Probabilistic Sequential Multi-Objective Optimization of Convolutional Neural Networks

Zixuan Yin, Warren Gross, and **Brett H. Meyer**
McGill University

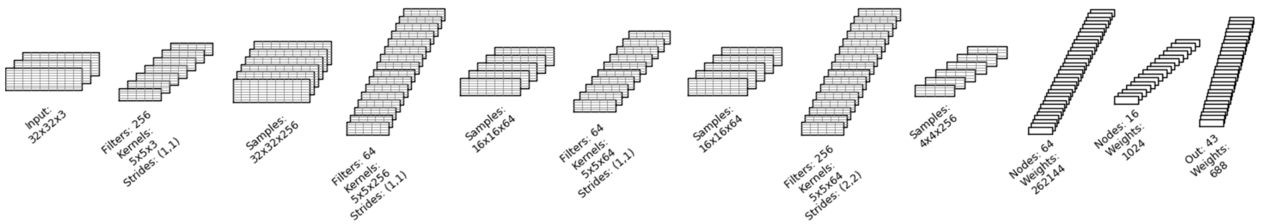
December 8, 2020

So Many Hyper-parameters, So Little Time

- Artificial neural networks are appearing everywhere
  - Embedded and mobile devices
  - In the cloud, and at the edge of the IoT
  - *Different domains have different constraints*
- Hyper-parameter selection affects performance (*accuracy*) and cost (e.g., *energy* or *delay*)
  - E.g., number of layers, types of neurons, etc.
- We *must* jointly optimize software and hardware, but how?
  - No intuitive patterns in large design spaces

Architecture Search is Complex

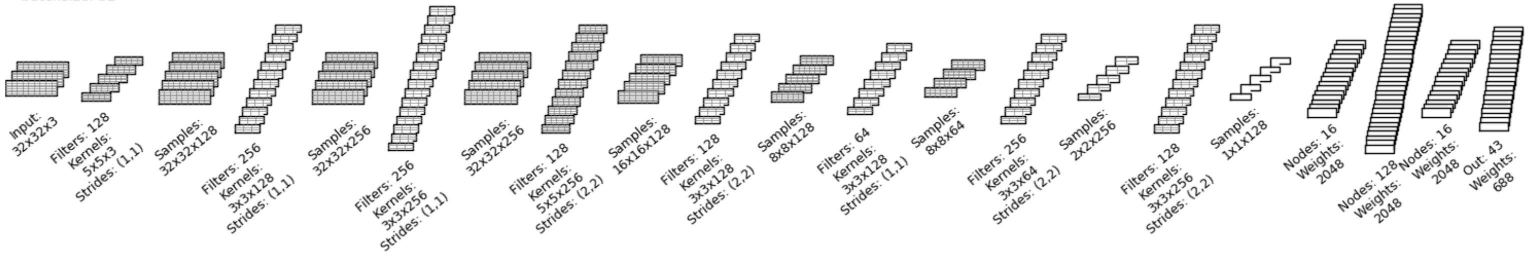
Selected: 101
Error: 0.0478 Accuracy: 0.9522
Cost: 4745500
fc_layers: [64, 16], fc_dropout: [0., 0.5], conv_filters: [256, 64, 64, 256],
conv_depthwise: [False, False, False, True], conv_dropout: [0., 0.2, 0., 0.],
conv_pools: [1, 2, 1, 2], conv_kernels: [5, 5, 5, 5], conv_strides: [1, 1, 1, 2],
learning_rate: 0.00278256, epochs: 40, updates: ADAM, batchsize: 64



Hardware Cost

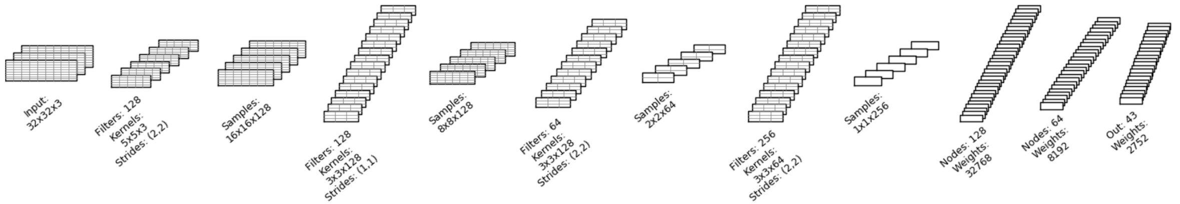
1

Selected: 75
Error: 0.0415 Accuracy: 0.9585
Cost: 10608711
fc_layers: [16, 128, 16], fc_dropout: [0., 0., 0.],
conv_filters: [128, 256, 256, 128, 128, 64, 256, 128], conv_depthwise: [False, True, False, True,
False, False, False, True], conv_dropout: [0., 0., 0., 0., 0.2, 0.2, 0.2, 0.2],
conv_pools: [1, 1, 1, 1, 1, 1, 2, 1], conv_kernels: [5, 3, 3, 5, 3, 3, 3, 3],
conv_strides: [1, 1, 1, 2, 2, 1, 2, 2], learning_rate: 0.02154435, epochs: 20, updates: ADAM,
batchsize: 32



2.2

Selected: 155
Error: 0.0555 Accuracy: 0.9445
Cost: 721307
fc_layers: [128, 64], fc_dropout: [0.5, 0.5], conv_filters: [128, 128, 64, 256],
conv_depthwise: [False, True, False, False], conv_dropout: [0.2, 0.2, 0., 0.],
conv_pools: [1, 2, 2, 1], conv_kernels: [5, 3, 3, 3], conv_strides: [2, 1, 2, 2],
learning_rate: 0.00774264, epochs: 40, updates: SGD, batchsize: 32



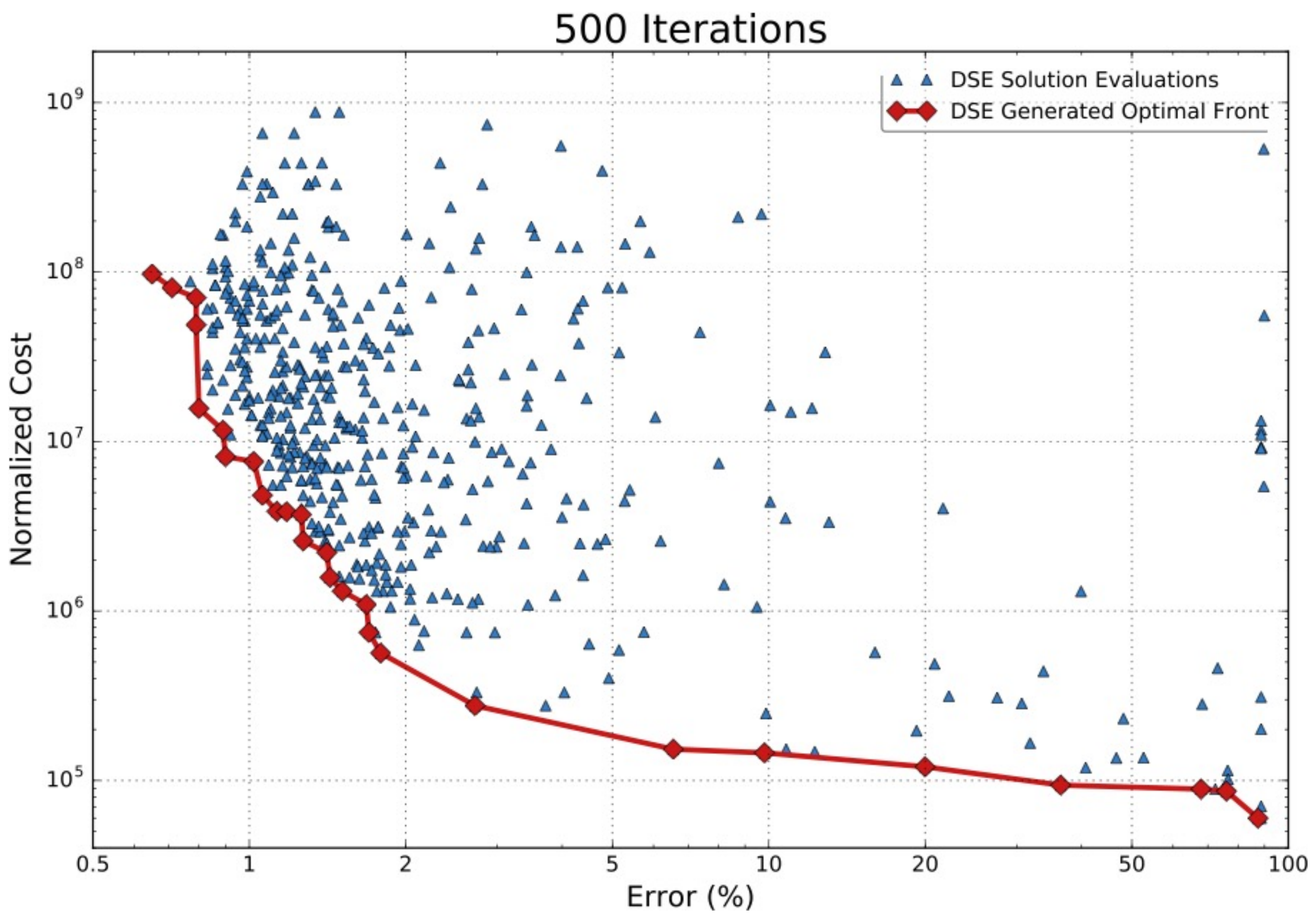
0.15

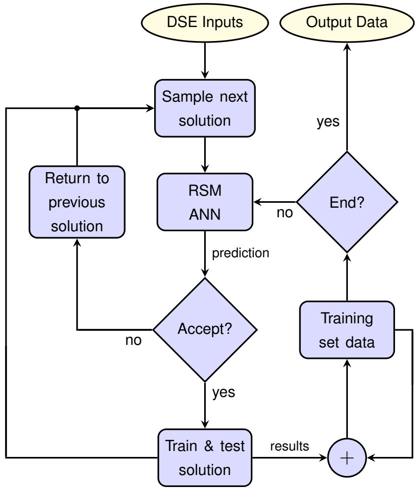
Ordinary People Accelerating Learning

- OPAL models the DNN design space with a many-dimensional *response surface* (hyperplane)
- OPAL is a *sequential, model-based* optimizer (SMBO)
- A meta DNN (*mDNN*) learns the relationship between target DNN (*tDNN*) hyper-parameters, and accuracy
  - Select a tDNN, predict its performance
  - If the tDNN is promising, evaluate it; results are used to retrain mDNN
- Returns a near-Pareto-optimal set
  - E.g., from *high accuracy*, **high cost**, to **low accuracy**, *low cost*, and everything in between

Example

- Pareto-optimal front evolves with each iteration

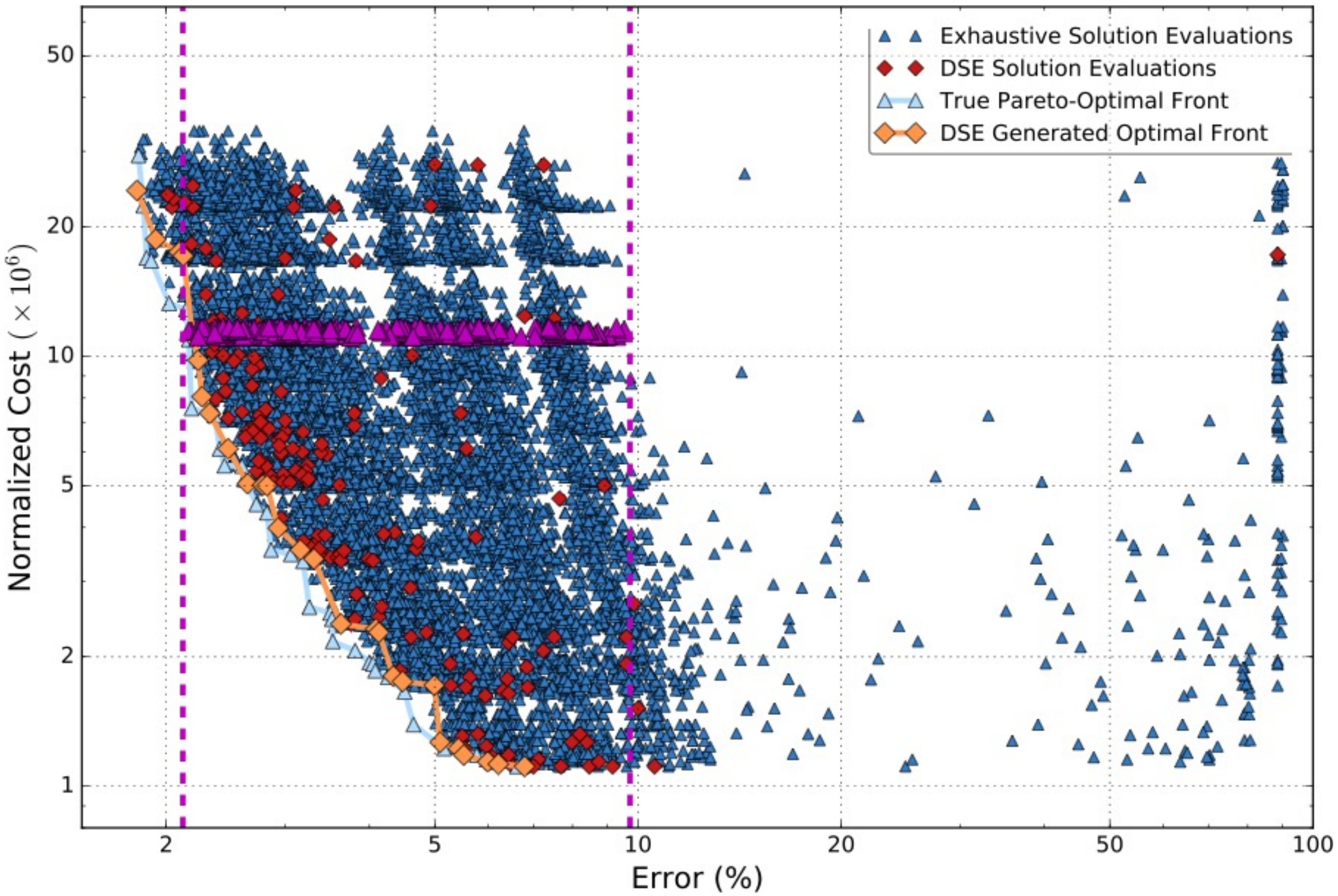




10^7 configurations

- 1-2 CNN layers
- 8-128 filters per CNN
- Kernel: 1x1-5x5
- Max-pool: 2x2-4x4
- 1-2 FC layers
- 10-250 nodes per FC
- LR: 0.01-0.8

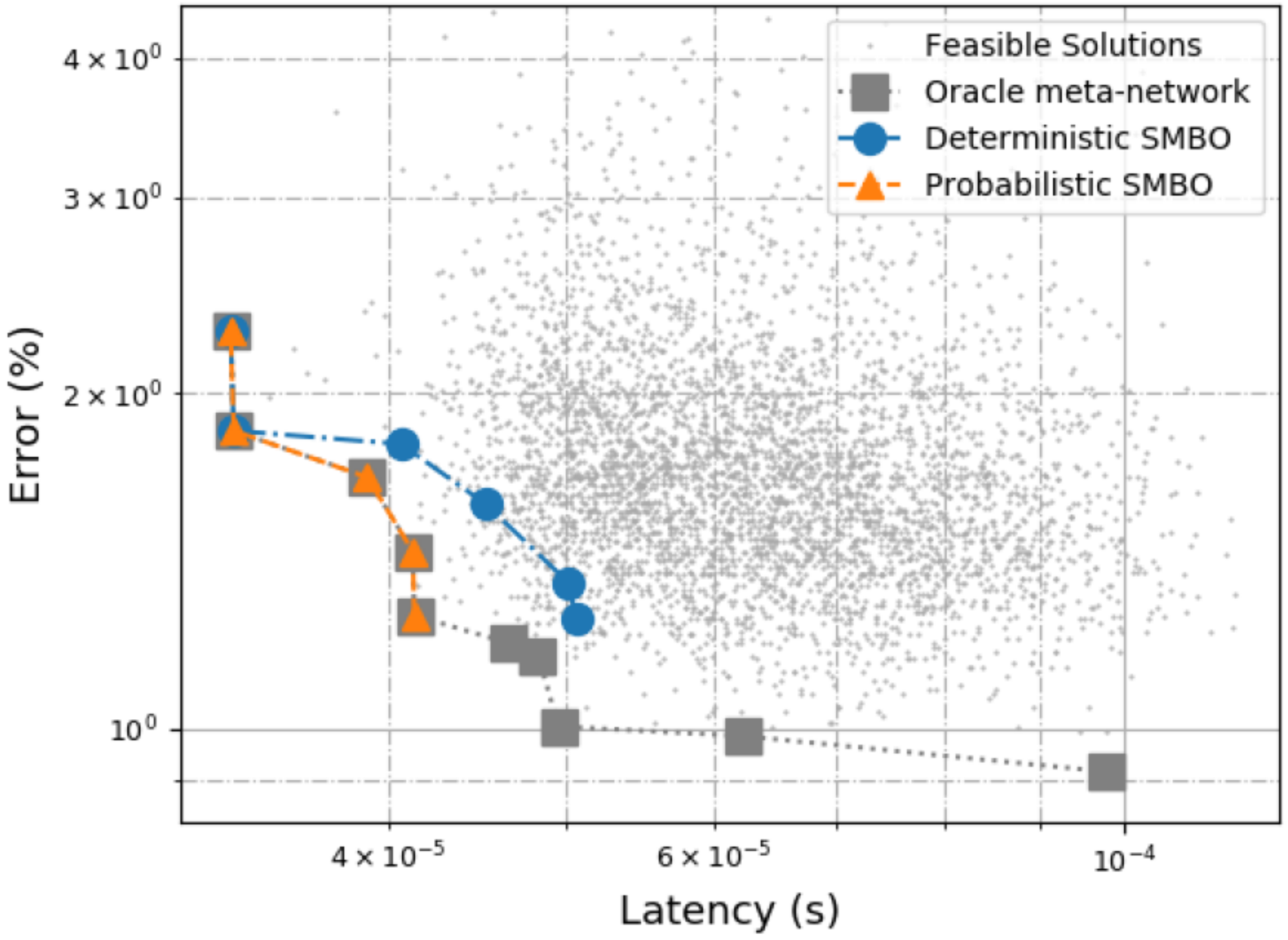
OPAL Gives the Best* Trade-offs



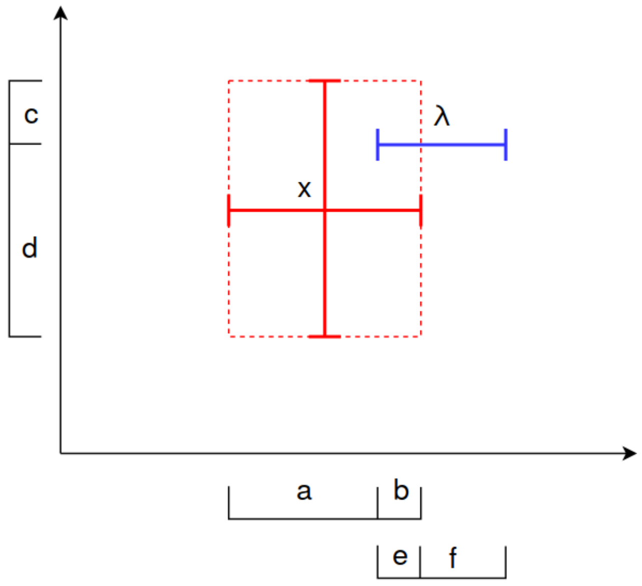
- *Majority of explored points* are near the Pareto-optimal front
- Many fewer *objectively bad* solution are evaluated

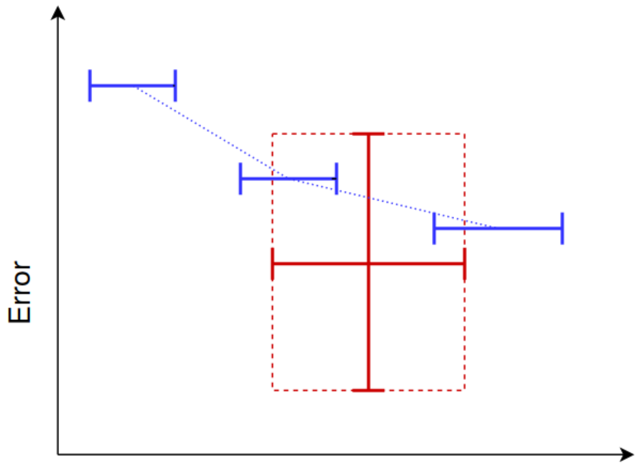
A New Challenge(r) Has Appeared

- We only find good solutions quickly if we make good predictions



Probabilistically Pareto-Optimal DNNs

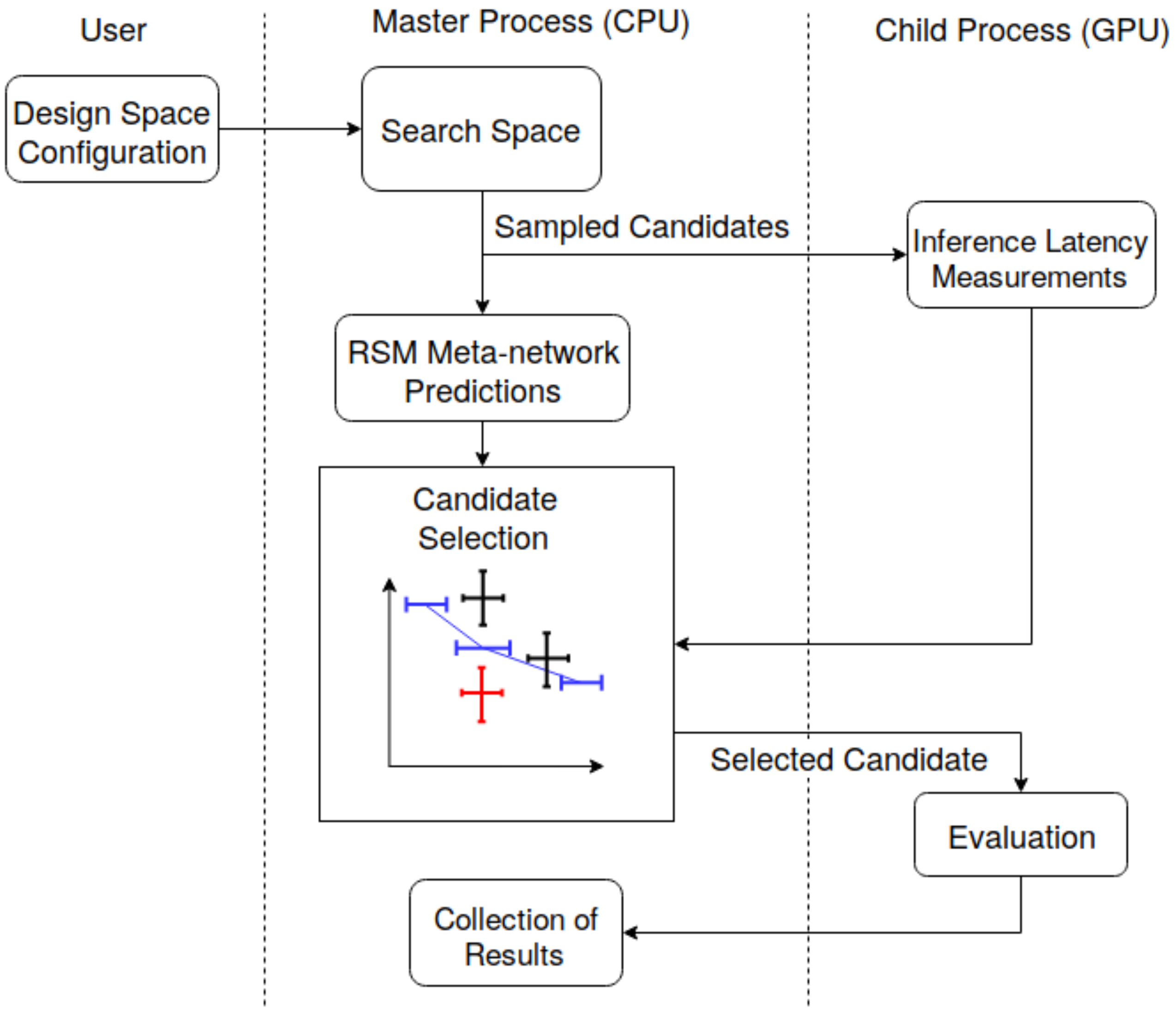




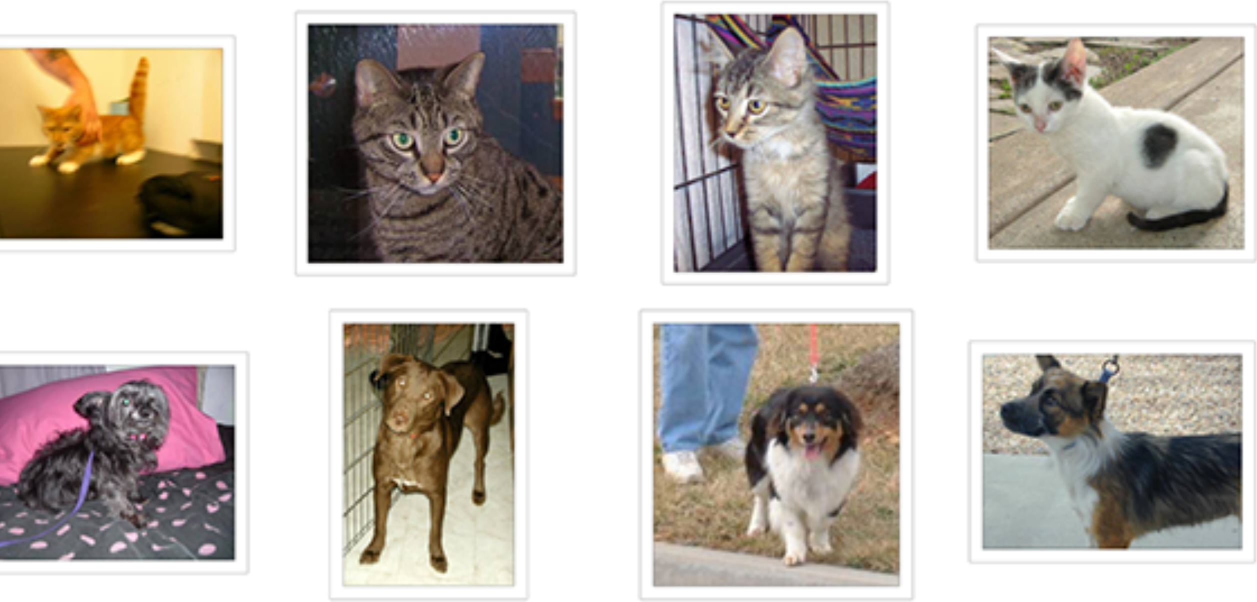
- Uncertainty in candidate comparison → poor choices
  - Strong candidates are discarded
  - Weak candidates are evaluated
- *Probabilistic Pareto Efficiency* (PPE) captures the likelihood a candidate advances the POF

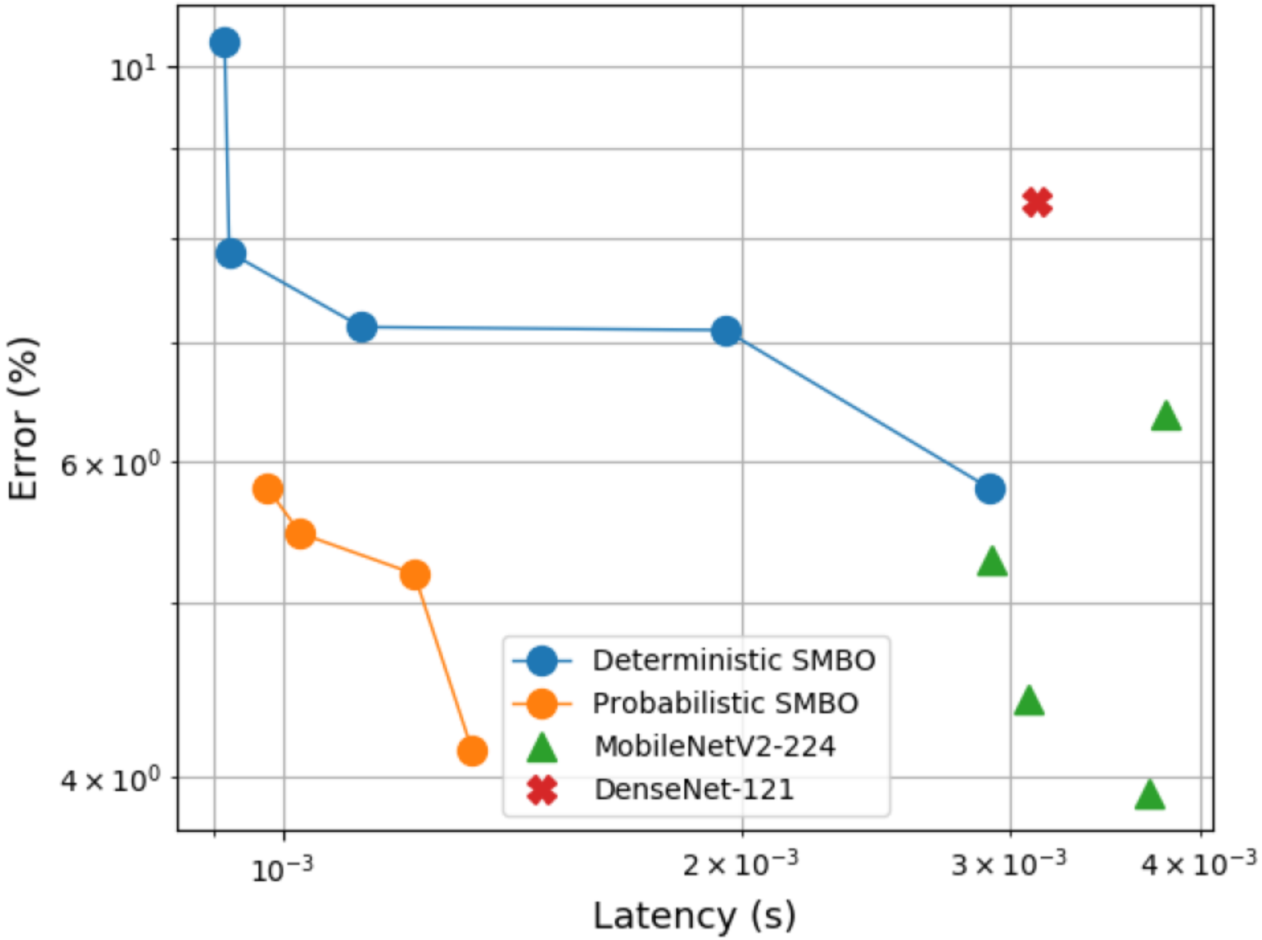
$$PPE(x|\Lambda) = \prod_{\lambda \in \Lambda} P(\lambda \nprec x) + \sum_{\lambda \in \Lambda} P(x \prec \lambda)$$

Searching with Probabilistic Pareto Efficiency

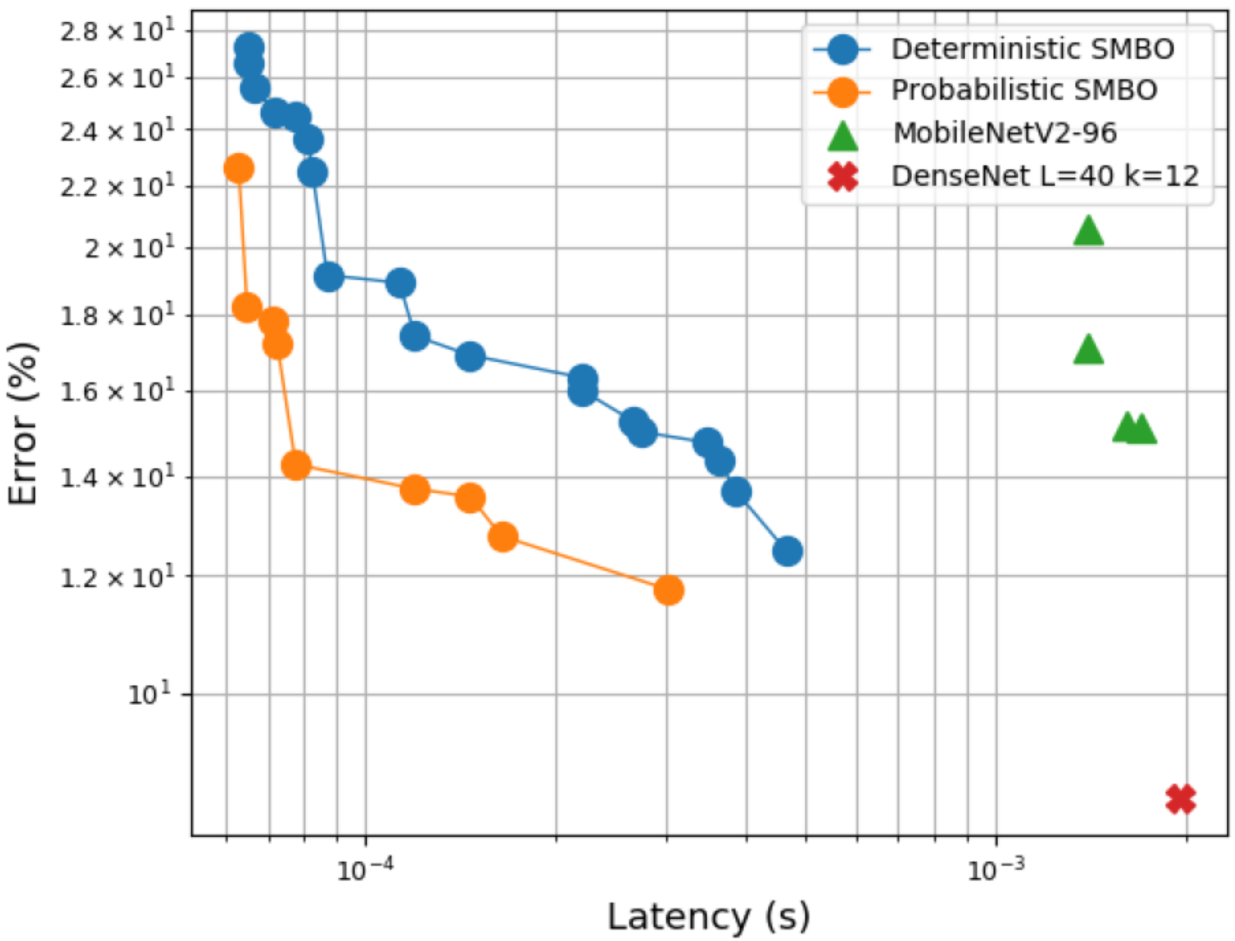


Results: Cats vs Dogs

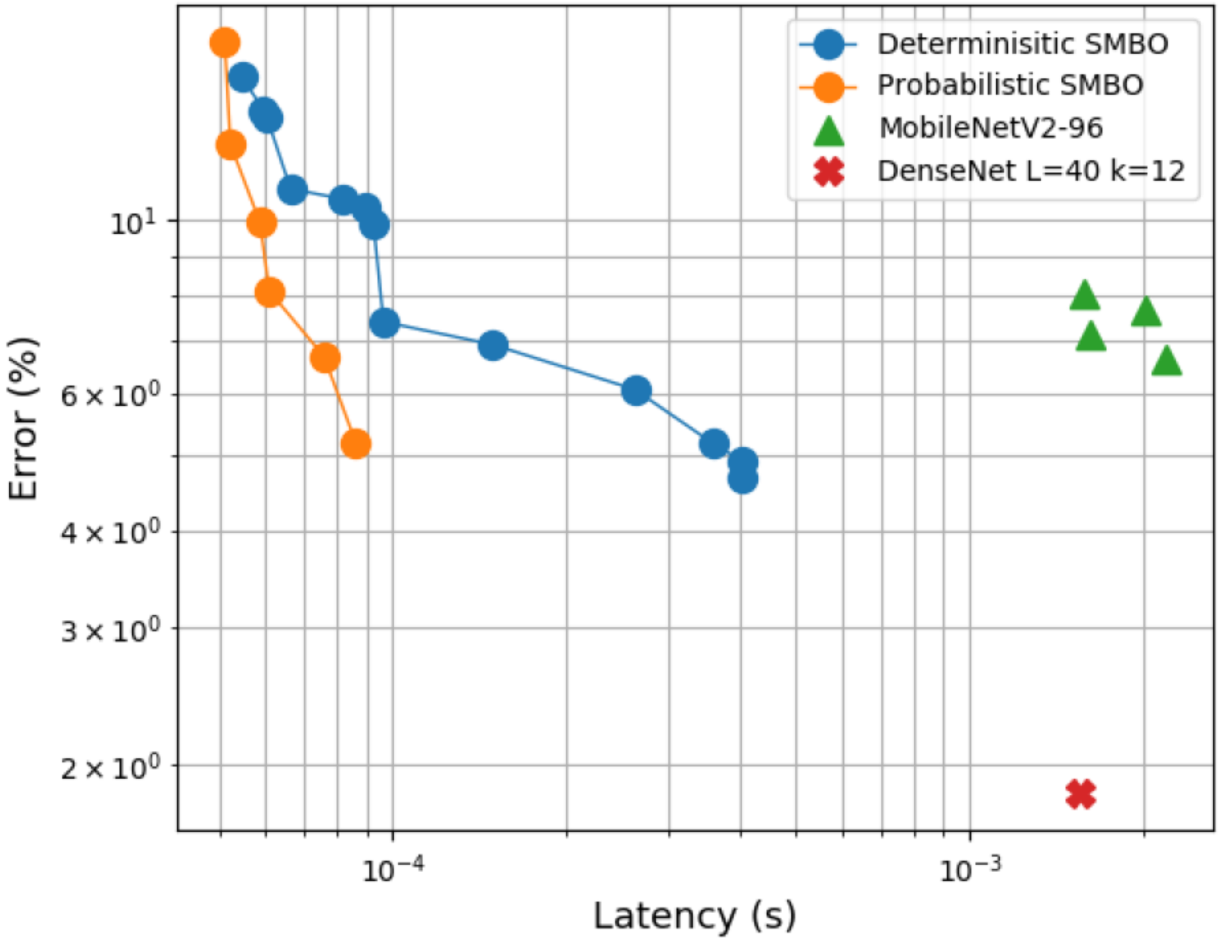




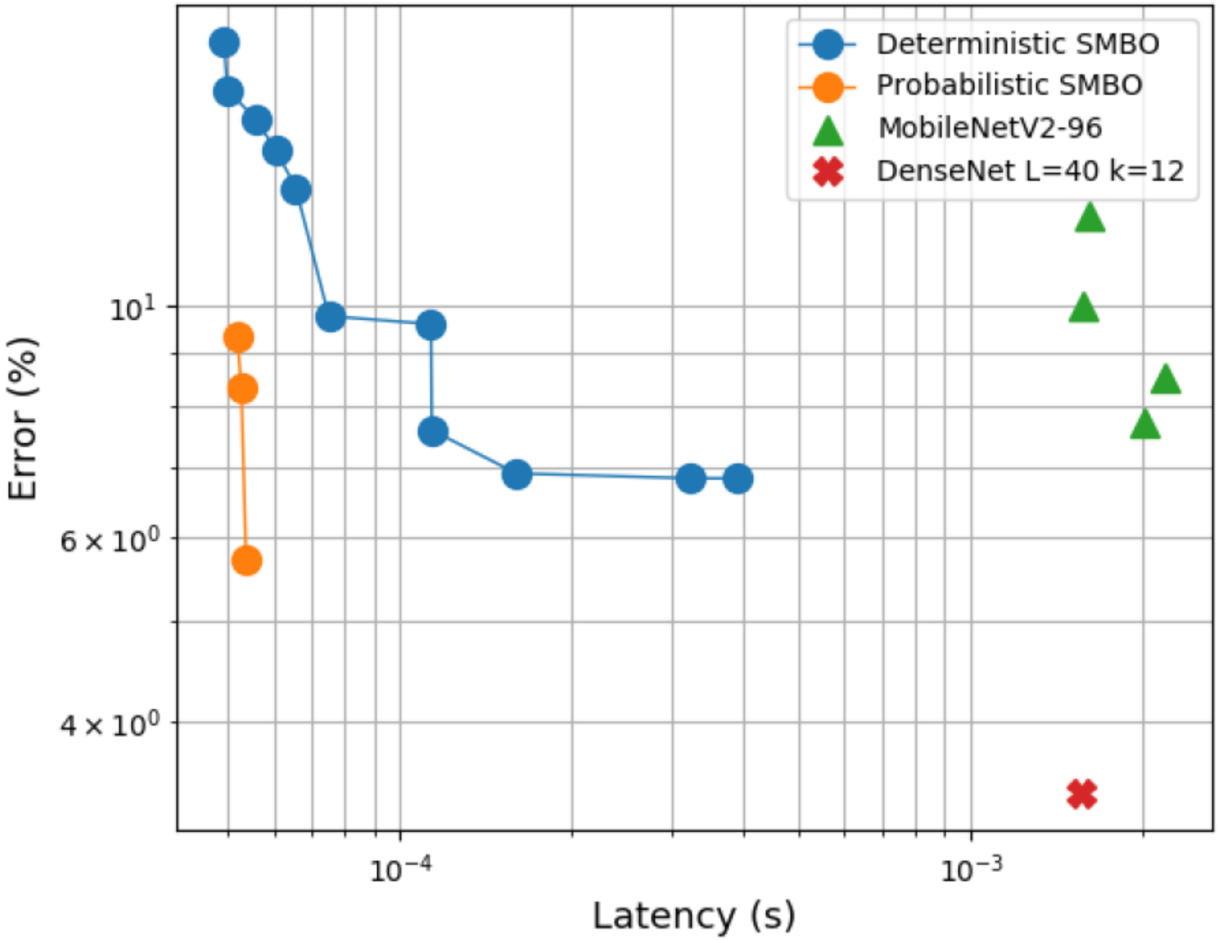
Results: CIFAR-10



Results: German Traffic Signs

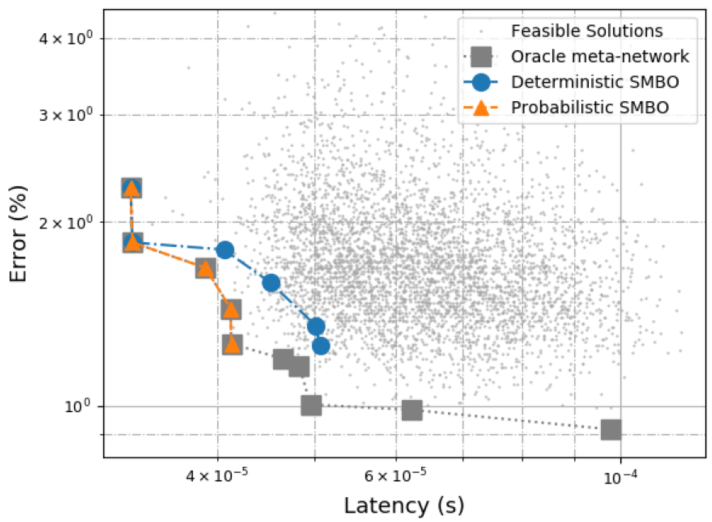


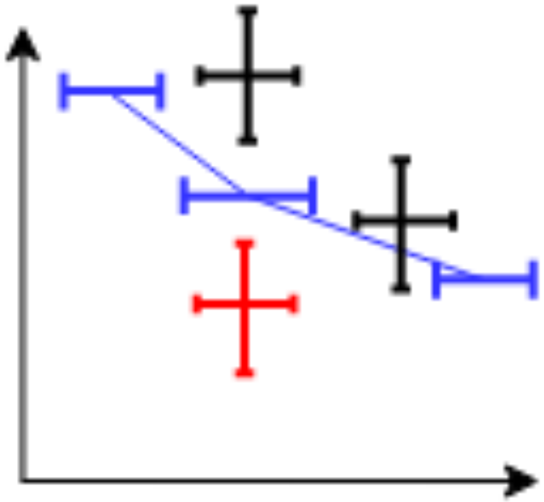
Results: Street View House Numbers

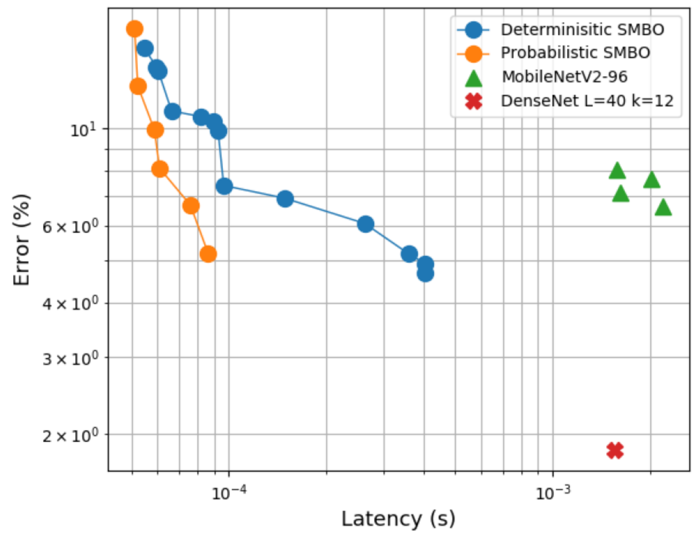


Conclusions

- SMBO optimization depends on accurate performance estimation to make good choices
- Accounting for uncertainty in candidate selection improves SMBO performance dramatically







References

S. C. Smithson, G. Yang, W. J. Gross, and B. H. Meyer, “Neural networks designing neural networks: Multi-objective hyper-parameter optimization,” *ICCAD 2016*.

Z. Yin, W. J. Gross, and B. H. Meyer, “Probabilistic sequential multi-objective optimization of convolutional neural networks,” *DATE 2020*.

Thank you!
Questions?