Incorporating Uncertainty into Reinforcement Learning through Gaussian Processes

Master's Thesis

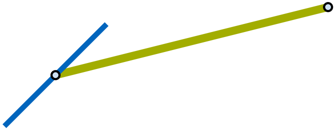
Markus Kaiser

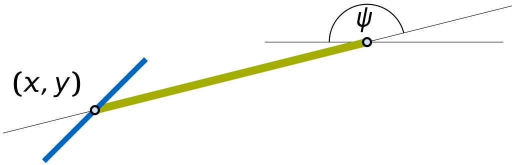
4. July 2016

Outline

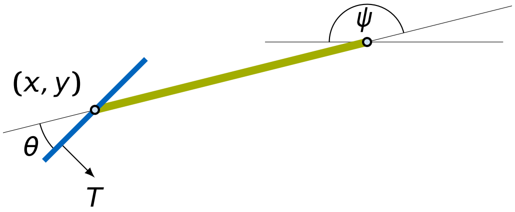
Bicycle Benchmark – Driving

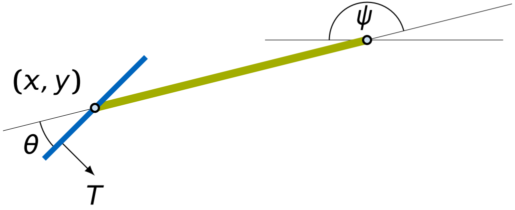
Bicycle Benchmark – Driving

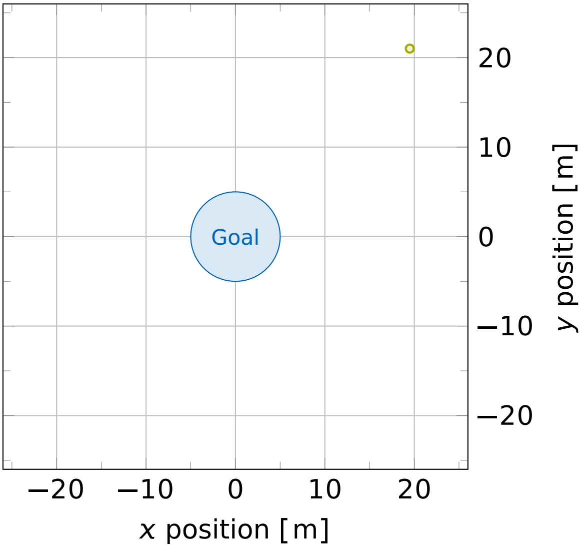




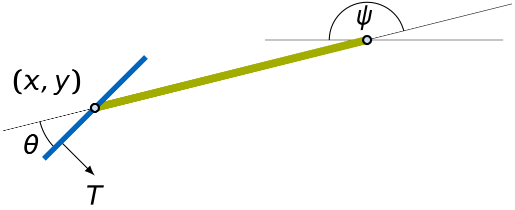
ψ
(x, y)

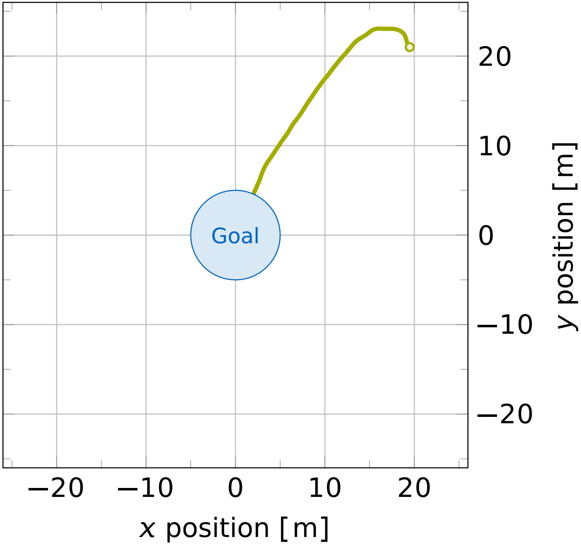

ψ
(x, y)
θ
T

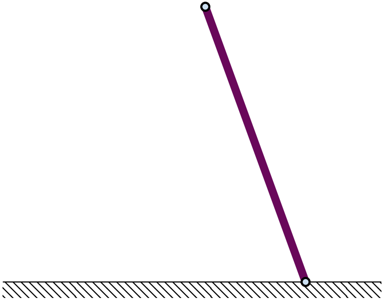

ψ
(x, y)
θ
T


Goal
20
10
0
−10
−20
y position [m]
−20
−10
0
10
20
x position [m]

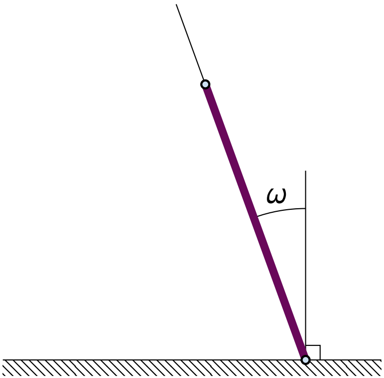
Bicycle Benchmark – Driving



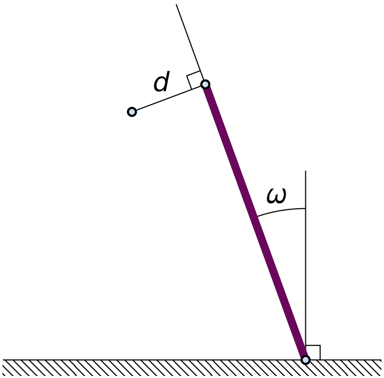




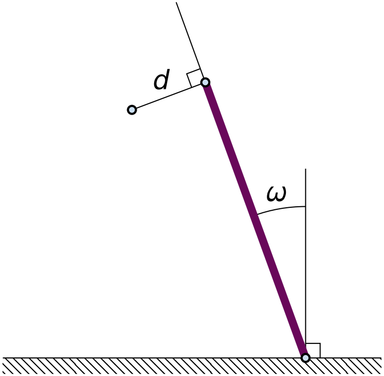
Bicycle Benchmark – Balancing

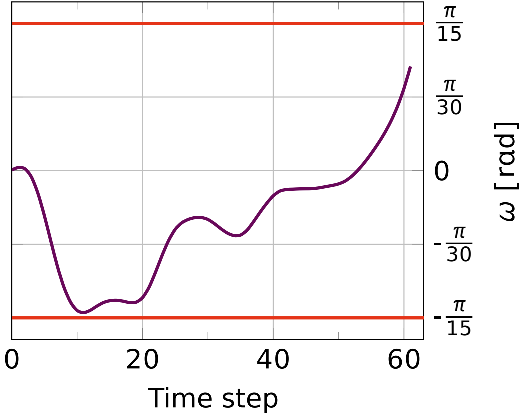


Bicycle Benchmark – Balancing



Bicycle Benchmark – Balancing





State Variables

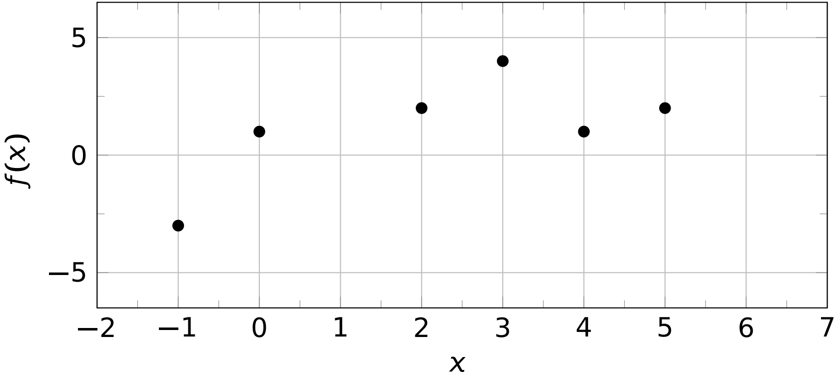
Notation	Description
θ	Steering Angle
$\dot{\theta}$	
ω	Leaning Angle
$\dot{\omega}$	
x	Front tyre position
y	
ψ	Bicycle orientation

Actions

Notation	Description
d	Lean distance
T	Steering force

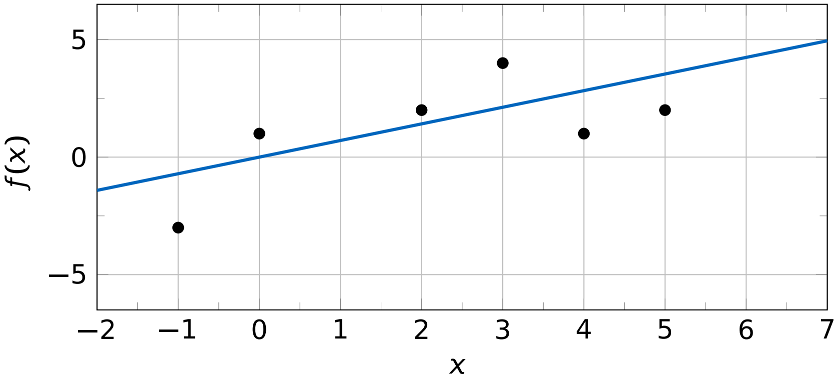
Data

- Observe a noisy data set $\{(\boldsymbol{x_i}, \boldsymbol{y_i})\}$
- Assume that $\boldsymbol{y_i} = f(\boldsymbol{x_i}) + \epsilon$



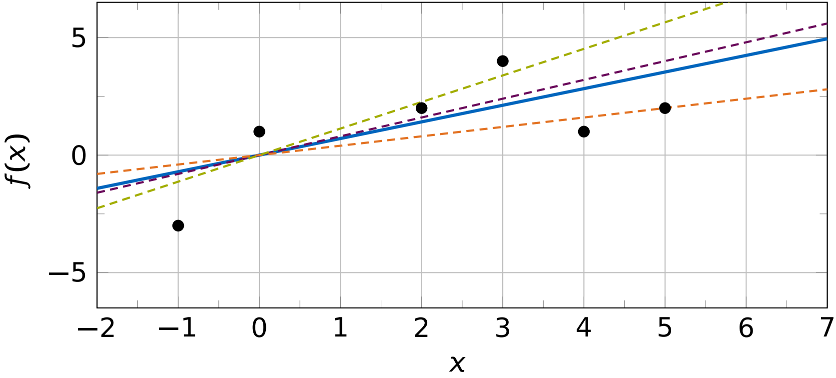
Parametric Models

- Assume structure about f, such as $f(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{x}$
- Find a single $\boldsymbol{W}$ which is optimal w.r.t. some criterion



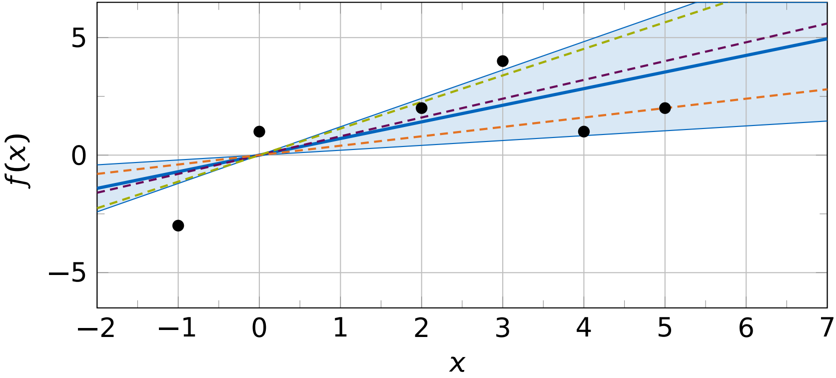
Parametric Models

- Assume structure about f, such as $f(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{x}$
- Find a single $\boldsymbol{W}$ which is optimal w.r.t. some criterion



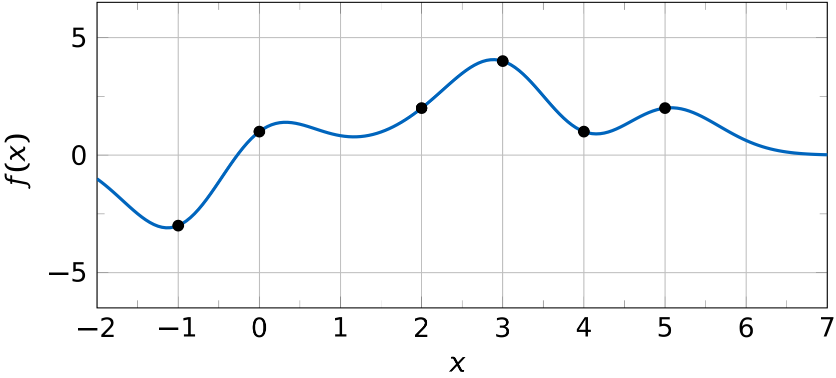
Bayesian Parametric Models

- Assume structure about f, such as $f(\boldsymbol{x}) = \boldsymbol{W}\boldsymbol{x}$
- Find a distribution $\mathrm{p}(\boldsymbol{W} \mid \boldsymbol{X}, \boldsymbol{y})$ of plausible parameters



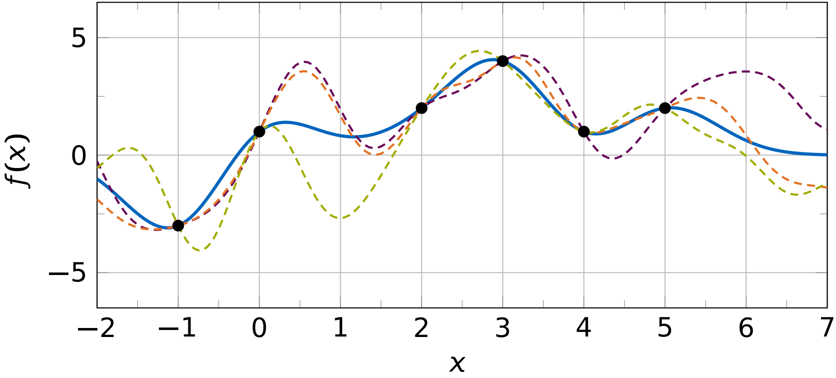
Bayesian Non-Parametric Models

- f can be an arbitrary function
- Find a distribution $\mathrm{p}(f \mid \boldsymbol{X}, \boldsymbol{y})$ over functions



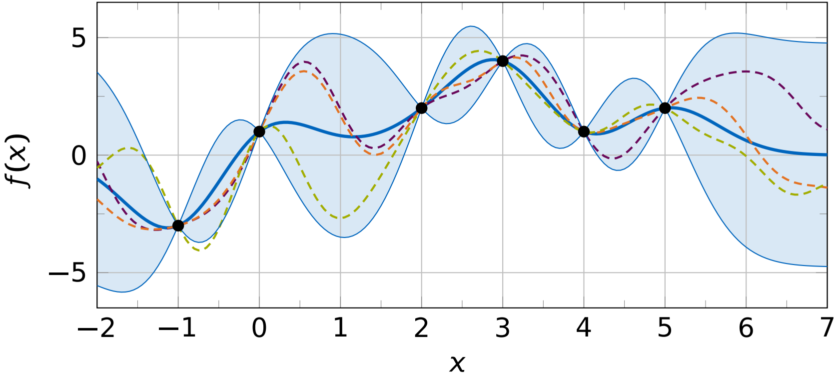
Bayesian Non-Parametric Models

- f can be an arbitrary function
- Find a distribution $p(f \mid \boldsymbol{X}, \boldsymbol{y})$ over functions



Bayesian Non-Parametric Models

- f can be an arbitrary function
- Find a distribution $p(f \mid \boldsymbol{X}, \boldsymbol{y})$ over functions



Definition (Gaussian Process)

A Gaussian Process (GP) is a collection of random variables $\{\boldsymbol{F}_{\boldsymbol{x}}\}$, any finite subset of which has a joint Gaussian distribution.

- Extension of Gaussians to (infinite) function spaces
- $\boldsymbol{F}_{\boldsymbol{x}}$ models the function value $f(x)$

Definition (Gaussian Process)

A Gaussian Process (GP) is a collection of random variables $\{\boldsymbol{F}_{\boldsymbol{x}}\}$, any finite subset of which has a joint Gaussian distribution.

- Extension of Gaussians to (infinite) function spaces
- $\boldsymbol{F}_{\boldsymbol{x}}$ models the function value $f(x)$

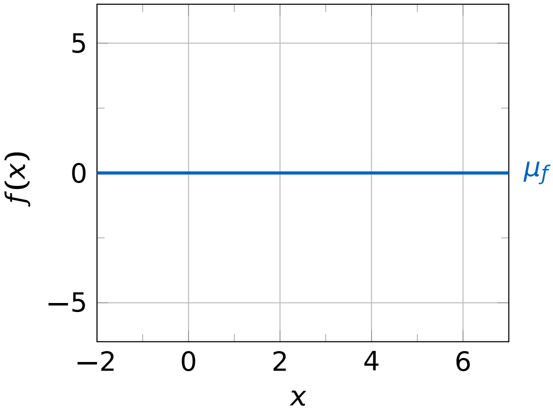
Mean and Kernel Functions

A GP is completely determined by two functions.

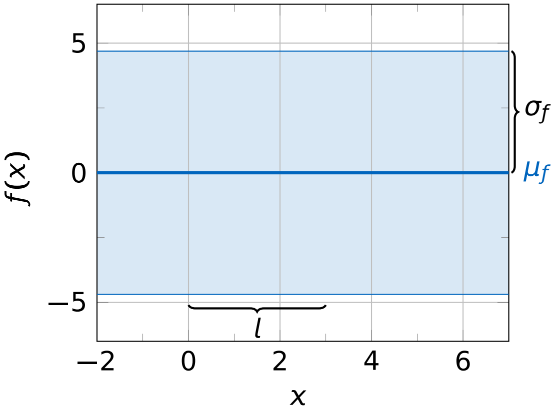
Mean function $\mu_f(x) = \mathbb{E}[f(x)]$

Kernel function $\mathcal{K}(x, x') = \mathrm{cov}[f(x), f(x')]$

- The kernel encodes the prior assumptions about the function



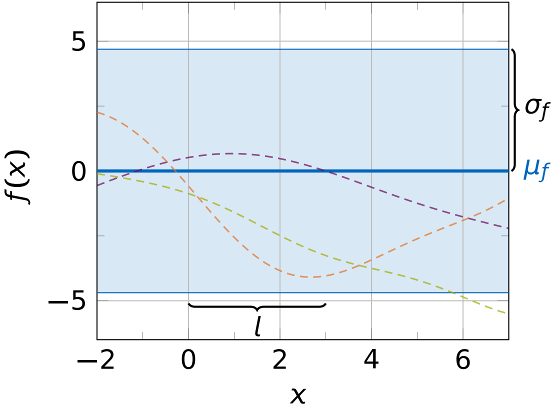
$$\mu_f(x) = 0$$



$$\mu_f(x) = 0$$

$$\mathcal{K}(x, x') = \sigma_f \cdot \exp\left(-\frac{(x - x')^2}{2 \cdot l^2}\right)$$

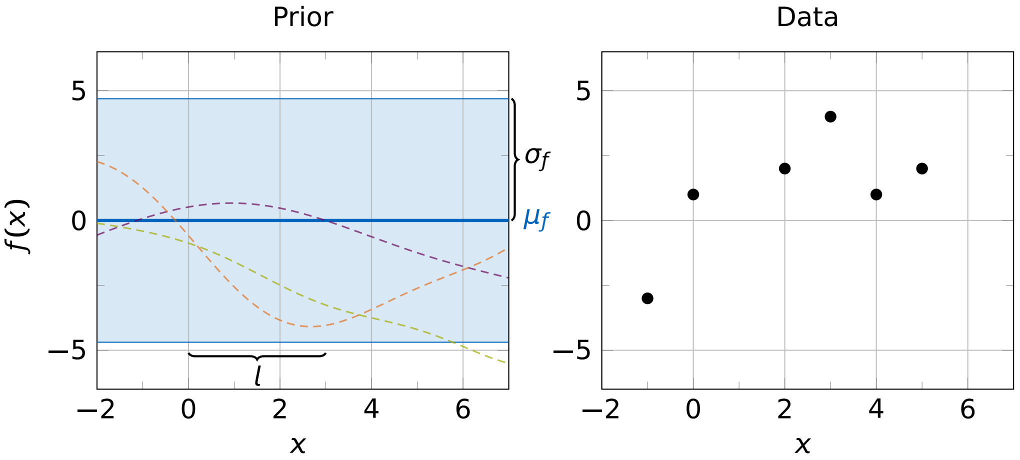
Hyperparameters and Posterior



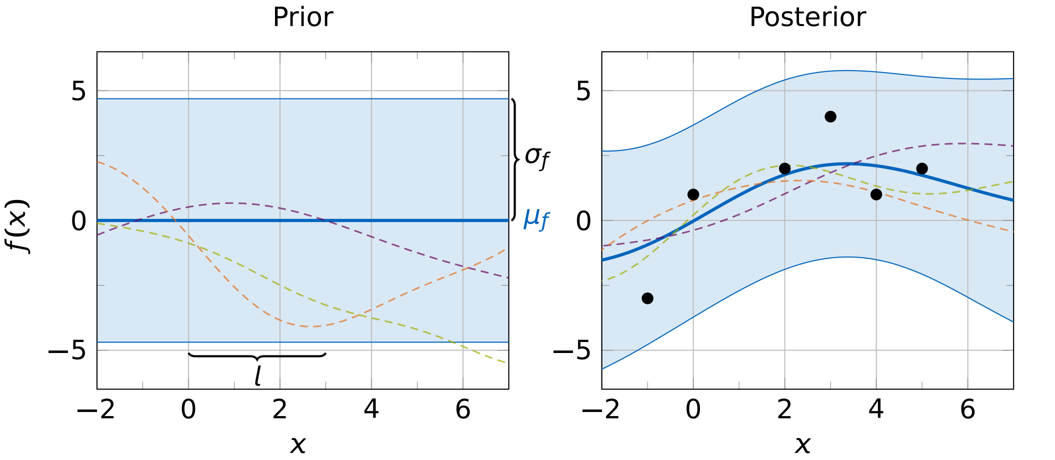
$$\mu_f(x) = 0$$

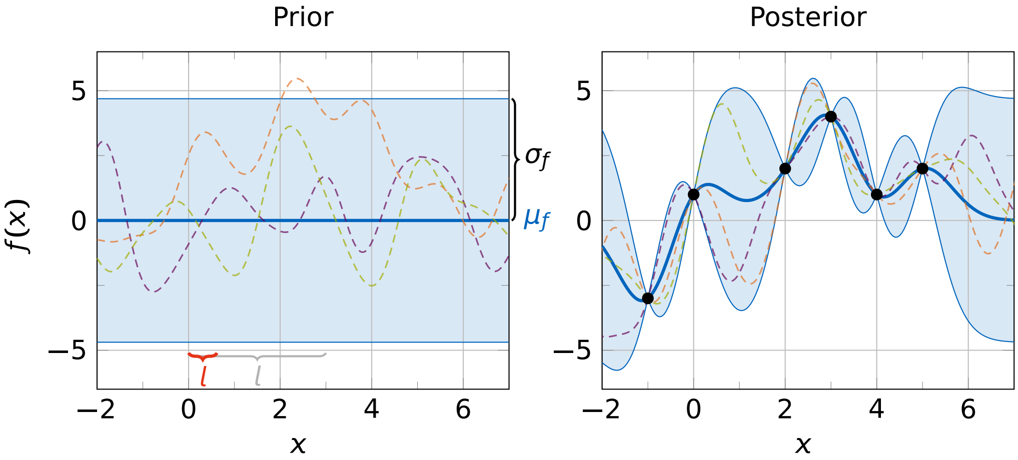
$$\mathcal{K}(x, x') = \sigma_f \cdot \exp\left(-\frac{(x - x')^2}{2 \cdot l^2}\right)$$

Hyperparameters and Posterior



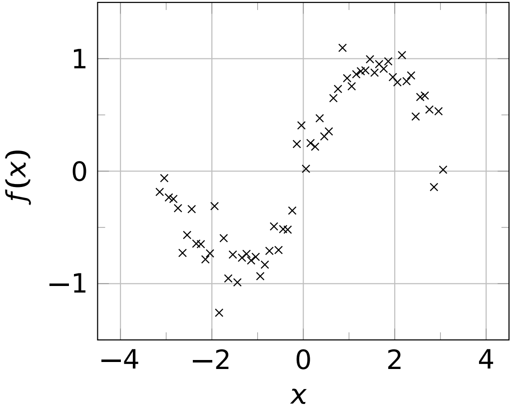
Hyperparameters and Posterior




Prior
Posterior
5
0
−5
−2
0
2
4
6
$f(x)$
x
σ_f
μ_f
l
l

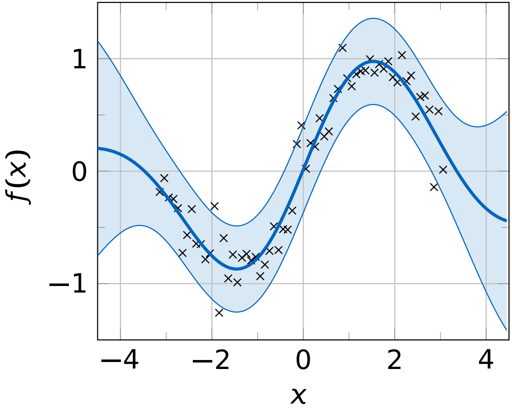
Full Gaussian Processes

- Calculation of the posterior in $\mathcal{O}(N^3)$
- Not feasible for large data sets



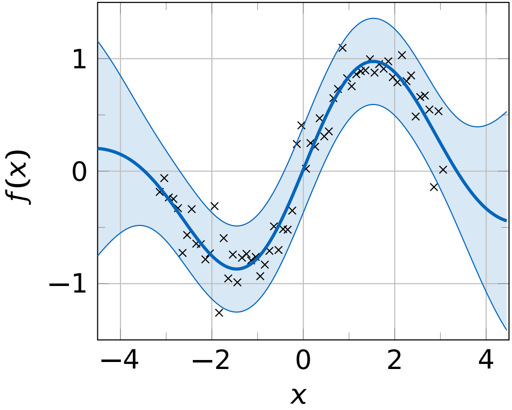
Full Gaussian Processes

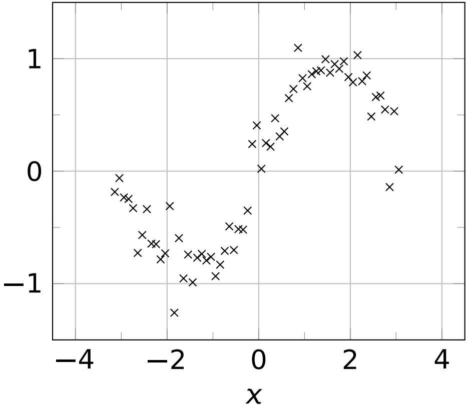
- Calculation of the posterior in $\mathcal{O}(N^3)$
- Not feasible for large data sets



Sparse Gaussian Processes

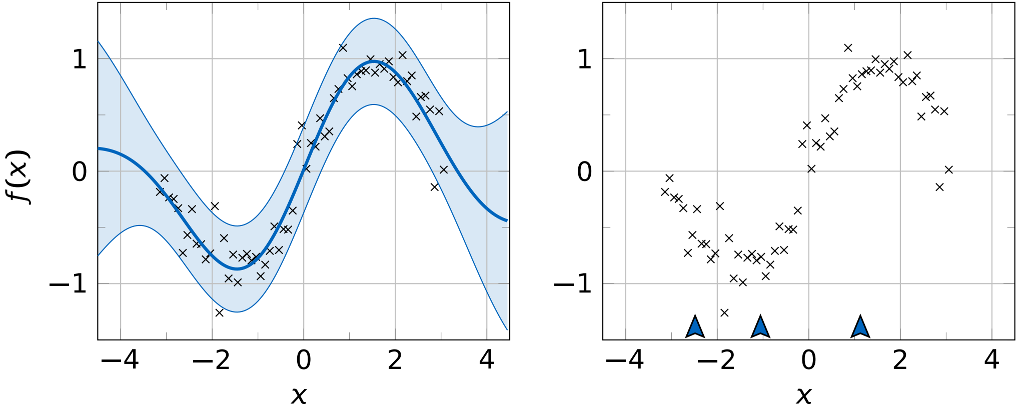
- Find a set of M Pseudo Inputs
- Calculation of the posterior in $\mathcal{O}(NM^2)$





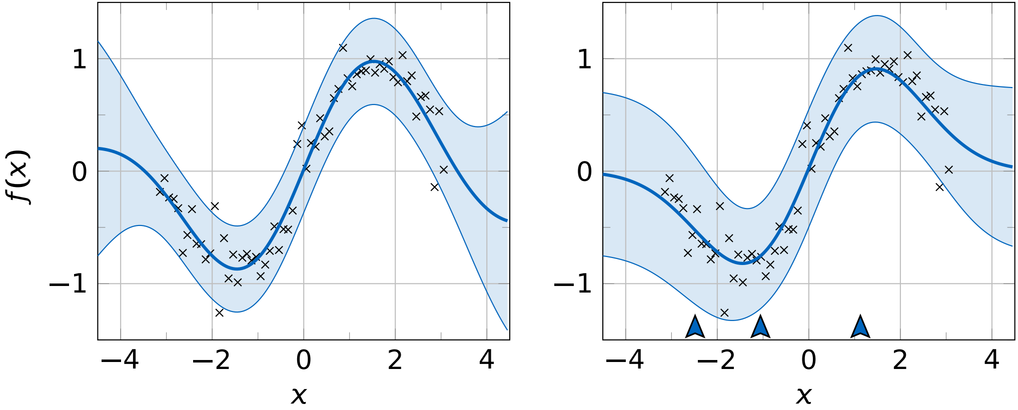
Sparse Gaussian Processes

- Find a set of M Pseudo Inputs
- Calculation of the posterior in $\mathcal{O}(NM^2)$

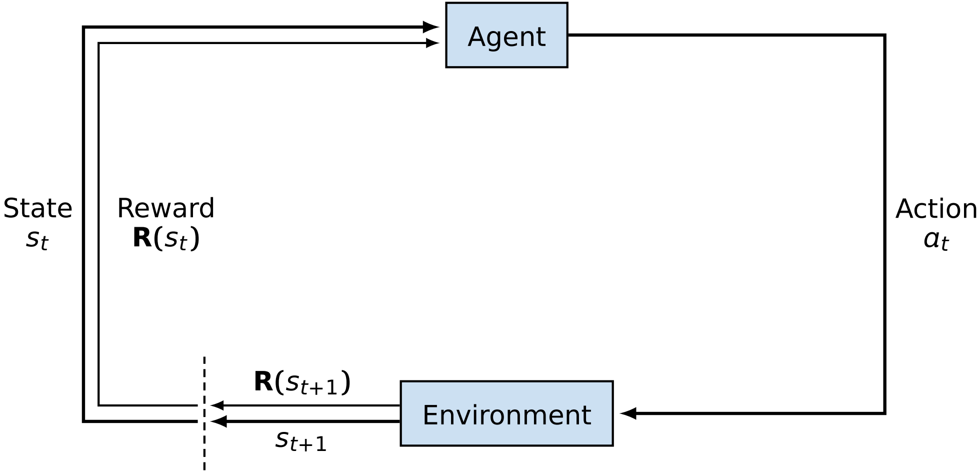


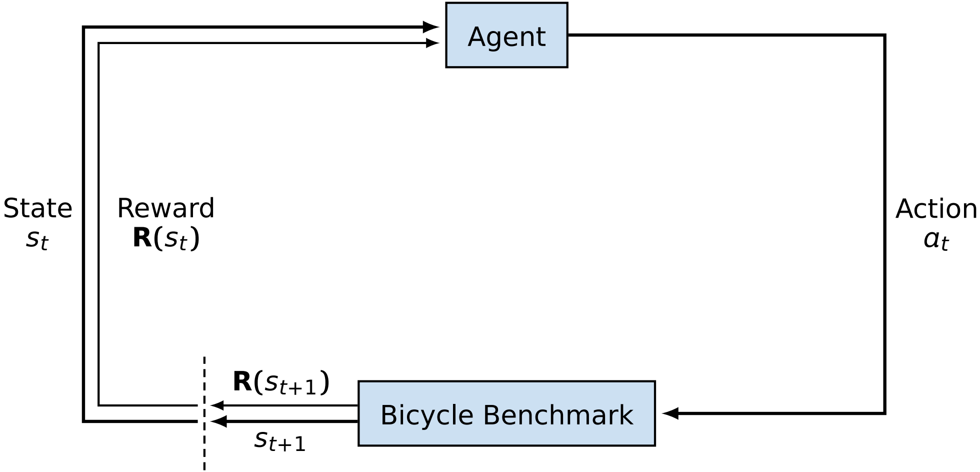
Sparse Gaussian Processes

- Find a set of M Pseudo Inputs
- Calculation of the posterior in $\mathcal{O}(NM^2)$



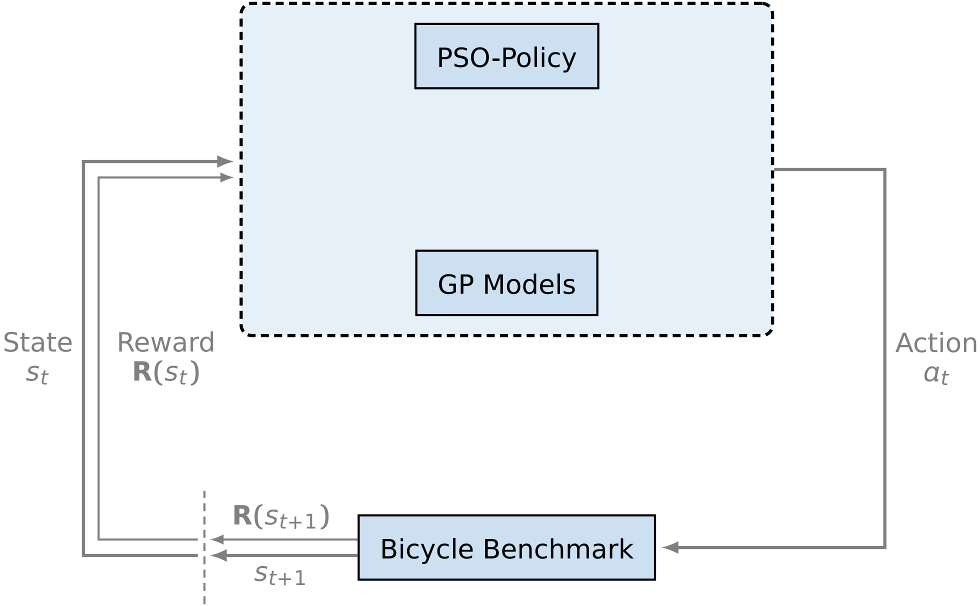
Solution Architecture



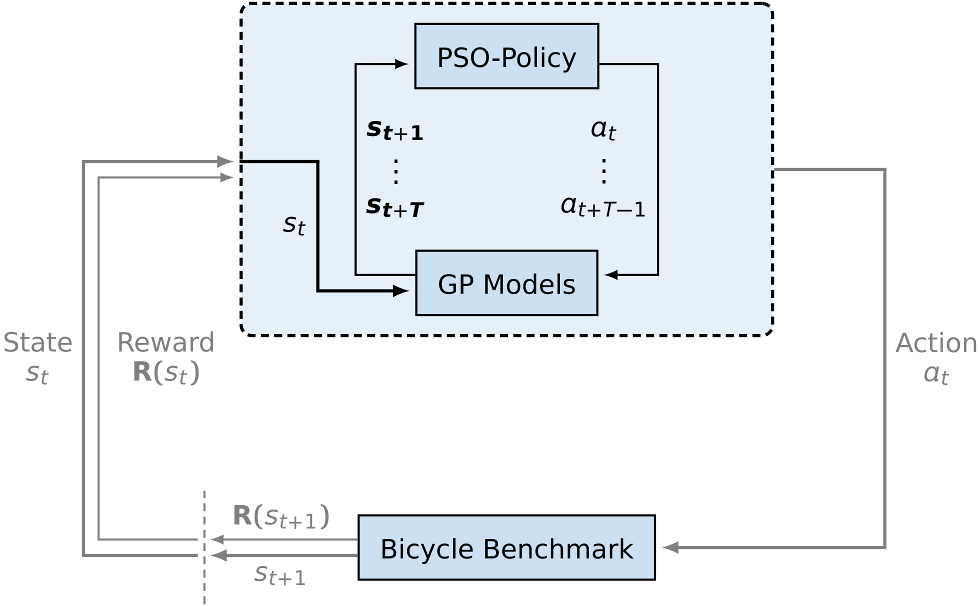


Agent
State
s_t
Reward
$\mathbf{R}(s_t)$
Action
a_t
$\mathbf{R}(s_{t+1})$
Bicycle Benchmark
s_{t+1}

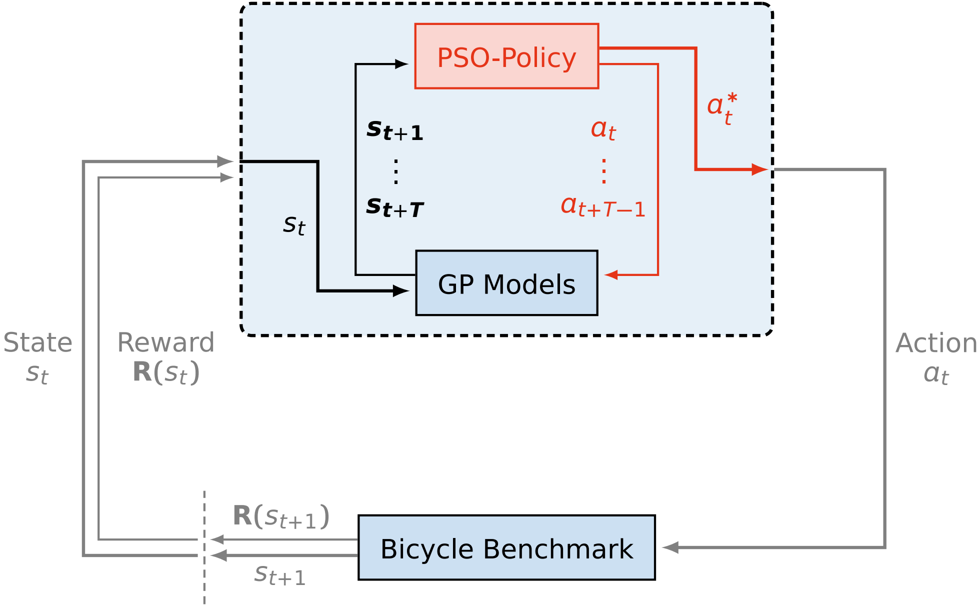
Solution Architecture



Solution Architecture



Solution Architecture



Definition (PSO-Policy)

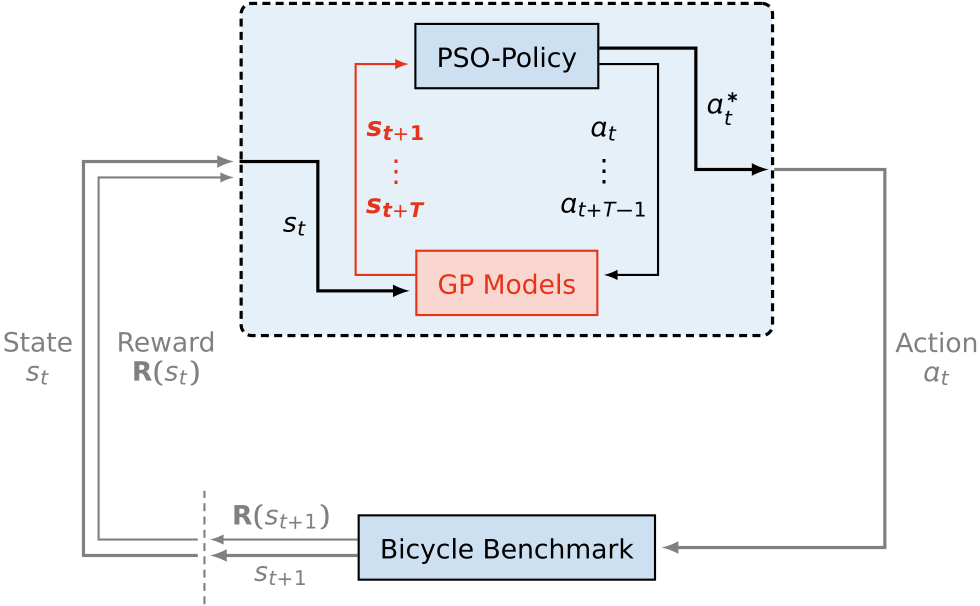
The Particle Swarm Optimization-Policy (PSO-P) chooses actions via optimization of the expected accumulated reward.

$$\pi_{\text{PSO-P}}(\boldsymbol{s}) := \boldsymbol{a}_0^*, \text{ where}$$

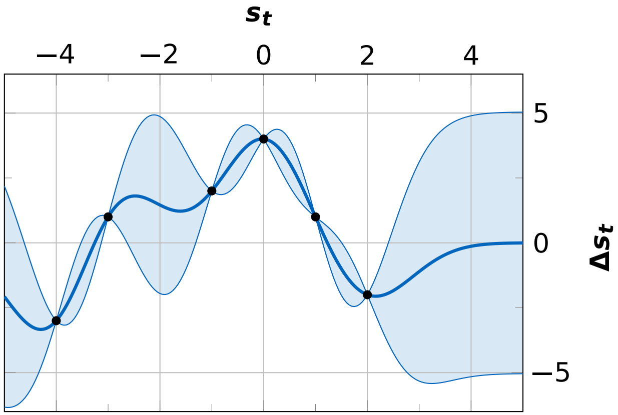
$$\boldsymbol{a}^* \in \operatorname*{argmax}_{\boldsymbol{a} \in \mathcal{A}^T} \left\{ \mathbb{E}\left[\sum_{t=1}^{T} \gamma^t \, \mathbf{R}(\boldsymbol{s_t}) \,\middle|\, \mathcal{GP}, \boldsymbol{s_0} = \boldsymbol{s}, \boldsymbol{a_0}, \ldots, \boldsymbol{a_{T-1}} \right] \right\}$$

- PSO is a gradient-free heuristic
- Directly exploits the transition models

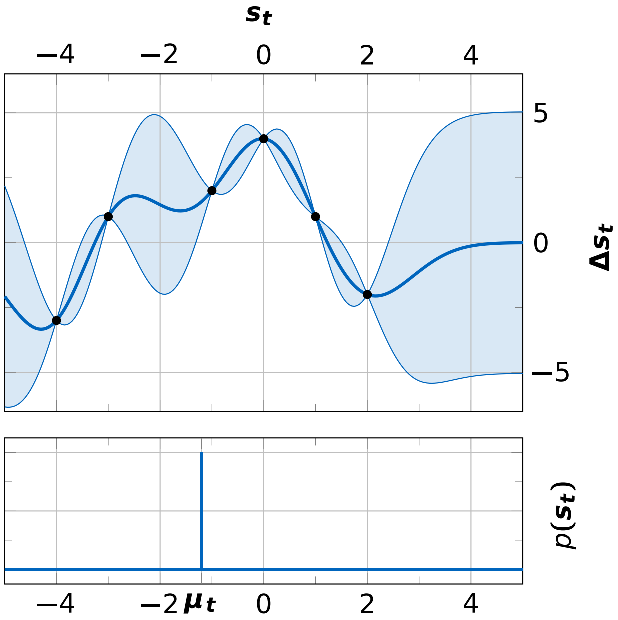
Solution Architecture



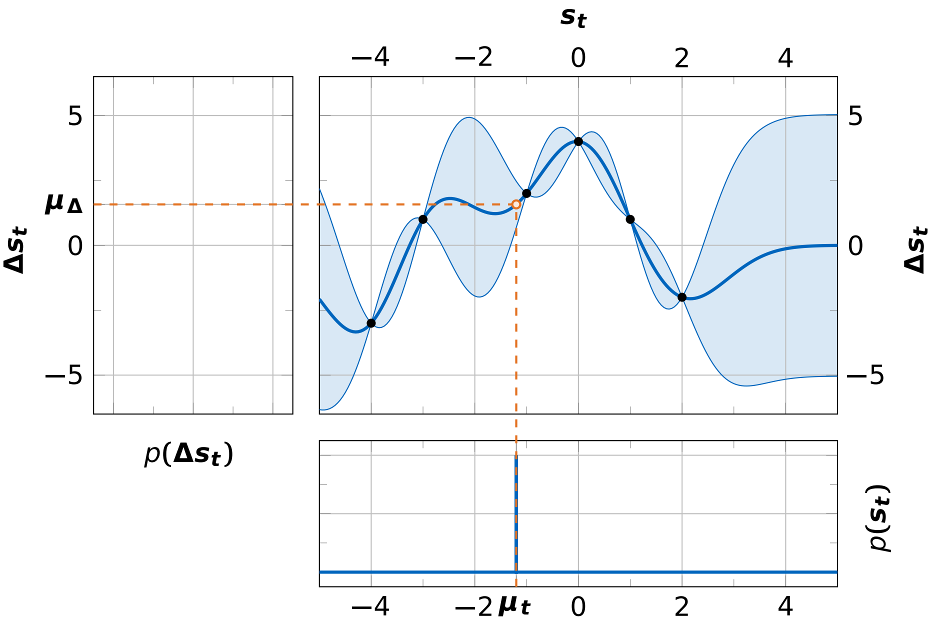
MAP without Uncertainties



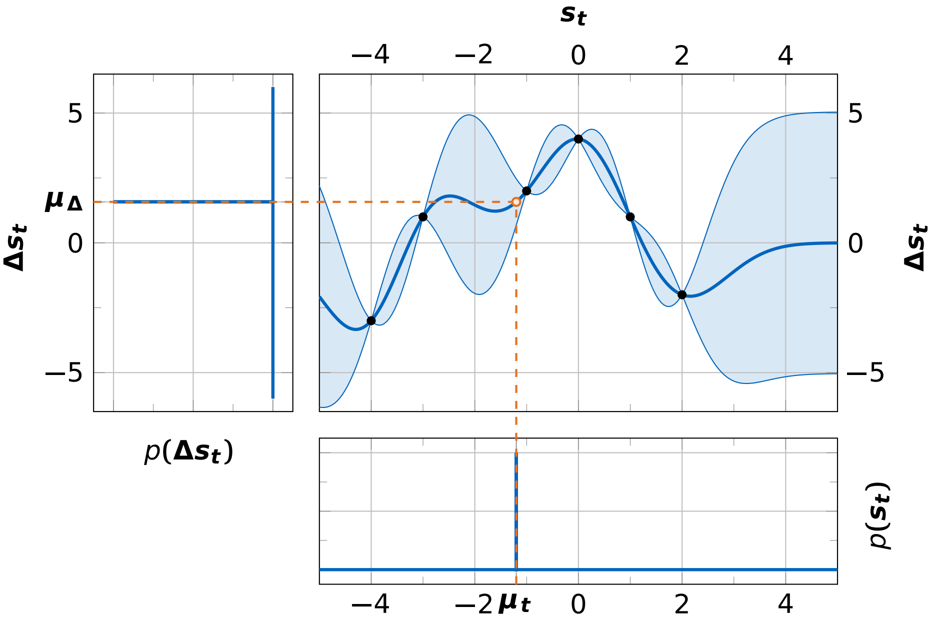
MAP without Uncertainties



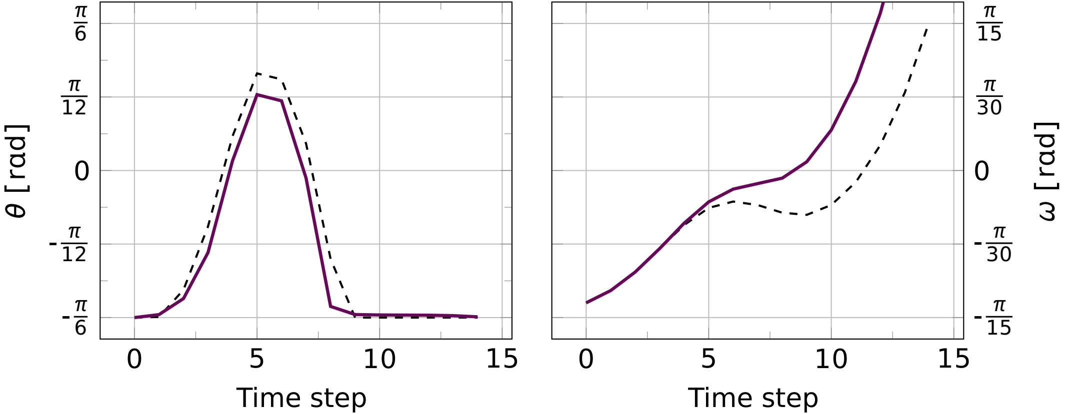
MAP without Uncertainties



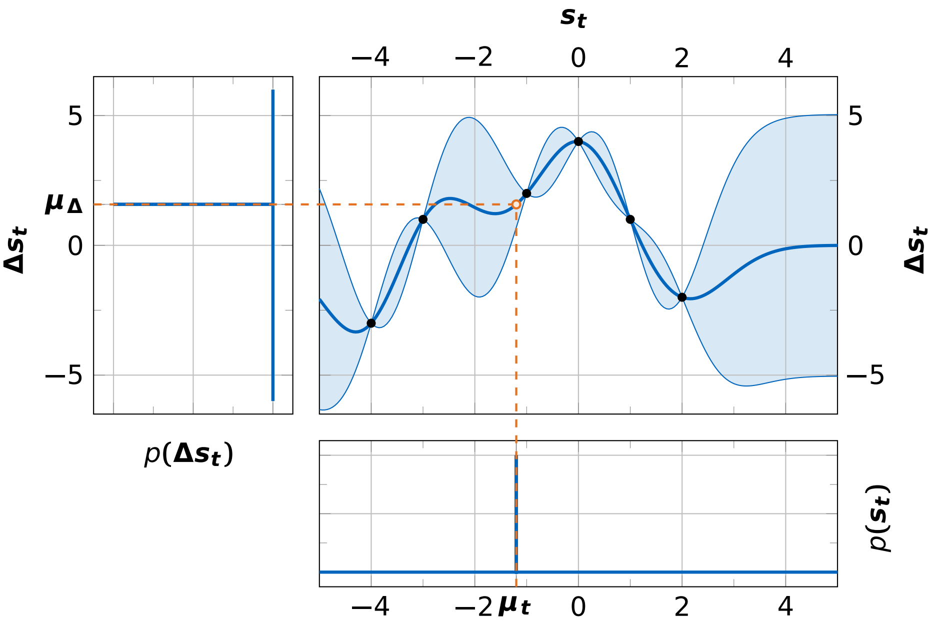
MAP without Uncertainties



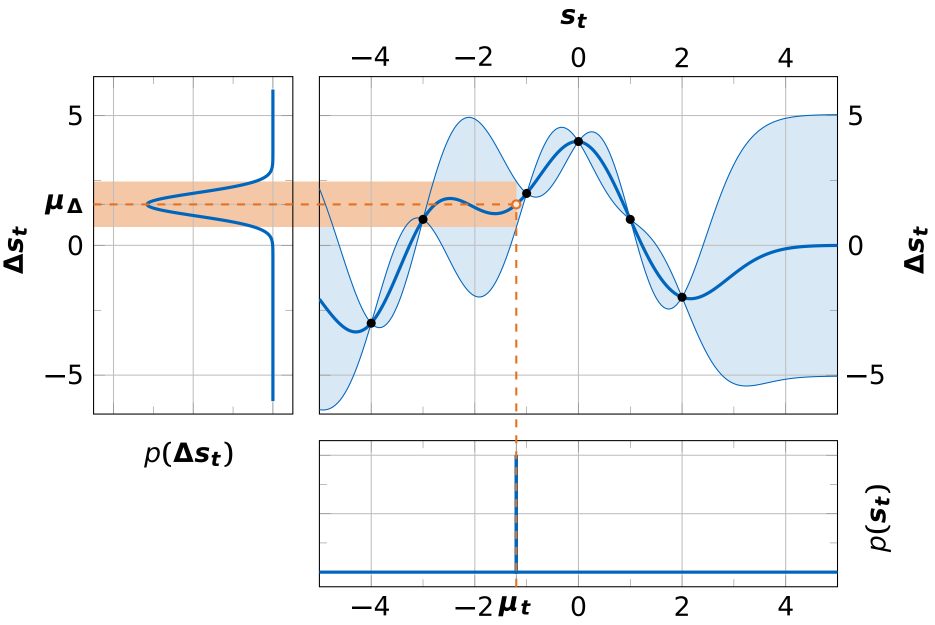
MAP Predictions

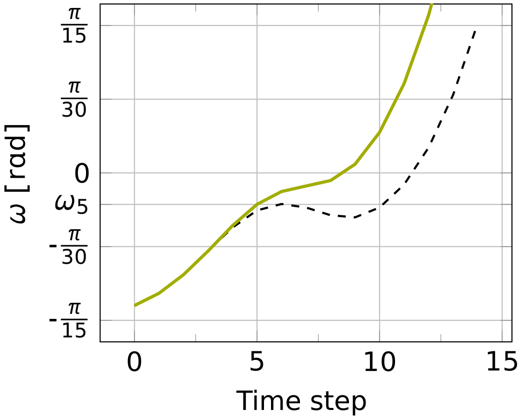


One-Step Uncertainties



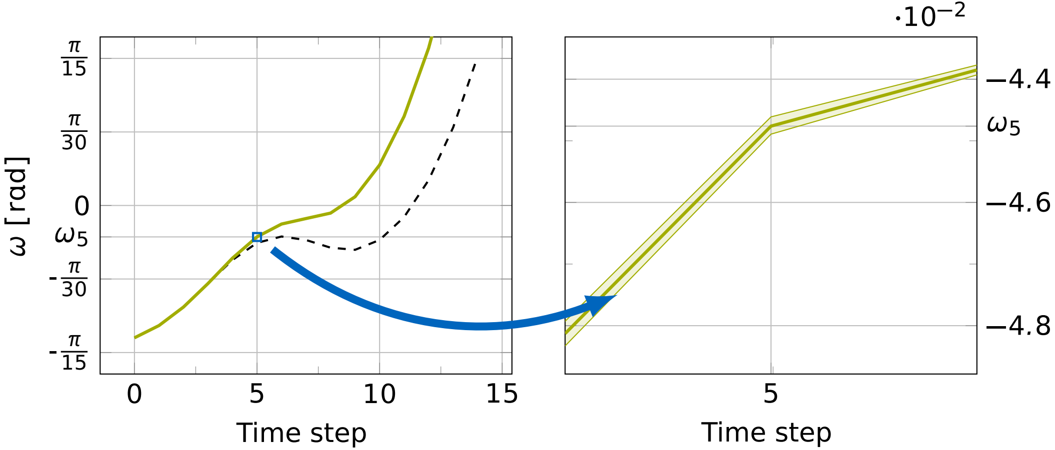
One-Step Uncertainties



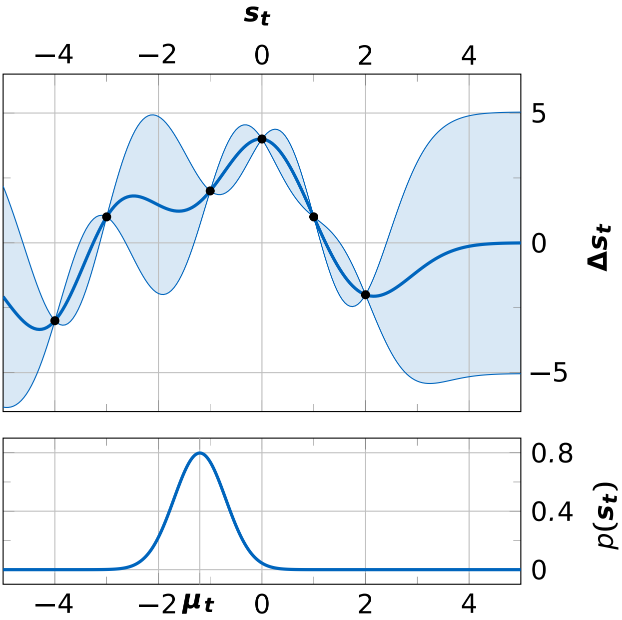


$\frac{\pi}{15}$
$\frac{\pi}{30}$
0
ω_5
$-\frac{\pi}{30}$
$-\frac{\pi}{15}$
ω [rad]
0
5
10
15
Time step

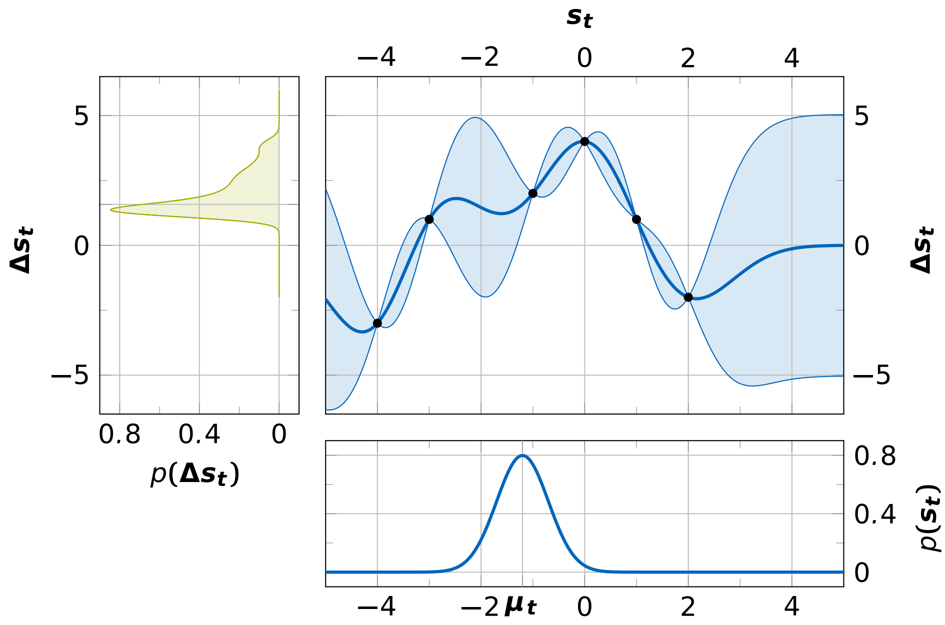
One-Step Predictions



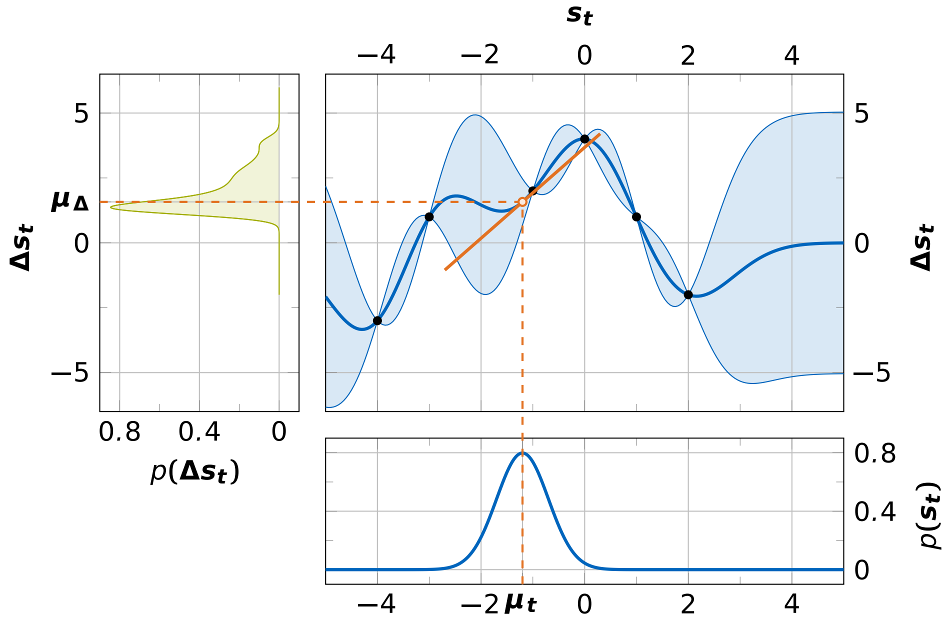
Multi-Step Uncertainties



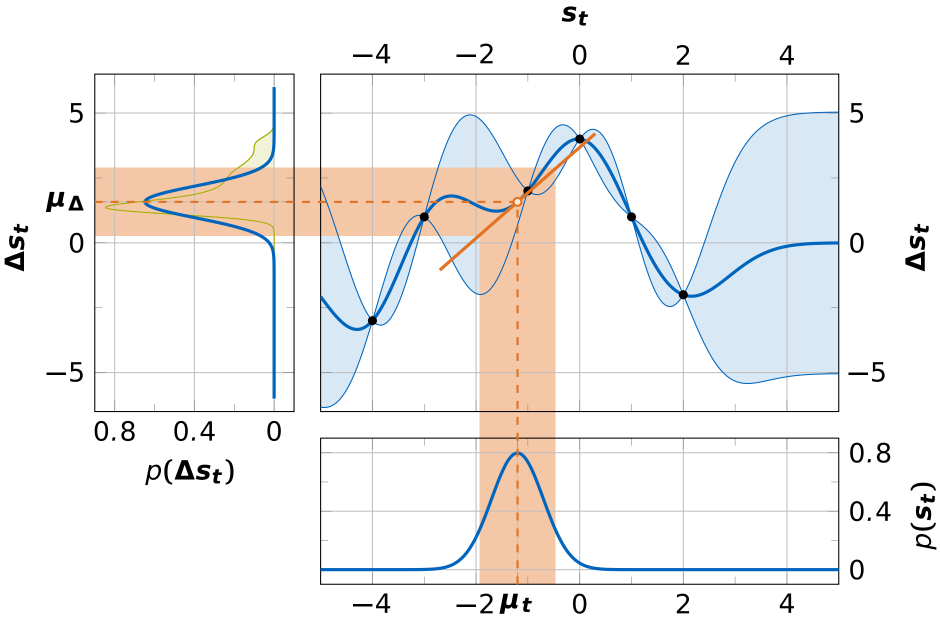
Multi-Step Uncertainties



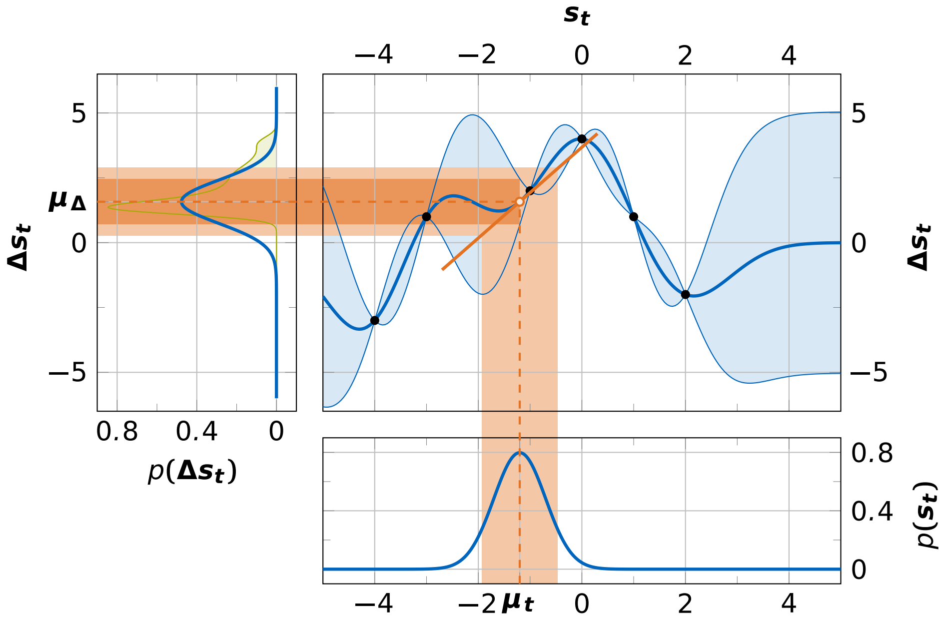
Multi-Step Uncertainties

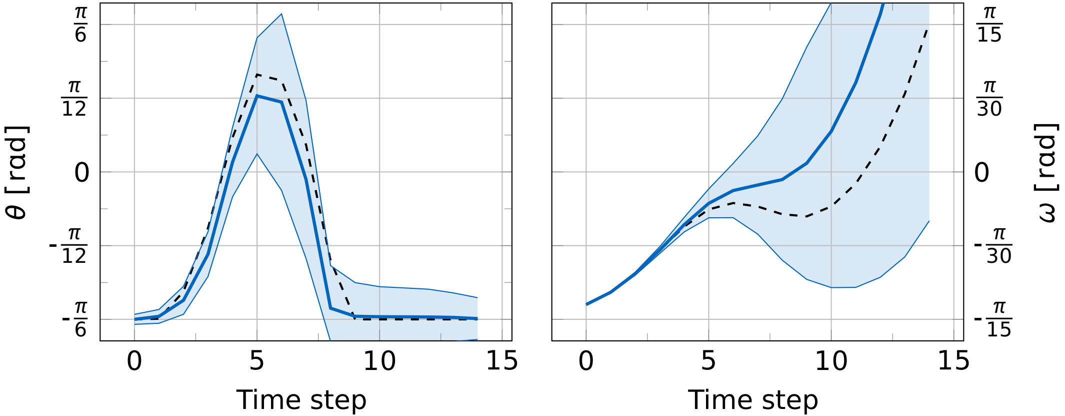


Multi-Step Uncertainties

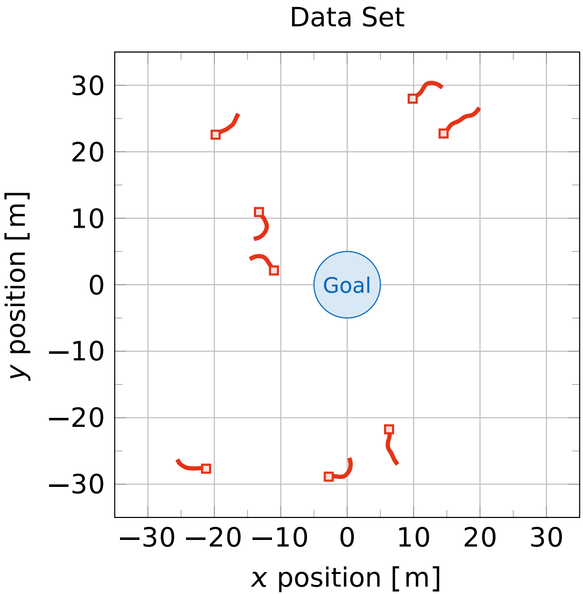


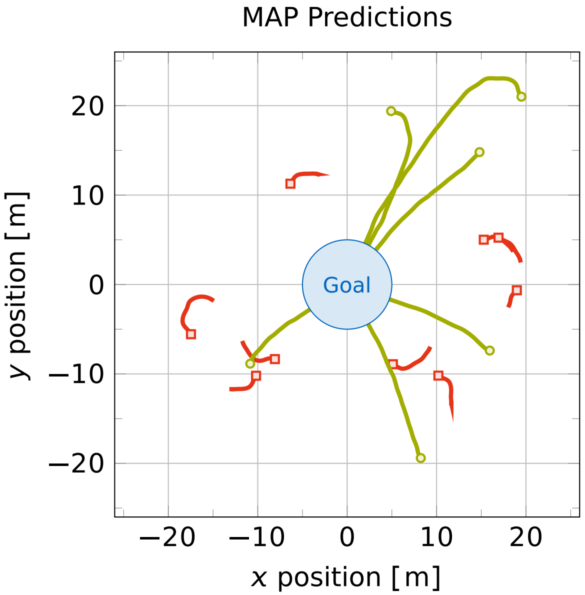
Multi-Step Uncertainties

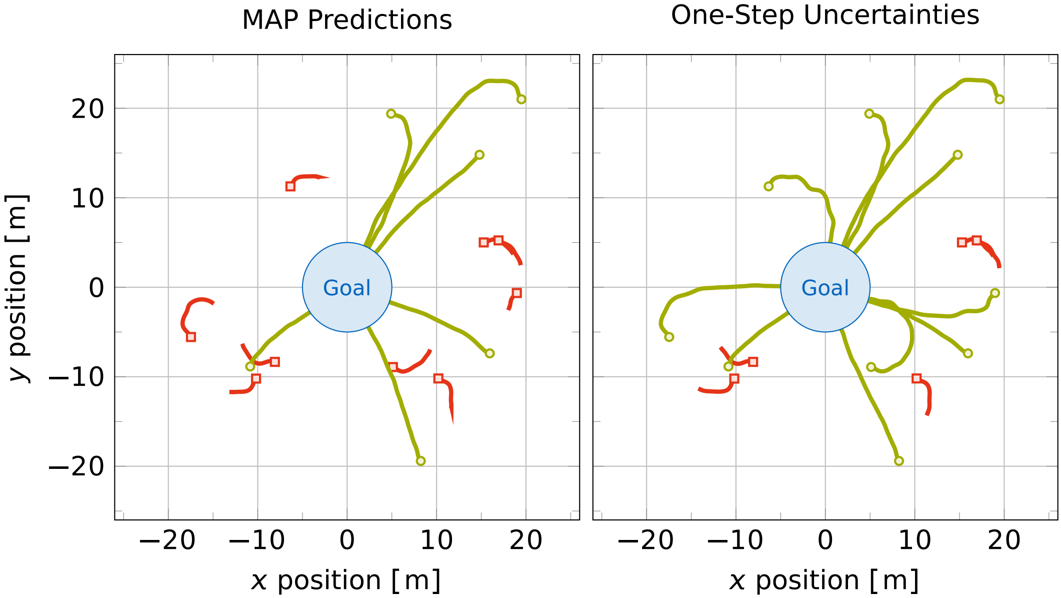




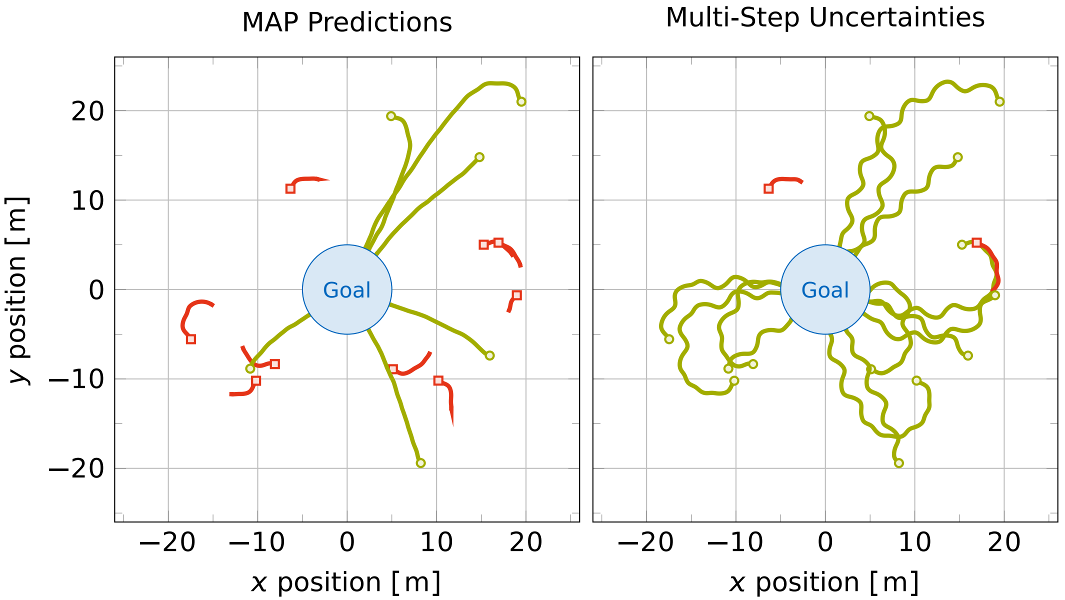
π/6
π/12
0
-π/12
-π/6
θ [rad]
0
5
10
15
Time step
π/15
π/30
0
-π/30
-π/15
ω [rad]
0
5
10
15
Time step


Data Set
Goal
30
20
10
0
−10
−20
−30
−30 −20 −10 0 10 20 30
x position [m]
y position [m]


MAP Predictions
20
10
0
−10
−20
y position [m]
Goal
−20
−10
0
10
20
x position [m]



MAP Predictions
One-Step Uncertainties
Goal
Goal
y position [m]
x position [m]
x position [m]
20
10
0
−10
−20
−20
−10
0
10
20



MAP Predictions
Multi-Step Uncertainties
20
10
0
−10
−20
y position [m]
−20
−10
0
10
20
x position [m]
Goal
Goal
−20
−10
0
10
20
x position [m]

Quantitative Results

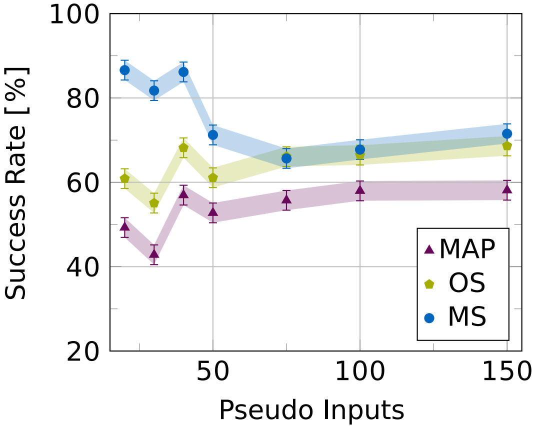
Metric	MAP	OS	MS
Trajectories	9660	9660	9660

MAP Deterministic predictions
OS One-Step Uncertainties
MS Multi-Step Uncertainties

Quantitative Results

Metric	MAP	OS	MS
Trajectories	9660	9660	9660
Success Rate	53.4 %	63.8 %	**75.8 %**
Time to Goal			
Mean	**59.9**	62.0	66.5
Median	**60**	**60**	63

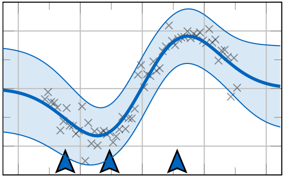
MAP Deterministic predictions
OS One-Step Uncertainties
MS Multi-Step Uncertainties



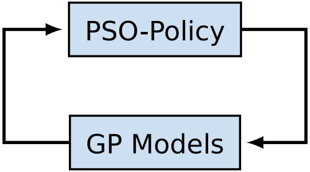
Metric	MAP	OS	MS
Trajectories	9660	9660	9660
Success Rate	53.4 %	63.8 %	**75.8 %**
Time to Goal			
Mean	**59.9**	62.0	66.5
Median	**60**	**60**	63

MAP Deterministic predictions
OS One-Step Uncertainties
MS Multi-Step Uncertainties

- Sparse GPs
  - Scale to large data sets
  - First test at LSY



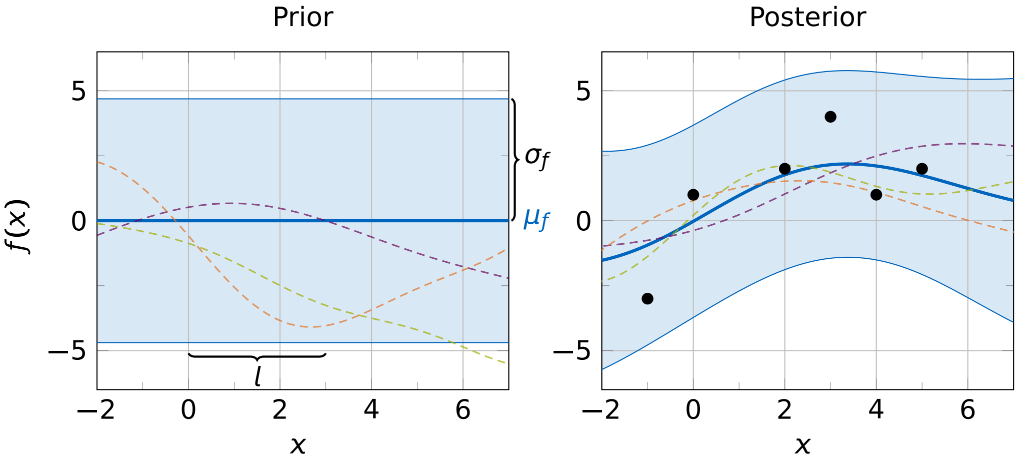
- Combination of GP and PSO
  - Directly exploit models
  - Fully non-parametric



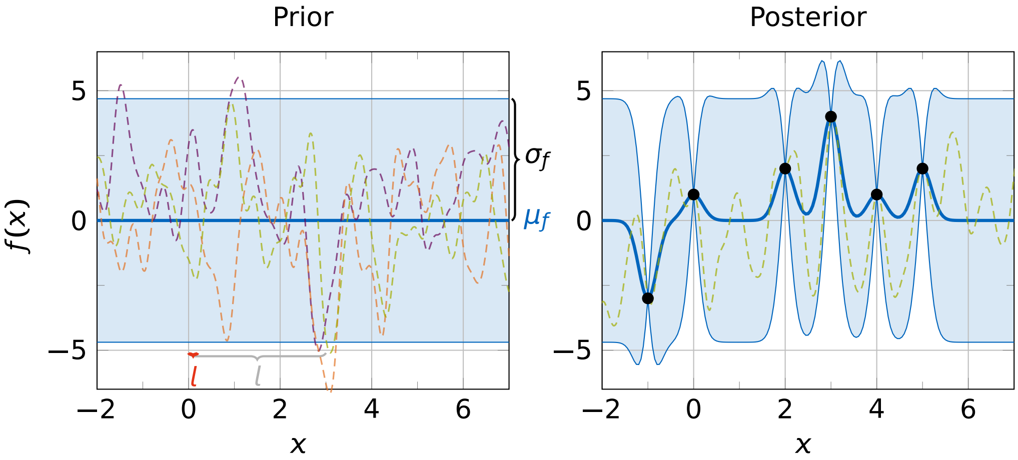
- Bayesian Models in RL
  - Reduce model-bias
  - Improve performance

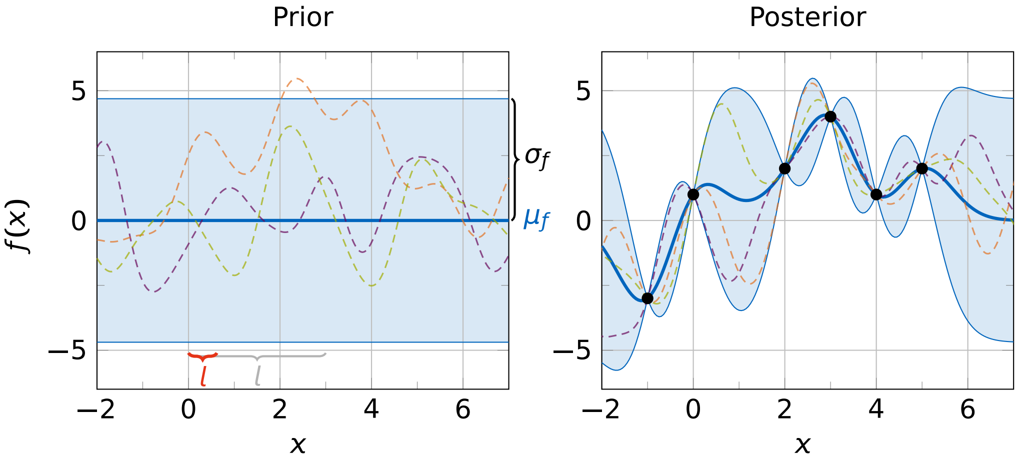
Metric	MAP	OS	MS
Success Rate	53.4 %	63.8 %	**75.8 %**

Additional Material


Prior
Posterior
5
0
−5
−2
0
2
4
6
x
$f(x)$
σ_f
μ_f
l

Hyperparameters and Posterior




Prior
Posterior
5
0
−5
−2
0
2
4
6
x
f(x)
σ_f
μ_f
l
l

Definition (Bicycle Reward Function)

The bicycle reward function is defined as

$$\mathbf{R}_{\text{bicycle}}(\boldsymbol{s}) := \begin{cases} 2 & \text{if goal reached} \\ 0 & \text{if fallen down} \\ c \cdot \mathcal{N}(\Delta^{\psi}_{\boldsymbol{s}} \mid 0, \sigma^2_{\text{angle}}) & \text{otherwise} \end{cases}$$

where $\Delta^{\psi}_{\boldsymbol{s}}$ denotes the angle towards the goal.