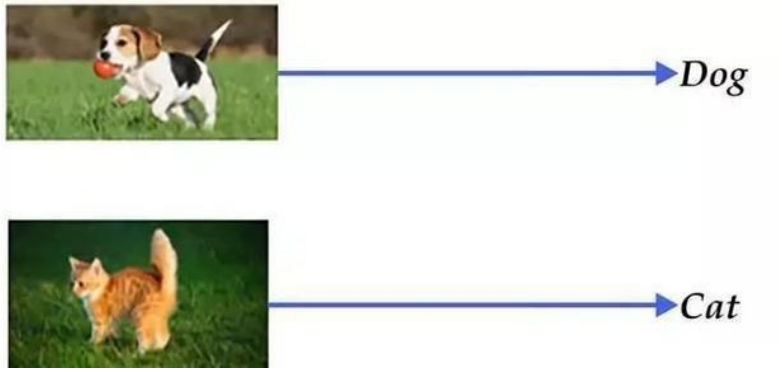
IEEE INFOCOM
IEEE International Conference on Computer Communications
6-9 July 2020 // Virtual Conference

Automating Cloud Deployment for Deep Learning Inference of Real-time Online Services

Yang Li Zhenhua Han Quanlu Zhang Zhenhua Li Haisheng Tan

DNN-driven Real-time Services

Image Classification



Speech Recognition

Neural Machine Translation

Cloud Deployment

DNN Model

require

- Low Latency
  - Network transmission time
  - Task scheduling time
  - DNN inference time
  - ……
- Cost Efficiency

Trade-off between execution time and economic cost

Cloud Deployment

DNN Model

require

- Low Latency
  - Network transmission time
  - Task scheduling time
  - DNN inference time
  - .....
- Cost Efficiency

Trade-off between execution time and economic cost

Model	Min Inference Cost	Max Inference Cost
RNNLM	$0.17	$1.15
Inception-V3	$0.40	$6.39
VGG16	$0.58	$7.26
ResNet-50	$0.60	$4.74
AlexNet	$0.59	$4.45

Inference cost (10000 times) of different models across different cloud configurations.

Here come the problems

I want to deploy my face recognition service on the cloud.

How should I choose the cloud configuration?

Given a configuration, how can I minimize the DNN inference time?

Choose Cloud Configurations

- Choose cloud configurations

Both of them provide over 100 types of cloud configurations!

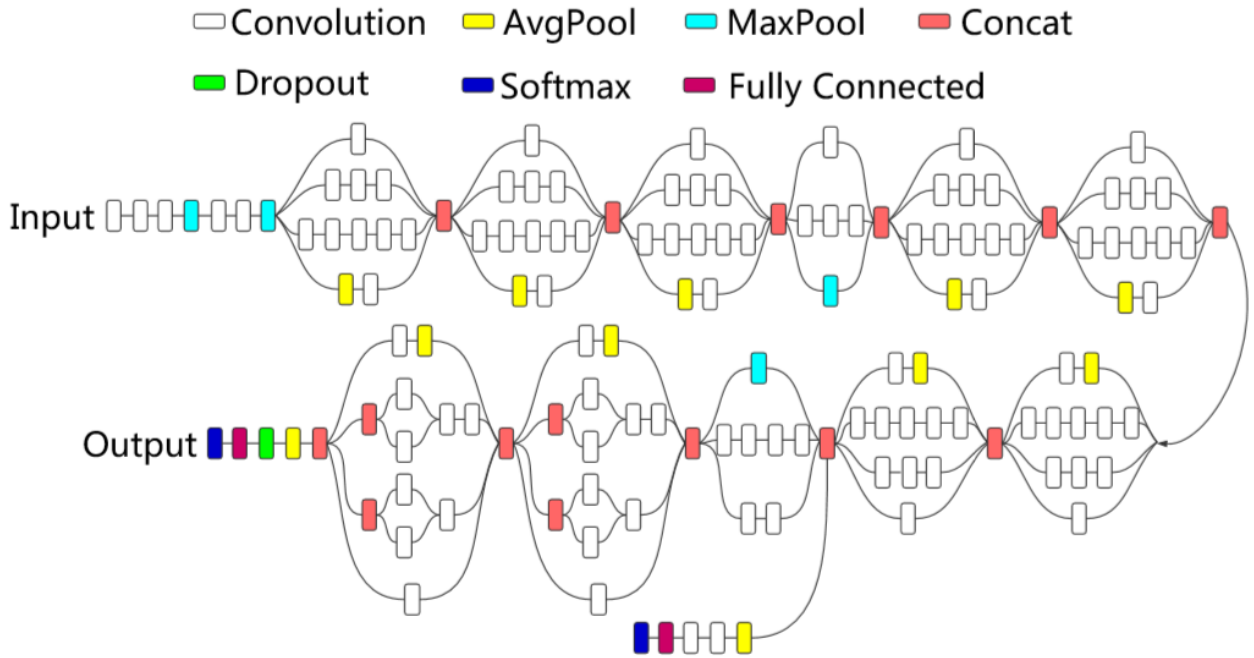
Example: 2 series from over 40 series on Azure!

Instance	Core	RAM	Temporary storage	GPU
NC6 Promo	6	56 GiB	340 GiB	1x K80
NC12 Promo	12	112 GiB	680 GiB	2x K80
NC24 Promo	24	224 GiB	1,440 GiB	4X K80
NC24r Promo	24	224 GiB	1,440 GiB	4X K80

Instance	Core	RAM	Temporary storage	GPU
NC6s v2	6	112 GiB	736 GiB	1X P100
NC12s v2	12	224 GiB	1,474 GiB	2X P100
NC24rs v2	24	448 GiB	2,948 GiB	4X P100
NC24s v2	24	448 GiB	2,948 GiB	4X P100

Reduce DNN Inference Time

- A DNN model can have hundreds to thousands of operations.
- Each operation can be placed on a list of feasible devices (e.g., CPUs or GPUs) to reduce execution time.

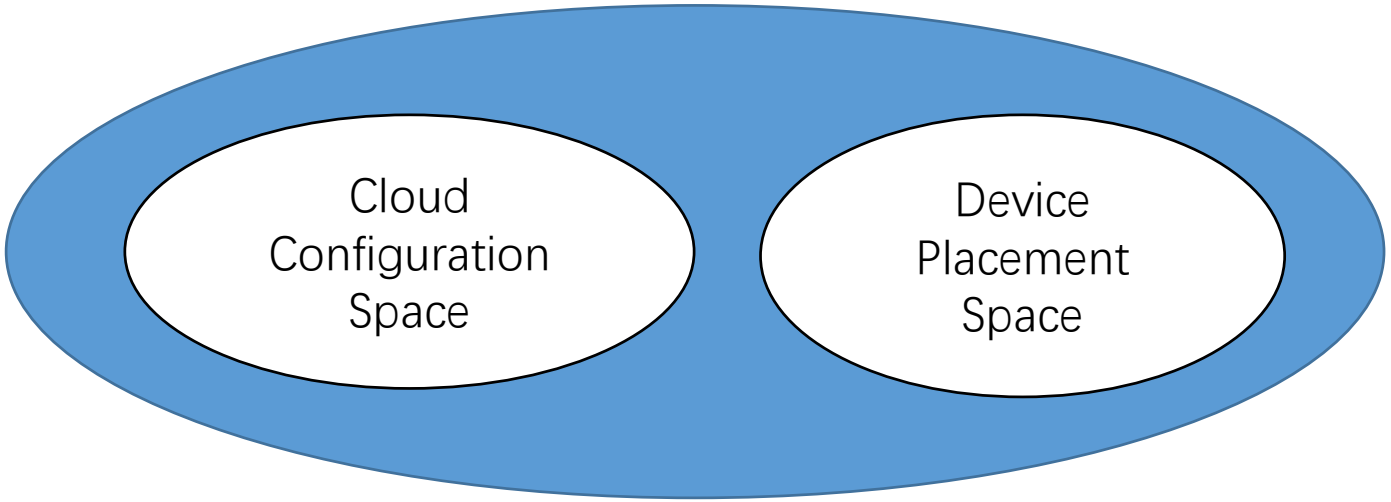


Example: the computation graph of Inception-V3

How to choose the optimal device placement plan?

Challenge

- Huge search space



- Inference cost
  - the price of the cloud configuration * inference time.
    ($/hour) (second/request)

How to automatically determine the cloud configuration and device placement for the inference of a DNN model, so as to minimize the inference cost while satisfying the inference time constraint (QoS)?

Black-box Optimization!

AutoDeep

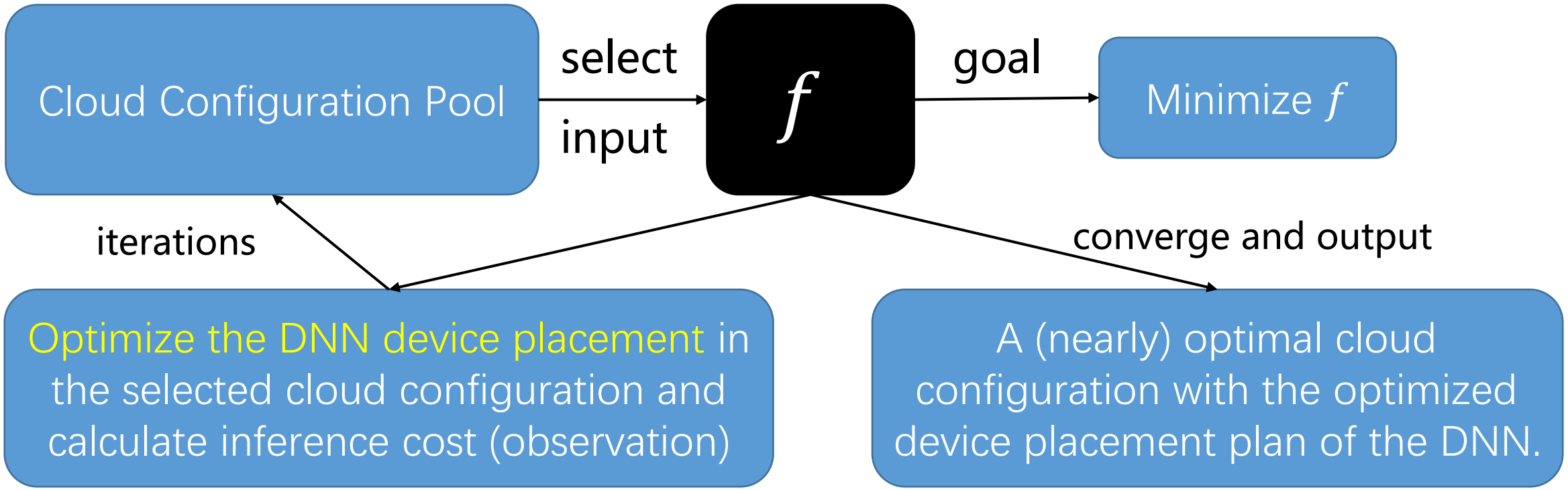
- Given
  - A DNN model
  - Inference time constraint (QoS constraint)
- Goal
  - Compute the cloud deployment with the lowest inference cost

- Two-fold joint optimization
  - Cloud configuration searching
    - Black-box method: Bayesian Optimization (BO)
  - Device placement optimization
    - Markov decision process: Deep Reinforcement Learning (DRL)

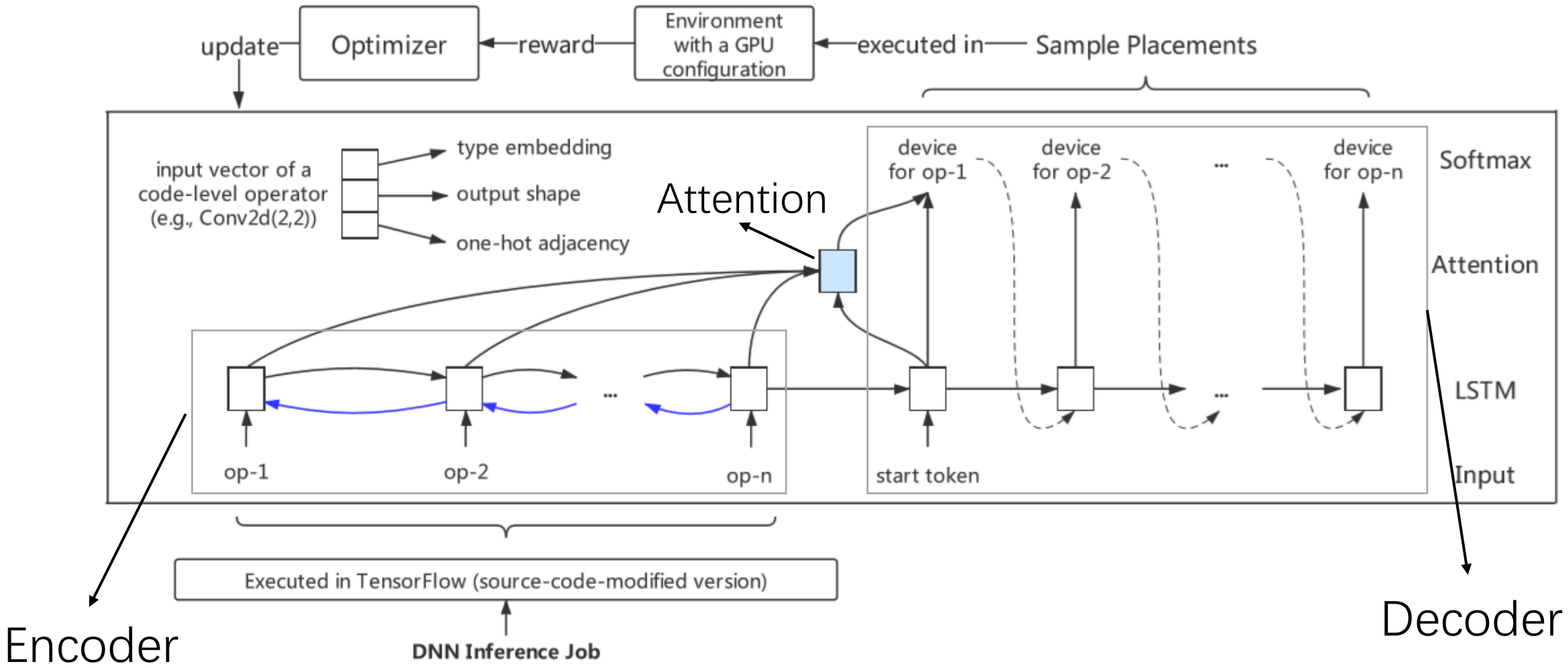
Black-box Optimization

Bayesian Optimization!

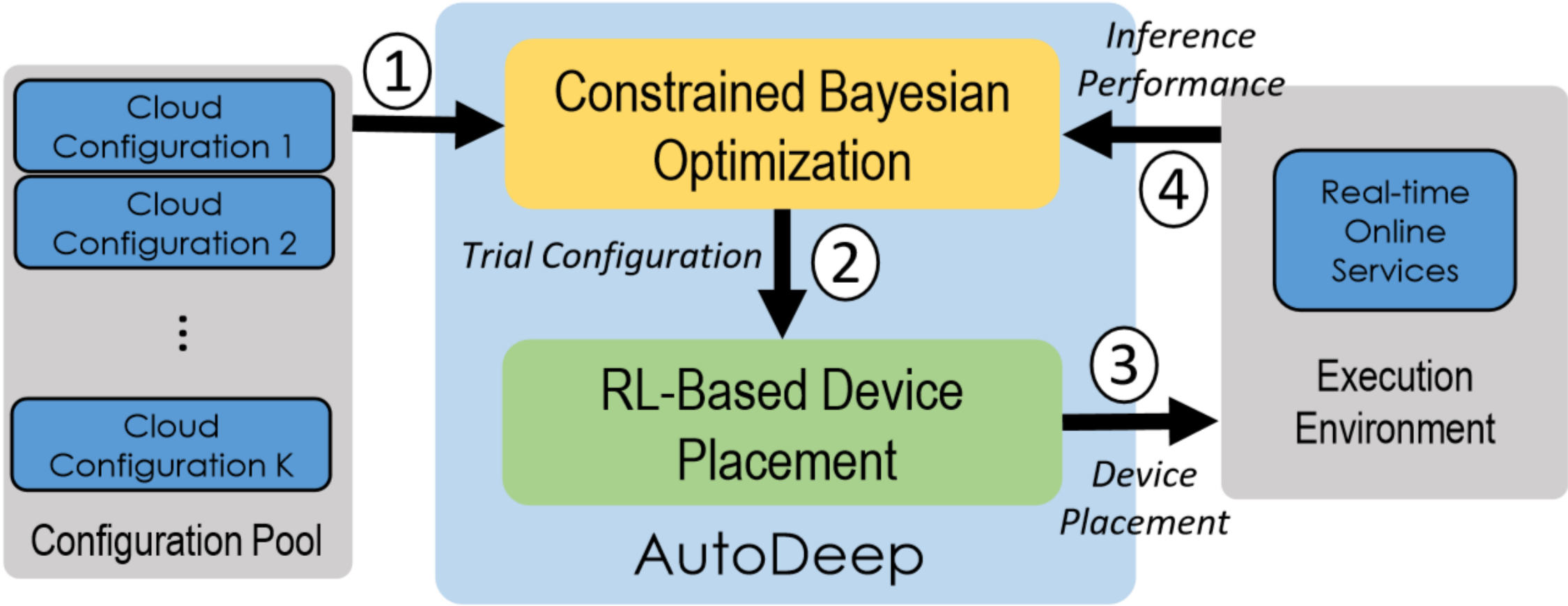
- Regard the inference cost of a given DNN model with a QoS constraint as a black-box function f.



Optimize Device Placement – DRL Model



AutoDeep: Architectural Overview

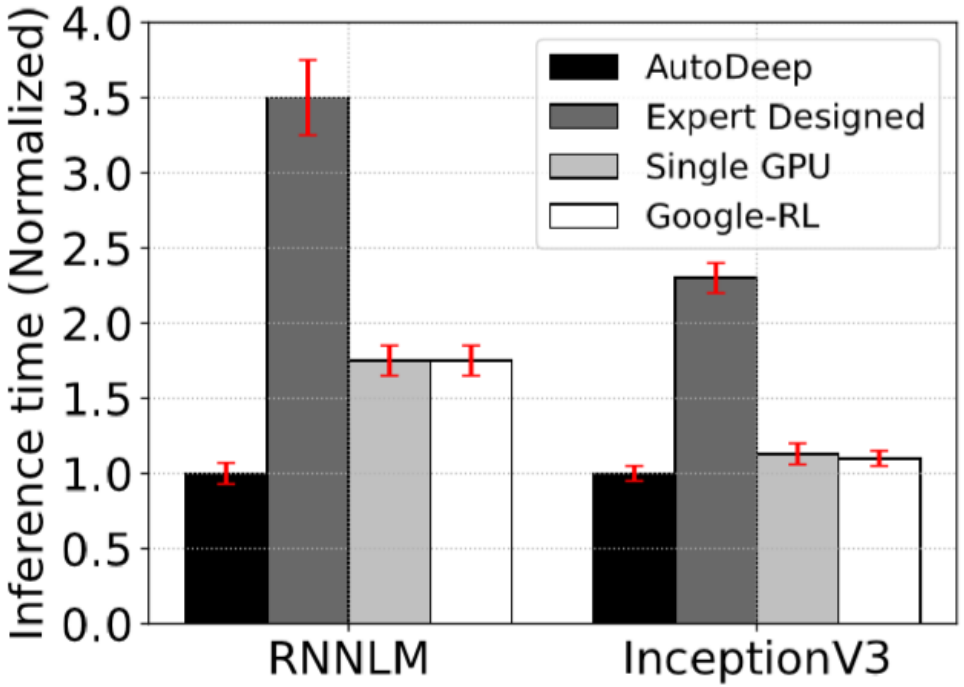


Experiments – Device Placement

- Google RL
  - Algorithm designed by Mirhoseini et al.
  - *[ICML17] Device placement optimization with reinforcement learning*
- Expert Designed
  - Hand-crafted placements given by Mirhoseini et al.
- Single GPU
  - Execution on a single GPU.

CPU	GPU	GPU Number	Price (USD/hour/GPU)
Core i7-5930K	GTX 980Ti	1-3	0.56
Core i7-6850K	GTX 1080	1-4	0.70
Xeon E5-2690 v4	P100	1-4	2.07
Xeon E5-2690 v3	K80	1-4	0.90

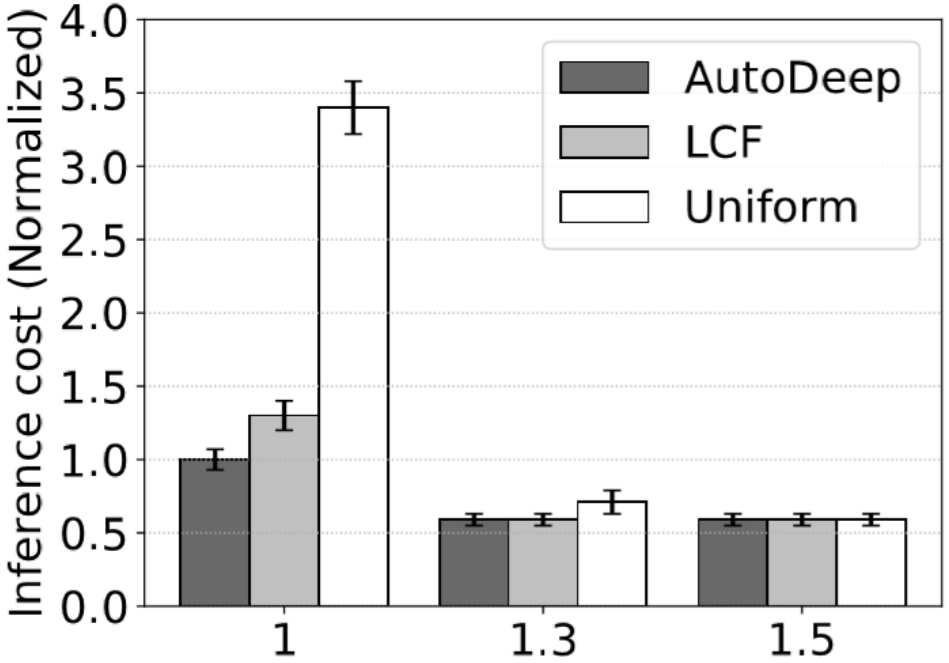
Experiments on 4 K80 GPUs



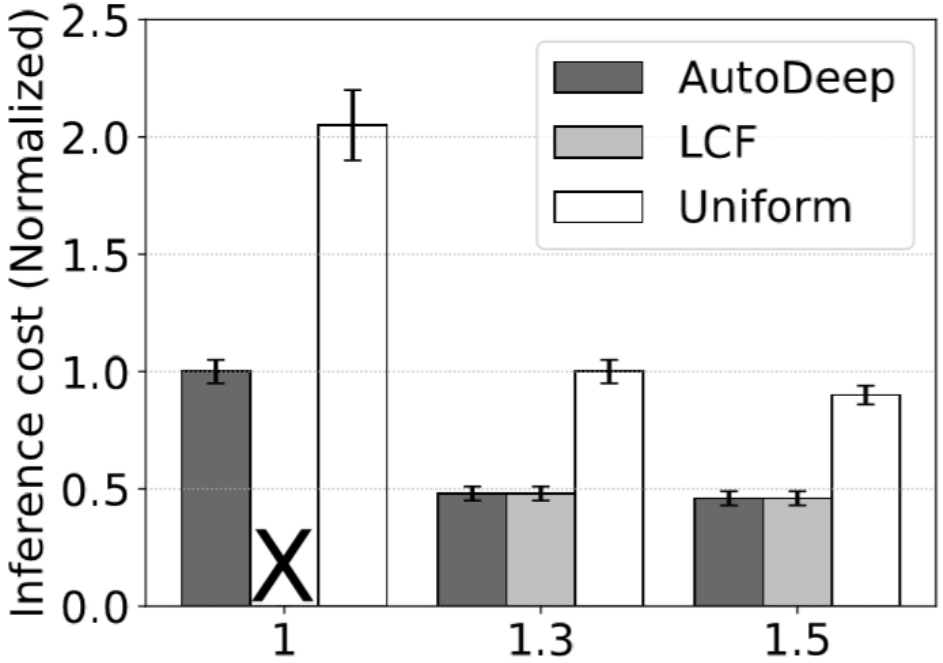
Experiments

- LCF (Lowest Cost First)
  - Try configurations in the ascending order of their unit price
- Uniform
  - Try configurations with uniform probability

QoS Constraint Increasing



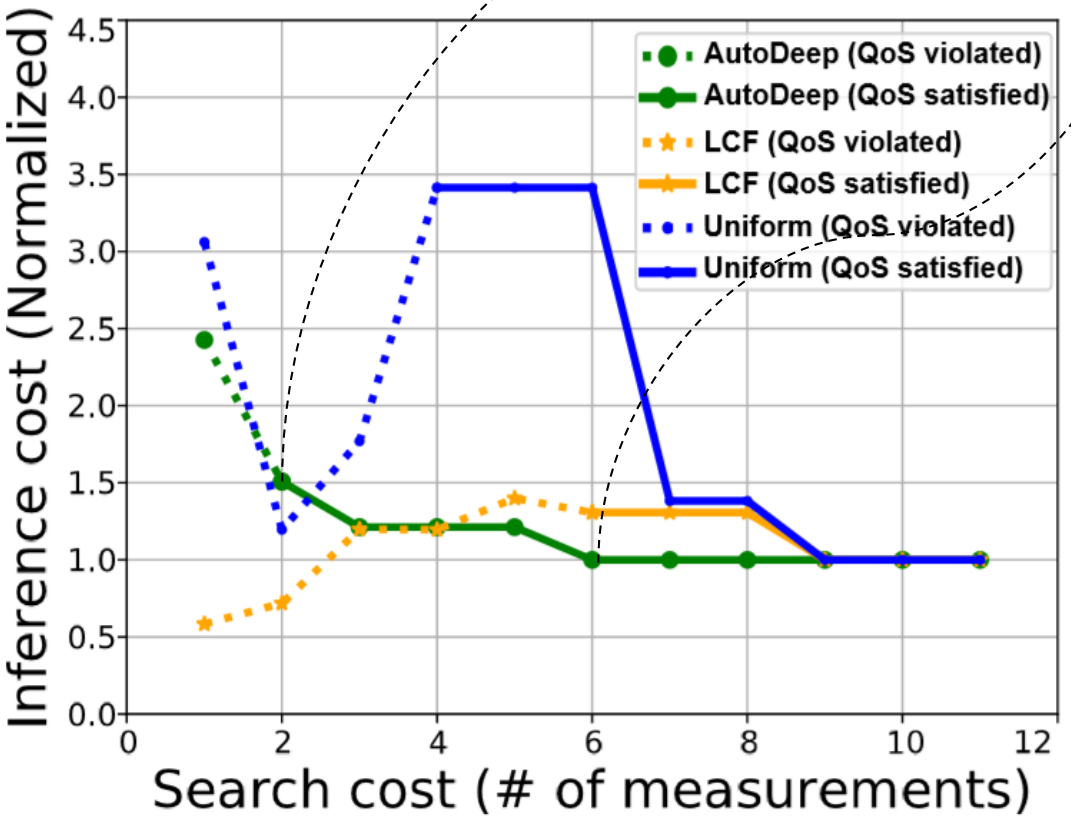
Inference cost of RNNLM under varying QoS constraint



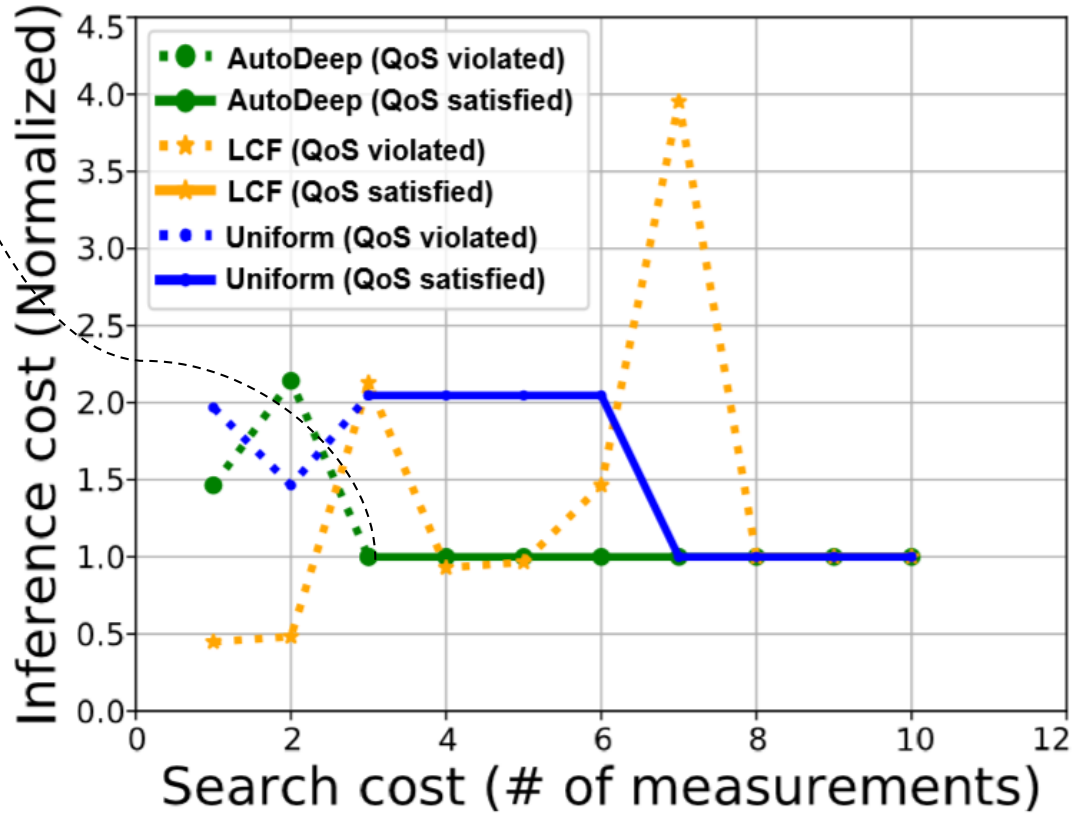
Inference cost of Inception-V3 under varying QoS constraint

Experiments

AutoDeep: Lowest search cost



RNNLM



Inception-V3

Future work

My Email: liyang14thu@gmail.com

- Improve learning efficiency
  - Developing a general network architecture so that re-training is not needed for new DNN inference models
  - Accelerate DRL training process
  - …

- Optimize the system efficiency
  - Over 90% of searching time is wasted to initialize the DNN computation graph
  - Allowing placing operations in a fine-grained manner (i.e., without restarting a job)