Simultaneous System Identification and Model Predictive Control with No Dynamic Regret

Hongyu Zhou, Vasileios Tzoumas

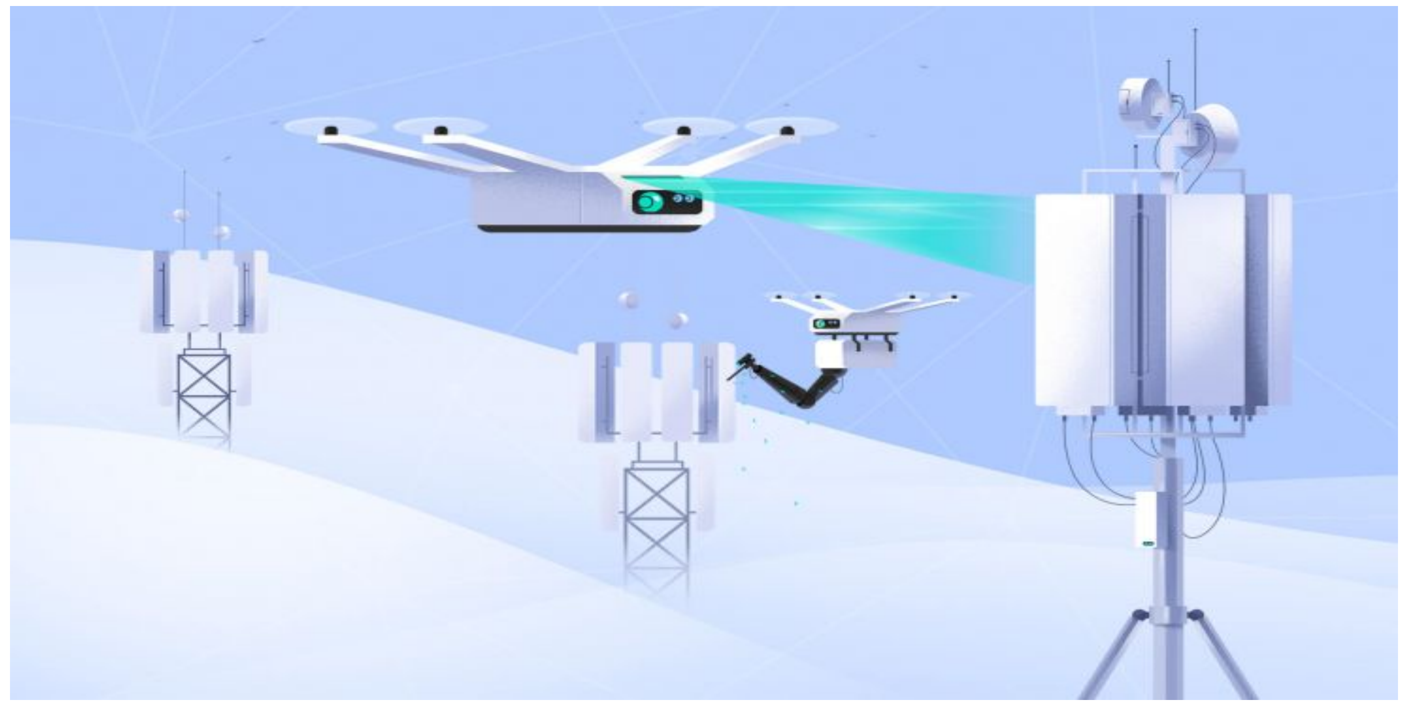
MICHIGAN ENGINEERING
UNIVERSITY OF MICHIGAN

Agile and Precise Motion Control Under Uncertainty

Drone Delivery

Target Tracking

Inspection & Maintenance



Packages with unknown weights

Unknown aero-dynamic effects

Complication: Unknown dynamics or disturbances

Problem: How to achieve agile and precise tracking control under unknown dynamics and disturbances?[1,2,3]

[1] Ackerman, IEEE Spectrum '13
[2] Chen, Liu, Shen, IROS '16
[3] Seneviratne, Dammika, et al., Acta Imeko '18

Model Predictive Control Under Uncertainty

All above scenarios are control problems under uncertainty

Goal: Find control input to minimize loss under unknown system dynamics or disturbances:

$$\begin{aligned} \underset{u_t,\ \ldots,\ u_{t+N-1}}{min} \quad & \sum_{k=t}^{t+N-1} c_k\left(x_k, u_k\right) \\ subject\ to \quad & x_{k+1} = f\left(x_k\right) + g\left(x_k\right)u_k + h\left(z_k\right), \\ & u_k \in \mathcal{U}, \\ & k \in \{t, \ldots, t+N-1\}. \end{aligned}$$

convex

known system dynamics

unknown dynamics
or disturbances

control constraints

All above scenarios are control problems under uncertainty

Goal: Find control input to minimize loss under unknown system dynamics or disturbances:

$$\begin{aligned} \min_{u_t,\ \ldots,\ u_{t+N-1}} \quad & \sum_{k=t}^{t+N-1} c_k\left(x_k, u_k\right) \\ \text{subject to} \quad & x_{k+1} = f\left(x_k\right) + g\left(x_k\right) u_k + h\left(z_k\right), \\ & u_k \in \mathcal{U}, \\ & k \in \{t, \ldots, t+N-1\}. \end{aligned}$$

convex

known system dynamics

unknown dynamics
or disturbances

control constraints

Difficulty: $h\left(z_k\right)$ is unknown and adaptive to the state and the control

All above scenarios are control problems under uncertainty

Goal: Find control input to minimize loss under unknown system dynamics or disturbances:

$$\begin{aligned} \min_{u_t,\ \ldots,\ u_{t+N-1}} \quad & \sum_{k=t}^{t+N-1} c_k\left(x_k, u_k\right) \\ \text{subject to} \quad & x_{k+1} = f\left(x_k\right) + g\left(x_k\right) u_k + \hat{h}\left(z_k; \hat{\alpha}\right), \\ & u_k \in \mathcal{U}, \\ & k \in \{t, \ldots, t+N-1\}. \end{aligned}$$

convex

known system dynamics

estimated unknown dynamics/disturbances

control constraints

Assumptions

Assumption (Function Space[4])

$h : \mathbb{R}^{d_z} \to \mathbb{R}^{d_x}$ lies in a subspace of a Reproducing Kernel Hilbert Space *(RKHS)* $\mathcal{H}$ where the kernel K is considered known[5] and can be written via a feature map $\Phi : \mathbb{R}^{d_z} \times \Theta \to \mathbb{R}^{d_x \times d_1}$ as

$$K(z_1, z_2) = \int_\Theta \Phi(z_1, \theta) \Phi(z_2, \theta)^\top \mathrm{d}\nu(\theta),$$

where $d_1 \leq d_x$, ν is a known probability measure on a measurable space Θ.

Assumption (Operator-Valued Bochner's Theorem[6])

The measurable space Θ is a subset of $\mathbb{R}^{d_z+1}$ such that $\theta \in \Theta$ can be written as $\theta = (w, b)$, where $w \in \mathbb{R}^{d_z}$ and $b \in \mathbb{R}$. Also, the feature map can be factorized as $\Phi(z, \theta) = B(w)\phi\left(w^\top z + b\right)$, where $B : \mathbb{R}^{d_z} \to \mathbb{R}^{d_x \times d_1}$ and $\phi : \mathbb{R} \to [-1, 1]$ is a 1-Lipschitz function.

[4] Bach, Journal of Machine Learning Research '17

[5] Carmel, et al., Analysis and Applications '16

[6] Brault, et al., Asian Conference on Machine Learning '16

Assumptions

Definition (Value Function[7])

Given a state x and parameter $\hat{\alpha}$, the *value function* $V_t\left(x;\hat{\alpha}\right)$ is defined as the optimal value of

$$
\begin{aligned}
\underset{u_t,\ldots,\ u_{t+N-1}}{min} \quad & \sum_{k=t}^{t+N-1} c_k\left(x_k, u_k\right) \\
subject\ to \quad & x_{k+1} = f\left(x_k\right) + g\left(x_k\right) u_k + \hat{h}\left(z_k\right), \\
& x_t = x,\ u_k \in \mathcal{U}, \\
& k \in \{t, \ldots, t+N-1\}.
\end{aligned}
$$

[7] Grimm, et al., TAC '05

Assumptions

Assumption (Stability Condition)

There exist positive scalars $\underline{\lambda}$, $\overline{\lambda}$, and a continuous function $\sigma : \mathbb{R}^{d_x} \to \mathbb{R}_+$, such that

1. $c_t(x, u) \geq \underline{\lambda}\sigma(x), \forall x,\ u,\ t;$
2. $V_t(x; \hat{\alpha}) \leq \overline{\lambda}\sigma(x), \forall x,\ t;$
3. $\lim_{\|x\| \to \infty} \sigma(x) \to \infty.$

Assumption (Lipschitzness of Value Function)

Assume that $c_t(x, u)$ is locally Lipschitz in x and u, and $\hat{h}(\cdot)$ is locally Lipschitz in $\hat{\alpha}$. Then, the value function is Lipschitz in the initial condition x and the parameter $\hat{\alpha}$ over bounded domain sets.

Remark: The stability condition ensures that the system $x_{k+1} = f(x_k) + g(x_k)\ u_k + \hat{h}(z_k; \hat{\alpha})$ is globally asymptotic stable under MPC policy[7].

[7] Grimm, et al., TAC '05

Assumptions

Assumption (Input-to-State-Stability of MPC against Estimation Errors)

The MPC policy renders the estimated system dynamics $x_{k+1} = f(x_k) + g(x_k)\,u_k + \hat{h}(z_k)$ input-to-state-stable against the estimation error $e_t \triangleq \hat{h}(z_t) - h(z_t)$: there exists a class $\mathcal{KL}$ function β and a class $\mathcal{K}$ function γ, such that

$$\|x_t\| \leq \beta\left(\|x_{t_0}\|,\ t - t_0\right) + \gamma\left(e_{max}\right),\ \forall t \geq t_0.$$

where $e_{max} \triangleq \sup_{t_0 \leq \tau \leq t} \|e_\tau\|$,

Suboptimality Metric against Optimal Control Policies in Hindsight

Definition (Dynamic Regret)

Assume a total time horizon of operation T, and loss functions c_t, $t = 1, \ldots, T$. Then, *dynamic regret* is

$$\text{Regret}_T^D = \sum_{t=1}^{T} c_t\left(x_t, u_t, h(z_t)\right) - \sum_{t=1}^{T} c_t\left(x_t^\star, u_t^\star, h(z_t^\star)\right),$$

where $x_t^\star$ and $u_t^\star$ are the optimal trajectory and control input in hindsight, and the cost c_t depends on the unknown disturbance h explicit.

Remark:

- The regret is sublinear if $\lim_{T\to\infty} \frac{\text{Regret}_T^D}{T} \to 0$, which implies $c_t\left(x_t, u_t, h(z_t)\right) - c_t\left(x_t^\star, u_t^\star, h(z_t^\star)\right) \to 0$.
- h adapts (possibly differently) to the state and control sequences (x_1, u_1), $\ldots$, (x_T, u_T) and $(x_1^\star, u_1^\star)$, $\ldots$, $(x_T^\star, u_T^\star)$ since h is a function of the state and the control input.

Problem

At each $t = 1, \ldots, T$,

- estimate the unknown disturbance $\hat{h}\,(\cdot)$;
- identify a control input u_t using MPC.

The goal is to minimize dynamic regret:

$$\sum_{t=1}^{T} c_t\left(x_t, u_t, h(z_t)\right) - \sum_{t=1}^{T} c_t\left(x_t^\star, u_t^\star, h(z_t^\star)\right).$$

State of the Art

Offline Learning for Control[8]

- collects data offline and trains neural-networks or Gaussian-process models BUT
  - data-collection can be expensive and time-consuming
  - may not generalize to unseen environments

Robust Control[9]

- selects control inputs over a lookahead horizon BUT
  - conservative since assuming worst-case disturbances

[8] Sánchez-Sánchez et al., '18; Carron et al., RAL '19; Torrente et al., RAL '21; Shi et al., ICRA '19; O'Connell, et al., SR '22; …
[9] Goel et al., '20; Sabag et al., ACC '21; Goel et al., L4DC '21; Martin et al., L4DC '22; Didier et al., L-CSS '22; Zhou et al., CDC '23; …

State of the Art

Adaptive Control[10]

- estimates uncertainty and compensates control input with estimated uncertainty BUT
  - do not learn a model of uncertainty for predictive control

Non-Stochastic Control[11]

- updates control input online to adapt to observed uncertainty BUT
  - sensitive to tuning parameters
  - do not learn a model of uncertainty for predictive control

[10] Slotine, '91; Krstic, et al., '95; Ioannou et al., '96; Tal et al., TCST '20; Wu, et al., '23; Das et al., '24; Jia, et al., TRO ’23;…

[11] Agarwal et al., ICML '19; Hazan et al., ALT '20; Gradu et al., L4DC '23; Zhou et al., CDC '23; Zhou et al., RAL '23;...

Initialization:

- Gradient descent learning rate η, number of random Fourier features M, domain set $\mathcal{D}$, estimated parameter $\hat{\alpha}_{i,1} \in \mathcal{D}$;
- Randomly sample $\theta_i \sim \nu$ and formulate $\Phi\left(\cdot, \theta_i\right)$, where $i \in \{1, \ldots, M\}$;

At each iteration $t = 1, \ldots, T$:

1. Apply control input u_t using MPC with $\hat{h}(\cdot) \triangleq \frac{1}{M}\sum_{i=1}^{M} \Phi\left(\cdot, \theta_i\right) \hat{\alpha}_{i,t}$;
2. Observe state x_{t+1}, and calculate disturbance via $h\left(z_t\right) = x_{t+1} - f(x_t) - g(x_t)u_t$;
3. Formulate estimation loss $l_t\left(\hat{\alpha}_t\right) \triangleq \|h\left(z_t\right) - \frac{1}{M}\sum_{i=1}^{M} \Phi\left(z_t, \theta_i\right) \hat{\alpha}_{i,t}\|^2$;
4. Calculate gradient $\nabla_t \triangleq \nabla_{\hat{\alpha}_t} l_t\left(\hat{\alpha}_t\right)$;
5. Update $\hat{\alpha}'_{t+1} = \hat{\alpha}_t - \eta \nabla_t$;
6. Project $\hat{\alpha}'_{i,t+1}$ onto $\mathcal{D}$, *i.e.*, $\hat{\alpha}_{i,t+1} = \Pi_{\mathcal{D}}(\hat{\alpha}'_{i,t+1})$, for $i \in \{1, \ldots, M\}$.

Dynamic Regret Analysis

Theorem (No-Regret)

Our algorithm with $\eta = \mathcal{O}\left(1/\sqrt{T}\right)$ achieves $\text{Regret}_T^D \leq \mathcal{O}\left(T^{\frac{3}{4}}\right)$.

Remark:

- Our algorithm converges asymptotically to the optimial controller since $\lim_{T\to\infty} \frac{\text{Regret}_T^D}{T} \to 0$.

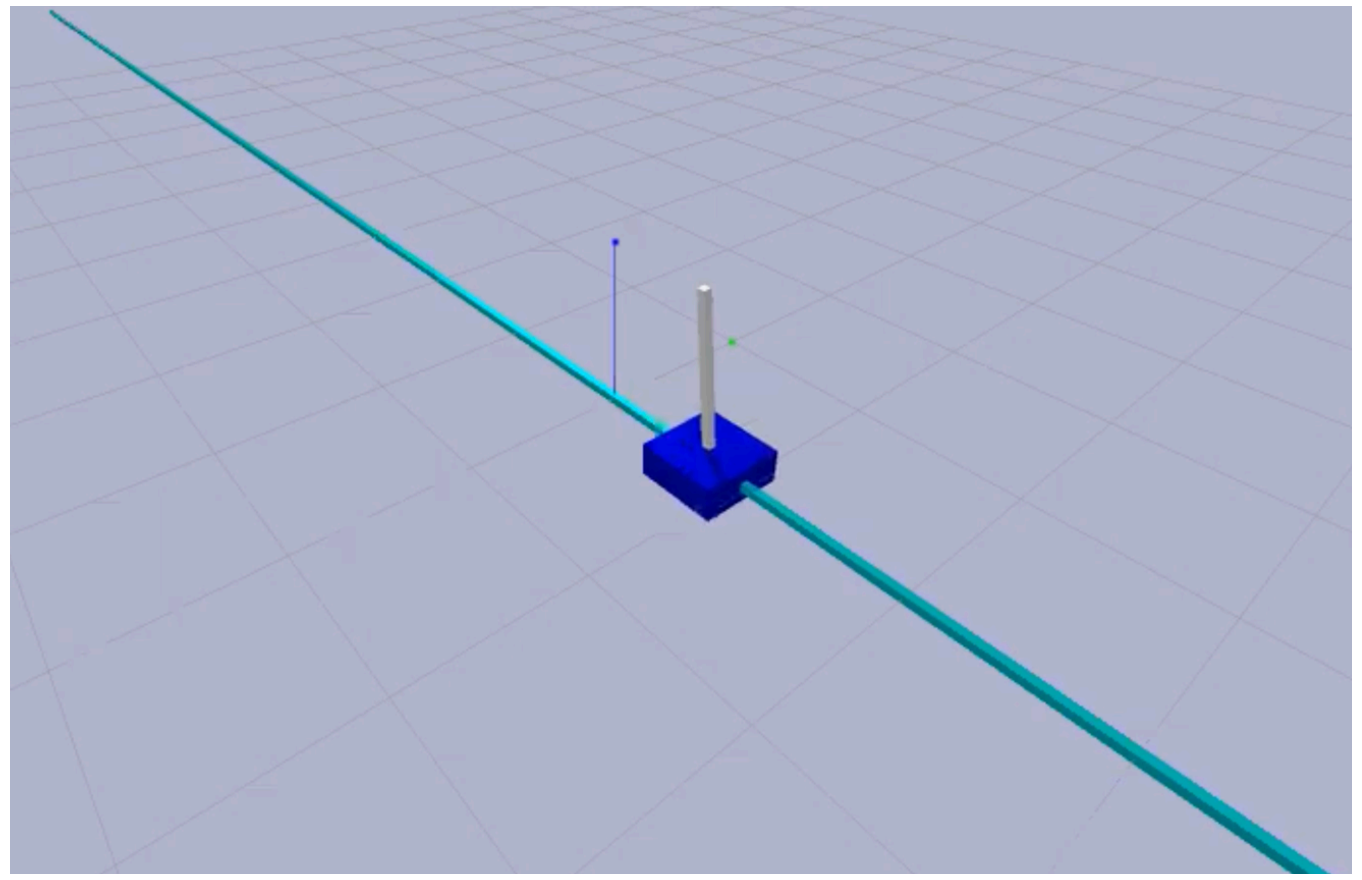
Simulations on Cart-Pole Stabilization with Inaccurate Model

Goal:

- Stabilize the cart-pole around the upright position of the pole

Setup:

- The model parameters (pole length & mass, cart mass) are inaccurate



Simulations on Cart-Pole Stabilization with Inaccurate Model

Goal:

- Stabilize the cart-pole around the upright position of the pole

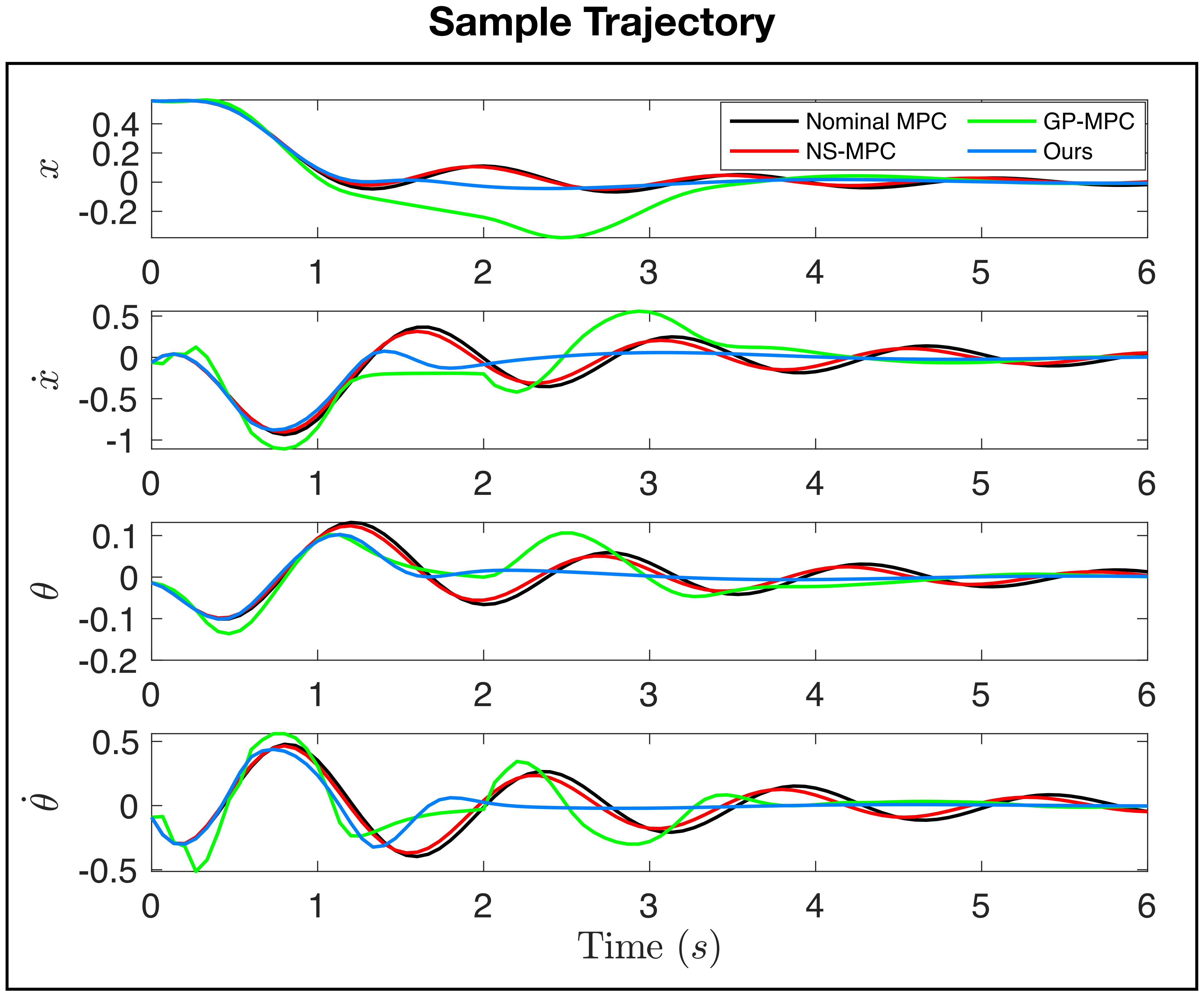
Setup:

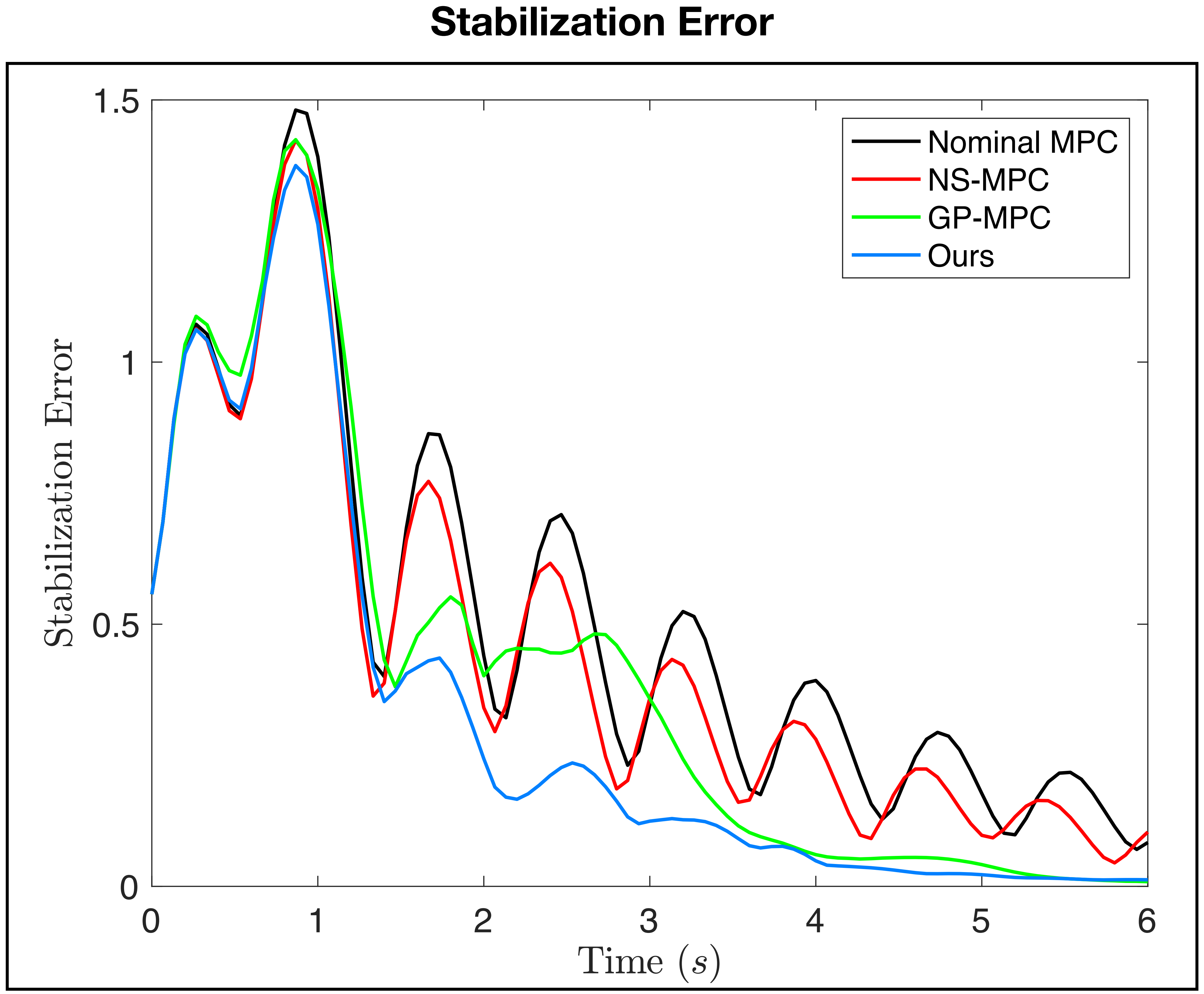
- The model parameters (pole length & mass, cart mass) are inaccurate

Compared algorithms: (i) Non-stochastic MPC[12] and (ii) Gaussian Process MPC[13]

[12] Zhou et al., RAL '23 [13] Hewing et al., TCST '19

Simulations on Cart-Pole Stabilization with Inaccurate Model





Results:

- Our method achieves fastest stabilization while other algorithms:
  - NS-MPC has marginal improvement over Nominal MPC
  - GP-MPC has larger deviation than our method

Simulations on Cart-Pole Stabilization with Inaccurate Model

TABLE I: **Computational Performance.** The red numbers correspond to the worse performance.

	MPC	NS-MPC	GP-MPC	Ours
Time (ms)	9.63 ± 3.34	9.94 ± 3.39	160.01 ± 26.96	16.28 ± 21.77

Results:

- Our method is computationally more efficient than GP-MPC

Simulations on Trajectory Tracking with Unknown Aerodynamic Effect

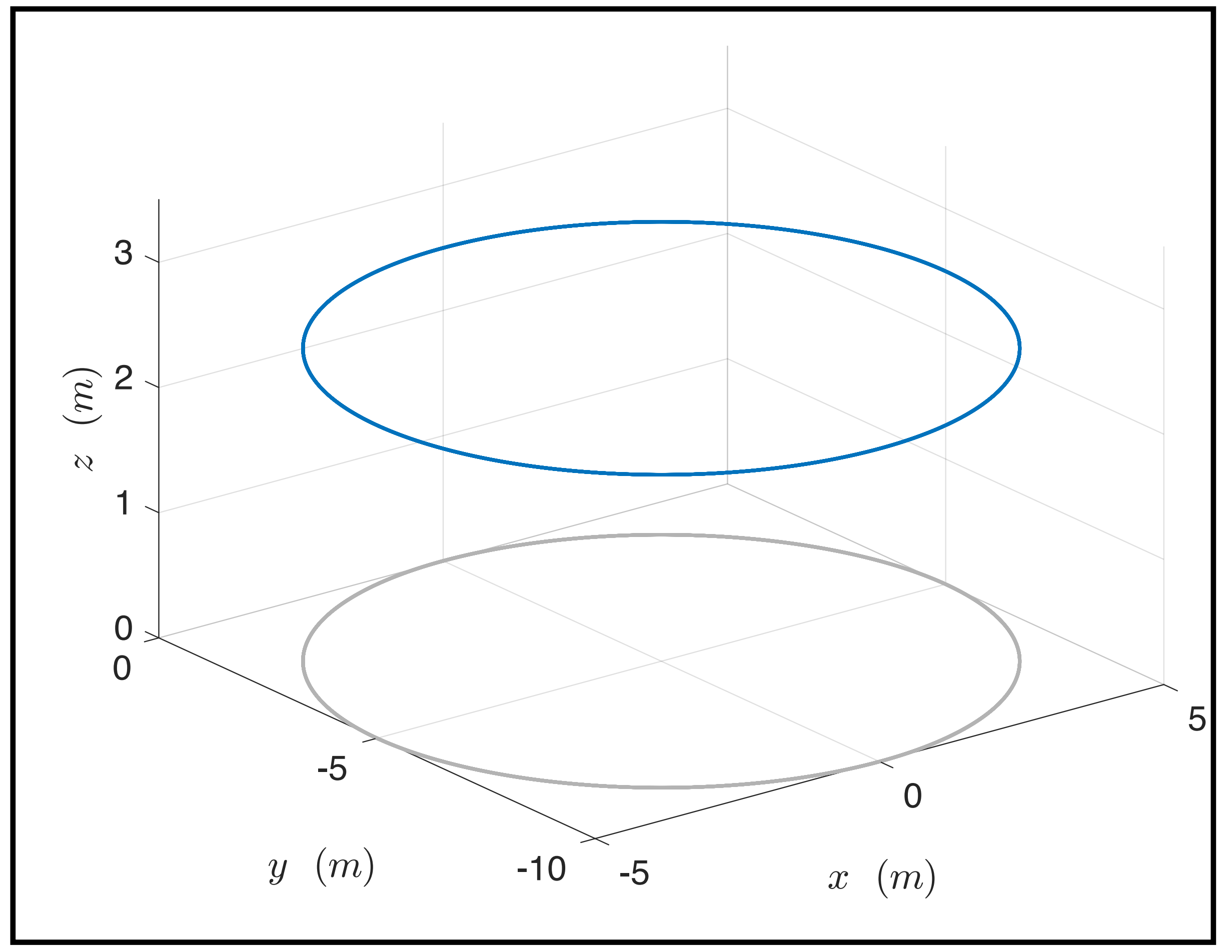
Goal:

- Track reference trajectories with a drone

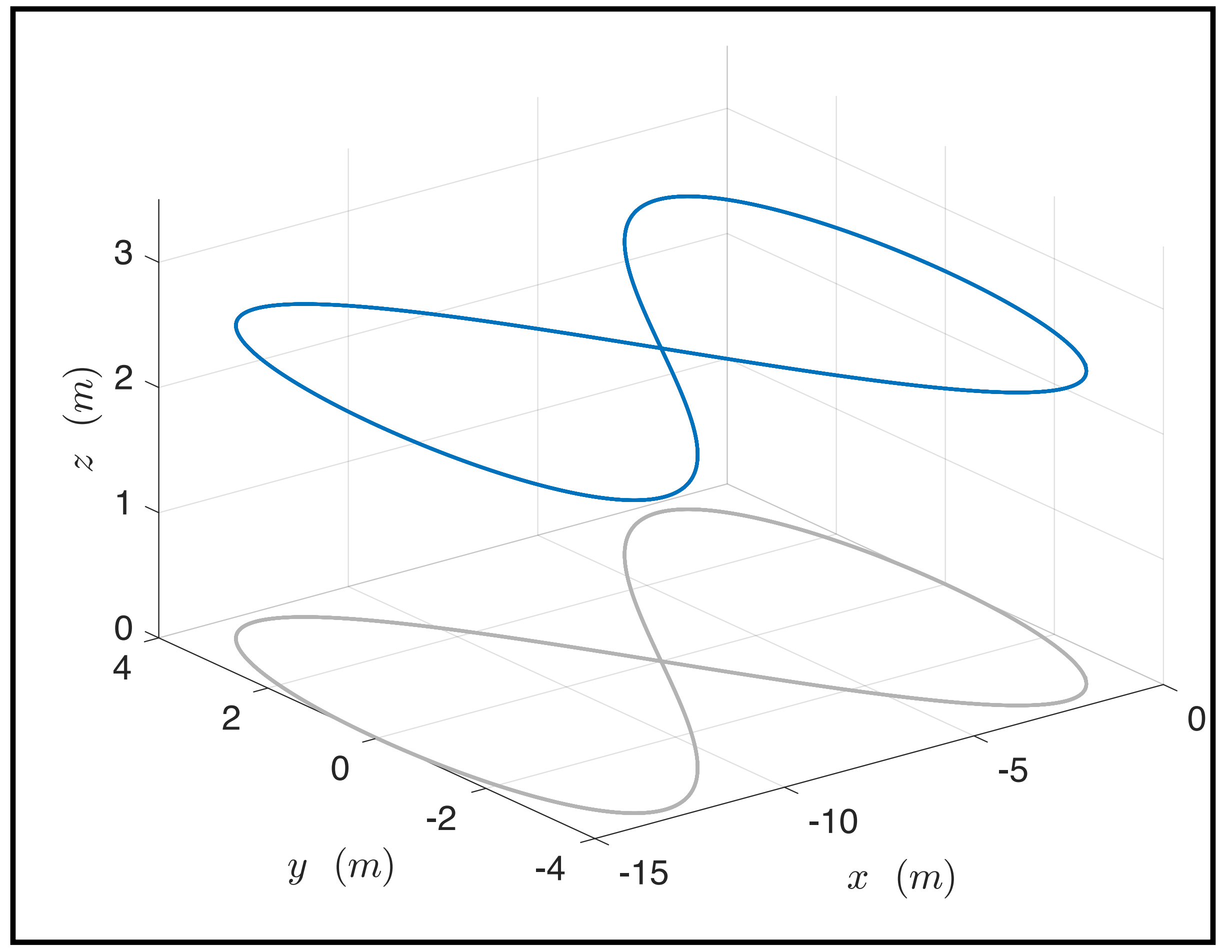
Setup:

- The system dynamics of the drone are corrupted with unknown drag effects

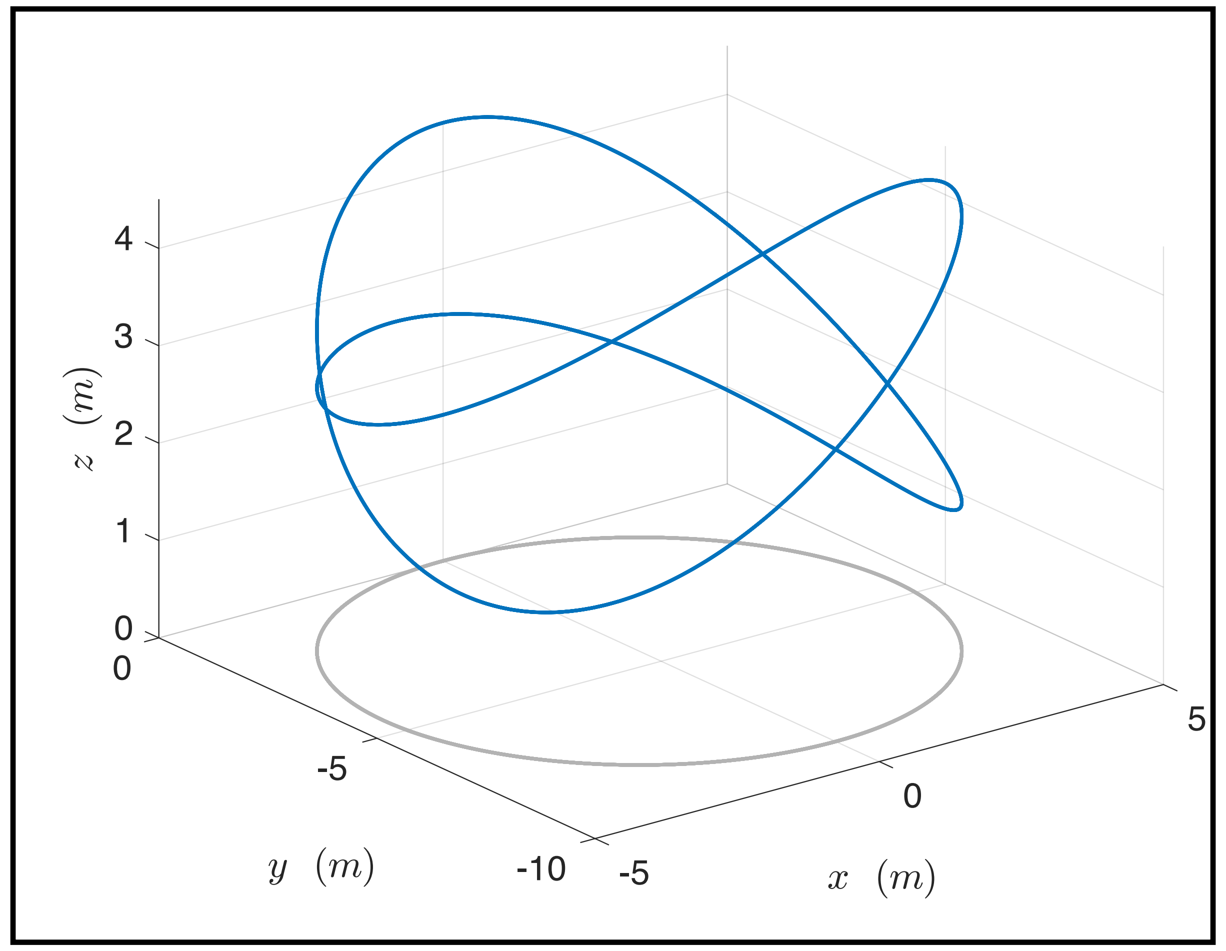
Circle



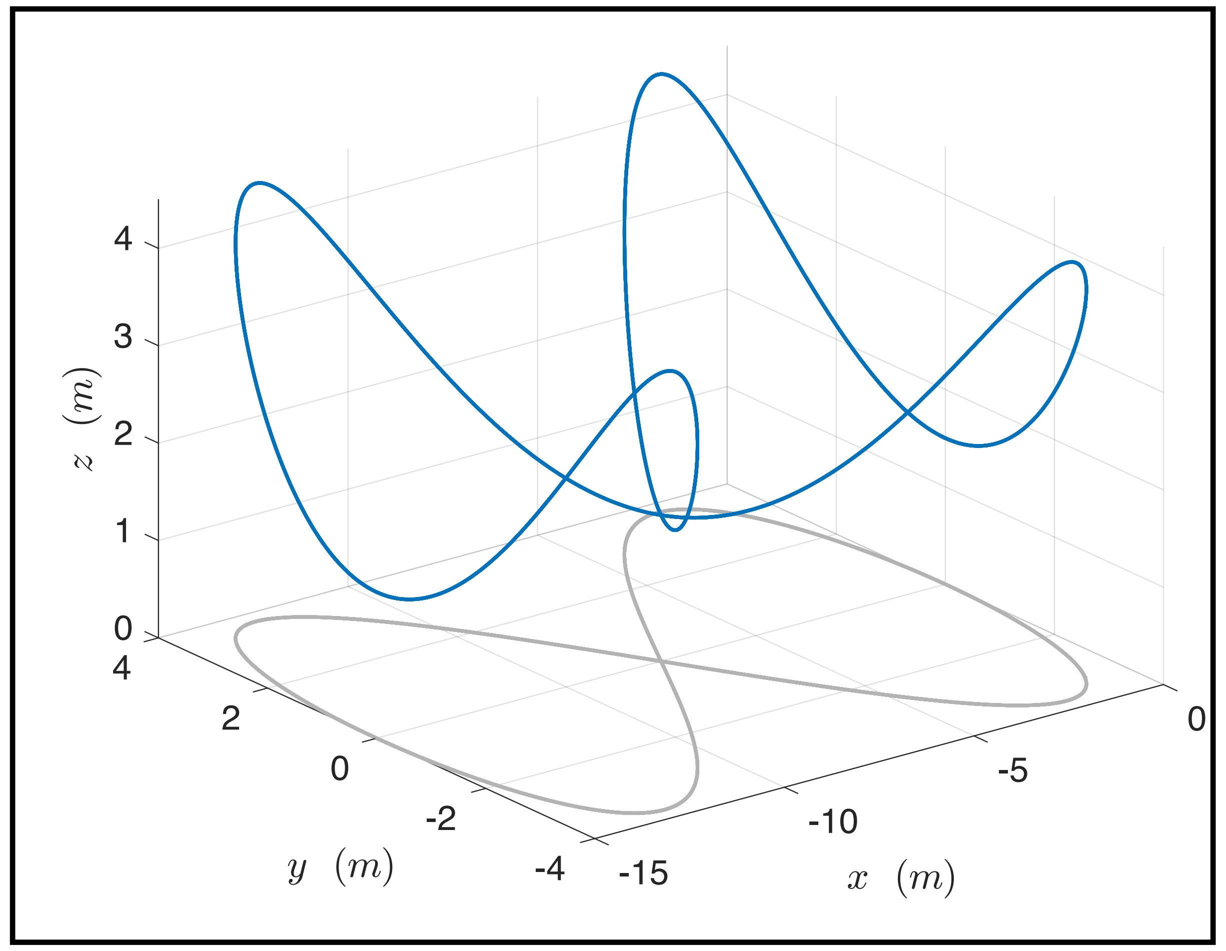
Lemniscate



Wrapped Circle



Wrapped Lemniscate



Simulations on Trajectory Tracking with Unknown Aerodynamic Effect

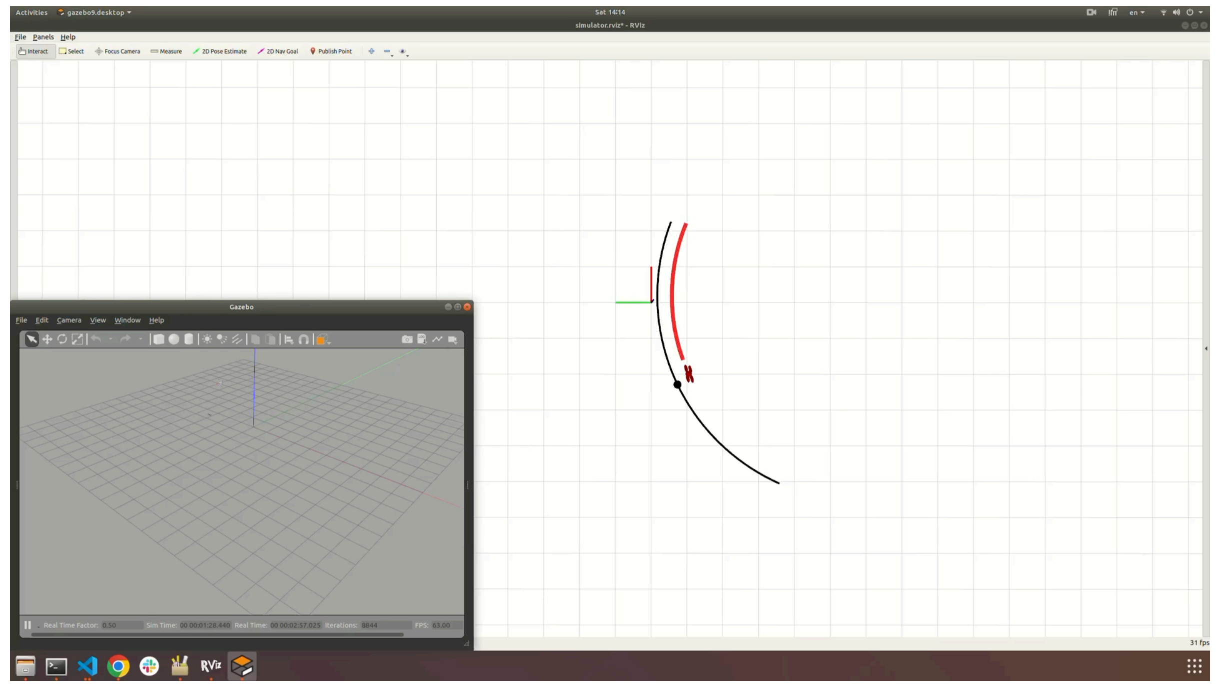
Goal:

- Track reference trajectories with a drone

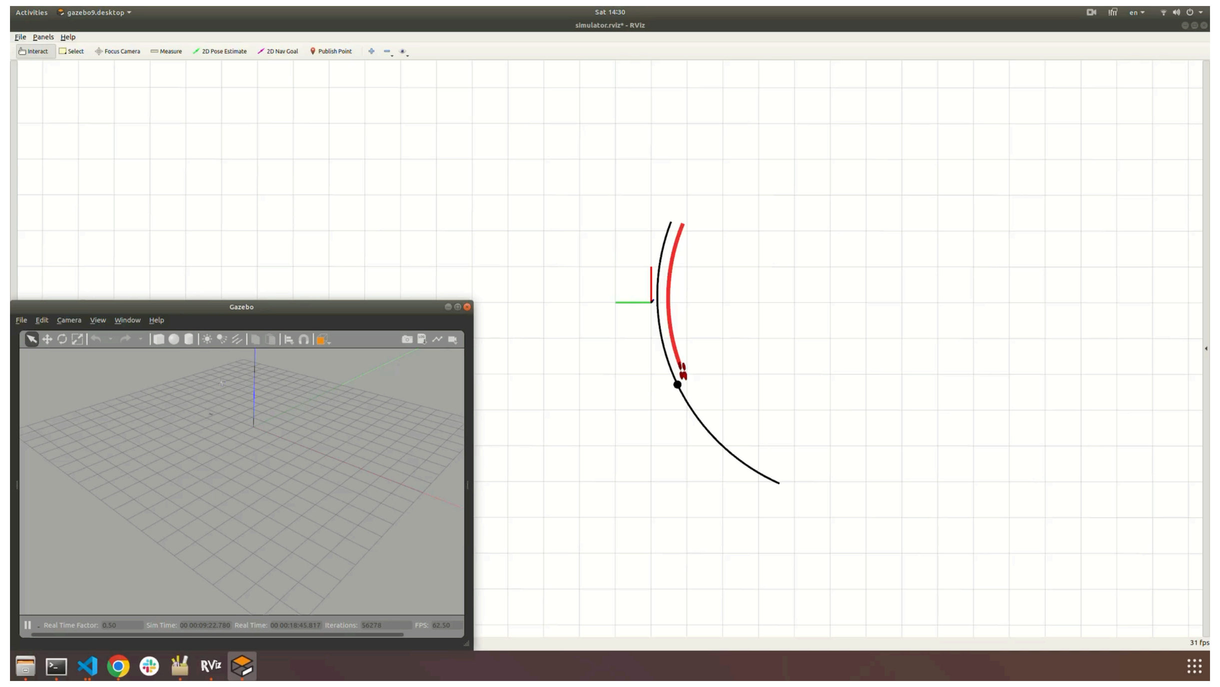
Setup:

- The system dynamics of the drone are corrupted with unknown drag effects

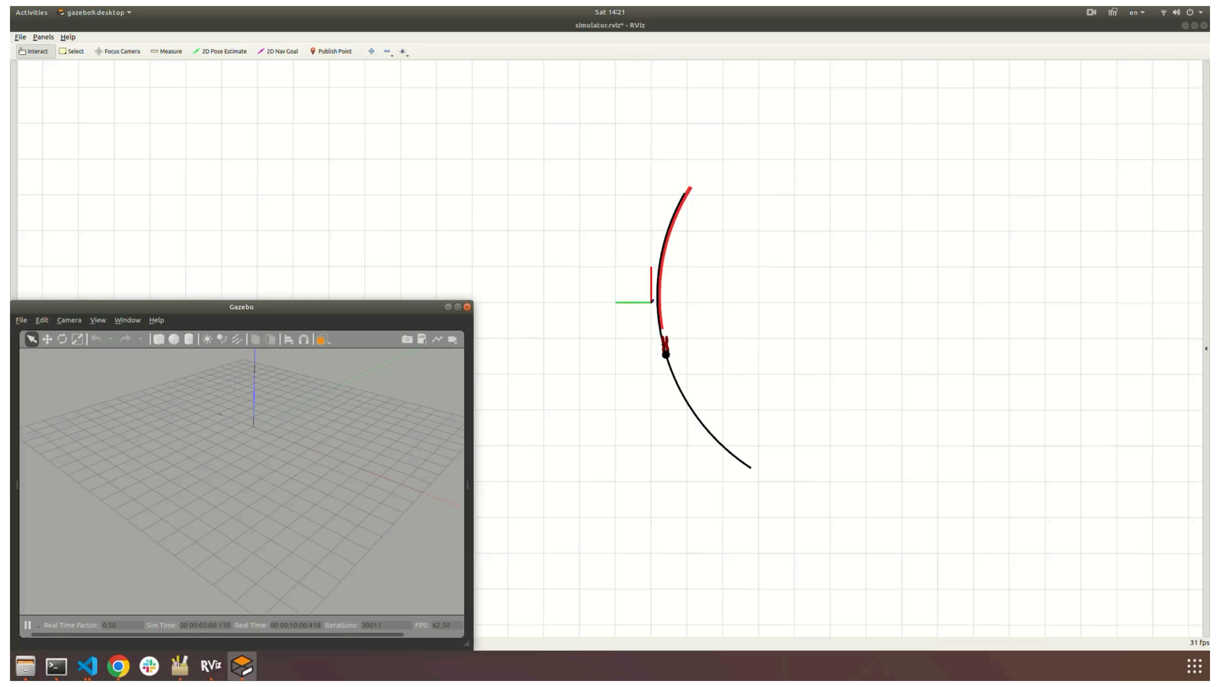
Compared algorithms: (i) Nominal MPC, (ii) Gaussian Process MPC[14], and (iii) Ours w/ INDI inner loop[15]



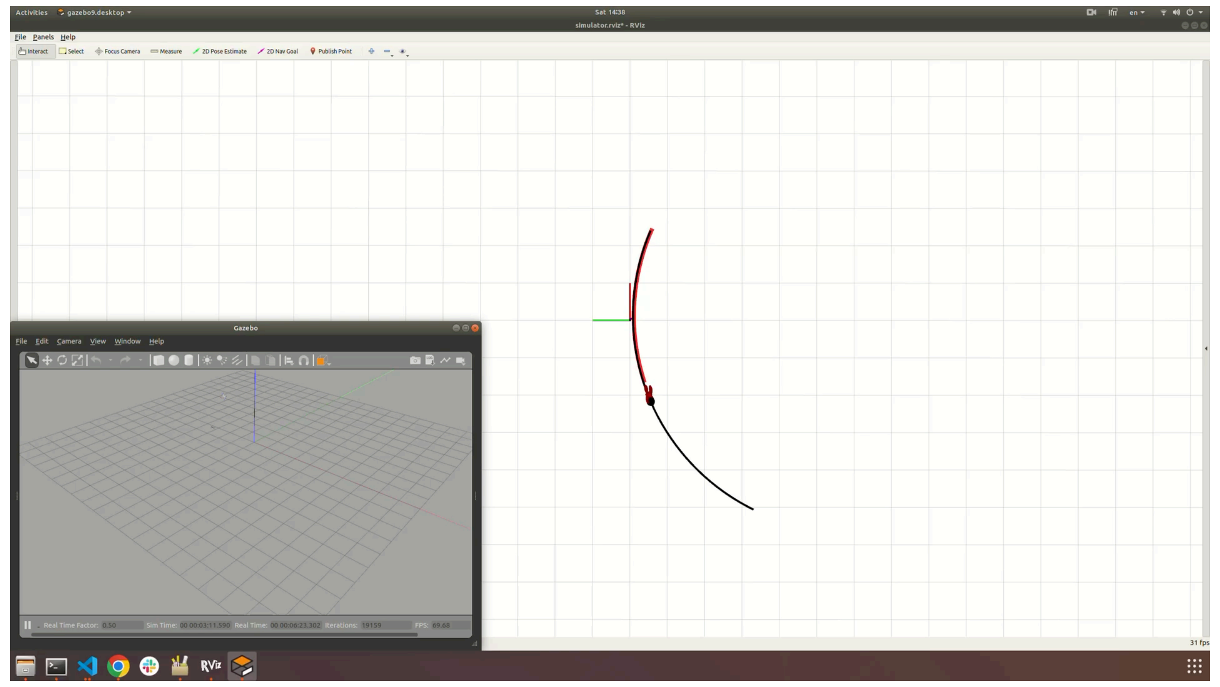
Nominal MPC



GP-MPC



Ours



Ours w/ INDI

[14] Z Torrente et al., RAL '21 [15] Tal et al., TCST '20

Simulations on Trajectory Tracking with Unknown Aerodynamic Effect

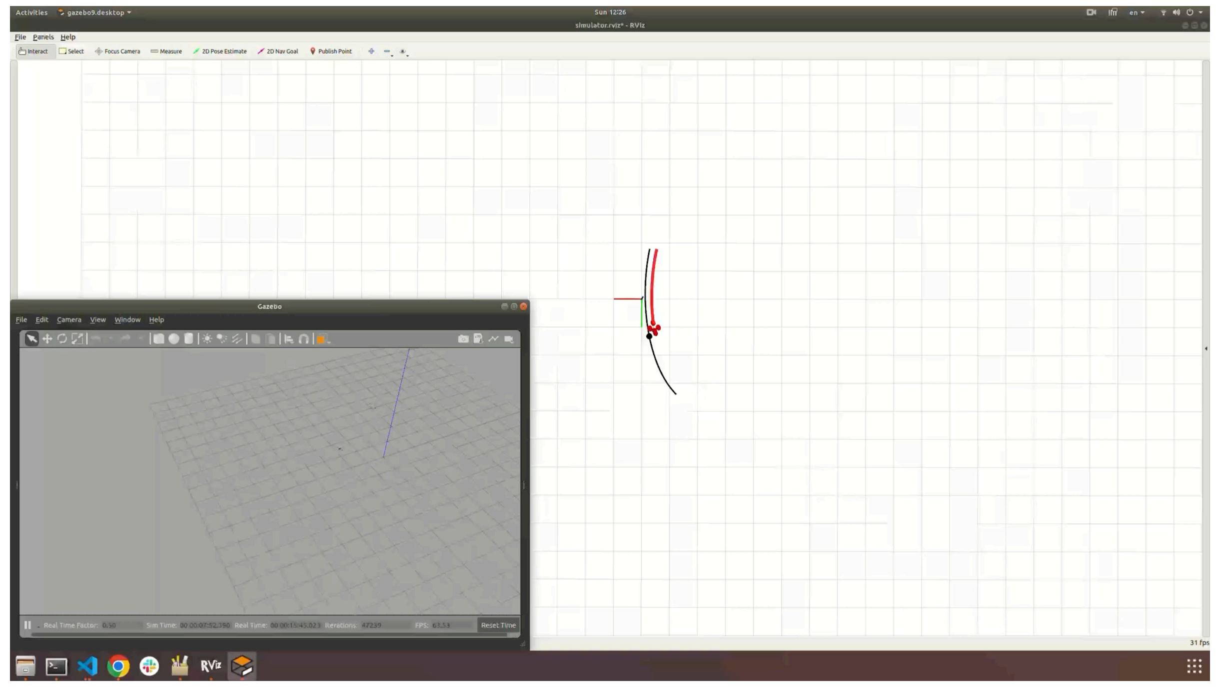
Goal:

- Track reference trajectories with a drone

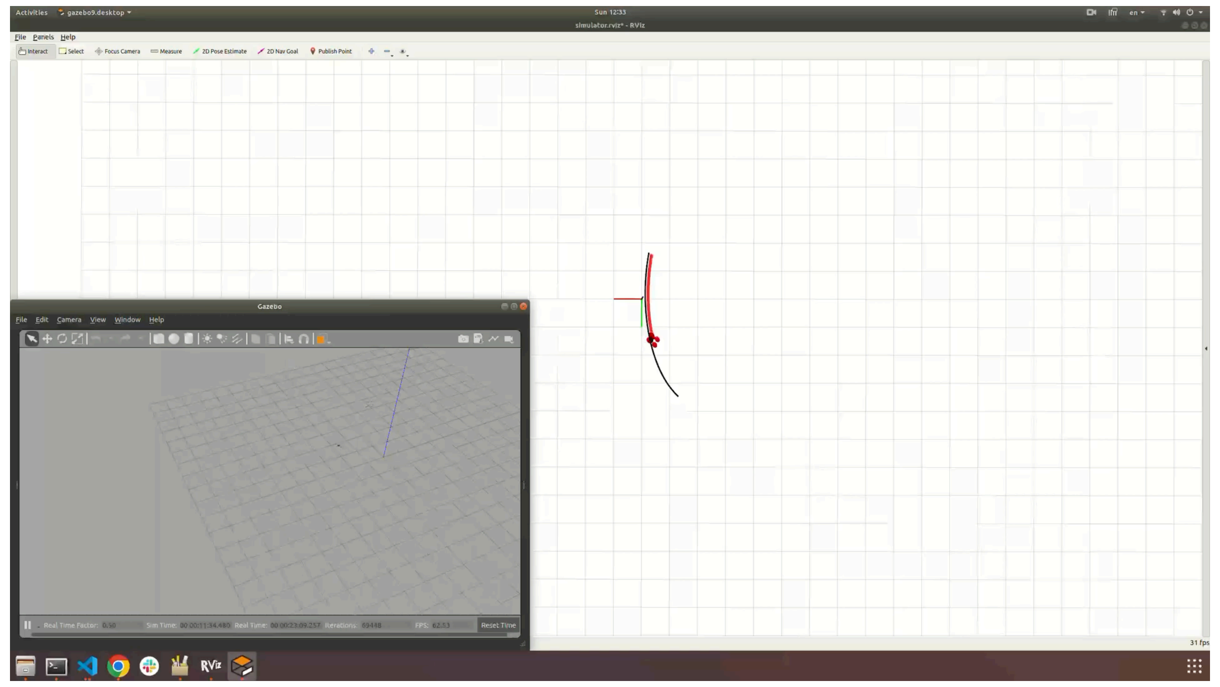
Setup:

- The system dynamics of the drone are corrupted with unknown drag effects

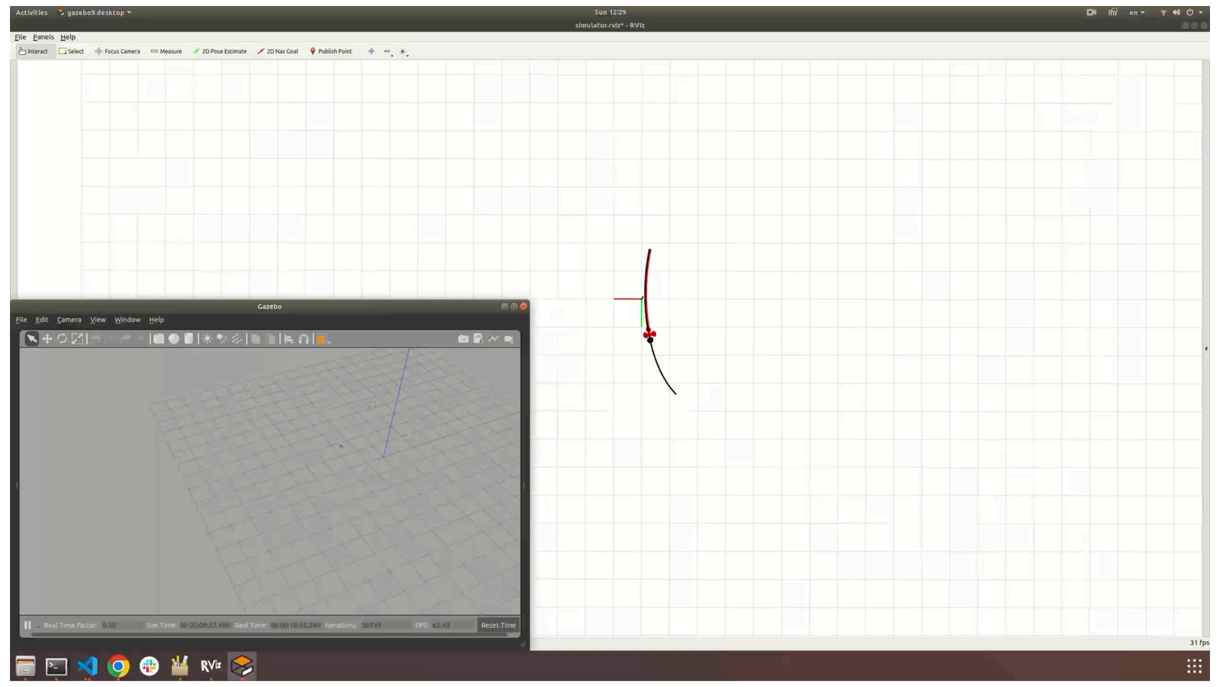
Compared algorithms: (i) Nominal MPC, (ii) Gaussian Process MPC[14], and (iii) Ours w/ INDI inner loop[15]



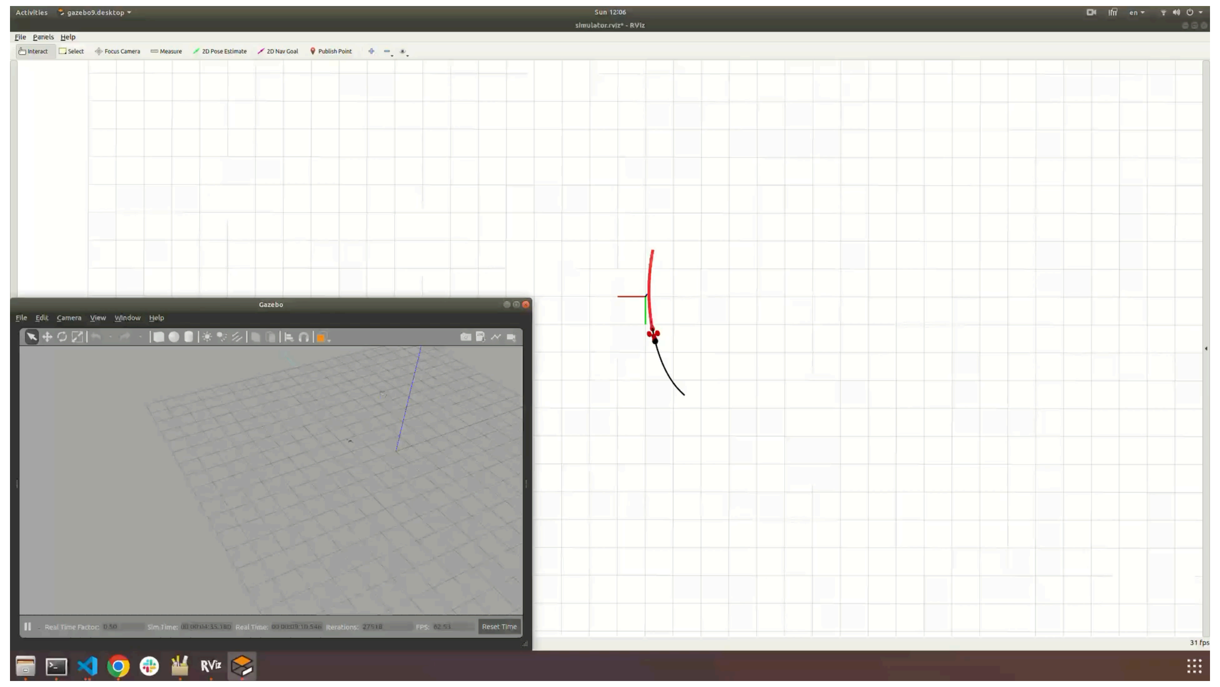
Nominal MPC



GP-MPC



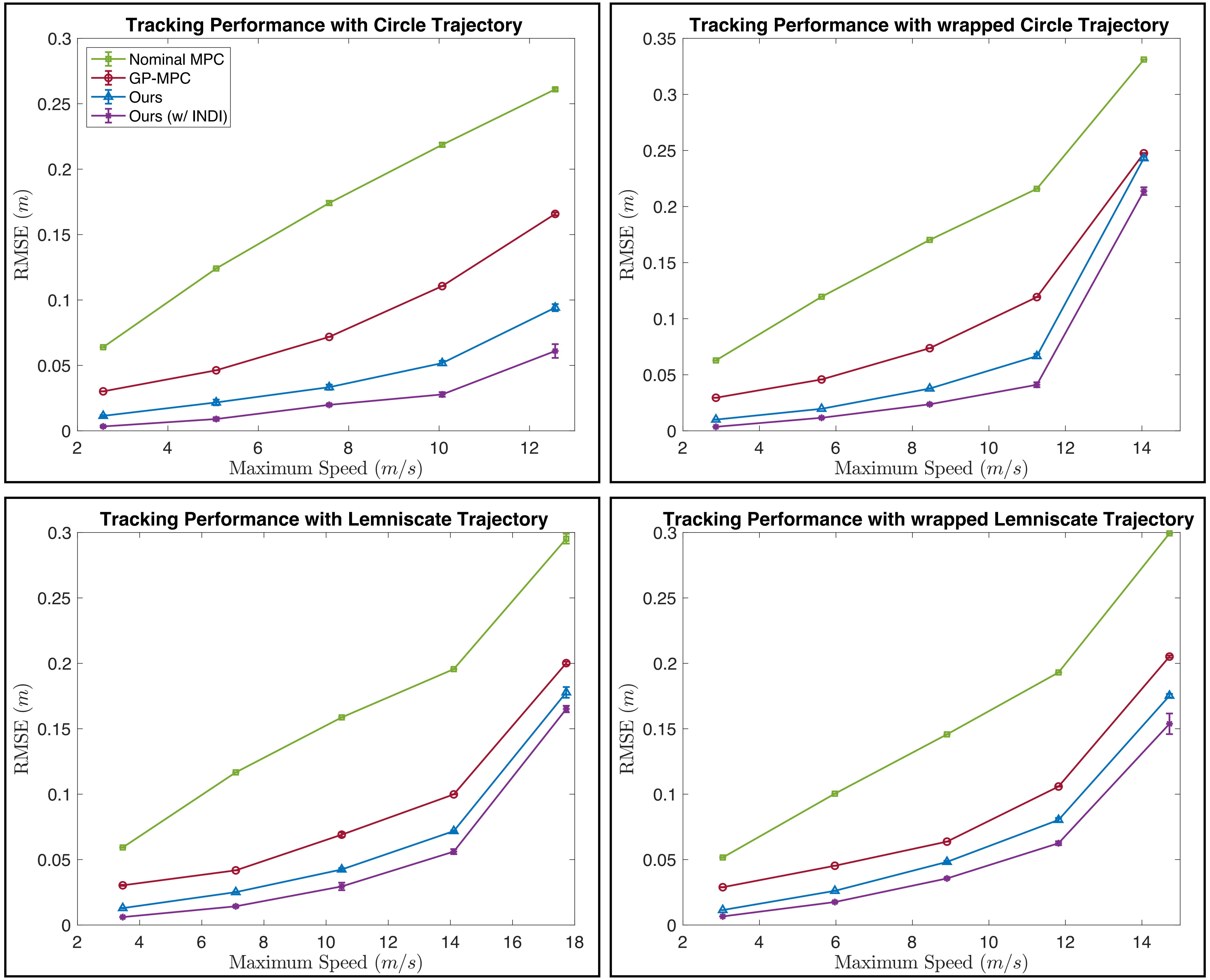
Ours



Ours w/ INDI

[14] Z Torrente et al., RAL '21 [15] Tal et al., TCST '20

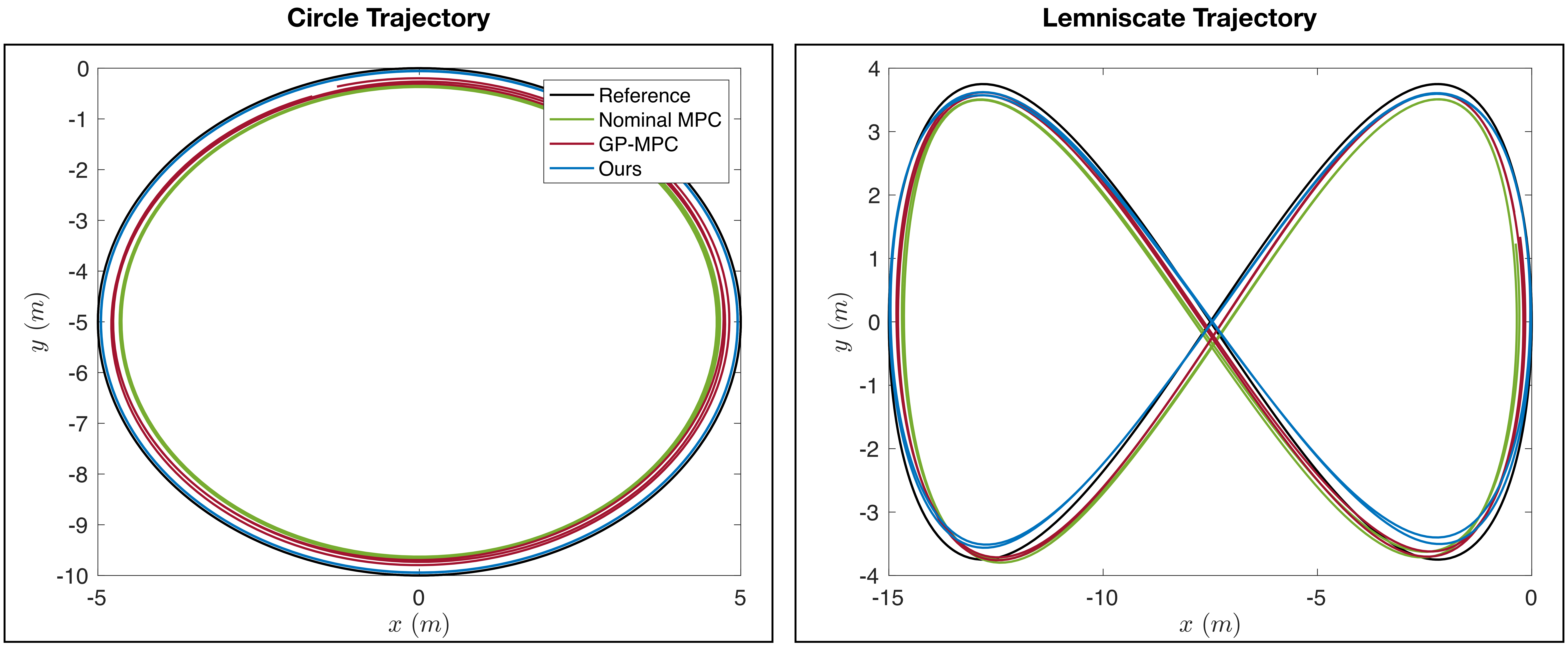
Simulations on Trajectory Tracking with Unknown Aerodynamic Effect



Results:

- Ours achieves lower tracking errors than GP
- Ours + INDI performs the best

Simulations on Trajectory Tracking with Unknown Aerodynamic Effect



Results:

- Ours achieves lower tracking errors than GP-MPC

Hardware on Trajectory Tracking with Unknown Aerodynamic Effect

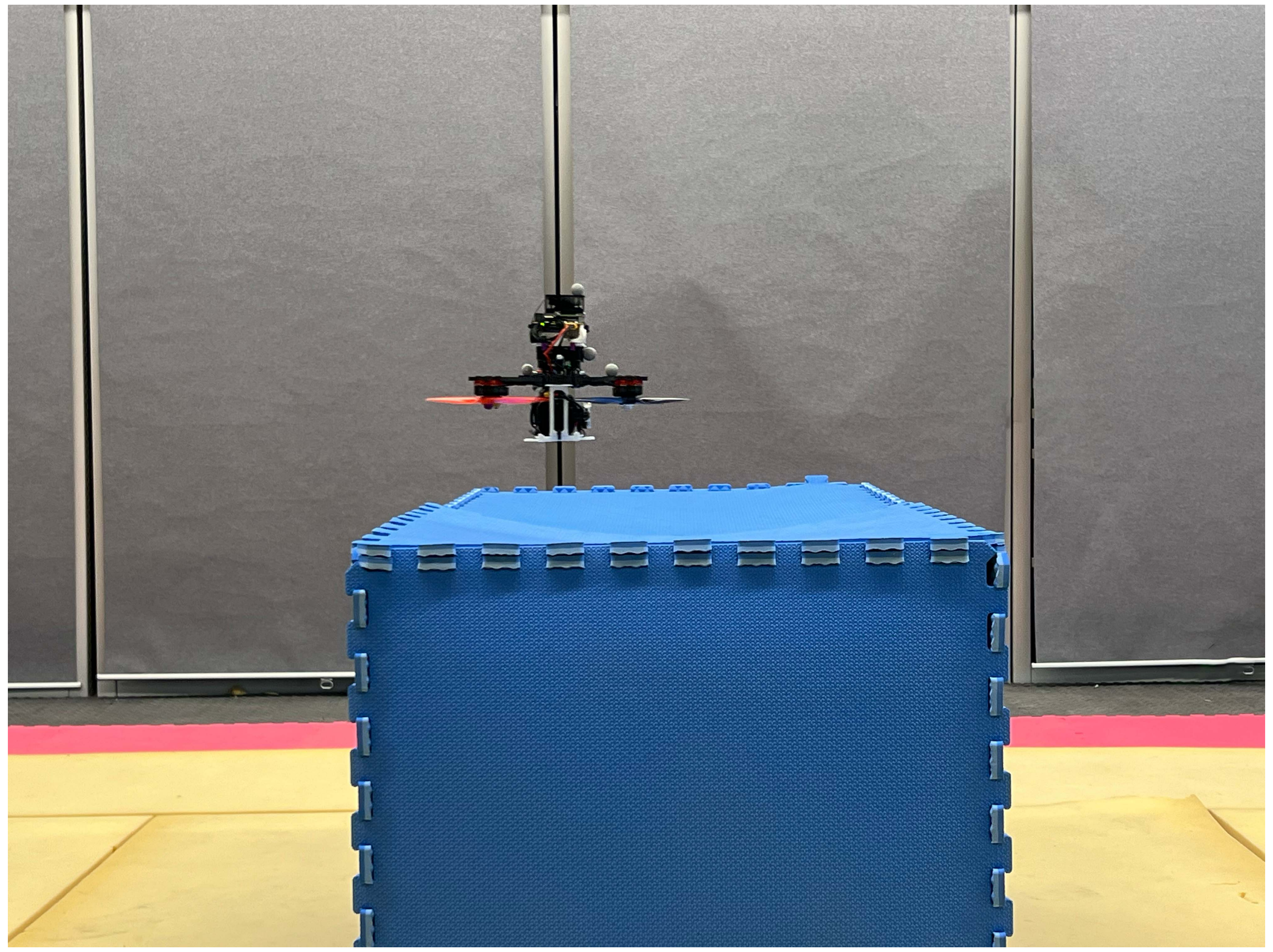
Goal:

- Track a circular trajectory with a drone

Setup:

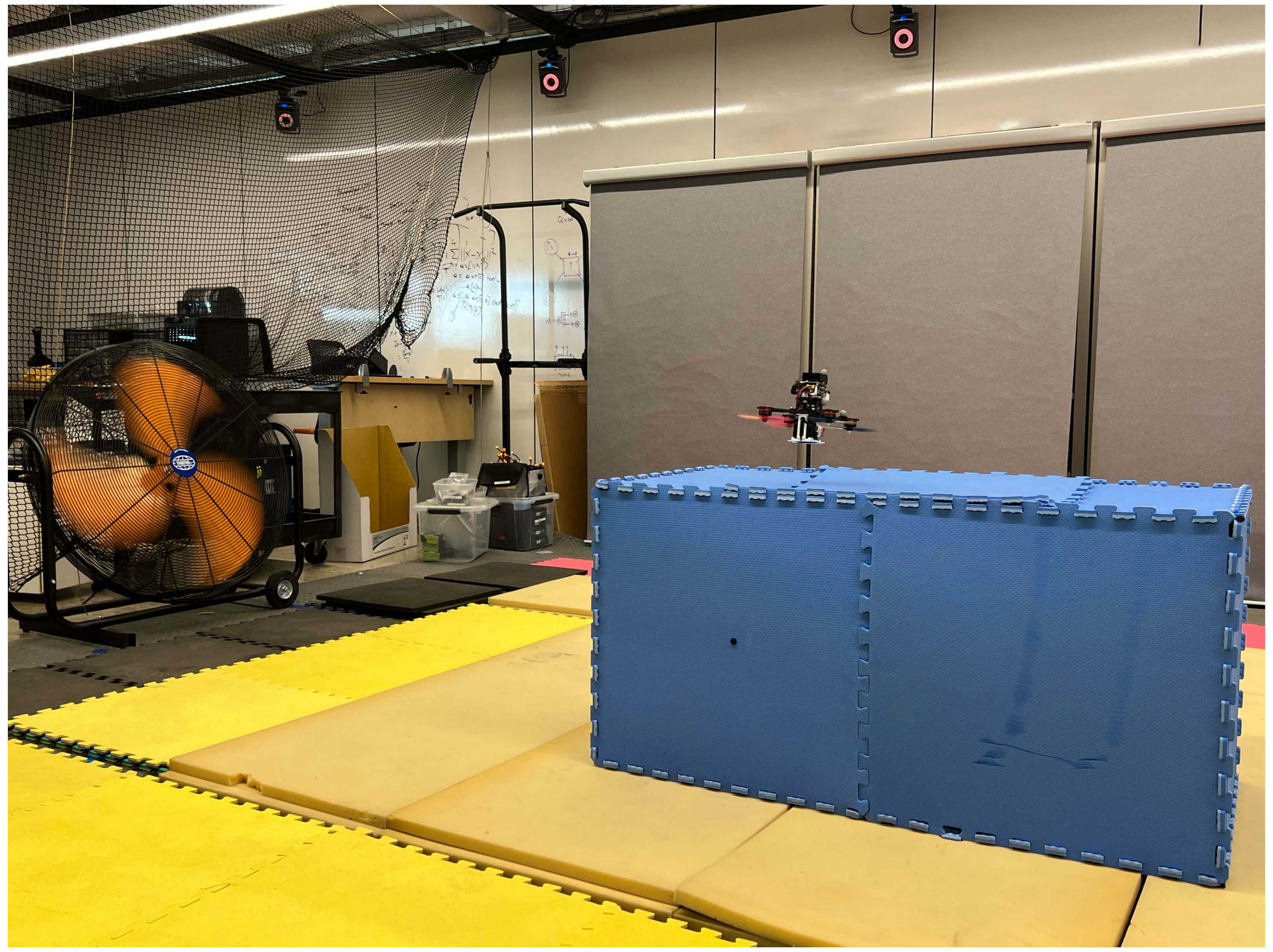
- The circular trajectory is $1m$ in diameter
- The speed is 0.8 m/s
- The drone suffers from:

(i) ground effect



(ii) wind disturbances

(iii) ground effect + wind disturbances



Hardware on Trajectory Tracking with Unknown Aerodynamic Effect

Goal:

- Track a circular trajectory with a drone

Setup:

- The circular trajectory is $1m$ in diameter
- The speed is 0.8 *m/s*
- The drone suffers from: (i) ground effect, (ii) wind disturbances, and (iii) ground effect + wind disturbances

Compared algorithms: (i) Nominal MPC and (ii) L1 adaptive MPC[16]

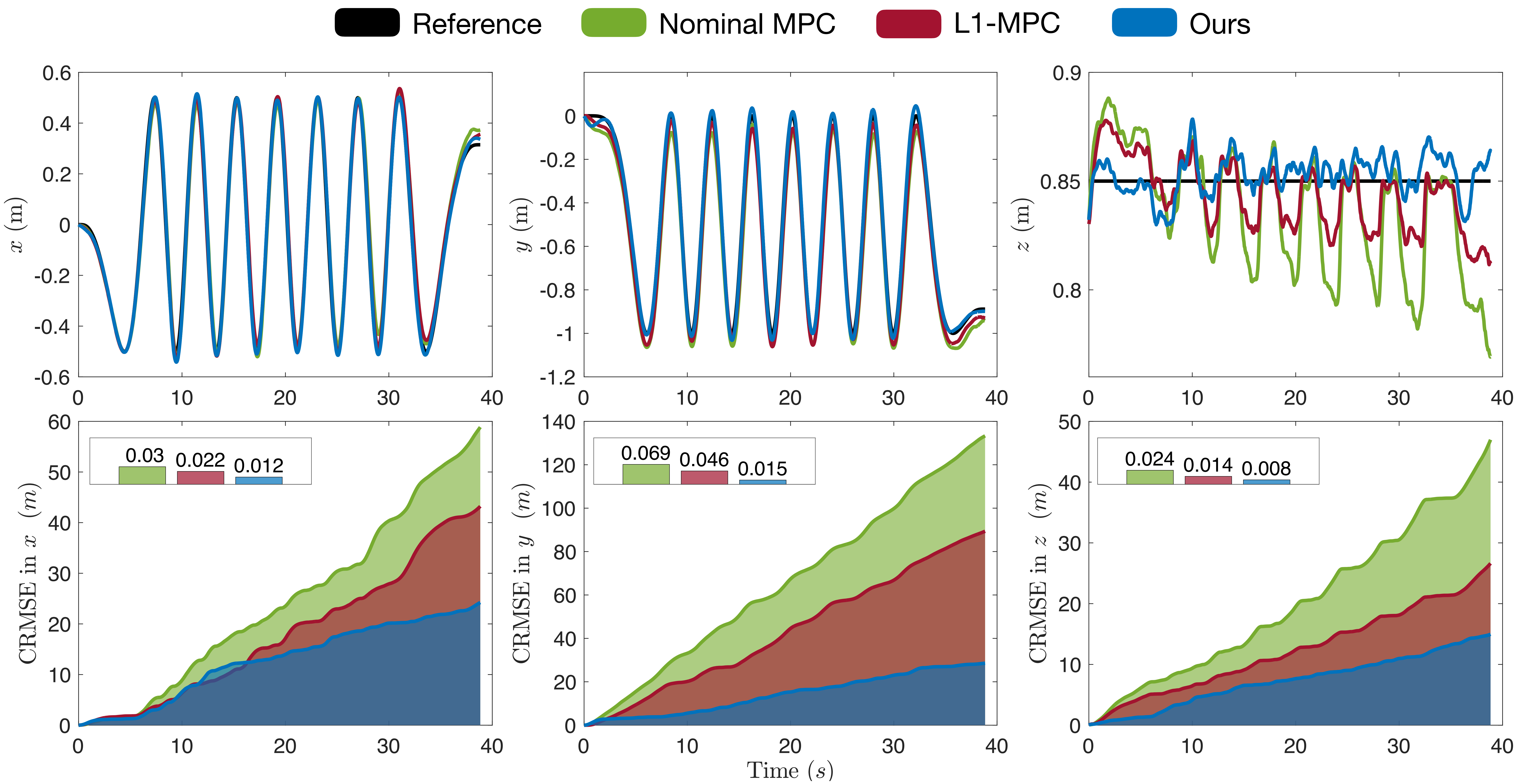
[16] Wu et al., arxiv '23

Hardware on Trajectory Tracking with Unknown Aerodynamic Effect

Results:

- Ours achieves lower tracking errors than L1-MPC
- Nominal MPC crashes under ground effect & wind disturbances

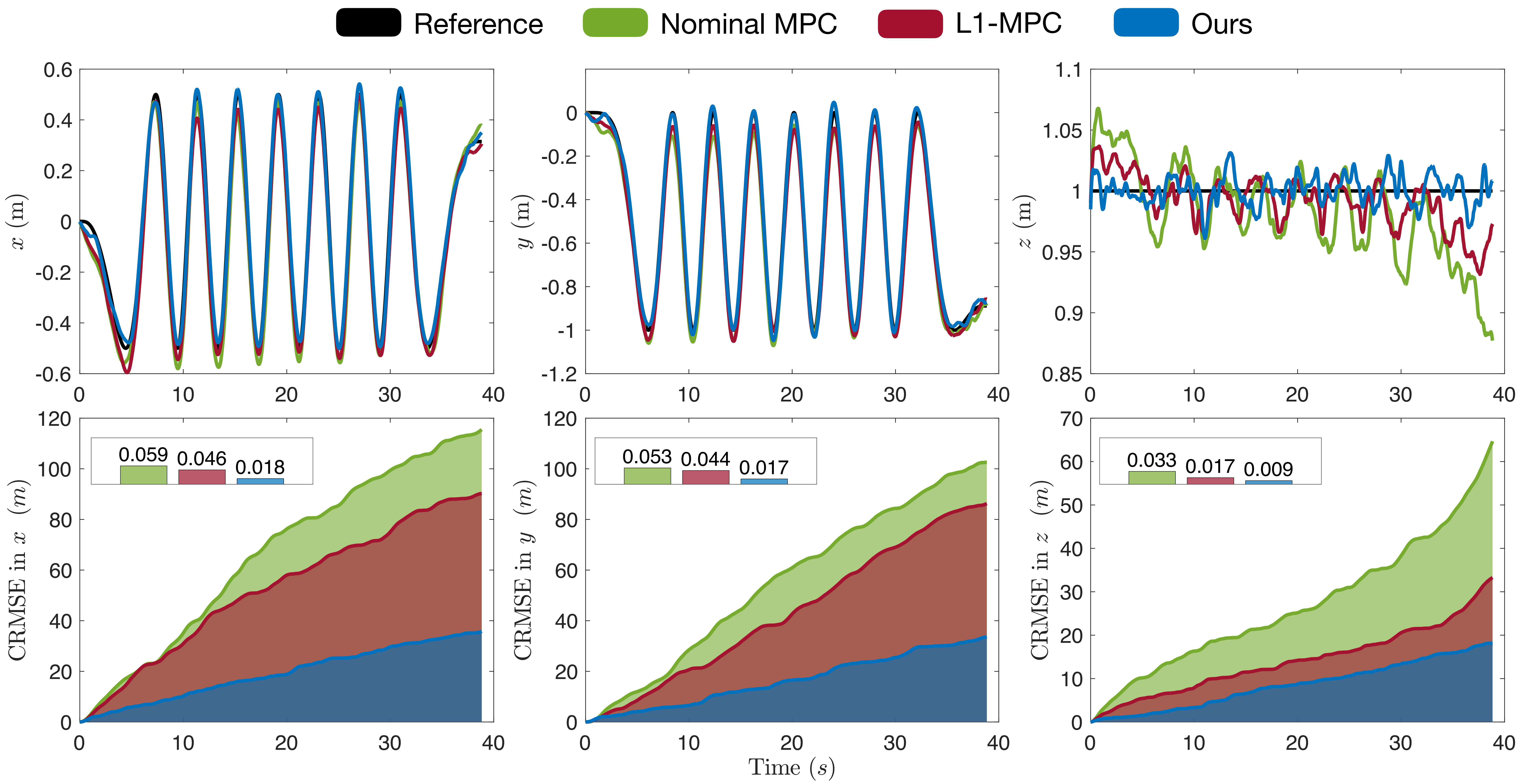
Sample Trajectory with Ground Effect



Results:

- Ours achieves lowest tracking errors
- L1-MPC do not learn the disturbance model and has large tracking error when the disturbances change abruptly
- Nominal MPC has poor *z*-direction tracking error due to ground effect and battery's voltage drop

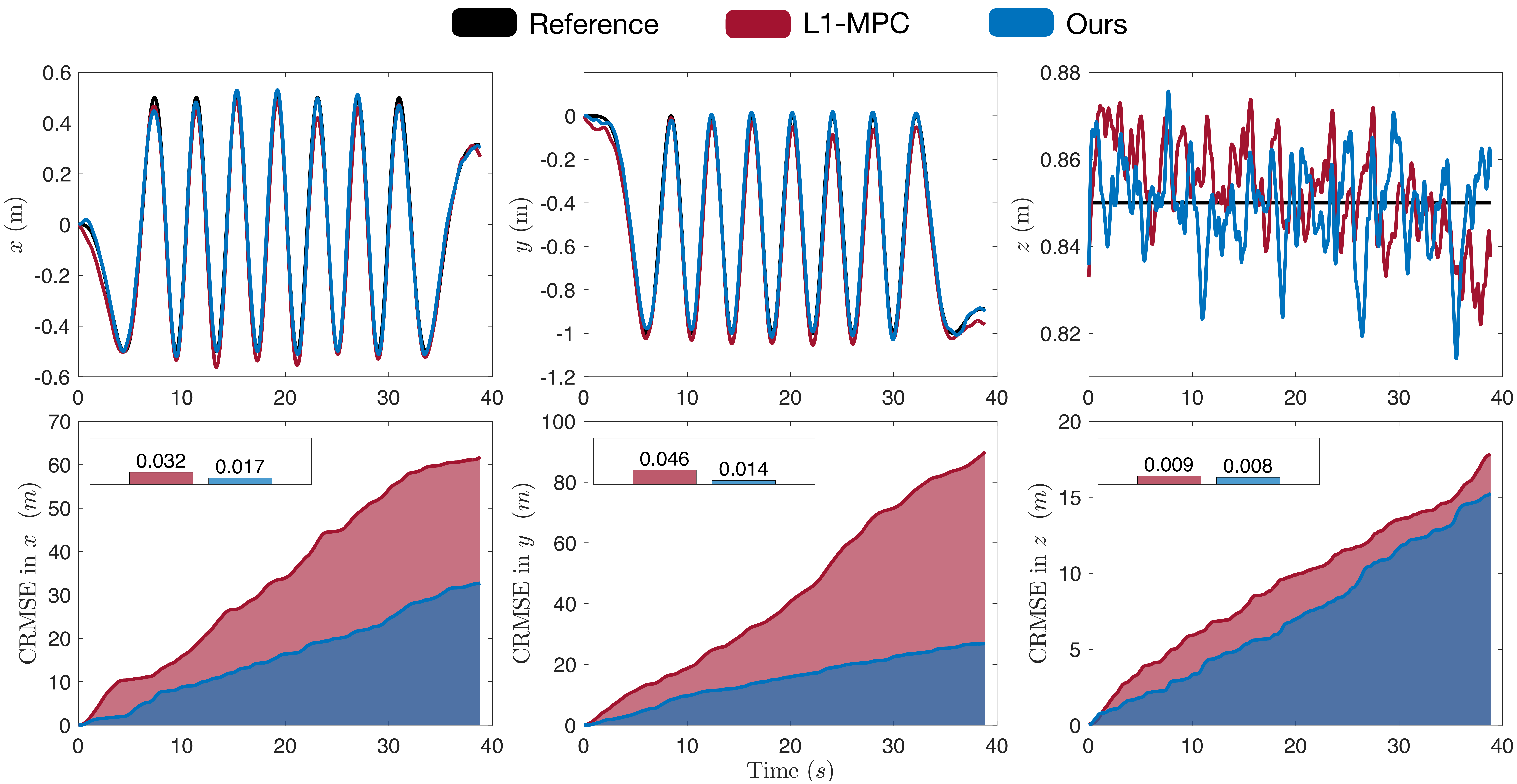
Sample Trajectory with Ground Effect



Results:

- Ours achieves lowest tracking errors
- Nominal MPC has poor z-direction tracking error due to wind and battery's voltage drop

Sample Trajectory with Ground Effect



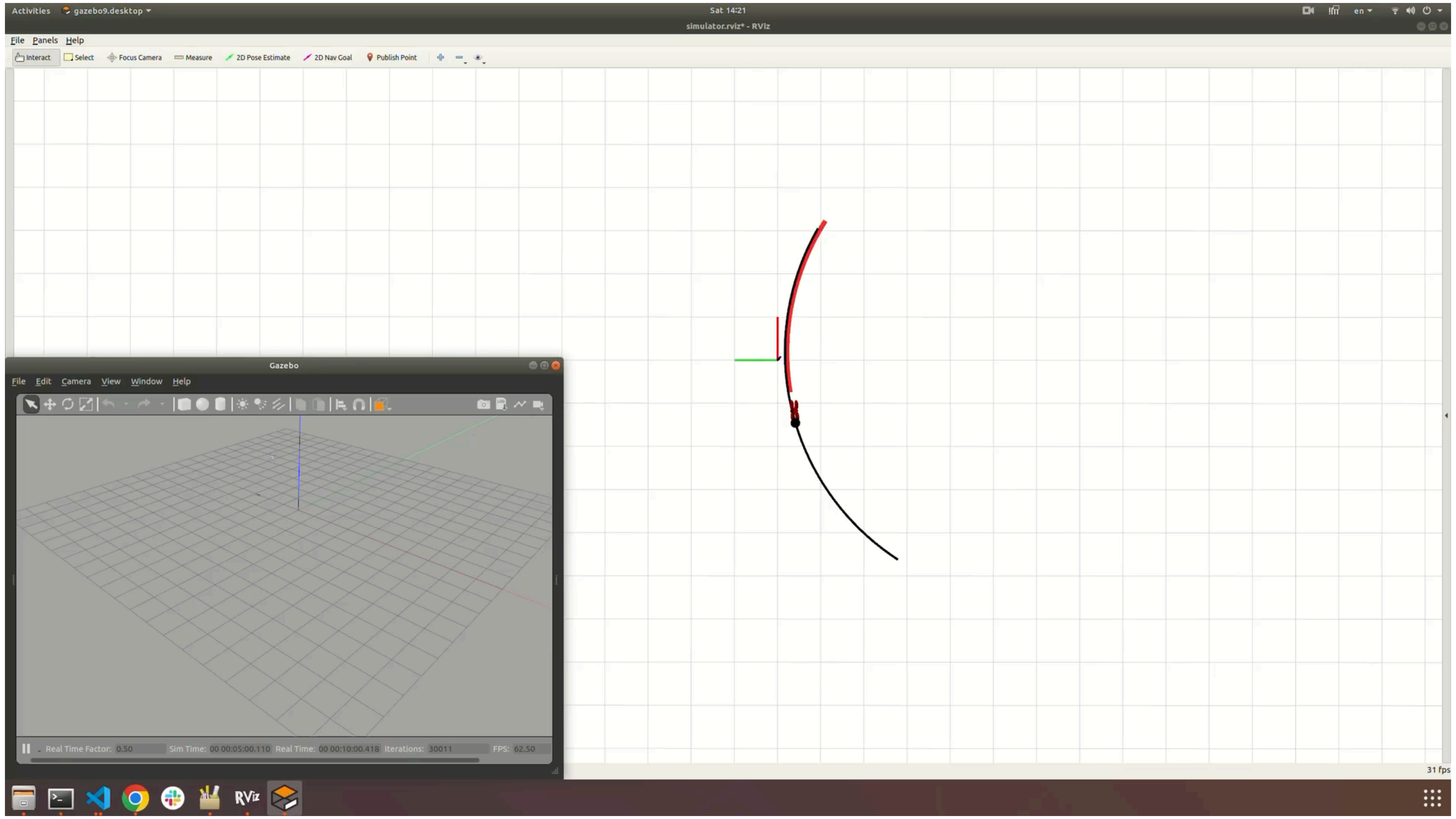
Results:

- Ours achieves lowest tracking errors
- Nominal MPC crashes under ground effect & wind disturbances

Summary and Extensions

Online control algorithm for partially unknown control-affine systems with:

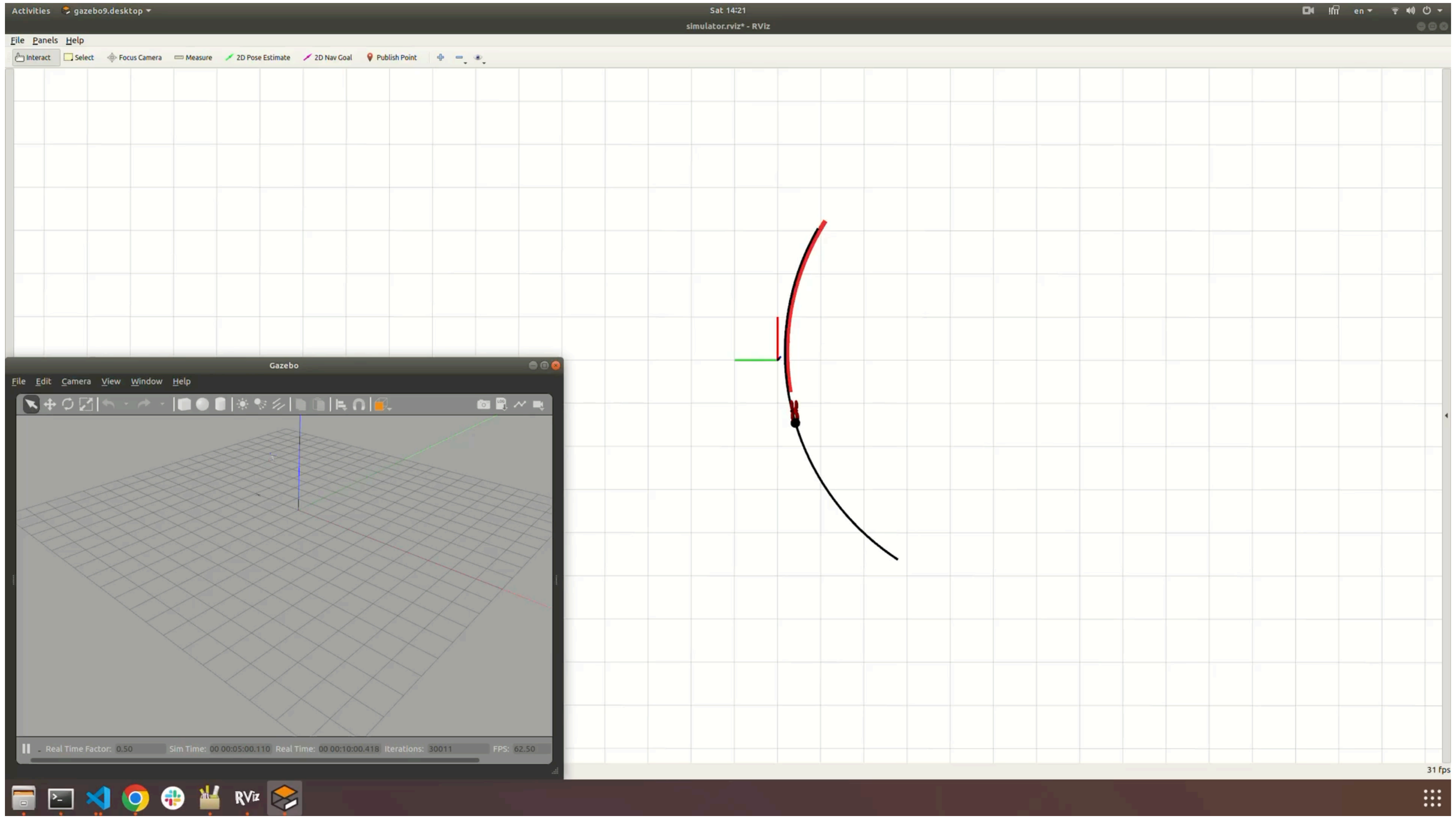
- simultaneous system identification and model predictive control
- no-dynamic-regret performance guarantees



Summary and Extensions

Online control algorithm for partially unknown control-affine systems with:

- simultaneous system identification and model predictive control
- no-dynamic-regret performance guarantees



Extensions:

- active learning
- optimality of the regret bounds