Arvin Islam-Gomes

Paper: "iTPlace: Machine Learning-Based Delay-Aware Transistor Placement for Standard Cell Synthesis" by Lee, Yang, and Li (ICCAD 2020)

What the Paper Studies

General Topic: Cell layout synthesis in modern digital IC design, specifically automating standard cell library synthesis.

Specific Activity: Delay-aware transistor placement at the sign-off level using machine learning. Previous tools were only optimized for area and routability, so the authors add delay into the equation.

Research Question: "This is the first work to propose a method of delay-aware transistor placement for cell library synthesis at the sign-off level." Essentially, can ML bridge the gap between automated layout and high-accuracy cell characterization by optimizing delay during placement?

Problem, Claims, Evidence, and Analysis

Problem: Previous automatic synthesis tools only consider cell area and routability, not delay. Manual design iteration between layout and sign-off simulation is slow and exhausting, and traditional automation does not match industry sign-off quality.

Claims: (1) Their CNN routability classifier achieves up to 98% accuracy. (2) Their CNN delay ranker achieves up to 94.6% accuracy. (3) 1.77% average delay improvement over the NCTUcell automated library. (4) 0.97% better delay than the handcrafted ASAP library (human-level design).

Evidence: Experiments on a 7nm FinFET node using the ASAP7 PDK. Designs were routed with NCTUcell, extracted with Mentor Calibre, and characterized with Synopsys SiliconSmart. All cells passed LVS and DRC. Tables 3 through 8 compare delay values in picoseconds across sequential and combinational cells. An exhaustive search of all DHL1x placements showed their solution ranked 2nd overall, only 0.56% off from the best possible result.

Statistical Analysis: Accuracy, miss rate, specificity, and precision are used for the routability classifier (98.7% ACC, 97.6% SPC, 100% precision on average). The delay ranker compares three architectures, with Model 2 hitting 94.6% test accuracy. Cell delay is averaged across timing arcs per Equation 1. No traditional hypothesis testing is used, which is typical for EDA/ML benchmark papers.