RL and MPC
Safety, Stability, and some more recent results

Sébastien Gros

Dept. of Cybernetic, NTNU
Faculty of Information Tech.

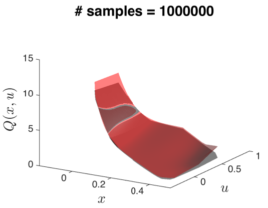
Freiburg PhD School

Outline

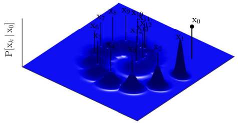
1. Safe RL via MPC
2. Stability-constrained Learning with MPC
3. Some more results (in brief)
4. Applications & Reflections

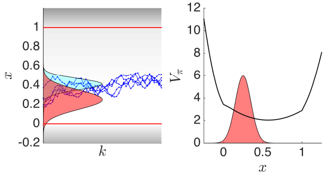
What are we going to discuss?

1. Learning for MPC - A focus on closed-loop performance
2. Safety & stability in Learning for MPC
3. MPC and Markov Decision Processes - When is learning beneficial?



$$Q_{+}(\mathbf{x},\mathbf{u}) \leftarrow L(\mathbf{x},\mathbf{u}) + \gamma \mathbb{E}\left[V\left(\mathbf{x}_{+}\right) \mid \mathbf{x},\mathbf{u}\right]$$





Outline

1. Safe RL via MPC
2. Stability-constrained Learning with MPC
3. Some more results (in brief)
4. Applications & Reflections

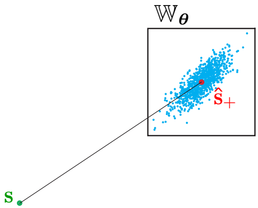
Robust MPC - Uncertainty model

$$\text{True system:} \quad \mathbf{s}_+ \sim \mathbb{P}\left[\,\cdot\,|\mathbf{s},\mathbf{a}\,\right]$$
$$\text{Deterministic model:} \quad \hat{\mathbf{s}}_+ = \mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right)$$

Robust MPC - Uncertainty model

True system: $\mathbf{s}_+ \sim \mathbb{P}\left[\,\cdot\,|\mathbf{s},\mathbf{a}\,\right]$

Deterministic model: $\hat{\mathbf{s}}_+ = \mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right)$



Dispersion: $\mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right) + \mathbb{W}_{\boldsymbol{\theta}}$ contains the support of $\mathbb{P}\left[\,\cdot\,|\mathbf{s},\mathbf{a}\,\right]$, i.e.

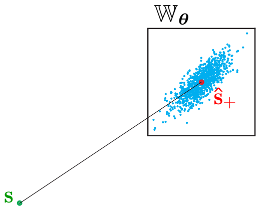
$$\mathbf{s}_+ \in \mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right) + \mathbb{W}_{\boldsymbol{\theta}} \tag{1}$$

with probability 1

Robust MPC - Uncertainty model

True system: $\mathbf{s}_+ \sim \mathbb{P}\left[\,\cdot\,|\mathbf{s},\mathbf{a}\,\right]$

Deterministic model: $\hat{\mathbf{s}}_+ = \mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right)$



Dispersion: $\mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right) + \mathbb{W}_{\boldsymbol{\theta}}$ contains the support of $\mathbb{P}\left[\,\cdot\,|\mathbf{s},\mathbf{a}\,\right]$, i.e.

$$\mathbf{s}_+ \in \mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right) + \mathbb{W}_{\boldsymbol{\theta}} \tag{1}$$

with probability 1

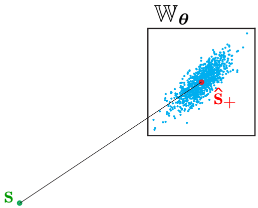
Remarks:

- Identifying $\mathbb{W}_{\boldsymbol{\theta}}$ is a set-membership identification problem, well studied
- Obviously $\mathbb{W}_{\boldsymbol{\theta}}$ is not unique
- Ensuring probability 1 from data is impossible $\rightarrow$ probabilistic guarantees
- Model parameters $\boldsymbol{\theta}$ must be such that (1) holds on every known data point

Robust MPC - Uncertainty model

True system: $\mathbf{s}_+ \sim \mathbb{P}\left[\,\cdot\,|\mathbf{s},\mathbf{a}\,\right]$

Deterministic model: $\hat{\mathbf{s}}_+ = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{s},\mathbf{a})$



Dispersion: $\mathbf{f}_{\boldsymbol{\theta}}(\mathbf{s},\mathbf{a}) + \mathbb{W}_{\boldsymbol{\theta}}$ contains the support of $\mathbb{P}\left[\,\cdot\,|\mathbf{s},\mathbf{a}\,\right]$, i.e.

$$\mathbf{s}_+ \in \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{s},\mathbf{a}) + \mathbb{W}_{\boldsymbol{\theta}} \tag{1}$$

with probability 1

Condition

$$\mathbf{s}_+ - \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{s},\mathbf{a}) \in \mathbb{W}_{\boldsymbol{\theta}}$$

for all observed triplets $(\mathbf{s},\mathbf{a},\mathbf{s}_+)$

$\rightarrow$ constraints on $\boldsymbol{\theta}$

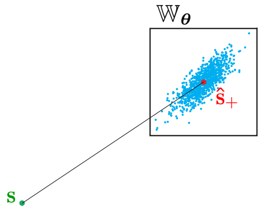
Remarks:

- Identifying $\mathbb{W}_{\boldsymbol{\theta}}$ is a set-membership identification problem, well studied
- Obviously $\mathbb{W}_{\boldsymbol{\theta}}$ is not unique
- Ensuring probability 1 from data is impossible $\rightarrow$ probabilistic guarantees
- Model parameters $\boldsymbol{\theta}$ must be such that (1) holds on every known data point

Robust MPC - Uncertainty model

True system: $\mathbf{s}_+ \sim \mathbb{P}\left[\,\cdot\,|\mathbf{s},\mathbf{a}\,\right]$

Deterministic model: $\hat{\mathbf{s}}_+ = \mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right)$



Dispersion: $\mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right) + \mathbb{W}_{\boldsymbol{\theta}}$ contains the support of $\mathbb{P}\left[\,\cdot\,|\mathbf{s},\mathbf{a}\,\right]$, i.e.

$$\mathbf{s}_+ \in \mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right) + \mathbb{W}_{\boldsymbol{\theta}} \tag{1}$$

with probability 1

Condition

$$\mathbf{s}_+ - \mathbf{f}_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right) \in \mathbb{W}_{\boldsymbol{\theta}}$$

for all observed triplets $\left(\mathbf{s},\mathbf{a},\mathbf{s}_+\right)$

$\rightarrow$ constraints on $\boldsymbol{\theta}$

Remarks:

- Identifying $\mathbb{W}_{\boldsymbol{\theta}}$ is a set-membership identification problem, well studied
- Obviously $\mathbb{W}_{\boldsymbol{\theta}}$ is not unique
- Ensuring probability 1 from data is impossible $\rightarrow$ probabilistic guarantees
- Model parameters $\boldsymbol{\theta}$ must be such that (1) holds on every known data point

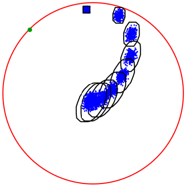
Containing the model-system mismatch becomes constraints in the parameters $\boldsymbol{\theta}$. Constraints can be readily formulated in terms of data.

Safe policies via Robust (N)MPC

Robust (N)MPC delivers policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{x}_0) = \mathbf{u}_0^\star$ from

$$\mathbf{u}^\star = \arg\min_{\mathbf{u}} \max_{\mathbf{w}\in\mathbb{W}_{\boldsymbol{\theta}}{}^N} T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k)$$

$$\text{s.t.} \quad \mathbf{u}_{0,\dots,N} \in \mathbb{U}$$

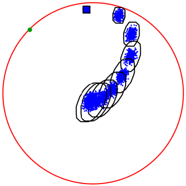


- $\mathbf{x}_{0,\dots,N}$ is the propagation of the state dispersion
- max cost treats worst-case scenario, required for "classic" stability
- $\mathbf{w} = \{\mathbf{w}_0, \dots, \mathbf{w}_N\}$ is the disturbance with $\mathbf{w}_k \in \mathbb{W}_{\boldsymbol{\theta}}$

Safe policies via Robust (N)MPC

Robust (N)MPC delivers policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{x}_0) = \mathbf{u}_0^\star$ from

$$\begin{aligned} \mathbf{u}^\star = \arg\min_{\mathbf{u}} \ & \max_{\mathbf{w}\in\mathbb{W}_{\boldsymbol{\theta}}^N} \ T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \ & \mathbf{u}_{0,\dots,N} \in \mathbb{U} \\ & \mathbf{x}_{1,\dots,N-1}(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}) \in \mathbb{X}, \quad \forall\, \mathbf{w} \in \mathbb{W}_{\boldsymbol{\theta}}^{N-1} \end{aligned}$$

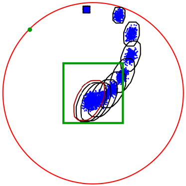


- $\mathbf{x}_{0,\dots,N}$ is the propagation of the state dispersion
- max cost treats worst-case scenario, required for “classic” stability
- $\mathbf{w} = \{\mathbf{w}_0, \dots, \mathbf{w}_N\}$ is the disturbance with $\mathbf{w}_k \in \mathbb{W}_{\boldsymbol{\theta}}$
- $\mathbf{x}_{1,\dots,N-1}(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w})$ are the trajectories subject to $\mathbf{w}$ and $\mathbf{f}_{\boldsymbol{\theta}}$
- $\mathbb{X}$ is the “safe” set where the state should be at all time

Safe policies via Robust (N)MPC

Robust (N)MPC delivers policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{x}_0) = \mathbf{u}_0^\star$ from

$$\begin{aligned}\mathbf{u}^\star = \arg\min_{\mathbf{u}} \quad & \max_{\mathbf{w}\in\mathbb{W}_{\boldsymbol{\theta}}{}^N} \quad T_{\boldsymbol{\theta}}\left(\mathbf{x}_N\right) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}\left(\mathbf{x}_k, \mathbf{u}_k\right) \\ \text{s.t.} \quad & \mathbf{u}_{0,\ldots,N} \in \mathbb{U} \\ & \mathbf{x}_{1,\ldots,N-1}\left(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}\right) \in \mathbb{X}, \quad \forall\, \mathbf{w} \in \mathbb{W}_{\boldsymbol{\theta}}{}^{N-1} \\ & \mathbf{x}_N\left(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}\right) \in \mathbb{T}_{\boldsymbol{\theta}}, \quad \forall\, \mathbf{w} \in \mathbb{W}_{\boldsymbol{\theta}}{}^{N-1}\end{aligned}$$



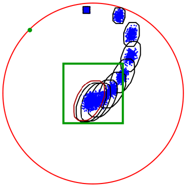
- $\mathbf{x}_{0,\ldots,N}$ is the propagation of the state dispersion
- max cost treats worst-case scenario, required for "classic" stability
- $\mathbf{w} = \{\mathbf{w}_0, \ldots, \mathbf{w}_N\}$ is the disturbance with $\mathbf{w}_k \in \mathbb{W}_{\boldsymbol{\theta}}$
- $\mathbf{x}_{1,\ldots,N-1}\left(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}\right)$ are the trajectories subject to $\mathbf{w}$ and $\mathbf{f}_{\boldsymbol{\theta}}$
- $\mathbb{X}$ is the "safe" set where the state should be at all time
- Terminal set $\mathbb{T}_{\boldsymbol{\theta}}$ (required for recursive feasibility & stability)

Safe policies via Robust (N)MPC

Robust (N)MPC delivers policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{x}_0) = \mathbf{u}_0^\star$ from

$$\mathbf{u}^\star = \arg\min_{\mathbf{u}} \max_{\mathbf{w}\in\mathbb{W}_{\boldsymbol{\theta}}{}^N} T_{\boldsymbol{\theta}}\left(\mathbf{x}_N\right) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}\left(\mathbf{x}_k, \mathbf{u}_k\right)$$

$$\begin{aligned} \text{s.t.}\quad & \mathbf{u}_{0,\ldots,N} \in \mathbb{U} \\ & \mathbf{x}_{1,\ldots,N-1}\left(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}\right) \in \mathbb{X}, \quad \forall\, \mathbf{w} \in \mathbb{W}_{\boldsymbol{\theta}}{}^{N-1} \\ & \mathbf{x}_N\left(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}\right) \in \mathbb{T}_{\boldsymbol{\theta}}, \quad \forall\, \mathbf{w} \in \mathbb{W}_{\boldsymbol{\theta}}{}^{N-1} \end{aligned}$$

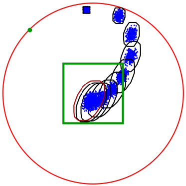


- $\mathbf{x}_{0,\ldots,N}$ is the propagation of the state dispersion
- max cost treats worst-case scenario, required for "classic" stability
- $\mathbf{w} = \{\mathbf{w}_0, \ldots, \mathbf{w}_N\}$ is the disturbance with $\mathbf{w}_k \in \mathbb{W}_{\boldsymbol{\theta}}$
- $\mathbf{x}_{1,\ldots,N-1}\left(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}\right)$ are the trajectories subject to $\mathbf{w}$ and $\mathbf{f}_{\boldsymbol{\theta}}$
- $\mathbb{X}$ is the "safe" set where the state should be at all time
- Terminal set $\mathbb{T}_{\boldsymbol{\theta}}$ (required for recursive feasibility & stability)
- If $\boldsymbol{\theta}$ is such that $\mathbb{W}_{\boldsymbol{\theta}}$ encloses state dispersion, **MPC yields safe policy**

Safe policies via Robust (N)MPC

Robust (N)MPC delivers policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{x}_0) = \mathbf{u}_0^\star$ from

$$\begin{aligned}\mathbf{u}^\star = \arg\min_{\mathbf{u}} \max_{\mathbf{w}\in\mathbb{W}_{\boldsymbol{\theta}}{}^N} \quad & T_{\boldsymbol{\theta}}\left(\mathbf{x}_N\right) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}\left(\mathbf{x}_k, \mathbf{u}_k\right) \\ \text{s.t.} \quad & \mathbf{u}_{0,\dots,N} \in \mathbb{U} \\ & \mathbf{x}_{1,\dots,N-1}\left(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}\right) \in \mathbb{X}, \quad \forall\, \mathbf{w} \in \mathbb{W}_{\boldsymbol{\theta}}{}^{N-1} \\ & \mathbf{x}_N\left(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}\right) \in \mathbb{T}_{\boldsymbol{\theta}}, \quad \forall\, \mathbf{w} \in \mathbb{W}_{\boldsymbol{\theta}}{}^{N-1}\end{aligned}$$



- $\mathbf{x}_{0,\dots,N}$ is the propagation of the state dispersion
- max cost treats worst-case scenario, required for “classic” stability
- $\mathbf{w} = \{\mathbf{w}_0, \dots, \mathbf{w}_N\}$ is the disturbance with $\mathbf{w}_k \in \mathbb{W}_{\boldsymbol{\theta}}$
- $\mathbf{x}_{1,\dots,N-1}\left(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}\right)$ are the trajectories subject to $\mathbf{w}$ and $\mathbf{f}_{\boldsymbol{\theta}}$
- $\mathbb{X}$ is the “safe” set where the state should be at all time
- Terminal set $\mathbb{T}_{\boldsymbol{\theta}}$ (required for recursive feasibility & stability)
- If $\boldsymbol{\theta}$ is such that $\mathbb{W}_{\boldsymbol{\theta}}$ encloses state dispersion, **MPC yields safe policy**

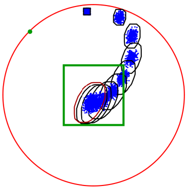
Closed-loop stability under some conditions on $\boldsymbol{\theta}$ (not trivial), need $\gamma = 1$ (for now)

Safe policies via Robust (N)MPC

Robust (N)MPC delivers policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{x}_0) = \mathbf{u}_0^\star$ from

$$\mathbf{u}^\star = \arg\min_{\mathbf{u}} \max_{\mathbf{w}\in\mathbb{W}_{\boldsymbol{\theta}}^N} T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k)$$

$$\begin{aligned} \text{s.t.} \quad & \mathbf{u}_{0,\dots,N} \in \mathbb{U} \\ & \mathbf{x}_{1,\dots,N-1}(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}) \in \mathbb{X}, \quad \forall\, \mathbf{w} \in \mathbb{W}_{\boldsymbol{\theta}}^{N-1} \\ & \mathbf{x}_N(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w}) \in \mathbb{T}_{\boldsymbol{\theta}}, \quad \forall\, \mathbf{w} \in \mathbb{W}_{\boldsymbol{\theta}}^{N-1} \end{aligned}$$



$$\nabla_{\boldsymbol{\theta}} J = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{a}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

- $\mathbf{x}_{0,\dots,N}$ is the propagation of the state dispersion
- max cost treats worst-case scenario, required for “classic” stability
- $\mathbf{w} = \{\mathbf{w}_0, \dots, \mathbf{w}_N\}$ is the disturbance with $\mathbf{w}_k \in \mathbb{W}_{\boldsymbol{\theta}}$
- $\mathbf{x}_{1,\dots,N-1}(\mathbf{u}, \mathbf{x}_0, \boldsymbol{\theta}, \mathbf{w})$ are the trajectories subject to $\mathbf{w}$ and $\mathbf{f}_{\boldsymbol{\theta}}$
- $\mathbb{X}$ is the “safe” set where the state should be at all time
- Terminal set $\mathbb{T}_{\boldsymbol{\theta}}$ (required for recursive feasibility & stability)
- If $\boldsymbol{\theta}$ is such that $\mathbb{W}_{\boldsymbol{\theta}}$ encloses state dispersion, **MPC yields safe policy**

Closed-loop stability under some conditions on $\boldsymbol{\theta}$ (not trivial), need $\gamma = 1$ (for now)

Safe Learning via Robust MPC

Robust NMPC parameters $\boldsymbol{\theta}$

Policy gradient

$$\nabla_{\boldsymbol{\theta}} J = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{u}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

adjusts $\boldsymbol{\theta}$ for performance

Condition

$$\mathbf{s}_{+} - \mathbf{f}\left(\mathbf{s}, \mathbf{a}, \boldsymbol{\theta}\right) \in \mathbb{W}_{\boldsymbol{\theta}}$$

enforces safety through $\boldsymbol{\theta}$

Safe Learning via Robust MPC

Robust NMPC parameters $\boldsymbol{\theta}$

Policy gradient

$$\nabla_{\boldsymbol{\theta}} J = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{u}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

adjusts $\boldsymbol{\theta}$ for performance

Condition

$$\mathbf{s}_{+} - \mathbf{f}\left(\mathbf{s}, \mathbf{a}, \boldsymbol{\theta}\right) \in \mathbb{W}_{\boldsymbol{\theta}}$$

enforces safety through $\boldsymbol{\theta}$

- No clear connection to SYSID
- Sometimes does opposite of SYSID

Safe Learning via Robust MPC

Robust NMPC parameters θ

Policy gradient

$$\nabla_{\boldsymbol{\theta}} J = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{u}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

adjusts $\boldsymbol{\theta}$ for performance

- No clear connection to SYSID
- Sometimes does opposite of SYSID

Condition

$$\mathbf{s}_{+} - \mathbf{f}\left(\mathbf{s}, \mathbf{a}, \boldsymbol{\theta}\right) \in \mathbb{W}_{\boldsymbol{\theta}}$$

enforces safety through $\boldsymbol{\theta}$

- Can be interpreted as a form of SYSID (see set-membership)

Safe Learning via Robust MPC

Robust NMPC parameters $\boldsymbol{\theta}$

Policy gradient

$$\nabla_{\boldsymbol{\theta}} J = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{u}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

adjusts $\boldsymbol{\theta}$ for performance

- No clear connection to SYSID
- Sometimes does opposite of SYSID

Condition

$$\mathbf{s}_{+} - \mathbf{f}\left(\mathbf{s}, \mathbf{a}, \boldsymbol{\theta}\right) \in \mathbb{W}_{\boldsymbol{\theta}}$$

enforces safety through $\boldsymbol{\theta}$

- Can be interpreted as a form of SYSID (see set-membership)

How to do Safe RL?

Classic RL steps: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} J$

Safe Learning via Robust MPC

Robust NMPC parameters $\boldsymbol{\theta}$

Policy gradient

$$\nabla_{\boldsymbol{\theta}} J = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{u}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

adjusts $\boldsymbol{\theta}$ for performance

- No clear connection to SYSID
- Sometimes does opposite of SYSID

Condition

$$\mathbf{s}_+ - \mathbf{f}\left(\mathbf{s}, \mathbf{a}, \boldsymbol{\theta}\right) \in \mathbb{W}_{\boldsymbol{\theta}}$$

enforces safety through $\boldsymbol{\theta}$

- Can be interpreted as a form of SYSID (see set-membership)

How to do Safe RL?

Classic RL steps: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} J$

Also reads as:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$$

$$\Delta\boldsymbol{\theta} = \arg\min_{\Delta\boldsymbol{\theta}} \frac{1}{2\alpha} \left\|\Delta\boldsymbol{\theta}\right\|^2 + \nabla_{\boldsymbol{\theta}} J^\top \Delta\boldsymbol{\theta}$$

Safe Learning via Robust MPC

Robust NMPC parameters $\boldsymbol{\theta}$

Policy gradient

$$\nabla_{\boldsymbol{\theta}} J = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{u}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

adjusts $\boldsymbol{\theta}$ for performance

- No clear connection to SYSID
- Sometimes does opposite of SYSID

Condition

$$\mathbf{s}_+ - \mathbf{f}\left(\mathbf{s}, \mathbf{a}, \boldsymbol{\theta}\right) \in \mathbb{W}_{\boldsymbol{\theta}}$$

enforces safety through $\boldsymbol{\theta}$

- Can be interpreted as a form of SYSID (see set-membership)

How to do Safe RL?

Classic RL steps: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} J$

Also reads as:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$$

$$\Delta\boldsymbol{\theta} = \arg\min_{\Delta\boldsymbol{\theta}} \frac{1}{2\alpha} \left\|\Delta\boldsymbol{\theta}\right\|^2 + \nabla_{\boldsymbol{\theta}} J^\top \Delta\boldsymbol{\theta}$$

Safe RL steps $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$:

$$\begin{aligned} \Delta\boldsymbol{\theta} = \arg\min_{\Delta\boldsymbol{\theta}} \ & \frac{1}{2\alpha} \left\|\Delta\boldsymbol{\theta}\right\|^2 + \nabla_{\boldsymbol{\theta}} J^\top \Delta\boldsymbol{\theta} \\ \text{s.t.} \ & \mathbf{s}_+ - \mathbf{f}\left(\mathbf{s}, \mathbf{a}, \boldsymbol{\theta} + \Delta\boldsymbol{\theta}\right) \in \mathbb{W}_{\boldsymbol{\theta}+\Delta\boldsymbol{\theta}} \\ & \forall \left(\mathbf{s}, \mathbf{a}, \mathbf{s}_+\right) \ \text{in data set} \end{aligned}$$

Safe Learning via Robust MPC

Robust NMPC parameters $\boldsymbol{\theta}$

Policy gradient

$$\nabla_{\boldsymbol{\theta}} J = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{u}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

adjusts $\boldsymbol{\theta}$ for performance

- No clear connection to SYSID
- Sometimes does opposite of SYSID

Condition

$$\mathbf{s}_{+} - \mathbf{f}\left(\mathbf{s}, \mathbf{a}, \boldsymbol{\theta}\right) \in \mathbb{W}_{\boldsymbol{\theta}}$$

enforces safety through $\boldsymbol{\theta}$

- Can be interpreted as a form of SYSID (see set-membership)

How to do Safe RL?

Classic RL steps: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} J$

Also reads as:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$$

$$\Delta\boldsymbol{\theta} = \arg\min_{\Delta\boldsymbol{\theta}} \frac{1}{2\alpha} \left\|\Delta\boldsymbol{\theta}\right\|^2 + \nabla_{\boldsymbol{\theta}} J^\top \Delta\boldsymbol{\theta}$$

Safe RL steps $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \Delta\boldsymbol{\theta}$:

$$\Delta\boldsymbol{\theta} = \arg\min_{\Delta\boldsymbol{\theta}} \frac{1}{2\alpha} \left\|\Delta\boldsymbol{\theta}\right\|^2 + \nabla_{\boldsymbol{\theta}} J^\top \Delta\boldsymbol{\theta}$$

$$\text{s.t.} \quad \mathbf{s}_{+} - \mathbf{f}\left(\mathbf{s}, \mathbf{a}, \boldsymbol{\theta} + \Delta\boldsymbol{\theta}\right) \in \mathbb{W}_{\boldsymbol{\theta}+\Delta\boldsymbol{\theta}}$$

$$\forall \left(\mathbf{s}, \mathbf{a}, \mathbf{s}_{+}\right) \text{ in data set}$$

Safe RL steps seek performance under safety constraints

Safety filters - Safe RL via projections

- RL can discover policy parameters $\boldsymbol{\theta}$ such that policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ has good closed-loop performances, ignoring safety (e.g. $\boldsymbol{\pi}_{\boldsymbol{\theta}}$ stems from a DNN). "Learning" safety implicitly is difficult.

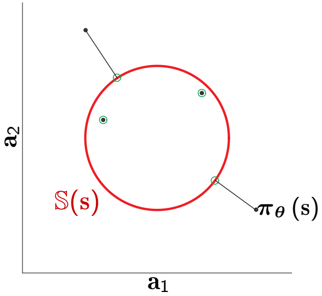
Safety filters - Safe RL via projections

- RL can discover policy parameters $\boldsymbol{\theta}$ such that policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ has good closed-loop performances, ignoring safety (e.g. $\boldsymbol{\pi}_{\boldsymbol{\theta}}$ stems from a DNN). "Learning" safety implicitly is difficult.
- If safe action set $\mathbb{S}(\mathrm{s})$ is somehow known, then we can
  - follow learned policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ when

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) \in \mathbb{S}(\mathbf{s})$$

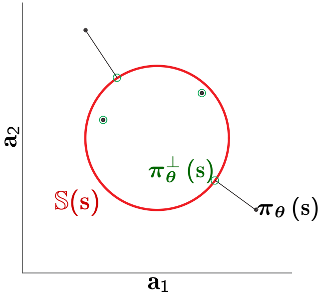
  - take "closest" action $\mathbf{a} \in \mathbb{S}(\mathbf{s})$ when

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) \notin \mathbb{S}(\mathbf{s})$$



Safety filters - Safe RL via projections

- RL can discover policy parameters $\boldsymbol{\theta}$ such that policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ has good closed-loop performances, ignoring safety (e.g. $\boldsymbol{\pi}_{\boldsymbol{\theta}}$ stems from a DNN). "Learning" safety implicitly is difficult.
- If safe action set $\mathbb{S}(\mathrm{s})$ is somehow known, then we can
  - follow learned policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ when
    $$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) \in \mathbb{S}(\mathrm{s})$$
  - take "closest" action $\mathbf{a} \in \mathbb{S}(\mathrm{s})$ when
    $$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) \notin \mathbb{S}(\mathrm{s})$$



More formally, safe policy e.g. reads as...

$$\begin{aligned} \boldsymbol{\pi}^{\perp}_{\boldsymbol{\theta}}(\mathrm{s}) = \arg\min_{\mathbf{a}} \quad & \|\mathbf{a} - \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})\|^2 \\ \text{s.t.} \quad & \mathbf{a} \in \mathbb{S}(\mathrm{s}) \end{aligned}$$

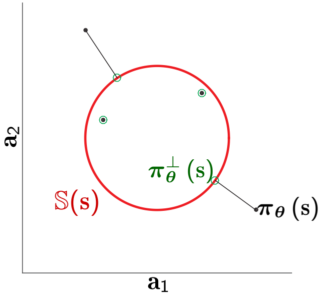
Safety filters - Safe RL via projections

- RL can discover policy parameters $\boldsymbol{\theta}$ such that policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ has good closed-loop performances, ignoring safety (e.g. $\boldsymbol{\pi}_{\boldsymbol{\theta}}$ stems from a DNN). "Learning" safety implicitly is difficult.
- If safe action set $\mathbb{S}(\mathrm{s})$ is somehow known, then we can
  - follow learned policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ when

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) \in \mathbb{S}(\mathrm{s})$$

  - take "closest" action $\mathbf{a} \in \mathbb{S}(\mathrm{s})$ when

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) \notin \mathbb{S}(\mathrm{s})$$



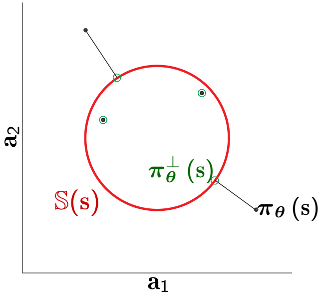
More formally, safe policy e.g. reads as...

$$\begin{aligned} \boldsymbol{\pi}_{\boldsymbol{\theta}}^{\perp}(\mathrm{s}) = \arg\min_{\mathbf{a}} \quad & \|\mathbf{a} - \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})\|^2 \\ \text{s.t.} \quad & \mathbf{a} \in \mathbb{S}(\mathrm{s}) \end{aligned}$$

Is that a good idea?

Safety filters - Safe RL via projections

- RL can discover policy parameters $\boldsymbol{\theta}$ such that policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ has good closed-loop performances, ignoring safety (e.g. $\boldsymbol{\pi}_{\boldsymbol{\theta}}$ stems from a DNN). "Learning" safety implicitly is difficult.
- If safe action set $\mathbb{S}(\mathrm{s})$ is somehow known, then we can
  - follow learned policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ when
  $$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) \in \mathbb{S}(\mathrm{s})$$
  - take "closest" action $\mathbf{a} \in \mathbb{S}(\mathrm{s})$ when
  $$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) \notin \mathbb{S}(\mathrm{s})$$



More formally, safe policy e.g. reads as...

$$\begin{aligned} \boldsymbol{\pi}_{\boldsymbol{\theta}}^{\perp}(\mathrm{s}) = \arg\min_{\mathbf{a}} \quad & \|\mathbf{a} - \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})\|^2 \\ \text{s.t.} \quad & \mathbf{a} \in \mathbb{S}(\mathrm{s}) \end{aligned}$$

Is that a good idea? It depends...

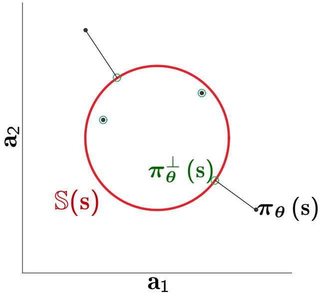
Safety filters - Safe RL via projections

- RL can discover policy parameters $\boldsymbol{\theta}$ such that policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ has good closed-loop performances, ignoring safety (e.g. $\boldsymbol{\pi}_{\boldsymbol{\theta}}$ stems from a DNN). "Learning" safety implicitly is difficult.
- If safe action set $\mathbb{S}(\mathrm{s})$ is somehow known, then we can
  - follow learned policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$ when

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) \in \mathbb{S}(\mathrm{s})$$

  - take "closest" action $\mathbf{a} \in \mathbb{S}(\mathrm{s})$ when

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) \notin \mathbb{S}(\mathrm{s})$$



More formally, safe policy e.g. reads as...

$$\begin{aligned} \boldsymbol{\pi}_{\boldsymbol{\theta}}^{\perp}(\mathrm{s}) = \arg\min_{\mathbf{a}} \quad & \left\| \mathbf{a} - \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) \right\|^2 \\ \text{s.t.} \quad & \mathbf{a} \in \mathbb{S}(\mathrm{s}) \end{aligned}$$

- Built from "Robust MPC" methods?
- Interaction with learning?

Is that a good idea? It depends...

Safety filters - How to obtain optimality?

Q learning: $Q_\theta \approx Q^\star$ learned via classic RL, ignoring safety.

Safety filters - How to obtain optimality?

Q learning: $Q_{\boldsymbol{\theta}} \approx Q^\star$ learned via classic RL, ignoring safety. Then

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\perp}(\mathrm{s}) = \arg\min_{\mathbf{a}} \quad \|\mathbf{a} - \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})\|^2 \qquad \text{s.t.} \quad \mathbf{a} \in \mathbb{S}(\mathrm{s})$$

where

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) = \arg\min_{\mathbf{a}} \; Q_{\boldsymbol{\theta}}(\mathrm{s}, \mathbf{a})$$

Safety filters - How to obtain optimality?

Q learning: $Q_\theta \approx Q^\star$ learned via classic RL, ignoring safety. Then

$$\boldsymbol{\pi}^\perp_{\boldsymbol{\theta}}\left(\mathrm{s}\right) = \arg\min_{\mathbf{a}} \quad \left\|\mathbf{a} - \boldsymbol{\pi}_{\boldsymbol{\theta}}\left(\mathrm{s}\right)\right\|^2 \qquad \text{s.t.} \quad \mathbf{a} \in \mathbb{S}\left(\mathrm{s}\right)$$

where

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}\left(\mathrm{s}\right) = \arg\min_{\mathbf{a}} \; Q_{\boldsymbol{\theta}}\left(\mathrm{s}, \mathbf{a}\right)$$

yields suboptimal policy $\boldsymbol{\pi}^\perp_{\boldsymbol{\theta}}$

Safety filters - How to obtain optimality?

Q learning: $Q_{\boldsymbol{\theta}} \approx Q^\star$ learned via classic RL, ignoring safety. Then

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\perp}(\mathbf{s}) = \arg\min_{\mathbf{a}} \quad Q_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{a}) \qquad \text{s.t.} \quad \mathbf{a} \in \mathbb{S}(\mathbf{s})$$

instead of a least-squares approach. Provably optimal (safe) policy.

Safety filters - How to obtain optimality?

Q learning: $Q_\theta \approx Q^\star$ learned via classic RL, ignoring safety. Then

$$\begin{aligned} \pi_\theta^\perp(\mathbf{s}) = \arg\min_{\mathbf{a}} \quad & Q_\theta(\mathbf{s},\mathbf{a}) \\ \text{s.t.} \quad & \mathbf{a} \in \mathbb{S}(\mathbf{s}) \end{aligned}$$

instead of a least-squares approach. Provably optimal (safe) policy.

Deterministic Policy gradient (actor-critic): the "regular expression"

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}\left[\nabla_\theta \pi_\theta \nabla_{\mathbf{a}} A_{\pi_\theta^\perp}\right]$$

yields incorrect gradients

Safety filters - How to obtain optimality?

Q learning: $Q_{\theta} \approx Q^{\star}$ learned via classic RL, ignoring safety. Then

$$\pi_{\theta}^{\perp}(\mathbf{s}) = \arg\min_{\mathbf{a}} \quad Q_{\theta}(\mathbf{s}, \mathbf{a}) \qquad \text{s.t.} \quad \mathbf{a} \in \mathbb{S}(\mathbf{s})$$

instead of a least-squares approach. Provably optimal (safe) policy.

Deterministic Policy gradient (actor-critic): make sure to evaluate the gradient using

$$\nabla_{\theta} J\left(\pi_{\theta}^{\perp}\right) = \mathbb{E}\left[\nabla_{\theta}\pi_{\theta}^{\perp}\nabla_{\mathbf{a}}A_{\pi_{\theta}^{\perp}}\right]$$

where

$$\pi_{\theta}^{\perp}(\mathbf{s}) = \arg\min_{\mathbf{a}} \quad \|\mathbf{a} - \pi_{\theta}(\mathbf{s})\|^2 \qquad \text{s.t.} \quad \mathbf{a} \in \mathbb{S}(\mathbf{s})$$

i.e. **account for projection (⇒differentiate NLP). Provably correct gradients.**

Safety filters - How to obtain optimality?

Q learning: $Q_{\boldsymbol{\theta}} \approx Q^\star$ learned via classic RL, ignoring safety. Then

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\perp}(\mathbf{s}) = \arg\min_{\mathbf{a}} \quad Q_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{a}) \qquad \text{s.t.} \quad \mathbf{a} \in \mathbb{S}(\mathbf{s})$$

instead of a least-squares approach. Provably optimal (safe) policy.

Deterministic Policy gradient (actor-critic): make sure to evaluate the gradient using

$$\nabla_{\boldsymbol{\theta}} J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\perp}\right) = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}}^{\perp} \nabla_{\mathbf{a}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\perp}}\right]$$

where

$$\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\perp}(\mathbf{s}) = \arg\min_{\mathbf{a}} \quad \|\mathbf{a} - \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s})\|^2 \qquad \text{s.t.} \quad \mathbf{a} \in \mathbb{S}(\mathbf{s})$$

i.e. **account for projection (⇒differentiate NLP). Provably correct gradients.**

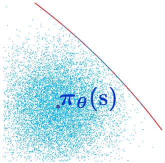
Stochastic policy gradient: where $\pi_{\boldsymbol{\theta}}$ is a probability density over the actions

$$\nabla_{\boldsymbol{\theta}} J\left(\pi_{\boldsymbol{\theta}}^{\perp}\right) = \mathbb{E}\left[\log \nabla_{\boldsymbol{\theta}} \pi_{\boldsymbol{\theta}} A_{\pi_{\boldsymbol{\theta}}^{\perp}}\right]$$

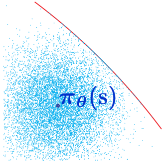
i.e. **do not account for projection. Provably correct gradients.**

Safe (feasible) exploration with MPC

Learning requires exploration. E.g. apply $\mathbf{a} = \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ to the real system where $\mathbf{d}$ is a "disturbance"



Safe (feasible) exploration with MPC



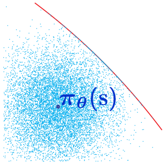
Learning requires exploration. E.g. apply $\mathbf{a} = \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ to the real system where $\mathbf{d}$ is a "disturbance"

Explore while keeping feasibility?

- Clearly an arbitrary "policy disturbance" $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ is a poor idea...
- NLP-based policy: "disturb" the cost function instead! (different options)

Safe (feasible) exploration with MPC

Learning requires exploration. E.g. apply $\mathbf{a} = \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) + \mathbf{d}$ to the real system where $\mathbf{d}$ is a "disturbance"



Explore while keeping feasibility?

- Clearly an arbitrary "policy disturbance" $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s}) + \mathbf{d}$ is a poor idea...
- NLP-based policy: "disturb" the cost function instead! (different options)

Feasible exploration: $\boldsymbol{\pi}^{\mathrm{e}}_{\boldsymbol{\theta}}(\mathrm{s}) = \mathbf{a}_0^\star$:

$$\min_{\mathbf{x},\mathbf{u}} \quad T(\mathbf{x}_N) - \mathbf{d}^\top \mathbf{u}_0 + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k)$$

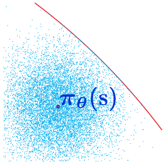
$$\text{s.t.} \quad \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$$

$$\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s}$$

satisfies the constraints by construction

Safe (feasible) exploration with MPC

Learning requires exploration. E.g. apply $\mathbf{a} = \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ to the real system where $\mathbf{d}$ is a "disturbance"



Explore while keeping feasibility?

- Clearly an arbitrary "policy disturbance" $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ is a poor idea...
- NLP-based policy: "disturb" the cost function instead! (different options)

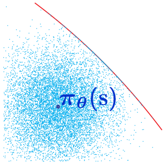
Feasible exploration: $\boldsymbol{\pi}^{\mathrm{e}}_{\boldsymbol{\theta}}(\mathbf{s}) = \mathbf{a}_0^\star$:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) - \mathbf{d}^\top \mathbf{u}_0 + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \mathrm{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s} \end{aligned}$$

satisfies the constraints by construction

Safe (feasible) exploration with MPC

Learning requires exploration. E.g. apply $\mathbf{a} = \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ to the real system where $\mathbf{d}$ is a "disturbance"



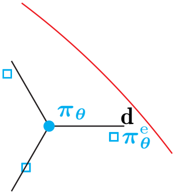
Explore while keeping feasibility?

- Clearly an arbitrary "policy disturbance" $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ is a poor idea...
- NLP-based policy: "disturb" the cost function instead! (different options)

Feasible exploration: $\boldsymbol{\pi}^{\mathrm{e}}_{\boldsymbol{\theta}}(\mathbf{s}) = \mathbf{a}_0^\star$:

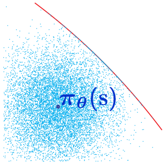
$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) - \mathbf{d}^\top \mathbf{u}_0 + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \mathrm{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s} \end{aligned}$$

satisfies the constraints by construction



Safe (feasible) exploration with MPC

Learning requires exploration. E.g. apply $\mathbf{a} = \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ to the real system where $\mathbf{d}$ is a "disturbance"



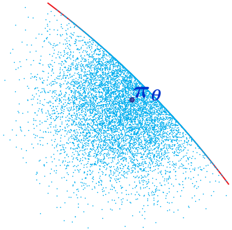
Explore while keeping feasibility?

- Clearly an arbitrary "policy disturbance" $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ is a poor idea...
- NLP-based policy: "disturb" the cost function instead! (different options)

Feasible exploration: $\boldsymbol{\pi}^{\mathrm{e}}_{\boldsymbol{\theta}}(\mathbf{s}) = \mathbf{a}_0^\star$:

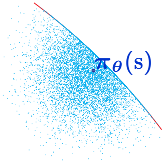
$$\begin{aligned}\min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) - \mathbf{d}^\top \mathbf{u}_0 + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s}\end{aligned}$$

satisfies the constraints by construction



Safe (feasible) exploration with MPC

Learning requires exploration. E.g. apply $\mathbf{a} = \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ to the real system where $\mathbf{d}$ is a "disturbance"



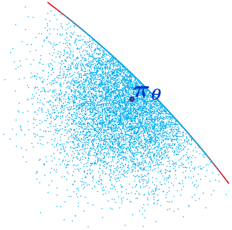
Explore while keeping feasibility?

- Clearly an arbitrary "policy disturbance" $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ is a poor idea...
- NLP-based policy: "disturb" the cost function instead! (different options)

Feasible exploration: $\boldsymbol{\pi}^{\mathrm{e}}_{\boldsymbol{\theta}}(\mathbf{s}) = \mathbf{a}_0^\star$:

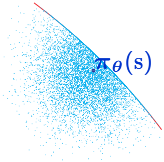
$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) - \mathbf{d}^\top \mathbf{u}_0 + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s} \end{aligned}$$

satisfies the constraints by construction



Safe (feasible) exploration with MPC

Learning requires exploration. E.g. apply $\mathbf{a} = \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ to the real system where $\mathbf{d}$ is a "disturbance"



Explore while keeping feasibility?

- Clearly an arbitrary "policy disturbance" $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s}) + \mathbf{d}$ is a poor idea...
- NLP-based policy: "disturb" the cost function instead! (different options)

Feasible exploration: $\boldsymbol{\pi}^{\mathrm{e}}_{\boldsymbol{\theta}}(\mathbf{s}) = \mathbf{a}_0^\star$:

$$\begin{aligned}\min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) - \mathbf{d}^\top \mathbf{u}_0 + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \mathrm{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s}\end{aligned}$$

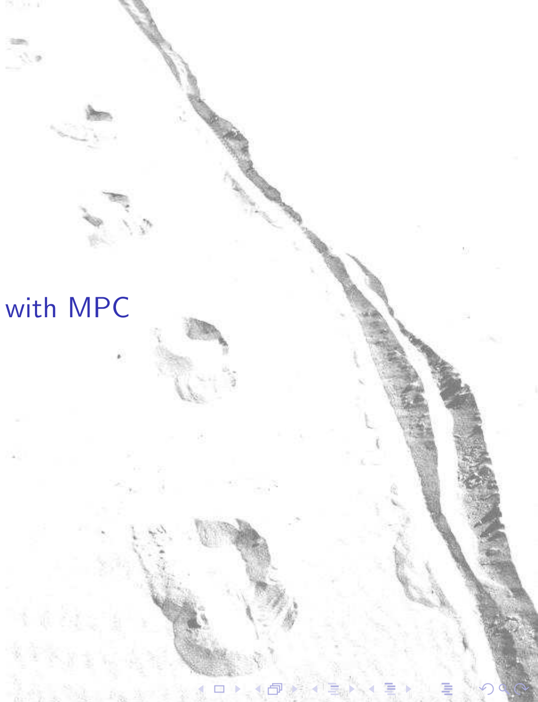
satisfies the constraints by construction

Remarks:

- Exploration $\mathbf{e} = \boldsymbol{\pi}^{\mathrm{e}}_{\boldsymbol{\theta}} - \boldsymbol{\pi}_{\boldsymbol{\theta}}$ is not centred-isotopric
- Can create some technical issues with actor-critic methods (linear compatible $A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}$), yields biased policy gradient estimation
- Bias seems small in practice

Outline

1. Safe RL via MPC
2. Stability-constrained Learning with MPC
3. Some more results (in brief)
4. Applications & Reflections



Stability of MPC

Policy π^{MPC} from

$$\begin{aligned}\min_{\mathbf{x},\mathbf{u}} \quad & T\left(\mathbf{x}_N\right) + \sum_{k=0}^{N-1} L\left(\mathbf{x}_k, \mathbf{u}_k\right) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right), \;\; \mathbf{x}_0 = \mathbf{s} \\ & \mathbf{h}\left(\mathbf{x}_k, \mathbf{u}_k\right) \leq 0\end{aligned}$$

Stability of MPC

Policy π^{MPC} from

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T\left(\mathbf{x}_N\right) + \sum_{k=0}^{N-1} L\left(\mathbf{x}_k, \mathbf{u}_k\right) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right), \quad \mathbf{x}_0 = \mathbf{s} \\ & \mathbf{h}\left(\mathbf{x}_k, \mathbf{u}_k\right) \leq 0 \end{aligned}$$

MPC scheme is (nominally) stabilizing if there is λ such that

$$\ell\left(\mathbf{s}, \mathbf{a}\right) := L\left(\mathrm{s}, \mathrm{a}\right) + \lambda\left(\mathrm{s}\right) - \lambda\left(\mathbf{f}\left(\mathrm{s}, \mathrm{a}\right)\right) \geq \kappa\left(\|\mathbf{s} - \mathbf{s}_{\mathrm{s}}\|\right), \quad \forall\, \mathbf{s}, \mathbf{a}$$

where κ is K_∞ (+conditions on T)

Stability of MPC

Policy π^{MPC} from

$$\min_{\mathbf{x},\mathbf{u}} \quad T\left(\mathbf{x}_N\right) + \sum_{k=0}^{N-1} L\left(\mathbf{x}_k, \mathbf{u}_k\right)$$

$$\text{s.t.} \quad \mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right), \quad \mathbf{x}_0 = \mathbf{s}$$

$$\mathbf{h}\left(\mathbf{x}_k, \mathbf{u}_k\right) \leq 0$$

Equivalent MPC

$$\min_{\mathbf{s},\mathbf{a}} \quad -\lambda\left(\mathbf{s}\right) + \tilde{T}\left(\mathbf{x}_N\right) + \sum_{k=0}^{N-1} \ell\left(\mathbf{x}_k, \mathbf{u}_k\right)$$

$$\text{s.t.} \quad \mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right), \quad \mathbf{x}_0 = \mathbf{s}$$

$$\mathbf{h}\left(\mathbf{x}_k, \mathbf{u}_k\right) \leq 0$$

brings us back to classic stability theory

MPC scheme is (nominally) stabilizing if there is λ such that

$$\ell\left(\mathbf{s}, \mathbf{a}\right) := L\left(\mathbf{s}, \mathbf{a}\right) + \lambda\left(\mathbf{s}\right) - \lambda\left(\mathbf{f}\left(\mathbf{s}, \mathbf{a}\right)\right) \geq \kappa\left(\|\mathbf{s} - \mathbf{s}_{\mathrm{s}}\|\right), \quad \forall\, \mathbf{s}, \mathbf{a}$$

where κ is K_∞ (+conditions on T)

Stability of MPC

Policy π^{MPC} from

$$\min_{\mathbf{x},\mathbf{u}} \quad T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k)$$

$$\text{s.t.} \quad \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k), \quad \mathbf{x}_0 = \mathbf{s}$$

$$\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0$$

Equivalent MPC

$$\min_{\mathbf{s},\mathbf{a}} \quad -\lambda(\mathbf{s}) + \tilde{T}(\mathbf{x}_N) + \sum_{k=0}^{N-1} \ell(\mathbf{x}_k, \mathbf{u}_k)$$

$$\text{s.t.} \quad \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k), \quad \mathbf{x}_0 = \mathbf{s}$$

$$\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0$$

brings us back to classic stability theory

MPC scheme is (nominally) stabilizing if there is λ such that

$$\ell(\mathbf{s}, \mathbf{a}) := L(\mathbf{s}, \mathbf{a}) + \lambda(\mathbf{s}) - \lambda(\mathbf{f}(\mathbf{s}, \mathbf{a})) \geq \kappa(\|\mathbf{s} - \mathbf{s}_{\mathrm{s}}\|), \quad \forall\, \mathbf{s}, \mathbf{a}$$

where κ is K_∞ (+conditions on T)

Remarks

- Modifying the MPC cost is a concept already present in dissipativity theory!
- Aligned with modifying the cost for MPC performance
- $\rightarrow$ Merge the RL & stability modifications for "Stability by design"

Stability-constrained Learning-based MPC - Deterministic case

Given arbitrary stage cost $L(\mathbf{s}, \mathbf{a})$, build a **stable policy** $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ minimizing:

$$J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}\right) = \sum_{k=0}^{\infty} L\left(\mathbf{s}_k, \mathbf{a}_k\right)$$

Stability-constrained Learning-based MPC - Deterministic case

Given arbitrary stage cost $L(\mathbf{s}, \mathbf{a})$, build a **stable policy** $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ minimizing:

$$J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}\right) = \sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)$$

Parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ from:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & -\lambda_{\boldsymbol{\theta}}(\mathbf{s}) + T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k), \;\; \mathbf{x}_0 = \mathbf{s} \\ & \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \leq 0 \end{aligned}$$

Stability-constrained Learning-based MPC - Deterministic case

Given arbitrary stage cost $L(\mathbf{s}, \mathbf{a})$, build a **stable policy** $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ minimizing:

$$J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}\right) = \sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)$$

- Learning based on L
- Impose constraint:

$$L_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{a}) \geq \kappa\left(\|\mathbf{s} - \mathbf{s}_{\mathrm{s}}\|\right), \quad \forall\, \mathbf{s}, \mathbf{a}$$

  throughout the learning
- $L_{\boldsymbol{\theta}}$ different than L from constraint & model error

Parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ from:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & -\lambda_{\boldsymbol{\theta}}(\mathbf{s}) + T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \\ \mathrm{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k), \ \mathbf{x}_0 = \mathbf{s} \\ & \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \leq 0 \end{aligned}$$

Stability-constrained Learning-based MPC - Deterministic case

Given arbitrary stage cost $L(\mathbf{s},\mathbf{a})$, build a **stable policy** $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ minimizing:

$$J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}\right) = \sum_{k=0}^{\infty} L(\mathbf{s}_k,\mathbf{a}_k)$$

- Learning based on L
- Impose constraint:

  $$L_{\boldsymbol{\theta}}(\mathbf{s},\mathbf{a}) \geq \kappa\left(\|\mathbf{s}-\mathbf{s}_{\mathrm{s}}\|\right), \quad \forall\, \mathbf{s},\mathbf{a}$$

  throughout the learning
- $L_{\boldsymbol{\theta}}$ different than L from constraint & model error

Parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ from:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & -\lambda_{\boldsymbol{\theta}}(\mathbf{s}) + T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k,\mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k,\mathbf{u}_k), \ \ \mathbf{x}_0 = \mathbf{s} \\ & \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k,\mathbf{u}_k) \leq 0 \end{aligned}$$

Theorem: under some conditions

- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \rightarrow \boldsymbol{\pi}_\star$ if $\boldsymbol{\pi}_\star$ is stabilizing
- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \rightarrow$ best stabilizing policy otherwise

Stability-constrained Learning-based MPC - Deterministic case

Given arbitrary stage cost $L(\mathbf{s}, \mathbf{a})$, build a **stable policy** $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ minimizing:

$$J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}\right) = \sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)$$

- Learning based on L
- Impose constraint:

$$L_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{a}) \geq \kappa(\|\mathbf{s} - \mathbf{s}_{\mathrm{s}}\|), \quad \forall\, \mathbf{s}, \mathbf{a}$$

throughout the learning
- $L_{\boldsymbol{\theta}}$ different than L from constraint & model error

Parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ from:

$$\begin{aligned}
\min_{\mathbf{x},\mathbf{u}} \quad & -\lambda_{\boldsymbol{\theta}}(\mathbf{s}) + T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \\
\text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k), \;\; \mathbf{x}_0 = \mathbf{s} \\
& \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \leq 0
\end{aligned}$$

Theorem: under some conditions

- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \rightarrow \boldsymbol{\pi}_\star$ if $\boldsymbol{\pi}_\star$ is stabilizing
- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \rightarrow$ best stabilizing policy otherwise

Change of philosophy from "classic" dissipativity framework:
stability analysis $\rightarrow$ stable design

Stability-constrained Learning-based MPC - Deterministic case

Given arbitrary stage cost $L(\mathbf{s}, \mathbf{a})$, build a **stable policy** $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ minimizing:

$$J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}\right) = \sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)$$

Constraint

$$L_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{a}) \geq \kappa\left(\|\mathbf{s} - \mathbf{s}_{\mathrm{s}}\|\right), \quad \forall\, \mathbf{s}, \mathbf{a}$$

is semi-infinite programming, not trivial

Some solutions:

- Sum-of-Squares (SOS) prog.
- Convex $L_{\boldsymbol{\theta}}$ (+ radially unbounded)
- Something else?

Parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ from:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & -\lambda_{\boldsymbol{\theta}}(\mathbf{s}) + T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k), \quad \mathbf{x}_0 = \mathbf{s} \\ & \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \leq 0 \end{aligned}$$

Theorem: under some conditions

- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \to \boldsymbol{\pi}_\star$ if $\boldsymbol{\pi}_\star$ is stabilizing
- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \to$ best stabilizing policy otherwise

Change of philosophy from "classic" dissipativity framework:
stability analysis $\to$ **stable design**

Stability-constrained Learning-based MPC - Deterministic case

Given arbitrary stage cost $L(\mathbf{s},\mathbf{a})$, build a **stable policy** $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ minimizing:

$$J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}\right) = \sum_{k=0}^{\infty} L\left(\mathbf{s}_k, \mathbf{a}_k\right)$$

Note that $\lambda_{\boldsymbol{\theta}}$ is redundant for policy gradient, needed for Q-learning... Combining both is meaningful!

Parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ from:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & -\lambda_{\boldsymbol{\theta}}(\mathbf{s}) + T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k), \;\; \mathbf{x}_0 = \mathbf{s} \\ & \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \leq 0 \end{aligned}$$

Theorem: under some conditions

- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \rightarrow \boldsymbol{\pi}_\star$ if $\boldsymbol{\pi}_\star$ is stabilizing
- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \rightarrow$ best stabilizing policy otherwise

Change of philosophy from "classic" dissipativity framework:
stability analysis $\rightarrow$ **stable design**

Stability-constrained Learning-based MPC - Deterministic case

Given arbitrary stage cost $L(\mathbf{s},\mathbf{a})$, build a **stable policy** $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ minimizing:

$$J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}\right) = \sum_{k=0}^{\infty} L\left(\mathbf{s}_k, \mathbf{a}_k\right)$$

Extension to stable policy for MDPs?

- Need stability with discount
- Need "stochastic dissipativity"

Parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ from:

$$\begin{aligned}\min_{\mathbf{x},\mathbf{u}} \quad & -\lambda_{\boldsymbol{\theta}}(\mathbf{s}) + T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k), \ \mathbf{x}_0 = \mathbf{s} \\ & \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \leq 0\end{aligned}$$

Theorem: under some conditions

- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \to \boldsymbol{\pi}_\star$ if $\boldsymbol{\pi}_\star$ is stabilizing
- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \to$ best stabilizing policy otherwise

Change of philosophy from "classic" dissipativity framework:
stability analysis $\to$ **stable design**

Stability-constrained Learning-based MPC - Deterministic case

Given arbitrary stage cost $L(\mathbf{s}, \mathbf{a})$, build a **stable policy** $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ minimizing:

$$J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}\right) = \sum_{k=0}^{\infty} L\left(\mathbf{s}_k, \mathbf{a}_k\right)$$

Extension to stable policy for MDPs?

- Need stability with discount
- Need "stochastic dissipativity"

MDP dissipativity: (2x Automatica '22)

- Use Strong Discounted Strict Dissipativity conditions
- Form the dissipativity equations in the measure space of the MDP

Parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ from:

$$\begin{aligned}
\min_{\mathbf{x},\mathbf{u}} \quad & -\lambda_{\boldsymbol{\theta}}(\mathbf{s}) + T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \\
\text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k), \quad \mathbf{x}_0 = \mathbf{s} \\
& \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \leq 0
\end{aligned}$$

Theorem: under some conditions

- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \to \boldsymbol{\pi}_\star$ if $\boldsymbol{\pi}_\star$ is stabilizing
- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \to$ best stabilizing policy otherwise

Change of philosophy from "classic" dissipativity framework:
stability analysis $\to$ stable design

Stability-constrained Learning-based MPC - Deterministic case

Given arbitrary stage cost $L(\mathbf{s}, \mathbf{a})$, build a **stable policy** $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ minimizing:

$$J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}\right) = \sum_{k=0}^{\infty} L\left(\mathbf{s}_k, \mathbf{a}_k\right)$$

Extension to stable policy for MDPs?

- Need stability with discount
- Need "stochastic dissipativity"

MDP dissipativity: (2x Automatica '22)

- Use Strong Discounted Strict Dissipativity conditions
- Form the dissipativity equations in the measure space of the MDP

Parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}}$ from:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & -\lambda_{\boldsymbol{\theta}}(\mathbf{s}) + T_{\boldsymbol{\theta}}(\mathbf{x}_N) + \sum_{k=0}^{N-1} L_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k), \ \mathbf{x}_0 = \mathbf{s} \\ & \mathbf{h}_{\boldsymbol{\theta}}(\mathbf{x}_k, \mathbf{u}_k) \leq 0 \end{aligned}$$

Theorem: under some conditions

- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \to \boldsymbol{\pi}_\star$ if $\boldsymbol{\pi}_\star$ is stabilizing
- $\boldsymbol{\pi}_{\boldsymbol{\theta}}^{\mathrm{MPC}} \to$ best stabilizing policy otherwise

Change of philosophy from "classic" dissipativity framework:
stability analysis $\to$ **stable design**

We have the maths to treat this, not yet the algorithms...

Stability of MPC - Stochastic dynamics

Policy π_{MPC} from

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T\left(\mathbf{x}_N\right) + \sum_{k=0}^{N-1} L\left(\mathbf{x}_k, \mathbf{u}_k\right) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right) \\ & \mathbf{h}\left(\mathbf{x}_k, \mathbf{u}_k\right) \leq 0, \quad \mathbf{x}_0 = \mathbf{s} \end{aligned}$$

MDP:

$$\min_{\boldsymbol{\pi}} \quad \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{\infty} L\left(\mathbf{s}_k, \mathbf{a}_k\right)\right]$$

where $\mathbf{a}_k = \boldsymbol{\pi}\left(\mathbf{s}_k\right)$ and system dynamics

$$\mathbf{s}_{k+1} \sim \mathbb{P}\left[\,\cdot\,|\,\mathbf{s}_k, \mathbf{a}_k\,\right]$$

Stability of MPC - Stochastic dynamics

Policy π_{MPC} from

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s} \end{aligned}$$

MDP:

$$\min_{\boldsymbol{\pi}} \quad \mathbb{E}_{\boldsymbol{\pi}} \left[\sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k) \right]$$

where $\mathbf{a}_k = \boldsymbol{\pi}(\mathbf{s}_k)$ and system dynamics

$$\mathbf{s}_{k+1} \sim \mathbb{P}[\,\cdot\,|\,\mathbf{s}_k, \mathbf{a}_k\,]$$

Classic stability via Lyapunov:

- $V^{\mathrm{MPC}}(\mathbf{s})$ **decrease** along the system trajectories, i.e.

$$V^{\mathrm{MPC}}\left(\mathbf{f}\left(\mathbf{s}, \boldsymbol{\pi}^{\mathrm{MPC}}(\mathbf{s})\right)\right) < V^{\mathrm{MPC}}(\mathbf{s})$$

  is ensured by construction

Stability of MPC - Stochastic dynamics

Policy π_{MPC} from

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s} \end{aligned}$$

MDP:

$$\min_{\boldsymbol{\pi}} \quad \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)\right]$$

where $\mathbf{a}_k = \boldsymbol{\pi}(\mathbf{s}_k)$ and system dynamics

$$\mathbf{s}_{k+1} \sim \mathbb{P}[\,\cdot\,|\,\mathbf{s}_k, \mathbf{a}_k\,]$$

Classic stability via Lyapunov:

- $V^{\text{MPC}}(\mathbf{s})$ **decrease** along the system trajectories, i.e.

$$V^{\text{MPC}}\left(\mathbf{f}\left(\mathbf{s}, \boldsymbol{\pi}^{\text{MPC}}(\mathbf{s})\right)\right) < V^{\text{MPC}}(\mathbf{s})$$

  is ensured by construction

- What if $\mathbf{s}_+ \sim \mathbb{P}[\,.\,|\,\mathbf{s}, \boldsymbol{\pi}^\star(\mathbf{s})\,]$ is stochastic (with know density)?

$$V^{\text{MPC}}(\mathbf{s}_+) < V^{\text{MPC}}(\mathbf{s}), \quad \forall\,\mathbf{s}$$

in *some sense*? **Not really**... (unless strong assumptions)

Stability of MPC - Stochastic dynamics

Policy π_{MPC} from

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s} \end{aligned}$$

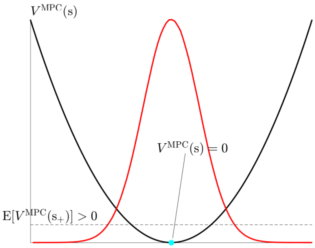
MDP:

$$\min_{\boldsymbol{\pi}} \quad \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)\right]$$

where $\mathbf{a}_k = \boldsymbol{\pi}(\mathbf{s}_k)$ and system dynamics

$$\mathbf{s}_{k+1} \sim \mathbb{P}[\,\cdot\,|\,\mathbf{s}_k, \mathbf{a}_k\,]$$

E.g. thought experiment: V^{MPC} convex, s at the minimum...



Stability of MPC - Stochastic dynamics

Policy π_{MPC} from

$$\min_{\mathbf{x},\mathbf{u}} \quad T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k)$$

$$\text{s.t.} \quad \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$$

$$\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s}$$

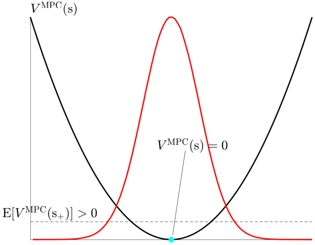
MDP:

$$\min_{\boldsymbol{\pi}} \quad \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)\right]$$

where $\mathbf{a}_k = \boldsymbol{\pi}(\mathbf{s}_k)$ and system dynamics

$$\mathbf{s}_{k+1} \sim \mathbb{P}[\,\cdot\,|\,\mathbf{s}_k, \mathbf{a}_k\,]$$

E.g. thought experiment: V^{MPC} convex, s at the minimum...



A Lyapunov stability theory for MDP in terms of state (beyond "stability to a set") is in general not possible.

Yet MDPs can be stable

Stability of MPC - Stochastic dynamics

Policy π_{MPC} from

$$\min_{\mathbf{x},\mathbf{u}} \quad T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k)$$
$$\text{s.t.} \quad \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$$
$$\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s}$$

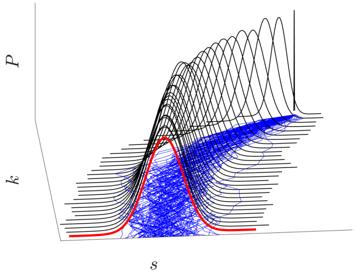
MDP:

$$\min_{\boldsymbol{\pi}} \quad \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)\right]$$

where $\mathbf{a}_k = \boldsymbol{\pi}(\mathbf{s}_k)$ and system dynamics

$$\mathbf{s}_{k+1} \sim \mathbb{P}[\,\cdot\,|\,\mathbf{s}_k, \mathbf{a}_k\,]$$

E.g. thought experiment: V^{MPC} convex, s at the minimum...



A Lyapunov stability theory for MDP in terms of state (beyond "stability to a set") is in general not possible.

Yet MDPs can be stable

Stability of MPC - Stochastic dynamics

Policy π_{MPC} from

$$\min_{\mathbf{x},\mathbf{u}} \quad T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k)$$

$$\text{s.t.} \quad \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$$

$$\mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s}$$

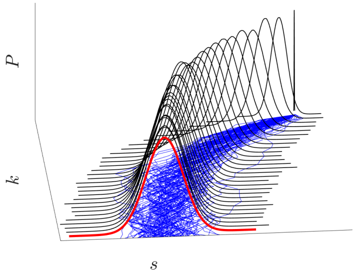
MDP:

$$\min_{\pi} \quad \mathbb{E}_{\pi}\left[\sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)\right]$$

where $\mathbf{a}_k = \pi(\mathbf{s}_k)$ and system dynamics

$$\mathbf{s}_{k+1} \sim \mathbb{P}[\,\cdot\,|\,\mathbf{s}_k, \mathbf{a}_k\,]$$

E.g. thought experiment: V^{MPC} convex, s at the minimum...



A Lyapunov stability theory for MDP in terms of state (beyond "stability to a set") is in general not possible.

Yet MDPs can be stable

Key idea: Lyapunov stability in the **state measure** rather than state space

Stability of MPC - Stochastic dynamics

Policy π_{MPC} from

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s} \end{aligned}$$

MDP:

$$\min_{\boldsymbol{\pi}} \quad \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)\right]$$

where $\mathbf{a}_k = \boldsymbol{\pi}(\mathbf{s}_k)$ and system dynamics

$$\mathbf{s}_{k+1} \sim \mathbb{P}[\,\cdot\,|\,\mathbf{s}_k, \mathbf{a}_k\,]$$

Key idea: Lyapunov stability in the **state measure** rather than state space

Functional dissipativity: if there is a functional λ such that:

$$\mathcal{L}[\rho, \boldsymbol{\pi}] - \lambda[\rho_+] + \lambda[\rho] \geq \kappa(D(\rho\,||\,\rho^{\text{s}})), \qquad \mathbf{s} \sim \rho, \; \mathbf{s}_+ \sim \rho_+$$

then the state distribution ρ converges to ρ^{s}

where

- $\mathcal{L}$ is the problem cost functional, e.g. $\mathcal{L} = \mathbb{E}[L(\mathbf{s}, \mathbf{a})]$
- $D(\cdot\,||\,\cdot)$ is a dissimilarity measure, e.g. Kullback-Liebler Divergence
- Choice of dissimilarity measure defines the form of stability

Stability of MPC - Stochastic dynamics

Policy π_{MPC} from

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s} \end{aligned}$$

MDP:

$$\min_{\boldsymbol{\pi}} \quad \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)\right]$$

where $\mathbf{a}_k = \boldsymbol{\pi}(\mathbf{s}_k)$ and system dynamics

$$\mathbf{s}_{k+1} \sim \mathbb{P}[\,\cdot\,|\,\mathbf{s}_k, \mathbf{a}_k\,]$$

Key idea: Lyapunov stability in the **state measure** rather than state space

Functional dissipativity: if there is a functional λ such that:

$$\mathcal{L}[\rho, \boldsymbol{\pi}] - \lambda[\rho_+] + \lambda[\rho] \geq \kappa(D(\rho\,||\,\rho^{\mathrm{s}})), \qquad \mathbf{s} \sim \rho,\ \mathbf{s}_+ \sim \rho_+$$

then the state distribution ρ converges to ρ^{s}

where

- $\mathcal{L}$ is the problem cost functional, e.g. $\mathcal{L} = \mathbb{E}[L(\mathbf{s}, \mathbf{a})]$
- $D(\,\cdot\,||\,\cdot\,)$ is a dissimilarity measure, e.g. Kullback-Liebler Divergence
- Choice of dissimilarity measure defines the form of stability

Not obvious how to use it in RL yet...

Outline

1. Safe RL via MPC

2. Stability-constrained Learning with MPC

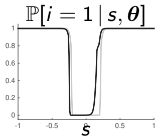
3. **Some more results (in brief)**

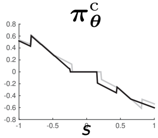
4. Applications & Reflections

RL & Mixed integer problem in MPC

Mixed-integer problems are common. Can we do RL over Mixed-integer MPC schemes?

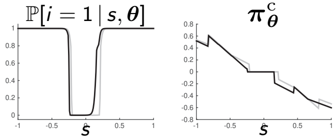
Assume mixed-integer actions





RL & Mixed integer problem in MPC

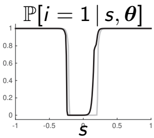
Mixed-integer problems are common. Can we do RL over Mixed-integer MPC schemes?

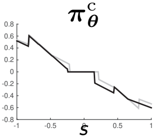


Assume mixed-integer actions

- With Q-learning, fairly trivial... incorrect if no exploration, though

RL & Mixed integer problem in MPC





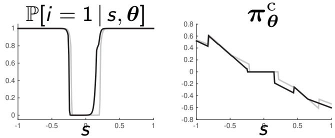
Mixed-integer problems are common. Can we do RL over Mixed-integer MPC schemes?

Assume mixed-integer actions

- With Q-learning, fairly trivial... incorrect if no exploration, though
- For policy gradient, devil is in the details
  - ✓ Integer inputs are best treated via stochastic policy gradient
  - ✓ Continuous inputs are "best treated" via deterministic policy gradient (in the presence of constraints)
  - ✓ Propose a hybrid policy gradient method combining deterministic and stochastic policies, with corresponding compatible linear A_{π_θ} approximations
  - ✓ Works well on mixed-integer MPC examples

RL & Mixed integer problem in MPC

Mixed-integer problems are common. Can we do RL over Mixed-integer MPC schemes?



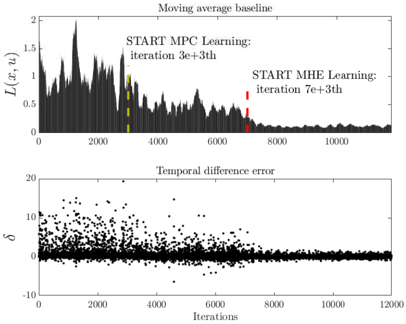
Assume mixed-integer actions

- With Q-learning, fairly trivial... incorrect if no exploration, though
- For policy gradient, devil is in the details
  - ✓ Integer inputs are best treated via stochastic policy gradient
  - ✓ Continuous inputs are "best treated" via deterministic policy gradient (in the presence of constraints)
  - ✓ Propose a hybrid policy gradient method combining deterministic and stochastic policies, with corresponding compatible linear $A_{\pi_{\theta}}$ approximations
  - ✓ Works well on mixed-integer MPC examples

More to be done on efficiency & control of the integer exploration

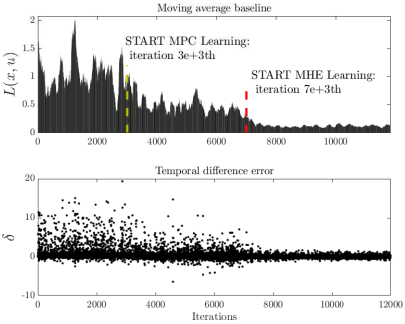
RL & MHE-MPC

The full state of the system is often not available, or not even modelled, use observer (e.g. MHE). Can we still do RL and how?



RL & MHE-MPC

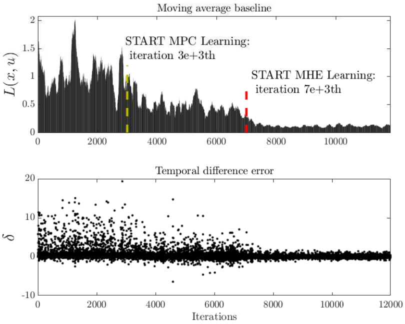
The full state of the system is often not available, or not even modelled, use observer (e.g. MHE). Can we still do RL and how?



- Problem becomes POMDP when MPC model does not include all states

RL & MHE-MPC

The full state of the system is often not available, or not even modelled, use observer (e.g. MHE). Can we still do RL and how?



- Problem becomes POMDP when MPC model does not include all states
- MHE becomes a component of the policy, must be treated in RL as well
  - ✓ RL can tune MHE and MPC jointly for closed loop performance in the context of Q learning
  - ✓ Algorithmic is simple, performances on example are good
  - ✓ The MHE tuning has a strong impact on performance (on our examples)
  - ✓ Extension to policy gradient is simple, yet to publish
  - ✓ Works also if MPC model omits some of the real states

Tuning of the MPC "meta"-parameters

MPC "meta"-parameters:

- Horizon length N
- When to recompute control sequence (event-based MPC)

MPC:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0 \end{aligned}$$

yields $\boldsymbol{\pi}_{\text{MPC}}(\mathbf{s}_0) = \mathbf{u}_0^\star$

Event-triggered:

- apply input profile $\mathbf{u}_{0,\ldots,n}^\star$ until re-computation is triggered
- often used to reduce computational demand, energy, communication, etc.
- Triggering is state-based, to be tuned

Tuning of the MPC "meta"-parameters

MPC "meta"-parameters:

- Horizon length N
- When to recompute control sequence (event-based MPC)

MPC:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T\left(\mathbf{x}_N\right) + \sum_{k=0}^{N-1} L\left(\mathbf{x}_k, \mathbf{u}_k\right) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right) \\ & \mathbf{h}\left(\mathbf{x}_k, \mathbf{u}_k\right) \leq 0 \end{aligned}$$

yields $\boldsymbol{\pi}_{\mathrm{MPC}}\left(\mathbf{s}_0\right) = \mathbf{u}_0^\star$

Event-triggered:

- apply input profile $\mathbf{u}^\star_{0,\ldots,n}$ until re-computation is triggered
- often used to reduce computational demand, energy, communication, etc.
- Triggering is state-based, to be tuned

Fairly simple idea, requires some care to be treated correctly:

✓ Define augmented state to preserve Markov property (essential for RL methods)

✓ Stochastic policy gradient methods required, must define the densities very carefully

RL to evaluate the storage function

Policy π_{MPC} from

$$\begin{aligned} \min_{\mathbf{x},\mathbf{u}} \quad & T(\mathbf{x}_N) + \sum_{k=0}^{N-1} L(\mathbf{x}_k, \mathbf{u}_k) \\ \text{s.t.} \quad & \mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k) \\ & \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k) \leq 0, \quad \mathbf{x}_0 = \mathbf{s} \end{aligned}$$

If for some λ function:

$$L(\mathbf{s}, \mathbf{a}) + \lambda(\mathbf{s}) - \lambda(\mathbf{f}(\mathbf{s}, \mathbf{a})) \geq \kappa(\|\mathbf{s} - \mathbf{s}_{\text{s}}\|), \quad \forall \mathbf{s}, \mathbf{a}$$

holds, then MPC scheme is stabilizing

How to evaluate λ?

- Approximate $\mathbf{f}$ as a polynomial, then Sum-of-Squares technique can be used
- We propose: parametrize λ and evaluate it via Q-learning
- On some examples, provides a more accurate λ than SOS
- Combination would arguably be good, to be done

MPC Beyond State Space

Systems with

- $\sim$Linear dynamics
- Input-output data
- Significant stochasticity
- Modelling is difficult

MPC Beyond State Space

Systems with

- ∼Linear dynamics
- Input-output data
- Significant stochasticity
- Modelling is difficult

Multi-step linear predictors

$$\hat{\mathbf{y}} = \Phi \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix}$$

- Recent input-output sequence $\mathbf{u}, \mathbf{y}$
- Planned input sequence $\mathbf{u}$
- Predicted output sequence $\hat{\mathbf{y}}$

MPC Beyond State Space

Multi-step linear predictors

SPC for $\mathbf{u}$, $\mathbf{y}$ given

$$\min_{\mathbf{u},\,\hat{\mathbf{y}}} \quad \sum_{k=0}^{N} L\left(\hat{\mathbf{y}}_k, \mathbf{u}_k\right)$$

$$\text{s.t.} \quad \hat{\mathbf{y}} = \Phi \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix}$$

$$\mathbf{h}\left(\hat{\mathbf{y}}_k, \mathbf{u}_k\right) \leq 0$$

yields policy $\boldsymbol{\pi}\left(\mathbf{u}, \mathbf{y}\right) = \mathbf{u}_0^\star$

$$\hat{\mathbf{y}} = \Phi \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix}$$

- Recent input-output sequence $\mathbf{u}, \mathbf{y}$
- Planned input sequence $\mathbf{u}$
- Predicted output sequence $\hat{\mathbf{y}}$

MPC Beyond State Space

SPC for $\mathbf{u}$, $\mathbf{y}$ given

$$\min_{\mathbf{u},\,\hat{\mathbf{y}}} \quad \sum_{k=0}^{N} L\left(\hat{\mathbf{y}}_k, \mathbf{u}_k\right)$$

$$\text{s.t.} \quad \hat{\mathbf{y}} = \Phi \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix}$$

$$\mathbf{h}\left(\hat{\mathbf{y}}_k, \mathbf{u}_k\right) \leq 0$$

yields policy $\boldsymbol{\pi}\left(\mathbf{u}, \mathbf{y}\right) = \mathbf{u}_0^\star$

Multi-step linear predictors

$$\hat{\mathbf{y}} = \Phi \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix}$$

- Recent input-output sequence $\mathbf{u}, \mathbf{y}$
- Planned input sequence $\mathbf{u}$
- Predicted output sequence $\hat{\mathbf{y}}$
- Measured output sequences $\mathbf{y}$

Where Φ can be **built from past data** $\mathcal{D}$, e.g.

$$\min_{\Phi} \quad \sum_{i\in\mathcal{D}} \frac{1}{2} \left\| \mathbf{y}_i - \Phi \begin{bmatrix} \mathbf{u}_i \\ \mathbf{y}_i \\ \mathbf{u}_i \end{bmatrix} \right\|^2 + R(\Phi)$$

$$\text{s.t.} \quad \Phi \text{ causal}$$

or other regressions

MPC Beyond State Space

SPC for $\mathbf{u}$, $\mathbf{y}$ given

$$\min_{\mathbf{u},\,\hat{\mathbf{y}}} \quad \sum_{k=0}^{N} L\left(\hat{\mathbf{y}}_k, \mathbf{u}_k\right)$$

$$\text{s.t.} \quad \hat{\mathbf{y}} = \Phi \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix}$$

$$\mathbf{h}\left(\hat{\mathbf{y}}_k, \mathbf{u}_k\right) \leq 0$$

yields policy $\boldsymbol{\pi}\left(\mathbf{u}, \mathbf{y}\right) = \mathbf{u}_0^\star$

Suffers from the same limitations as classic MPC

Multi-step linear predictors

$$\hat{\mathbf{y}} = \Phi \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix}$$

- Recent input-output sequence $\mathbf{u}, \mathbf{y}$
- Planned input sequence $\mathbf{u}$
- Predicted output sequence $\hat{\mathbf{y}}$
- Measured output sequences $\mathbf{y}$

Where Φ can be **built from past data** $\mathcal{D}$, e.g.

$$\min_{\Phi} \quad \sum_{i\in\mathcal{D}} \frac{1}{2} \left\| \mathbf{y}_i - \Phi \begin{bmatrix} \mathbf{u}_i \\ \mathbf{y}_i \\ \mathbf{u}_i \end{bmatrix} \right\|^2 + R(\Phi)$$

$$\text{s.t.} \quad \Phi \text{ causal}$$

or other regressions

MPC Beyond State Space

SPC for $\mathbf{u}$, $\mathbf{y}$ given

$$\min_{\mathbf{u},\,\hat{\mathbf{y}}} \quad \sum_{k=0}^{N} L\left(\hat{\mathbf{y}}_k, \mathbf{u}_k\right)$$

$$\text{s.t.} \quad \hat{\mathbf{y}} = \Phi \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix}$$

$$\mathbf{h}\left(\hat{\mathbf{y}}_k, \mathbf{u}_k\right) \leq 0$$

yields policy $\boldsymbol{\pi}\left(\mathbf{u}, \mathbf{y}\right) = \mathbf{u}_0^\star$

Can we do RL? Yes!

- RL-MPC theory applies with some twists
- State becomes $\mathbf{u}$, $\mathbf{y}$ (window of input-output)
- Modifications in principle not localized in time

MPC Beyond State Space

SPC for $\mathbf{u}$, $\mathbf{y}$ given

$$\min_{\mathbf{u},\, \hat{\mathbf{y}}} \quad \Psi_{\theta}\left(\mathbf{u}, \hat{\mathbf{y}}, \mathbf{u}, \mathbf{y}\right)$$

$$\text{s.t.} \quad \hat{\mathbf{y}} = \Phi_{\theta} \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix}$$

$$\mathbf{H}_{\theta}\left(\mathbf{u}, \hat{\mathbf{y}}, \mathbf{u}, \mathbf{y}\right) \leq 0$$

yields policy $\boldsymbol{\pi}_{\theta}\left(\mathbf{u}, \mathbf{y}\right) = \mathbf{u}_0^{\star}$

Can we do RL? Yes!

- RL-MPC theory applies with some twists
- State becomes $\mathbf{u}$, $\mathbf{y}$ (window of input-output)
- Modifications in principle not localized in time

MPC Beyond State Space

SPC for $\mathbf{u}$, $\mathbf{y}$ given

$$\begin{aligned}
\min_{\mathbf{u},\,\hat{\mathbf{y}}} \quad & \Psi_{\theta}\left(\mathbf{u},\, \hat{\mathbf{y}}, \mathbf{u}, \mathbf{y}\right) \\
\text{s.t.} \quad & \hat{\mathbf{y}} = \Phi_{\theta} \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix} \\
& \mathbf{H}_{\theta}\left(\mathbf{u},\, \hat{\mathbf{y}}, \mathbf{u}, \mathbf{y}\right) \leq 0
\end{aligned}$$

yields policy $\boldsymbol{\pi}_{\theta}\left(\mathbf{u}, \mathbf{y}\right) = \mathbf{u}_0^{\star}$

Can we do RL? Yes!

- RL-MPC theory applies with some twists
- State becomes $\mathbf{u}$, $\mathbf{y}$ (window of input-output)
- Modifications in principle not localized in time
- High-dimensional parameter space for RL
- Better behaved for learning than one-step simulation models *(?)*

MPC Beyond State Space

SPC for $\mathbf{u}$, $\mathbf{y}$ given

$$\min_{\mathbf{u},\,\hat{\mathbf{y}}} \quad \Psi_{\theta}\left(\mathbf{u},\, \hat{\mathbf{y}}, \mathbf{u}, \mathbf{y}\right)$$

$$\text{s.t.} \quad \hat{\mathbf{y}} = \Phi_{\theta} \begin{bmatrix} \mathbf{u} \\ \mathbf{y} \\ \mathbf{u} \end{bmatrix}$$

$$\mathbf{H}_{\theta}\left(\mathbf{u},\, \hat{\mathbf{y}}, \mathbf{u}, \mathbf{y}\right) \leq 0$$

yields policy $\boldsymbol{\pi}_{\theta}\left(\mathbf{u}, \mathbf{y}\right) = \mathbf{u}_0^{\star}$

Can we do RL? Yes!

- RL-MPC theory applies with some twists
- State becomes $\mathbf{u}$, $\mathbf{y}$ (window of input-output)
- Modifications in principle not localized in time
- High-dimensional parameter space for RL
- Better behaved for learning than one-step simulation models *(?)*

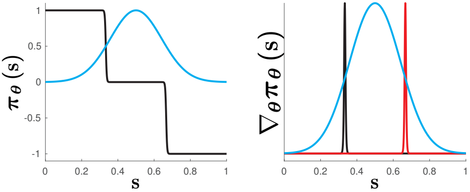
Nonlinear extension possible. Best way to do it is to be investigated.

RL & MPC for "strongly economic" problems

Some policies are dominated by "switches", difficult to treat in RL because $\nabla_{\theta}\pi_{\theta} = 0$ on most of the state space. Hence

$$\nabla_{\theta} J(\pi_{\theta}) = \mathbb{E}\left[\nabla_{\theta}\pi_{\theta}\nabla_{\mathrm{a}} A_{\pi_{\theta}}\right]$$

is based on contributions from a very small number of samples. Parameter updates become "infrequent and jumpy".



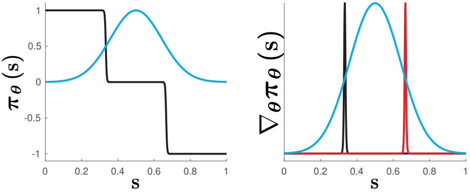
RL & MPC for "strongly economic" problems

Some policies are dominated by "switches", difficult to treat in RL because $\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} = 0$ on most of the state space. Hence

$$\nabla_{\boldsymbol{\theta}} J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}\right) = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}}\nabla_{\mathrm{a}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

is based on contributions from a very small number of samples. Parameter updates become "infrequent and jumpy".

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



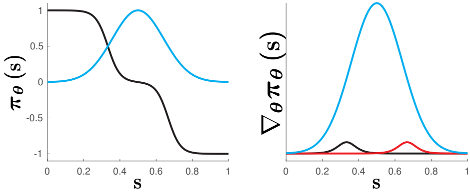
RL & MPC for “strongly economic” problems

Some policies are dominated by “switches”, difficult to treat in RL because $\nabla_\theta \pi_\theta = 0$ on most of the state space. Hence

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}\left[\nabla_\theta \pi_\theta \nabla_a A_{\pi_\theta}\right]$$

is based on contributions from a very small number of samples. Parameter updates become “infrequent and jumpy”.

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_\theta \pi_\theta \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



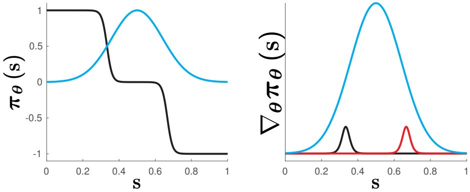
RL & MPC for "strongly economic" problems

Some policies are dominated by "switches", difficult to treat in RL because $\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} = 0$ on most of the state space. Hence

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}}) = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathbf{a}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

is based on contributions from a very small number of samples. Parameter updates become "infrequent and jumpy".

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



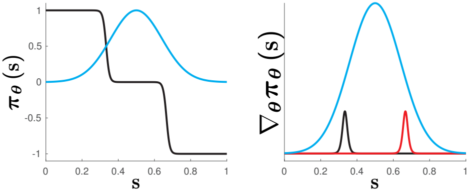
RL & MPC for “strongly economic” problems

Some policies are dominated by “switches”, difficult to treat in RL because $\nabla_{\theta}\pi_{\theta} = 0$ on most of the state space. Hence

$$\nabla_{\theta} J(\pi_{\theta}) = \mathbb{E}\left[\nabla_{\theta}\pi_{\theta}\nabla_{a} A_{\pi_{\theta}}\right]$$

is based on contributions from a very small number of samples. Parameter updates become “infrequent and jumpy”.

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_{\theta}\pi_{\theta} \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



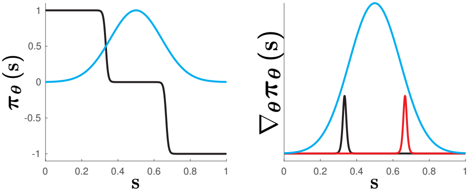
RL & MPC for "strongly economic" problems

Some policies are dominated by "switches", difficult to treat in RL because $\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} = 0$ on most of the state space. Hence

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}}) = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{a}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

is based on contributions from a very small number of samples. Parameter updates become "infrequent and jumpy".

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



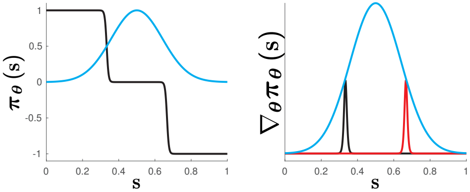
RL & MPC for "strongly economic" problems

Some policies are dominated by "switches", difficult to treat in RL because $\nabla_{\theta}\pi_{\theta} = 0$ on most of the state space. Hence

$$\nabla_{\theta} J(\pi_{\theta}) = \mathbb{E}\left[\nabla_{\theta}\pi_{\theta}\nabla_{a} A_{\pi_{\theta}}\right]$$

is based on contributions from a very small number of samples. Parameter updates become "infrequent and jumpy".

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_{\theta}\pi_{\theta} \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



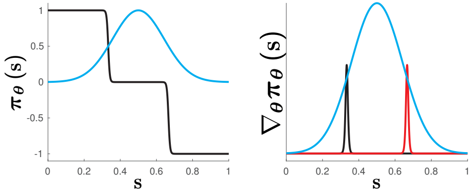
RL & MPC for “strongly economic” problems

Some policies are dominated by “switches”, difficult to treat in RL because $\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} = 0$ on most of the state space. Hence

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}}) = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{a}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

is based on contributions from a very small number of samples. Parameter updates become “infrequent and jumpy”.

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



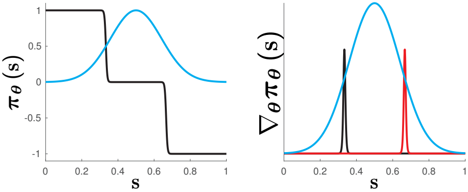
RL & MPC for "strongly economic" problems

Some policies are dominated by "switches", difficult to treat in RL because $\nabla_\theta \pi_\theta = 0$ on most of the state space. Hence

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}\left[\nabla_\theta \pi_\theta \nabla_{\mathrm{a}} A_{\pi_\theta}\right]$$

is based on contributions from a very small number of samples. Parameter updates become "infrequent and jumpy".

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_\theta \pi_\theta \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



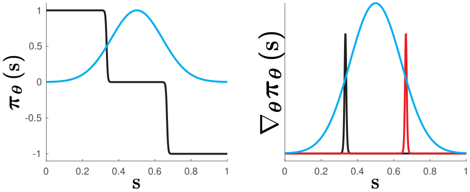
RL & MPC for "strongly economic" problems

Some policies are dominated by "switches", difficult to treat in RL because $\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} = 0$ on most of the state space. Hence

$$\nabla_{\boldsymbol{\theta}} J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}\right) = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}}\nabla_{\mathrm{a}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

is based on contributions from a very small number of samples. Parameter updates become "infrequent and jumpy".

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_{\boldsymbol{\theta}}\boldsymbol{\pi}_{\boldsymbol{\theta}} \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



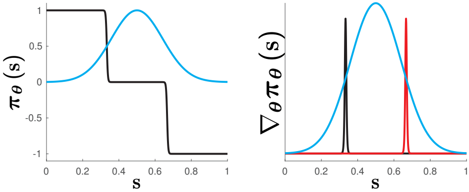
RL & MPC for "strongly economic" problems

Some policies are dominated by "switches", difficult to treat in RL because $\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} = 0$ on most of the state space. Hence

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}}) = \mathbb{E}\left[\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} \nabla_{\mathrm{a}} A_{\boldsymbol{\pi}_{\boldsymbol{\theta}}}\right]$$

is based on contributions from a very small number of samples. Parameter updates become "infrequent and jumpy".

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_{\boldsymbol{\theta}} \boldsymbol{\pi}_{\boldsymbol{\theta}} \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



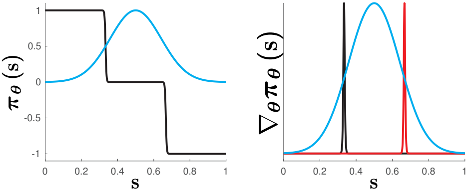
RL & MPC for "strongly economic" problems

Some policies are dominated by "switches", difficult to treat in RL because $\nabla_{\theta}\pi_{\theta} = 0$ on most of the state space. Hence

$$\nabla_{\theta} J(\pi_{\theta}) = \mathbb{E}\left[\nabla_{\theta}\pi_{\theta}\nabla_{\mathrm{a}} A_{\pi_{\theta}}\right]$$

is based on contributions from a very small number of samples. Parameter updates become "infrequent and jumpy".

- ✓ Proposed policy relaxation techniques based on Interior-Point formulations, such that $\nabla_{\theta}\pi_{\theta} \neq 0$ almost everywhere
- ✓ Converge the policy to the true one over the learning



Outline

Reflections for today

Focus on Economic problems

- RLMPC is for performance
- Optimality "driven by external disturbances" seems the most interesting

Reflections for today

Focus on Economic problems

- RLMPC is for performance
- Optimality "driven by external disturbances" seems the most interesting

Not a competitor to other ideas

- Keep classic approaches!
- Combinations are possible and beneficial
- RL-MPC "milks" the performance of other approaches

Reflections for today

Focus on Economic problems

- RLMPC is for performance
- Optimality "driven by external disturbances" seems the most interesting

Not a competitor to other ideas

- Keep classic approaches!
- Combinations are possible and beneficial
- RL-MPC "milks" the performance of other approaches

RLMPC for constraint satisfaction

- Can "learn" to respect constraints
- Indirect approach, though
- ML-based "model-learning" better?

Reflections for today

Focus on Economic problems

- RLMPC is for performance
- Optimality "driven by external disturbances" seems the most interesting

Not a competitor to other ideas

- Keep classic approaches!
- Combinations are possible and beneficial
- RL-MPC "milks" the performance of other approaches

RLMPC for constraint satisfaction

- Can "learn" to respect constraints
- Indirect approach, though
- ML-based "model-learning" better?

Software integration is a bottleneck

- A lot of software for AI / RL
- Integration of MPC is not trivial

Reflections for today

Focus on Economic problems

- RLMPC is for performance
- Optimality "driven by external disturbances" seems the most interesting

Not a competitor to other ideas

- Keep classic approaches!
- Combinations are possible and beneficial
- RL-MPC "milks" the performance of other approaches

RLMPC for constraint satisfaction

- Can "learn" to respect constraints
- Indirect approach, though
- ML-based "model-learning" better?

Software integration is a bottleneck

- A lot of software for AI / RL
- Integration of MPC is not trivial

Prospects:

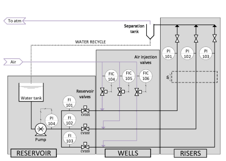
- Software
- Stochastic constraints
- Dual mode / Optimized exploration
- Data efficiency
- Multi-agent problems, FATE
- More applications
- Can we make it a "technology"?

Energy, Processes & Mobile robots

- Smart building
- Mobile robotics (UAV, USV)
- Wind energy
- Chemical process

- Smart house
- House with PV + Battery
- Energy Communities

Mix of experiments and simulations



Next lecture

When does the best model fit produce the optimal policy?
I.e. when can we expect "classic MPC" to give us the highest performance?

- Will do some repeats to put us in the right position to get there
- Introduce some "corollary" to the theory to explain our current understanding
- Show some basic examples

This is brand new lecture :-)

Thanks for your attention!