

# Robust Control using Control Lyapunov Function and Hamilton-Jacobi Reachability

Chun-Ming Yang, Pranav A. Bhounsule

Department of Mechanical and Industrial Engineering,  
University of Illinois at Chicago, 842 W Taylor St, Chicago, IL 60607,  
USA. (e-mail: jyang241@uic.edu; pranav@uic.edu)

**Abstract:** The paper presents a robust control technique that combines the Control Lyapunov function and Hamilton-Jacobi Reachability to compute a controller and its Region of Attraction (ROA). The Control Lyapunov function uses a linear system model with an assumed additive uncertainty to calculate a control gain and the level sets of the ROA as a function of the worst-case uncertainty. Next, Hamilton-Jacobi reachability uses the nonlinear model with the modeled uncertainty, which need not be additive, to compute the backward reachable set (BRS). Finally, by juxtaposing the level sets of the ROA with BRS, we can calculate the worst-case additive disturbance and the ROA of the nonlinear model. We illustrate our approach on a 2D quadcopter tracking a trajectory in the presence of disturbances and a 2D quadruped achieving height and velocity regulation in the presence of added mass.

**Keywords:** Control Lyapunov Function, Hamilton-Jacobi Reachability, Model Uncertainty, Quadcopter, Quadruped

## 1. INTRODUCTION

Model predictive control (MPC), an online optimal control technique, has received increased adaptation because of its ability to quickly adapt to the situation in real-time (Di Carlo et al., 2018; Merabti et al., 2015). However, MPC may perform poorly when there is model uncertainty. Hence there is a need to investigate techniques to enable robustification of MPC controllers.

Robust control offers the best possible control for the worst-case uncertainty. The H infinity method is a robust control approach for linear or linearized systems but requires a reasonably accurate model (Babar et al., 2013). On the other hand, adaptive control uses experimental data to either update the model or the controller (Sombolstan and Nguyen, 2023). Learning approaches such as Reinforcement learning can provide robustness using dynamic randomization. Here the controller is a neural network tuned by varying the model and environmental parameters in the simulator, thus making the controller robust to uncertainty (Tan et al., 2018). While robust and adaptive control offers stability guarantees, learning methods do not.

The safety of controllers is often guaranteed by computing the region of attraction (ROA). The sum-of-squares method uses a polynomial approximation of the system dynamics and convex optimization to estimate the ROA (Tedrake et al., 2009). The control-Lyapunov and control-barrier function (CLF-CBF) approaches do not compute the ROA. Still, they guarantee the system would not violate safety constraints through the barrier function and ensure reachability to the goal using the Lyapunov function (Ames et al., 2016). More recently, CLF-CBF has been formulated as constraints in the MPC optimization to

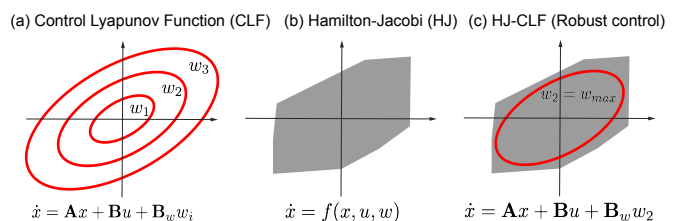


Fig. 1. (a) Control Lyapunov function (CLF) uses the linear dynamics to compute the controller and the level set for the Region of Attraction (ROA) for a given maximum disturbance  $w_i$  (red ellipses); (b) Hamilton-Jacobi (HJ) Reachability uses the nonlinear dynamics with the disturbance  $w$  to compute the Backward Reachable set (gray region); (c) Superimposing the BRS with level sets enables computation of the worst case disturbance (here  $w_2 = w_{max}$ ) and the safe set (red ellipse)

enable optimal control under uncertainty. (Minniti et al., 2021). The Hamilton-Jacobi (HJ) reachability analysis is a tool to compute the target sets the system can reach from a given state under constraints on the control and disturbances. It has been used to calculate the safety regions for quadcopter (Aswani et al., 2013) and path planning of vehicles (Chen et al., 2018).

We introduce Robust Model Predictive Control as a planning method with safety guarantees for bounded model uncertainty. We assume that a nominal MPC controller works for the uncertainty-free nominal system. We linearize the system, and use a Control Lyapunov Function approach to compute the feedback control gain and the level sets for the ROA for different values of the additive disturbances (see Fig. 1 (a)). Note that the additive disturbance in the linearized model is an assumption as the actual model

\* The work was supported by NSF grant 2128568

may not have an additive disturbance. We use the nonlinear model with the Hamilton Jacobi (HJ) reachability to compute the safe set (see Fig. 1 (b)). Finally, by superimposing the level sets on the safe set, we can compute the worst case additive disturbance for the linear model as well as the ROA (see Fig. 1 (c)). Our CLF method is similar to (Yu et al., 2010), but the novelty lies in using HJ to estimate the ROA. The technique is demonstrated on a 2D quadcopter tracking a trajectory under disturbance and a 2D quadruped regulating the height in the presence of added weight and tracking a horizontal speed while pushing a box.

## 2. METHODS

### 2.1 Model Predictive Control (MPC)

Model predictive control (MPC) is an online optimization technique. It uses the current estimate of the state and a model of the plant to estimate the state over a time horizon and a projected cost. The minimization of the cost gives the control over the time horizon. However, only the first computed control is applied, and the process is repeated for the next time step and so on.

The MPC problem at time  $t$  and with horizon length  $k$  can be written as follows

$$\min_{x,u} \sum_{i=0}^{k-1} \|x(t+i) - x_{ref}(t)\|_{Q_{MPC}} + \|u(t+i)\|_{R_{MPC}} \quad (1)$$

$$\text{s.t. } \dot{x}_{t+i} = f(x_{t+i}, u_{t+i}), \quad (2)$$

$$x_{t+i+1} = x_{t+i} + f(x_{t+i}, u_{t+i})\Delta t_{MPC}, \quad (3)$$

$$u_i \in \mathbb{U}, x_{t+i} \in \mathbb{X}, i = 0, 1 \dots k-1 \quad (4)$$

where  $x \in \mathbb{R}^n$  is the system state,  $u \in \mathbb{R}^m$  is the control input,  $Q_{MPC} \in \mathbb{R}^{n \times n}$  and  $R_{MPC} \in \mathbb{R}^{m \times m}$  are user-chosen diagonal positive weight matrix, scalar  $\Delta t_{MPC}$  is the time step, and  $\mathbb{U}, \mathbb{X}$  are feasible sets for the control and state.

### 2.2 Hamilton-Jacobi Reachability Analysis (HJ)

Hamilton-Jacobi Reachability Analysis (HJ) provides formal guarantees for nonlinear control systems' performance and safety properties. Here HJ is used to compute the backward reachable set (BRS). BRS  $\mathcal{R}(t)$  is the set of states  $x \in \mathbb{R}^n$  from which the system can be driven into the target set of states  $\mathcal{T}$  in a predefined time.

Consider the dynamical system

$$\dot{x}(t) = f(x(t), u(t), w(t)) \quad (5)$$

with state  $x \in \mathbb{R}^n$ , time  $t$ , control  $u$ , disturbance  $w$ , and dynamics  $f : \mathbb{R}^n \times \mathcal{U} \times \mathcal{W} \rightarrow \mathbb{R}^n$  is assumed to be continuous, bounded, and Lipschitz continuous with respect to all inputs. The disturbance is bounded as given below:

$$\mathbb{W} = \{w(t) \in \mathbb{R}^p \mid \|w(t)\|_\infty \leq w_{\max}\} \quad (6)$$

Let the solution to the Eqn. 5 be  $\chi(t; x, t_0, u(\cdot), w(\cdot)) : [t_0, 0] \rightarrow \mathbb{R}^n$ . The initial condition is  $\chi(t_0; x, t_0, u(\cdot), w(\cdot)) = x_0$ .

We define the target set of states as  $\mathcal{T}$ . The BRS is the set of states  $\mathcal{R}(t)$  such that we can compute a control

$u$  that will drive the  $\mathcal{R}(t)$  to the  $\mathcal{T}$  for the worst case disturbance  $w_{\max}$  exactly in time  $t_0$ . The BRS is defined mathematically as follows

$$\mathcal{R}(t) = \{x : \forall u(\cdot) \in \mathbb{U}, \exists w(\cdot) \in \mathbb{W}, \exists t \in [t_0, 0], \dots \\ \chi(t; x, t_0, u(\cdot), w(\cdot)) \in \mathcal{T}\} \quad (7)$$

In the HJ formulation, the target set is a sublevel set of a function  $l(x)$ , where  $x \in \mathcal{T} \Leftrightarrow l(x) \leq 0$ . The BRS in HJ reachability becomes the sublevel set of the value function  $V(x, t)$  defined as follows

$$V(x, t) := \min_{w(\cdot)} \max_{u(\cdot)} l(\chi(t_0; x, t_0, u(\cdot), w(\cdot))) \quad (8)$$

The value function  $V(x, t)$  may be obtained by solving the HJ partial differential equation:

$$\frac{\partial V}{\partial t} + H^*(x, \nabla V(x(t), t), t) = 0 \\ V(x, 0) = l(x), t \in [t_0, 0] \\ H^* = \min_{w(\cdot)} \max_{u(\cdot)} \nabla V(x, t)^T f(x, u) \quad (9)$$

### 2.3 Control Lyapunov Function (CLF)

The MPC control presented earlier assumes that there is no external disturbance. By design, MPC control can handle small disturbances but can fail in the presence of large disturbances. Here we present a Control Lyapunov Function to robustify the controller.

Consider the nominal system obtained by setting  $w(t) = 0$  in Eqn. 5

$$\dot{\bar{x}}(t) = f(\bar{x}(t), \bar{u}(t), 0) \quad (10)$$

By defining the error  $e = x - \bar{x}$  between the nominal and actual system, the error dynamics can be described as:

$$\dot{e} = f(x(t), u(t), w(t)) - f(\bar{x}(t), \bar{u}(t), 0) \quad (11)$$

In order to achieve,  $|e| \rightarrow 0$ , we propose a robust controller of the form:

$$u = \bar{u}_{MPC} + K(x - \bar{x}) \quad (12)$$

where  $\bar{u}_{MPC}$  is the feed-forward control from the MPC (see Sec. 2.1),  $x$  is the measured state,  $\bar{x}$  is the reference trajectory, and  $K(\cdot) := \mathbb{R}^n \rightarrow \mathbb{R}^m$  is the ancillary feedback control law derived from *Lemma 2*. It can guarantee all solutions of (11) are decreasing towards zero and are inside an invariant set defined by *Lemma 1*.

**Lemma 1:** Suppose there exists  $E(x(t)) > 0$ ,  $\lambda > 0$  and  $\mu > 0$  such that

$$\frac{d}{dt} E(x(t)) + \lambda E(x(t)) - \mu^T w(t)w(t) < 0, \quad (13)$$

and if  $\Omega(x)$  is the invariant set, then the system trajectory starting from  $x(t_0) \in \Omega(x)$  will remain in  $\Omega(x)$ , where

$$\Omega(x) := \left\{ x \mid E(x(t)) < \frac{\mu w_{\max}^2}{\lambda} \right\} \quad (14)$$

*Proof:* See (Yu et al., 2010).  $\square$

*Remark 1:* The size of the invariant set  $\Omega(x)$  may be tuned by adjusting  $\lambda$  and  $\mu$  to guarantee  $\Omega(x) \in \mathbb{X}$ , that is, the invariant set is inside the feasible set of states.

*Remark 2:* It is usually hard to model the uncertainty and find the disturbance bound (Cockburn and Morton, 1997). In this paper, instead of directly modeling uncertainty, we

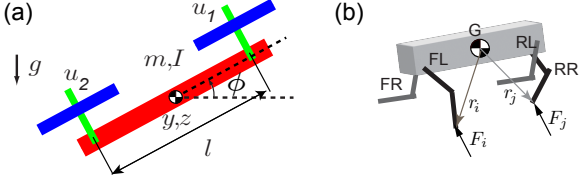


Fig. 2. (a) 2D quadcopter (b) 2D quadruped

utilize the Hamilton-Jacobi analysis (HJ) to determine the bound on the disturbance,  $\omega_{max}$ .

**Lemma 2:** Suppose there exist parameter scalars  $\lambda > 0$ ,  $\mu > 0$  and matrix variables  $0 < Y \in \mathbb{R}^{n \times n}$ ,  $L \in \mathbb{R}^{m \times n}$ , the cost function  $J = \int_0^\infty x^T Q x + u^T R u$  can be minimized by the following optimization:

$$\max_{Y, L} \text{tr}(Y), \quad (15)$$

$$\text{s.t.} \quad \begin{bmatrix} R_{11} & Y^T & L^T & B_w \\ Y & -Q^{-1} & 0 & 0 \\ L & 0 & -R^{-1} & 0 \\ B_w^T & 0 & 0 & -\mu I \end{bmatrix} \prec 0, \quad (16)$$

$$R_{11} = (AY + BL)^T + (AY + BL) + \lambda Y, \quad (17)$$

where  $Q \in \mathbb{R}^{n \times n}$  and  $R \in \mathbb{R}^{m \times m}$  are diagonal positive weight matrix. Then with  $u = Kx$ ,  $E(x) = x^T P x$ , where  $Y = P^{-1}$  and  $L = KY$ , Lemma 1 is satisfied for the linearized form,  $\dot{x} = Ax + Bu + B_w w$ , of the nonlinear system  $\dot{x} = f(x, u, w)$ .

*Proof:* Given in appendix.  $\square$

#### 2.4 Example 1: 2D Quadcopter

*Nonlinear Model:* The 2D Quadcopter is shown in Fig. 2 (a), and the equations derived from using the Newton-Euler method are given below

$$\dot{x} = f(x, u, w) = \begin{bmatrix} \dot{y} \\ \dot{z} \\ \dot{\phi} \\ -m^{-1}u_s \sin \phi + w_1 \\ m^{-1}u_s \cos \phi - g + w_2 \\ 0.5I_{xx}^{-1}l u_d \end{bmatrix} \quad (18)$$

where  $y, z, \phi$  are horizontal, vertical position and orientation,  $\dot{y}, \dot{z}, \dot{\phi}$  are horizontal, vertical velocity and angular velocity,  $m, l, I_{xx}, g$  are mass, length, the moment of inertia and gravity,  $u_s = u_1 + u_2$ ,  $u_d = u_1 - u_2$  are the sum of the thrust forces and the difference in the thrust forces. We assume there is an additive disturbance  $w_i \in \mathbb{R}$ , where  $i = 1, 2$ , and is bounded  $\|w_i\|_\infty \leq w_{max}$ .

*Linearized Model:* The system may be linearized by assuming  $\sin \phi \approx \phi$  and  $\cos \phi \approx 1$ ,  $u_s = mg + \delta u_s$ ,  $u_d = \delta u_d$ , and note that  $\delta u_s \sin \phi \approx \delta u_s \phi \approx 0$

$$\begin{aligned} \dot{x} &= Ax + Bu + B_w w + G \\ \dot{\bar{x}} &= A\bar{x} + B\bar{u} + G \end{aligned} \quad (19)$$

where  $A$  can be populated as follows (assuming index starts from 1):  $A = \text{zeros}(6, 6)$ ,  $A(1, 4) = A(2, 5) = A(3, 6) = 1$ ,  $A(4, 3) = -g$ ;  $B = \text{zeros}(6, 2)$ ,  $B(5, 1) = 1/m$  and  $B(6, 2) = 0.5I_{xx}^{-1}l$ ;  $B_w = \text{zeros}(6, 2)$ ,  $B(4, 1) = B(5, 2) = 1$ , and  $G = \text{zeros}(6, 1)$ ,  $G(5, 1) = -g$ , where the actual state is  $x = [y \ z \ \phi \ \dot{y} \ \dot{z} \ \dot{\phi}]^T$ , control is  $u =$

$[\delta u_s \ \delta u_d]^T$ , the nominal state and control  $\bar{x}$ ,  $\bar{u}$  are similar to  $x$ ,  $u$  respectively. The disturbance is  $w = [w_1 \ w_2]^T$ .

*Robust control:* The proposed controller is of the form 
$$u = \bar{u} + K(x - \bar{x}) \quad (20)$$

where the ancillary feedback control gain is  $K \in \mathbb{R}^{2 \times 6}$  for the given  $\lambda > 0$  and  $\mu > 0$  and is computed from Lemma 2. Furthermore, given the bound  $w_{max}$ , the region of attraction can be computed using Lemma 1.

#### 2.5 Example 2: 2D Quadruped

*Nonlinear Model:* The 2D quadruped is shown in Fig. 2 (b), and the equations are derived from the Newton-Euler method. We have assumed that all mass is concentrated at the torso, and the legs are massless.

$$\dot{x} = f(x, u, w) = \begin{bmatrix} \dot{y} \\ \dot{z} \\ \dot{\phi} \\ \bar{m}^{-1}(F_i + F_j) - G \\ I_{xx}^{-1}(r_i \times F_i + r_j \times F_j) \end{bmatrix} \quad (21)$$

where the positions of the center of mass in the global frame are  $y$  and  $z$ , the angular position of the body with respect to the horizontal direction is  $\phi$ , the ground reaction forces on the diagonal legs are  $F_i, F_j \in \mathbb{R}^2$ , where  $i$  and  $j$  are diagonal legs, the position vectors from the respective feet to the center of mass are  $r_i, r_j \in \mathbb{R}^2$ , the constant vector is  $G = \{\mu g, g\}$  where gravity is  $g$  and coefficient of static friction is  $\mu$ , inertia is  $I_{xx} \in \mathbb{R}$ , and the uncertainty is in the mass,  $\bar{m} = (m + \Delta m) \in \mathbb{R}$ , where the actual mass is  $m$ . The uncertainty in mass is  $\Delta m$ . We assume that the quadruped moves in a trot gait where diagonal legs contact the ground. The step time is considered to be constant.

*Linearized Model:* We assume that the linearized equation in the  $y$ - and  $z$ -direction (lines 1, 2, 4, 5 in Eqn. 21) can be written as

$$\begin{aligned} \dot{x}_i &= A_i x_i + B_i u_i + B_{wi} w_i + G_i \\ \dot{\bar{x}}_i &= A_i \bar{x}_i + B_i \bar{u} + G_i \end{aligned} \quad (22)$$

where  $x_i \in \mathbb{R}^2$ ,  $i = 1, 2$  such that  $x_1 = [y, \dot{y}]$  and  $x_2 = [z, \dot{z}]$   $u_1 = \{\delta F x_f, \delta F x_r\}$  and  $u_2 = \{\delta F y_f, \delta F y_r\}$  where  $F x, F y$  are the control forces on the front and rear leg and subscript  $f r$  refer to the diagonal front and rear feet which are in contact with the ground,  $A_i, B_i, B_{wi}$  can be populated as follows (assuming index starts from 1):  $A_i = \text{zeros}(2, 2)$ ,  $A_i(1, 2) = 1$ ;  $B = \text{zeros}(2, 2)$ ,  $B_i(2, 1) = B_i(2, 2) = 1/m$ ;  $B_{wi} = \text{zeros}(2, 1)$ ,  $B_{wi}(2, 1) = 1$ ;  $G_i = \text{zeros}(2, 1)$ ,  $G_1(2, 1) = \mu$ ,  $G_2(2, 1) = -g$ . We model the uncertainty in mass (see  $\Delta m$  in Eqn 22) as an additive disturbance,  $w_i$ , with a bound given by  $\mathbb{W} = \{w_i \in \mathbb{R} \mid \|w_i\|_\infty \leq w_{max}\}$ . Here  $w_{max}$  is unknown, but we will use HJ to estimate it as shown in HJ section below.

*Robust Control:* The control is the same from (Bhounsule and Yang, 2023), but the stance phase control is modified for robustness as given below.

$$u_i = \bar{u}_i + K(x_i - \bar{x}_i) \quad (23)$$

where the ancillary feedback control gain is  $K \in \mathbb{R}^{2 \times 2}$  for the given  $\lambda > 0$  and  $\mu > 0$  and is computed from Lemma

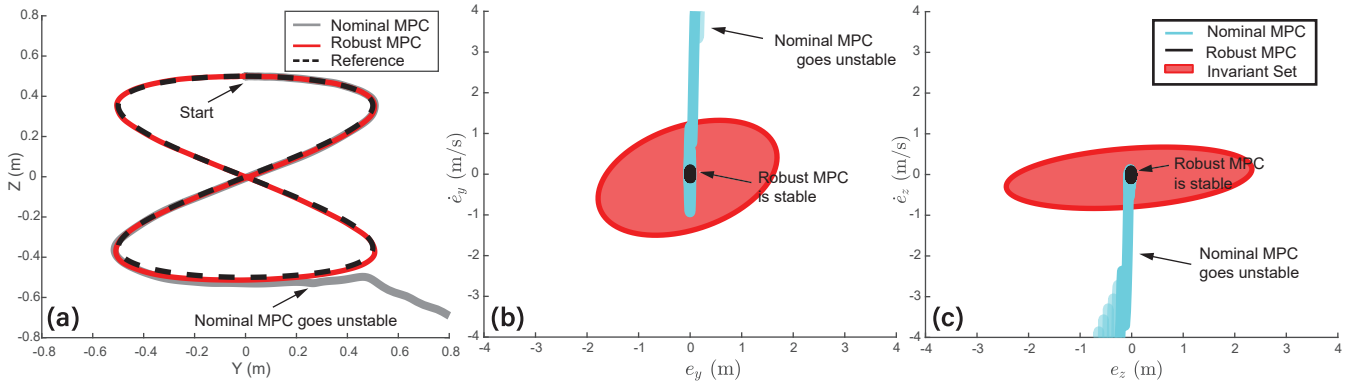


Fig. 3. Bicopter is tracking the figure 8 curve in the presence of bounded disturbance. (a) Figure 8 tracking in the y-z plane. (b) Horizontal tracking errors with the invariant set or the Region of attraction (ROA). (c) Vertical tracking errors with invariant set.

2. Furthermore, given the bound  $w_{\max}$ , the region of attraction may be computed using *Lemma 1*. However, the bound  $w_{\max}$  is unknown, and we will use HJ to estimate it, as shown in the HJ section below.

*Hamilton-Jacobi Reachability:* The uncertainty in the model arises due to an unknown mass  $\Delta m$  placed on top of the robot. Thus, the mass of the robot,  $\bar{m} = m + \Delta m$ , is uncertain. From Eqn 21, we observe that this is a multiplicative uncertainty that cannot be isolated symbolically as an additive term,  $w$ , as we have assumed in the linearized Eqn. 22. We use HJ to identify a bound on  $w$ ,  $w_{\max}$ , as follows

- (1) The HJ uses the nonlinear dynamics Eqn. 21 and the uncertainty (here the added mass  $\Delta m$ ) to compute the Backward Reachable Set (BRS)  $\mathcal{R}$  for a given target set  $\mathcal{T}$ . The safe set,  $\mathcal{S}$ , is the set of all states that are guaranteed to be invariant and are given by  $\mathcal{S} = \mathcal{R} \cap \mathcal{T}$ .
- (2) The CLF using *Lemma 2* to compute the gain  $K$  and then *Lemma 1* to compute the region of attraction for a given  $w_{\max}$ .
- (3) Finally, we combine HJ and CLF as follows. We compute the value of the worst-case uncertainty,  $w_{\max}$ , using a line search method such that the region of attraction computed using the CLF lies within the safe set computed using HJ.

### 3. RESULTS AND DISCUSSION

*Overview:* We present results on a quadcopter tracking a lemniscate curve with disturbances, and a quadruped maintaining a set height while carrying an unknown mass and following a reference velocity as it pushes an unknown mass. We compare the results of model predictive control and robust model predictive control that combines Control Lyapunov Function and Hamilton-Jacobi Reachability.

For CLF, we used YALMIP (Lofberg, 2004), a toolbox for solving Linear Matrix Inequality and other features in MATLAB, to compute the feedback gain  $K$  and positive definite matrix,  $P$ . For HJ, we used *helperOC* (Bansal et al., 2017), optimal control toolbox for Hamilton-Jacobi Reachability Analysis, to compute the maximum disturbance  $w_{\max}$  for the quadruped. We used a custom simu-

lation written in Python for 2D quadcopter and MuJoCo (Todorov et al., 2012) for the 2D quadruped. All computations are on Ubuntu 20.04 on Intel Core i7.

*2D quadcopter:* The model parameters are  $m = 1$  kg,  $l = 0.2$  m,  $I_{xx} = 0.1$  kg·m<sup>2</sup>,  $g = 9.81$  m/s<sup>2</sup>, and the additive uniformly distributed disturbance  $w_i \in \mathbb{W}$ , where  $\mathbb{W} = \{w_i \in \mathbb{R} \mid \|w_i\|_{\infty} \leq 3.5\}$  where  $i = 1, 2$ . For the quadcopter we track the figure 8 which is given by  $x = 0.5 \sin(2\tau)$  and  $y = 0.5 \cos(\tau)$  where  $\tau$  is assumed to be a fifth order polynomial and the constants are solved such that  $\tau(t = 0) = \dot{\tau}(t = 0) = \ddot{\tau}(t = 0) = \dot{\tau}(t = T) = \ddot{\tau}(t = T) = 0$  and  $\tau(t = T) = T$  where the end time is  $T = 5$ .

For the nominal MPC, the time step discretization is  $dt = 0.05$  s, and the prediction horizon is two time steps. The cost function matrices are  $Q_{MPC} = \text{diag}([100, 10, 1e9, 1e5, 1e14, 1e4])$  and  $R_{MPC} = \text{diag}([1e6, 1e6])$ .

For the Robust MPC we used  $R = \text{diag}([1e - 2, 1e - 4])$ ,  $Q = \text{diag}([1e - 1, 1, 1, 1, 1, 1e - 2])$ ,  $\mu = 0.1$ , and  $\lambda = 0.5$ . Using *Lemma 1* and *Lemma2*, we computed the gain  $K$  and the positive definite matrix  $P$ .

Figure 3 (a) shows the quadcopter’s ability to track figure 8 using nominal MPC and robust MPC. It can be seen that the nominal MPC fails to track about halfway while the robust MPC can track successfully. Figure 3 (b) and (c) shows the tracking error in the y- and z-direction.

*2D quadruped:* The model parameters  $m = 12.454$  kg,  $I_{xx} = 0.0565$  kg·m<sup>2</sup>, and  $g = 9.81$  m/s<sup>2</sup>, where the length of the thigh and shank are  $l = 0.2$  m, considering an additional mass disturbance  $\Delta m = 5$  kg loaded on the robot’s torso and in front of the robot to be the uncertainty, interfering with height and velocity regulation. The nominal robot height is 0.32 m, and the forward velocity is 0.45 m/s.

For the nominal MPC, the time step discretization is  $dt = 0.05$  s, and the MPC horizon is 2 time steps. The cost function matrices are  $Q_{MPC} = \text{diag}([1e5, 1e3, 1e7, 1e2, 1e1, 1e2])$  and  $R_{MPC} = 0$ .

For the control in the horizontal direction (y-direction)  $R = \text{diag}([1])$ ,  $Q = \text{diag}([500, 10])$ ,  $\mu = 200$ , and  $\lambda = 0.3$ .

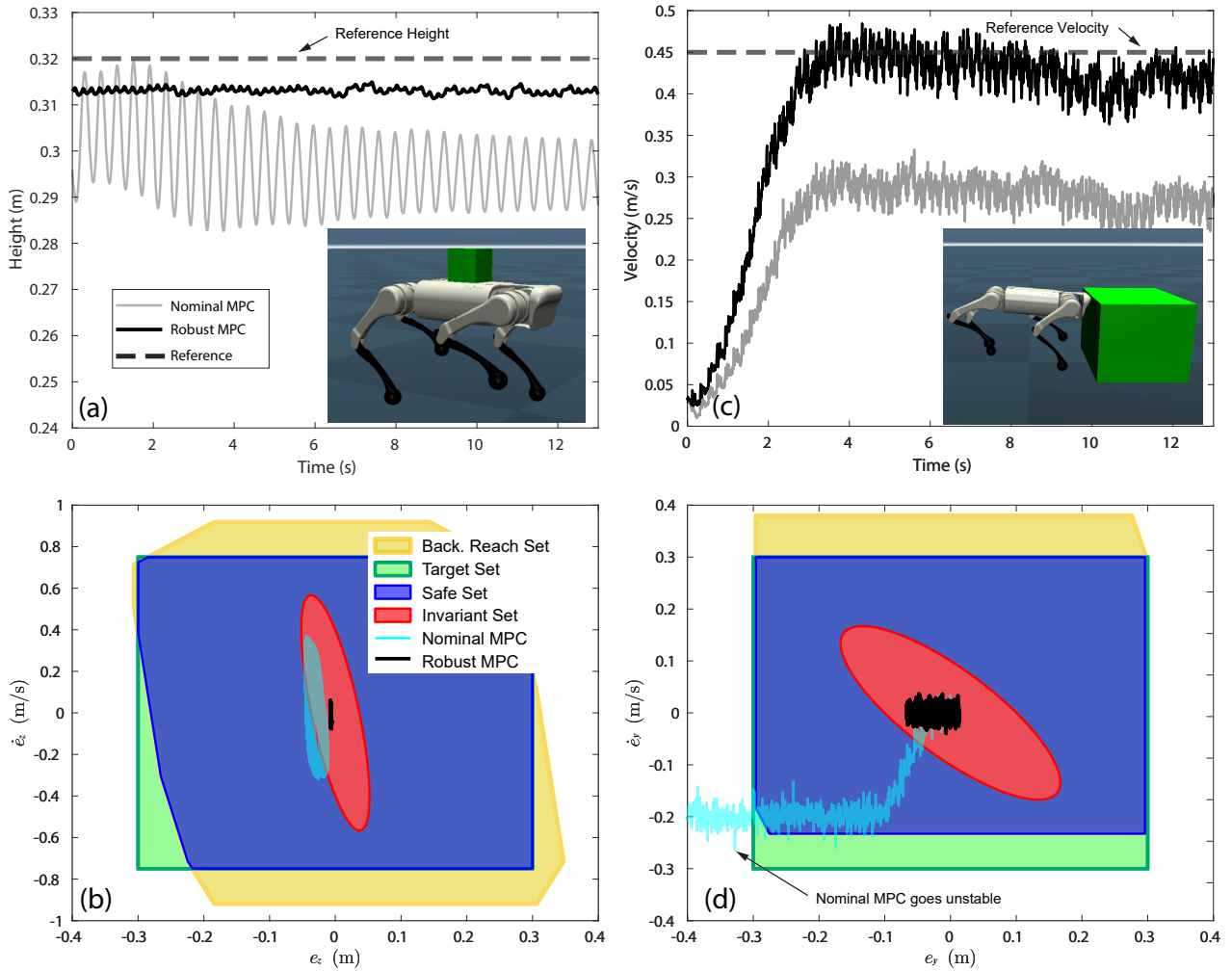


Fig. 4. For the 2D quadruped, two separate experiments are conducted. (a) simulation snapshot of quadruped while loading an unknown 5 kg object, along with the vertical height tracking result. (b) vertical height tracking errors with invariant set (c) simulation snapshot of quadruped while pushing an unknown 5 kg object along with the forward velocity tracking result (d) forward velocity tracking errors with invariant set

Using Lemma 1 and Lemma 2 we computed the gain  $K$  and the positive definite matrix  $P$  For the control in the vertical direction ( $z$ -direction),  $R = \text{diag}([0.01])$ ,  $Q = \text{diag}([1000, 1])$ ,  $\mu = 90$ , and  $\lambda = 0.8$ . Using Lemma 1 and Lemma 2 we computed the gain  $K$  and the positive definite matrix  $P$

Figure 4 (a) and (b) shows reference height tracking in a trot gait with added weight. The reference height is the dashed gray line. The robust MPC (black line) shows substantially lower tracking error than the nominal MPC (gray line). Figure 4 (c) and (d) show reference velocity tracking when the robot is pushing a weight. The reference velocity is shown as the dashed gray line. The robust MPC (black line) shows a substantially lower tracking error than the nominal MPC (gray line). The latter eventually goes beyond the safe region while the robust MPC stays within the Invariant set.

This paper shows that CLF can be combined with HJ reachability to create a robust controller with safety guarantees for an existing nominal controller (here MPC).

The method capitalizes on convex optimization with a quadratic cost and a linear model to compute the ROAs as a function of the additive disturbances. Then HJ is used to calculate the correct approximation for the additive disturbance to select the correct ROA. The results are demonstrated on a 2D quadcopter and 2D quadruped in simulation.

The method has its limitations. The process is conservative and requires extensive trial and error to compute the gains and invariant set for the CLF. The technique is strictly offline and relies on understanding the nonlinear model and the model uncertainty. The method works well for fully actuated or under-actuated systems without strong coupling (e.g., for 2D quadruped, we assumed the coupling between pitching motion and translation was minimal).

## REFERENCES

- Ames, A.D., Xu, X., Grizzle, J.W., and Tabuada, P. (2016). Control barrier function based quadratic pro-

- grams for safety critical systems. *IEEE Transactions on Automatic Control*, 62(8), 3861–3876.
- Aswani, A., Gonzalez, H., Sastry, S.S., and Tomlin, C. (2013). Provably safe and robust learning-based model predictive control. *Automatica*, 49(5), 1216–1226.
- Babar, M., Ali, S., Shah, M., Samar, R., Bhatti, A., and Afzal, W. (2013). Robust control of uavs using h control paradigm. In *2013 IEEE 9th International Conference on Emerging Technologies (ICET)*, 1–5. IEEE.
- Bansal, S., Chen, M., Herbert, S., and Tomlin, C.J. (2017). Hamilton-jacobi reachability: A brief overview and recent advances. 2242–2253.
- Bhounsule, P.A. and Yang, C.M. (2023). A simple controller for omnidirectional trotting of quadrupedal robots: Command following and waypoint tracking. *Robotics*, 12(2), 35.
- Chen, M., Bansal, S., Fisac, J.F., and Tomlin, C.J. (2018). Robust sequential trajectory planning under disturbances and adversarial intruder. *IEEE Transactions on Control Systems Technology*, 27(4), 1566–1582.
- Cockburn, J.C. and Morton, B.G. (1997). Linear fractional representations of uncertain systems. *Automatica*, 33(7), 1263–1271.
- Di Carlo, J., Wensing, P.M., Katz, B., Bledt, G., and Kim, S. (2018). Dynamic locomotion in the mit cheetah 3 through convex model-predictive control. 1–9.
- Lofberg, J. (2004). Yalmip: A toolbox for modeling and optimization in matlab. In *2004 IEEE international conference on robotics and automation (IEEE Cat. No. 04CH37508)*, 284–289. IEEE.
- Merabti, H., Bouchachi, I., and Belarbi, K. (2015). Nonlinear model predictive control of quadcopter. In *2015 16th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, 208–211. doi:10.1109/STA.2015.7505151.
- Minniti, M.V., Grandia, R., Farshidian, F., and Hutter, M. (2021). Adaptive clf-mpc with application to quadrupedal robots. *IEEE Robotics and Automation Letters*, 7(1), 565–572.
- Sombolestan, M. and Nguyen, Q. (2023). Hierarchical adaptive loco-manipulation control for quadrupedal robots. 12156–12162.
- Tan, J., Zhang, T., Coumans, E., Iscen, A., Bai, Y., Hafner, D., Bohez, S., and Vanhoucke, V. (2018). Sim-to-real: Learning agile locomotion for quadrupedal robots. *arXiv preprint arXiv:1804.10332*.
- Tedrake, R. et al. (2009). Lqr-trees: Feedback motion planning on sparse randomized trees. In *Robotics: Science and Systems*, volume 2009.
- Todorov, E., Erez, T., and Tassa, Y. (2012). Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*, 5026–5033. IEEE.
- Yu, S., Böhm, C., Chen, H., and Allgöwer, F. (2010). Robust model predictive control with disturbance invariant sets. 6262–6267.

## Appendix A. PROOF OF LEMMA 2

We prove that *Lemma 2* holds for the system as follow:

$$\dot{x} = Ax + Bu + B_\omega \omega \quad (\text{A.1})$$

$$u = Kx \quad (\text{A.2})$$

Firstly, to achieve equilibrium point tracking, an optimization problem is formulated. This involves using a standard infinite horizon quadratic cost as the performance measure.

$$\min \int_0^\infty x^T Q x + u^T R u \quad (\text{A.3})$$

A controller which minimizes the proposed cost is given by solving the following semidefinite programming problem:

$$\mathcal{P} : \min_{P, K} \text{tr}(P) \quad (\text{A.4})$$

$$s.t. \quad (A + BK)^T P + P(A + BK) \prec -Q - K^T R K \quad (\text{A.5})$$

where positive definite matrix  $P$  is defined by the propose CLF  $E(x) = x^T P x$  and  $Q, R$  are diagonal weight matrices. Assuming the existence of positive scalars  $\mu$  and  $\lambda$ , the original constraint can be rewritten to include additional negative terms  $-PB_\omega(\mu)^{-1}(PB_\omega)^T, -\lambda P$ . This modification preserves the validity of (A.5) the original constraint, leading to the new formulation as follows:

$$(A + BK)^T P + P(A + BK) \prec -Q - K^T R K - PB_\omega(\mu)^{-1}(PB_\omega)^T - \lambda P \quad (\text{A.6})$$

Using Schur complement, the constraint (A.6) can be rewritten as:

$$\bullet := (A + BK)^T P + P(A + BK) + \lambda P \quad (\text{A.7})$$

$$\begin{bmatrix} \bullet & I & K^T & PB_\omega \\ I & (-Q)^{-1} & 0 & 0 \\ K & 0 & (-R_1)^{-1} & 0 \\ B_\omega^T P & 0 & 0 & -\mu \end{bmatrix} \prec 0 \quad (\text{A.8})$$

Performing a congruence transformation on (A.8) with the matrix  $\text{diag}(x, \Xi, \Phi, \omega)$ , the following inequality is obtained:

$$\begin{aligned} & x^T [(A + BK)^T P + P(A + BK) + \lambda P] x + \\ & \Xi^T I x + \Phi^T K x + \omega^T (PB_\omega)^T x + x^T I \Xi - \Xi^T Q^{-1} \Xi + \\ & x^T K^T \Phi - \Phi^T R^{-1} \Phi + x^T PB_\omega \omega - \mu \omega^T \omega < 0 \quad (\text{A.9}) \end{aligned}$$

Assuming  $\Xi, \Phi = 0$ , and considering the CLF  $E(x) = x^T P x$  along with its derivative  $\frac{d}{dt} E(x) = \dot{x}^T P x + x^T P \dot{x}$ , (A.9) can be reformulated as follows:

$$\frac{d}{dt} E(x) + \lambda E(x) - \mu \omega^T \omega < 0 \quad (\text{A.10})$$

Thus, according to *Lemma 1*, by solving semidefinite programming problem  $\mathcal{P}_1$  subject to constraint (A.5), it can be guaranteed that system remains robustly invariant within the set  $\Omega(x) := \left\{ x \mid E(x) < \frac{\mu \omega_{\max}^2}{\lambda} \right\}$ .

By using the standard method in control, performing a congruence transformation with  $Y = P^{-1}$ , introducing  $L = KY$ , and applying a Schur complement, nonconvex problem  $\mathcal{P}$  can be formulated as a convex problem as follow:

$$\max_{Y, L} \text{tr}(Y) \quad (\text{A.11})$$

$$* := (AY + BL)^T + (AY + BL) + \lambda Y \quad (\text{A.12})$$

$$s.t. \quad \begin{bmatrix} * & Y^T & L^T & B_\omega \\ Y & -Q^{-1} & 0 & 0 \\ L & 0 & -R^{-1} & 0 \\ B_\omega^T & 0 & 0 & -\mu \end{bmatrix} \prec 0 \quad (\text{A.13})$$

□