Final Presentation

Reducing Maximization Bias and Risk in Hyperparameter Optimization for Reinforcement Learning and Learning-Based Control

Master Thesis

Tsung Yuan Tseng [1,2]

Supervisors: Dr. Siqi Zhou[2], Lukas Brunke[2], Alexander von Rohr[1], Prof. Dr. Angela Schoellig[2], and Prof. Dr. Sebastian Trimpe[1]

[1] Institute for Data Science in Mechanical Engineering, RWTH Aachen University
[2] Learning Systems and Robotics Lab, Technical University of Munich & University of Toronto

October 31, 2023

Introduction and Motivation

Introduction and Motivation

- Most algorithms have hyperparameters h.

Introduction and Motivation

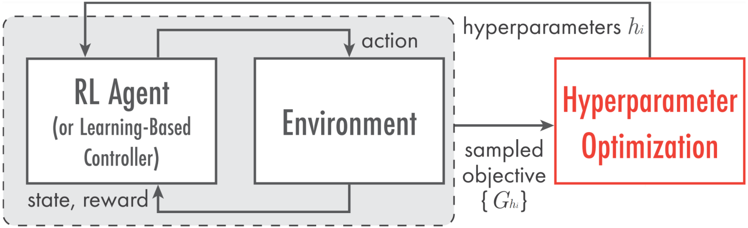
- Most algorithms have hyperparameters h.
- Reinforcement learning and learning-based control.

Introduction and Motivation

- Most algorithms have hyperparameters h.
- Reinforcement learning and learning-based control.
  - Optimize h that maximizes algorithm performance G under limited computational budget.

Introduction and Motivation

- Most algorithms have hyperparameters h.
- Reinforcement learning and learning-based control.
  - Optimize h that maximizes algorithm performance G under limited computational budget.

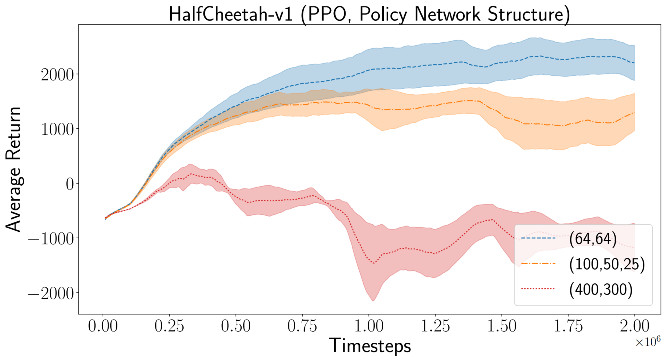


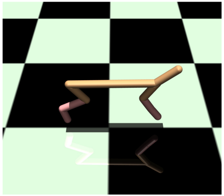
Decision-Making Algorithms

- Hyperparameters have a great impact. [a]

Decision-Making Algorithms

- Hyperparameters have a great impact. [a]





[a] (Bischl et al. 2021; Henderson et al. 2018)

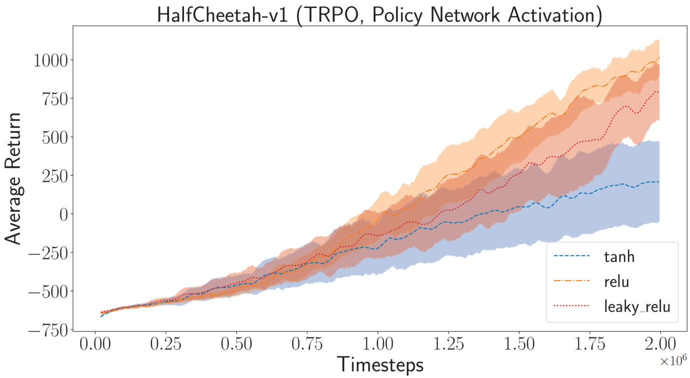
Decision-Making Algorithms

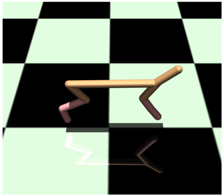
- Robust hyperparameters are essential.
  - Ensure consistent algorithm performance.

[1] (Henderson et al. 2018)

Decision-Making Algorithms

- Robust hyperparameters are essential.
  - Ensure consistent algorithm performance.





1

[1](Henderson et al. 2018)

Related Works

- Hyperparameter optimization (HPO)

[2](Bergstra and Bengio 2012)
[3](Snoek et al. 2012)
[4](Eimer et al. 2023; Zhang et al. 2021; Eggensperger et al. 2019)

Related Works

- Hyperparameter optimization (HPO)
  - Manual HPO?

[2](Bergstra and Bengio 2012)
[3](Snoek et al. 2012)
[4](Eimer et al. 2023; Zhang et al. 2021; Eggensperger et al. 2019)

Related Works

- Hyperparameter optimization (HPO)
  - Manual HPO?
- Automated tuning

[2] (Bergstra and Bengio 2012)
[3] (Snoek et al. 2012)
[4] (Eimer et al. 2023; Zhang et al. 2021; Eggensperger et al. 2019)

Related Works

- Hyperparameter optimization (HPO)
  - Manual HPO?
- Automated tuning
  - HPO algorithms such as random search [2] and Bayesian optimization (BO). [3]

[2] (Bergstra and Bengio 2012)
[3] (Snoek et al. 2012)
[4] (Eimer et al. 2023; Zhang et al. 2021; Eggensperger et al. 2019)

Related Works

- Hyperparameter optimization (HPO)
  - Manual HPO?
- Automated tuning
  - HPO algorithms such as random search [2] and Bayesian optimization (BO). [3]
- Several advantages: [4]

[2] (Bergstra and Bengio 2012)
[3] (Snoek et al. 2012)
[4] (Eimer et al. 2023; Zhang et al. 2021; Eggensperger et al. 2019)

Related Works

- Hyperparameter optimization (HPO)
  - Manual HPO?
- Automated tuning
  - HPO algorithms such as random search [2] and Bayesian optimization (BO). [3]
- Several advantages: [4]
  - More configurations evaluated

[2] (Bergstra and Bengio 2012)
[3] (Snoek et al. 2012)
[4] (Eimer et al. 2023; Zhang et al. 2021; Eggensperger et al. 2019)

Related Works

- Hyperparameter optimization (HPO)
  - Manual HPO?
- Automated tuning
  - HPO algorithms such as random search [2] and Bayesian optimization (BO). [3]
- Several advantages: [4]
  - More configurations evaluated
  - Performance

[2] (Bergstra and Bengio 2012)
[3] (Snoek et al. 2012)
[4] (Eimer et al. 2023; Zhang et al. 2021; Eggensperger et al. 2019)

Related Works

- Hyperparameter optimization (HPO)
  - Manual HPO?
- Automated tuning
  - HPO algorithms such as random search [2] and Bayesian optimization (BO). [3]
- Several advantages: [4]
  - More configurations evaluated
  - Performance
  - Reproducibility

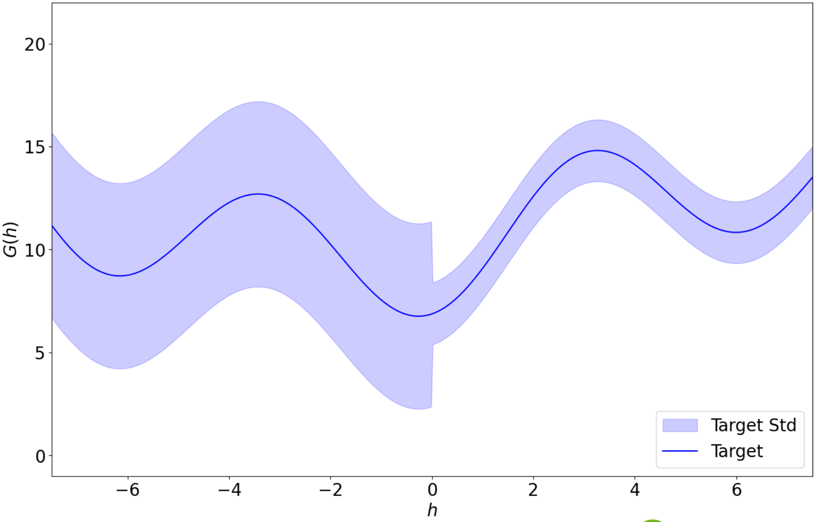
[2] (Bergstra and Bengio 2012)
[3] (Snoek et al. 2012)
[4] (Eimer et al. 2023; Zhang et al. 2021; Eggensperger et al. 2019)

Related Works

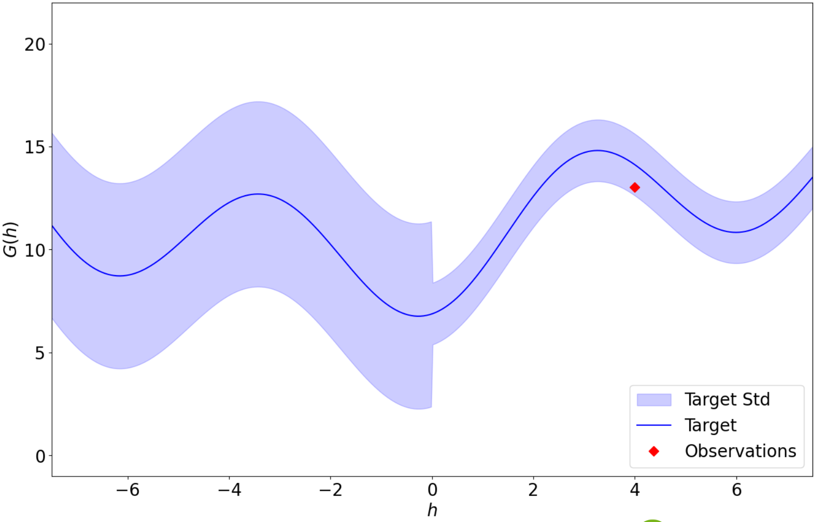
- Hyperparameter optimization (HPO)
  - Manual HPO?
- Automated tuning
  - HPO algorithms such as random search [2] and Bayesian optimization (BO). [3]
- Several advantages: [4]
  - More configurations evaluated
  - Performance
  - Reproducibility
  - Less human-time

[2] (Bergstra and Bengio 2012)
[3] (Snoek et al. 2012)
[4] (Eimer et al. 2023; Zhang et al. 2021; Eggensperger et al. 2019)

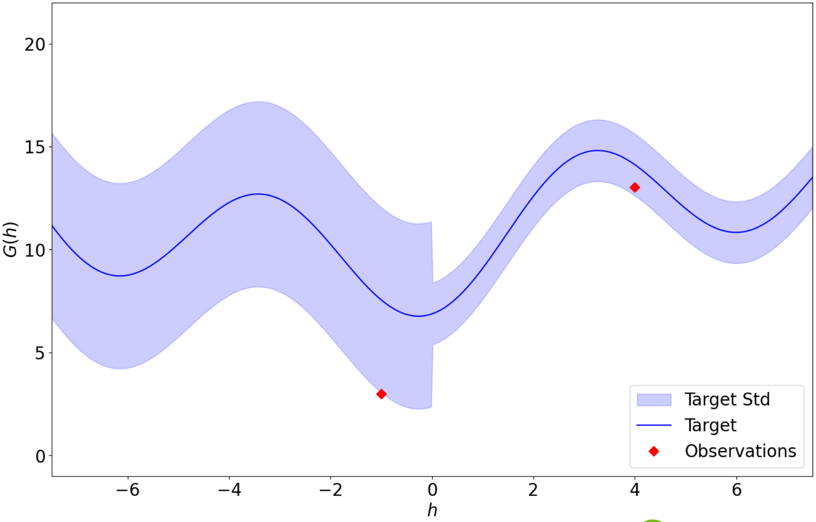
Random Search



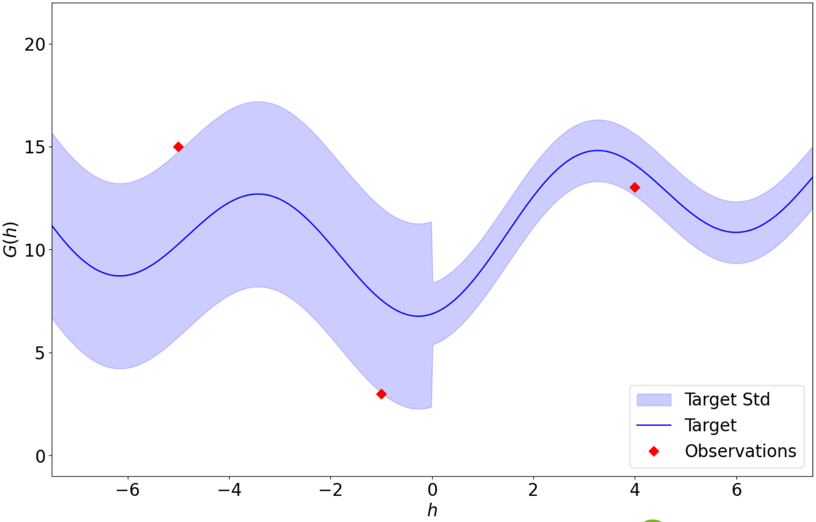
Random Search



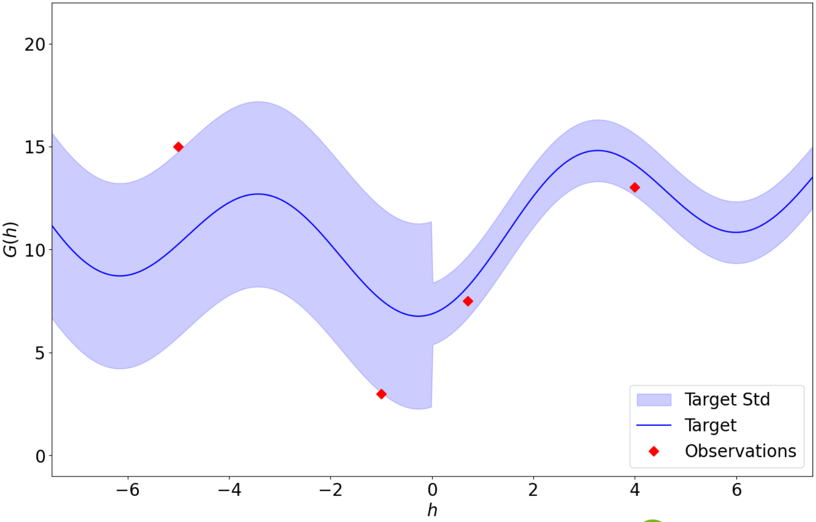
Random Search



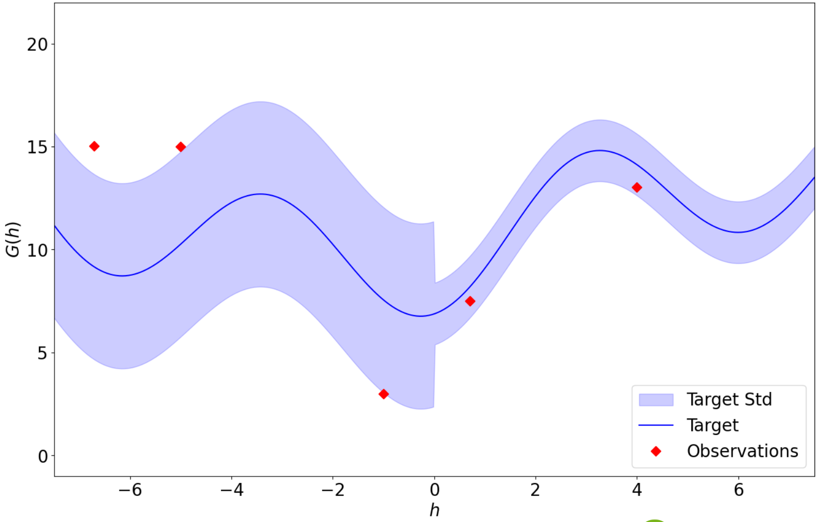
Random Search



Random Search



Random Search



Current Challenges: Maximization bias

- Maximization bias in HPO (synonyms are overconfidence, overtuning, and overfitting). [5]

[5] Hutter et al. 2007; Patterson et al. 2023

Problem Statement

Problem Formulation

- Evaluation metrics G;
- Hyperparameter $\boldsymbol{h} \in \mathcal{H}$;

$$\boldsymbol{h}^* = \arg\max_{\boldsymbol{h} \in \mathcal{H}} \rho\left[G|\boldsymbol{h}\right],$$

where ρ is a risk functional.

Goal of the Thesis

Problem Statement

Problem Formulation

- Evaluation metrics G;
- Hyperparameter $\boldsymbol{h} \in \mathcal{H}$;

$$\boldsymbol{h}^* = \arg\max_{\boldsymbol{h} \in \mathcal{H}} \rho \left[G | \boldsymbol{h} \right],$$

where ρ is a risk functional.

Goal of the Thesis

Extract robust $\boldsymbol{h}^*$ by observing data sequences $\{(\boldsymbol{h}_i, \hat{\rho}[G|\boldsymbol{h}_i])\}_{i=1}^{M}$.

Analysis on Maximization Bias

- Risk functional ρ: expectations $\mathbb{E}$.

Analysis on Maximization Bias

- Risk functional ρ: expectations $\mathbb{E}$.
- HPO

$$\boldsymbol{h}^* = \arg\max_{\boldsymbol{h}\in\mathcal{H}} \mathbb{E}\left[G|\boldsymbol{h}\right].$$

Analysis on Maximization Bias

- Risk functional ρ: expectations $\mathbb{E}$.
- HPO

$$\boldsymbol{h}^* = \underset{\boldsymbol{h}\in\mathcal{H}}{\arg\max}\, \mathbb{E}\left[G|\boldsymbol{h}\right].$$

- $\mathbb{E}[G|\boldsymbol{h}]$ (shorthand $\mathbb{E}[G_{\boldsymbol{h}}]$) $\approx \hat{\mu}_{\boldsymbol{h}}$

Analysis on Maximization Bias

- Risk functional ρ: expectations $\mathbb{E}$.
- HPO

$$\boldsymbol{h}^* = \arg\max_{\boldsymbol{h}\in\mathcal{H}} \mathbb{E}\left[G|\boldsymbol{h}\right].$$

- $\mathbb{E}[G|\boldsymbol{h}]$ (shorthand $\mathbb{E}[G_{\boldsymbol{h}}]$) $\approx \hat{\mu}_{\boldsymbol{h}}$
- Observed data $\{(\boldsymbol{h}_i, \hat{\mu}_{\boldsymbol{h}_i})\}_{i=1}^{M}$

Analysis on Maximization Bias

- Risk functional ρ: expectations $\mathbb{E}$.
- HPO

$$\boldsymbol{h}^* = \arg\max_{\boldsymbol{h}\in\mathcal{H}} \mathbb{E}\left[G|\boldsymbol{h}\right].$$

- $\mathbb{E}[G|\boldsymbol{h}]$ (shorthand $\mathbb{E}[G_{\boldsymbol{h}}]$) $\approx \hat{\mu}_{\boldsymbol{h}}$
- Observed data $\{(\boldsymbol{h}_i, \hat{\mu}_{\boldsymbol{h}_i})\}_{i=1}^{M}$
- Optimization is done by

$$\boldsymbol{h}^* = \arg\max_{\boldsymbol{h}_i\in\mathbb{H}} \hat{\mu}_{\boldsymbol{h}_i},$$

where $\mathbb{H} = \{\boldsymbol{h}_i\}_{i=1}^{M}$.

Analysis on Maximization Bias

- Risk functional ρ: expectations $\mathbb{E}$.
- HPO

$$\boldsymbol{h}^* = \arg\max_{\boldsymbol{h}\in\mathcal{H}} \mathbb{E}\left[G|\boldsymbol{h}\right].$$

- $\mathbb{E}[G|\boldsymbol{h}]$ (shorthand $\mathbb{E}[G_{\boldsymbol{h}}]$) $\approx \hat{\mu}_{\boldsymbol{h}}$
- Observed data $\{(\boldsymbol{h}_i, \hat{\mu}_{\boldsymbol{h}_i})\}_{i=1}^{M}$
- Optimization is done by

$$\boldsymbol{h}^* = \arg\max_{\boldsymbol{h}_i\in\mathbb{H}} \hat{\mu}_{\boldsymbol{h}_i},$$

where $\mathbb{H} = \{\boldsymbol{h}_i\}_{i=1}^{M}$.

Analysis on Maximization Bias

- The value of $\max\{\hat{\mu}_{\boldsymbol{h}_i}\}_{i=1}^{M}$ has a great impact on final selection of $\boldsymbol{h}^*$
- $\max\{\hat{\mu}_{\boldsymbol{h}_i}\}_{i=1}^{M} \stackrel{?}{=} \max\{\mathbb{E}[G_{\boldsymbol{h}_i}]\}_{i=1}^{M}$

Analysis on Maximization Bias

Lemma: Lower Bound of Maximization Bias

Consider

- Sampled set of hyperparameters $\mathbb{H} = \{\boldsymbol{h}_i\}_{i=1}^{M}$

Use

Then

Analysis on Maximization Bias

Lemma: Lower Bound of Maximization Bias

Consider

- Sampled set of hyperparameters $\mathbb{H} = \{\boldsymbol{h}_i\}_{i=1}^{M}$
- Set of random variables $\mathcal{G} = \{G_{\boldsymbol{h}_i}\}_{i=1}^{M}$

Use

Then

Analysis on Maximization Bias

Lemma: Lower Bound of Maximization Bias

Consider

- Sampled set of hyperparameters $\mathbb{H} = \{\boldsymbol{h}_i\}_{i=1}^{M}$
- Set of random variables $\mathcal{G} = \{G_{\boldsymbol{h}_i}\}_{i=1}^{M}$
- Set of true means $\mathbb{E}[\mathcal{G}] = \{\mu_{\boldsymbol{h}_i}\}_{i=1}^{M}$.

Use

Then

Analysis on Maximization Bias

Lemma: Lower Bound of Maximization Bias

Consider

- Sampled set of hyperparameters $\mathbb{H} = \{\boldsymbol{h}_i\}_{i=1}^M$
- Set of random variables $\mathcal{G} = \{G_{\boldsymbol{h}_i}\}_{i=1}^M$
- Set of true means $\mathbb{E}[\mathcal{G}] = \{\mu_{\boldsymbol{h}_i}\}_{i=1}^M$.

Use

- Set of unbiased estimators $\hat{\mathcal{G}} = \{\hat{\mu}_{\boldsymbol{h}_i}\}_{i=1}^M$

Then

Analysis on Maximization Bias

Lemma: Lower Bound of Maximization Bias

Consider

- Sampled set of hyperparameters $\mathbb{H} = \{\boldsymbol{h}_i\}_{i=1}^{M}$
- Set of random variables $\mathcal{G} = \{G_{\boldsymbol{h}_i}\}_{i=1}^{M}$
- Set of true means $\mathbb{E}[\mathcal{G}] = \{\mu_{\boldsymbol{h}_i}\}_{i=1}^{M}$.

Use

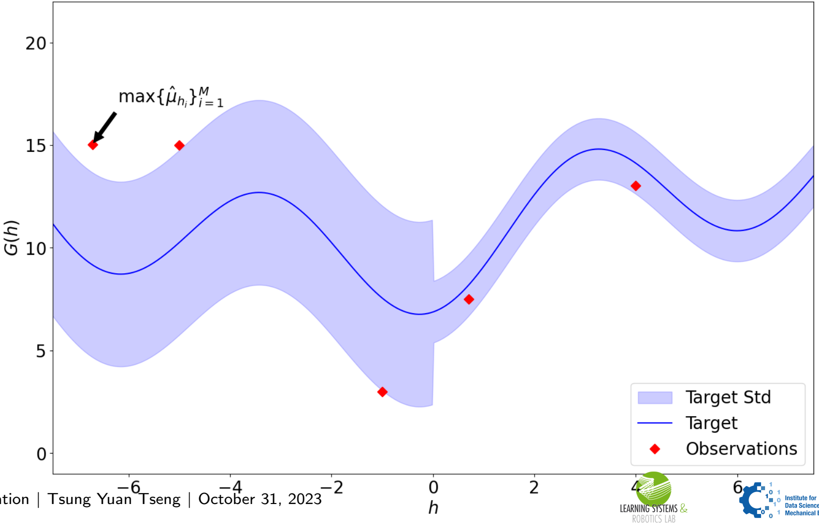
- Set of unbiased estimators $\hat{\mathcal{G}} = \{\hat{\mu}_{\boldsymbol{h}_i}\}_{i=1}^{M}$

Then

- $\mathbb{E}[\max_{\boldsymbol{h}_i \in \mathbb{H}} \hat{\mathcal{G}}] - \max_{\boldsymbol{h}_i \in \mathbb{H}} \mathbb{E}[\mathcal{G}] \geq 0$

Analysis on Maximization Bias

- Uncertainties in $\hat{\mathcal{G}} = \{\hat{\mu}_{h_i}\}_{i=1}^{M}$ leads to sub-optimal decisions.



Methods: Adaptive Variance Reduction

- Limitation

Methods: Adaptive Variance Reduction

- Limitation
  - Reduce uncertainties $\rightarrow$ Increase samples.
  - Insufficient samples $\rightarrow$ High uncertainties.

Methods: Adaptive Variance Reduction

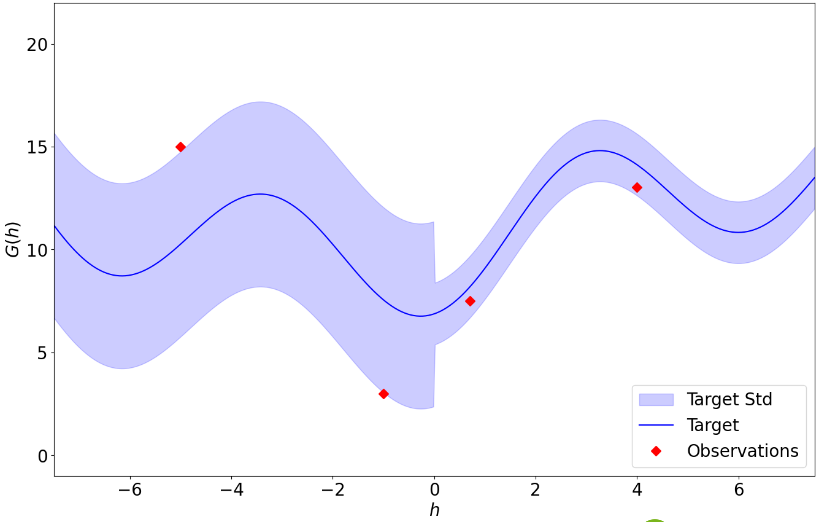
- Limitation
  - Reduce uncertainties $\rightarrow$ Increase samples.
  - Insufficient samples $\rightarrow$ High uncertainties.
- We propose an adaptive sampling scheme termed "adaptive multiple runs algorithm" (AMRA).
  - Increase samples (runs) when necessary.

Methods: Adaptive Variance Reduction

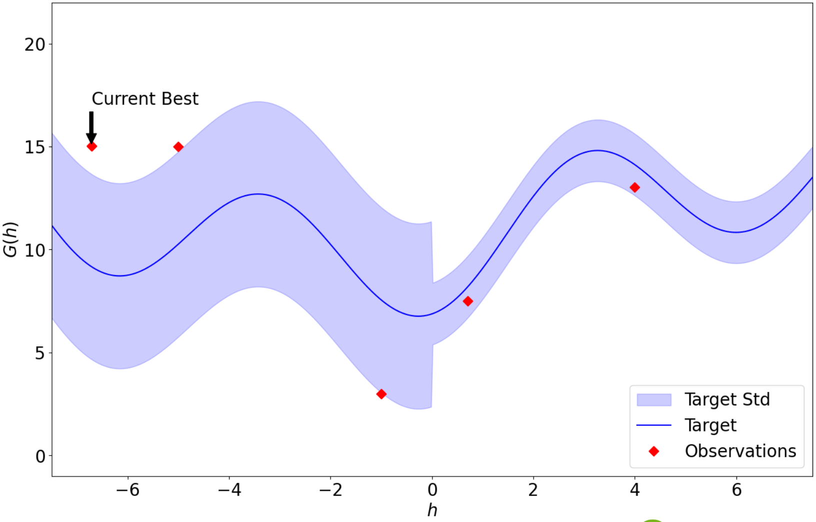
- Limitation
  - Reduce uncertainties $\rightarrow$ Increase samples.
  - Insufficient samples $\rightarrow$ High uncertainties.
- We propose an adaptive sampling scheme termed "adaptive multiple runs algorithm" (AMRA).
  - Increase samples (runs) when necessary.

A run means one realization of G

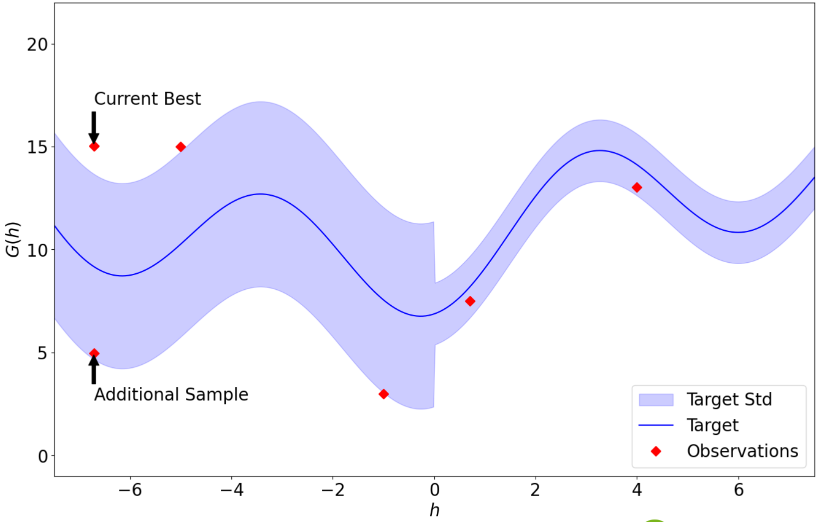
Methods: Adaptive Variance Reduction



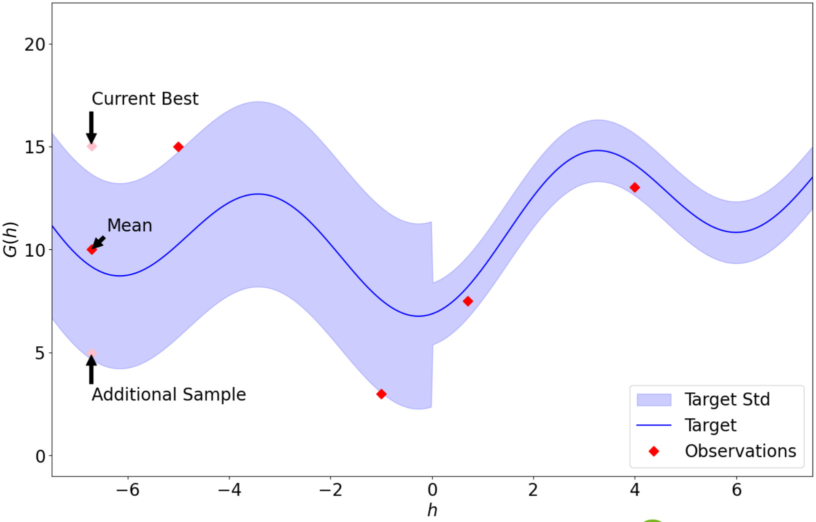
Methods: Adaptive Variance Reduction



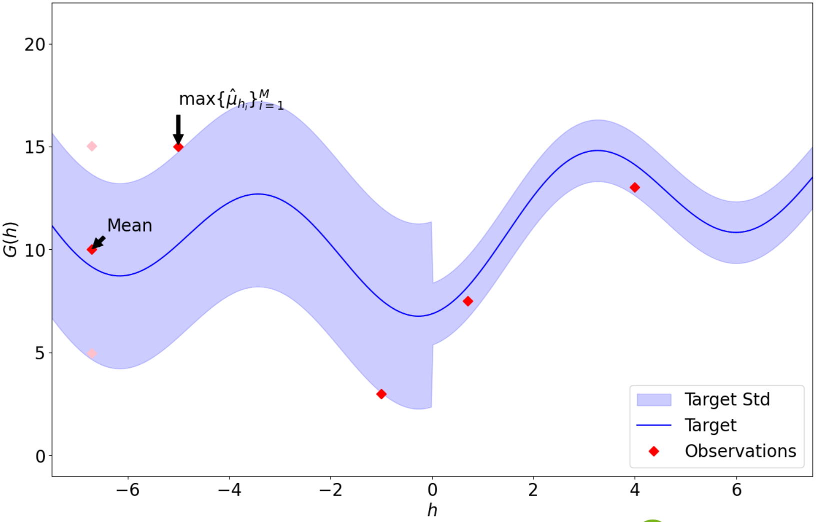
Methods: Adaptive Variance Reduction



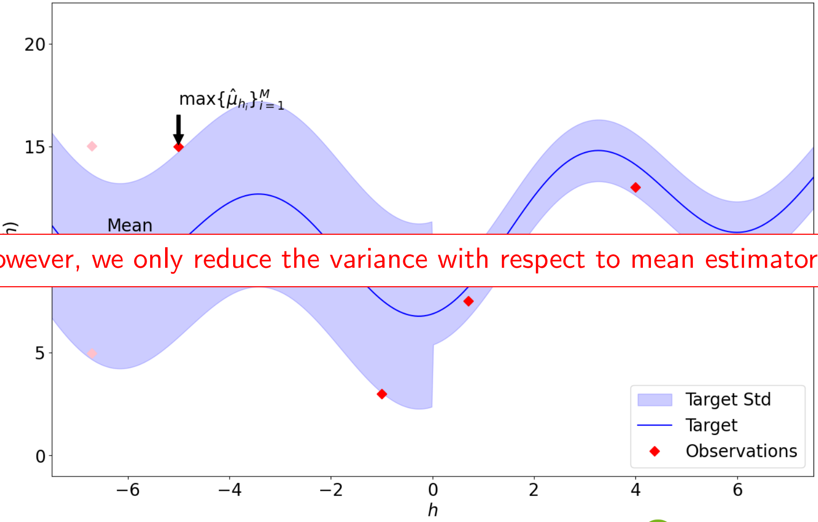
Methods: Adaptive Variance Reduction



Methods: Adaptive Variance Reduction



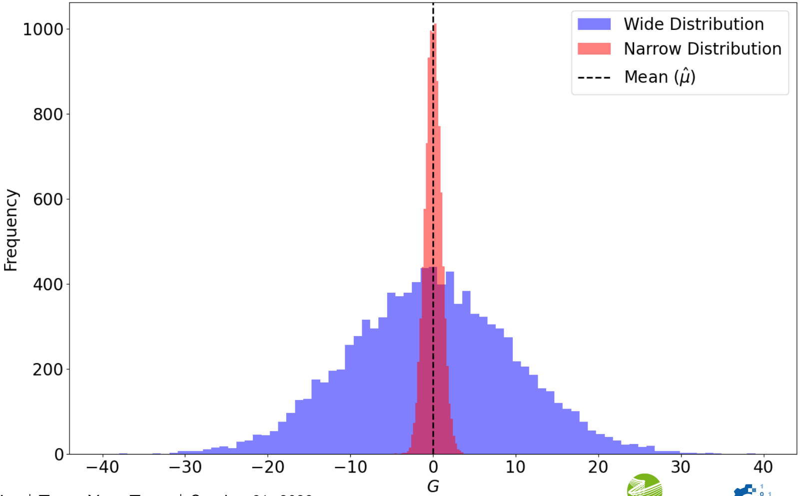
Methods: Adaptive Variance Reduction



However, we only reduce the variance with respect to mean estimators...

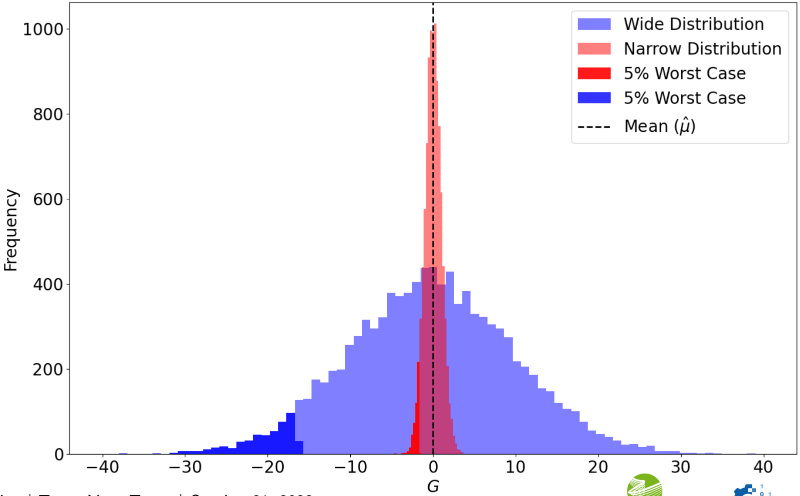
Methods: Risk in Tail Distributions

- Is the expectation a good risk measure (risk functional)?



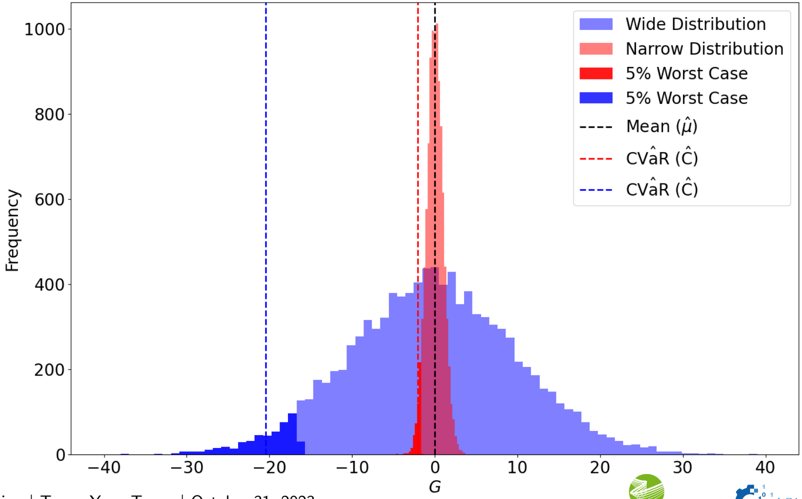
Methods: Risk in Tail Distributions

- Is the expectation a good risk measure (risk functional)?

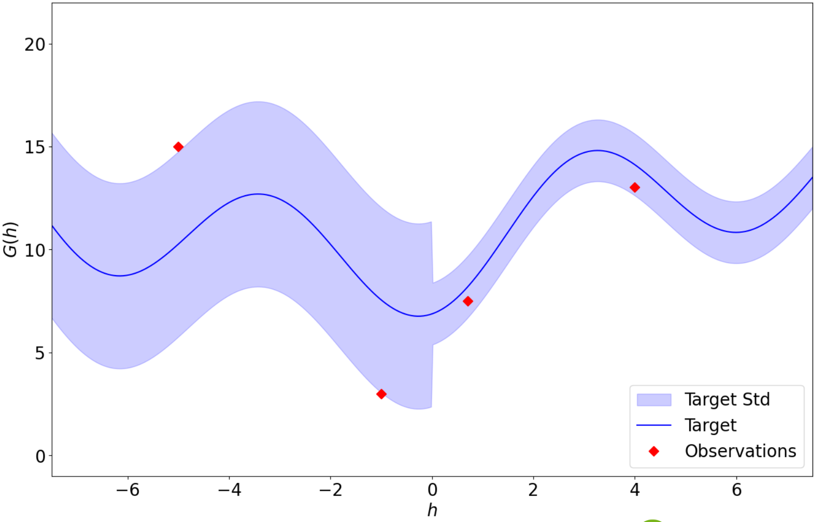


Methods: Risk in Tail Distributions

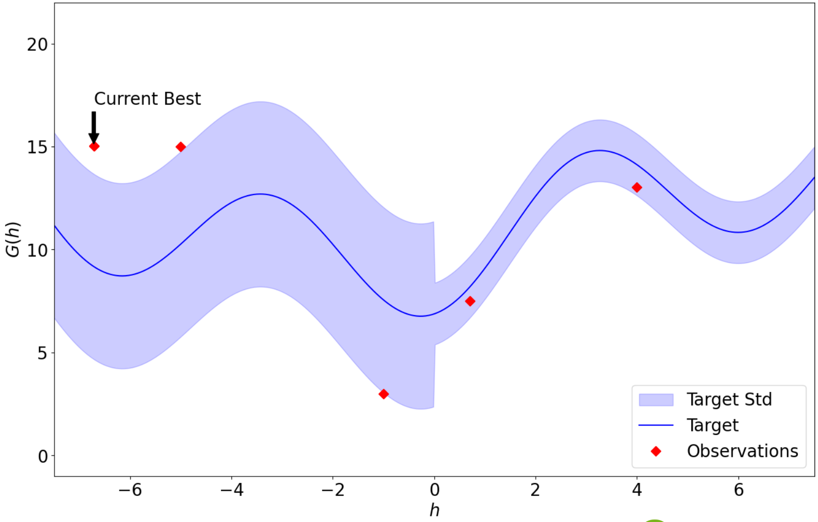
- Conditional Value at Risk (CVaR)



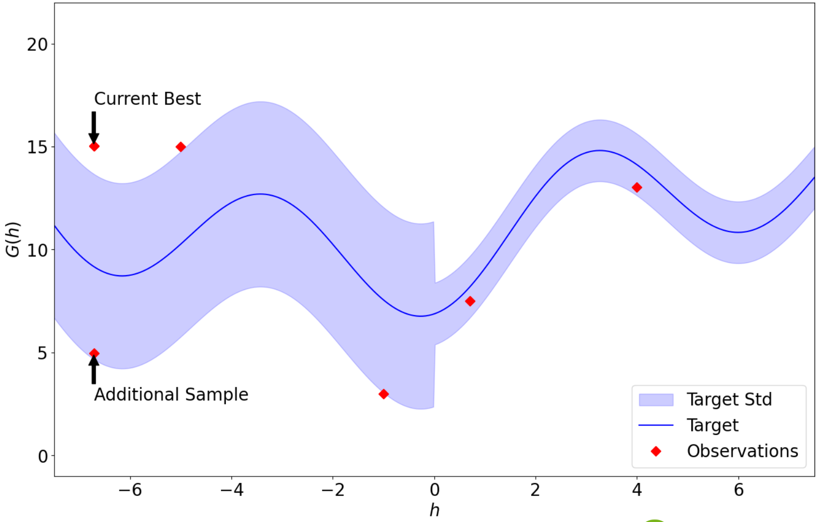
Methods: AMRA with CVaR



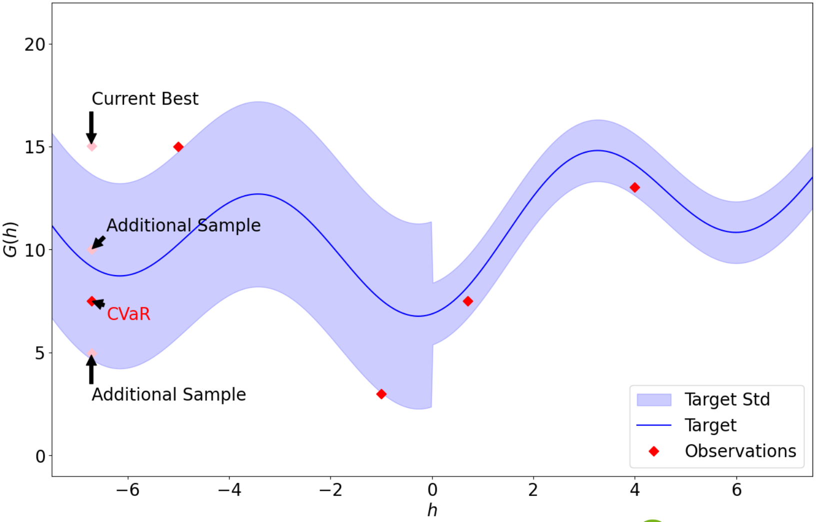
Methods: AMRA with CVaR



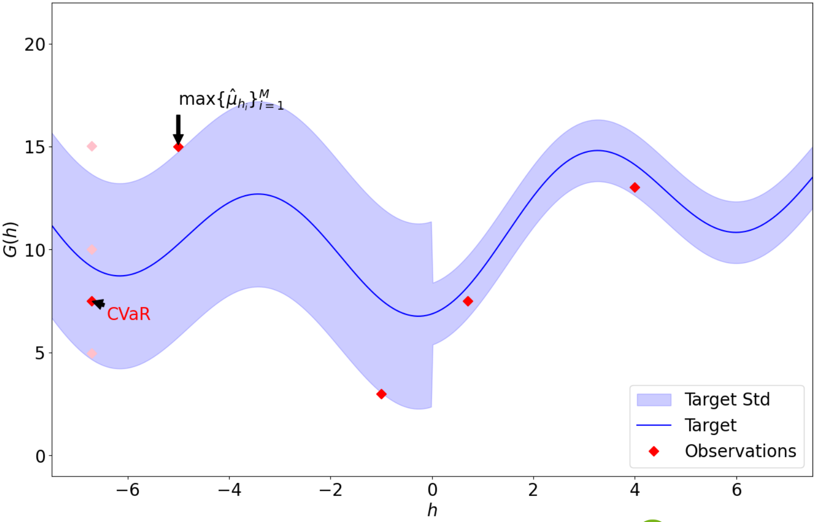
Methods: AMRA with CVaR



Methods: AMRA with CVaR



Methods: AMRA with CVaR



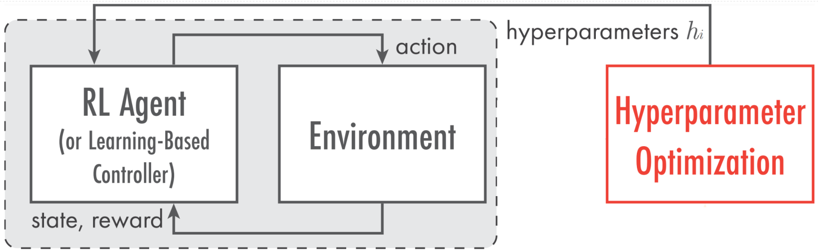
Methods: AMRA with CVaR

- Convergence criterion?

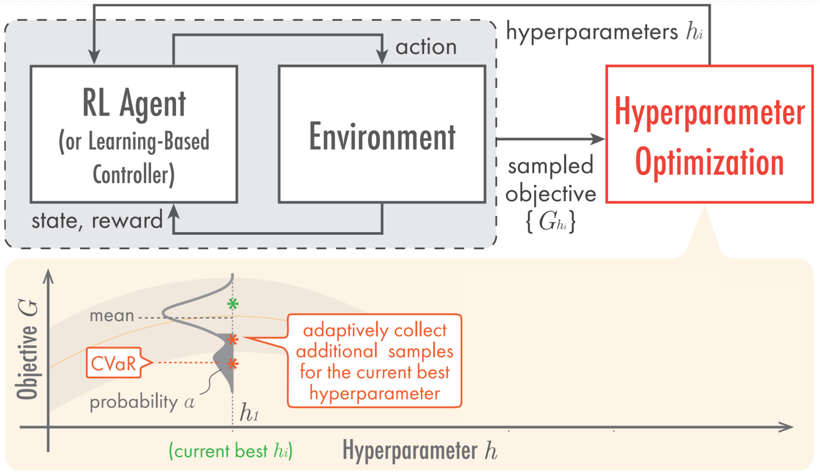
Methods: AMRA with CVaR

- Convergence criterion?
- $|\hat{\text{CVaR}}^{s}[G_{\boldsymbol{h}_i}] - \hat{\text{CVaR}}^{ss}[G_{\boldsymbol{h}_i}]| < \delta$.

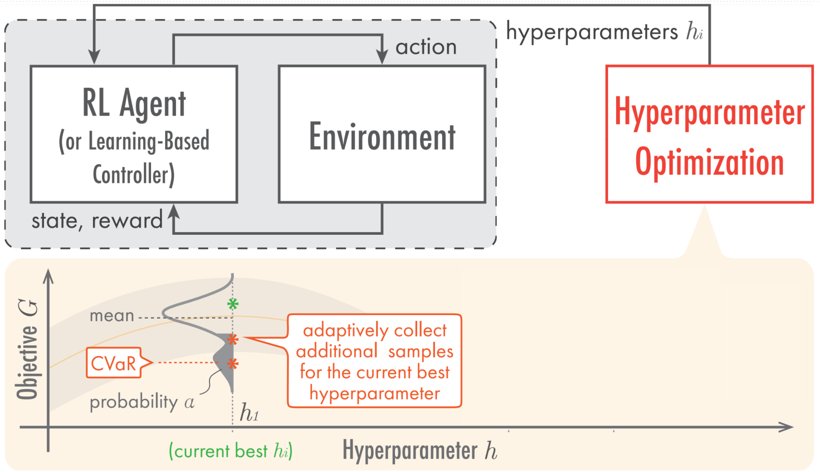
Methods: Overview



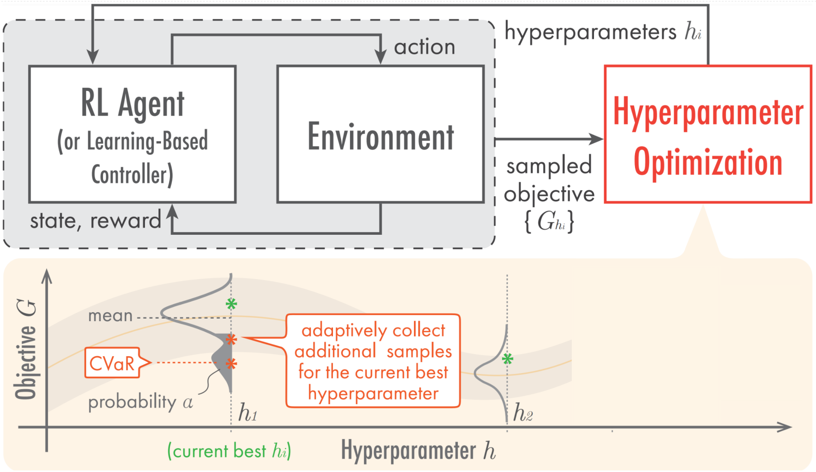
Methods: Overview



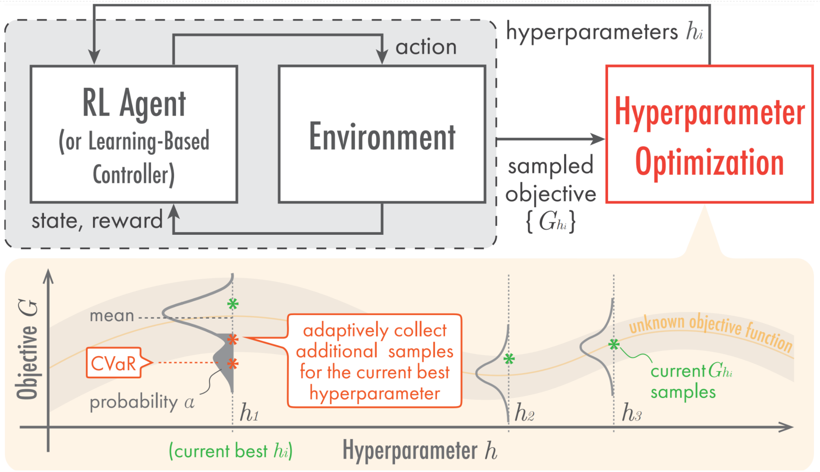
Methods: Overview



Methods: Overview



Methods: Overview



HPO Strategy Comparisons

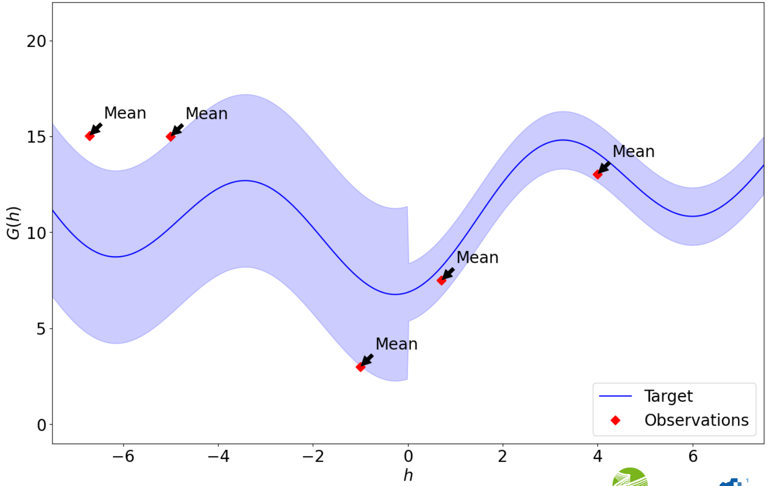
- Impact of the risk functional?

HPO Strategy Comparisons

- Impact of the risk functional?
- Impact of number of samples per $G(h)$?

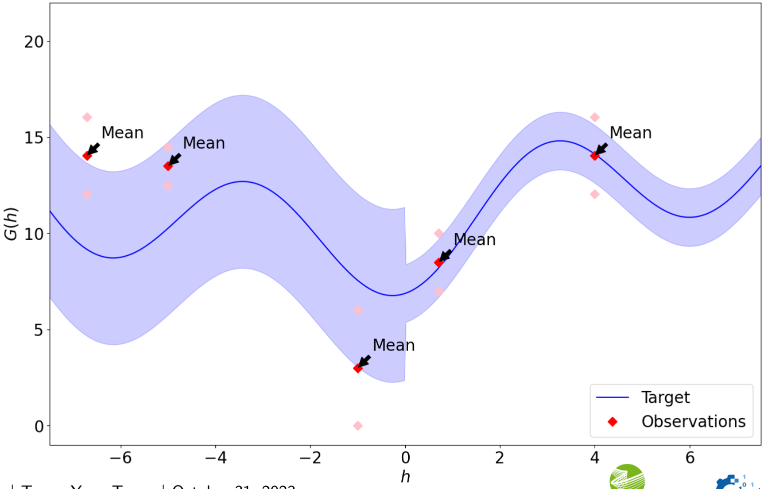
HPO Strategy Comparisons

- Single run: $\mathbb{E}[G_{h_i}] \approx S_{G_{h_i}}$.



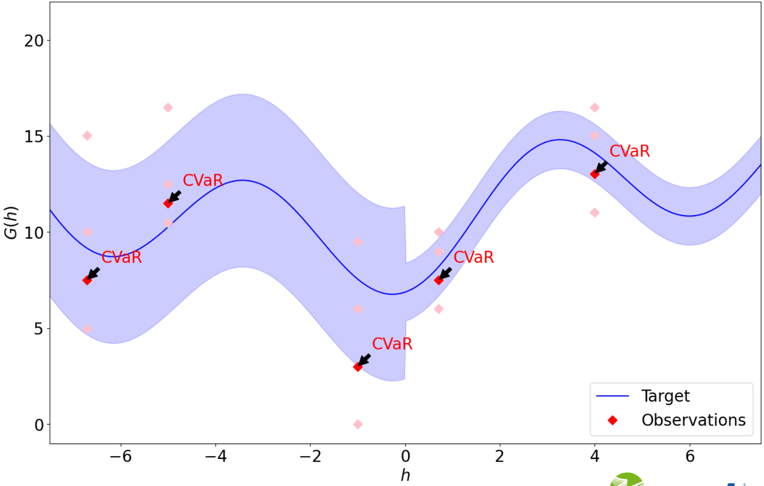
HPO Strategy Comparisons

- Multiple runs: $\mathbb{E}[G_{h_i}] \approx \sum_j S^j_{G_{h_i}}/N$.



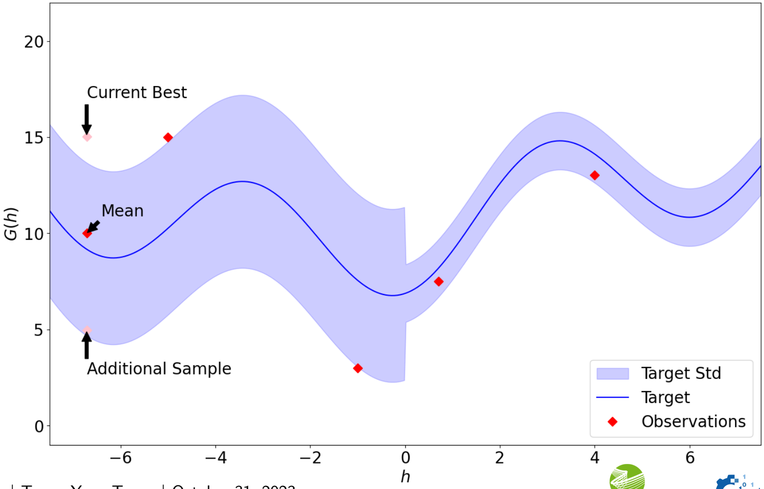
HPO Strategy Comparisons

- Multiple runs with CVaR: CVaR$[G_{h_i}] \approx \hat{\text{CVaR}}_N[G_{h_i}]$.



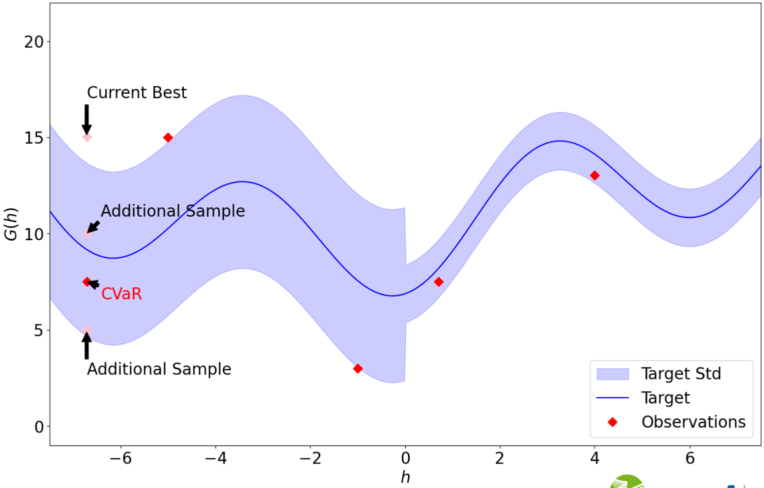
HPO Strategy Comparisons

- AMRA with expectations: $\mathbb{E}[G_{h_i}] \approx \sum_j S^j_{G_{h_i}} / N_i$.

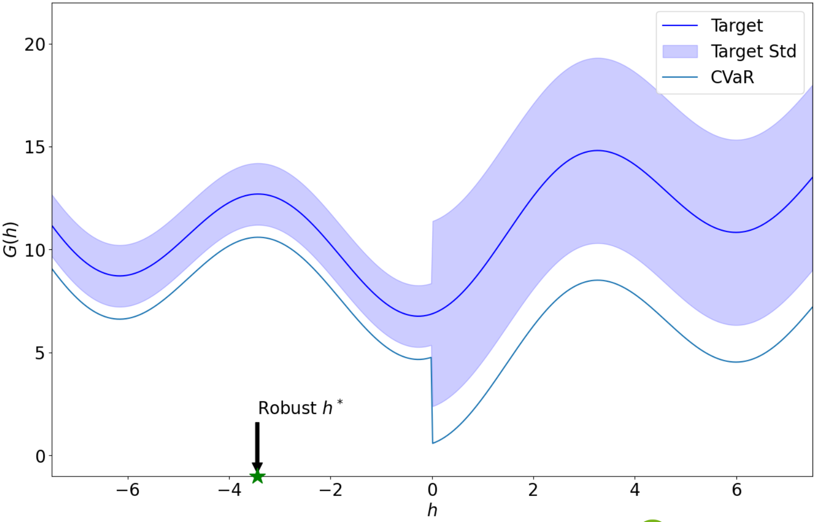


HPO Strategy Comparisons

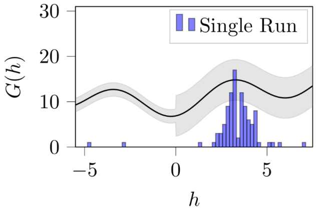
- AMRA with CVaR: $\mathrm{CVaR}[G_{h_i}] \approx \hat{\mathrm{CVaR}}_{N_i}[G_{h_i}]$.



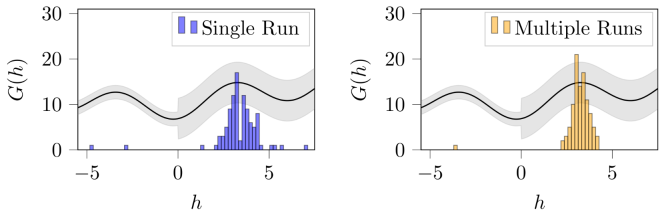
Experiments: Synthetic Examples



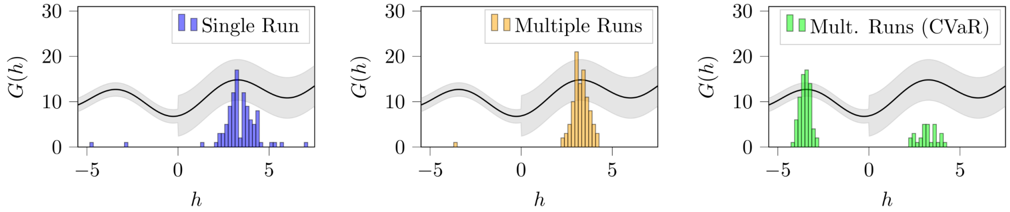
Experiments: Synthetic Examples



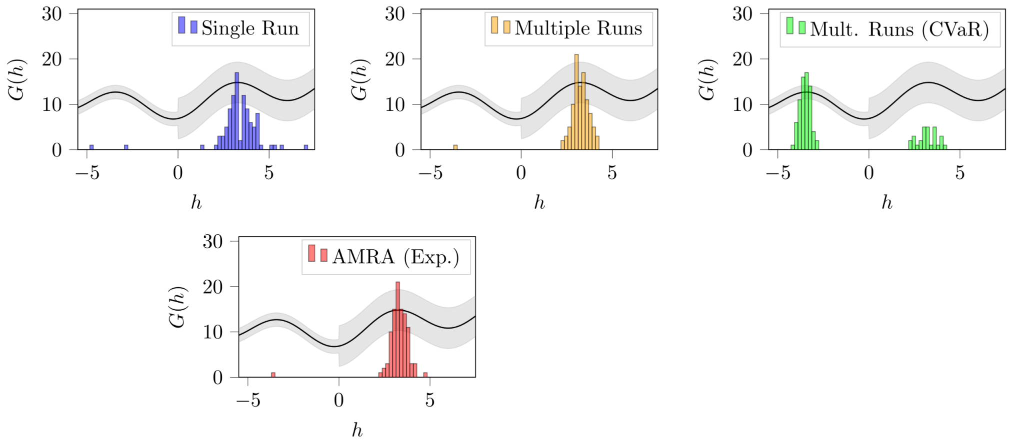
Experiments: Synthetic Examples



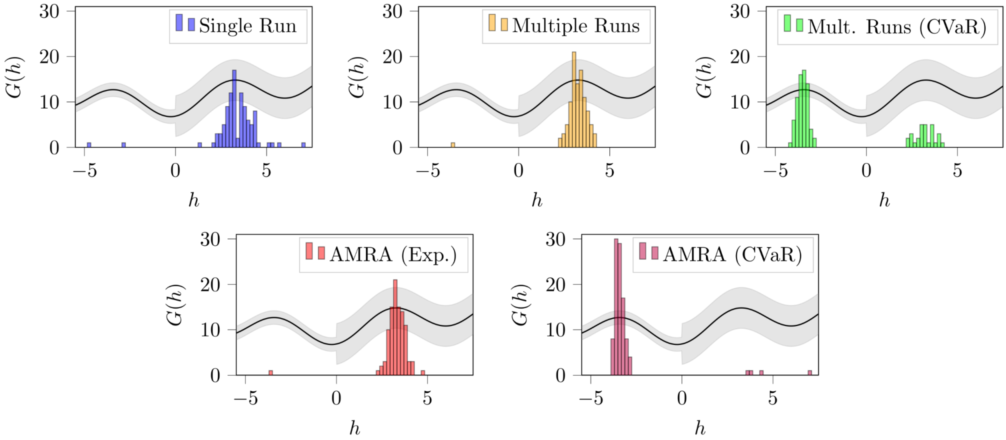
Experiments: Synthetic Examples



Experiments: Synthetic Examples

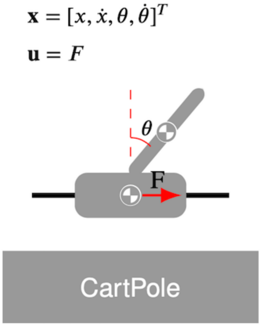


Experiments: Synthetic Examples



Experiments: Decision-Making Algorithms

- Benchmarking environment: Safe-Control-Gym (SCG).
- Control tasks: Cartpole system.
- DDPG, PPO, SAC, and GP-MPC.

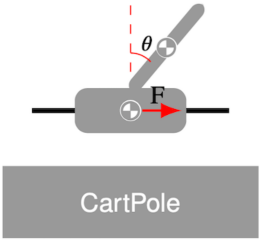


Experiments: Decision-Making Algorithms

- Benchmarking environment: Safe-Control-Gym (SCG).
- Control tasks: Cartpole system.
- DDPG, PPO, SAC, and GP-MPC.

$$\mathbf{x} = [x, \dot{x}, \theta, \dot{\theta}]^T$$

$$\mathbf{u} = F$$



Experiments: Decision-Making Algorithms

Experiments: Decision-Making Algorithms

- Unifying evaluation metric G

Experiments: Decision-Making Algorithms

- Unifying evaluation metric G
- Reward per step i as $J_i = \exp\left(-\left(\boldsymbol{x}_i^T \boldsymbol{Q} \boldsymbol{x}_i + \boldsymbol{u}_i^T \boldsymbol{R} \boldsymbol{u}_i\right)\right)$

Experiments: Decision-Making Algorithms

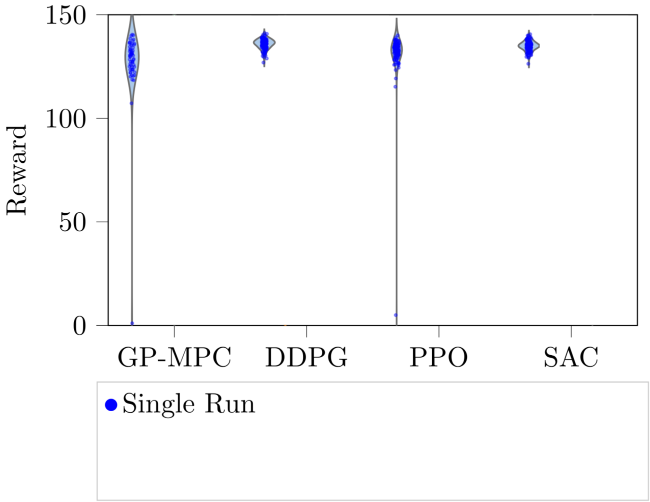
- Unifying evaluation metric G
- Reward per step i as $J_i = \exp\left(-\left(\boldsymbol{x}_i^T \boldsymbol{Q} \boldsymbol{x}_i + \boldsymbol{u}_i^T \boldsymbol{R} \boldsymbol{u}_i\right)\right)$
- $G = \sum_{i=1}^{L} J_i$

Experiments: Decision-Making Algorithms

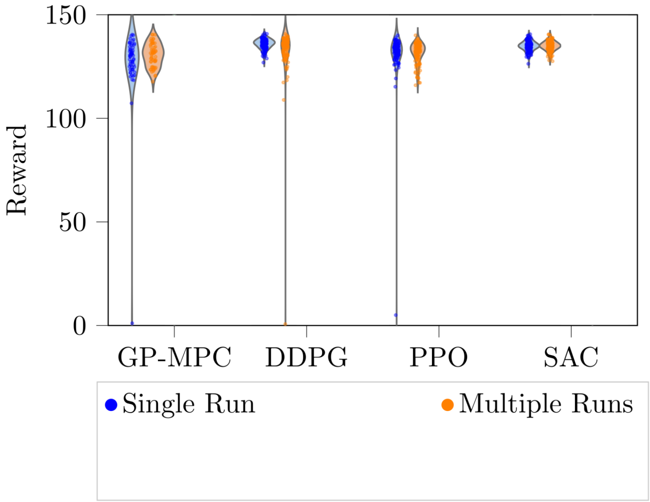
- Unifying evaluation metric G
- Reward per step i as $J_i = \exp\left(-\left(\boldsymbol{x}_i^T \boldsymbol{Q} \boldsymbol{x}_i + \boldsymbol{u}_i^T \boldsymbol{R} \boldsymbol{u}_i\right)\right)$
- $G = \sum_{i=1}^{L} J_i$

A run means one training-and-evaluation loop of a control algorithm

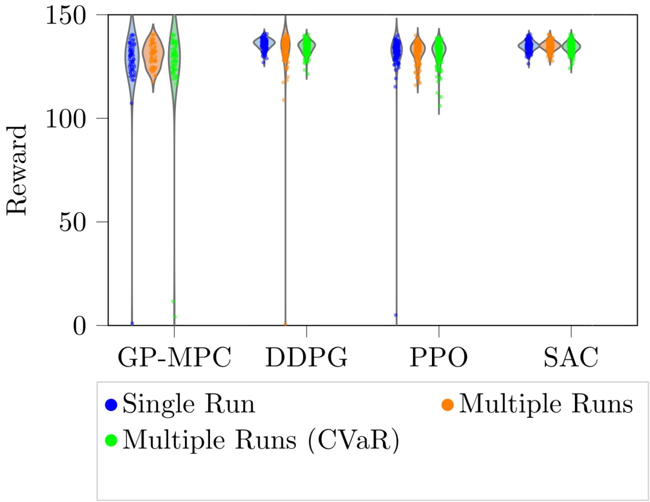
Experiments: Decision-Making Algorithms



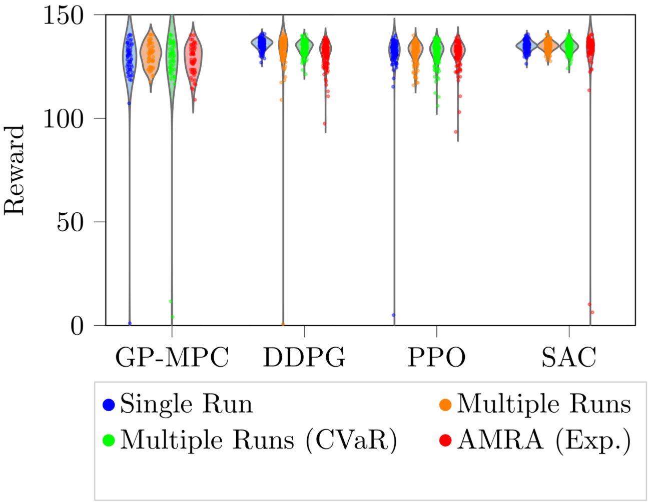
Experiments: Decision-Making Algorithms



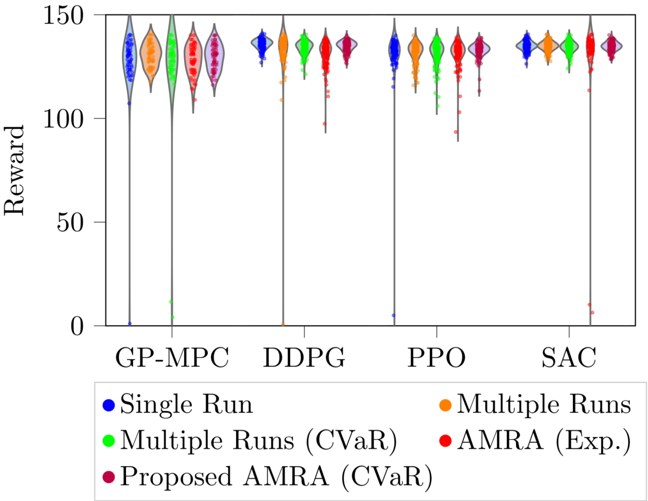
Experiments: Decision-Making Algorithms



Experiments: Decision-Making Algorithms



Experiments: Decision-Making Algorithms

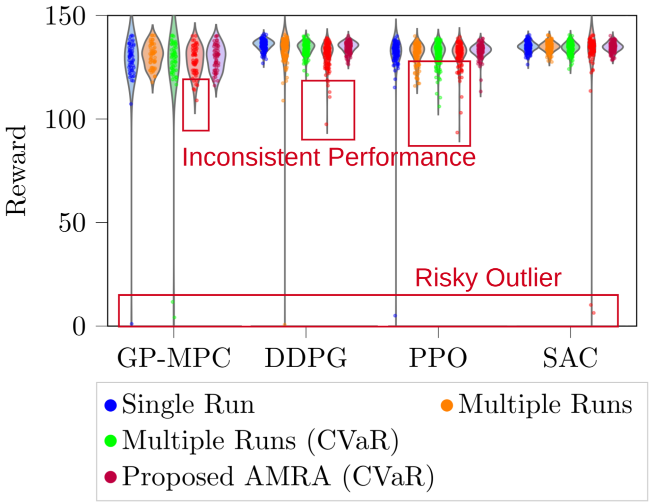


LEARNING SYSTEMS & ROBOTICS LAB

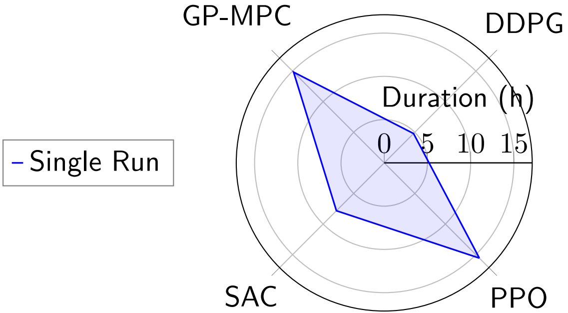
Institute for Data Science in Mechanical Engineering

RWTH AACHEN UNIVERSITY

Experiments: Decision-Making Algorithms

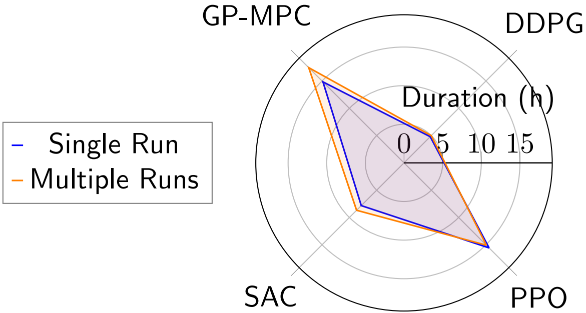


Experiments: Decision-Making Algorithms

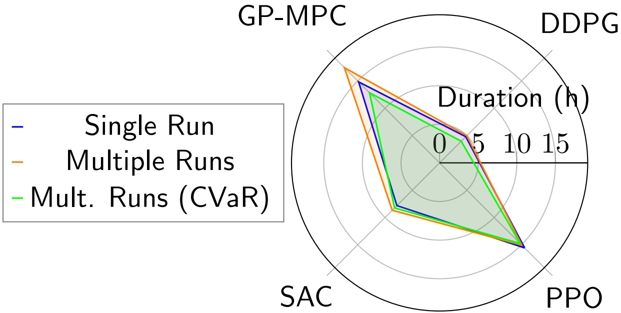


LEARNING SYSTEMS & ROBOTICS LAB TUM AND UofT
Institute for Data Science in Mechanical Engineering
RWTH AACHEN UNIVERSITY

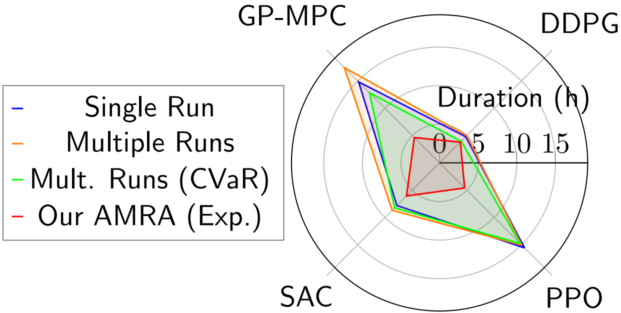
Experiments: Decision-Making Algorithms



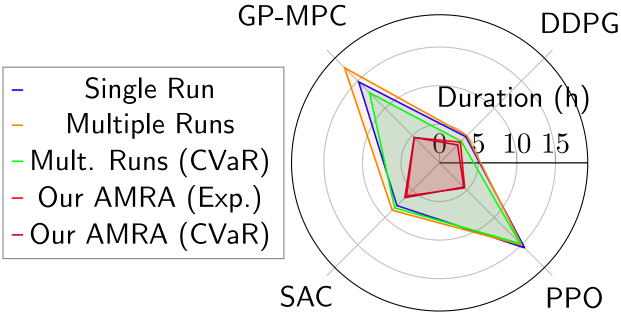
Experiments: Decision-Making Algorithms



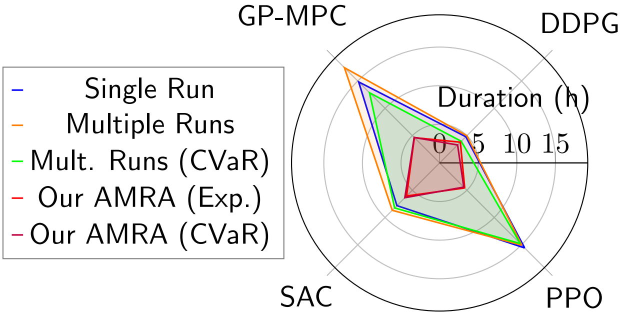
Experiments: Decision-Making Algorithms



Experiments: Decision-Making Algorithms



Experiments: Decision-Making Algorithms



AMRA requires less computational time.

Summary

- We propose a robust and efficient HPO algorithm (AMRA with CVaR).

Summary

- We propose a robust and efficient HPO algorithm (AMRA with CVaR).
  - Adaptive sampling scheme $\rightarrow$ (i) efficiency (ii) bias.

Summary

- We propose a robust and efficient HPO algorithm (AMRA with CVaR).
  - Adaptive sampling scheme $\rightarrow$ (i) efficiency (ii) bias.
  - CVaR $\rightarrow$ risk in tail distributions.

Summary

- We propose a robust and efficient HPO algorithm (AMRA with CVaR).
  - Adaptive sampling scheme $\rightarrow$ (i) efficiency (ii) bias.
  - CVaR $\rightarrow$ risk in tail distributions.
- Robust performance of selected control algorithms is finally attained.

Summary

- We propose a robust and efficient HPO algorithm (AMRA with CVaR).
  - Adaptive sampling scheme $\rightarrow$ (i) efficiency (ii) bias.
  - CVaR $\rightarrow$ risk in tail distributions.
- Robust performance of selected control algorithms is finally attained.
- Reduce computation effort by 25% to 66% across different control algorithms.

Future Works

- Evaluate our method on well-established and widely-used RL platforms such as OpenAI gym.

Future Works

- Evaluate our method on well-established and widely-used RL platforms such as OpenAI gym.
- Toward benchmarking

Future Works

- Evaluate our method on well-established and widely-used RL platforms such as OpenAI gym.
- Toward benchmarking
  - Optimize h that yields consistent good performance for selected algorithms for fairer comparisons.

Q & A

Thank You!

Bibliography I

- Bergstra, James and Yoshua Bengio (2012). "Random search for hyper-parameter optimization.". In: *Journal of machine learning research* 13.2.
- Bischl, Bernd et al. (2021). "Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, e1484.
- Eggensperger, Katharina, Marius Lindauer, and Frank Hutter (2019). "Pitfalls and best practices in algorithm configuration". In: *Journal of Artificial Intelligence Research* 64, pp. 861–893.
- Eimer, Theresa, Marius Lindauer, and Roberta Raileanu (2023). "Hyperparameters in Reinforcement Learning and How To Tune Them". In: *arXiv preprint arXiv:2306.01324*.
- Henderson, Peter et al. (2018). "Deep reinforcement learning that matters". In: *Proceedings of the AAAI conference on artificial intelligence*. Vol. 32. 1.

Bibliography II

Hutter, Frank, Holger H Hoos, and Thomas Stützle (2007). "Automatic algorithm configuration based on local search". In: *Aaai*. Vol. 7, pp. 1152–1157.

Patterson, Andrew et al. (2023). "Empirical Design in Reinforcement Learning". In: *arXiv preprint arXiv:2304.01315*.

Snoek, Jasper, Hugo Larochelle, and Ryan P Adams (2012). "Practical bayesian optimization of machine learning algorithms". In: *Advances in neural information processing systems* 25.

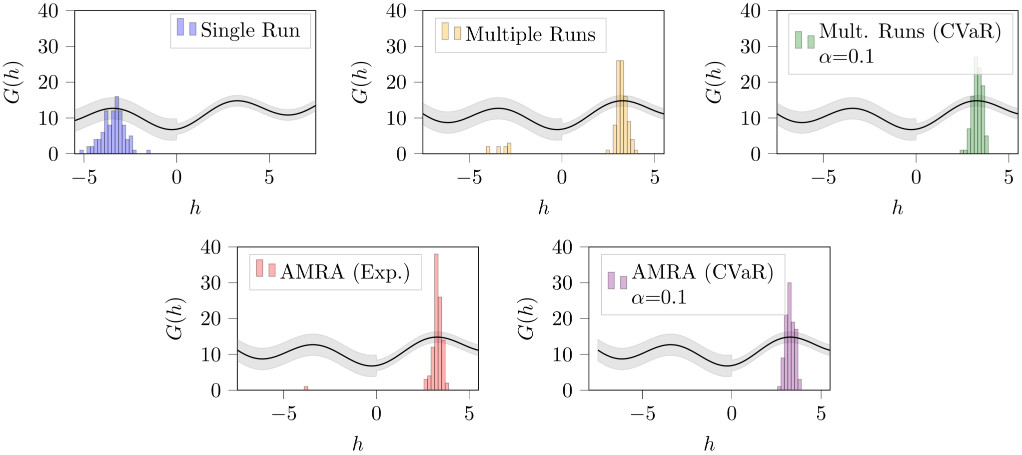
Zhang, Baohe et al. (2021). "On the importance of hyperparameter optimization for model-based reinforcement learning". In: *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 4015–4023.

Appendix: Method

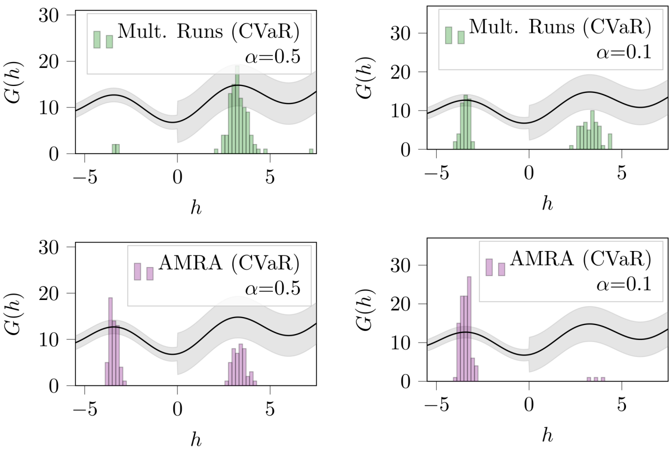
Algorithm 2 Adaptive Multiple Runs Algorithm with General Risk Functional

procedure ADAPTIVE MULTIPLE RUNS ALGORITHM WITH GENERAL RISK FUNCTIONAL
    Initialize an HPO algorithm $\mathcal{A}$, estimator $\hat{\rho}_N[G_{\boldsymbol{h}}]$ of an agent given a environment ϵ, number of basic agent runs N, incremented agent runs N^+, number of trials M, dataset $\mathcal{D} = \emptyset$, and $\hat{\rho}^{ss}[G_{\boldsymbol{h}}] = \inf$
    for each trial **do**
        $\boldsymbol{h}_i \leftarrow \mathcal{A}(\mathcal{D})$ ▷ Sample next HPs
        Reset N to the initial value
        $\hat{\rho}^s[G_{\boldsymbol{h}_i}] \leftarrow \hat{\rho}_N[G_{\boldsymbol{h}_i}]$
        $\hat{\rho}[G_{\boldsymbol{h}_i}] \leftarrow \hat{\rho}^s[G_{\boldsymbol{h}_i}]$
        if $\hat{\rho}^s[G_{\boldsymbol{h}_i}]$ is current best trial **then**
            while $|\hat{\rho}^s[G_{\boldsymbol{h}_i}] - \hat{\rho}^{ss}[G_{\boldsymbol{h}_i}]| > \delta$ **do**
                if not the first iteration **then**
                    $\hat{\rho}^s[G_{\boldsymbol{h}_i}] \leftarrow \hat{\rho}^{ss}[G_{\boldsymbol{h}_i}]$
                end if
                $N \leftarrow N + N^+$
                $\hat{\rho}^{ss}[G_{\boldsymbol{h}_i}] \leftarrow \hat{\rho}_N[G_{\boldsymbol{h}_i}]$
            end while
            $\hat{\rho}[G_{\boldsymbol{h}_i}] \leftarrow \hat{\rho}^{ss}[G_{\boldsymbol{h}_i}]$
        end if
        $\mathcal{D} \leftarrow \mathcal{D} \cup (\boldsymbol{h}_i, \hat{\rho}[G_{\boldsymbol{h}_i}])$
    end for
    return $\boldsymbol{h}^* \leftarrow \underset{\boldsymbol{h}_i \in \mathbb{H}}{\arg\max}\ \hat{\rho}[G_{\boldsymbol{h}_i}]$
end procedure

Appendix: Additional Experiments



Appendix: Additional Experiments



Appendix: Additional Experiments