Parameterized Differential Dynamic Programming

Robotics: Science and Systems, June 29, 2022

Alex Oshin, Matthew D. Houghton, Michael J. Acheson,
Irene M. Gregory, Evangelos A. Theodorou

Urban Air Mobility

Impact:

- $9.6B - $28.3B industry by 2030
- Scaled transportation
- Air-Taxis
- Robotic Cargo Transportation

Planner Specifications:

- Principled Solutions/Guarantees
- Handle dynamical constraints
- Robust to epistemic uncertainty
- Multiple operational modes and flight regimes
- Transferable to different vehicles

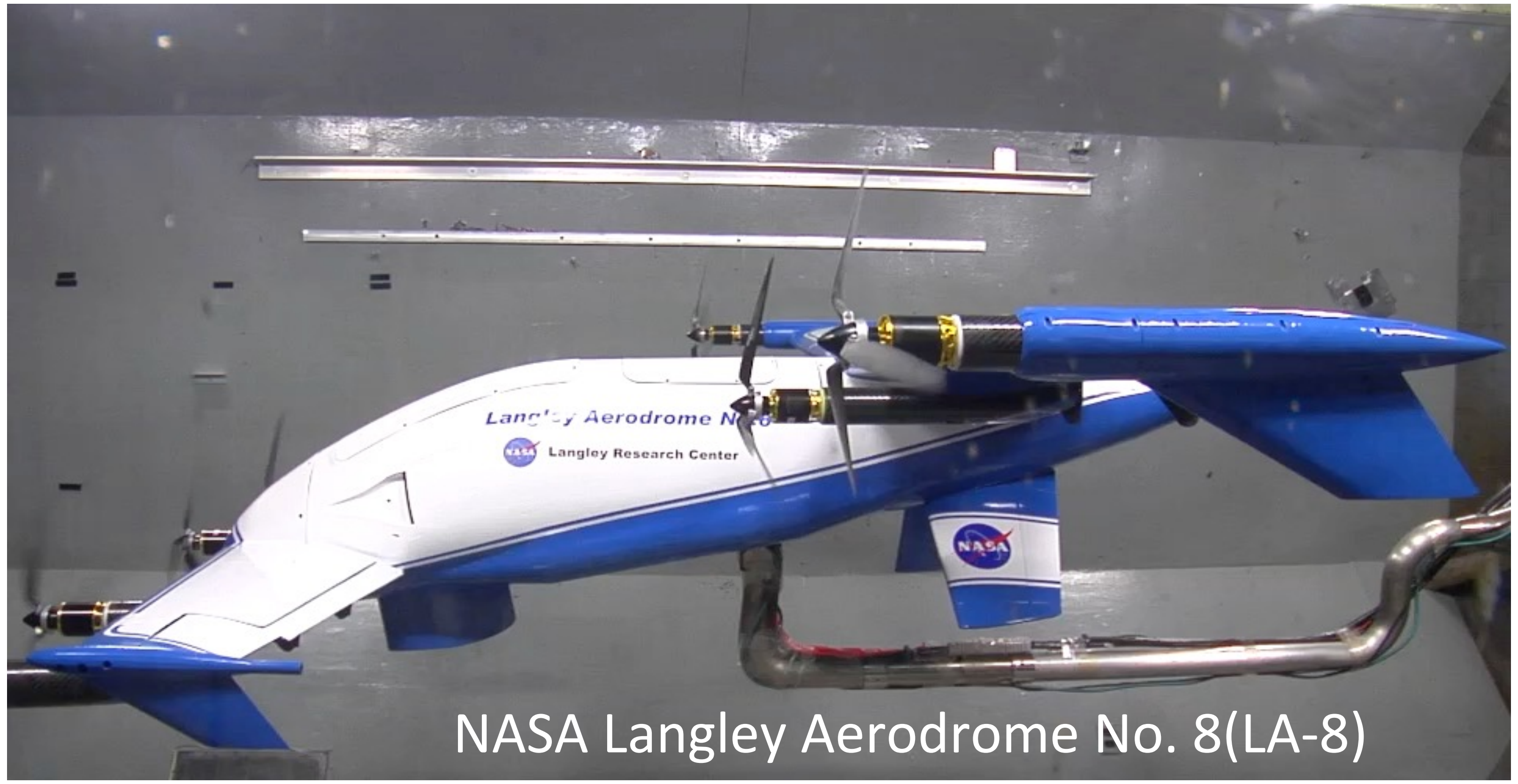
Why Urban Air Mobility

Vehicle Characteristics:

- Highly Nonlinear Dynamics
- Hybrid/Time Varying
- Poor Aero-propulsive Modeling
- Different Operational Modes:
  - Rotor-Borne Flight
  - Fixed-Wing like Cruise
  - Transition

Challenges:

- Accurate Trajectory Planning & Replanning
- Optimal Transitions between Modes of Flight
- Parameter Adjustments mid flight
- Large differences in propulsion methods of vehicles



NASA Langley Aerodrome No. 8(LA-8)

Differential Dynamic Programming (DDP)

Past extensions of DDP address:

- Parametric uncertainty [1]
- Nonparametric uncertainty using GPs [2]
- Trajectory Optimization using Polynomials Chaos [3]

We propose a method that:

- Co-optimizes both controls and parameters simultaneously
- Generalizes to different robotics tasks and robotics systems
- Incorporates dynamics with multiple modes and handles transitions between them

[1] Kobilarov, M., Ta, D. N., & Dellaert, F. (2015). Differential dynamic programming for optimal estimation. In *2015 IEEE International Conference on Robotics and Automation (ICRA)* (pp. 863-869). IEEE.

[2] Pan, Y., & Theodorou, E. (2014). Probabilistic differential dynamic programming. *Advances in Neural Information Processing Systems*, *27*.

[3] Boutselis, G. I., Pan, Y., & Theodorou, E. A. (2019). Numerical trajectory optimization for stochastic mechanical systems. *SIAM Journal on scientific computing*, *41*(4), A2065-A2087.

Parameterized Differential Dynamic Programming (PDDP)

General Problem Formulation

$$\min_{\mathbf{U}} \mathcal{J}(\mathbf{U}) = \min_{\mathbf{U}} \sum_{t=1}^{T} \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t) + \phi(\mathbf{x}_{T+1})$$

$$\min_{\mathbf{U},\boldsymbol{\theta}} \mathcal{J}(\mathbf{U};\boldsymbol{\theta}) = \min_{\mathbf{U}} \min_{\boldsymbol{\theta}} \sum_{t=1}^{T} \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t;\boldsymbol{\theta}) + \phi(\mathbf{x}_{T+1};\boldsymbol{\theta})$$

subject to $\mathbf{x}_{t+1} = \mathbf{F}(\mathbf{x}_t, \mathbf{u}_t;\boldsymbol{\theta})$, $\mathbf{x}_1$ given

Backward Pass Equations

$$V_t^0 = Q_t^0 + (\frac{1}{2}\epsilon^2 - \epsilon)(Q_t^u)^\top (Q_t^{uu})^{-1} Q_t^u,$$

$$V_t^x = Q_t^x - Q_t^{xu}(Q_t^{uu})^{-1} Q_t^u,$$

$$V_t^\theta = Q_t^\theta - Q_t^{\theta u}(Q_t^{uu})^{-1} Q_t^u,$$

$$V_t^{xx} = Q_t^{xx} - Q_t^{xu}(Q_t^{uu})^{-1} Q_t^{ux},$$

$$V_t^{x\theta} = Q_t^{x\theta} - Q_t^{xu}(Q_t^{uu})^{-1} Q_t^{u\theta},$$

$$V_t^{\theta\theta} = Q_t^{\theta\theta} - Q_t^{\theta u}(Q_t^{uu})^{-1} Q_t^{u\theta}.$$

Control Update Law

$$\delta\mathbf{u}_t^* = \epsilon\mathbf{k}_t + \mathbf{K}_t\delta\mathbf{x}_t + \mathbf{M}_t\delta\boldsymbol{\theta}^*$$

$$\delta\boldsymbol{\theta}^* = \epsilon\mathbf{n}$$

$\epsilon \in (0,1]$ determined through line search

Theoretical Analysis of Improvements

- Second-order algorithm derived in a principled manner by extending classical optimal control

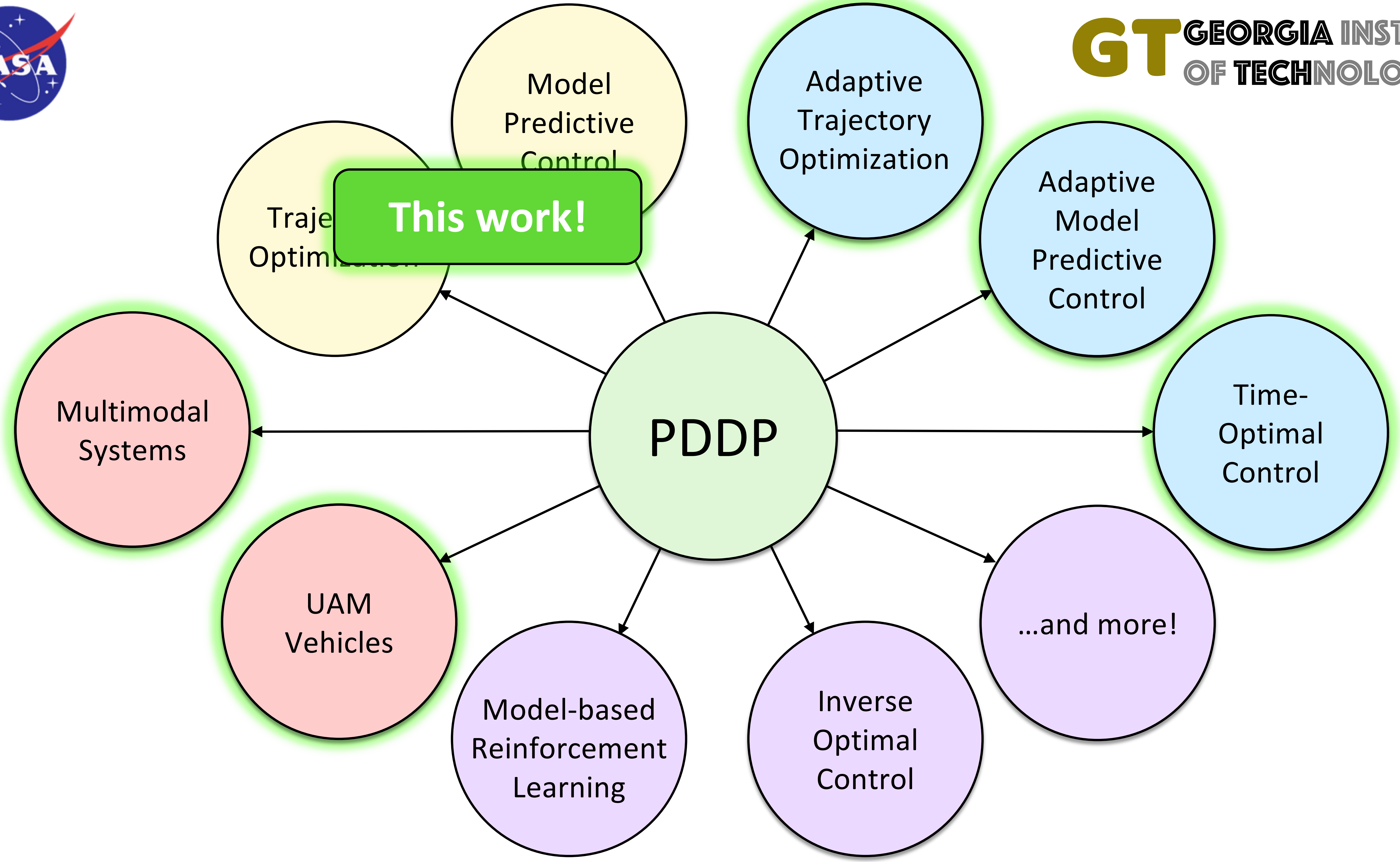
Proposition 1. *The parameter update $\delta\boldsymbol{\theta}^*$ is a (damped) Newton step towards the minimum of the value function.*

- Convergence guarantees independent of initialization

Theorem 1. *As the number of iterations of PDDP approaches infinity, the cost $\mathcal{J}$, the control trajectory $\mathbf{U}$, and the parameters $\boldsymbol{\theta}$ converge to a stationary point regardless of initialization.*

- Co-optimizes for controls and parameters simultaneously

- General parameterization

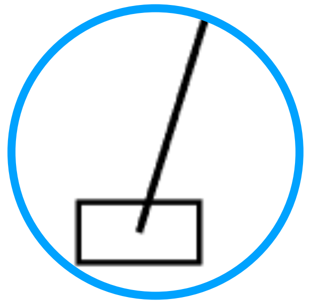
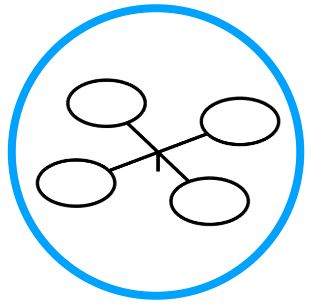

Model Predictive Control
Traje
Optim
This work!
Adaptive Trajectory Optimization
Adaptive Model Predictive Control
Multimodal Systems
PDDP
Time-Optimal Control
UAM Vehicles
Model-based Reinforcement Learning
Inverse Optimal Control
...and more!

Applications

Adaptive Model Predictive Control

$$\mathcal{J}_{\text{combined}}(\mathbf{u}_{t:T};\boldsymbol{\theta},\mathbf{x}_1) = \mathcal{J}_{\text{est}}(\boldsymbol{\theta},\mathbf{x}_1) + \mathcal{J}_{\text{mpc}}(\mathbf{u}_{t:T})$$

- $\mathcal{J}_{\text{est}}$ → Moving Horizon Estimation → Maximize likelihood of observed states
- $\mathcal{J}_{\text{mpc}}$ → Model Predictive Control → Plan future trajectory

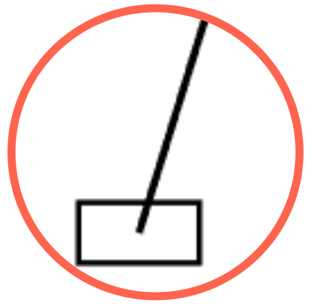
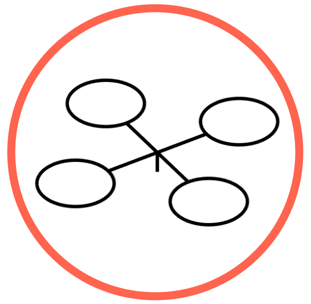

Switching Time Optimization

Switching Time System

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \theta_i \mathbf{f}^{(i)}(\mathbf{x}_t,\mathbf{u}_t)\Delta t$$

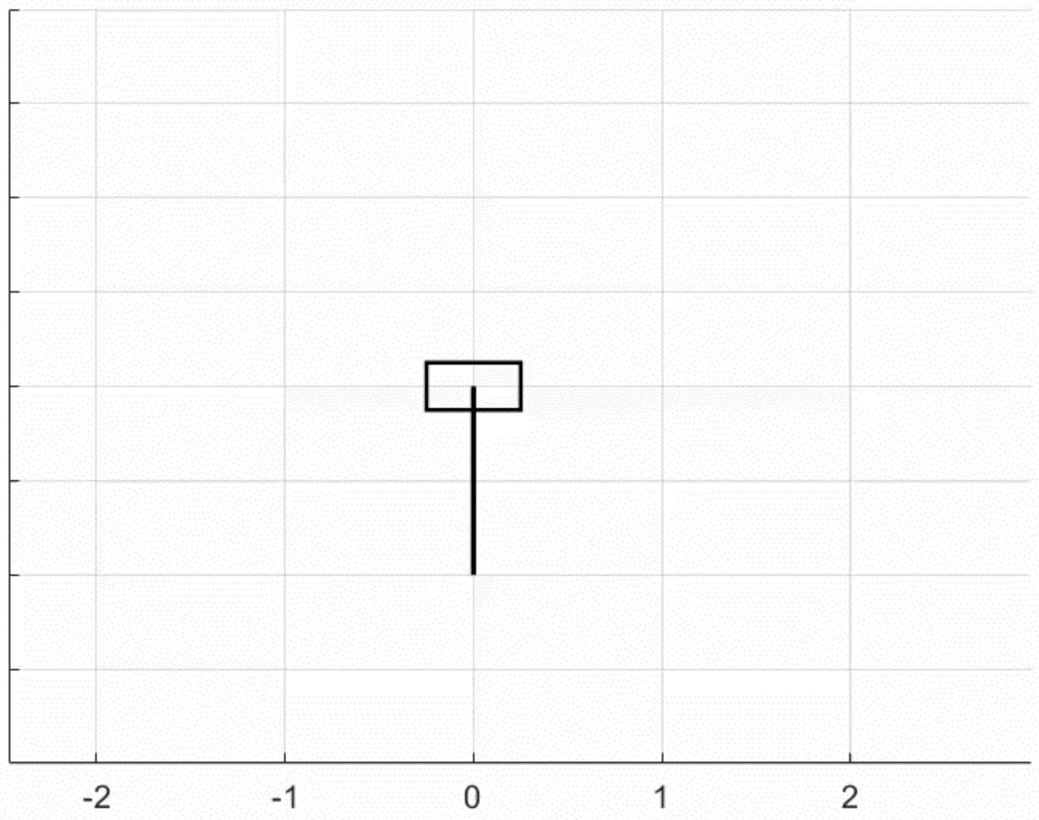
$$\sum_{t=T_{i-1}+1}^{T_i} \theta_i \mathcal{L}^{(i)}(\mathbf{x}_t,\mathbf{u}_t) + \phi^{(i)}(\mathbf{x}_{T_i+1})$$

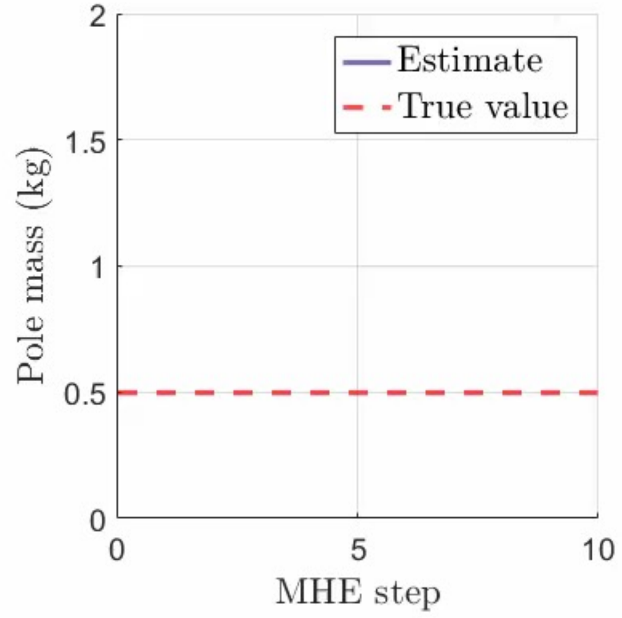
$$i = 1, 2, \ldots, N$$

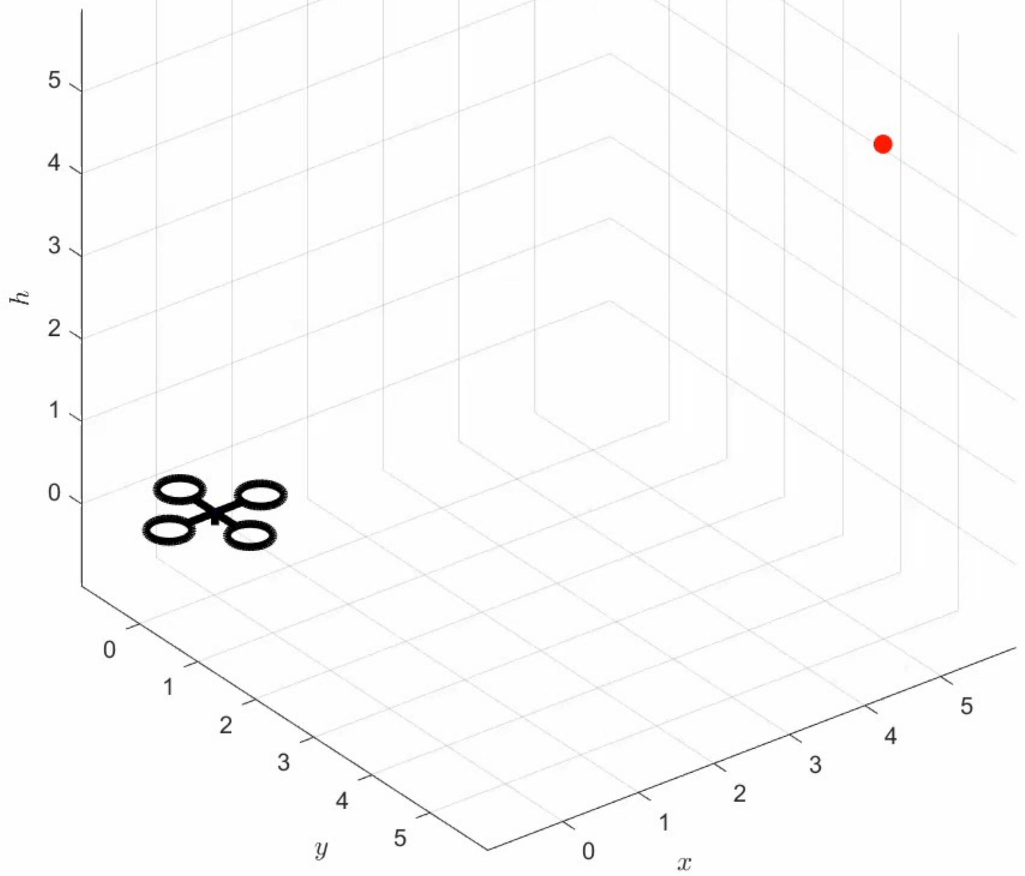

Adaptive MPC

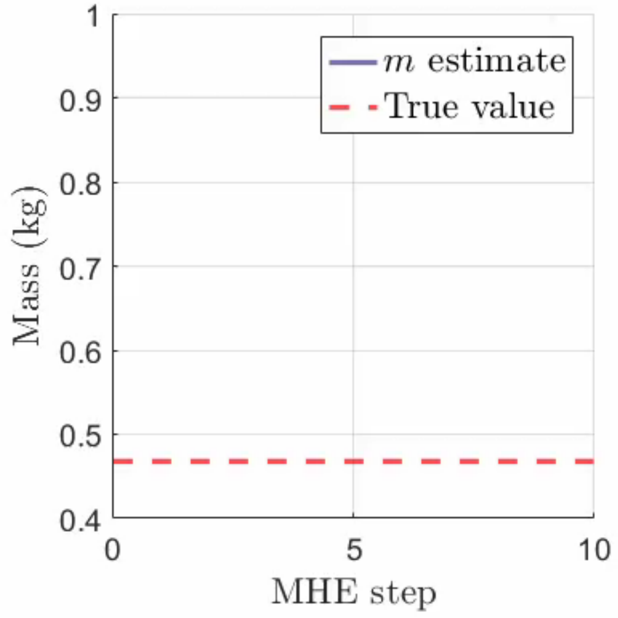
Cartpole

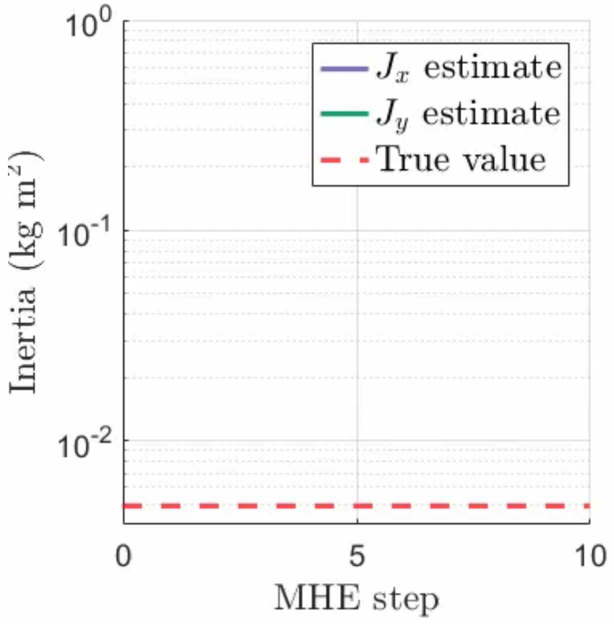


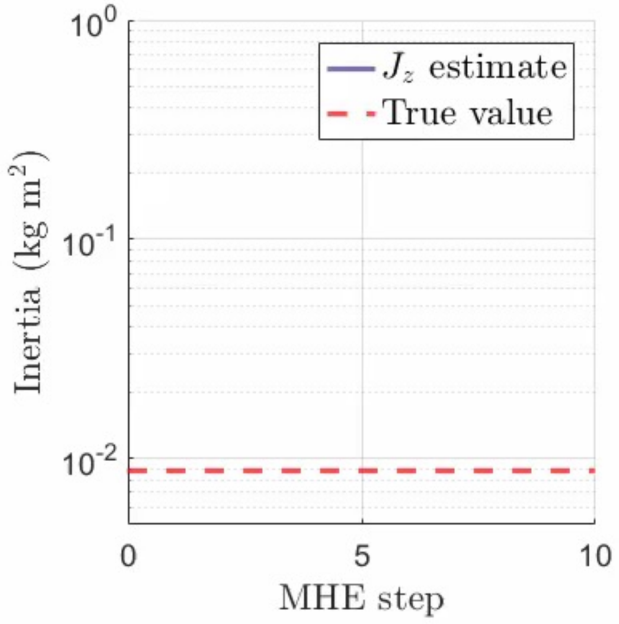


Quadrotor



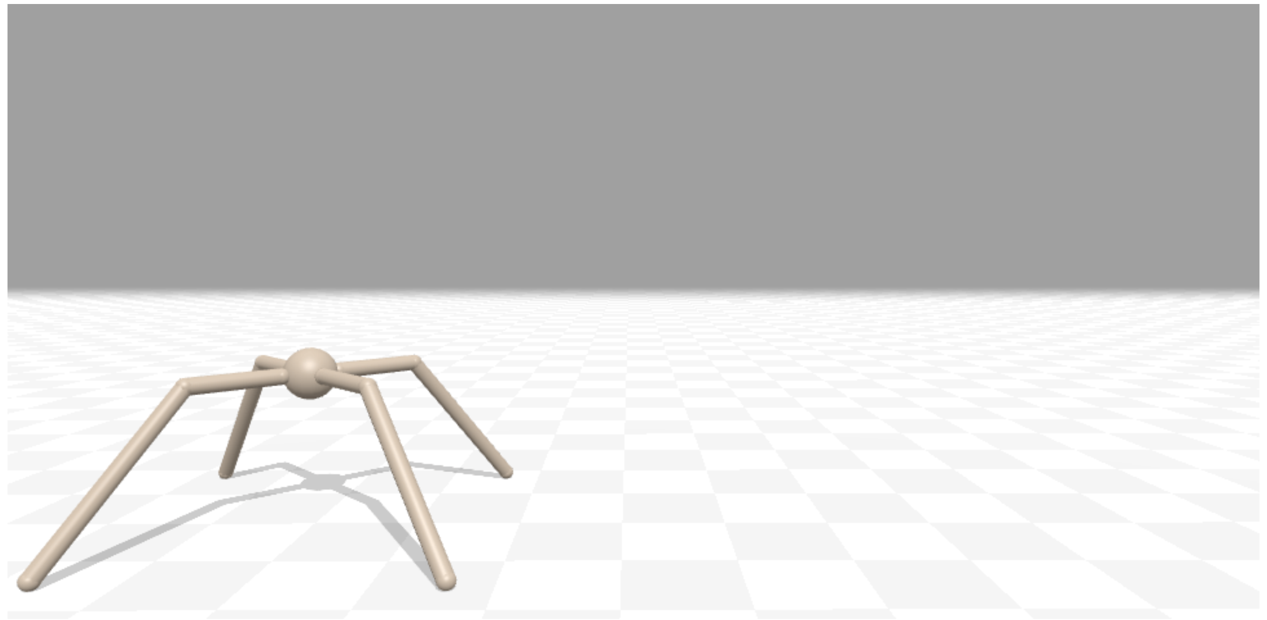




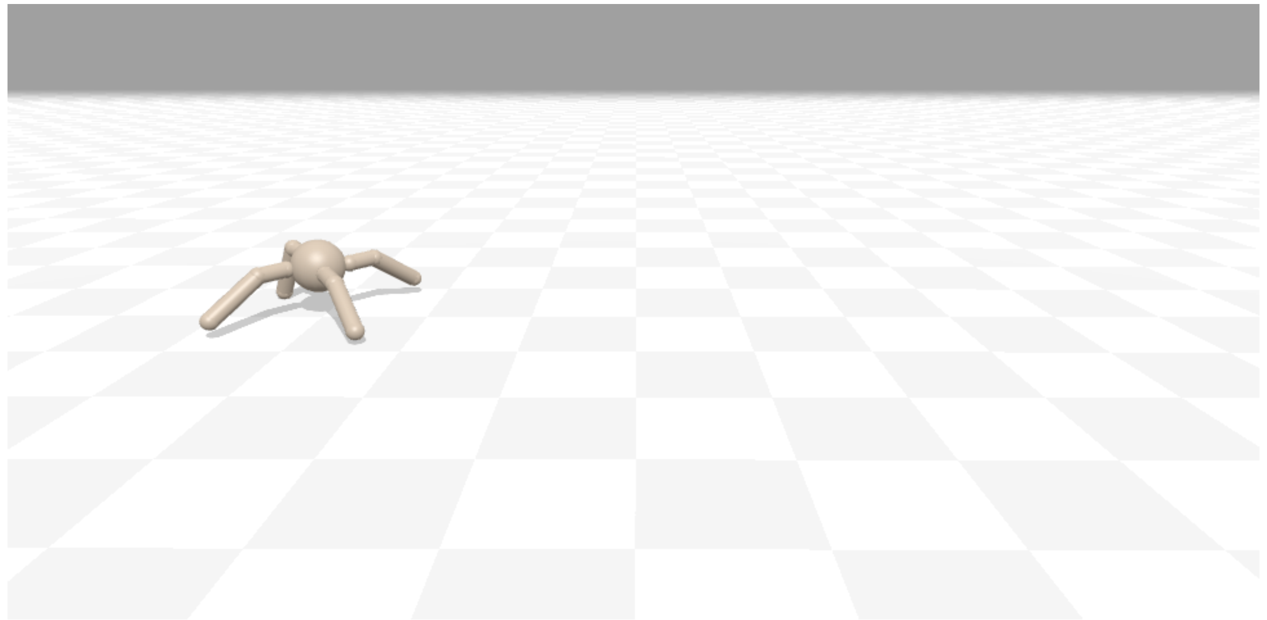


Adaptive MPC

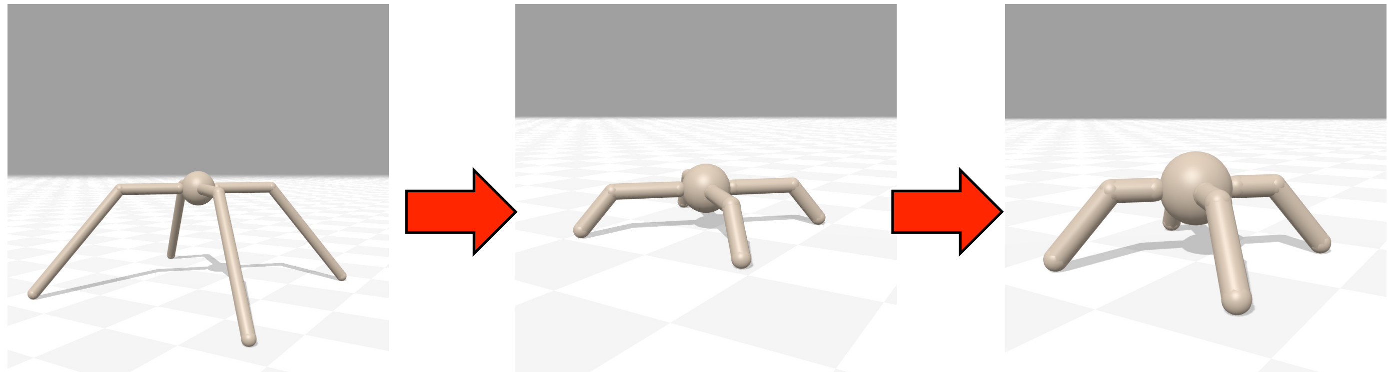
Ant

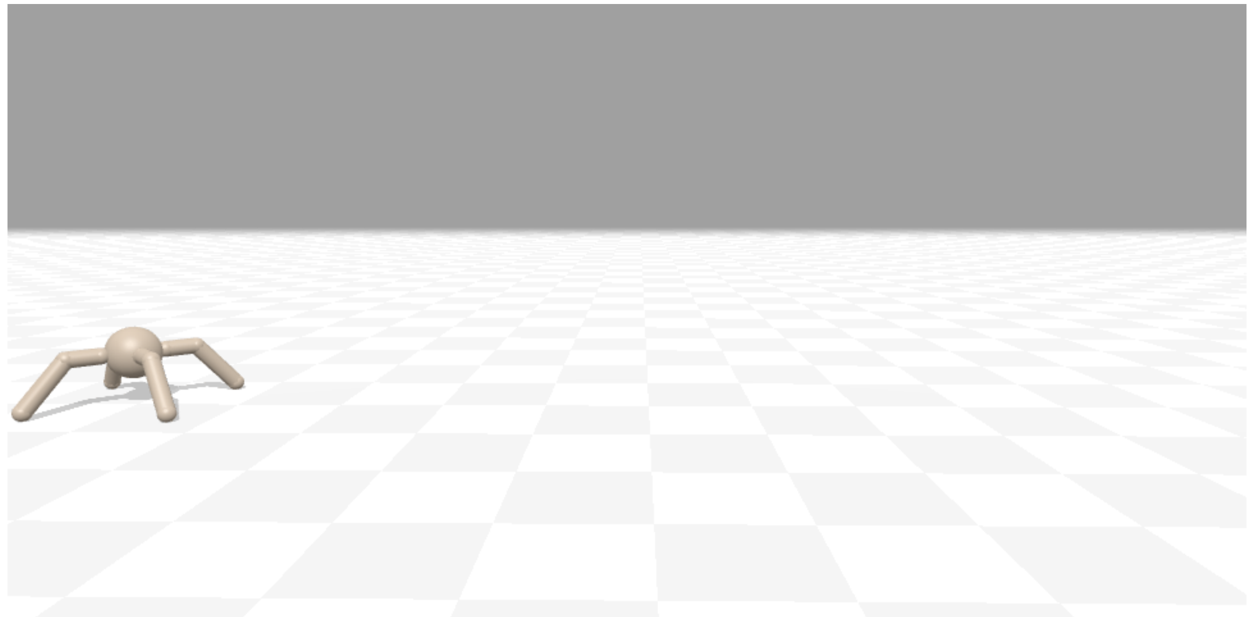


DDP planning on model with incorrect parameters

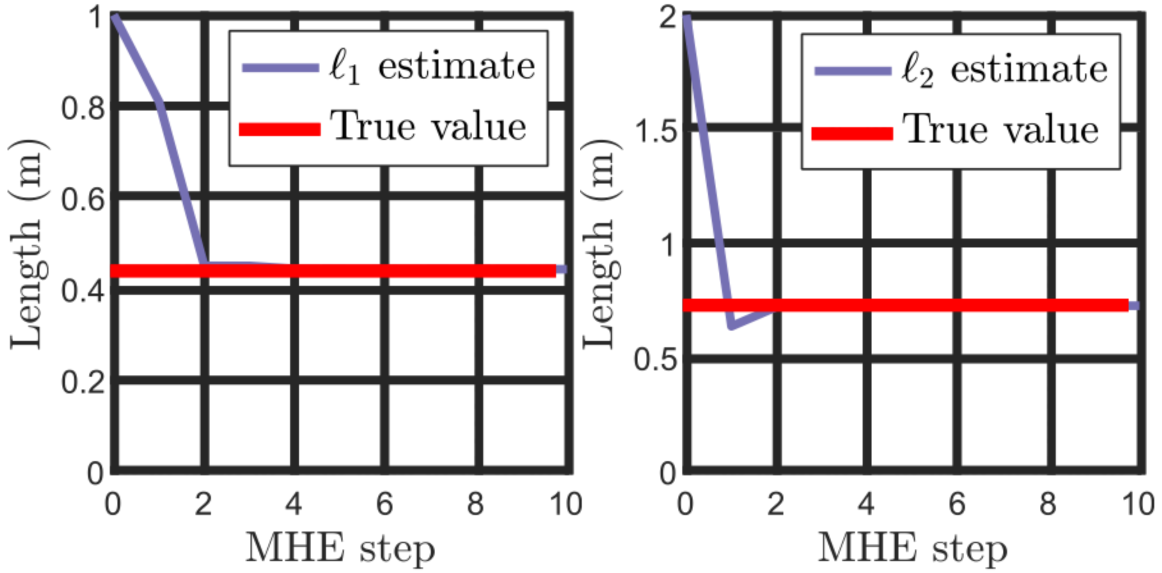


Executing plan on true model: Failure





PDDP with adaptive control: Success



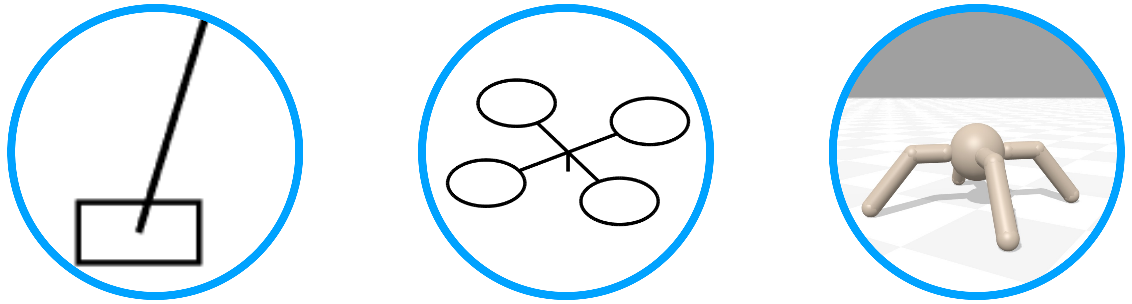
Applications

Adaptive Model Predictive Control

$$\mathcal{J}_{\text{combined}}(\mathbf{u}_{t:T}; \boldsymbol{\theta}, \mathbf{x}_1) = \mathcal{J}_{\text{est}}(\boldsymbol{\theta}, \mathbf{x}_1) + \mathcal{J}_{\text{mpc}}(\mathbf{u}_{t:T})$$

Moving Horizon Estimation → Maximize likelihood of observed states

Model Predictive Control → Plan future trajectory



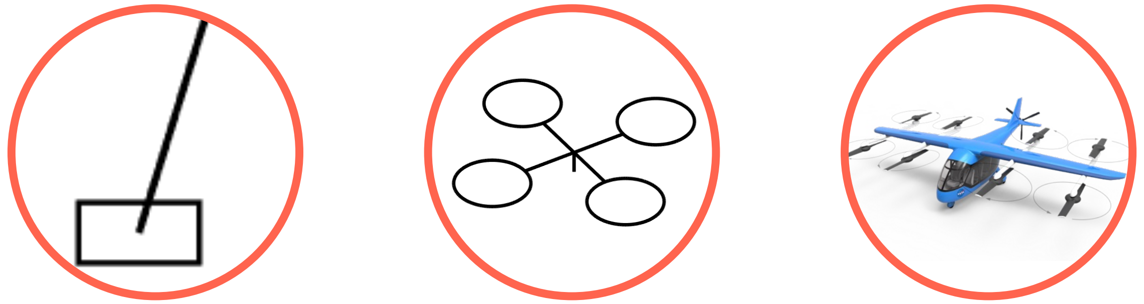
Switching Time Optimization

Switching Time System

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \theta_i \mathbf{f}^{(i)}(\mathbf{x}_t, \mathbf{u}_t)\Delta t$$

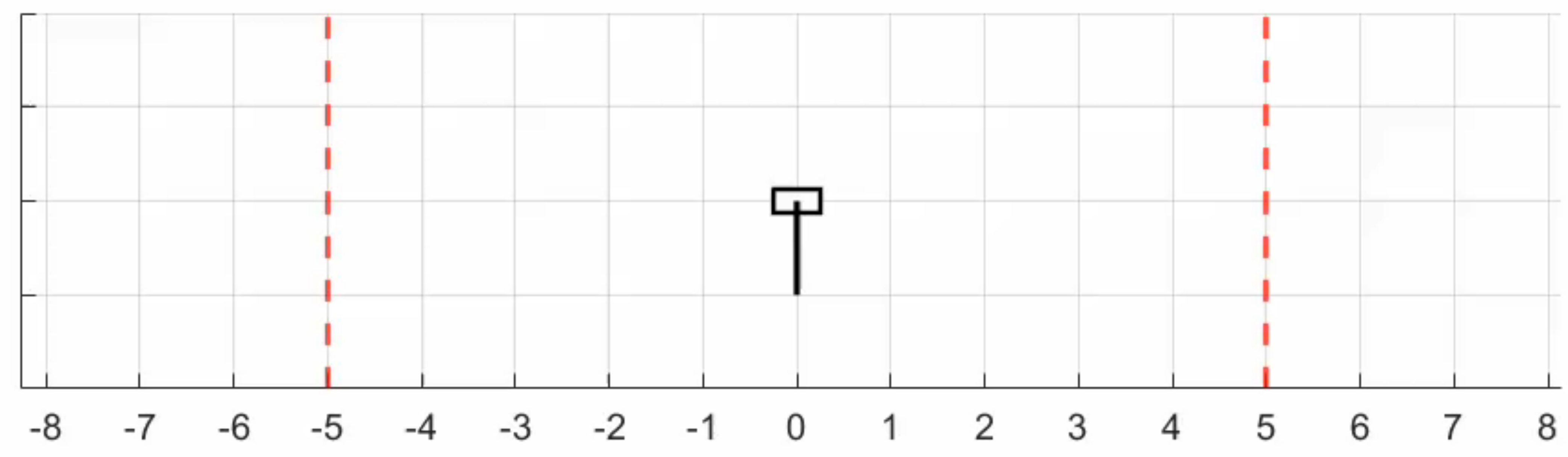
$$\sum_{t=T_{i-1}+1}^{T_i} \theta_i \mathcal{L}^{(i)}(\mathbf{x}_t, \mathbf{u}_t) + \phi^{(i)}(\mathbf{x}_{T_i+1})$$

$$i = 1, 2, \ldots, N$$

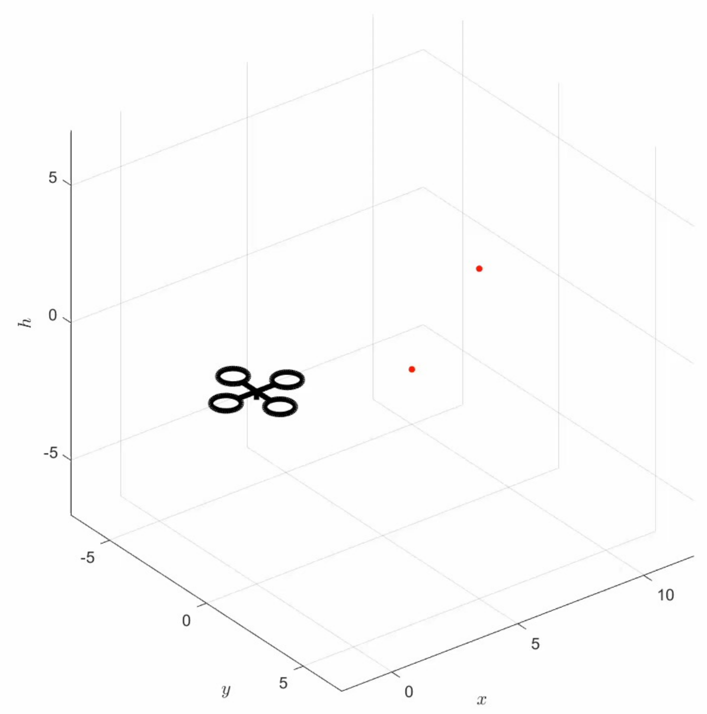


Switching Time Optimization (STO)

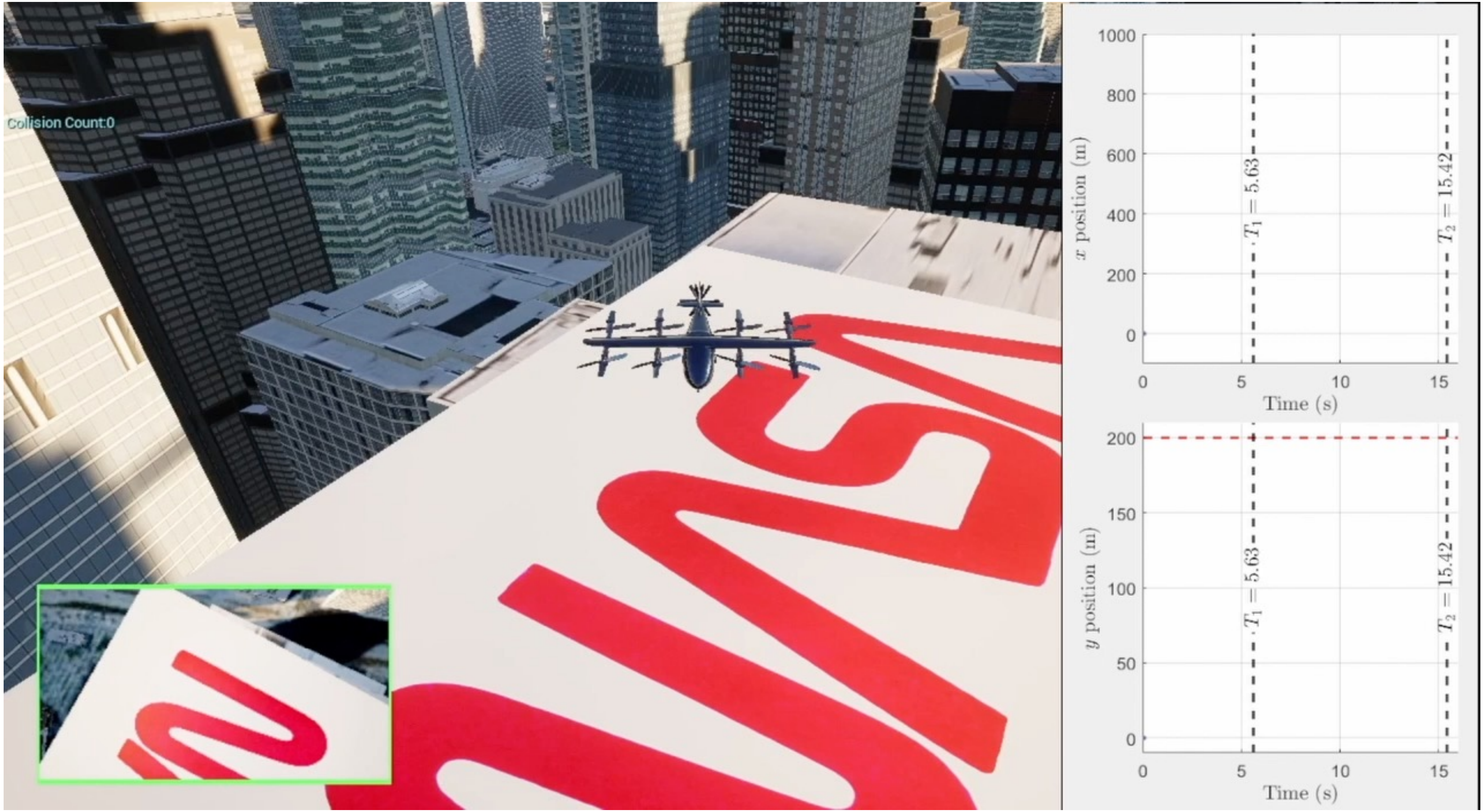
Cartpole



Quadrotor



STO – Lift+Cruise Vehicle



Conclusion

Parameterized DDP:

- Second-order algorithm derived by extending classical optimal control
- Convergence guarantees independent of initialization
- Co-optimizes for controls and parameters simultaneously
- Generality enables application to various robotics applications, including adaptive MPC and STO tasks

Any Further Questions?
Visit our poster @ 4pm!
Or reach out to:
alexoshin@gatech.edu

UNUSED SLIDES AFTER THIS POINT

Parametric Differential Dynamic Programming (PDDP)

General Problem Formulation

$$\min_{\mathbf{U},\boldsymbol{\theta}} \mathcal{J}(\mathbf{U};\boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \min_{\mathbf{U}} \sum_{t=1}^{T} \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t; \boldsymbol{\theta}) + \phi(\mathbf{x}_{T+1}; \boldsymbol{\theta})$$

$$\text{subject to } \mathbf{x}_{t+1} = \mathbf{F}(\mathbf{x}_t, \mathbf{u}_t; \boldsymbol{\theta}), \ \mathbf{x}_1 \text{ given}$$

Backward Pass Equations

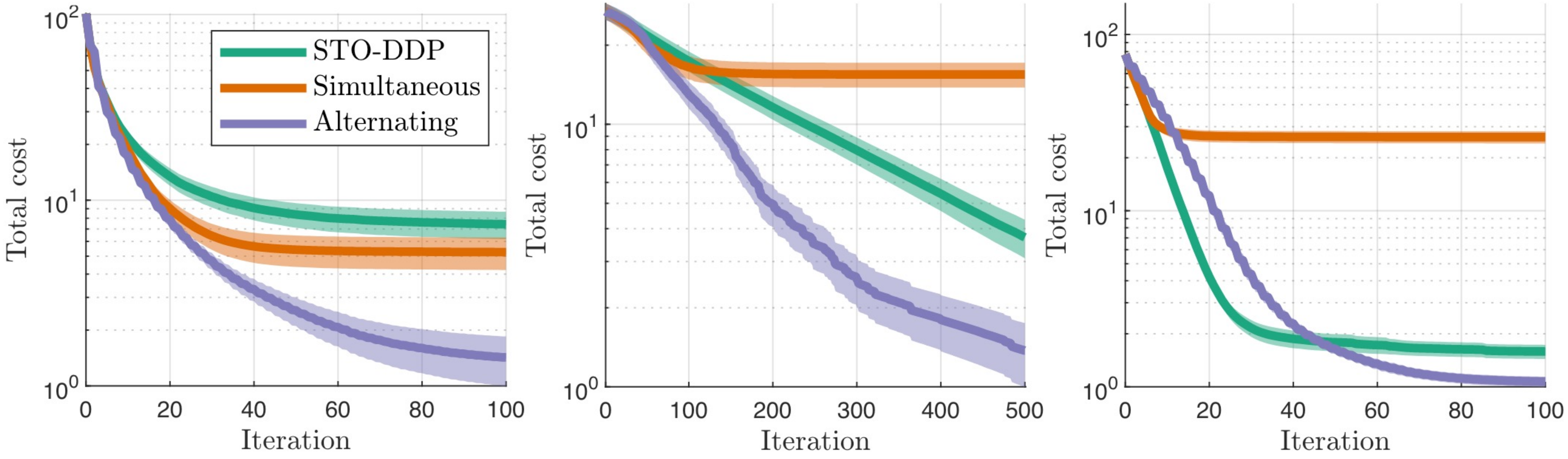
$$\begin{aligned}
V_t^0 &= Q_t^0 + (\frac{1}{2}\epsilon^2 - \epsilon)(Q_t^u)^\top (Q_t^{uu})^{-1} Q_t^u, \\
V_t^x &= Q_t^x - Q_t^{xu}(Q_t^{uu})^{-1} Q_t^u, \\
V_t^\theta &= Q_t^\theta - Q_t^{\theta u}(Q_t^{uu})^{-1} Q_t^u, \\
V_t^{xx} &= Q_t^{xx} - Q_t^{xu}(Q_t^{uu})^{-1} Q_t^{ux}, \\
V_t^{x\theta} &= Q_t^{x\theta} - Q_t^{xu}(Q_t^{uu})^{-1} Q_t^{u\theta} = (V_t^{\theta x})^\top, \\
V_t^{\theta\theta} &= Q_t^{\theta\theta} - Q_t^{\theta u}(Q_t^{uu})^{-1} Q_t^{u\theta}.
\end{aligned}$$

Control Update Law

$$\begin{aligned}
\delta \mathbf{u}_t^* &= \epsilon \mathbf{k}_t + \mathbf{K}_t \delta \mathbf{x}_t + \mathbf{M}_t \delta \boldsymbol{\theta}^*, \\
\delta \boldsymbol{\theta}^* &= \epsilon \mathbf{m},
\end{aligned}$$

$$\epsilon \in (0, 1] \text{ determined through line search}$$

Switching Time Optimization



(a) Cartpole (b) Quadrotor (c) Lift+Cruise

Problem Formulation

Parameterized optimal control problem:

$$\min_{\mathbf{U},\boldsymbol{\theta}} \mathcal{J}(\mathbf{U};\boldsymbol{\theta}) = \min_{\boldsymbol{\theta}} \min_{\mathbf{U}} \sum_{t=1}^{T} \mathcal{L}(\mathbf{x}_t, \mathbf{u}_t; \boldsymbol{\theta}) + \phi(\mathbf{x}_{T+1}; \boldsymbol{\theta})$$

$$\text{subject to } \mathbf{x}_{t+1} = \mathbf{F}(\mathbf{x}_t, \mathbf{u}_t; \boldsymbol{\theta}), \ \ \mathbf{x}_1 \text{ given}$$

Value Function

$$V(\mathbf{x}_t;\boldsymbol{\theta}) := \min_{\mathbf{u}_{t:T}} \left[\sum_{i=t}^{T} \mathcal{L}(\mathbf{x}_i, \mathbf{u}_i; \boldsymbol{\theta}) + \phi(\mathbf{x}_{T+1}; \boldsymbol{\theta}) \right]$$

$$= \min_{\mathbf{u}_t} \Big[\underbrace{\mathcal{L}(\mathbf{x}_t, \mathbf{u}_t; \boldsymbol{\theta}) + V(\mathbf{x}_{t+1}; \boldsymbol{\theta})}_{:= Q(\mathbf{x}_t, \mathbf{u}_t; \boldsymbol{\theta})} \Big]$$

Optimal parameters are given as

$$\boldsymbol{\theta}^* = \operatorname*{argmin}_{\boldsymbol{\theta}} V(\mathbf{x}_1; \boldsymbol{\theta})$$

Value Function Solution

Quadratic approximation explicitly incorporates parameters:

$$\begin{aligned} V(\mathbf{x}_t;\boldsymbol{\theta}) &= V(\bar{\mathbf{x}}_t + \delta\mathbf{x}_t; \bar{\boldsymbol{\theta}} + \delta\boldsymbol{\theta}) \\ &\approx V_t^0 + (V_t^x)^\top \delta\mathbf{x}_t + (V_t^\theta)^\top \delta\boldsymbol{\theta} \\ &\quad + \frac{1}{2}\begin{bmatrix}\delta\mathbf{x}_t \\ \delta\boldsymbol{\theta}\end{bmatrix}^\top \begin{bmatrix} V_t^{xx} & V_t^{x\theta} \\ V_t^{\theta x} & V_t^{\theta\theta}\end{bmatrix}\begin{bmatrix}\delta\mathbf{x}_t \\ \delta\boldsymbol{\theta}\end{bmatrix}, \\ &\text{where } V_t^0 = V(\bar{\mathbf{x}}_t, \bar{\boldsymbol{\theta}}),\ \text{etc.} \\ \implies \ \delta\boldsymbol{\theta}^* &= -(V_1^{\theta\theta})^{-1} V_1^\theta := \mathbf{m} \end{aligned}$$

Value Function Solution

$$V(\mathbf{x}_t; \boldsymbol{\theta}) = \min_{\mathbf{u}_t} Q(\mathbf{x}_t, \mathbf{u}_t; \boldsymbol{\theta})$$

$$\begin{aligned} Q(\mathbf{x}_t, \mathbf{u}_t; \boldsymbol{\theta}) &= Q(\bar{\mathbf{x}}_t + \delta\mathbf{x}_t, \bar{\mathbf{u}}_t + \delta\mathbf{u}_t; \bar{\boldsymbol{\theta}} + \delta\boldsymbol{\theta}) \\ &\approx Q_t^0 + (Q_t^x)^\top \delta\mathbf{x}_t + (Q_t^u)^\top \delta\mathbf{u}_t + (Q_t^\theta)^\top \delta\boldsymbol{\theta} \\ &\quad + \frac{1}{2} \begin{bmatrix} \delta\mathbf{x}_t \\ \delta\mathbf{u}_t \\ \delta\boldsymbol{\theta} \end{bmatrix}^\top \begin{bmatrix} Q_t^{xx} & Q_t^{xu} & Q_t^{x\theta} \\ Q_t^{ux} & Q_t^{uu} & Q_t^{u\theta} \\ Q_t^{\theta x} & Q_t^{\theta u} & Q_t^{\theta\theta} \end{bmatrix} \begin{bmatrix} \delta\mathbf{x}_t \\ \delta\mathbf{u}_t \\ \delta\boldsymbol{\theta} \end{bmatrix}, \end{aligned}$$

$$\text{where } Q_t^0 = Q(\bar{\mathbf{x}}_t, \bar{\mathbf{u}}_t; \bar{\boldsymbol{\theta}}), \text{ etc.}$$

Value Function Solution

$$\implies \delta \mathbf{u}_t^* = \underbrace{-(Q_t^{uu})^{-1} Q_t^{u}}_{:=\mathbf{k}_t} \underbrace{-(Q_t^{uu})^{-1} Q_t^{ux}}_{:=\mathbf{K}_t} \delta \mathbf{x}_t \underbrace{-(Q_t^{uu})^{-1} Q_t^{u\theta}}_{:=\mathbf{M}_t} \delta \boldsymbol{\theta}$$

Optimal control and parameter variations:

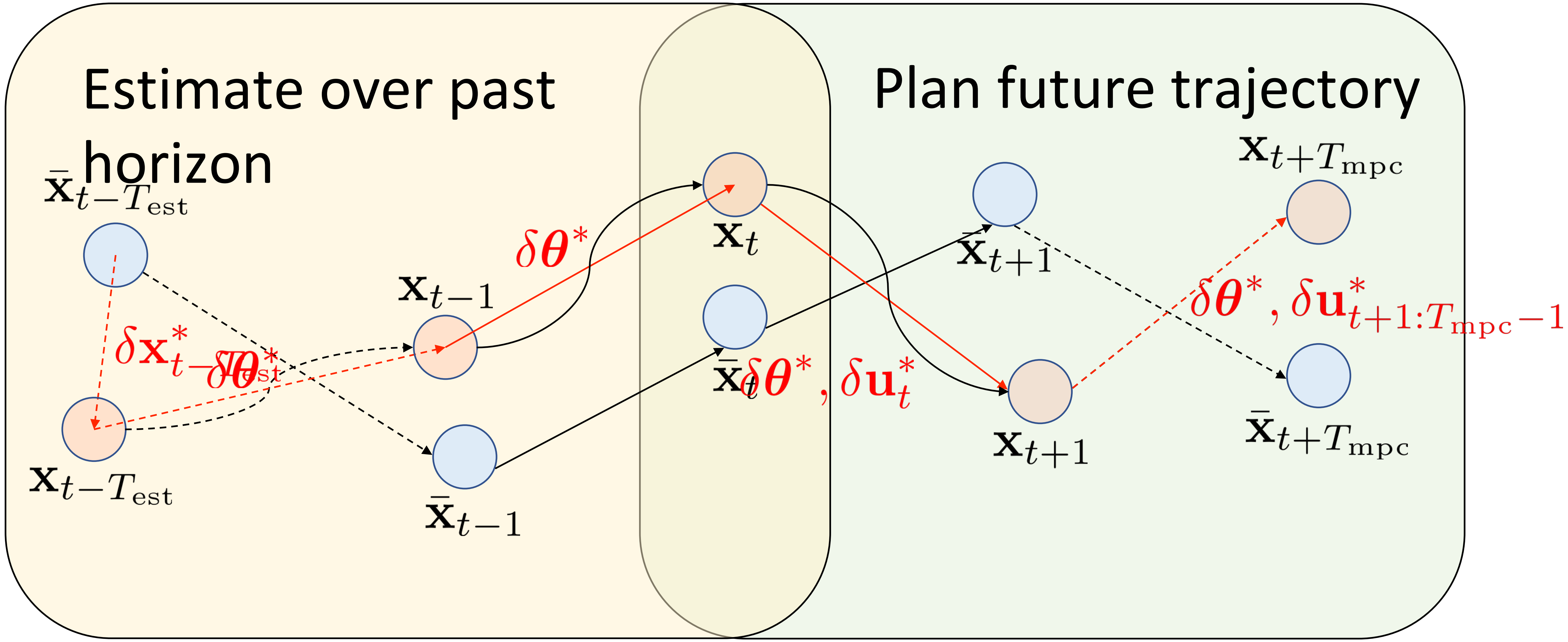
$$\begin{aligned} \delta \mathbf{u}_t^* &= \epsilon \mathbf{k}_t + \mathbf{K}_t \delta \mathbf{x}_t + \mathbf{M}_t \delta \boldsymbol{\theta}^*, \\ \delta \boldsymbol{\theta}^* &= \epsilon \mathbf{m}, \end{aligned}$$
$$\epsilon \in (0, 1] \text{ determined through line search}$$

Value Function Solution

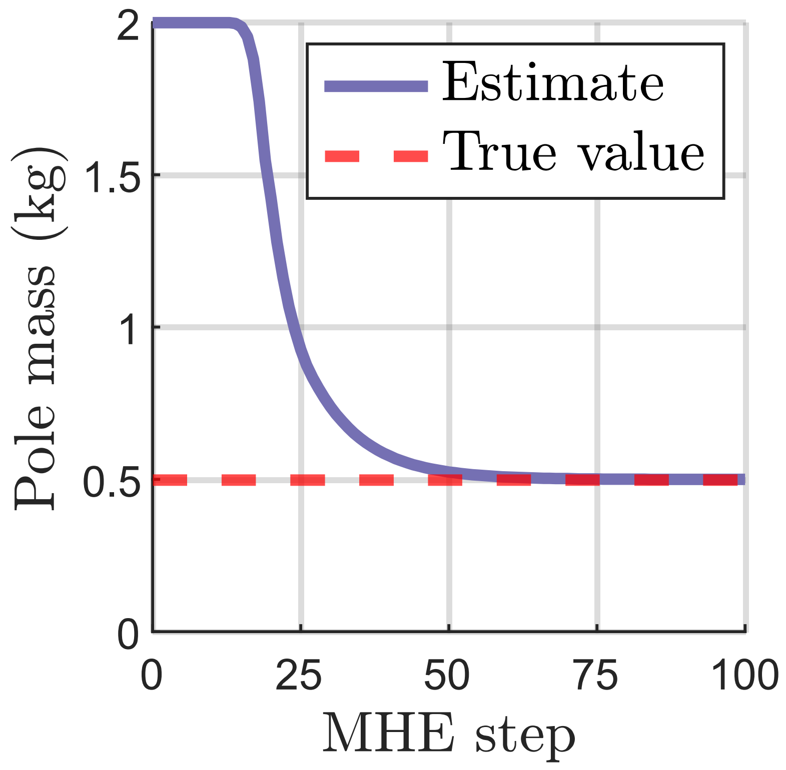
After substitution:

$$
\begin{aligned}
V_t^0 &= Q_t^0 + (\frac{1}{2}\epsilon^2 - \epsilon)(Q_t^u)^\top (Q_t^{uu})^{-1} Q_t^u, \\
V_t^x &= Q_t^x - Q_t^{xu}(Q_t^{uu})^{-1} Q_t^u, \\
V_t^\theta &= Q_t^\theta - Q_t^{\theta u}(Q_t^{uu})^{-1} Q_t^u, \\
V_t^{xx} &= Q_t^{xx} - Q_t^{xu}(Q_t^{uu})^{-1} Q_t^{ux}, \\
V_t^{x\theta} &= Q_t^{x\theta} - Q_t^{xu}(Q_t^{uu})^{-1} Q_t^{u\theta} = (V_t^{\theta x})^\top, \\
V_t^{\theta\theta} &= Q_t^{\theta\theta} - Q_t^{\theta u}(Q_t^{uu})^{-1} Q_t^{u\theta}.
\end{aligned}
$$

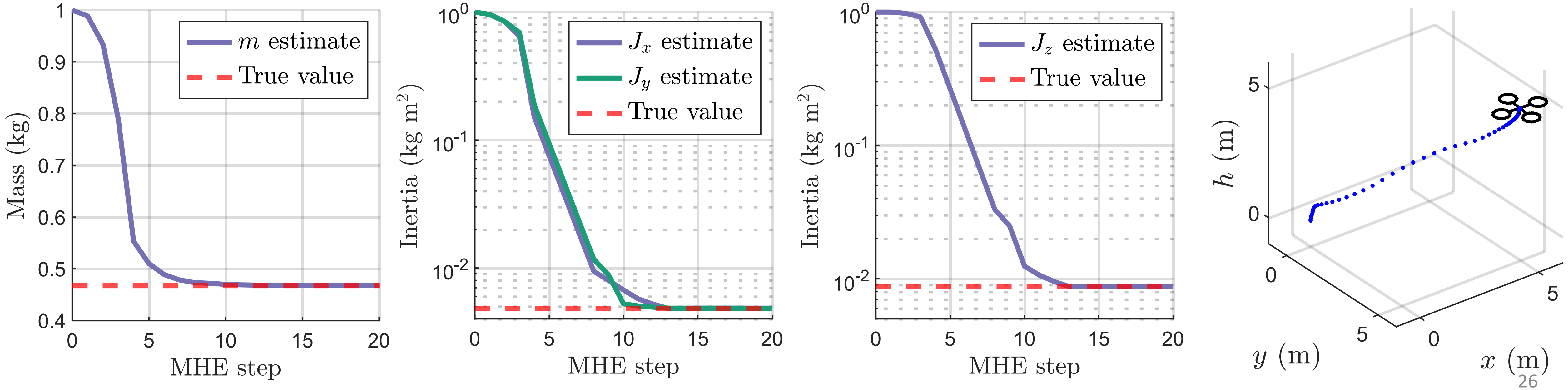
Adaptive MPC using Moving Horizon Estimation (MHE)



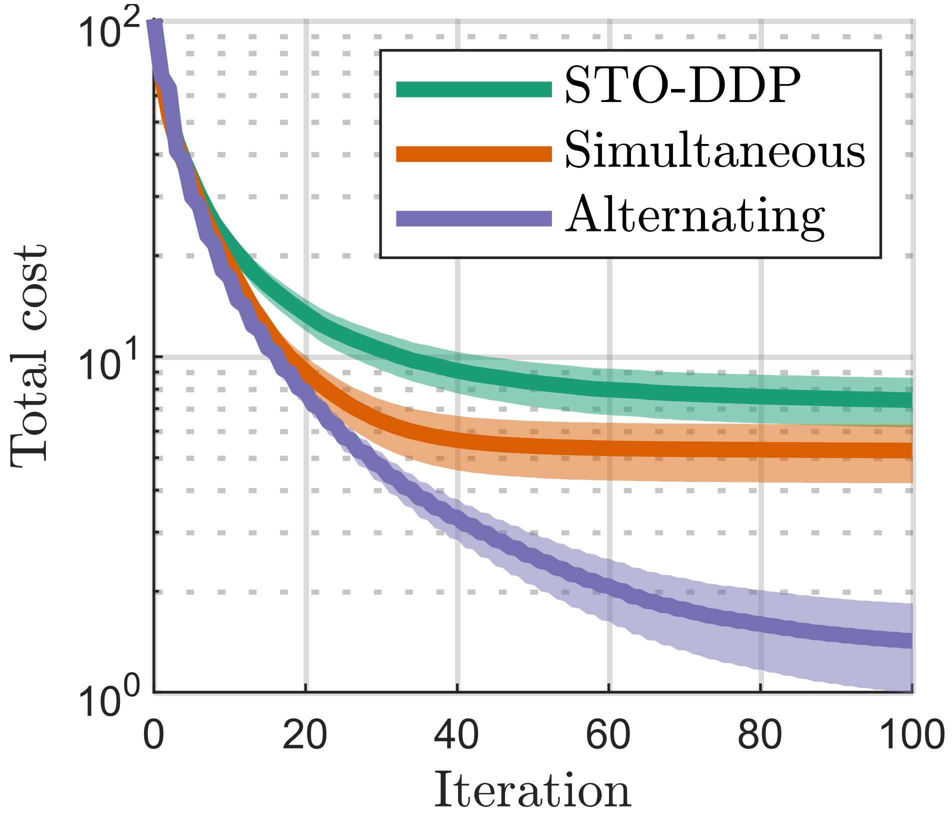
Adaptive MPC - Cartpole

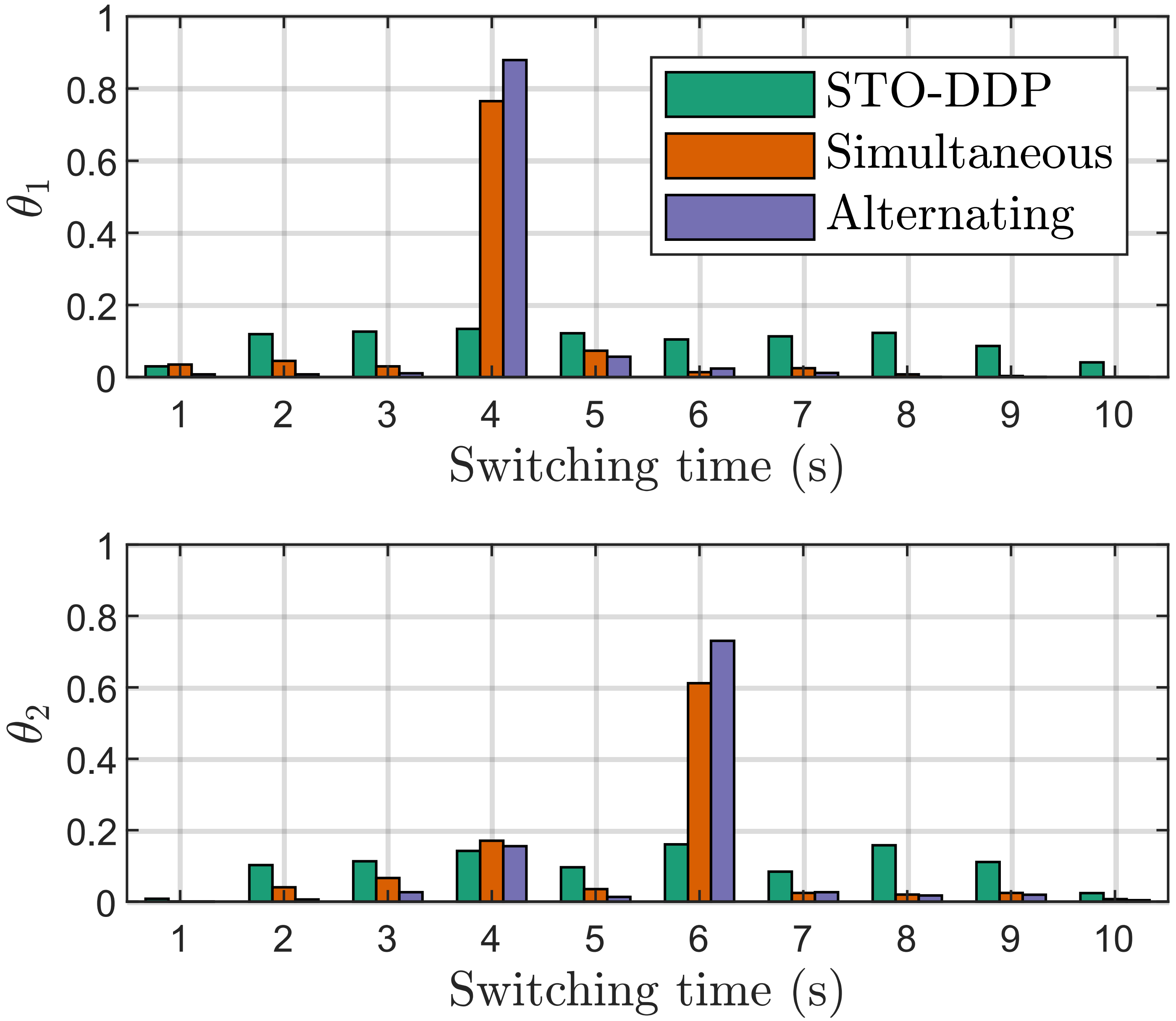


Adaptive MPC – Quadrotor

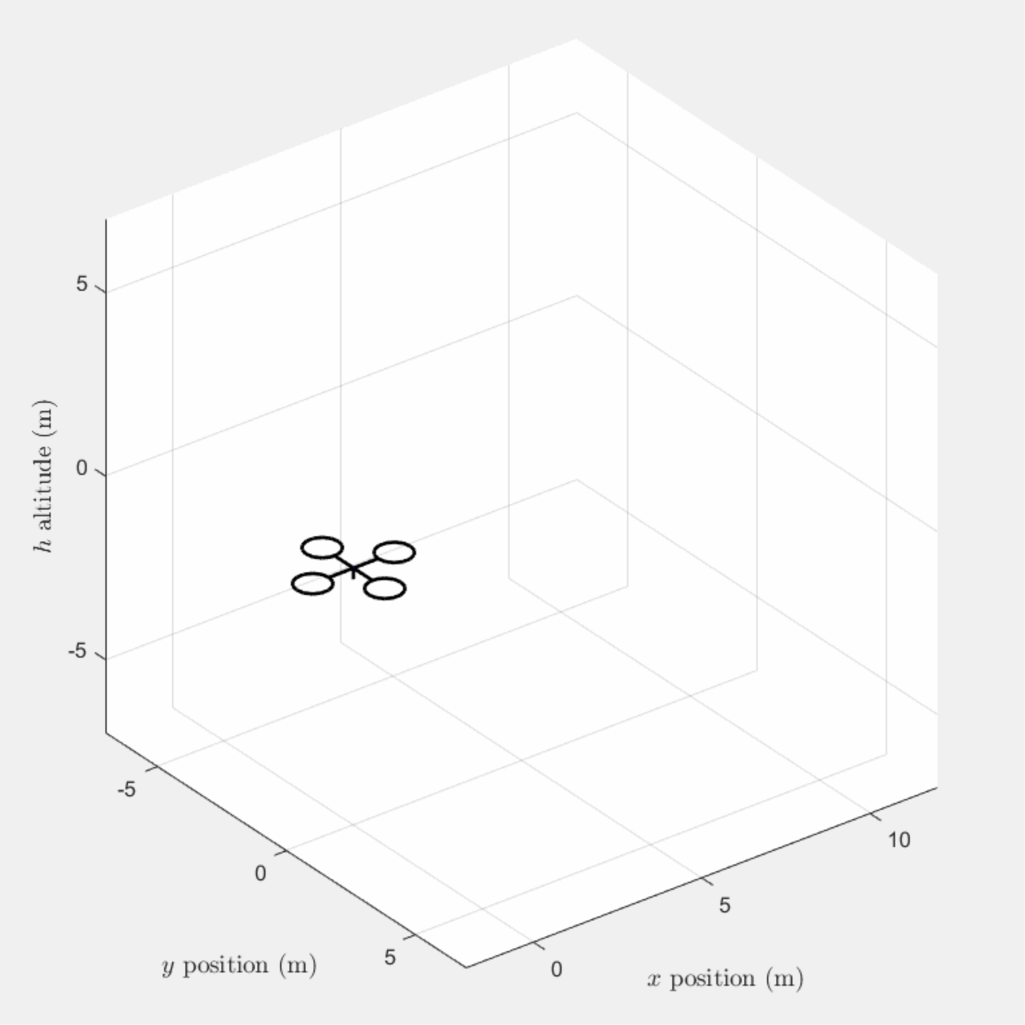


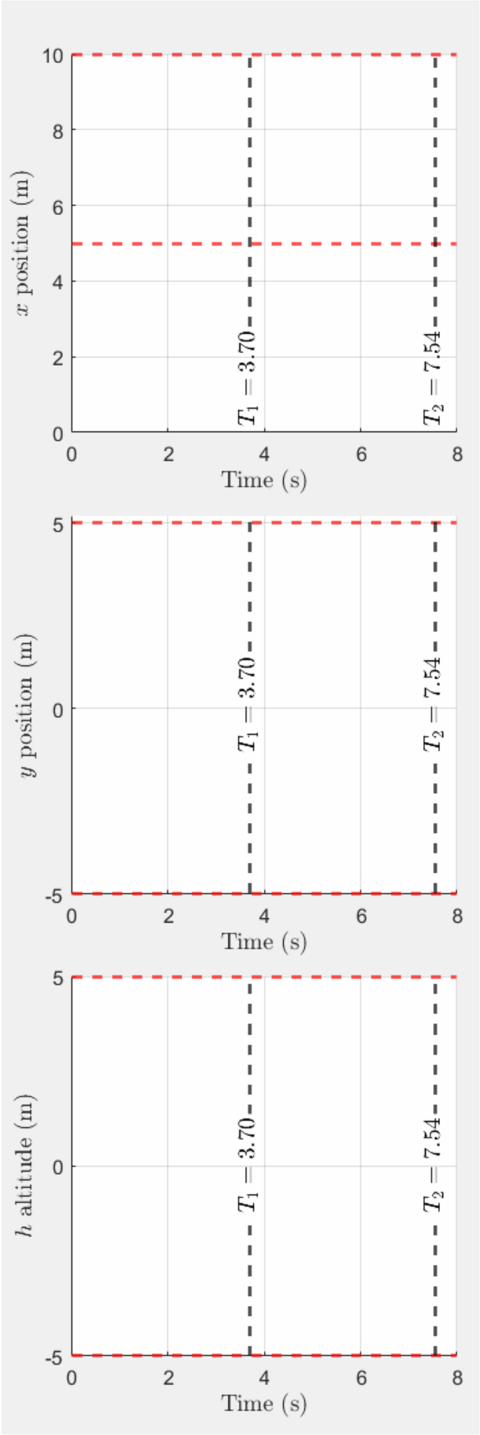
STO – Cartpole





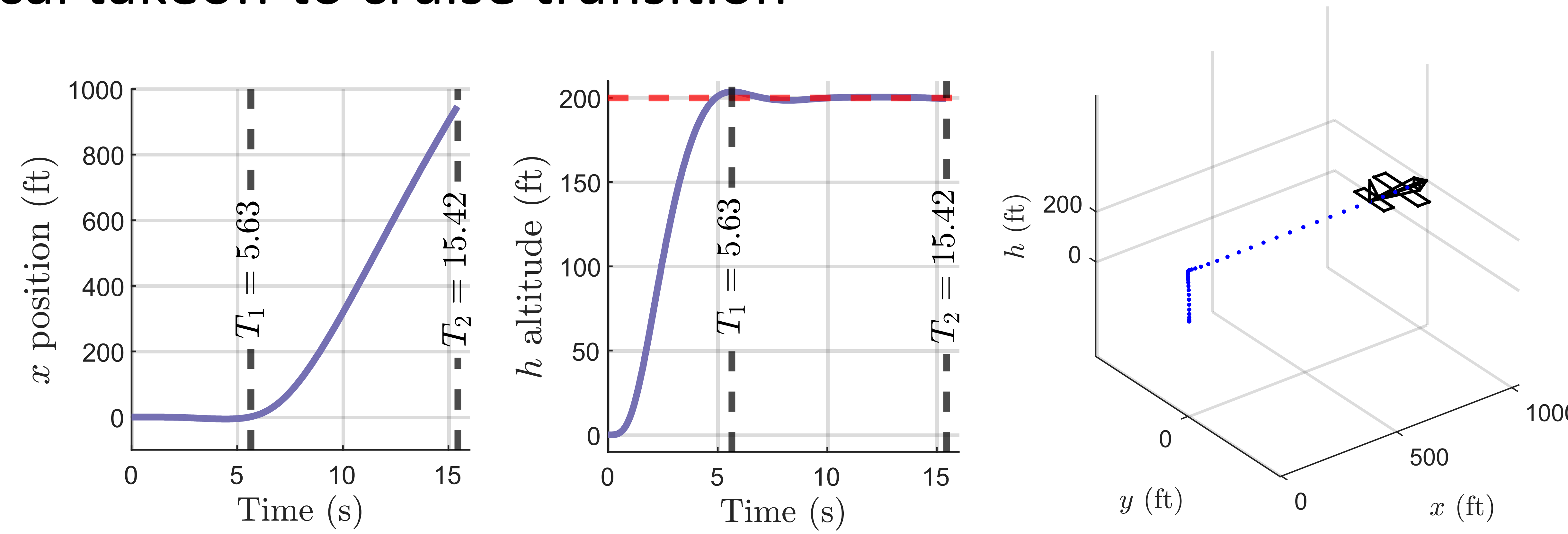
STO – Quadrotor





STO – Lift+Cruise

- Vertical takeoff to cruise transition



STO – Lift+Cruise

- Cruise to hover to vertical landing transition