Friedrich-Alexander-Universität
Lehrstuhl für Intelligente
Technische Elektronik und Systeme

Gefördert durch:
Bundesministerium
für Forschung, Technologie
und Raumfahrt

COBRA: An AI-Assisted Circuit-Level Optimizer for Open Source Based RFIC Design

Gianluca Simone
Institute for Smart Electronics and Systems, Friedrich-Alexander Universität Erlangen-Nürnberg, Germany

RFIC Design Challenges

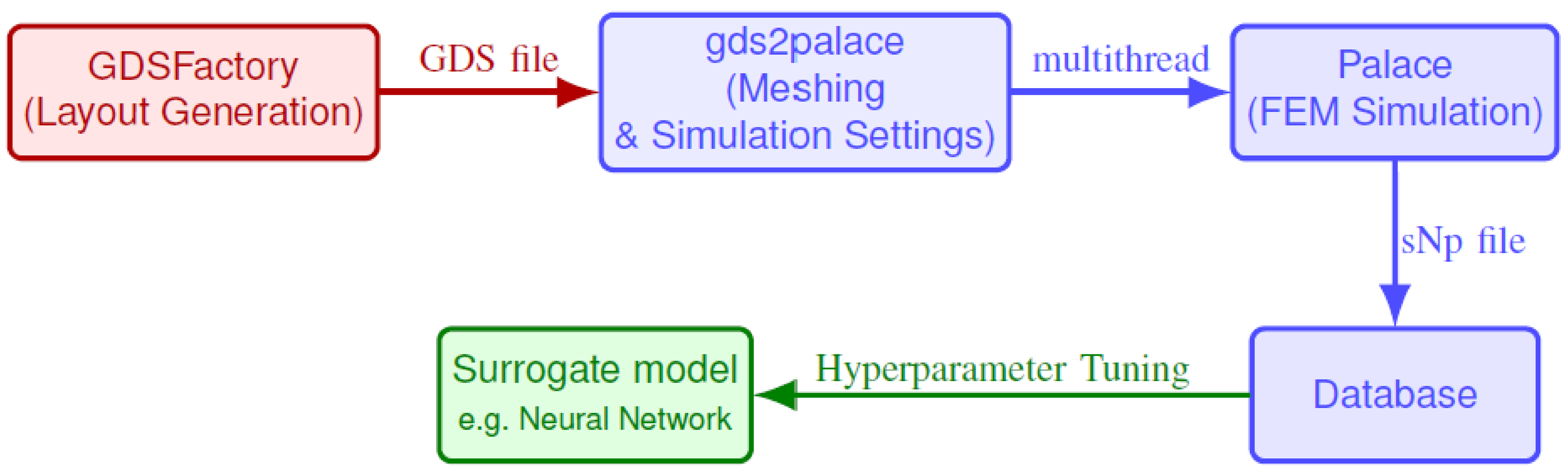
- EM simulations are slow
- Long design cycles
- Expensive, closed-source tools

Idea: AI + Open Source →

AI surrogates: Instant S-Parameter predictions for faster optimization cycles
Open-source: No license constraints, reproducible, reusable models

ORCA

GUI-driven setup: User defines parametric geometry, sweep ranges, and sampling strategy
Automated data generation: EM database is generated via Palace simulations
ML surrogate model: AI trains a user-selected model (e.g. neural network) to replace EM simulations



Idea: Joint EM-circuit co-optimization with EM surrogates integrated into circuit-level loop → COBRA

COBRA

1. Optimization Setup

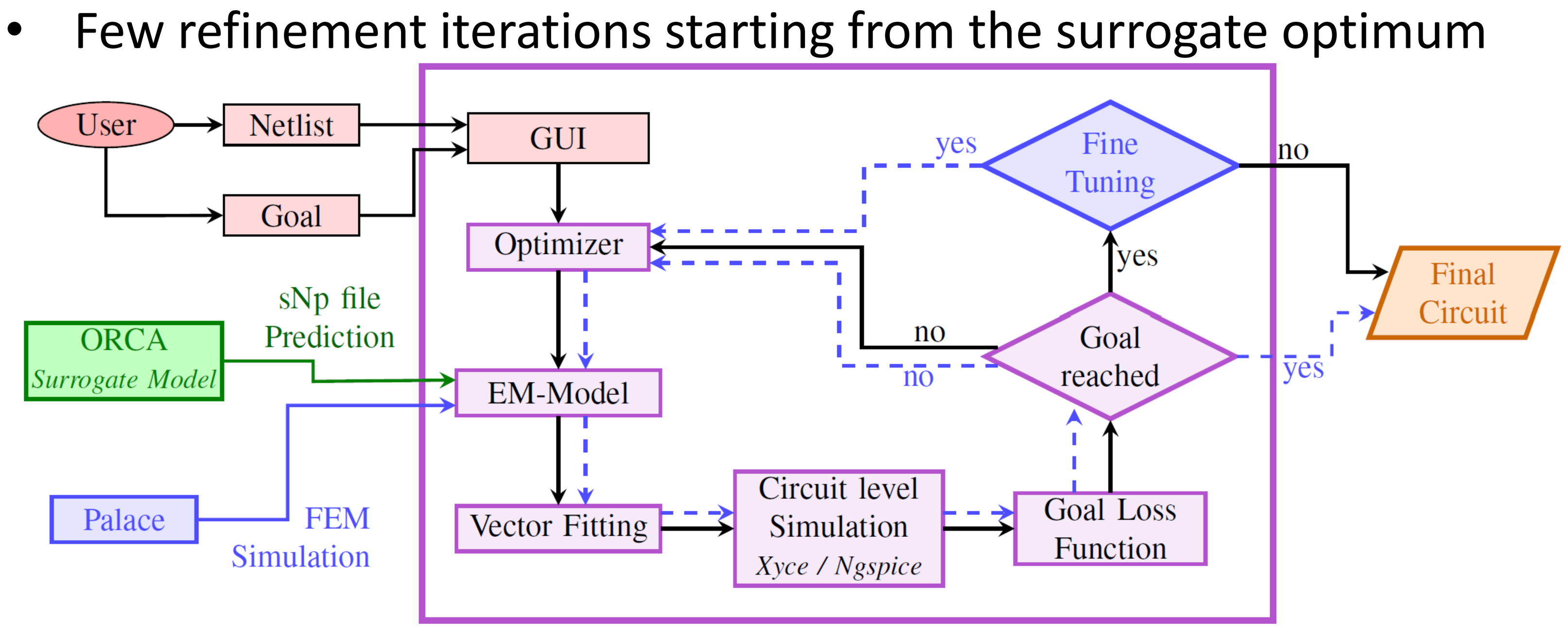
- Qucs-S schematic with EM models as S-Parameter blocks
- Export netlist to COBRA GUI & Define goals

2. Optimization Loop

Optimizer proposes parameters → EM surrogates predict S-parameters → vector fitting generates SPICE model → Xyce simulates circuit → loss vs. targets → parameters updated until goals are met

3. Fine-Tuning

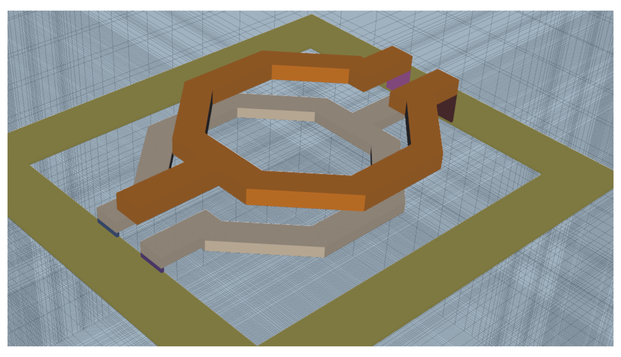
- Swap surrogate for Palace full-wave EM
- Few refinement iterations starting from the surrogate optimum



Example: Matching Network Design

1. EM Dataset Generation

- Generate EM dataset for a vertically coupled octagonal transformer
- Design space > 10^{11} configurations → exhaustive simulation infeasible
- Randomly sample 3000 geometries and simulate with FEM Solver Palace



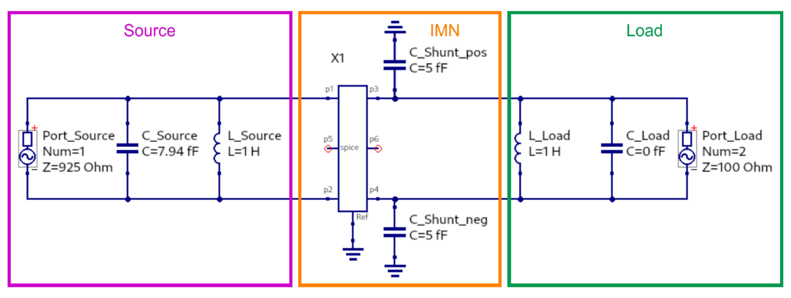
2. Surrogate Model Training and Validation

- MLP (Neural Network)
- Input: 5 geometry parameters + frequency
- Output: Full 6-port S-Parameters (Re & Im)
- Training: MSE loss on S-Parameters with 70/15/15 train/val/test
- Model optimized via automated hyperparameter tuning

Samples	RMSE	ε_L	ε_{SRF}	ε_Q	δ_{max}
1000	0.39	1.4 %	2.6 %	8.4 %	7e-3
2000	0.18	0.8 %	0.3 %	4.2 %	6e-3
3000	0.17	0.7 %	0.3 %	4.9 %	5e-3

3. Circuit-Level Optimization

- Match $Z_{Source} = (25 - j150)\ \Omega$ to $Z_{Load} = 100\ \Omega$
- S21 > −2 dB & S22 < −10 dB



4. Optimization Results

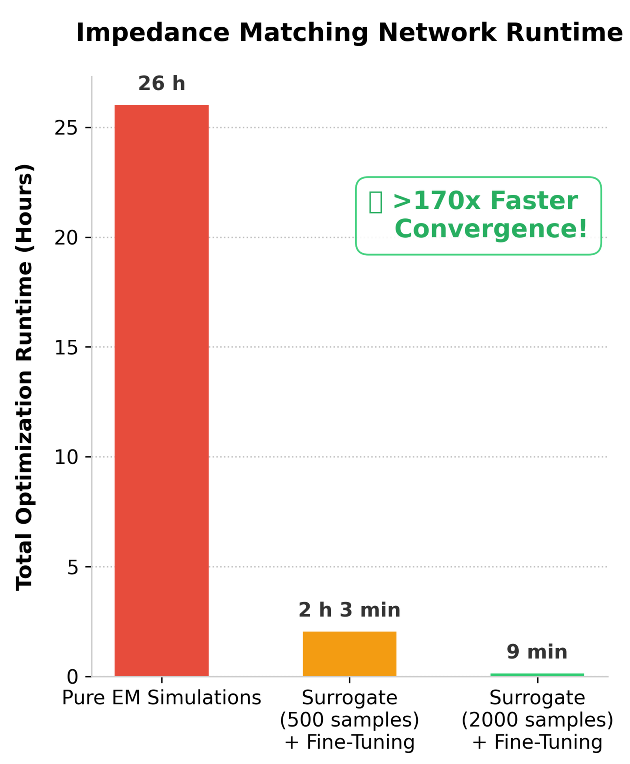
- Achieved Goals!
- Higher training size → more accurate predictions
- Few fine-tuning EM simulations to correct surrogate errors near optimum

Break-even:

- 2000 samples → 2000/200 ≈ 10 designs
- 500 samples → 500/200 ≈ 3 designs

Scaling Insight:

- + 1500 samples → 150 h extra simulation time
- Pays off after 75 designs



State-of-the-Art & Conclusion

- Prior work uses surrogate models to optimize mainly EM-level objectives for specific components
- Commercial tools support optimization & co-simulation, but are license-bound and less flexible

This Work →

- Fully open-source AI-assisted optimization framework
- Reduces hours of simulation time to minutes
- Ensures correctness via fine-tuning

Future Work

- Optimize complete open-source RF systems
- Hierarchical system-level optimization
- Additional geometries and circuit classes (mixers, PAs, oscillators)
- New simulation modes (HB, transient, noise) and simulators (Ngspice, VACASK)

gianluca.simone@fau.de
https://github.com/DI-PASSIONATE/COBRA
gianluca-simone2000
Giannix00