Adaptation of MPC via RL: fundamental principles

Sébastien Gros

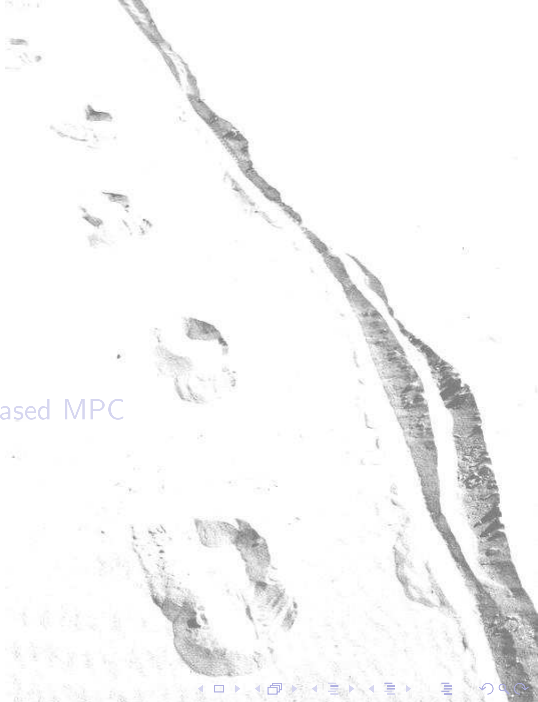
Dept. of Cybernetic, NTNU
Faculty of Information Tech.

Freiburg

Outline

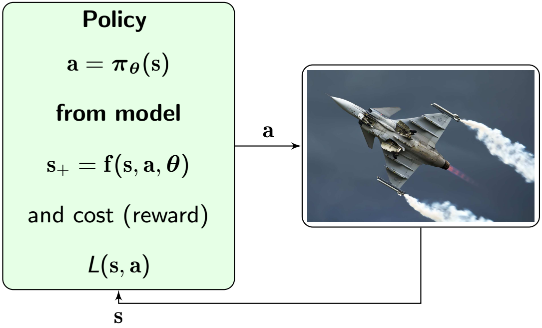
1. Forewords
2. MPC & MDPs
3. A central result on Learning-based MPC
4. RL for Learning-based MPC

Outline



1. Forewords
2. MPC & MDPs
3. A central result on Learning-based MPC
4. RL for Learning-based MPC

Why RL and MPC?



Why RL and MPC?

$\boldsymbol{\theta}$

Policy

$\mathbf{a} = \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s})$

from model

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

and cost (reward)

$L(\mathbf{s}, \mathbf{a})$

$\mathbf{a}$

$\mathbf{s}_+, \mathbf{s}, \mathbf{a}$

SYSID

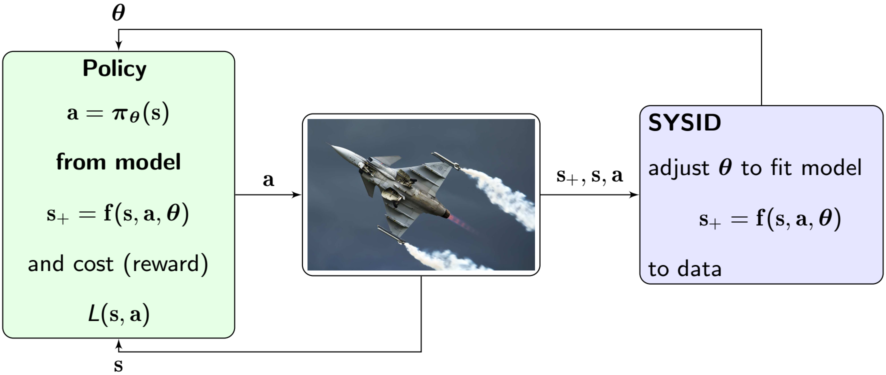
adjust $\boldsymbol{\theta}$ to fit model

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

to data

$\mathbf{s}$

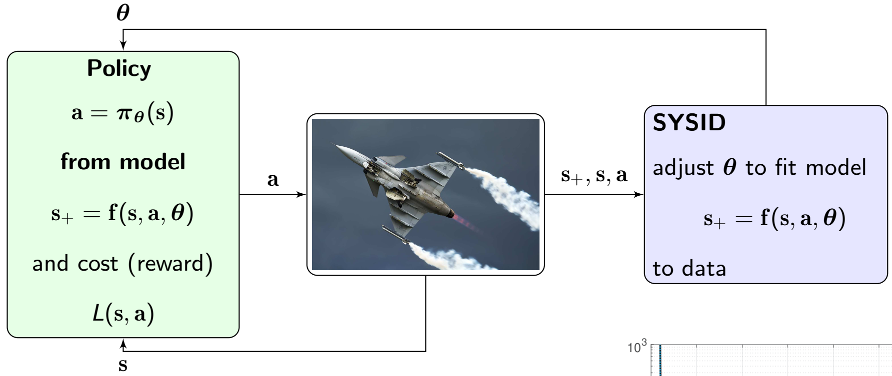
Why RL and MPC?



Does this work? Not necessarily...

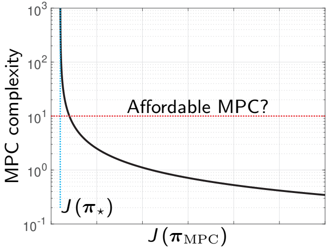
- Problem: does not capture the real system
- E.g. what $\mathbf{f}$ should be if real system is stochastic?
- Can degrade performance compared to keeping initial $\boldsymbol{\theta}$
- Well-known issue is data-based process optimization (RTO)
- Well-known issue in adaptive control

Why RL and MPC?

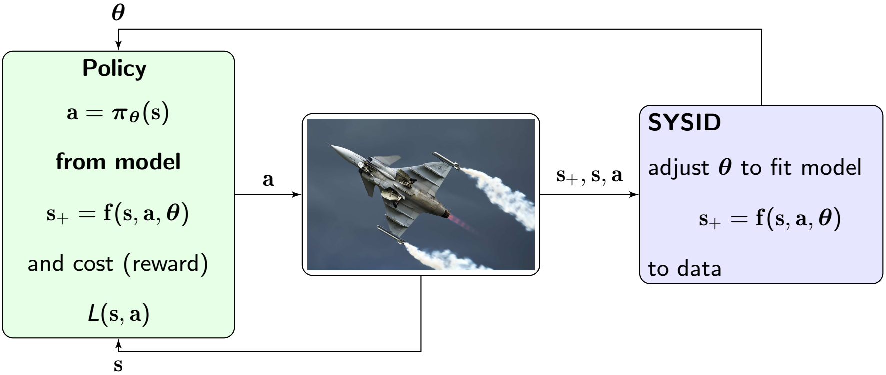


SYSID-like solutions

- Learn real dynamics $\mathbf{s}_{k+1} \sim \mathbb{P}\left[\,\cdot\,|\,\mathbf{s}_k, \mathbf{a}_k\,\right]$ using statistical tools (Gaussian Processes, RKHS, etc)
- Embed these statistical models in MPC
- Increments towards $\pi_\star$ via SYSID+MPC are "exponentially" costly



Why RL and MPC?



RL-MPC approach

- "Milk" the performance of the MPC scheme for a given MPC structure / modelling choice
- Focuses directly on closed-loop performance rather than on "ever better models"
- Not a competing strategy to "better models", can be used in combination

In this lecture: basic principles / Next lecture: recent results

Notation

- Real system dynamics

$$\mathbb{P}\left[\,\mathbf{s}_+\,|\,\mathbf{s},\mathbf{a}\,\right] \in \mathbb{R}_+$$

denotes the probability (density) of observing a transition from the state-action pair $\mathbf{s}, \mathbf{a}$ to the subsequent state $\mathbf{s}_+$

Notation

- Real system dynamics

$$\mathbb{P}\left[\,\mathbf{s}_+ \,|\, \mathbf{s}, \mathbf{a}\,\right] \in \mathbb{R}_+$$

denotes the probability (density) of observing a transition from the state-action pair $\mathbf{s}, \mathbf{a}$ to the subsequent state $\mathbf{s}_+$

- Cost (reward):

$$L\left(\mathbf{s}, \mathbf{a}\right) \in \mathbb{R}$$

assigns a value to each state-action pair. To be minimized here (RL often wants to maximize, no difference)

Notation

- Real system dynamics

$$\mathbb{P}\left[\mathbf{s}_+ \,|\, \mathbf{s}, \mathbf{a}\right] \in \mathbb{R}_+$$

denotes the probability (density) of observing a transition from the state-action pair $\mathbf{s}, \mathbf{a}$ to the subsequent state $\mathbf{s}_+$

- Cost (reward):

$$L\left(\mathbf{s}, \mathbf{a}\right) \in \mathbb{R}$$

assigns a value to each state-action pair. To be minimized here (RL often wants to maximize, no difference)

- Deterministic policy

$$\mathbf{a} = \boldsymbol{\pi}\left(\mathbf{s}\right)$$

maps a state $\mathbf{s}$ into an action $\mathbf{a}$

Notation

- Real system dynamics

$$\mathbb{P}\left[\,\mathbf{s}_+ \,|\, \mathbf{s}, \mathbf{a}\,\right] \in \mathbb{R}_+$$

denotes the probability (density) of observing a transition from the state-action pair $\mathbf{s}, \mathbf{a}$ to the subsequent state $\mathbf{s}_+$

- Cost (reward):

$$L\left(\mathbf{s}, \mathbf{a}\right) \in \mathbb{R}$$

assigns a value to each state-action pair. To be minimized here (RL often wants to maximize, no difference)

- Deterministic policy

$$\mathbf{a} = \boldsymbol{\pi}\left(\mathbf{s}\right)$$

maps a state $\mathbf{s}$ into an action $\mathbf{a}$

- Stochastic policy

$$\pi\left[\,\mathbf{a} \,|\, \mathbf{s}\,\right] \in \mathbb{R}_+$$

assigns the probability (density) of taking action $\mathbf{a}$ for a given state $\mathbf{s}$

Markov Decision Processes (MDP)

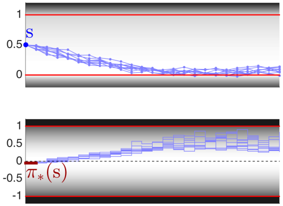
A very general way of describing optimal control

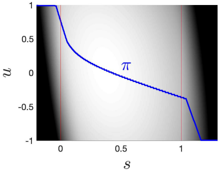
- Expected cost (return):

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

with discount $\gamma \in [0, 1]$

- Fixed or random initial conditions $\mathbf{s}_0$





State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

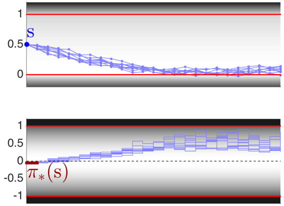
$$J(\pi) = \mathbb{E}_{\pi}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \pi(\mathbf{s}_k))\right]$$

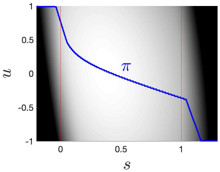
with discount $\gamma \in [0, 1]$

- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\pi_\star$ solution of

$$\min_{\pi} J(\pi)$$

(optimization over policies, i.e. functions)





State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

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

  with discount $\gamma \in [0, 1]$

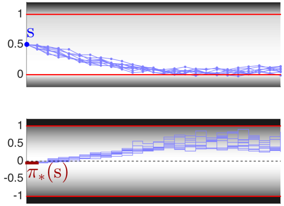
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

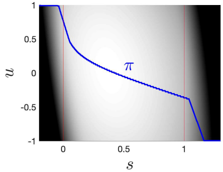
$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

  (optimization over policies, i.e. functions)

- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$





State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

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

with discount $\gamma \in [0, 1]$

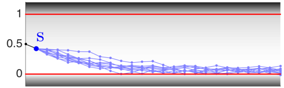
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

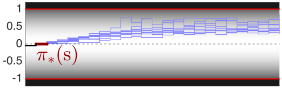
$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

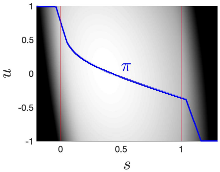
(optimization over policies, i.e. functions)

- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$







State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

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

  with discount $\gamma \in [0, 1]$

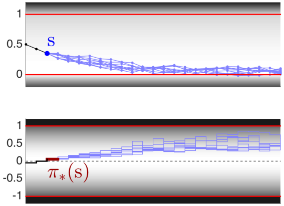
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

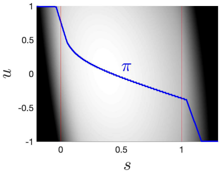
$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

  (optimization over policies, i.e. functions)

- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$





State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

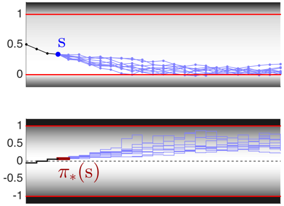
$$J(\boldsymbol{\pi}) = \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \boldsymbol{\pi}(\mathbf{s}_k))\right]$$

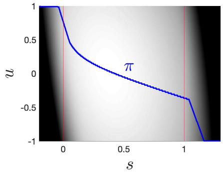
  with discount $\gamma \in [0, 1]$
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

  (optimization over policies, i.e. functions)
- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$





State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

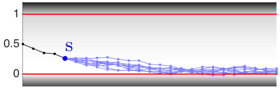
$$J(\boldsymbol{\pi}) = \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{\infty} \gamma^k L\left(\mathbf{s}_k, \boldsymbol{\pi}(\mathbf{s}_k)\right)\right]$$

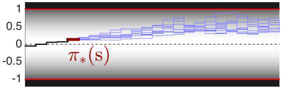
  with discount $\gamma \in [0, 1]$
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

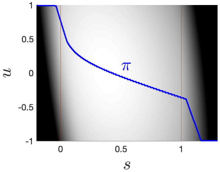
$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

  (optimization over policies, i.e. functions)
- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$







State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

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

with discount $\gamma \in [0, 1]$

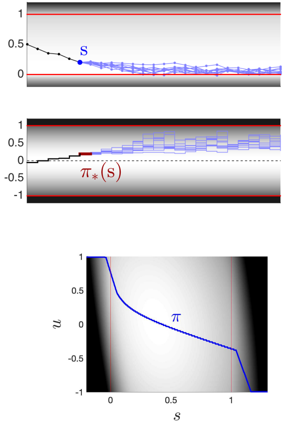
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

(optimization over policies, i.e. functions)

- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$



State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

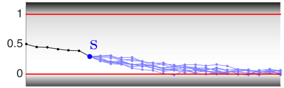
$$J(\boldsymbol{\pi}) = \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \boldsymbol{\pi}(\mathbf{s}_k))\right]$$

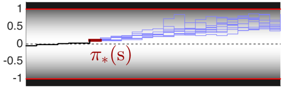
  with discount $\gamma \in [0, 1]$
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

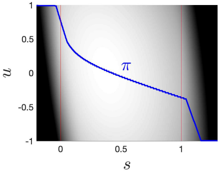
$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

  (optimization over policies, i.e. functions)
- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$







State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

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

with discount $\gamma \in [0, 1]$

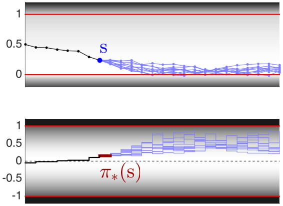
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

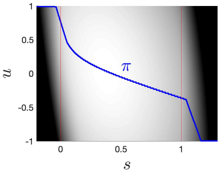
$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

(optimization over policies, i.e. functions)

- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$





State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

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

  with discount $\gamma \in [0, 1]$

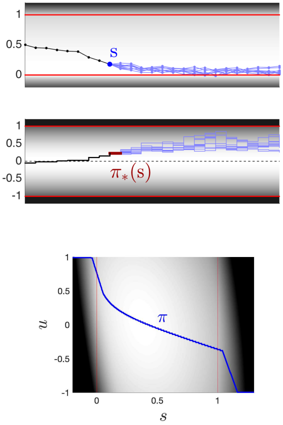
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

  (optimization over policies, i.e. functions)

- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$



State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

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

with discount $\gamma \in [0,1]$

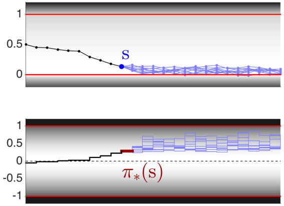
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

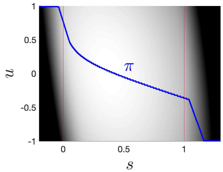
$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

(optimization over policies, i.e. functions)

- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$





State-action spaces can be continuous of discrete (e.g. integer)

Markov Decision Processes (MDP)

A very general way of describing optimal control

- Expected cost (return):

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

  with discount $\gamma \in [0, 1]$

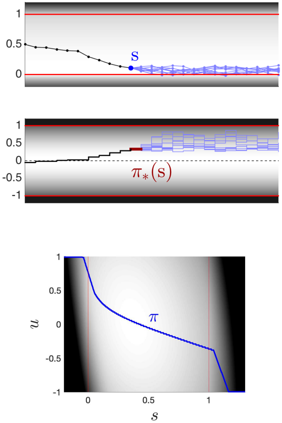
- Fixed or random initial conditions $\mathbf{s}_0$
- **MDP**: find $\boldsymbol{\pi}_\star$ solution of

$$\min_{\boldsymbol{\pi}} J(\boldsymbol{\pi})$$

  (optimization over policies, i.e. functions)

- Optimal parametrized policy $\boldsymbol{\pi}_{\boldsymbol{\theta}_\star}$ given by:

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$



State-action spaces can be continuous of discrete (e.g. integer)

Optimal Value Functions

- **Value function**:

$$V_\star(\mathbf{s}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty}\gamma^k L(\mathbf{s}_k, \mathbf{a}_k)\,\middle|\, \mathbf{s}_0 = \mathbf{s},\ \mathbf{a}_k = \boldsymbol{\pi}_\star(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}_\star$, starting from given initial conditions $\mathbf{s}$

Optimal Value Functions

- **Value function**:

$$V_\star(\mathbf{s}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{s}_0 = \mathbf{s},\ \mathbf{a}_k = \boldsymbol{\pi}_\star(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}_\star$, starting from given initial conditions $\mathbf{s}$

- **Action-Value function**:

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{s}_0 = \mathbf{s},\ \mathbf{a}_0 = \mathbf{a},\ \mathbf{a}_{k>0} = \boldsymbol{\pi}_\star(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}_\star$, starting from given initial conditions $\mathbf{s}$, and using action $\mathbf{a}$ as first input (policy $\boldsymbol{\pi}_\star$ after that)

Optimal Value Functions

- **Value function**:

$$V_\star(\mathbf{s}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty}\gamma^k L(\mathbf{s}_k,\mathbf{a}_k)\,\middle|\,\mathbf{s}_0=\mathbf{s},\ \mathbf{a}_k=\boldsymbol{\pi}_\star(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}_\star$, starting from given initial conditions $\mathbf{s}$

- **Action-Value function**:

$$Q_\star(\mathbf{s},\mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty}\gamma^k L(\mathbf{s}_k,\mathbf{a}_k)\,\middle|\,\mathbf{s}_0=\mathbf{s},\ \mathbf{a}_0=\mathbf{a},\ \mathbf{a}_{k>0}=\boldsymbol{\pi}_\star(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}_\star$, starting from given initial conditions $\mathbf{s}$, and using action $\mathbf{a}$ as first input (policy $\boldsymbol{\pi}_\star$ after that)

- **Relationship**:

$$V_\star(\mathbf{s}) = \min_{\mathbf{a}}\ Q_\star(\mathbf{s},\mathbf{a})$$

Optimal Value Functions

- **Value function**:

$$V_\star(\mathbf{s}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty}\gamma^k L(\mathbf{s}_k,\mathbf{a}_k)\,\middle|\, \mathbf{s}_0=\mathbf{s},\ \mathbf{a}_k=\boldsymbol{\pi}_\star(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}_\star$, starting from given initial conditions $\mathbf{s}$

- **Action-Value function**:

$$Q_\star(\mathbf{s},\mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty}\gamma^k L(\mathbf{s}_k,\mathbf{a}_k)\,\middle|\, \mathbf{s}_0=\mathbf{s},\ \mathbf{a}_0=\mathbf{a},\ \mathbf{a}_{k>0}=\boldsymbol{\pi}_\star(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}_\star$, starting from given initial conditions $\mathbf{s}$, and using action $\mathbf{a}$ as first input (policy $\boldsymbol{\pi}_\star$ after that)

- **Relationship**:

$$V_\star(\mathbf{s}) = \min_{\mathbf{a}}\ Q_\star(\mathbf{s},\mathbf{a})$$

- **Optimal Policy**:

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

Optimal Value Functions

- **Value function**:

$$V_\star(\mathbf{s}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{s}_0 = \mathbf{s},\ \mathbf{a}_k = \boldsymbol{\pi}_\star(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}_\star$, starting from given initial conditions $\mathbf{s}$

- **Action-Value function**:

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{s}_0 = \mathbf{s},\ \mathbf{a}_0 = \mathbf{a},\ \mathbf{a}_{k>0} = \boldsymbol{\pi}_\star(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}_\star$, starting from given initial conditions $\mathbf{s}$, and using action $\mathbf{a}$ as first input (policy $\boldsymbol{\pi}_\star$ after that)

- **Relationship**:

$$V_\star(\mathbf{s}) = \min_{\mathbf{a}}\ Q_\star(\mathbf{s}, \mathbf{a})$$

- **Optimal Policy**:

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

Can be computed via the Bellman equations, intractable for "large" state-action spaces

Value Functions

- **Value function**:

$$V_{\pi}(\mathbf{s}) = \mathbb{E}_{\pi}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{s}_0 = \mathbf{s},\ \mathbf{a}_k = \boldsymbol{\pi}(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}$, starting from given initial conditions $\mathbf{s}$

- **Action-Value function**:

$$Q_{\pi}(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\pi}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{s}_0 = \mathbf{s},\ \mathbf{a}_0 = \mathbf{a},\ \mathbf{a}_{k>0} = \boldsymbol{\pi}(\mathbf{s}_k)\right]$$

gives the expected cost for policy $\boldsymbol{\pi}$, starting from given initial conditions $\mathbf{s}$, and using action $\mathbf{a}$ as first input (policy $\boldsymbol{\pi}_\star$ after that)

- **Relationship**:

$$V_{\pi}(\mathbf{s}) = Q_{\pi}(\mathbf{s}, \boldsymbol{\pi}(\mathbf{s}_k))$$

- **Advantage function**:

$$A_{\pi}(\mathbf{s}, \mathbf{a}) = Q_{\pi}(\mathbf{s}, \mathbf{a}) - V_{\pi}(\mathbf{s})$$

compares $\mathbf{a}$ to policy $\boldsymbol{\pi}$. Instrumental in policy gradient methods.

Note:

$$V_{\pi} \neq V_\star$$
$$Q_{\pi} \neq Q_\star$$
$$A_{\pi} \neq A_\star$$

Can be computed via the Bellman equations, intractable for "large" state-action

MDPs and “forbidden” states

What if the system is not allowed to leave a certain subset of the state space?

MDPs and “forbidden” states

What if the system is not allowed to leave a certain subset of the state space?

- Say there is a “feasible” set:

$$\mathbb{F} = \{\ \mathbf{s} \quad | \quad \mathbf{h}(\mathbf{s}) \leq 0\ \}$$

  where the state of the system should always be.

MDPs and "forbidden" states

What if the system is not allowed to leave a certain subset of the state space?

- Say there is a "feasible" set:

$$\mathbb{F} = \{\ \mathbf{s} \quad | \quad \mathbf{h}(\mathbf{s}) \leq 0\ \}$$

  where the state of the system should always be.

- In the "MDP theory", assign an infinite penalty to leaving $\mathbb{F}$, i.e. add:

$$\mathrm{I}_{\mathbb{F}}(\mathbf{s}, \mathbf{a}) = \begin{cases} 0 & \text{if} \quad \mathbf{s} \in \mathbb{F} \\ +\infty & \text{if} \quad \mathbf{s} \notin \mathbb{F} \end{cases}$$

  to stage cost L.

MDPs and "forbidden" states

What if the system is not allowed to leave a certain subset of the state space?

- Say there is a "feasible" set:

$$\mathbb{F} = \{ \ \mathbf{s} \quad | \quad \mathbf{h}(\mathbf{s}) \leq 0 \ \}$$

  where the state of the system should always be.

- In the "MDP theory", assign an infinite penalty to leaving $\mathbb{F}$, i.e. add:

$$\mathrm{I}_{\mathbb{F}}(\mathbf{s}, \mathbf{a}) = \begin{cases} 0 & \text{if} \quad \mathbf{s} \in \mathbb{F} \\ +\infty & \text{if} \quad \mathbf{s} \notin \mathbb{F} \end{cases}$$

  to stage cost L.

- In RL, ∞ penalties are not meaningful: "There is no backup from death"

MDPs and "forbidden" states

What if the system is not allowed to leave a certain subset of the state space?

- Say there is a "feasible" set:

$$\mathbb{F} = \{\, \mathbf{s} \quad | \quad \mathbf{h}(\mathbf{s}) \leq 0 \,\}$$

  where the state of the system should always be.

- In the "MDP theory", assign an infinite penalty to leaving $\mathbb{F}$, i.e. add:

$$\mathrm{I}_{\mathbb{F}}(\mathbf{s}, \mathbf{a}) = \begin{cases} 0 & \text{if} \quad \mathbf{s} \in \mathbb{F} \\ +\infty & \text{if} \quad \mathbf{s} \notin \mathbb{F} \end{cases}$$

  to stage cost L.

- In RL, ∞ penalties are not meaningful: "There is no backup from death"
- Common approach: assign a "very large" penalty to $\mathbf{s} \notin \mathbb{F}$ instead of $+\infty$.

MDPs and "forbidden" states

What if the system is not allowed to leave a certain subset of the state space?

- Say there is a "feasible" set:

$$\mathbb{F} = \{ \mathbf{s} \quad | \quad \mathbf{h}(\mathbf{s}) \leq 0 \}$$

  where the state of the system should always be.

- In the "MDP theory", assign an infinite penalty to leaving $\mathbb{F}$, i.e. add:

$$\mathrm{I}_{\mathbb{F}}(\mathbf{s}, \mathbf{a}) = \begin{cases} 0 & \text{if} \quad \mathbf{s} \in \mathbb{F} \\ +\infty & \text{if} \quad \mathbf{s} \notin \mathbb{F} \end{cases}$$

  to stage cost L.

- In RL, ∞ penalties are not meaningful: "There is no backup from death"
- Common approach: assign a "very large" penalty to $\mathbf{s} \notin \mathbb{F}$ instead of $+\infty$.
- Use of "barrier functions" in RL

Outline

1. Forewords
2. MPC & MDPs
3. A central result on Learning-based MPC
4. RL for Learning-based MPC

A conceptual comparison...

MDP:

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

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

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

MPC & MDPs

A conceptual comparison...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

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

yields $\mathbf{a}^\star_{0,\ldots,N-1}\left(\mathbf{s}_0\right)$ and $\boldsymbol{\pi}_{\mathrm{MPC}}\left(\mathbf{s}_0\right) = \mathbf{a}^\star_0$

MPC & MDPs

A conceptual comparison...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \quad \sum_{k=0}^{\infty} L\left(\mathbf{s}_k, \mathbf{a}_k\right)$$

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

yields $\mathbf{a}^\star_{0,\ldots,\infty}\left(\mathbf{s}_0\right)$ and $\boldsymbol{\pi}_{\mathrm{MPC}}\left(\mathbf{s}_0\right) = \mathbf{a}^\star_0$

Assume:

- MPC has an infinite horizon

MPC & MDPs

A conceptual comparison...

MDP:

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

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

$$\mathbf{s}_{k+1} \sim \delta\left(\mathbf{s}_{k+1} - \mathbf{f}(\mathbf{s}_k, \mathbf{a}_k)\right)$$

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \quad \sum_{k=0}^{\infty} L(\mathbf{s}_k, \mathbf{a}_k)$$

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

yields $\mathbf{a}^\star_{0,\ldots,\infty}(\mathbf{s}_0)$ and $\boldsymbol{\pi}_{\mathrm{MPC}}(\mathbf{s}_0) = \mathbf{a}^\star_0$

Assume:

- MPC has an infinite horizon
- MDP has a deterministic dynamics $\mathbf{f}$

A conceptual comparison...

MDP:

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

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

$$\mathbf{s}_{k+1} \sim \delta\left(\mathbf{s}_{k+1} - \mathbf{f}(\mathbf{s}_k, \mathbf{a}_k)\right)$$

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \quad \sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k)$$

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

yields $\mathbf{a}^\star_{0,\ldots,\infty}(\mathbf{s}_0)$ and $\boldsymbol{\pi}_{\mathrm{MPC}}(\mathbf{s}_0) = \mathbf{a}^\star_0$

Assume:

- MPC has an infinite horizon
- MDP has a deterministic dynamics $\mathbf{f}$
- MPC is discounted

MPC & MDPs

A conceptual comparison...

MDP:

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

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

$$\mathbf{s}_{k+1} \sim \delta\left(\mathbf{s}_{k+1} - \mathbf{f}(\mathbf{s}_k, \mathbf{a}_k)\right)$$

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \quad \sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k)$$

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

yields $\mathbf{a}^\star_{0,\ldots,\infty}(\mathbf{s}_0)$ and $\boldsymbol{\pi}_{\mathrm{MPC}}(\mathbf{s}_0) = \mathbf{a}^\star_0$

Assume:

- MPC has an infinite horizon
- MDP has a deterministic dynamics $\mathbf{f}$
- MPC is discounted

Then (without model error):

$$\underbrace{\boldsymbol{\pi}^\star(\mathbf{s}_k)}_{\text{MDP solution}} = \underbrace{\mathbf{a}^\star_k(\mathbf{s}_0)}_{\text{MPC sequence}} = \underbrace{\mathbf{a}^\star_0(\mathbf{s}_k)}_{\text{MPC 1}^{\text{st}}\text{ control}} = \boldsymbol{\pi}_{\mathrm{MPC}}(\mathbf{s}_k)$$

on the trajectories $\mathbf{s}_{0,\ldots,\infty}$

MPC & MDPs

A conceptual comparison...

MDP:

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

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

$$\mathbf{s}_{k+1} \sim \delta\left(\mathbf{s}_{k+1} - \mathbf{f}\left(\mathbf{s}_k, \mathbf{a}_k\right)\right)$$

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \quad \sum_{k=0}^{\infty}\gamma^k L\left(\mathbf{s}_k, \mathbf{a}_k\right)$$

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

yields $\mathbf{a}^\star_{0,\ldots,\infty}\left(\mathbf{s}_0\right)$ and $\boldsymbol{\pi}_{\mathrm{MPC}}\left(\mathbf{s}_0\right) = \mathbf{a}^\star_0$

Assume:

- MPC has an infinite horizon
- MDP has a deterministic dynamics $\mathbf{f}$
- MPC is discounted

Then (without model error):

$$\underbrace{\boldsymbol{\pi}^\star\left(\mathbf{s}_k\right)}_{\text{MDP solution}} = \underbrace{\mathbf{a}^\star_k\left(\mathbf{s}_0\right)}_{\text{MPC sequence}} = \underbrace{\mathbf{a}^\star_0\left(\mathbf{s}_k\right)}_{\text{MPC } 1^{\text{st}} \text{ control}} = \boldsymbol{\pi}_{\mathrm{MPC}}\left(\mathbf{s}_k\right)$$

on the trajectories $\mathbf{s}_{0,\ldots,\infty}$

Bottom line: MPC provides optimal policy approximation (finite horizon, deterministic model), i.e. $\boldsymbol{\pi}_{\mathrm{MPC}} \approx \boldsymbol{\pi}_\star$

MPC & MDPs

A conceptual comparison...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

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

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

yields $\mathbf{a}^\star_{0,\dots,N-1}(\mathbf{s}_0)$ and $\boldsymbol{\pi}_{\mathrm{MPC}}(\mathbf{s}_0) = \mathbf{a}^\star_0$

Assume:

- MPC has an infinite horizon
- MDP has a deterministic dynamics $\mathbf{f}$
- MPC is discounted

Then (without model error):

$$\underbrace{\boldsymbol{\pi}^\star(\mathbf{s}_k)}_{\text{MDP solution}} = \underbrace{\mathbf{a}^\star_k(\mathbf{s}_0)}_{\text{MPC sequence}} = \underbrace{\mathbf{a}^\star_0(\mathbf{s}_k)}_{\text{MPC 1}^{\text{st}}\text{ control}} = \boldsymbol{\pi}_{\mathrm{MPC}}(\mathbf{s}_k)$$

on the trajectories $\mathbf{s}_{0,\dots,\infty}$

Bottom line: MPC provides optimal policy approximation (finite horizon, deterministic model), i.e. $\boldsymbol{\pi}_{\mathrm{MPC}} \approx \boldsymbol{\pi}_\star$

MPC with stochastic model: better approximation, higher computational cost

Why discounting?

MDP:

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

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

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

Discounting is (in general) needed to make the MDP well defined, is that all?

Why discounting?

MDP:

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

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

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

Discounting is (in general) needed to make the MDP well defined, is that all?

System lifetime: assuming that the system can (irremediably) fail at any time k with probability $1-\gamma$, then discounting accounts for resulting probabilistic lifetime.

Why discounting?

MDP:

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

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

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

Discounting is (in general) needed to make the MDP well defined, is that all?

System lifetime: assuming that the system can (irremediably) fail at any time k with probability $1-\gamma$, then discounting accounts for resulting probabilistic lifetime.

E.g. a system with a sampling time of 1 second, and a 90% chance of having a lifetime of 20 years, should have $\gamma = 0.999999996349275$

Why discounting?

MDP:

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

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

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

Discounting is (in general) needed to make the MDP well defined, is that all?

Investment model: expected economic growth r (per time unit) implies that earning at time k is worth $(1+r)^{-k}$ the same earning at time 0. Hence $\gamma = (1+r)^{-1}$.

Why discounting?

MDP:

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

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

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

Discounting is (in general) needed to make the MDP well defined, is that all?

Investment model: expected economic growth r (per time unit) implies that earning at time k is worth $(1+r)^{-k}$ the same earning at time 0. Hence $\gamma = (1+r)^{-1}$.

E.g. a system with a sampling time of 1 second and an expected return of 10% per year should have $\gamma = 0.999999999848887$

Why discounting?

MDP:

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

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

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

Discounting is (in general) needed to make the MDP well defined, is that all?

Investment model: expected economic growth r (per time unit) implies that earning at time k is worth $(1+r)^{-k}$ the same earning at time 0. Hence $\gamma = (1+r)^{-1}$.

E.g. a system with a sampling time of 1 second and an expected return of 10% per year should have $\gamma = 0.999999999848887$

Bottom line: on "engineering applications", the discount tends to (should) be extremely close to 1

Why discounting?

Gain optimal MDP:

$$\min_{\boldsymbol{\pi}} \quad \lim_{N\to\infty} \mathbb{E}_{\boldsymbol{\pi}}\left[\sum_{k=0}^{N} \frac{1}{N} L\left(\mathbf{s}_k, \mathbf{a}_k\right)\right]$$

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

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

What about considering average cost?

Policy π

- is said to achieve "gain optimality"
- transients are irrelelvant as they have no contribution in the average return
- tend to yield "bang-bang" actions until optimal steady state is reached
- is not unique!

Why discounting?

Gain optimal MDP:

$$\min_{\pi} \quad \lim_{N\to\infty} \mathbb{E}_{\pi}\left[\sum_{k=0}^{N} \frac{1}{N} L\left(\mathbf{s}_k, \mathbf{a}_k\right)\right]$$

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

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

What about considering average cost?

Policy π

- is said to achieve "gain optimality"
- transients are irrelelvant as they have no contribution in the average return
- tend to yield "bang-bang" actions until optimal steady state is reached
- is not unique!

... gain optimal are of questionable use for control

Why discounting?

Bias optimal MDP:

$$\min_{\pi} \quad \mathbb{E}_{\pi}\left[\sum_{k=0}^{N} L\left(\mathbf{s}_k, \mathbf{a}_k\right) - V_{\mathrm{G}}^{\star}\left(\mathbf{s}_0\right)\right]$$

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

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

What about "removing" the average cost?

where $V_{\mathrm{G}}^{\star}$ is the value function associated to gain optimal problem.

Policy π

- is said to achieve "bias optimality"
- "best transient to gain-optimal state"
- there are RL algorithms for bias optimality

Why discounting?

Bias optimal MDP:

$$\min_{\pi} \quad \mathbb{E}_{\pi}\left[\sum_{k=0}^{N} L\left(\mathbf{s}_k, \mathbf{a}_k\right) - V_{\mathrm{G}}^{\star}\left(\mathbf{s}_0\right)\right]$$

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

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

What about "removing" the average cost?

where $V_{\mathrm{G}}^{\star}$ is the value function associated to gain optimal problem.

Policy π

- is said to achieve "bias optimality"
- "best transient to gain-optimal state"
- there are RL algorithms for bias optimality

A New Framework for Computing Bias-Optimal Policies Using Discounted Reinforcement Learning, NeurIPS 2021, M. Zanon, S. Gros (submitted)

Outline

1 Forewords

2 MPC & MDPs

3 A central result on Learning-based MPC

4 RL for Learning-based MPC

MPC-based value functions

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

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

yields $\mathbf{a}^\star_{0,\ldots,N-1}\left(\mathbf{s}_0\right)$ and $\pi_{\mathrm{MPC}}\left(\mathbf{s}_0\right) = \mathbf{a}^\star_0$

MPC-based value functions

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

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

yields $\mathbf{a}^\star_{0,\ldots,N-1}(\mathbf{s}_0)$ and $\boldsymbol{\pi}_{\mathrm{MPC}}(\mathbf{s}_0) = \mathbf{a}^\star_0$

Value Functions:

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

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{a}_0 = \mathbf{a}\right]$$

MPC-based value functions

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

$$V_{\mathrm{MPC}}\left(\mathbf{s}_0\right) = \min_{\mathbf{s},\mathbf{a}} \quad \gamma^N T\left(\mathbf{s}_N\right) + \sum_{k=0}^{N-1}\gamma^k L\left(\mathbf{s}_k,\mathbf{a}_k\right)$$

$$\mathrm{s.t.} \quad \mathbf{s}_{k+1} = \mathbf{f}\left(\mathbf{s}_k,\mathbf{a}_k\right)$$

i.e. MPC scheme provides a value function

Value Functions:

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

$$Q_\star\left(\mathbf{s},\mathbf{a}\right) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty}\gamma^k L\left(\mathbf{s}_k,\mathbf{a}_k\right)\,\middle|\,\mathbf{a}_0 = \mathbf{a}\right]$$

- MPC delivers a value function V_{MPC}

MPC-based value functions

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

$$Q_{\mathrm{MPC}}\left(\mathbf{s}_0, \mathbf{a}\right) = \min_{\mathbf{s},\mathbf{a}} \quad \gamma^N T\left(\mathbf{s}_N\right) + \sum_{k=0}^{N-1} \gamma^k L\left(\mathbf{s}_k, \mathbf{a}_k\right)$$

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

$$\mathbf{a}_0 = \mathbf{a}$$

i.e. MPC scheme provides an action-value function

Value Functions:

$$V_\star\left(\mathbf{s}\right) = \mathbb{E}_{\pi_\star}\left[\sum_{k=0}^{\infty} \gamma^k L\left(\mathbf{s}_k, \mathbf{a}_k\right)\right]$$

$$Q_\star\left(\mathbf{s}, \mathbf{a}\right) = \mathbb{E}_{\pi_\star}\left[\sum_{k=0}^{\infty} \gamma^k L\left(\mathbf{s}_k, \mathbf{a}_k\right) \,\middle|\, \mathbf{a}_0 = \mathbf{a}\right]$$

- MPC delivers a value function V_{MPC}
- MPC (can) deliver an action-value function Q_{MPC}

MPC-based value functions

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

$$Q_{\mathrm{MPC}}\left(\mathbf{s}_0,\mathbf{a}\right) = \min_{\mathbf{s},\mathbf{a}} \quad \gamma^N T\left(\mathbf{s}_N\right) + \sum_{k=0}^{N-1}\gamma^k L\left(\mathbf{s}_k,\mathbf{a}_k\right)$$

$$\mathrm{s.t.} \quad \mathbf{s}_{k+1} = \mathbf{f}\left(\mathbf{s}_k,\mathbf{a}_k\right)$$

$$\mathbf{a}_0 = \mathbf{a}$$

i.e. MPC scheme provides an action-value function

Value Functions:

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

$$Q_\star\left(\mathbf{s},\mathbf{a}\right) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty}\gamma^k L\left(\mathbf{s}_k,\mathbf{a}_k\right)\,\middle|\,\mathbf{a}_0 = \mathbf{a}\right]$$

- MPC delivers a value function V_{MPC}
- MPC (can) deliver an action-value function Q_{MPC}
- MPC delivers a policy $\boldsymbol{\pi}_{\mathrm{MPC}}$

MPC-based value functions

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

$$Q_{\mathrm{MPC}}(\mathbf{s}_0, \mathbf{a}) = \min_{\mathbf{s},\mathbf{a}} \quad \gamma^N T(\mathbf{s}_N) + \sum_{k=0}^{N-1} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k)$$
$$\text{s.t.} \quad \mathbf{s}_{k+1} = \mathbf{f}(\mathbf{s}_k, \mathbf{a}_k)$$
$$\mathbf{a}_0 = \mathbf{a}$$

i.e. MPC scheme provides an action-value function

Value Functions:

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

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k)\,\middle|\, \mathbf{a}_0 = \mathbf{a}\right]$$

- MPC delivers a value function V_{MPC}
- MPC (can) deliver an action-value function Q_{MPC}
- MPC delivers a policy $\boldsymbol{\pi}_{\mathrm{MPC}}$
- Fundamental relationships satisfied:

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

MPC-based value functions

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

$$Q_{\mathrm{MPC}}(\mathbf{s}_0, \mathbf{a}) = \min_{\mathbf{s},\mathbf{a}} \quad \gamma^N T(\mathbf{s}_N) + \sum_{k=0}^{N-1} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k)$$

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

$$\mathbf{a}_0 = \mathbf{a}$$

i.e. MPC scheme provides an action-value function

Value Functions:

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

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{a}_0 = \mathbf{a}\right]$$

- MPC delivers a value function V_{MPC}
- MPC (can) deliver an action-value function Q_{MPC}
- MPC delivers a policy $\boldsymbol{\pi}_{\mathrm{MPC}}$
- Fundamental relationships satisfied:

$$V_{\mathrm{MPC}}(\mathbf{s}) = \min_{\mathbf{a}} \; Q_{\mathrm{MPC}}(\mathbf{s}, \mathbf{a})$$

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

Similarly to $\boldsymbol{\pi}_{\mathrm{MPC}} \approx \boldsymbol{\pi}_\star$:

$$V_{\mathrm{MPC}}(\mathbf{s}) \approx V_\star(\mathbf{s})$$

$$Q_{\mathrm{MPC}}(\mathbf{s}, \mathbf{a}) \approx Q_\star(\mathbf{s}, \mathbf{a})$$

A central result...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \gamma^N T\left(\mathbf{s}_N\right) + \sum_{k=0}^{N-1}\gamma^k L\left(\mathbf{s}_k,\mathbf{a}_k\right)$$
$$\text{s.t.} \quad \mathbf{s}_{k+1} = \mathbf{f}\left(\mathbf{s}_k,\mathbf{a}_k\right)$$

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

Value and Action-Value Functions:

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

$$Q_\star\left(\mathbf{s},\mathbf{a}\right) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty}\gamma^k L\left(\mathbf{s}_k,\mathbf{a}_k\right)\,\middle|\,\mathbf{a}_0 = \mathbf{a}\right]$$

A central result...

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

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

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

Value and Action-Value Functions:

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

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{a}_0 = \mathbf{a}\right]$$

MPC: ($\mathbf{s}_0$ given)

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

In general

$$\boldsymbol{\pi}_{\mathrm{MPC}} \neq \boldsymbol{\pi}_\star, \; V_{\mathrm{MPC}} \neq V_\star, \; Q_{\mathrm{MPC}} \neq Q_\star$$

but...

A central result...

MDP:

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

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

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

Value and Action-Value Functions:

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

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{a}_0 = \mathbf{a}\right]$$

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \gamma^N \tilde{T}(\mathbf{s}_N) + \sum_{k=0}^{N-1} \gamma^k \tilde{L}(\mathbf{s}_k, \mathbf{a}_k)$$

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

In general

$$\boldsymbol{\pi}_{\mathrm{MPC}} \neq \boldsymbol{\pi}_\star, \; V_{\mathrm{MPC}} \neq V_\star, \; Q_{\mathrm{MPC}} \neq Q_\star$$

but...

A central result...

MDP:

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

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

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

Value and Action-Value Functions:

$$V_\star(\mathbf{s}) = \mathbb{E}_{\pi_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k)\right]$$

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\pi_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{a}_0 = \mathbf{a}\right]$$

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \gamma^N \tilde{T}(\mathbf{s}_N) + \sum_{k=0}^{N-1} \gamma^k \tilde{L}(\mathbf{s}_k, \mathbf{a}_k)$$

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

yields π_{MPC}, V_{MPC}, and Q_{MPC}

Under some assumptions, there are $\tilde{L}$, $\tilde{T}$ s.t.

$$\pi_{\mathrm{MPC}} = \pi_\star,\; V_{\mathrm{MPC}} = V_\star,\; Q_{\mathrm{MPC}} = Q_\star$$

Assumption: trajectories of model $\mathbf{f}$ under optimal policy $\pi_\star$ should yield bounded $\gamma^k L(\mathbf{s}_k, \mathbf{a}_k)$ for $k = 0, \ldots, \infty$

A central result...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \gamma^N \tilde{T}(\mathbf{s}_N) + \sum_{k=0}^{N-1} \gamma^k \tilde{L}(\mathbf{s}_k, \mathbf{a}_k)$$

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

Value and Action-Value Functions:

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

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k)\,\middle|\, \mathbf{a}_0 = \mathbf{a}\right]$$

Under some assumptions, there are $\tilde{L}$, $\tilde{T}$ s.t.

$$\boldsymbol{\pi}_{\mathrm{MPC}} = \boldsymbol{\pi}_\star,\ V_{\mathrm{MPC}} = V_\star,\ Q_{\mathrm{MPC}} = Q_\star$$

Assumption: trajectories of model $\mathbf{f}$ under optimal policy $\boldsymbol{\pi}_\star$ should yield bounded $\gamma^k L(\mathbf{s}_k, \mathbf{a}_k)$ for $k = 0, \ldots, \infty$

- MPC can "capture" $\boldsymbol{\pi}_\star$, $Q_\star$, $V_\star$, **even if MPC model is inaccurate**
- Requires modifications of the stage cost & constraints
- Valid for all MPC schemes (classic, robust, stochastic, economic, etc)

A central result...

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

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

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

Value and Action-Value Functions:

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

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\boldsymbol{\pi}_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{a}_0 = \mathbf{a}\right]$$

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \gamma^N \tilde{T}(\mathbf{s}_N) + \sum_{k=0}^{N-1} \gamma^k \tilde{L}(\mathbf{s}_k, \mathbf{a}_k)$$
$$\text{s.t.} \quad \mathbf{s}_{k+1} = \mathbf{f}(\mathbf{s}_k, \mathbf{a}_k)$$

yields $\boldsymbol{\pi}_{\text{MPC}}$, V_{MPC}, and Q_{MPC}

Under some assumptions, there are $\tilde{L}$, $\tilde{T}$ s.t.

$$\boldsymbol{\pi}_{\text{MPC}} = \boldsymbol{\pi}_\star,\ V_{\text{MPC}} = V_\star,\ Q_{\text{MPC}} = Q_\star$$

Assumption: trajectories of model $\mathbf{f}$ under optimal policy $\boldsymbol{\pi}_\star$ should yield bounded $\gamma^k L(\mathbf{s}_k, \mathbf{a}_k)$ for $k = 0, \ldots, \infty$

- MPC can "capture" $\boldsymbol{\pi}_\star$, $Q_\star$, $V_\star$, **even if MPC model is inaccurate**
- Requires modifications of the stage cost & constraints
- Valid for all MPC schemes (classic, robust, stochastic, economic, etc)

Data-driven Economic NMPC using Reinforcement Learning, S. Gros, M. Zanon, Transaction on Automatic Control, 2019

A central result...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

$$\min_{\mathbf{s},\mathbf{a}} \gamma^N \tilde{T}(\mathbf{s}_N) + \sum_{k=0}^{N-1} \gamma^k \tilde{L}(\mathbf{s}_k, \mathbf{a}_k)$$

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

yields π_{MPC}, V_{MPC}, and Q_{MPC}

Value and Action-Value Functions:

$$V_\star(\mathbf{s}) = \mathbb{E}_{\pi_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k)\right]$$

$$Q_\star(\mathbf{s}, \mathbf{a}) = \mathbb{E}_{\pi_\star}\left[\sum_{k=0}^{\infty} \gamma^k L(\mathbf{s}_k, \mathbf{a}_k) \,\middle|\, \mathbf{a}_0 = \mathbf{a}\right]$$

Under some assumptions, there are $\tilde{L}$, $\tilde{T}$ s.t.

$$\pi_{\text{MPC}} = \pi_\star, \; V_{\text{MPC}} = V_\star, \; Q_{\text{MPC}} = Q_\star$$

Assumption: trajectories of model $\mathbf{f}$ under optimal policy $\pi_\star$ should yield bounded $\gamma^k L(\mathbf{s}_k, \mathbf{a}_k)$ for $k = 0, \ldots, \infty$

If you do (any) Learning+MPC and adjust the cost and/or constraints, then this paper is formally justifying what you are doing

Data-driven Economic NMPC using Reinforcement Learning, S. Gros, M. Zanon, Transaction on Automatic Control, 2019

Practical consequences...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

Practical consequences...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

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

yields $\boldsymbol{\pi}_{\text{MPC}}$, V_{MPC}, and Q_{MPC}

- In principle, it is possible to "modify" the MPC scheme such that it produces

$$\boldsymbol{\pi}_{\text{MPC}} = \boldsymbol{\pi}_\star,\; V_{\text{MPC}} = V_\star,\; Q_{\text{MPC}} = Q_\star$$

Practical consequences...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

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

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

- In principle, it is possible to "modify" the MPC scheme such that it produces

$$\boldsymbol{\pi}_{\mathrm{MPC}} = \boldsymbol{\pi}_\star,\ V_{\mathrm{MPC}} = V_\star,\ Q_{\mathrm{MPC}} = Q_\star$$

- Unfortunately, computing $\tilde{L}$, $\tilde{T}$ is as difficult as solving the Bellman equations

Practical consequences...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

- In principle, it is possible to "modify" the MPC scheme such that it produces

$$\boldsymbol{\pi}_{\mathrm{MPC}} = \boldsymbol{\pi}_\star,\ V_{\mathrm{MPC}} = V_\star,\ Q_{\mathrm{MPC}} = Q_\star$$

- Unfortunately, computing $\tilde{L}$, $\tilde{T}$ is as difficult as solving the Bellman equations
- Not very useful in practice, **unless** we are working in a "learning" context...

Practical consequences...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

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

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

- In principle, it is possible to "modify" the MPC scheme such that it produces

$$\boldsymbol{\pi}_{\mathrm{MPC}} = \boldsymbol{\pi}_\star,\ V_{\mathrm{MPC}} = V_\star,\ Q_{\mathrm{MPC}} = Q_\star$$

- Unfortunately, computing $\tilde{L}$, $\tilde{T}$ is as difficult as solving the Bellman equations
- Not very useful in practice, **unless** we are working in a "learning" context...
- Then $\tilde{L}$, $\tilde{T}$ is something that we learn from the closed-loop trajectories

Practical consequences...

MDP:

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

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

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

MPC: ($\mathbf{s}_0$ given)

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

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

- In principle, it is possible to "modify" the MPC scheme such that it produces

$$\boldsymbol{\pi}_{\mathrm{MPC}} = \boldsymbol{\pi}_\star,\ V_{\mathrm{MPC}} = V_\star,\ Q_{\mathrm{MPC}} = Q_\star$$

- Unfortunately, computing $\tilde{L}$, $\tilde{T}$ is as difficult as solving the Bellman equations
- Not very useful in practice, **unless** we are working in a "learning" context...
- Then $\tilde{L}$, $\tilde{T}$ is something that we learn from the closed-loop trajectories
- E.g. RL can be used to learn $\tilde{L}$, $\tilde{T}$ (+possibly MPC model)

Outline

1 Forewords

2 MPC & MDPs

3 A central result on Learning-based MPC

4 RL for Learning-based MPC

Classic RL vs. RL-MPC

MDP:

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

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

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

MPC:

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

Classic RL vs. RL-MPC

MDP:

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

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

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

MPC:

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

RL with DNN

- correct structure is unknown
- good initialization is difficult
- respecting constraints is difficult & implicit

Classic RL vs. RL-MPC

MDP:

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

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

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

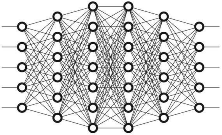
MPC:

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

yields $\boldsymbol{\pi}_{\mathrm{MPC}}$, V_{MPC}, and Q_{MPC}

RL with DNN

- correct structure is unknown
- good initialization is difficult
- respecting constraints is difficult & implicit



Classic RL vs. RL-MPC

MDP:

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

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

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

MPC:

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

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

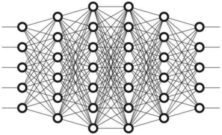
yields π_{MPC}, V_{MPC}, and Q_{MPC}

RL with DNN

- correct structure is unknown
- good initialization is difficult
- respecting constraints is difficult & implicit

MPC

- Provides $V_{\mathrm{MPC}} \equiv \hat{V}_\star$, $Q_{\mathrm{MPC}} \equiv \hat{Q}_\star$, $\pi_{\mathrm{MPC}} \equiv \hat{\pi}_\star$
- Structure and initialization given
- Constraints enforced explicitly
- Theory says that we can get $V_\star$, $Q_\star$, $\pi_\star$ from MPC



RL and MPC

Parametrized MPC:

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

yields $\boldsymbol{\pi}_{\boldsymbol{\theta}}$, $V_{\boldsymbol{\theta}}$, and $Q_{\boldsymbol{\theta}}$

RL: does

$$
\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})
$$

on the real system, where

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

RL and MPC

Parametrized MPC:

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

yields $\boldsymbol{\pi}_{\boldsymbol{\theta}}$, $V_{\boldsymbol{\theta}}$, and $Q_{\boldsymbol{\theta}}$

RL: does

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$

on the real system, where

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

- Parametrize all functions
- Constraints $\mathbf{h}_{\boldsymbol{\theta}}$ for forbidden state-actions

RL and MPC

Parametrized MPC:

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

yields $\boldsymbol{\pi}_{\boldsymbol{\theta}}$, $V_{\boldsymbol{\theta}}$, and $Q_{\boldsymbol{\theta}}$

- Parametrize all functions
- Constraints $\mathbf{h}_{\boldsymbol{\theta}}$ for forbidden state-actions

RL: does

$$\min_{\boldsymbol{\theta}} \; J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}\right)$$

on the real system, where

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

All RL techniques can be applied to an MPC scheme. RL adjusts the MPC parameters to minimize the closed-loop cost $J\left(\boldsymbol{\pi}_{\boldsymbol{\theta}}\right)$

RL and MPC

Parametrized MPC:

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

yields $\boldsymbol{\pi}_{\boldsymbol{\theta}}$, $V_{\boldsymbol{\theta}}$, and $Q_{\boldsymbol{\theta}}$

- Parametrize all functions
- Constraints $\mathbf{h}_{\boldsymbol{\theta}}$ for forbidden state-actions

All RL techniques can be applied to an MPC scheme. RL adjusts the MPC parameters to minimize the closed-loop cost $J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$

RL: does

$$\min_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$

on the real system, where

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

Good starting point: (MPC as usual)

- $L_{\boldsymbol{\theta}_0} = L$, $\mathbf{h}_{\boldsymbol{\theta}_0}$ selected according to the desired constraints
- $\mathbf{f}_{\boldsymbol{\theta}_0}$ selected from SYSID

but departing from that can help!!

RL and MPC

Parametrized MPC:

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

yields $\boldsymbol{\pi}_{\boldsymbol{\theta}}$, $V_{\boldsymbol{\theta}}$, and $Q_{\boldsymbol{\theta}}$

- Parametrize all functions
- Constraints $\mathbf{h}_{\boldsymbol{\theta}}$ for forbidden state-actions

All RL techniques can be applied to an MPC scheme. RL adjusts the MPC parameters to minimize the closed-loop cost $J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$

RL: does

$$\min_{\boldsymbol{\theta}} \; J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$$

on the real system, where

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

Good starting point: (MPC as usual)

- $L_{\boldsymbol{\theta}_0} = L$, $\mathbf{h}_{\boldsymbol{\theta}_0}$ selected according to the desired constraints
- $\mathbf{f}_{\boldsymbol{\theta}_0}$ selected from SYSID

but departing from that can help!!

Note: MPC model tuning via RL $\neq$ SYSID

RL methods - Reminder

Form function approximators:

$$Q_{\boldsymbol{\theta}}\left(\mathbf{s},\mathbf{a}\right),\quad V_{\boldsymbol{\theta}}\left(\mathbf{s}\right),\quad \boldsymbol{\pi}_{\boldsymbol{\theta}}\left(\mathbf{s}\right)$$

via ad-hoc parametrization

RL methods - Reminder

Form function approximators:

$$Q_{\boldsymbol{\theta}}(\mathrm{s}, \mathbf{a}), \quad V_{\boldsymbol{\theta}}(\mathrm{s}), \quad \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$$

via ad-hoc parametrization

- *Q*-**learning methods** adjust $\boldsymbol{\theta}$ to get

$$Q_{\boldsymbol{\theta}}(\mathrm{s}, \mathbf{a}) \approx Q_\star(\mathrm{s}, \mathbf{a})$$

Yields policy:

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

E.g. basic Q-learning uses:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha\delta\nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}(\mathrm{s}_k, \mathbf{a}_k)$$

$$\delta = L(\mathrm{s}_k, \mathbf{a}_k) + \gamma V_{\boldsymbol{\theta}}(\mathrm{s}_{k+1}) - Q_{\boldsymbol{\theta}}(\mathrm{s}_k, \mathbf{a}_k)$$

RL methods - Reminder

Form function approximators:

$$Q_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{a}), \quad V_{\boldsymbol{\theta}}(\mathbf{s}), \quad \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s})$$

via ad-hoc parametrization

- *Q*-**learning methods** adjust $\boldsymbol{\theta}$ to get

$$Q_{\boldsymbol{\theta}}(\mathbf{s}, \mathbf{a}) \approx Q_{\star}(\mathbf{s}, \mathbf{a})$$

Yields policy:

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

E.g. basic Q-learning uses:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha\delta\nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}(\mathbf{s}_k, \mathbf{a}_k)$$

$$\delta = L(\mathbf{s}_k, \mathbf{a}_k) + \gamma V_{\boldsymbol{\theta}}(\mathbf{s}_{k+1}) - Q_{\boldsymbol{\theta}}(\mathbf{s}_k, \mathbf{a}_k)$$

- **Policy gradient methods** adjust $\boldsymbol{\theta}$ to get

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}}) = 0$$

yields policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{x}) \approx \boldsymbol{\pi}_{\star}(\mathbf{x})$ directly.

RL methods - Reminder

Form function approximators:

$$Q_{\boldsymbol{\theta}}(\mathrm{s}, \mathbf{a}), \quad V_{\boldsymbol{\theta}}(\mathrm{s}), \quad \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$$

via ad-hoc parametrization

- Q-**learning methods** adjust $\boldsymbol{\theta}$ to get

$$Q_{\boldsymbol{\theta}}(\mathrm{s}, \mathbf{a}) \approx Q_{\star}(\mathrm{s}, \mathbf{a})$$

Yields policy:

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

E.g. basic Q-learning uses:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha\delta\nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}(\mathrm{s}_k, \mathbf{a}_k)$$

$$\delta = L(\mathrm{s}_k, \mathbf{a}_k) + \gamma V_{\boldsymbol{\theta}}(\mathrm{s}_{k+1}) - Q_{\boldsymbol{\theta}}(\mathrm{s}_k, \mathbf{a}_k)$$

- **Policy gradient methods** adjust $\boldsymbol{\theta}$ to get

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}}) = 0$$

yields policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{x}) \approx \boldsymbol{\pi}_{\star}(\mathbf{x})$ directly. E.g.

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

- **Derivative-free methods**
  - Build a surrogate of $J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$
  - Optimize over that model
  - Difficult over large parameter spaces

RL methods - Reminder

Form function approximators:

$$Q_{\boldsymbol{\theta}}(\mathbf{s},\mathbf{a}), \quad V_{\boldsymbol{\theta}}(\mathbf{s}), \quad \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{s})$$

via ad-hoc parametrization

Derivative-based methods require $Q_{\boldsymbol{\theta}}$, $V_{\boldsymbol{\theta}}$, $\boldsymbol{\pi}_{\boldsymbol{\theta}}$ and computing their sensitivities (i.e. $\nabla_{\boldsymbol{\theta}}$ or $\frac{\partial}{\partial \boldsymbol{\theta}}$)

- *Q*-**learning methods** adjust $\boldsymbol{\theta}$ to get

$$Q_{\boldsymbol{\theta}}(\mathbf{s},\mathbf{a}) \approx Q_\star(\mathbf{s},\mathbf{a})$$

Yields policy:

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

E.g. basic Q-learning uses:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha\delta\nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}(\mathbf{s}_k,\mathbf{a}_k)$$

$$\delta = L(\mathbf{s}_k,\mathbf{a}_k) + \gamma V_{\boldsymbol{\theta}}(\mathbf{s}_{k+1}) - Q_{\boldsymbol{\theta}}(\mathbf{s}_k,\mathbf{a}_k)$$

- **Policy gradient methods** adjust $\boldsymbol{\theta}$ to get

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}}) = 0$$

yields policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{x}) \approx \boldsymbol{\pi}_\star(\mathbf{x})$ directly. E.g.

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

- **Derivative-free methods**
  - Build a surrogate of $J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$
  - Optimize over that model
  - Difficult over large parameter spaces

RL methods - Reminder

Form function approximators:

$$Q_{\boldsymbol{\theta}}(\mathrm{s}, \mathbf{a}), \quad V_{\boldsymbol{\theta}}(\mathrm{s}), \quad \boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathrm{s})$$

via ad-hoc parametrization

Derivative-based methods require $Q_{\boldsymbol{\theta}}$, $V_{\boldsymbol{\theta}}$, $\boldsymbol{\pi}_{\boldsymbol{\theta}}$ and computing their sensitivities (i.e. $\nabla_{\boldsymbol{\theta}}$ or $\frac{\partial}{\partial \boldsymbol{\theta}}$)

In the RL-MPC context, $Q_{\boldsymbol{\theta}}$, $V_{\boldsymbol{\theta}}$, $\boldsymbol{\pi}_{\boldsymbol{\theta}}$ are coming from an **MPC scheme, typically cast as Nonlinear Program. What about the sensitivities?**

- Q**-learning methods** adjust $\boldsymbol{\theta}$ to get

$$Q_{\boldsymbol{\theta}}(\mathrm{s}, \mathbf{a}) \approx Q_\star(\mathrm{s}, \mathbf{a})$$

Yields policy:

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

E.g. basic Q-learning uses:

$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \alpha\delta\nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}(\mathrm{s}_k, \mathbf{a}_k)$$

$$\delta = L(\mathrm{s}_k, \mathbf{a}_k) + \gamma V_{\boldsymbol{\theta}}(\mathrm{s}_{k+1}) - Q_{\boldsymbol{\theta}}(\mathrm{s}_k, \mathbf{a}_k)$$

- **Policy gradient methods** adjust $\boldsymbol{\theta}$ to get

$$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\pi}_{\boldsymbol{\theta}}) = 0$$

yields policy $\boldsymbol{\pi}_{\boldsymbol{\theta}}(\mathbf{x}) \approx \boldsymbol{\pi}_\star(\mathbf{x})$ directly. E.g.

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

- **Derivative-free methods**
  - Build a surrogate of $J(\boldsymbol{\pi}_{\boldsymbol{\theta}})$
  - Optimize over that model
  - Difficult over large parameter spaces

Implementation of Basic RL Algorithms for MPC

MPC is a Nonlinear Program

Optimal value

$$\begin{aligned} V_{\boldsymbol{\theta}}(\mathrm{s}) = \min_{\mathbf{w}} \quad & \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) \\ \mathrm{s.t.} \quad & \mathbf{g}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) = 0 \\ & \mathbf{h}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) \leq 0 \end{aligned}$$

Optimal solution

$$\begin{aligned} \mathbf{w}^{\star}_{\boldsymbol{\theta}}(\mathrm{s}) = \mathrm{a}\min_{\mathbf{w}} \quad & \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) \\ \mathrm{s.t.} \quad & \dots \end{aligned}$$

Implementation of Basic RL Algorithms for MPC

MPC is a Nonlinear Program

Optimal value

$$V_{\boldsymbol{\theta}}(\mathrm{s}) = \min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta})$$
$$\text{s.t.} \quad \mathbf{g}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) = 0$$
$$\mathbf{h}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) \leq 0$$

Optimal solution

$$\mathbf{w}^{\star}_{\boldsymbol{\theta}}(\mathrm{s}) = \mathrm{a}\min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta})$$
$$\text{s.t.} \quad \ldots$$

How to obtain:

$$\nabla_{\boldsymbol{\theta}} V_{\boldsymbol{\theta}}, \ \nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}, \ \nabla_{\boldsymbol{\theta}} \mathbf{w}^{\star}_{\boldsymbol{\theta}}$$

?

Implementation of Basic RL Algorithms for MPC

MPC is a Nonlinear Program

Optimal value

$$V_{\boldsymbol{\theta}}(\mathrm{s}) = \min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta})$$
$$\text{s.t.} \quad \mathbf{g}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) = 0$$
$$\mathbf{h}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) \leq 0$$

Optimal solution

$$\mathbf{w}^\star_{\boldsymbol{\theta}}(\mathrm{s}) = \mathrm{a}\min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta})$$
$$\text{s.t.} \quad \ldots$$

How to obtain:

$$\nabla_{\boldsymbol{\theta}} V_{\boldsymbol{\theta}}, \; \nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}, \; \nabla_{\boldsymbol{\theta}} \mathbf{w}^\star_{\boldsymbol{\theta}}$$

?

NLP solution satisfies (KKT conditions)

$$\mathbf{r} = \begin{bmatrix} \nabla_{\mathbf{w}} \mathcal{L} \\ \mathbf{g} \\ \mathbf{h}_i \boldsymbol{\mu}_i \end{bmatrix} = 0$$
$$\mathbf{h} \leq 0, \; \boldsymbol{\mu} \geq 0$$

where Lagrange function is

$$\mathcal{L} = \Phi + \boldsymbol{\lambda}^\top \mathbf{g} + \boldsymbol{\mu}^\top \mathbf{h}$$

and $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ are "auxiliary variables" (multipliers)

Implementation of Basic RL Algorithms for MPC

MPC is a Nonlinear Program

Optimal value

$$V_{\boldsymbol{\theta}}(\mathrm{s}) = \min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta})$$
$$\text{s.t.} \quad \mathbf{g}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) = 0$$
$$\mathbf{h}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) \leq 0$$

Optimal solution

$$\mathbf{w}^{\star}_{\boldsymbol{\theta}}(\mathrm{s}) = \mathrm{a}\min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta})$$
$$\text{s.t.} \quad \ldots$$

How to obtain:

$$\nabla_{\boldsymbol{\theta}} V_{\boldsymbol{\theta}}, \;\; \nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}, \;\; \nabla_{\boldsymbol{\theta}} \mathbf{w}^{\star}_{\boldsymbol{\theta}}$$

?

NLP solution satisfies (KKT conditions)

$$\mathbf{r} = \begin{bmatrix} \nabla_{\mathbf{w}} \mathcal{L} \\ \mathbf{g} \\ \mathbf{h}_i \boldsymbol{\mu}_i \end{bmatrix} = 0$$
$$\mathbf{h} \leq 0, \; \boldsymbol{\mu} \geq 0$$

where Lagrange function is

$$\mathcal{L} = \Phi + \boldsymbol{\lambda}^{\top} \mathbf{g} + \boldsymbol{\mu}^{\top} \mathbf{h}$$

and $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ are "auxiliary variables" (multipliers)

Implementation of Basic RL Algorithms for MPC

MPC is a Nonlinear Program

Optimal value

$$\begin{aligned} V_{\boldsymbol{\theta}}(\mathrm{s}) = \min_{\mathbf{w}} \quad & \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) \\ \text{s.t.} \quad & \mathbf{g}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) = 0 \\ & \mathbf{h}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) \leq 0 \end{aligned}$$

Optimal solution

$$\begin{aligned} \mathbf{w}^\star_{\boldsymbol{\theta}}(\mathrm{s}) = \mathrm{a}\min_{\mathbf{w}} \quad & \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) \\ \text{s.t.} \quad & \ldots \end{aligned}$$

How to obtain:

$$\nabla_{\boldsymbol{\theta}} V_{\boldsymbol{\theta}}, \;\; \nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}, \;\; \nabla_{\boldsymbol{\theta}} \mathbf{w}^\star_{\boldsymbol{\theta}}$$

?

NLP solution satisfies (KKT conditions)

$$\mathbf{r} = \begin{bmatrix} \nabla_{\mathbf{w}} \mathcal{L} \\ \mathbf{g} \\ \mathbf{h}_i \boldsymbol{\mu}_i \end{bmatrix} = 0$$

$$\mathbf{h} \leq 0, \; \boldsymbol{\mu} \geq 0$$

where Lagrange function is

$$\mathcal{L} = \Phi + \boldsymbol{\lambda}^\top \mathbf{g} + \boldsymbol{\mu}^\top \mathbf{h}$$

and $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ are "auxiliary variables" (multipliers)

Solve NLP for $\mathrm{x}, \boldsymbol{\theta}$, provides $\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\mu}$, then:

$$\nabla_{\boldsymbol{\theta}} V_{\boldsymbol{\theta}}(\mathrm{s}) = \nabla_{\boldsymbol{\theta}} \mathcal{L}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}, \boldsymbol{\lambda}, \boldsymbol{\mu})$$

is a simple function evaluation

Implementation of Basic RL Algorithms for MPC

MPC is a Nonlinear Program

Optimal value

$$V_{\boldsymbol{\theta}}(\mathbf{s}) = \min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathbf{s}, \boldsymbol{\theta})$$
$$\text{s.t.} \quad \mathbf{g}(\mathbf{w}, \mathbf{s}, \boldsymbol{\theta}) = 0$$
$$\mathbf{h}(\mathbf{w}, \mathbf{s}, \boldsymbol{\theta}) \leq 0$$

Optimal solution

$$\mathbf{w}^\star_{\boldsymbol{\theta}}(\mathbf{s}) = \mathrm{a}\min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathbf{s}, \boldsymbol{\theta})$$
$$\text{s.t.} \quad \ldots$$

How to obtain:

$$\nabla_{\boldsymbol{\theta}} V_{\boldsymbol{\theta}}, \; \nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}, \; \nabla_{\boldsymbol{\theta}} \mathbf{w}^\star_{\boldsymbol{\theta}}$$

?

NLP solution satisfies (KKT conditions)

$$\mathbf{r} = \begin{bmatrix} \nabla_{\mathbf{w}} \mathcal{L} \\ \mathbf{g} \\ \mathbf{h}_i \boldsymbol{\mu}_i \end{bmatrix} = 0$$
$$\mathbf{h} \leq 0, \; \boldsymbol{\mu} \geq 0$$

where Lagrange function is

$$\mathcal{L} = \Phi + \boldsymbol{\lambda}^\top \mathbf{g} + \boldsymbol{\mu}^\top \mathbf{h}$$

and $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ are "auxiliary variables" (multipliers)

Solve NLP for $\mathbf{s}, \boldsymbol{\theta}$, provides $\mathbf{w}, \boldsymbol{\lambda}, \boldsymbol{\mu}$, then:

$$\frac{\partial \mathbf{w}^\star_{\boldsymbol{\theta}}}{\partial \boldsymbol{\theta}} = -\frac{\partial \mathbf{r}}{\partial \mathbf{w}}^{-1} \frac{\partial \mathbf{r}}{\partial \boldsymbol{\theta}}$$

with $\frac{\partial \mathbf{r}}{\partial \mathbf{w}}^{-1}$ already built in the solver, exists if LICQ / SOSC

Implementation of Basic RL Algorithms for MPC

MPC is a Nonlinear Program

Optimal value

$$V_{\boldsymbol{\theta}}(\mathrm{s}) = \min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta})$$
$$\text{s.t.} \quad \mathbf{g}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) = 0$$
$$\mathbf{h}(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta}) \leq 0$$

Optimal solution

$$\mathbf{w}^\star_{\boldsymbol{\theta}}(\mathrm{s}) = \mathrm{a}\min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathrm{s}, \boldsymbol{\theta})$$
$$\text{s.t.} \quad \ldots$$

How to obtain:

$$\nabla_{\boldsymbol{\theta}} V_{\boldsymbol{\theta}}, \ \nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}, \ \nabla_{\boldsymbol{\theta}} \mathbf{w}^\star_{\boldsymbol{\theta}}$$

?

NLP solution satisfies (KKT conditions)

$$\mathbf{r} = \begin{bmatrix} \nabla_{\mathbf{w}} \mathcal{L} \\ \mathbf{g} \\ \mathbf{h}_i \boldsymbol{\mu}_i \end{bmatrix} = 0$$
$$\mathbf{h} \leq 0, \ \boldsymbol{\mu} \geq 0$$

where Lagrange function is

$$\mathcal{L} = \Phi + \boldsymbol{\lambda}^\top \mathbf{g} + \boldsymbol{\mu}^\top \mathbf{h}$$

and $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ are “auxiliary variables” (multipliers)

Sensitivities do not exist for all s, a.
Does that matter?

Implementation of Basic RL Algorithms for MPC

MPC is a Nonlinear Program

Optimal value

$$V_{\boldsymbol{\theta}}(\mathbf{s}) = \min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathbf{s}, \boldsymbol{\theta}) \quad \text{s.t.} \quad \mathbf{g}(\mathbf{w}, \mathbf{s}, \boldsymbol{\theta}) = 0, \quad \mathbf{h}(\mathbf{w}, \mathbf{s}, \boldsymbol{\theta}) \leq 0$$

Optimal solution

$$\mathbf{w}^\star_{\boldsymbol{\theta}}(\mathbf{s}) = \mathrm{a}\min_{\mathbf{w}} \quad \Phi(\mathbf{w}, \mathbf{s}, \boldsymbol{\theta}) \quad \text{s.t.} \quad \dots$$

How to obtain:

$$\nabla_{\boldsymbol{\theta}} V_{\boldsymbol{\theta}}, \ \nabla_{\boldsymbol{\theta}} Q_{\boldsymbol{\theta}}, \ \nabla_{\boldsymbol{\theta}} \mathbf{w}^\star_{\boldsymbol{\theta}}$$

?

NLP solution satisfies (KKT conditions)

$$\mathbf{r} = \begin{bmatrix} \nabla_{\mathbf{w}} \mathcal{L} \\ \mathbf{g} \\ \mathbf{h}_i \boldsymbol{\mu}_i \end{bmatrix} = 0$$

$$\mathbf{h} \leq 0, \ \boldsymbol{\mu} \geq 0$$

where Lagrange function is

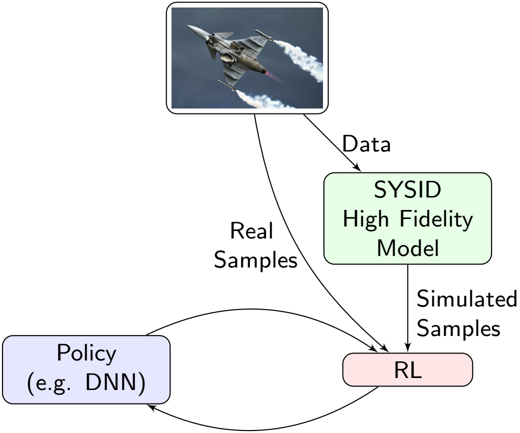
$$\mathcal{L} = \Phi + \boldsymbol{\lambda}^\top \mathbf{g} + \boldsymbol{\mu}^\top \mathbf{h}$$

and $\boldsymbol{\lambda}$, $\boldsymbol{\mu}$ are "auxiliary variables" (multipliers)

Sensitivities do not exist for all $\mathbf{s}, \mathbf{a}$.
Does that matter?

In general no: they exist *almost everywhere*, and always appear inside $\mathbb{E}[\cdot]$. If the MDP has well-defined underlying densities, then we are good.

Model-based RL methods vs. RL-MPC: Data flow



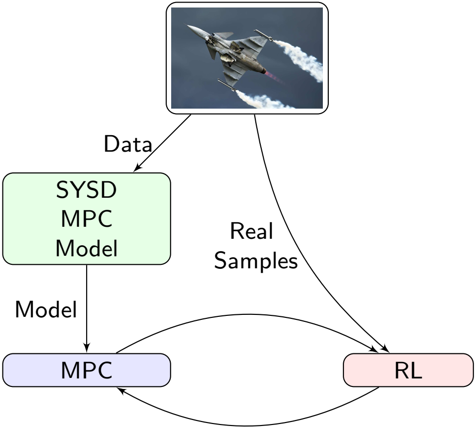
Common setup for "classic RL:

- Build statistical model of the real system
- Generate simulated samples
- Feed RL with real and simulated samples

Remarks:

- Simulated data much cheaper than real ones, most data will be simulated ones
- With mostly simulated data:
  - ≈equivalent to approximate DP
  - policy optimality relies on model quality

Model-based RL methods vs. RL-MPC: Data flow



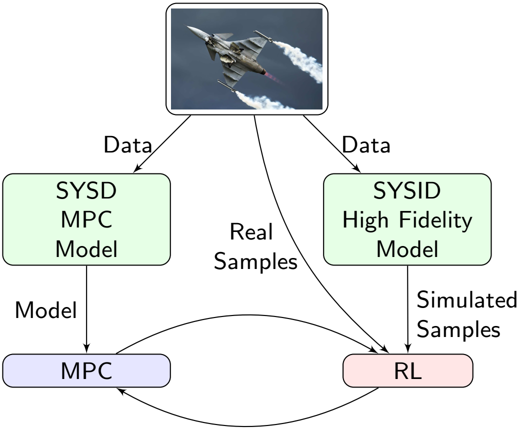
Basic setup for "RL-MPC":

- Build MPC model of the real system
- Pass it to MPC scheme
- Feed RL with real samples

Remarks:

- RL tunes MPC for real system
- MPC model may be "detuned" from SYSID version
- Real data are expensive...

Model-based RL methods vs. RL-MPC: Data flow



"Mixed" setup for "RL-MPC":

- Build MPC model of the real system
- MPC model is typically "simple"
- Build statistical model of the real system
- Generate simulated samples
- Feed RL with real and simulated samples

Remarks:

- Simple MPC model
- Complex simulation model
- MPC model may be "detuned" from SYSID version

What did we discuss?

- Learning-based MPC: we accept that the MPC model will never be "right", seek closed-loop performance rather than model fitting
- MPC serves as a policy & value functions approximation. This is a classic object in RL, but MPC is highly structured, while classic approximations in RL are not.
- Modifying the MPC cost and constraints allows MPC to be close-to optimality despite inaccurate model
- ... but it is also formally justified: in principle it allows to capture the optimal policy and value functions with a wrong model
- We discussed how to implement RL methods on MPC (basics)
- There is still room for high-fidelity modelling, can be used to produce virtual training data

So what's next?

- Stability of MPC under learning?
- Safety of MPC under learning?
- General MPC stability theory for deterministic, undiscounted problems. How to extend it to MDPs?
- Some more results:
    - Bias in policy gradient methods with constrained policies
    - Combining RL and SYSID?
    - RL and MPC for mixed-integer problems?
    - RL and MPC with state observers?
    - RL and MPC with strongly economic policies?
    - RL for tuning the '"meta" MPC parameters?