Safe Trajectory Sampling in Model-based Reinforcement Learning

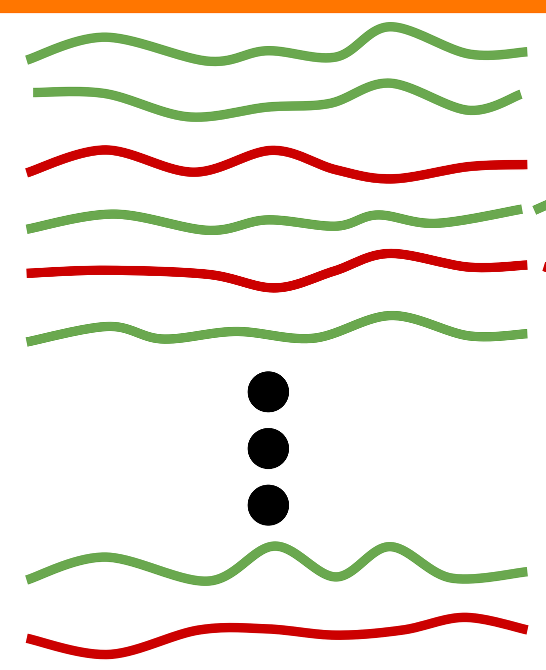
Sicelukwanda Zwane, Denis Hadjivelichkov, Yicheng Luo, Yasemin Bekiroglu, Dimitrios Kanoulas, Marc P. Deisenroth

UCL

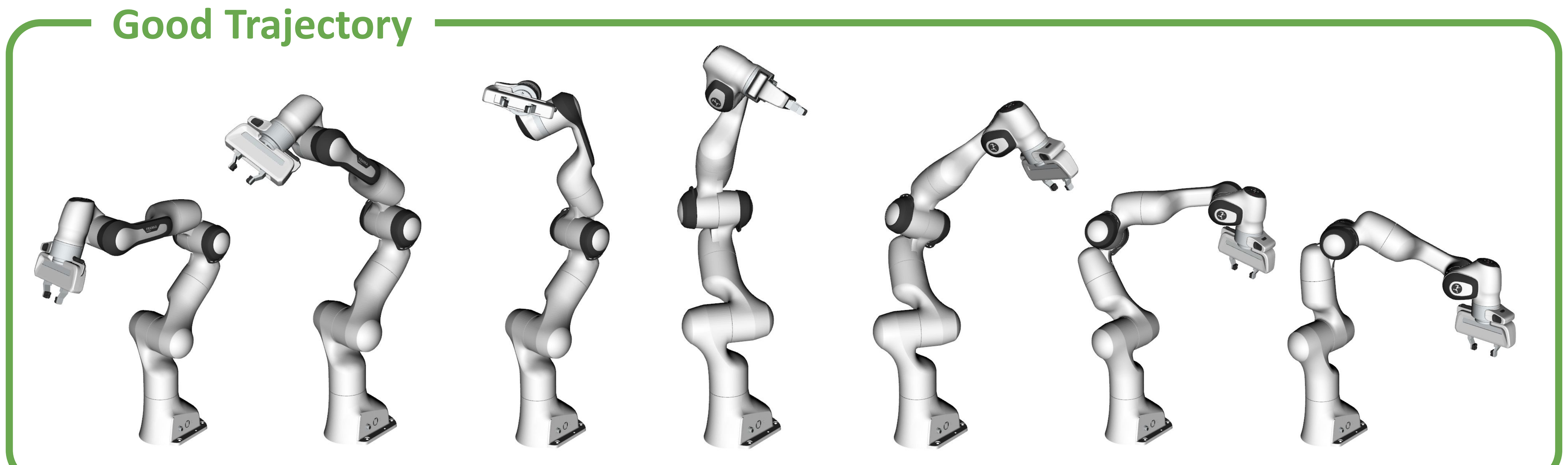
Background: Model-based Reinforcement Learning

Model-based reinforcement learning (MBRL) is a successful strategy for learning complex tasks in robotics. By Leveraging learned probabilistic dynamics models, MBRL can learn robust policies using minimal data. However, such data-driven models can be blind to safety and feasibility constraints present in the real world.

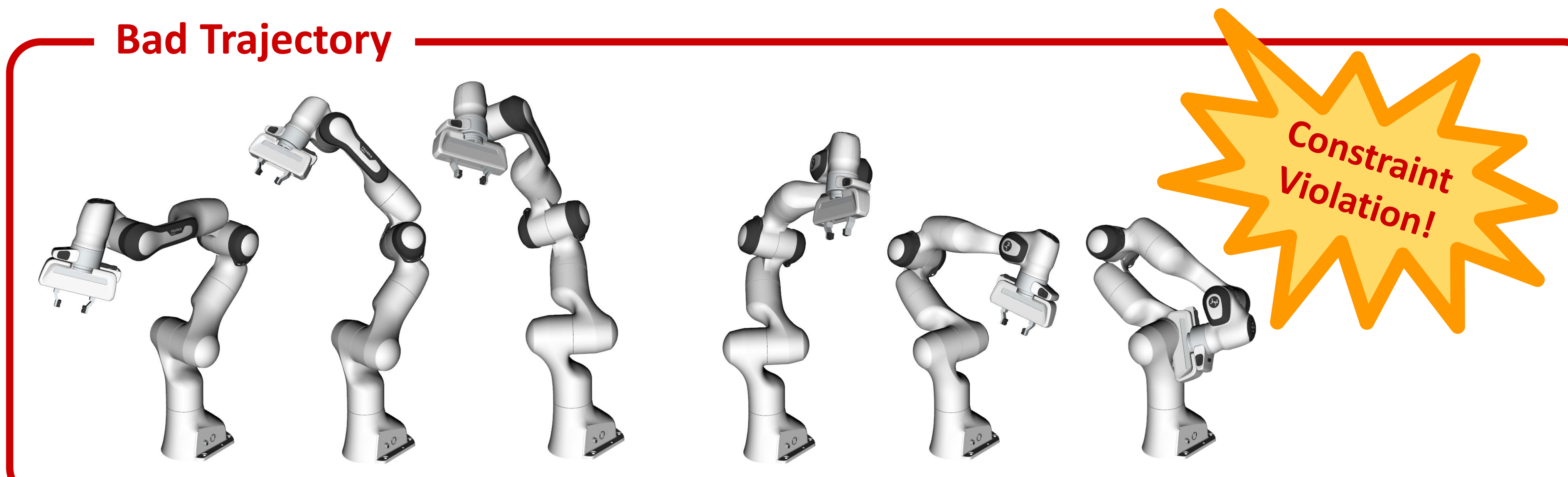
Sampled Trajectories



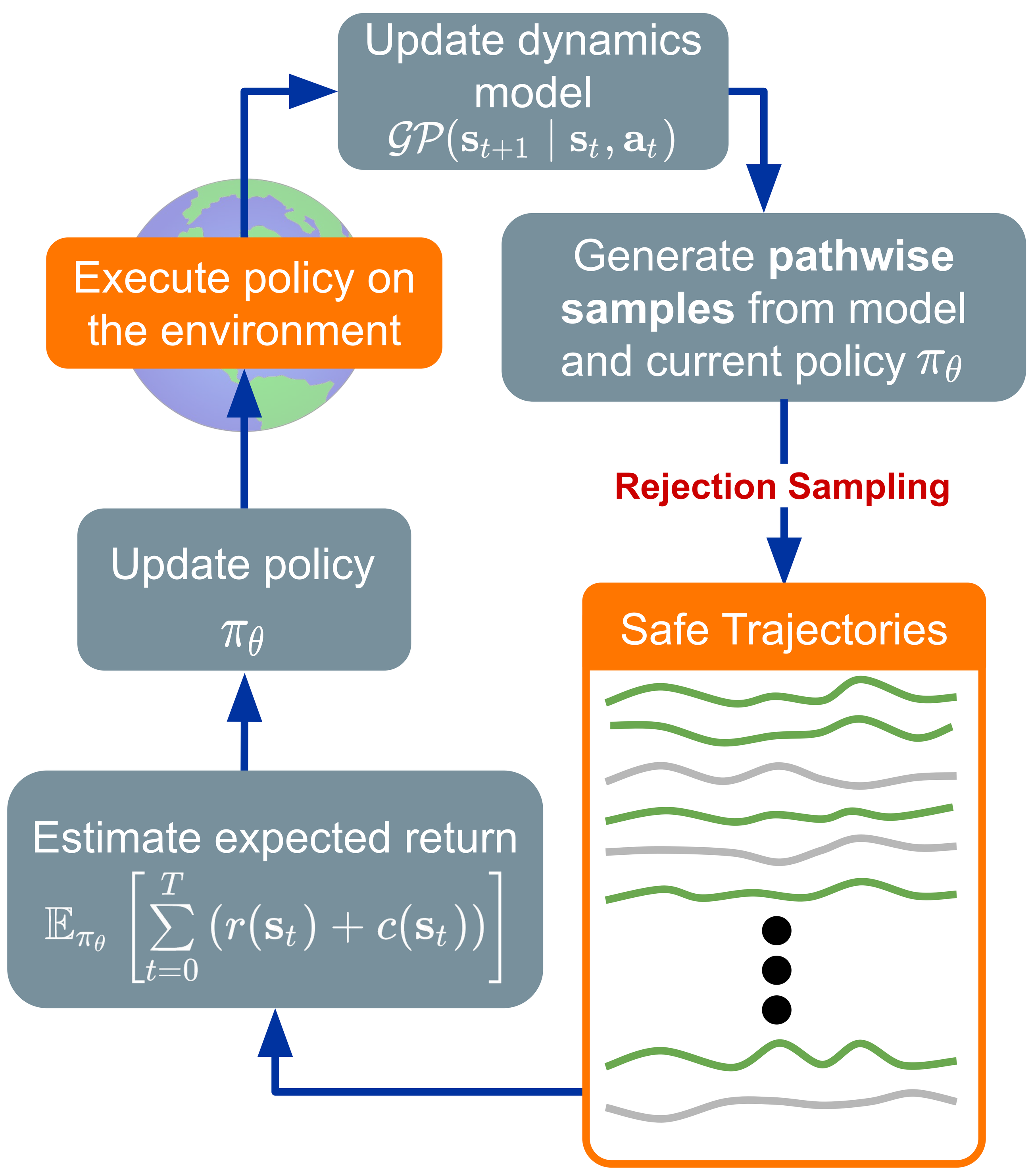
Good Trajectory



Bad Trajectory



Method: Safe Gaussian Process Policy Optimization



Gaussian Process Dynamics Model

- We setup trajectories into a dataset $\{\mathbf{x}_i, \mathbf{y}_i\}$, where $\mathbf{x} = [\mathbf{s}_t \; \mathbf{a}_t]$ and $\mathbf{y} = \mathbf{s}_{t+1} - \mathbf{s}_t$.
- We train the GP parameters on the data by maximising the marginal log likelihood

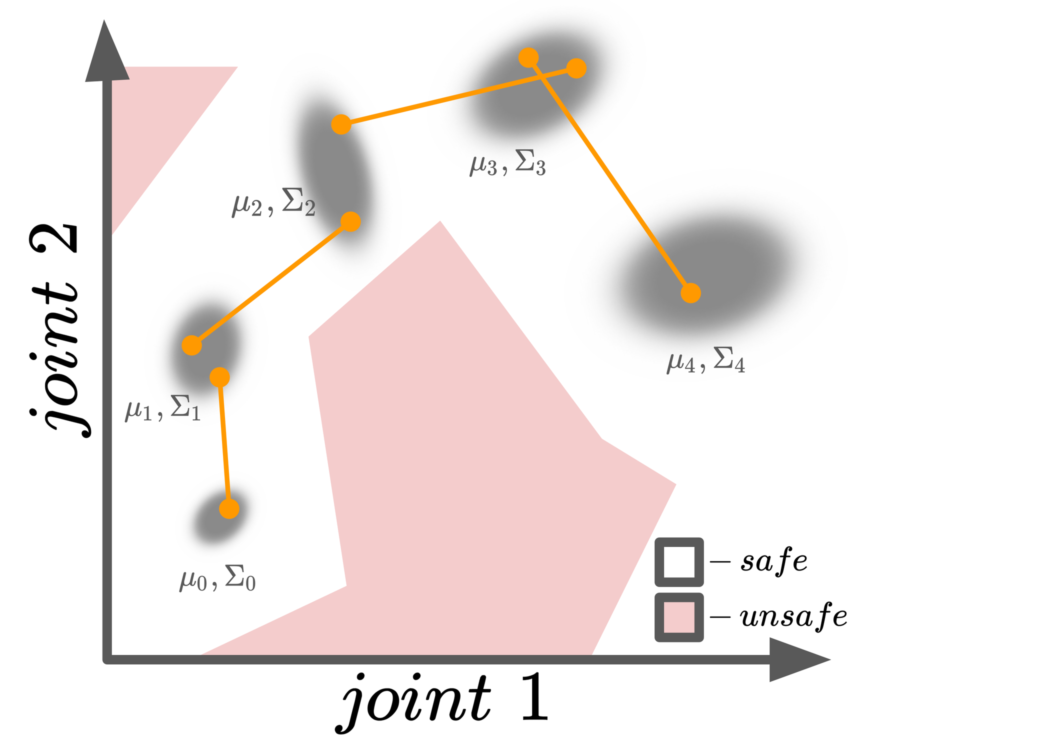
$$\log p(\mathbf{y} \mid \mathbf{X}) = -\frac{1}{2}\mathbf{y}^T(\mathbf{K} + \sigma_\varepsilon^2\mathbf{I})^{-1}\mathbf{y} - \frac{1}{2}\log|\mathbf{K} + \sigma_\varepsilon^2\mathbf{I}| - \frac{N}{2}\log 2\pi.$$

Long-term Predictions (Trajectory Sampling)

Standard Trajectory Sampling

- Iteratively perform random one-step predictions according to

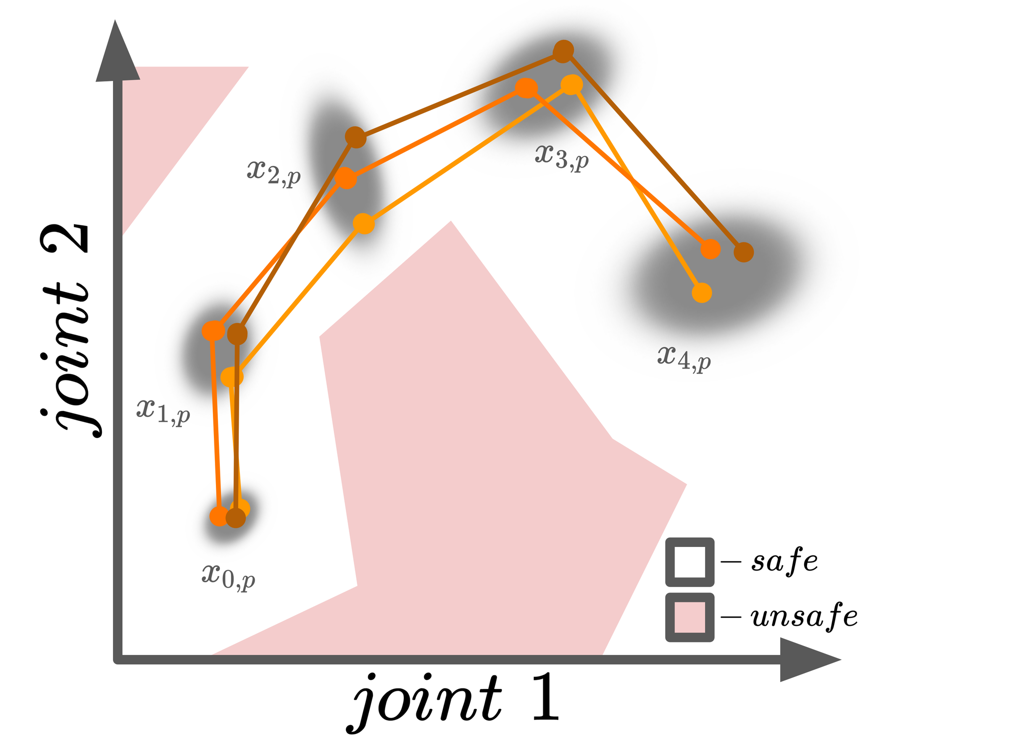
$$\mu^* = k(x^*, X)\big(\mathbf{K} + \sigma_\varepsilon^2\mathbf{I}\big)^{-1}\mathbf{y}$$
$$\Sigma^* = k(x^*, x^*) - k(x^*, X)\big(\mathbf{K} + \sigma_\varepsilon^2\mathbf{I}\big)^{-1}k(X, x^*)$$



Pathwise Trajectory Sampling

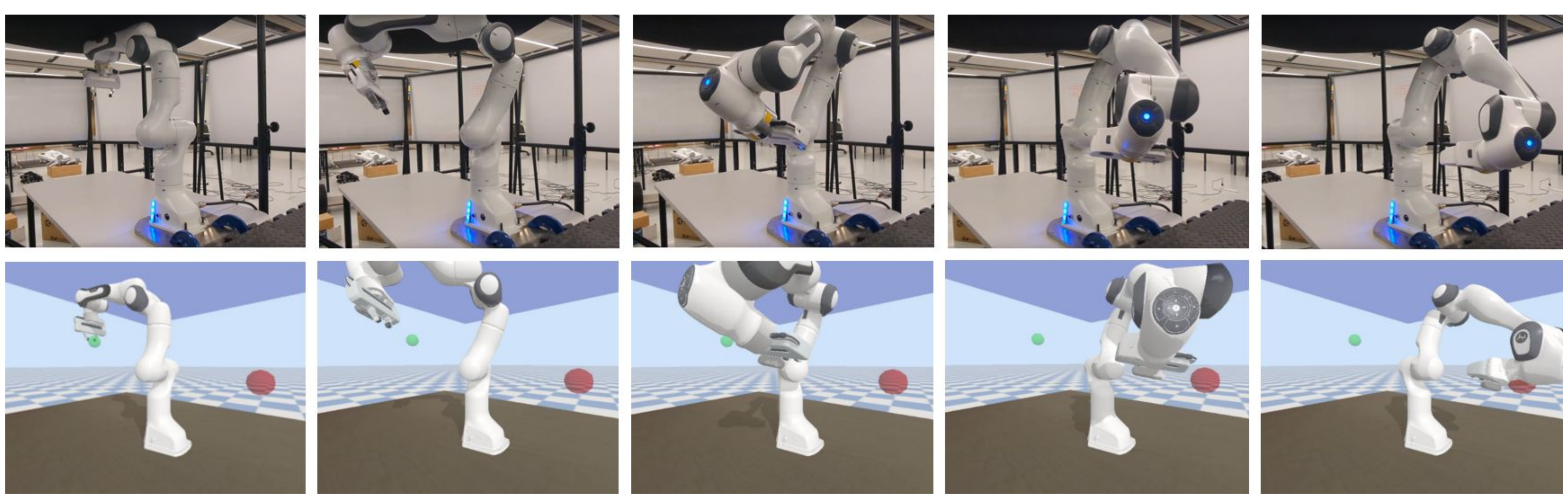
- Sample deterministic "paths" from prior and transform into posterior samples

$$\underbrace{(f \mid \mathbf{X}, \mathbf{y})(\cdot)}_{\text{posterior}} \stackrel{d}{=} \underbrace{f(\cdot)}_{\text{prior}} + \underbrace{k(\cdot, \mathbf{X})(\mathbf{K} + \sigma_\varepsilon^2\mathbf{I})^{-1}(\mathbf{y} - f(\mathbf{X}))}_{\text{data-dependent update}}.$$

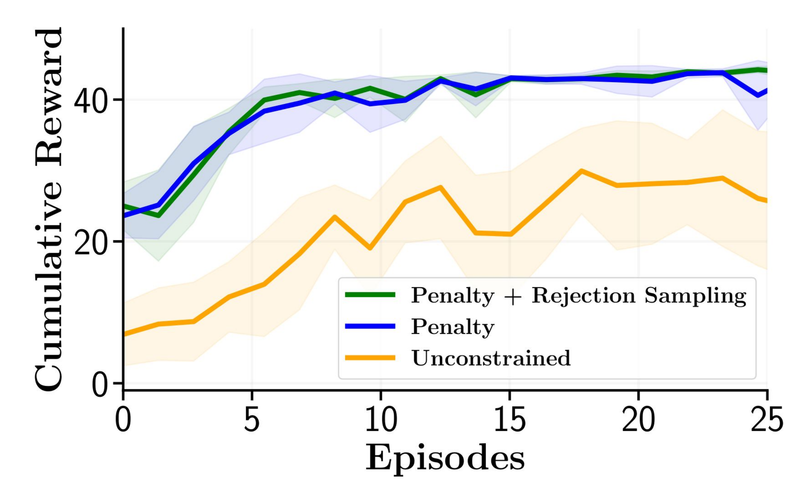


Experiments: Constrained Manipulation

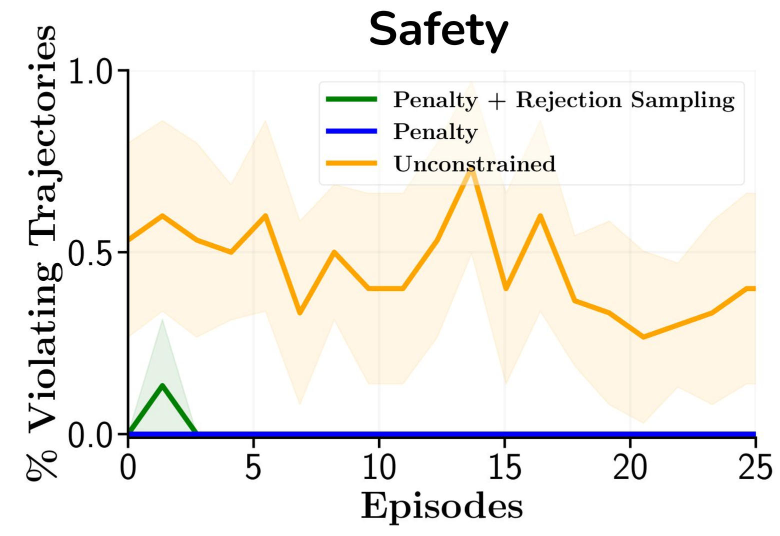
We evaluate our method by training a policy on a simulated reach task where the robot is constrained along the Z-axis by an overhead obstacle. The resulting policy is deployed *sim-to-real* on a physical robot with no further fine tuning.



Performance



Safety



UCL Centre for Artificial Intelligence
UKRI Engineering and Physical Sciences Research Council

sicelukwanda.zwane.20@ucl.ac.uk