. 1, pp.1877-1901, 2022
- J. K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How Powerful are Graph Neural Networks?," Proceedings of the International Conference on Learning Representations, pp. 1–17, 2019
- K. Li, J. Malik, "Learning to optimize," Proceedings of the International Conference on Learning Representations, 2017

References

- A. F. Budak, Z. Jiang, K. Zhu, A. Mirhoseini, A. Goldie and D. Z. Pan, "Reinforcement Learning for Electronic Design Automation: Case Studies and Perspectives," Proceedings of the Asia and South Pacific Design Automation Conference pp.500-505, 2022
- W. Lyu, et al., "An Efficient Bayesian Optimization Approach For Automated Optimization Of Analog Circuits," IEEE Transactions on Circuits and Systems, Vol. 65, No. 6, pp. 1954–1967, 2018
- N. Lourenço and N. Horta, "GENOM-POF: Multi-objective Evolutionary Synthesis Of Analog ICs With Corners Validation," Proceedings of the International Conference on Genetic and Evolutionary Computation, pp. 1119– 1126, 2012
- R. Lourenço, N. Lourenço, N. Horta, "AIDA-CMK: Multi-Algorithm Optimization Kernel Applied To Analog IC Sizing," Springer, 2015
- F. Silveira, D. Flandre and P. G. A. Jespers, "A gm/Id Based Methodology For The Design Of CMOS Analog Circuits And Its Application To The Synthesis Of A Silicon-on-insulator Micropower OTA," IEEE Journal of Solid-State Circuits, Vol. 31, No. 9, pp. 1314-1319, 1996
- R. Frevert, et al., "Modeling and Simulation for RF System Design," Springer, pp.291, 2005
- Z. Wu and I. Savidis, "Variation-aware Analog Circuit Sizing with Classifier Chains," Proceedings of the ACM/IEEE Workshop on Machine Learning for CAD, pp. 1–6, 2021
- T. Dhar, et al., "ALIGN: A System for Automating Analog Layout," IEEE Design & Test, Vol. 38, No. 2, pp. 8-18, 2021
- H. Graeb, S. Zizala, J. Eckmueller, and K. Antreich, "The Sizing Rules Method For Analog Integrated Circuit Design," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp. 343–349, 2001
- M. Liu, et al., "S3DET: Detecting System Symmetry Constraints for Analog Circuits with Graph Similarity," Proceedings of the Asia and South Pacific Design Automation Conference, pp. 193-198, 2020
- K. Kunal, et al., "GANA: Graph Convolutional Network Based Automated Netlist Annotation for Analog Circuits," Proceedings of the Design, Automation and Test in Europe Conference and Exhibition, pp. 55–60, 2020
- K. Zhu, et al., "GeniusRoute: A New Analog Routing Paradigm Using Generative Neural Network Guidance," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp.1-8, 2019
- B. Xu, et al., "MAGICAL: Toward Fully Automated Analog IC Layout Leveraging Human and Machine Intelligence: Invited Paper," Proceedings of the IEEE/ACM International Conference on Computer-Aided Design, pp.1-8, 2019
- A. Mirhoseini, et al., "A Graph Placement Methodology for Fast Chip Design," Nature, No. 594, pp. 207–212, 2021

References

- G. Pradipta, V. A. Chhabria, and S. S. Sapatnekar, "A Machine Learning Based Parasitic Extraction Tool," 2019
- B.W. Shook. et.al, "MLParest: Machine Learning based Parasitic Estimation for Custom Circuit Design," Proceedings of the ACM/IEEE Design Automation Conference, pp. 1-6, 2020
- H. Ren, G. F. Kokai, W. J. Turner and T. Ku. "ParaGraph: Layout Parasitics And Device Parameter Prediction Using Graph Neural Networks," Proceedings of the ACM/IEEE Design Automation Conference, pp. 1–6, 2020
- Z. Wu and I. Savidis, "Transfer Learning for Reuse of Analog Circuit Sizing Models Across Technology Nodes," Proceedings of the IEEE International Symposium on Circuits and Systems, pp. 1–5, 2022
- Z. Wu and I. Savidis, "Transfer of Performance Models Across Analog Circuit Topologies with Graph Neural Networks," Proceedings of the ACM/IEEE Workshop on Machine Learning for CAD, pp. 159-165, 2022
- H. Wang, et. al., "GCN-RL Circuit Designer: Transferable Transistor Sizing with Graph Neural Networks and Reinforcement Learning," Proceedings of the IEEE/ACM Design Automation Conference, pp. 1–6, 2020
- Y. Li et al., "A Customized Graph Neural Network Model for Guiding Analog IC Placement," Proceedings of the IEEE/ACM International Conference On Computer Aided Design, pp. 1-9, 2020
- A. B. Kahng, "Machine Learning Applications in Physical Design: Recent Results and Directions," Proceedings of the International Symposium on Physical Design, pp. 68-73, 2018
- M. Simonovsky and N. Komodakis, “Dynamic Edge-Conditioned Filters in Convolutional Neural Networks on Graphs,” Proceedings of the Conference on Computer Vision and Pattern Recognition, pp. 29–38, Jul. 2017
- D. Kingma and M. Welling, "Auto-Encoding Variational Bayes", Proceedings of the International Conference on Learning Representations, 2014
- I. Goodfellow, et al., "Generative Adversarial Nets", Proceedings of the International Conference on Neural Information Processing System, Vol. 2, No.1, pp.,2672-2680, 2014
- P. Shrestha and I. Savidis, "Graph Representation Learning for Parasitic Impedance Prediction of the Interconnect," Proceedings of the IEEE International Symposium on Circuits and Systems, pp. 1-5, May 2023.
- A. B. Kahng, J. Lienig, I. L. Markov, and J. Hu, "VLSI Physical Design: From Graph Partitioning to Timing Closure," Springer, 2011

Drexel UNIVERSITY

References

- **Z. Zhao and L. Zhang, “Graph-grammar-based analog circuit topology synthesis,” *Proceedings of the IEEE International Symposium on Circuits and Systems*, Sapporo, Japan, pp. 1–5, May. 2019**
- **Z. Zhao and L. Zhang, “Analog integrated circuit topology synthesis with deep reinforcement learning,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 41, no. 12, pp. 5138–5151, Dec. 2022**
- ***M. Campilho-Gomes, R. Tavares, and J. Goes, “Analog flat-level circuit synthesis with genetic algorithms,” IEEE Access, vol. 12, pp. 115 532–115 545, Aug. 2024***
- **Z. Yang, A. D. Gaidhane, K. Anderson, G. Workman, and Y. Cao, “Graph-based compact model (GCM) for efficient transistor parameter extraction: A machine learning approach on 12 nm finfets,” *IEEE Transactions on Electron Devices*, vol. 71, no. 1, pp. 254–262, Nov. 2024.**
- **S. Lee, S. Eom, J. Jeong, J. Lee, S. Lee, H. Yun, Y. Ahn, and R.-H. Baek, “Multi-task learning for real-time bsim-cmg parameter extraction of nsfets with multiple structural variations,” *IEEE Access*, vol. 12, pp. 184 619–184 630, Dec. 2024.**
- **Y. Li, W. Dai, K. Geng, L. Zhang, R. Wang and R. Huang, "An Automatic Parameter Extraction Method Based on Autoencoder for PIN Diode Model," *2022 IEEE 16th International Conference on Solid-State & Integrated Circuit Technology (ICSICT)*, Nangjing, China, pp. 1-3, Oct. 2022**
- **A. Hammoud, C. Goyal, S. Pathen, A. Dai, A. Li, G. Kielian, and M. Saligane, "Human Language to Analog Layout Using GLayout Layout Automation Framework," *2024 ACM/IEEE 6th Symposium on Machine Learning for CAD (MLCAD)*, Salt Lake City (Snowbird), UT, USA, pp. 1-7 Sept 2024.**
- **B. Liu, H. Zhang, X. Gao, Z. Kong, X. Tang, Y. Lin, R. Wang, and R. Huang, "LayoutCopilot: An LLM-Powered Multi-Agent Collaborative Framework for Interactive Analog Layout Design," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. early access, pp. 1–1, 2025**
- **C.-C. Chang, Y. Shen, S. Fan, J. Li, S. Zhang, N. Cao, Y. Chen, and X. Zhang. 2024. LaMAGIC: language-model-based topology generation for analog integrated circuits, *Proceedings of the International Conference on Machine Learning (ICML'24)*, Vol. 235. JMLR.org, Article 241, 6253–6262, Jul. 2024**

Drexel UNIVERSITY