Circuit Training: An open-source framework for generating chip floor plans with distributed deep reinforcement learning.

Ph.D. Sergio Guadarrama, Google Research, Brain Team
Collaboration with TPU design team

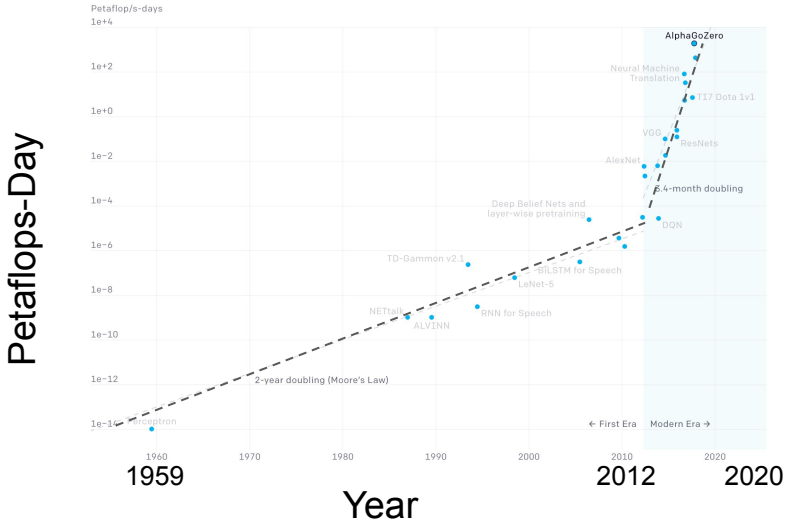
Presenting the work of many people at Google (*Leads):
Sergio Guadarrama*, Summer Yue, Toby Boyd, Joe Wenjie Jiang*,
Ebrahim Songhori, Terence Tam, Azalia Mirhoseini*

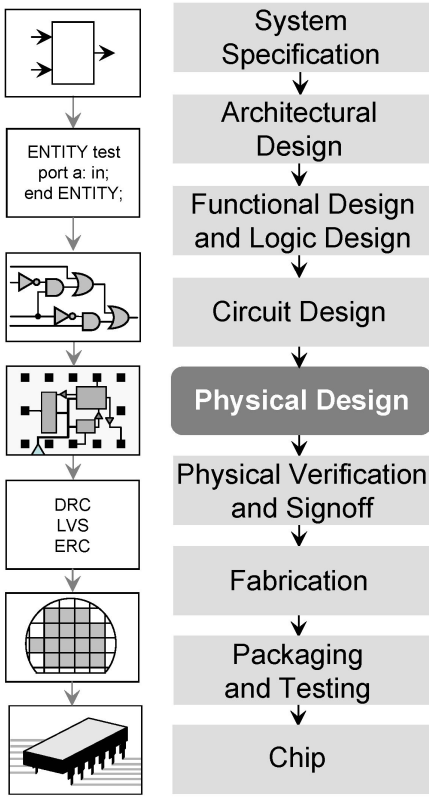
https://github.com/google-research/circuit_training

Google Research

Fast Chip Design to Keep Up with the Computational Demands of AI

Between 1959 to 2012, compute usage roughly doubled every two years
Since 2012, the amount of compute used in the largest AI training runs doubled every 3.4 months[1]





Circuit-Training: RL for Chip Placement

1.OpenAI'19

Approaches to Chip Placement

Partitioning-Based Methods (e.g. MinCut)	Stochastic/Hill-Climbing Methods (e.g. Simulated Annealing)
Analytic Solvers (e.g. RePlAce, DREAMPlace)	Learning-Based Methods (e.g. Reinforcement Learning)

nature

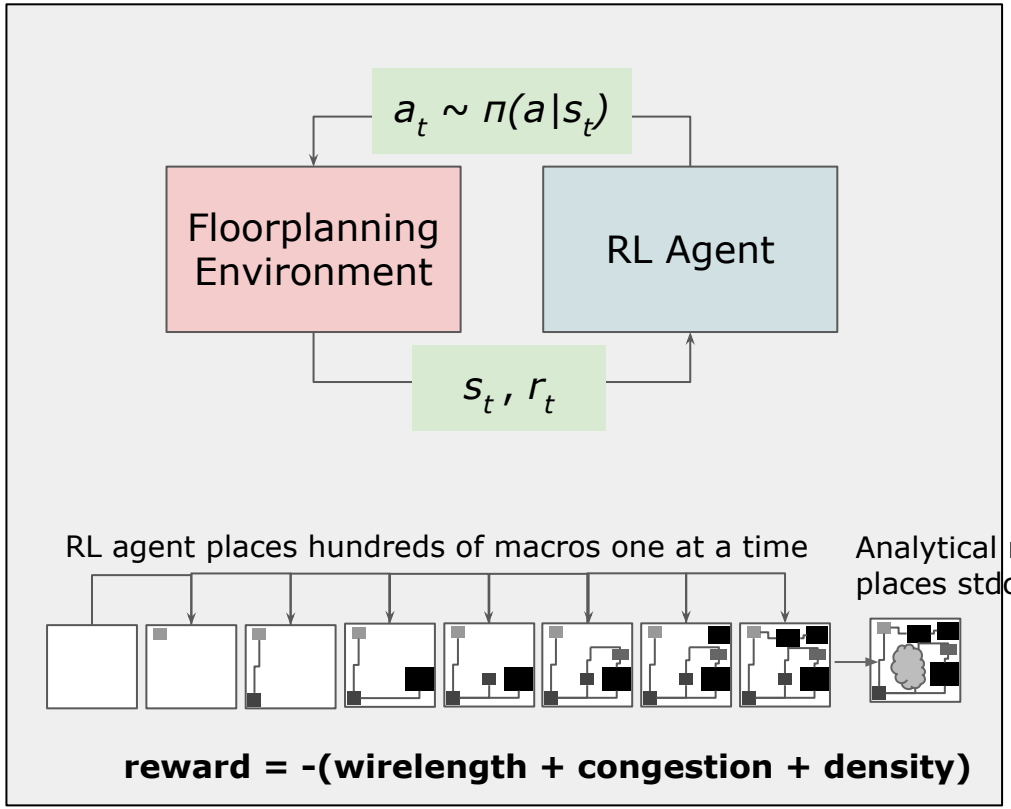
Article | Published: 09 June 2021

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

Chip Placement with Reinforcement Learning

- **Action (a):** Placing the current node on a grid cell.
- **Reward (r):** A weighted average of total wirelength, density, and congestion, after all nodes are placed.
- **State (s):** Embeddings of chip netlist and canvas.



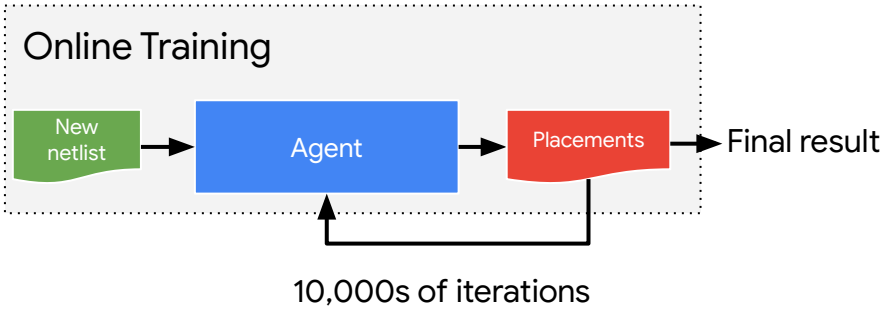
RL tool tried for placement and routing of 37 blocks of TPU-v5 chip design

Compared to placement by human experts:

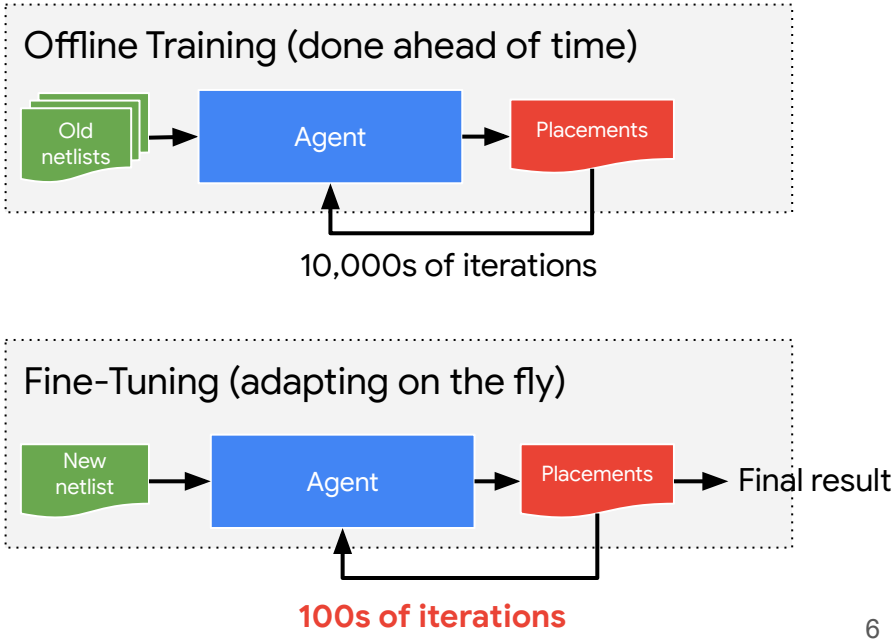
- **26 of 37** blocks better QoR than human experts
- **7 of 37** blocks equal QoR
- **4 of 37** blocks worse QoR

Moving Towards Generalized Placements

Before: Training from scratch for each chip netlist

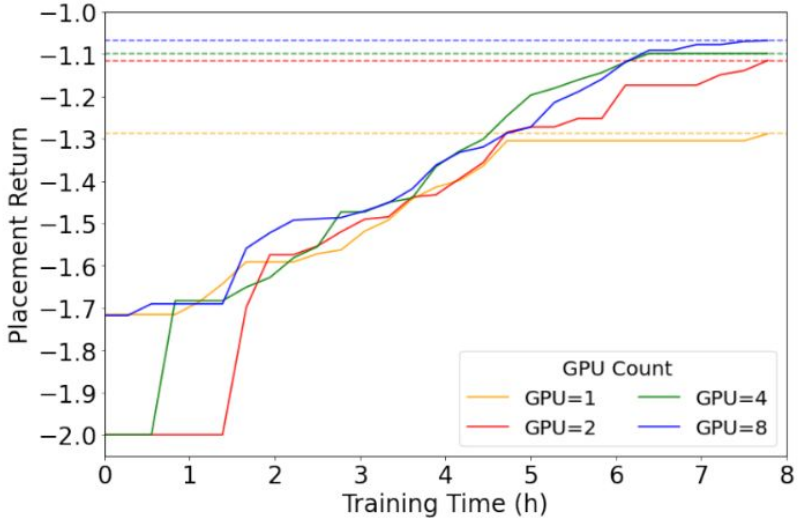


Now: Pre-training the agent on multiple netlists and finetuning on new netlists

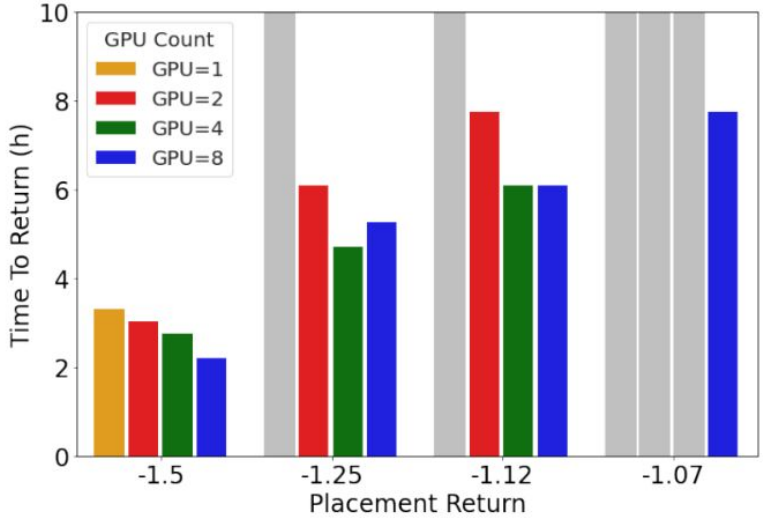


Open Sourced Circuit-Training: Scalable RL Floor-Planning

- https://github.com/google-research/circuit_training
- Example: RISC-V Ariane block.
- Increasing the number of GPUs results in better final placements.
- Increasing GPUs 1 -> 8 reduces 30% training time to reach a –1.5 return



Grey means unattainable.



Benefits of ML-Based Chip Floorplanning

Allow ASIC designers to **quickly generate many alternatives** with different tradeoffs of area, timing, etc.

Quickly re-do layout even if major changes in design upstream (can get pretty good estimates of QoR in seconds with zero-shot placement or within 1-2 hours with fine-tuning)

Can significantly reduce time/effort needed to develop new ASICs and **improve quality of the results**

Thank You

https://github.com/google-research/circuit_training

Google Research