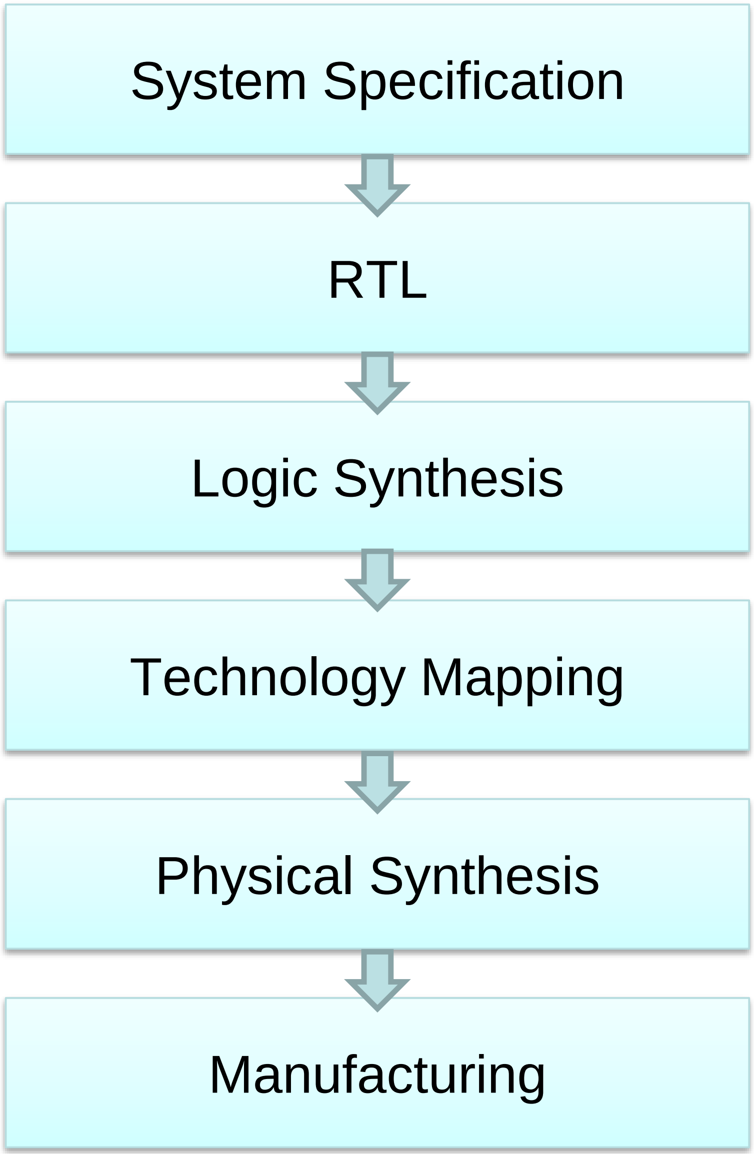
TEXAS
The University of Texas at Austin

Exploring Logic Optimizations with Reinforcement Learning and Graph Convolutional Network

Keren Zhu, Mingjie Liu, Hao Chen, Zheng Zhao and David Z. Pan
ECE Department
The University of Texas at Austin

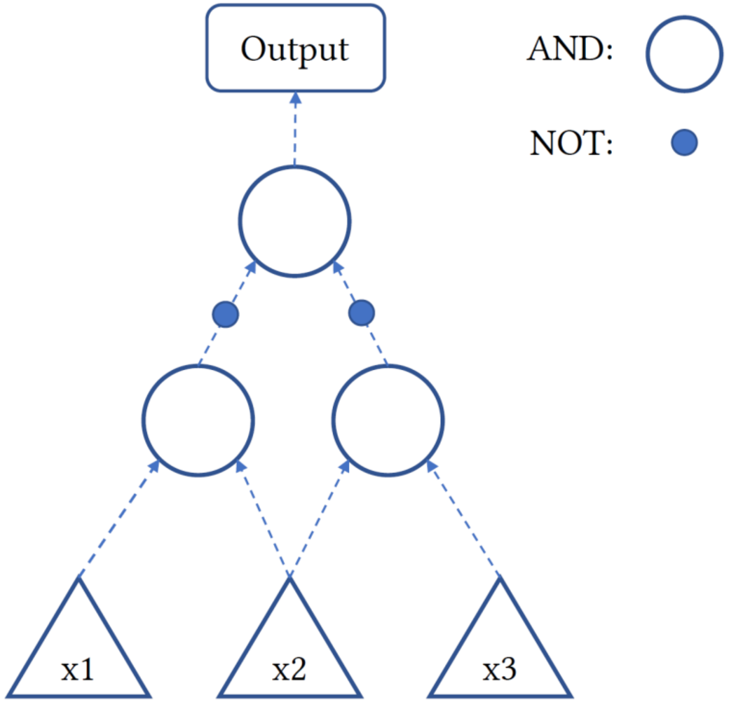
Background: logic optimization in VLSI design

- Modern VLSI design:
  - Abstract architecture -> physical layout
- Logic synthesis and technology mapping:
  - RTL -> Netlist
- Logic optimization:
  - Optimize the logic in logic synthesis
  - Sequential and combinational
- This work focus on combinational logic



Combinational logic is often represented as logic graph

- Standardized logic graph
  - AIG, MIG etc.
  - This work focus on AIG
- Operations on graph can preserve the Boolean logic but change the graph
  - Balance, rewrite etc.
- Modern heuristic use a sequence of operation to optimize the logic graph in number of nodes and logic depth

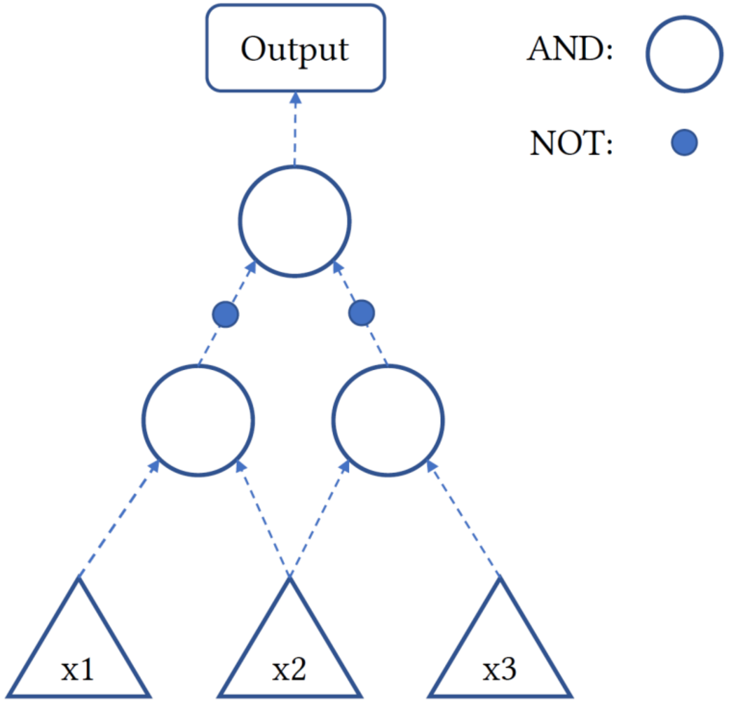


AIG logic $\neg(x1 \wedge x2 \vee x2 \wedge x3)$

Question: what is the optimal operation sequence?

- There are some well-known heuristic
  - E.g. resyn2 in ABC
- The effectiveness of an operation sequence is design-dependent
  - Different circuits have different optimal operation sequence

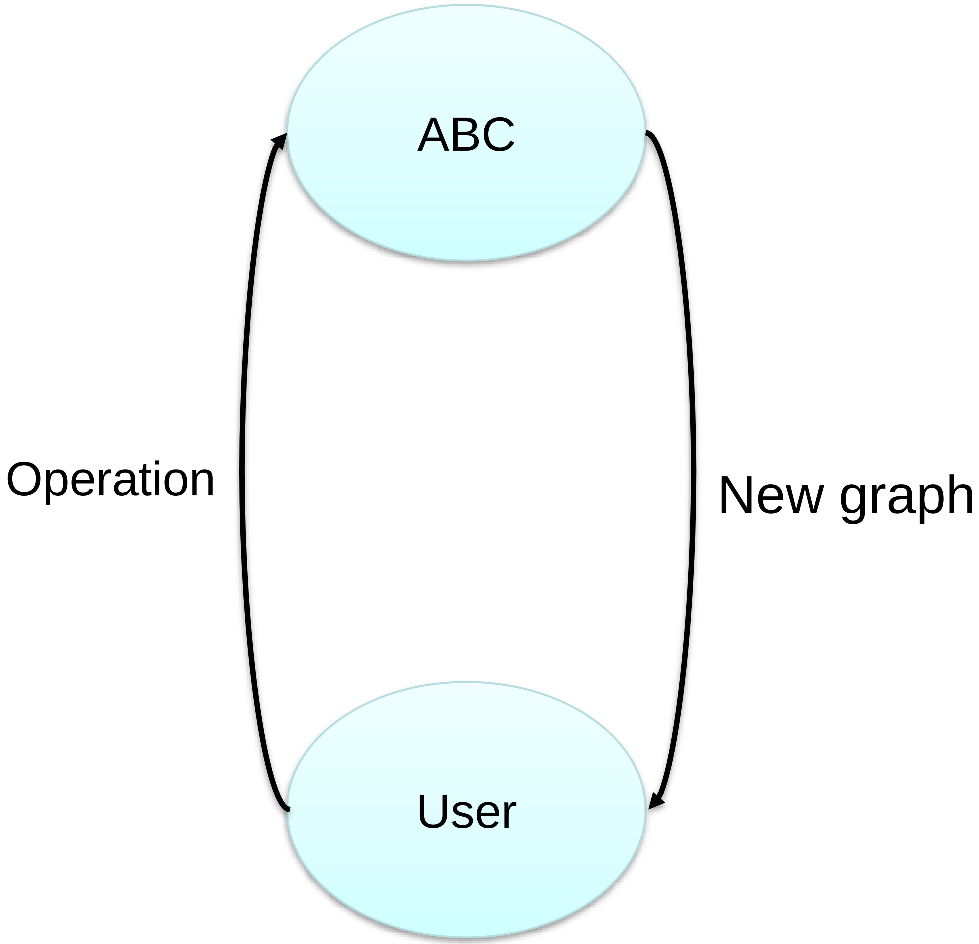
- Question: how to efficiently explore the search space and find good sequences for a new circuit?



AIG logic $\neg(\mathbf{x1} \wedge \mathbf{x2} \vee \mathbf{x2} \wedge \mathbf{x3})$

Operation sequence as an MDP

- The user can observe the current AIG graph
- The user command ABC to execute an operation on the graph
- The user then observe the new graph
- The user want to optimize the graph by repeating this process

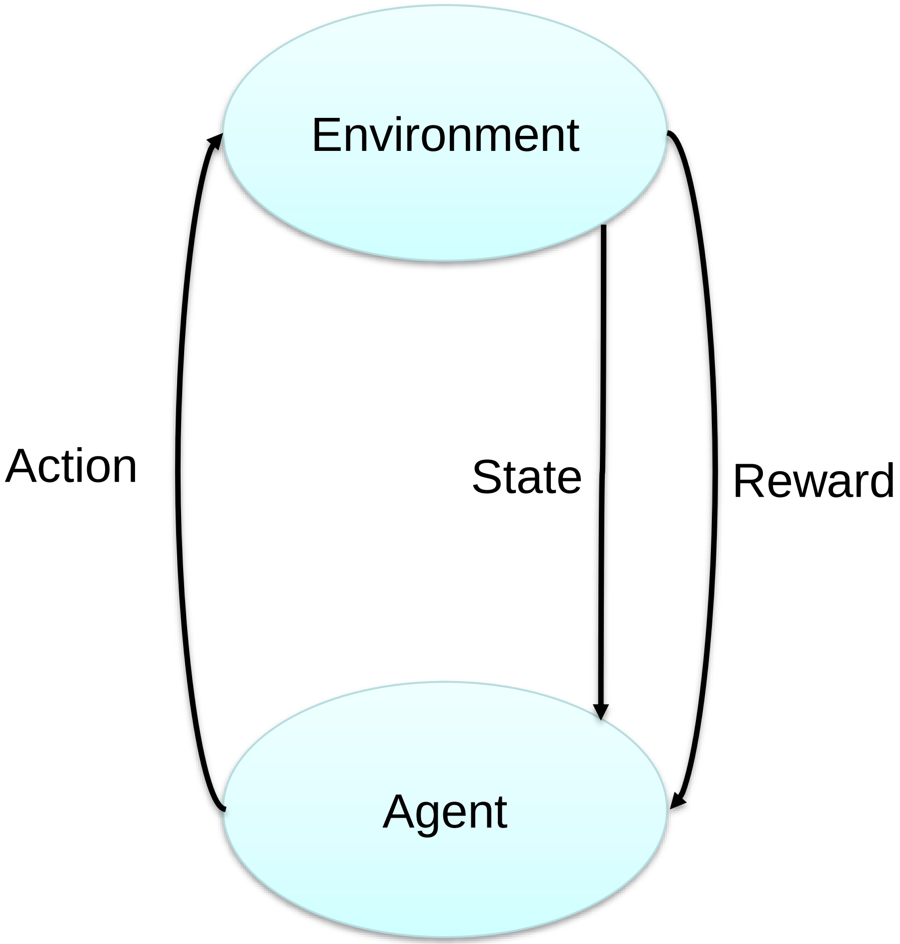


Operation sequence as an MDP

- Then we can formulate this process as a Markov Decision Process (MDP)

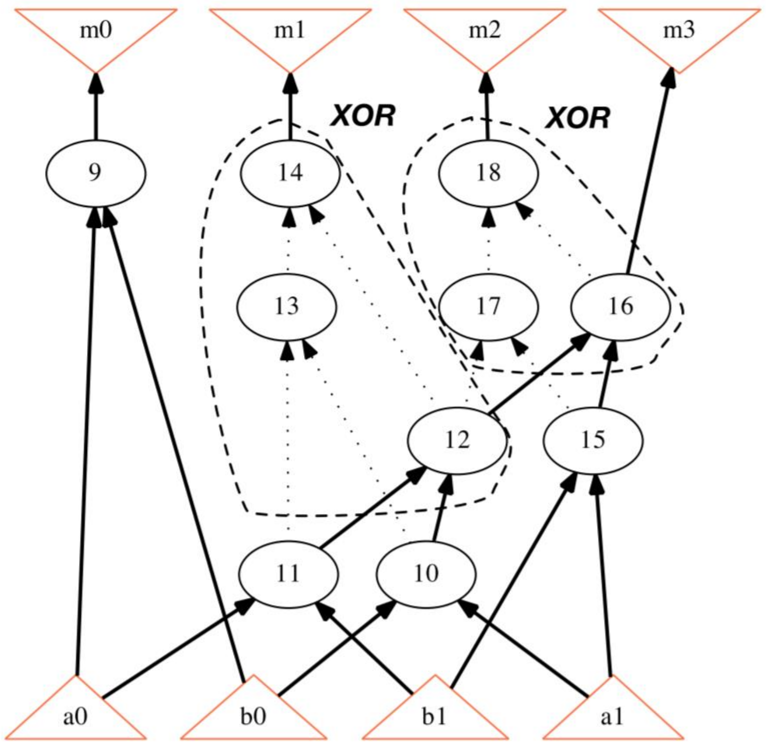
- Action: operation
- State: logic graph
- Reward: improvements

- The process is of Markov property
  - Each operation on the graph is deterministic and not depend on the past
- The state is fully observed
  - Logic graphs contain all the information we need



Key challenge: state representation

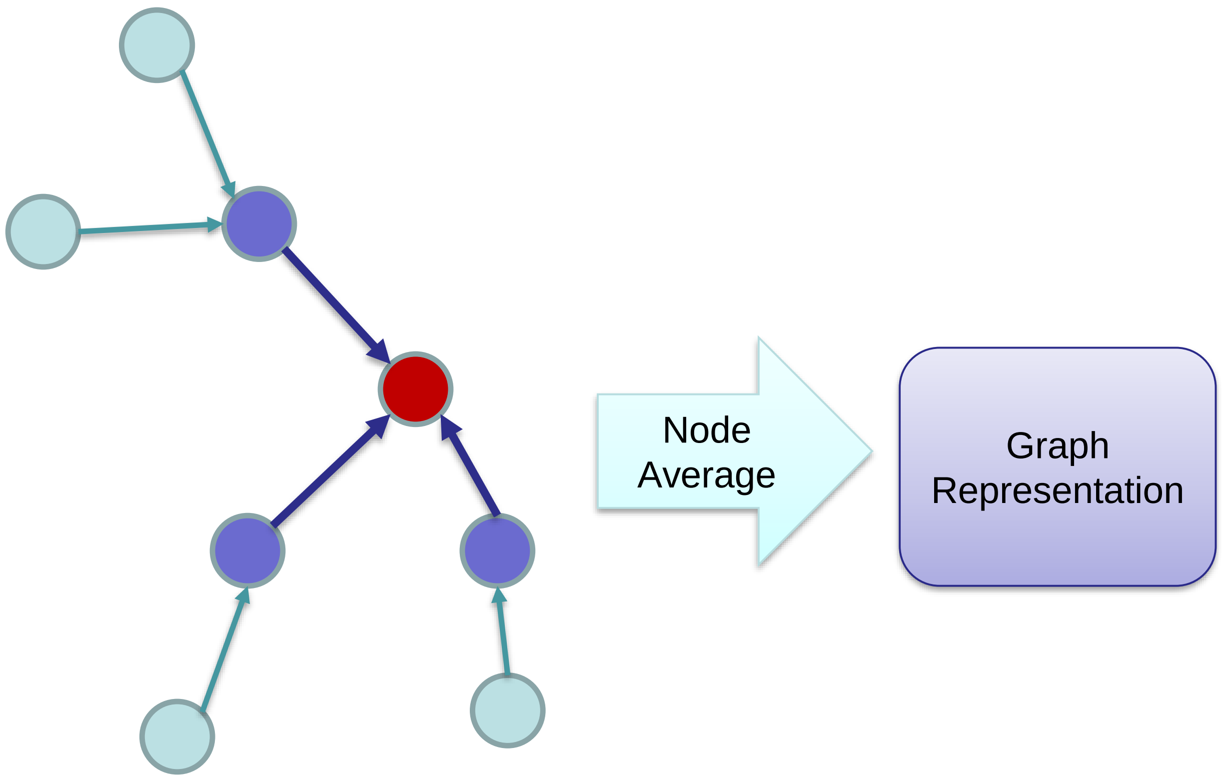
- Reinforcement learning (RL) algorithms often need vector state representation with fixed dimension
- Graph statists can help describing the graph, but not enough
- We also use past action record and graph convolutional network for better state representation



An AIG graph
Courtesy: [Yu+ TCAD18]

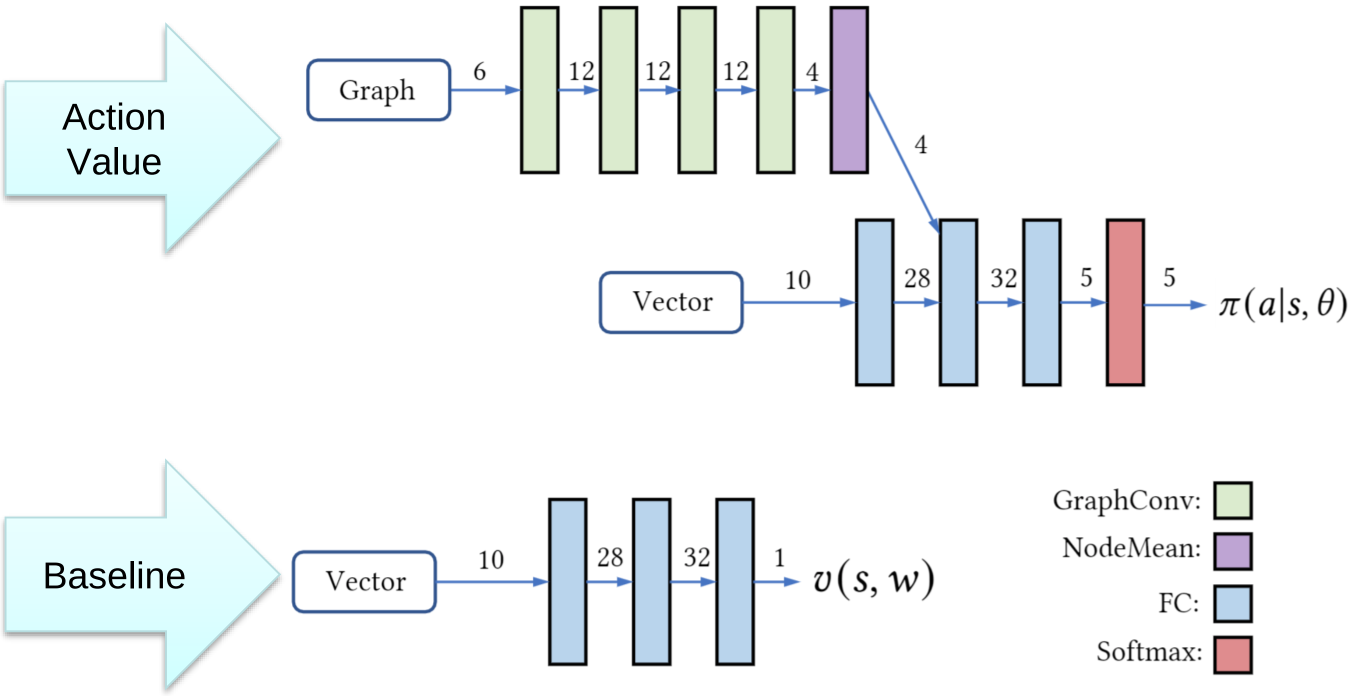
Graph convolution network for graph vectorization

- We use graph convolutional network for assisting state representation
- We use the type as node feature and let graph convolution to aggregate the neighboring features into the nodes
- We take the mean of the graph nodes to obtain a vectorized representation



Policy Gradient RL agent

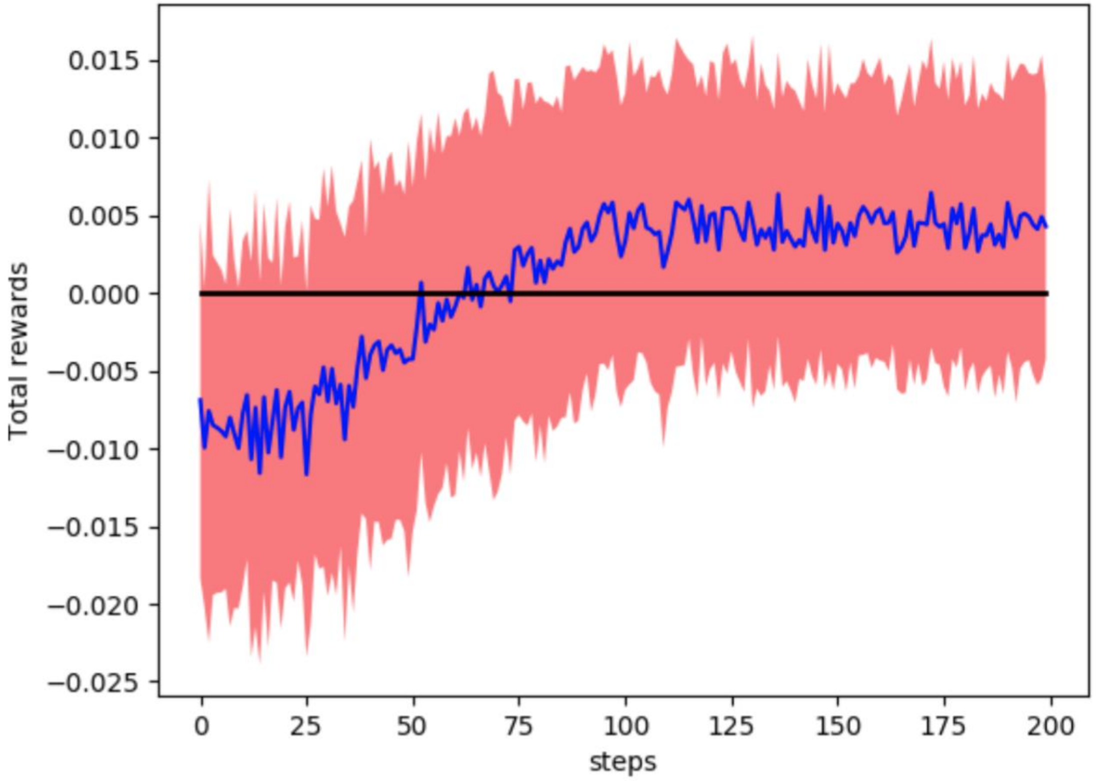
- Policy gradient methods estimate the value of actions
- The RL agent explore the space and update the action value estimation
- We use a simple network for estimating state value as baseline



Experiments: average return

- We use the RL agent to generate operation sequence of 18
  - The same length of running resyn2 twice

- We compare the total rewards over episode
  - RL agent is learning something



Experiments: optimizing the number of nodes

- Optimize number of nodes

Benchmark	Initial		Resyn2 twice		This work (averaged)	
	# Nodes	Depth	# Nodes	Depth	# Nodes	Depth
i10	2675	50	1804	32	1730.2	40.3
c1355	504	25	390	16	386.2	17.6
c7552	2093	29	1416	26	1395.4	27.4
c6288	2337	120	1870	89	1870.0	88.0
c5315	1780	37	1295	26	1337.4	27.2
dalu	1371	35	1103	31	1039.8	33.2
k2	1998	23	1186	13	1128.4	19.8
mainpla	5346	38	3583	26	3438.4	25.0
apex1	2665	27	1966	17	1921.6	19.2
bc0	1592	31	899	17	819.4	18.6
Ratio	1.0	1.0	0.717	0.702	0.698	0.757

Conclusion

- RL algorithm can be applied to the problem of finding a good operation sequence in optimizing combinational logic graph
- Graph mining techniques are useful in extracting information into vectors
- The source codes have been released to public
  - https://github.com/krzhu/abcRL

Future directions

- How to obtain how information from the graph?
  - Graph convolutional network cannot extract the dedicated logic hierarchy from the graphs
  - The usage of past experience in state representation break the perfect Markov property
  - More principal method on vectorizing the graph
- A more generalized action space
  - Currently assume a small discrete action space. How to extend it to general continuous space?
- Multi-objective optimization
- More efficient search space exploration