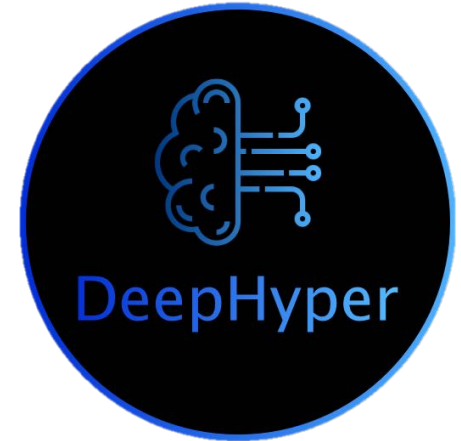


AUG 09, 2019

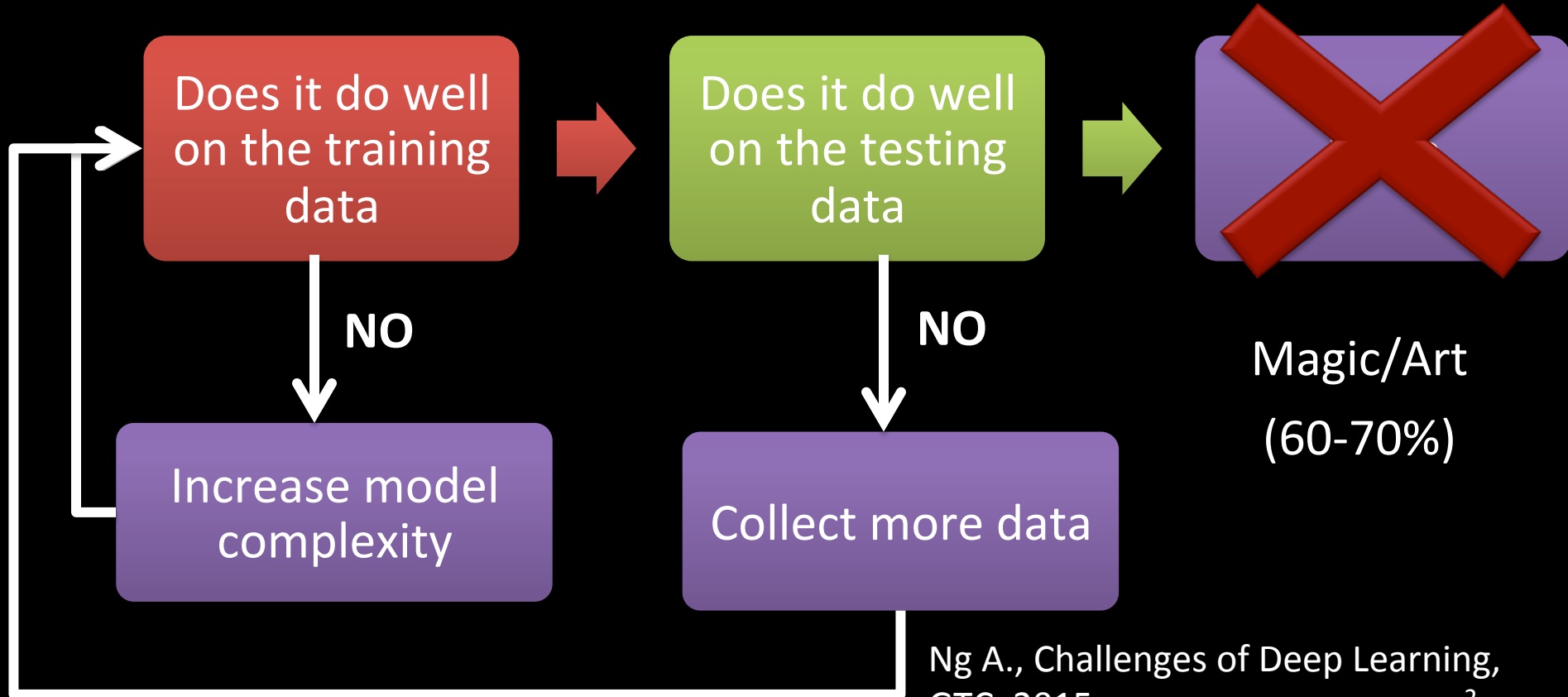


Hyperparameter Optimization and DeepHyper



Prasanna Balaprakash and Misha Salim
Argonne National Laboratory

Supervised (deep) learning



Automated machine learning for deep learning

Lower-level problem:

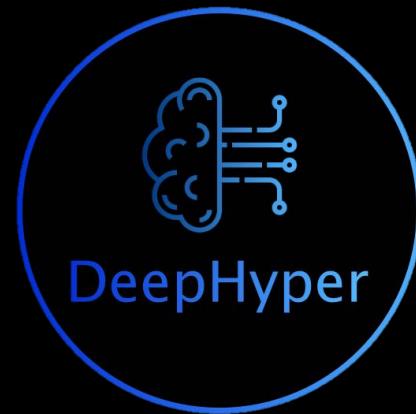
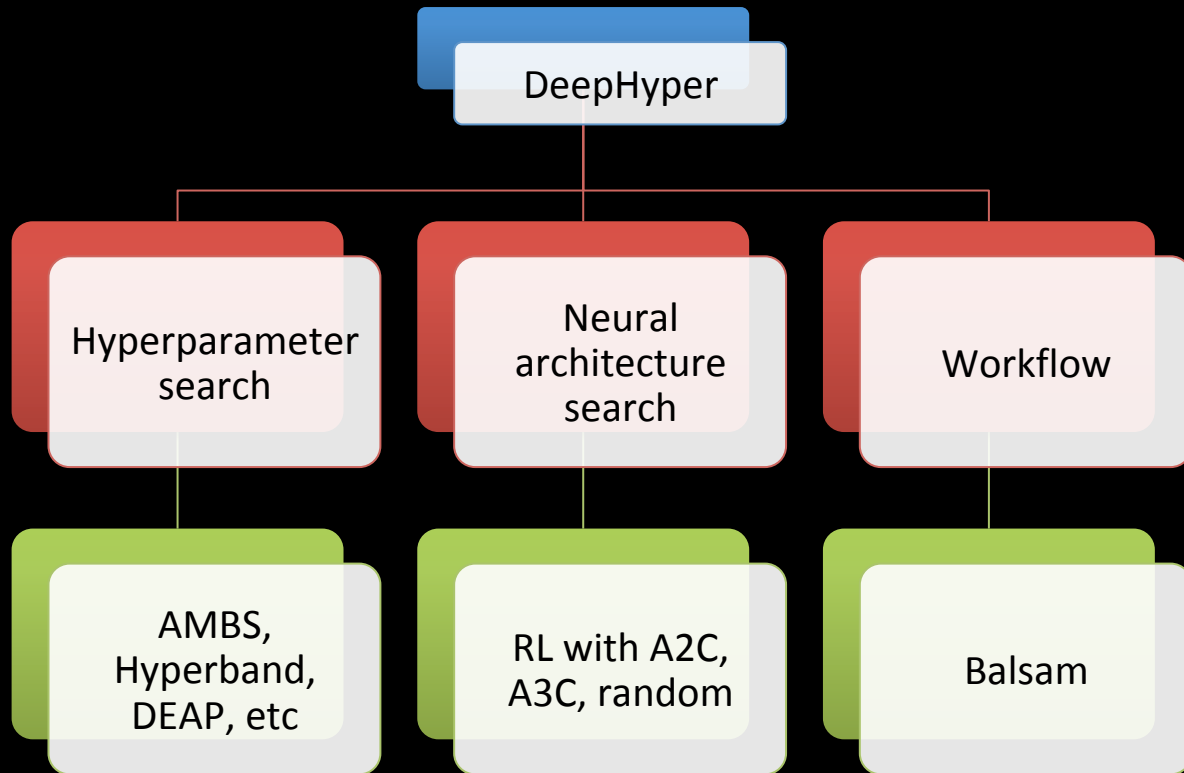
$$\text{solve} \quad \underset{w}{\text{minimize}} \quad \text{err}_T([\mathcal{X}_A, \mathcal{X}_P]; \mathcal{T}; w)$$

Upper-level problem:

$$\text{solve} \quad \underset{\mathcal{X}_A, \mathcal{X}_P}{\text{minimize}} \quad \text{err}_V([\mathcal{X}_A, \mathcal{X}_P]; \mathcal{V}; w^*[\mathcal{X}_A, \mathcal{X}_P])$$

Architecture space Hyperparameter space

DeepHyper: Scalable AutoML package

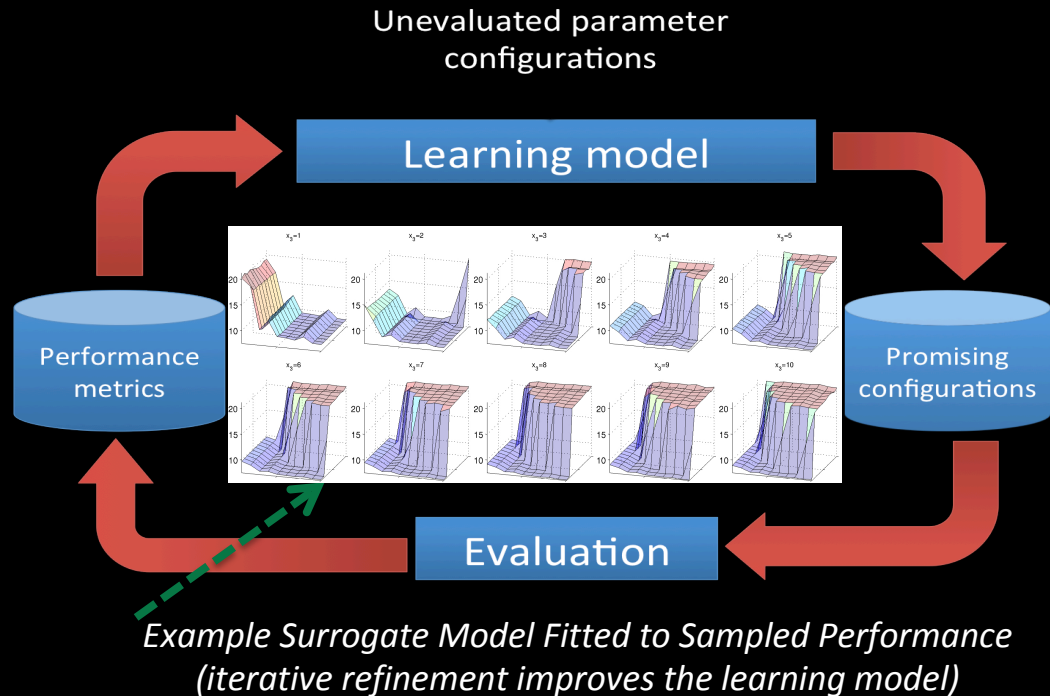


<https://github.com/deephyper/deephyper>

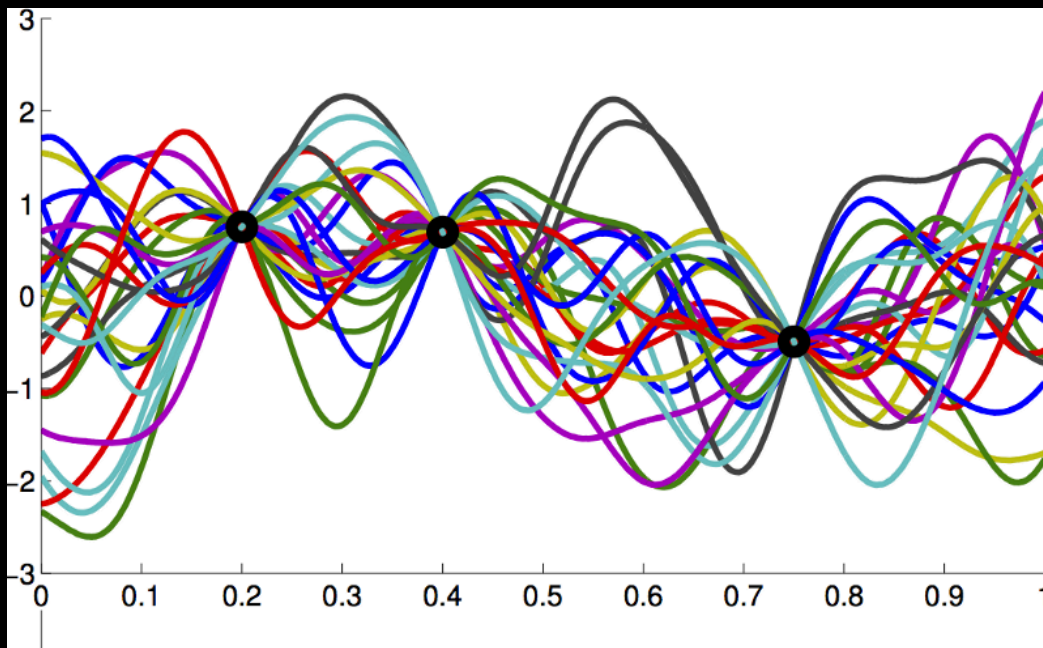
AMBS: Asynchronous model-based search

– Framework:

- Initialization phase
 - Random or Latin hypercube sampling
- Iterative phase
 - Fit model
 - Sample using the model



Bayesian optimization



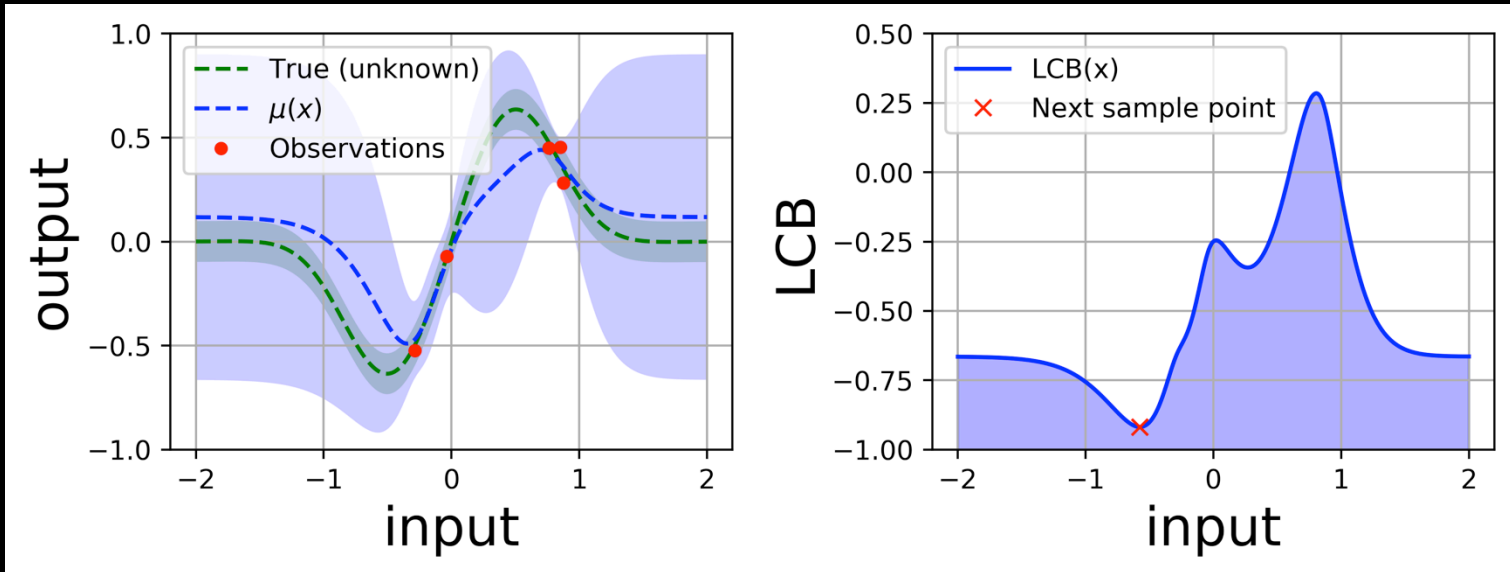
functions

- Usual Gaussian process regression cannot handle nonordinal space natively
- Appropriate methods: random forest, extra tree regressor, Bayesian NN
- We use **Random Forest**

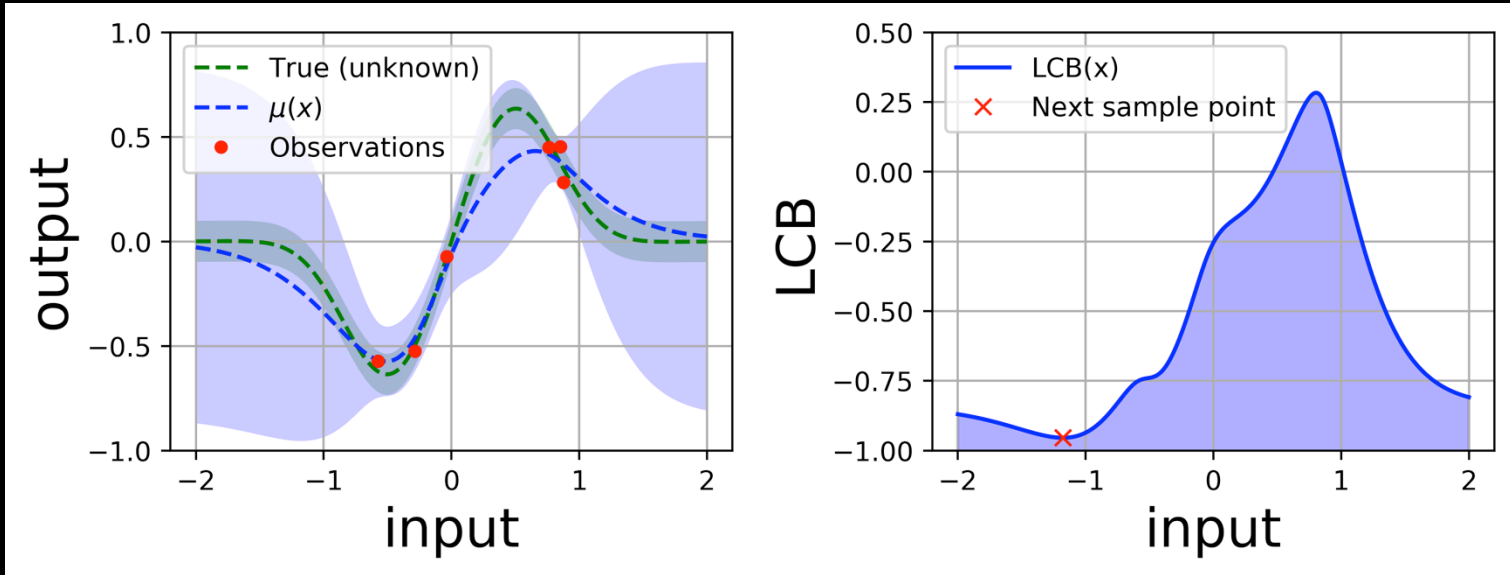
Bayesian optimization

$$LCB(x, \beta) = \mu(x) - \beta \times \sigma(x)$$

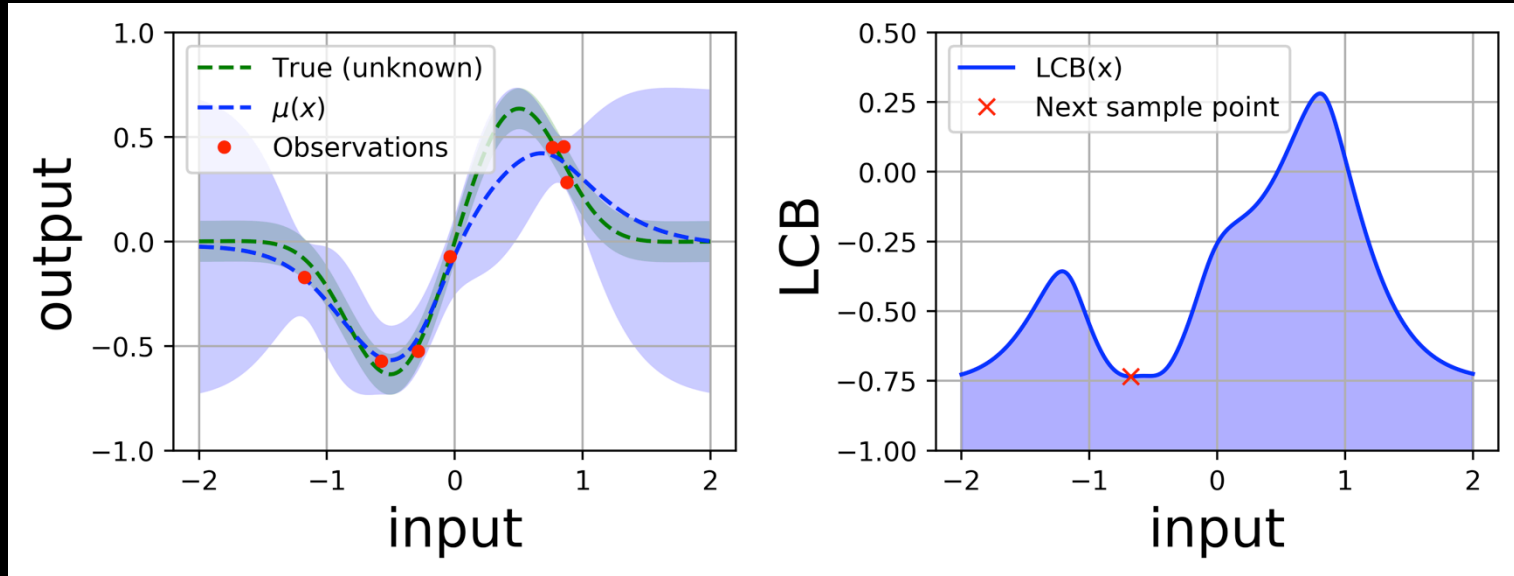
1.96



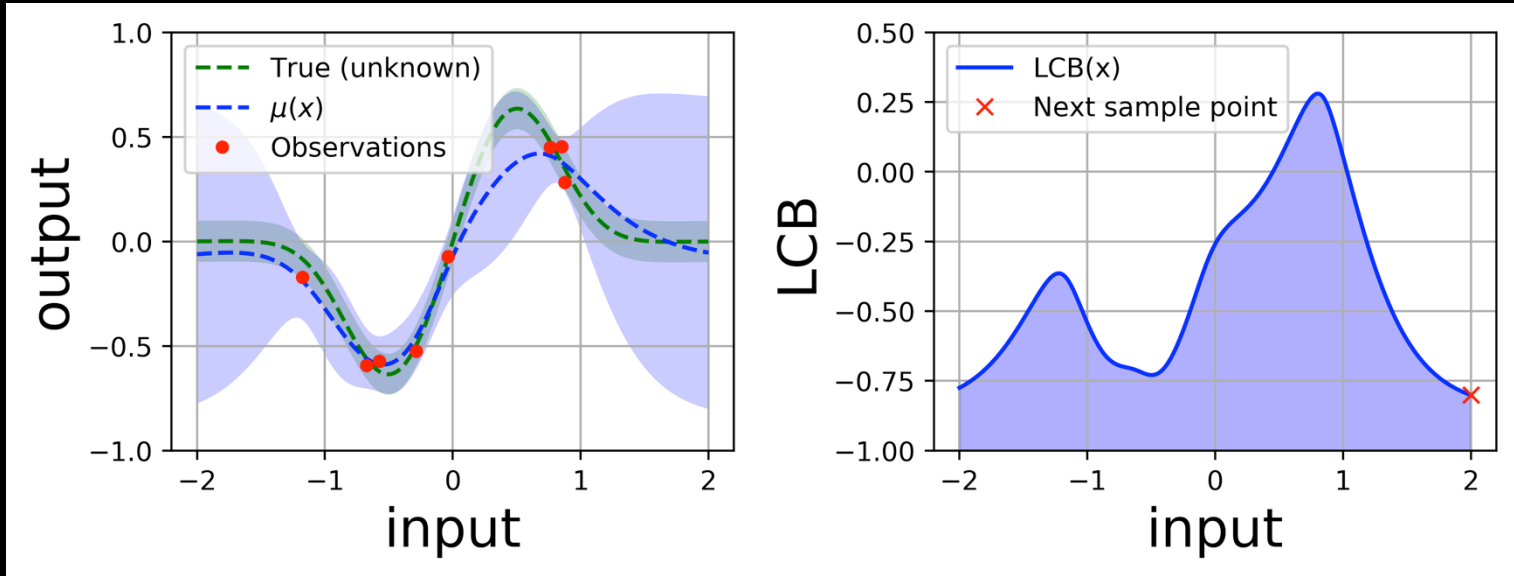
Bayesian optimization



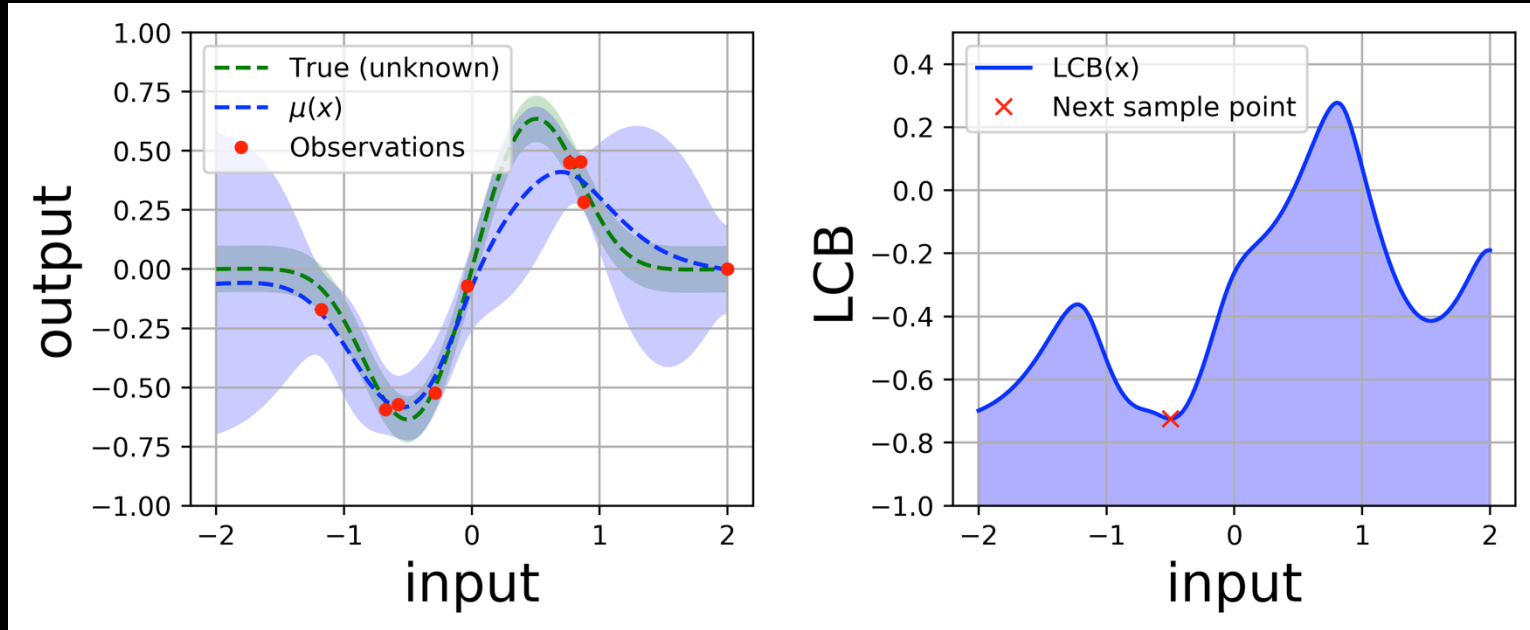
Bayesian optimization



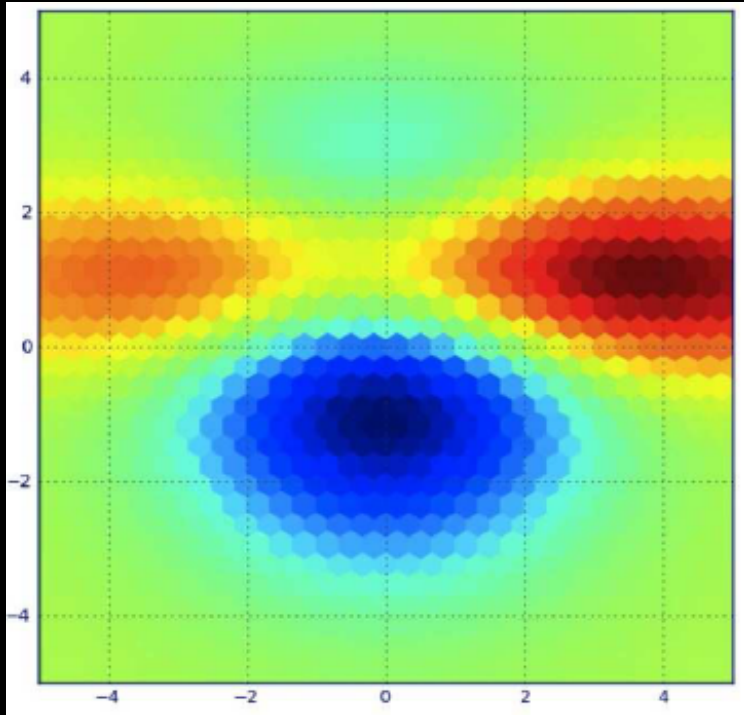
Bayesian optimization



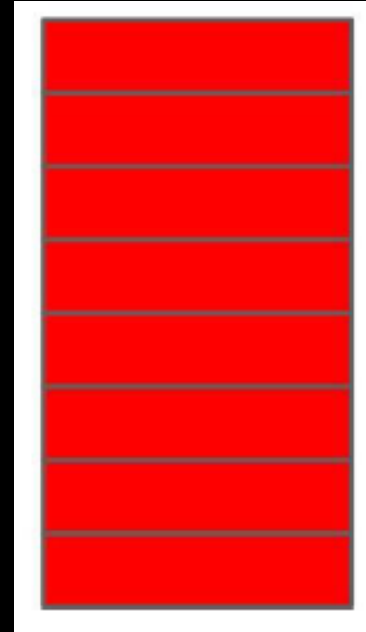
Bayesian optimization



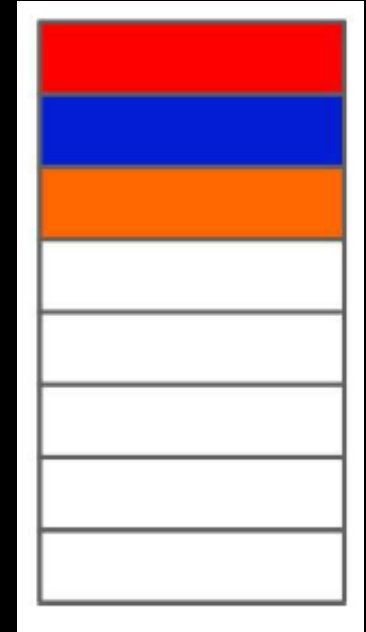
Multipoint asynchronous sampling



Naive



Conditioned



Constant liar scheme for asynchronous update

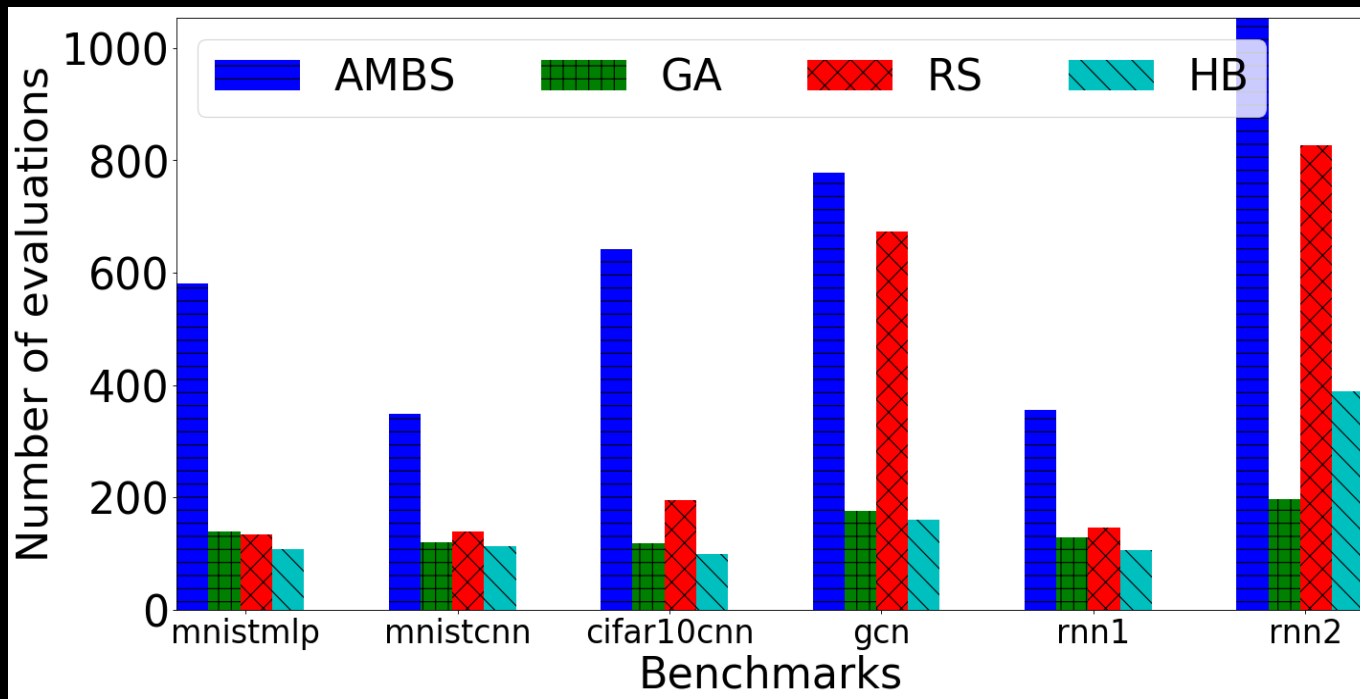
$$LCB_k(x, \beta) = \mu(x) - \beta \times \sigma(x)$$

$$f(\hat{x}_{LCB_k}) = \mu(\hat{x}_{LCB_k})$$

$$LCB_{k+1}(x, \beta) = \mu'(x) - \beta \times \sigma'(x)$$

$$f(\hat{x}_{LCB_{k+1}}) = \mu'(\hat{x}_{LCB_{k+1}})$$

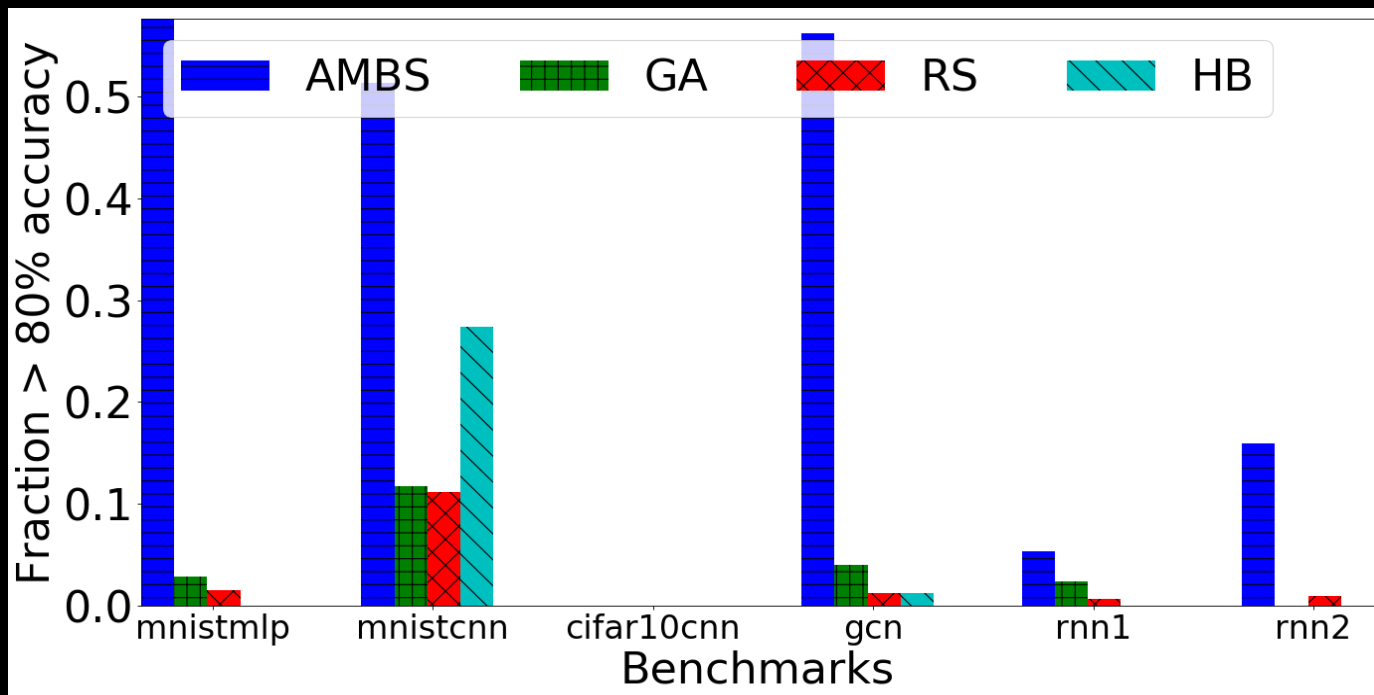
Comparison of search methods



Target platform: Theta@ALCF (128 KNL nodes)

Stopping criterion: 2 hours

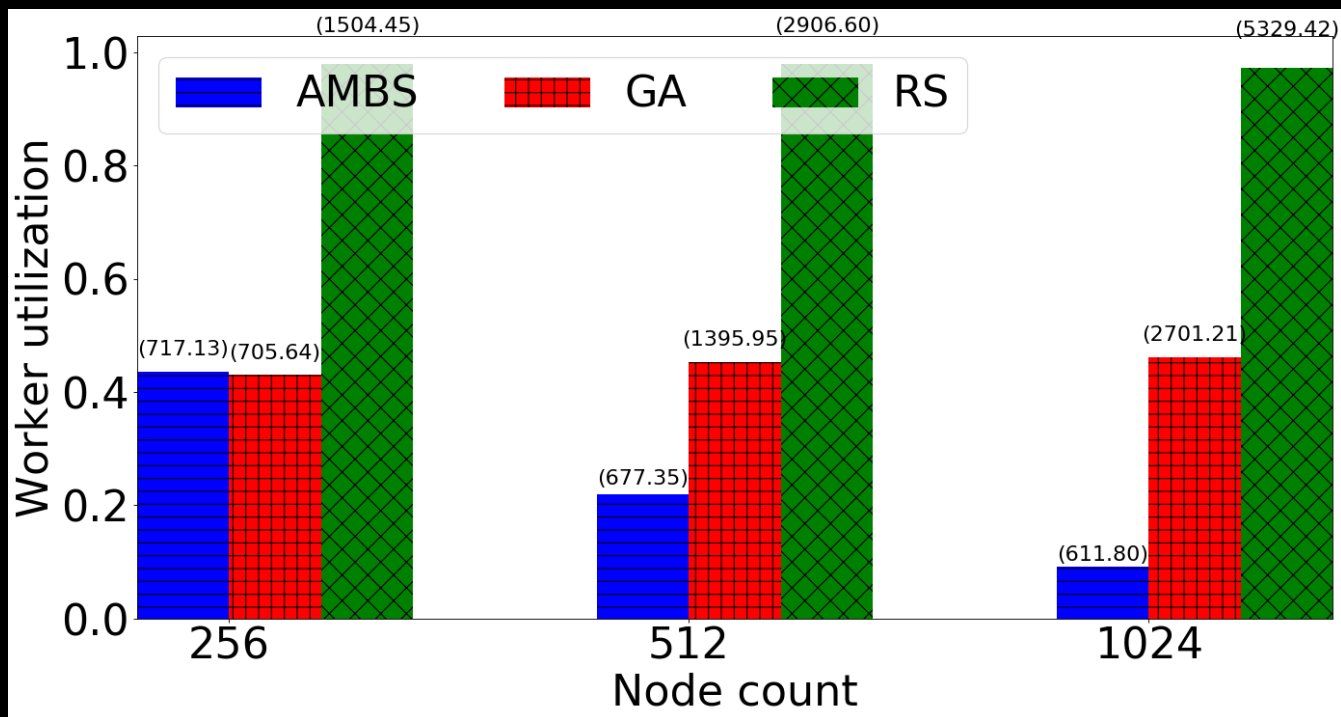
Comparison of search methods



Target platform: Theta@ALCF (128 KNL nodes)

Stopping criterion: 2 hours

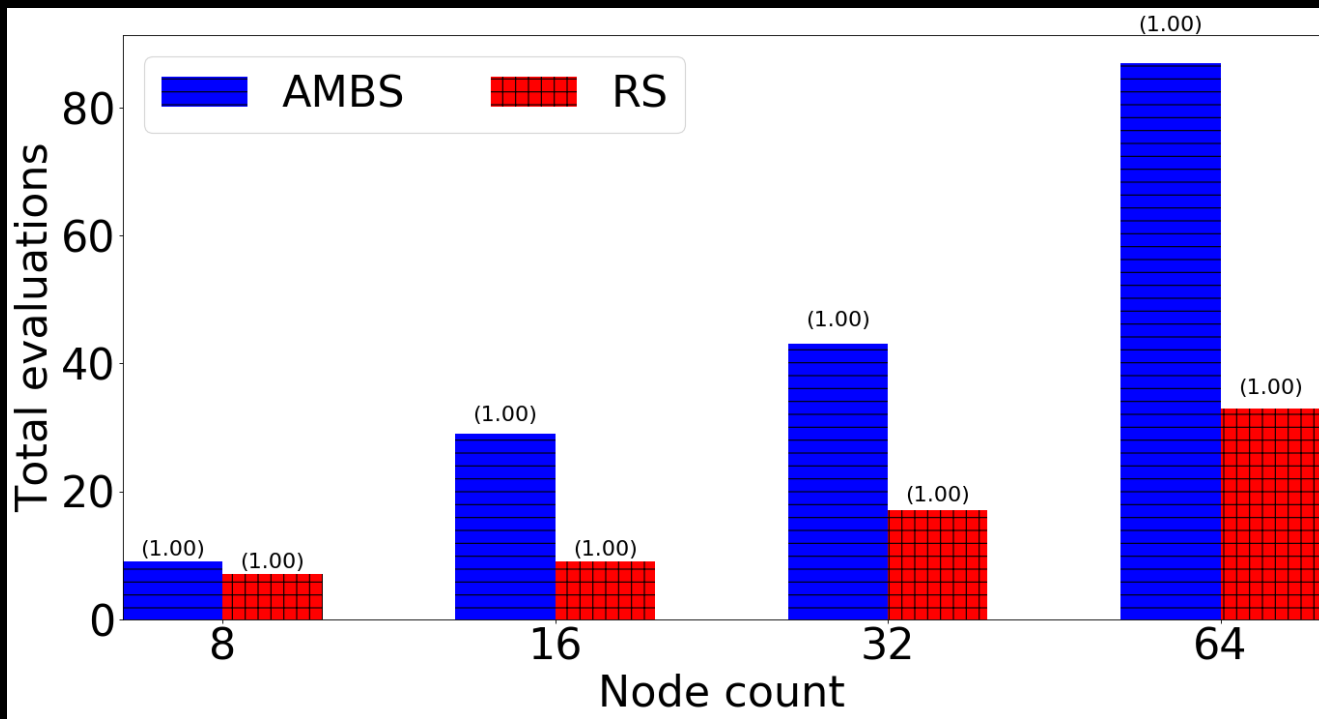
Scaling search methods



Target platform: Theta@ALCF (128 KNL nodes)

Benchmark: rnn2; Stopping criterion: 2 hours

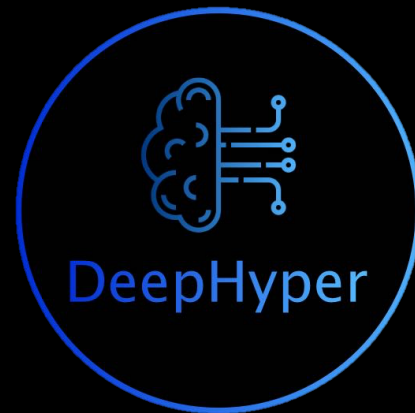
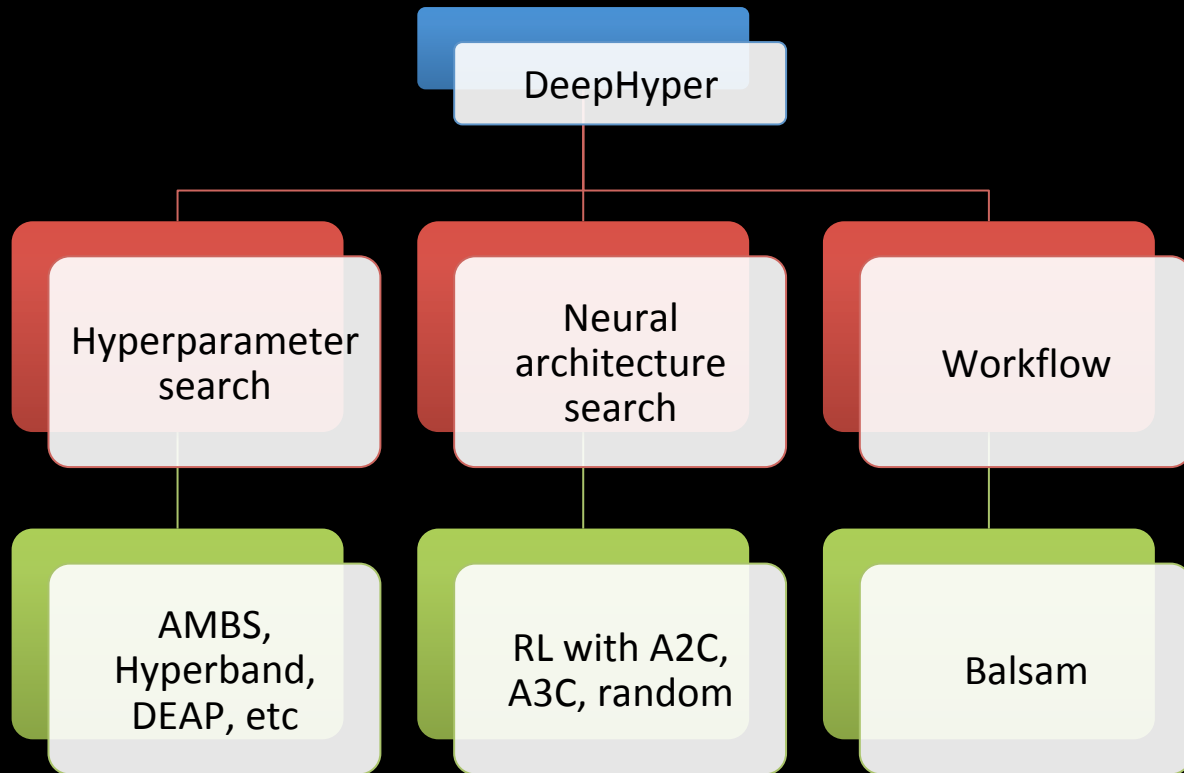
AMBS vs RS



Target platform: Cooley (64 nodes Haswell + NVIDIA Tesla K80)

Benchmark: cifar10cnn; Stopping criterion: 1 hour

DeepHyper: Scalable AutoML package



<https://github.com/deephyper/deephyper>

Acknowledgements



DOE Early Career Research Program, ASCR

CANDLE ECP Project



Argonne Leadership Computing Facility

Laboratory Directed Research and Development (LDRD)

DeepHyper

