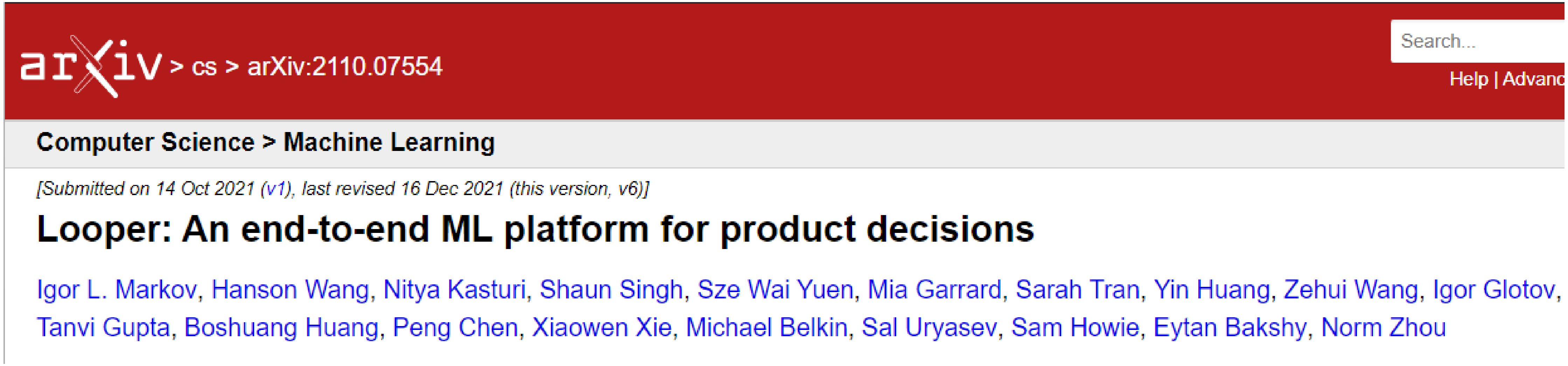
How Vertical ML Platforms Echo EDA Design Flows

Igor Markov

University of Michigan and Meta

Opinions stated here are mine and do not reflect those of my former, current of future employers



Computer Science > Machine Learning

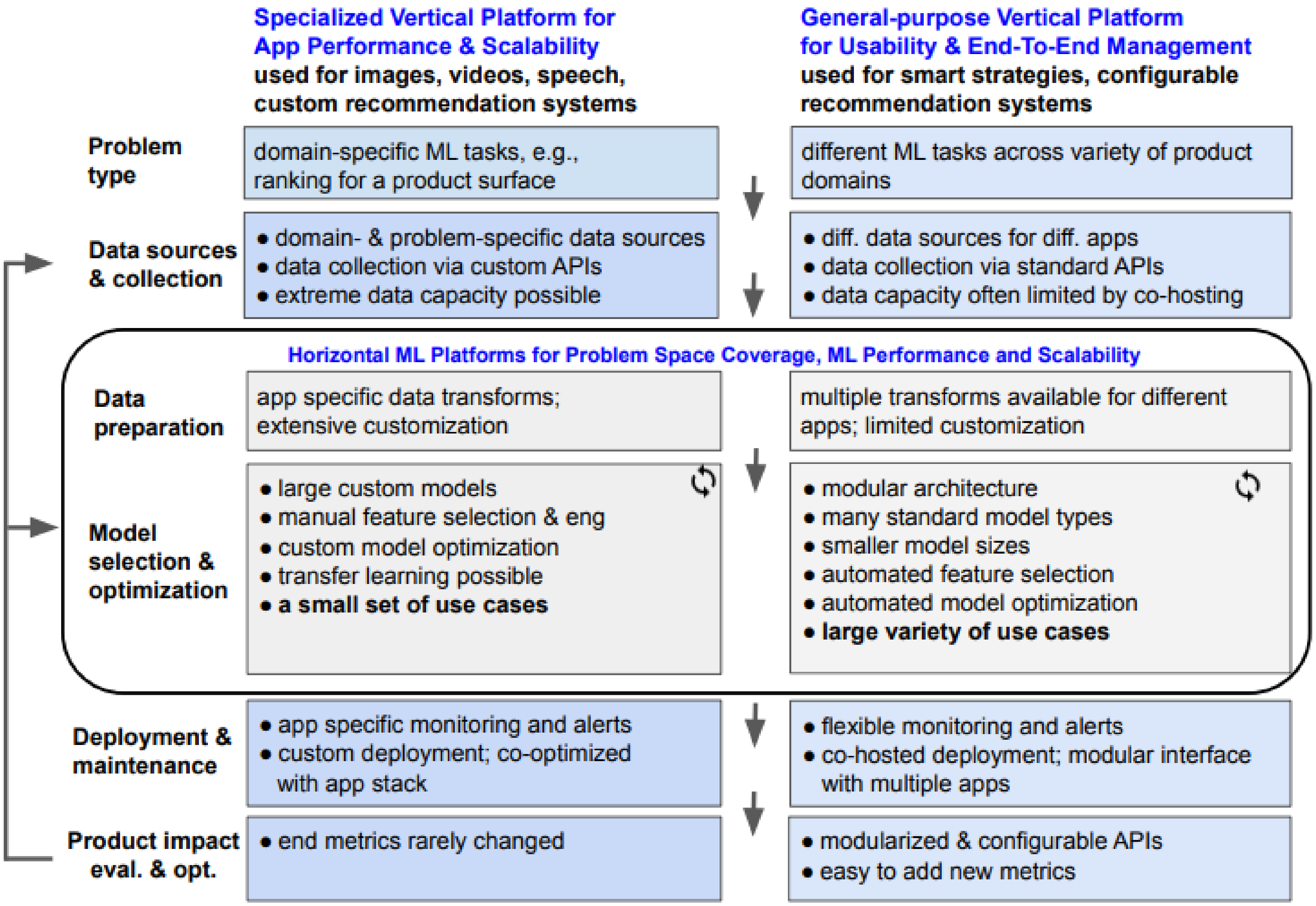
[Submitted on 14 Oct 2021 (v1), last revised 16 Dec 2021 (this version, v6)]

Looper: An end-to-end ML platform for product decisions

Igor L. Markov, Hanson Wang, Nitya Kasturi, Shaun Singh, Sze Wai Yuen, Mia Garrard, Sarah Tran, Yin Huang, Zehui Wang, Igor Glotov, Tanvi Gupta, Boshuang Huang, Peng Chen, Xiaowen Xie, Michael Belkin, Sal Uryasev, Sam Howie, Eytan Bakshy, Norm Zhou

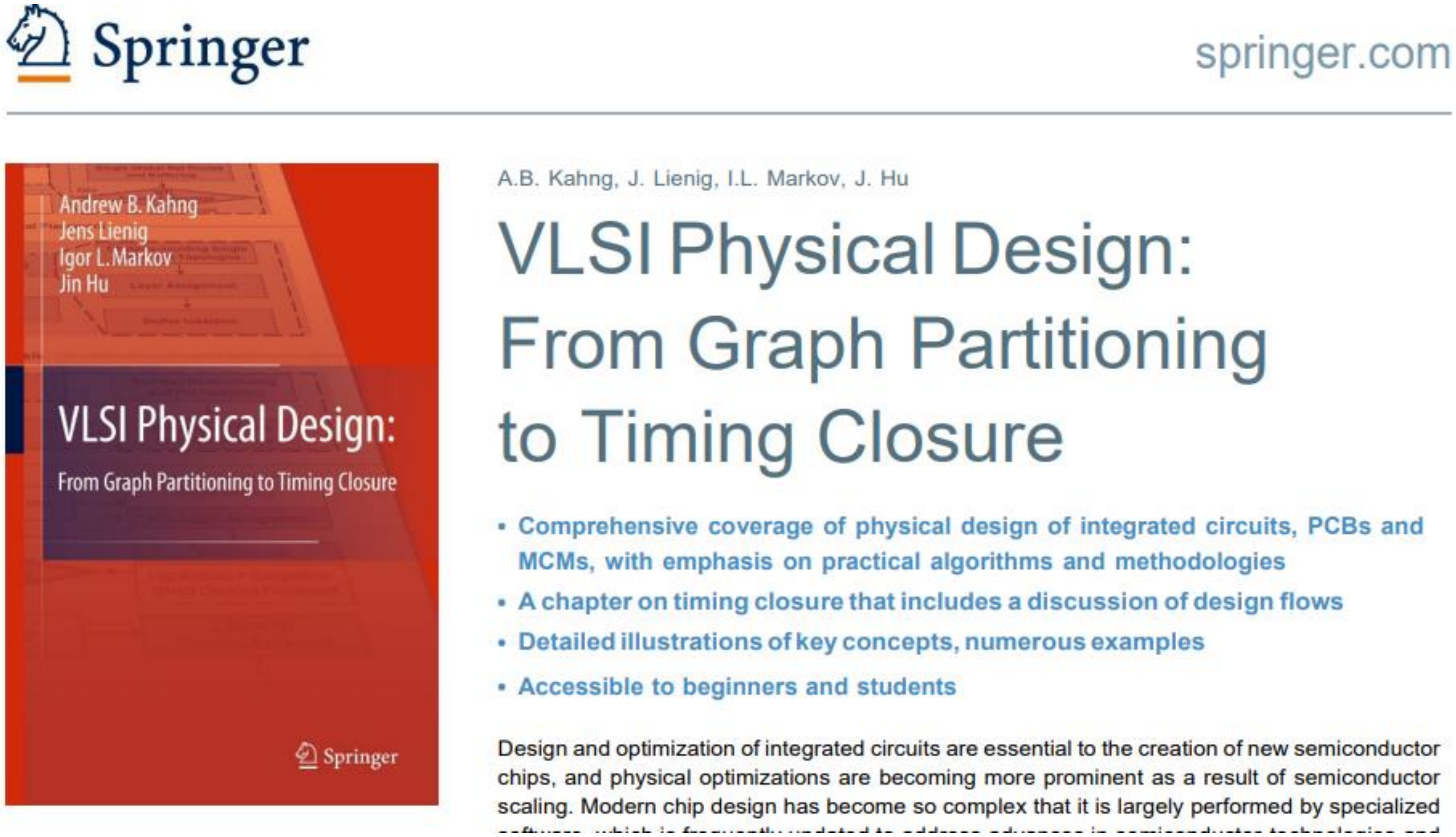
- API for decision/prediction points in SW systems
    - Online products interacting with many users
    - Must learn from data and personalize policies
- Examples
    - data-driven personalization and adaptive interfaces
    - controlling the frequency of user notifications
    - prefetching to reduce latency
    - content ranking and prioritizing available actions

- Supervised ML
    - Classification
    - Regression
- Reinforcement Learning
    - Contextual Bandit
    - MDP-style RL
- Deep Learning vs. XGBoost
- **90+ product teams**
- **400-1000 ML models running**
- **4-6M AI outputs/sec**



	Specialized Vertical Platform for App Performance & Scalability used for images, videos, speech, custom recommendation systems	General-purpose Vertical Platform for Usability & End-To-End Management used for smart strategies, configurable recommendation systems
Problem type	domain-specific ML tasks, e.g., ranking for a product surface	different ML tasks across variety of product domains
Data sources & collection	• domain- & problem-specific data sources • data collection via custom APIs • extreme data capacity possible	• diff. data sources for diff. apps • data collection via standard APIs • data capacity often limited by co-hosting
Horizontal ML Platforms for Problem Space Coverage, ML Performance and Scalability		
Data preparation	app specific data transforms; extensive customization	multiple transforms available for different apps; limited customization
Model selection & optimization	• large custom models • manual feature selection & eng • custom model optimization • transfer learning possible • **a small set of use cases**	• modular architecture • many standard model types • smaller model sizes • automated feature selection • automated model optimization • **large variety of use cases**
Deployment & maintenance	• app specific monitoring and alerts • custom deployment; co-optimized with app stack	• flexible monitoring and alerts • co-hosted deployment; modular interface with multiple apps
Product impact eval. & opt.	• end metrics rarely changed	• modularized & configurable APIs • easy to add new metrics

Figure 1. Categories of applied ML platforms: horizontal vs. vertical, specialized vs. general-purpose (back arrows show vertical optimizations based on product metrics). Specialized platforms are limited in their support for diverse applications.



Springer

springer.com

Andrew B. Kahng
Jens Lienig
Igor L. Markov
Jin Hu

VLSI Physical Design:
From Graph Partitioning to Timing Closure

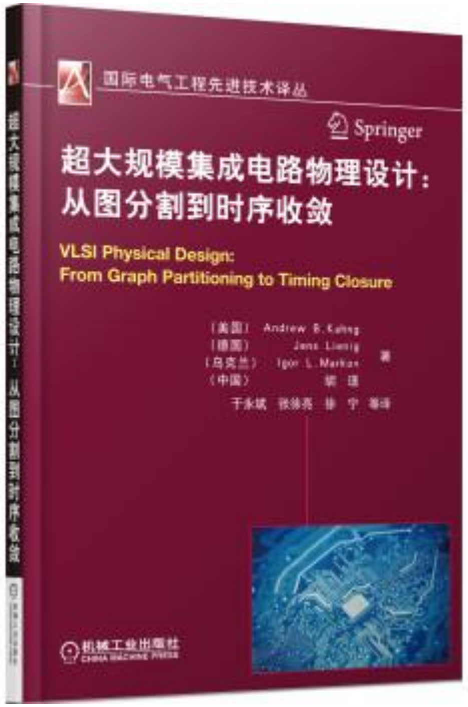
Springer

A.B. Kahng, J. Lienig, I.L. Markov, J. Hu

VLSI Physical Design: From Graph Partitioning to Timing Closure

- Comprehensive coverage of physical design of integrated circuits, PCBs and MCMs, with emphasis on practical algorithms and methodologies
- A chapter on timing closure that includes a discussion of design flows
- Detailed illustrations of key concepts, numerous examples
- Accessible to beginners and students

Design and optimization of integrated circuits are essential to the creation of new semiconductor chips, and physical optimizations are becoming more prominent as a result of semiconductor scaling. Modern chip design has become so complex that it is largely performed by specialized



国际电气工程先进技术译丛

Springer

超大规模集成电路物理设计：
从图分割到时序收敛

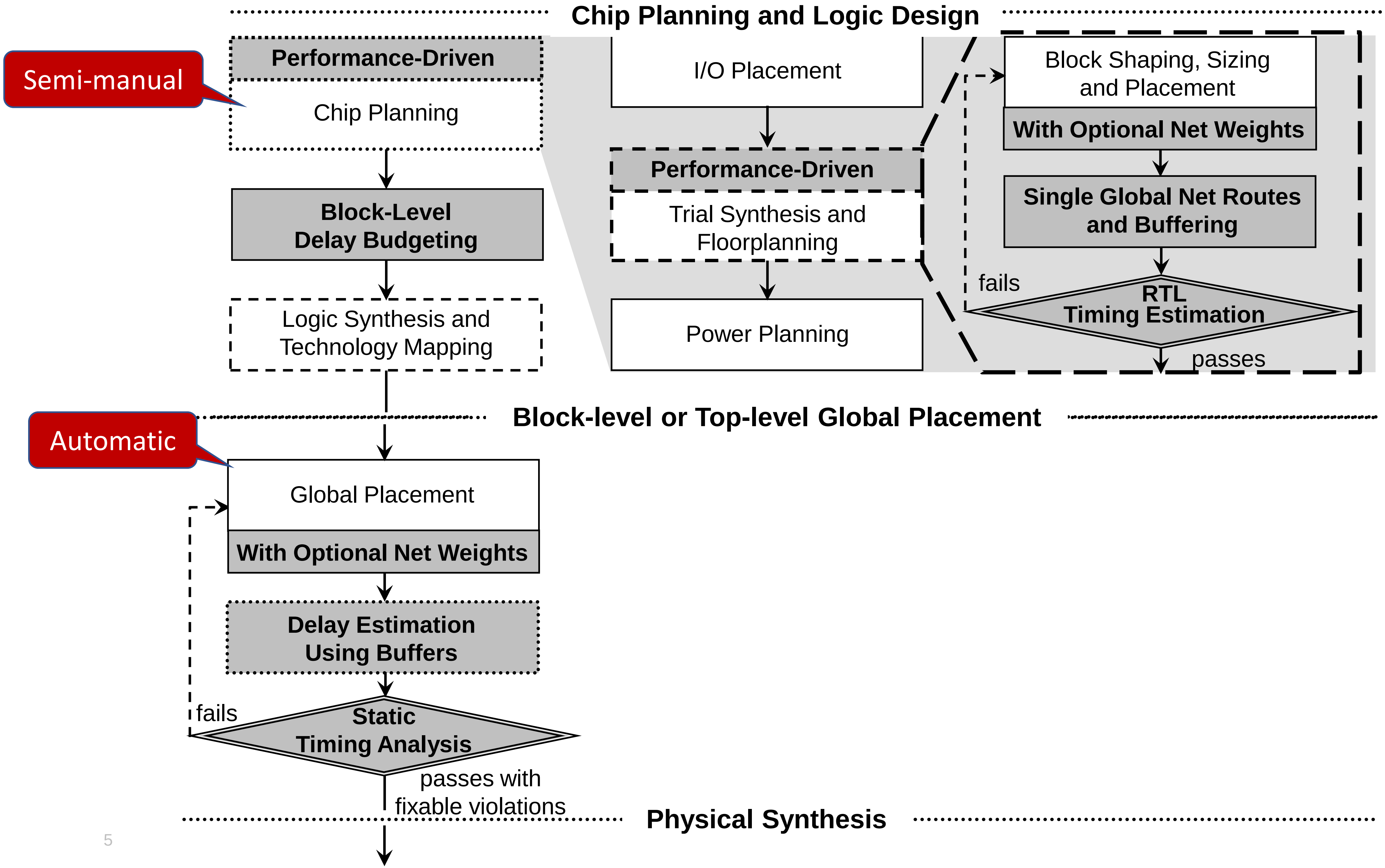
VLSI Physical Design:
From Graph Partitioning to Timing Closure

（美国） Andrew B.Kahng
（德国） Jens Lienig
（乌克兰） Igor L.Markov 著
（中国） 胡 瑾
于永斌 张徐亮 徐 宁 等译

机械工业出版社
CHINA MACHINE PRESS

超大规模集成电路物理设计：从图分割到时序收敛

2nd edition on the way



Chip Planning and Logic Design

Semi-manual

Performance-Driven

Chip Planning

I/O Placement

Performance-Driven

Trial Synthesis and Floorplanning

Power Planning

Block Shaping, Sizing and Placement

With Optional Net Weights

Single Global Net Routes and Buffering

fails

RTL Timing Estimation

passes

Block-Level Delay Budgeting

Logic Synthesis and Technology Mapping

Block-level or Top-level Global Placement

Automatic

Global Placement

With Optional Net Weights

Delay Estimation Using Buffers

fails

Static Timing Analysis

passes with fixable violations

Physical Synthesis

Two different domains?

The Looper vertical ML platform

- AI outputs: predictions/decisions
- A lot of data available on the same platform
- **A lot of data for training ML models that optimize data fit**

EDA tools and design flows

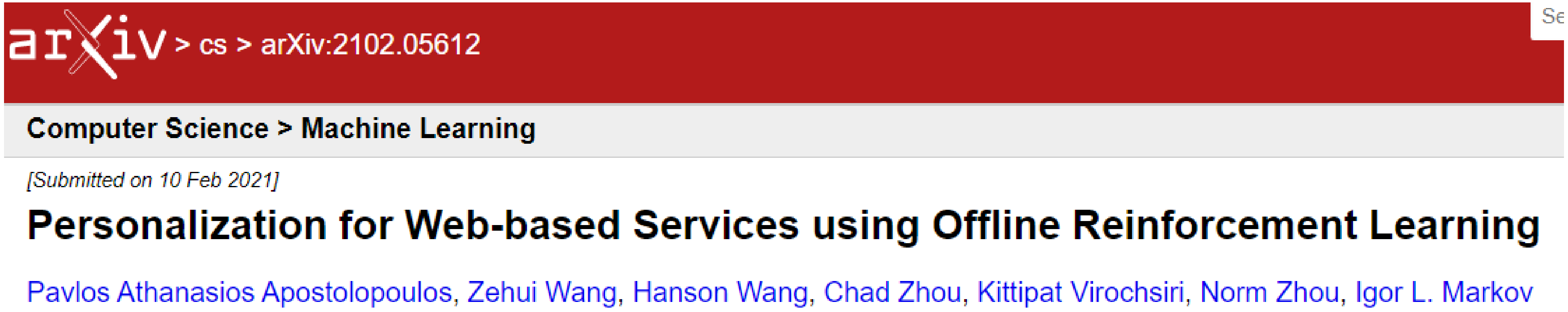
- Synthesis and optimization tasks
- No one sees a significant fraction of all chips designed + not many chips
- **Limited training data, easy to overfit**
- **Established “closed-form” optimizers optimize mathematical obj function**

ML platforms and EDA: anything in common?

- Point optimization vs. end-to-end optimization
- **Bad news**: “product impact” cannot be captured in closed form
  - User engagement metrics are measured on live data
  - Full-chip performance is measured algorithmically
- **Good news**: classic simple objectives get you to the ballpark
  - If some ML model loses by >10% in ROC AUC, NE, NDCG, it is likely useless
  - If some global placer loses by >10% HPWL, it likely loses by other metrics too
- Researchers are tempted to claim otherwise (w/o evidence) - a trap!
- In practice, optimize simple objectives first,
  then tune for product impact

ML platforms and EDA: anything in common?

- **A natural idea:** optimize *product objectives* using Reinforcement Learning



arXiv > cs > arXiv:2102.05612

Se

Computer Science > Machine Learning

[Submitted on 10 Feb 2021]

Personalization for Web-based Services using Offline Reinforcement Learning

Pavlos Athanasios Apostolopoulos, Zehui Wang, Hanson Wang, Chad Zhou, Kittipat Virochsiri, Norm Zhou, Igor L. Markov

- Hard to pull off, planets must align
  - Need a lot of training data, RL is slow
  - High-variance training, often unstable
  - Poor out-of-distribution performance?

Robust baselines are hard to beat:

- Supervised ML w XGBoost
- Heuristics (not ML)

ML platforms and EDA: anything in common?

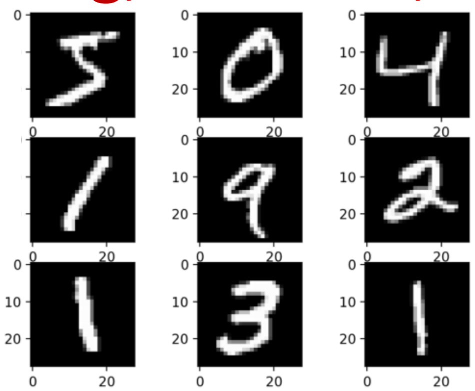
- Belief in *magic*
  - **“Any sufficiently advanced technology is indistinguishable from magic” – Arthur C. Clarke**
  - Striking breakthroughs – **simulated annealing**, **deep learning, RL for games**
  - **Many false starts, benchmarking goofs** (discussed for placement in P. Madden, 2001)
- To ensure progress, very important are: the culture of public benchmarking, contests, and reproducibilit

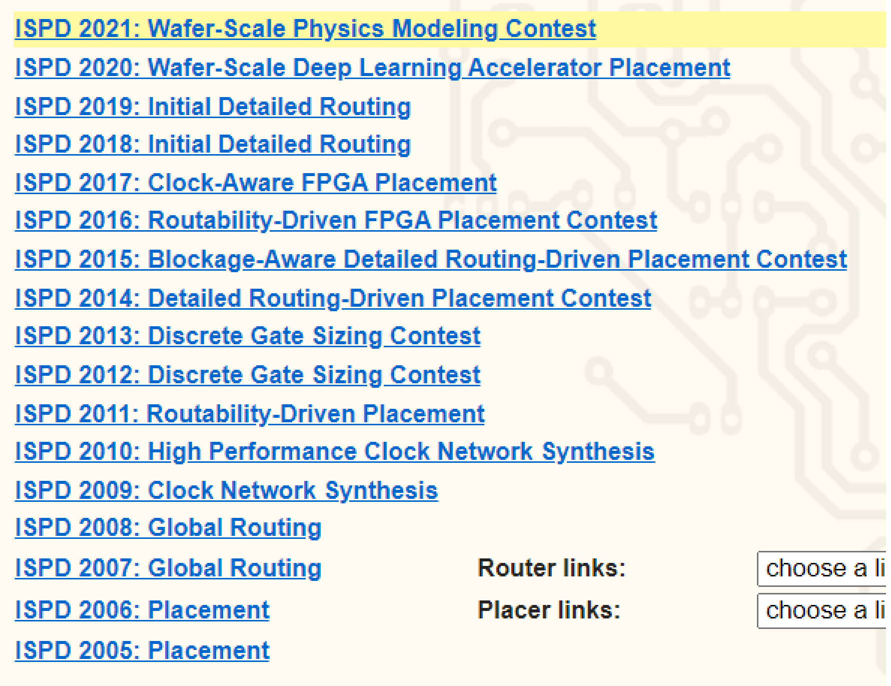
VLSI CAD Bookshelf 2

kaggle

IMAGENET

- https://paperswithcode.com/



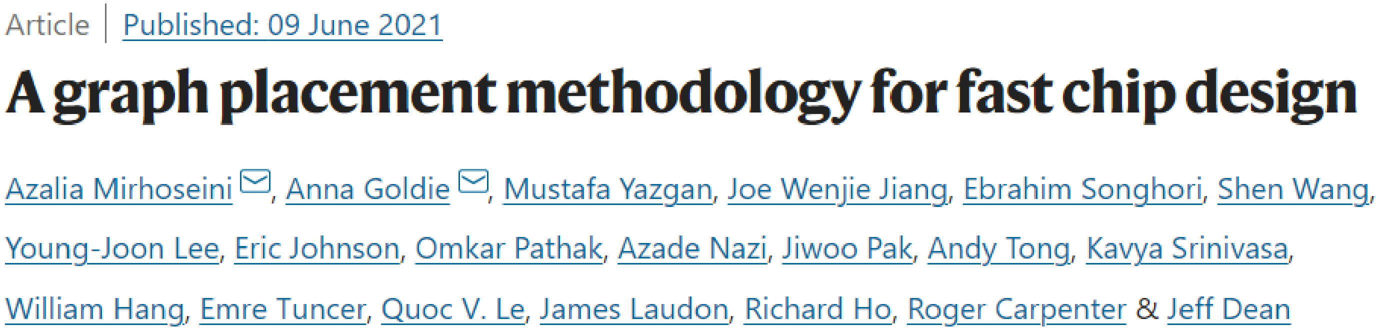


ISPD 2021: Wafer-Scale Physics Modeling Contest
ISPD 2020: Wafer-Scale Deep Learning Accelerator Placement
ISPD 2019: Initial Detailed Routing
ISPD 2018: Initial Detailed Routing
ISPD 2017: Clock-Aware FPGA Placement
ISPD 2016: Routability-Driven FPGA Placement Contest
ISPD 2015: Blockage-Aware Detailed Routing-Driven Placement Contest
ISPD 2014: Detailed Routing-Driven Placement Contest
ISPD 2013: Discrete Gate Sizing Contest
ISPD 2012: Discrete Gate Sizing Contest
ISPD 2011: Routability-Driven Placement
ISPD 2010: High Performance Clock Network Synthesis
ISPD 2009: Clock Network Synthesis
ISPD 2008: Global Routing
ISPD 2007: Global Routing
ISPD 2006: Placement
ISPD 2005: Placement

Router links: choose a li

Placer links: choose a li

A case study

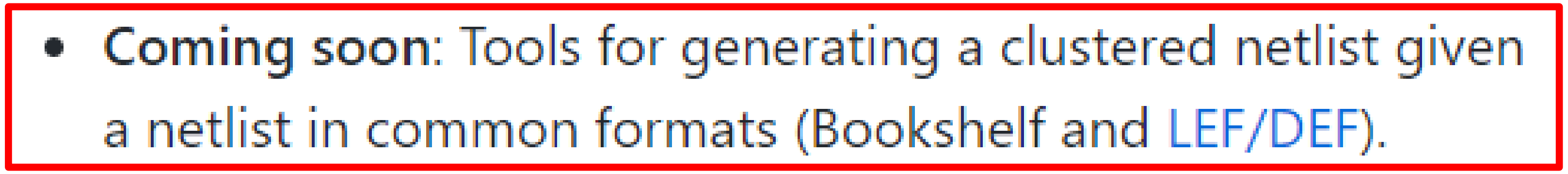


Article | Published: 09 June 2021

A graph placement methodology for fast chip design

Azalia Mirhoseini, Anna Goldie, Mustafa Yazgan, Joe Wenjie Jiang, Ebrahim Songhori, Shen Wang, Young-Joon Lee, Eric Johnson, Omkar Pathak, Azade Nazi, Jiwoo Pak, Andy Tong, Kavya Srinivasa, William Hang, Emre Tuncer, Quoc V. Le, James Laudon, Richard Ho, Roger Carpenter & Jeff Dean

- Idea: adapt RL breakthroughs to outdo mathematical optimization in mixed-size placement
  - Main results on selected proprietary circuits, with a full commercial design flow
- **Serious doubts about empirical results and top-line claims**
  - RL appears slow – competing analytical placers are way faster
  - Claimed improvements are small, possibly within the noise margin
  - RL is post-processed by Sim Annealing and Coordinate Descent, but no ablation studies
  - Impaired open-sourcing https://github.com/google-research/circuit_training



- **Coming soon**: Tools for generating a clustered netlist given a netlist in common formats (Bookshelf and LEF/DEF).

  - **RL is only compared to other methods after pre-processing** (a 20 year-old design flow) --- **modern mixed-size placers need no clustering**
  - **No compelling results for common mixed-size benchmarks**