Optimal Robust Control for Constrained Nonlinear Hybrid Systems with Application to Bipedal Locomotion

Quan Nguyen and Koushil Sreenath

Abstract— **Recent work on control Lyapunov functions and control Barrier functions has enabled addressing stability of nonlinear and underactuated hybrid systems while simultaneously enforcing input / state constraints and safety-critical constraints. However, under model uncertainty, these controllers break down and violate constraints. This paper presents a novel method of optimal robust control through quadratic programs that can handle stability, input / state dependent constraints, as well as safety-critical constraints, in the presence of high level of model uncertainty. Under the assumption of bounded uncertainty, the proposed controller strictly guarantees constraints without violating them. We evaluate our proposed control design for achieving dynamic bipedal locomotion that involves orbital stability of an underactuated nonlinear hybrid system subject to (a) torque saturation constraints (input constraints), (b) contact force constraints (state constraints), and (c) precise footstep placements (safety-critical constraints). We present numerical results on RABBIT, a five-link planar bipedal robot, subject to a large unknown load on its torso. Our proposed controller is able to demonstrate walking while strictly enforcing the above constraints with an unknown load of up to 15 Kg (47% of the robot mass.)**

I. Introduction

Designing controllers for nonlinear systems for achieving stability while simultaneously guaranteeing input, state, and safety constraints is challenging. Trying to do this when subject to high levels of model uncertainty is even harder. The proposed controller in this paper offers a solution under the assumption of bounded model uncertainty. We develop a QP-based controller using control Lyapunov functions for stability and control Barrier functions for safety and strictly enforce constraints without violations even under the presence of model uncertainty.

Using Control Lyapunov Functions (CLFs) for designing feedback control and analyzing the stability of the closed loop system for both linear and nonlinear systems is a well established approach [6]. Recently, Control Lyapunov Function (CLF) based controllers for nonlinear and hybrid systems have been introduced, see [2], [3]. Inspired by these approaches, expressing the CLF-based control via Quadratic Programs was introduced in [7], which opens an effective way for dealing with stability and additional input-based constraints at the same time, wherein the control input is solved through a QP pointwise in time.

Q. Nguyen is with the Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, email: quannguyen@cmu.edu.
K. Sreenath is with the Depts. of Mechanical Engineering, The Robotics Institute, and Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213, email: koushils@cmu.edu.
This work is supported in part by NSF grants IIS-1526515, CMMI-1538869, IIS-1464337 and in part by the Google Faculty Research Award.

Furthermore, currently, a novel method of Control Barrier Function incorporated with Control Lyapunov Function based Quadratic Program (CBF-CLF-QPs) was introduced in [1], that can handle state-dependent constraints effectively in real-time. Preliminary experimental validations were carried out on the problem of Adaptive Cruise Control in [11]. The methodology has also been extended to safety constraints on Riemmanian manifolds [18] and dynamic walking of bipedal robots [14], [8].

These approaches offer formal guarantees on stability and safety, however the question of robustness of this approach remains. Robust control has been extensively studied and there are well established methods, such as H_∞-based and linear quadratic Gaussian (LQG) based robust control [10] for linear systems, and input-to-state stability (ISS) [16] and sliding mode control (SMC) [5] methods for nonlinear systems. Robust control of hybrid systems can be achieved using the ISS technique, see [17].

The robustness of the CLF-QP has been recently addressed in [13], however the problem of robustly enforcing constraints was not considered. Preliminary robustness analysis of the CBF-based constraints was carried out recently in [19], wherein formal bounds on the violation of the constraints due to the model uncertainty were presented. However, for safety constraints, it's critical that constraints are enforced strictly without any violations. In this paper, an Optimal Robust Control via Quadratic Programs will offer a novel method to simultaneously handle robust stability, robust input-based constraints, and robust state-dependent constraints under bounded model uncertainty.

In particular, our work builds off our recent results on robust dynamic walking [13] (which offered robust stability but did not handle constraints), and dynamic walking with precise foot placements [14] (which guaranteed safety constraints but could not handle model uncertainty). The main contributions of the paper with respect to prior work are:

- The introduction of a new technique of optimal robust control via quadratic programs that can simultaneously handle stability, input-based constraints and state-dependent constraints under high level of model uncertainty.
- Strict enforcement of input / state-based and safety constraints without violations, even under the presence of large model uncertainty.
- Application to dynamic bipedal walking subject to torque constraints, contact force constraints, and precise footstep placement constraints.

The rest of the paper is organized as follows. Section II revisits Control Barrier Functions and Control Lyapunov Functions based Quadratic Programs (CBF-CLF-QPs). Section III discusses the adverse effects of uncertainty in the dynamics on the CBF-CLF-QP controllers. Section IV presents the proposed optimal robust control using CBF-CLF-QP. Section V presents numerical validation with application to dynamic bipedal walking. Finally, Section VI provides concluding remarks.

II. Control Lyapunov Functions and Control Barrier Function based Quadratic Programs Revisited

A. Model and Input-Output Linearizing Control

Consider a nonlinear control affine hybrid model

$$\mathcal{H}: \begin{cases} \dot{x} = f(x) + g(x)u, & x \notin S \\ x^+ = \Delta(x^-), & x \in S \end{cases} \qquad (1)$$
$$y = y(x),$$

where $x \in \mathbb{R}^n$ is the system state, $u \in \mathbb{R}^m$ is the control input, S is the switching surface of the hybrid system, and $y \in \mathbb{R}^m$ is a set of relative-degree two outputs. Thus,

$$\ddot{y} = L_f^2 y(x) + L_g L_f y(x) u. \qquad (2)$$

If the decoupling matrix $L_g L_f y(x)$ is invertible, then,

$$u(x, \mu) = u^*(x) + (L_g L_f y(x))^{-1}\mu, \qquad (3)$$

with

$$u^*(x) = -(L_g L_f y(x))^{-1} L_f^2 y(x), \qquad (4)$$

input-output linearizes the system. The dynamics of the system (1) can then be described in terms of dynamics of the transverse variables, $\eta = \begin{bmatrix} y(x) & \dot{y}(x) \end{bmatrix}^T \in \mathbb{R}^{2m}$, and the coordinates $z \in \mathcal{Z} = \{\mathbb{R}^n \mid \eta(x) \equiv 0\}$. The input-output linearized hybrid system then is,

$$\mathcal{H}_{I/O}: \begin{cases} \dot{\eta} = \bar{f}(\eta) + \bar{g}(\eta)\mu, & (\eta, z) \notin S \\ \dot{z} = p(\eta, z), & \\ (\eta^+, z^+) = \Delta(\eta^-, z^-), & (\eta, z) \in S \end{cases} \qquad (5)$$
$$y = y(\eta),$$

where z represents uncontrolled states [3], and

$$\begin{matrix} \bar{f}(\eta) = F\eta, \\ \bar{g}(\eta) = G, \end{matrix} \quad \text{with} \quad F = \begin{bmatrix} O & I \\ O & O \end{bmatrix}, G = \begin{bmatrix} O \\ I \end{bmatrix}. \qquad (6)$$

B. Control Lyapunov Function based Quadratic Programs

To enable directly controlling to the rate of convergence, we use a *rapidly exponentially stabilizing control Lyapunov function (RES-CLF)*, introduced in [3]. RES-CLFs provide guarantees of *rapid exponential stability* for the traverse variables η. In particular, a function $V_\varepsilon(\eta)$ is a *rapidly exponentially stabilizing control Lyapunov function (RES-CLF)* for the system (1) if there exist positive constants $c_1, c_2, c_3 > 0$ such that for all $0 < \varepsilon < 1$ and all states (η, z),

$$c_1 \|\eta\|^2 \le V_\varepsilon(\eta) \le \frac{c_2}{\varepsilon^2}\|\eta\|^2, \qquad (7)$$

$$\dot{V}_\varepsilon(\eta, \mu) + \frac{c_3}{\varepsilon} V_\varepsilon(\eta) \le 0. \qquad (8)$$

The RES-CLF will take the form

$$V_\varepsilon(\eta) = \eta^T \begin{bmatrix} \frac{1}{\varepsilon} I & 0 \\ 0 & I \end{bmatrix} P \begin{bmatrix} \frac{1}{\varepsilon} I & 0 \\ 0 & I \end{bmatrix} \eta =: \eta^T P_\varepsilon \eta, \qquad (9)$$

with its time derivative computed as

$$\dot{V}_\varepsilon(\eta, \mu) = L_{\bar{f}} V_\varepsilon(\eta) + L_{\bar{g}} V_\varepsilon(\eta)\mu, \qquad (10)$$

where

$$L_{\bar{f}} V_\varepsilon(\eta) = \eta^T (F^T P_\varepsilon + P_\varepsilon F)\eta, \; L_{\bar{g}} V_\varepsilon(\eta) = 2\eta^T P_\varepsilon G. \qquad (11)$$

The following (CLF-QP)-based controller, introduced in [7], directly selects μ through an online quadratic program to enforce (8), as well as additional constraints of the form,

$$A_c^u(x) u \le b_c^u(x). \qquad (12)$$

CLF-QP:

$$\mu^* = \underset{\mu, d}{\operatorname{argmin}} \quad \mu^T \mu + p d^2 \qquad (13)$$
$$\text{s.t.} \quad \psi_{0,\varepsilon}(\eta) + \psi_{1,\varepsilon}(\eta)\, \mu \le d \qquad \textbf{(CLF)}$$
$$\psi_0^c(x) + \psi_1^c(x)\mu \le 0 \qquad \textbf{(Constraints)}$$

---

where the CLF constraint is relaxed through d, and

$$\psi_{0,\varepsilon}(\eta) = L_{\bar{f}} V_\varepsilon(\eta) + \frac{c_3}{\varepsilon} V_\varepsilon(\eta), \quad \psi_{1,\varepsilon}(\eta) = L_{\bar{g}} V_\varepsilon(\eta),$$
$$\psi_0^c(x) = -b_c^u(x) - A_c^u(x)(L_g L_f y(x))^{-1} L_f^2 y(x),$$
$$\psi_1^c(x) = -A_c^u(x)(L_g L_f y(x))^{-1}. \qquad (14)$$

Having revisited control Lyapunov function based quadratic programs, we will next revisit control Barrier functions.

C. Control Barrier Function

Control barrier functions were introduced in [1] to enable designing controllers that enforce the forward invariance of the safety set

$$\mathcal{C} = \{x \in \mathbb{R}^n : h(x) \ge 0\}, \qquad (15)$$

where $h : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function. A function $B : \mathcal{C} \to \mathbb{R}$ is a Control Barrier Function (CBF) [1] if there exists class $\mathcal{K}$ function α_1 and α_2 such that, for all $x \in Int(\mathcal{C}) = \{x \in \mathbb{R}^n : h(x) > 0\}$,

$$\frac{1}{\alpha_1(h(x))} \le B(x) \le \frac{1}{\alpha_2(h(x))}, \qquad (16)$$

$$\dot{B}(x, u) = L_f B(x) + L_g B(x) u \le \frac{\gamma}{B(x)}. \qquad (17)$$

The existence of a Barrier function $B(x)$ that satisfies the CBF condition in (17), guarantees that if $x(0) = x_0 \in \mathcal{C}$, i.e., $h(x_0) \ge 0$, then $x = x(t) \in \mathcal{C}, \forall t$, i.e., $h(x(t)) \ge 0, \forall t$.

We now can incorporate the above CBF condition into CLF-QP controller as follow:

CBF-CLF-QP:

$$\begin{aligned} \mu^* = &\underset{\mu,d}{\operatorname{argmin}} \quad \mu^T \mu + p d^2 & (18)\\ &\text{s.t.} \quad \psi_{0,\varepsilon}(\eta) + \psi_{1,\varepsilon}(\eta)\ \mu \le d & \textbf{(CLF)}\\ &\quad\quad\ \psi_0^b(x) + \psi_1^b(x)\ \mu \le 0 & \textbf{(CBF)}\\ &\quad\quad\ \psi_0^c(x) + \psi_1^c(x)\mu \le 0 & \textbf{(Constraints)} \end{aligned}$$

where,

$$\begin{aligned} \psi_0^b(x) &= L_f B(x) + L_g B u^*(x) - \frac{\gamma}{B(x)},\\ \psi_1^b(x) &= L_g B(x) L_g L_f^{-1} y(x), \end{aligned} \tag{19}$$

with u^* being defined in (4).

D. Modification of CBF for position based safety constraints

The CBF introduced above is for $h(x)$ with relative degree one. For application of CBF to robotic systems, we need to consider position based safety constraints, or functions of the form $g_b(x) \ge 0$ with relative degree two. The modification of CBF with position based constraint was simultaneously developed in [18], [8]. Here, we employ the simpler construction from [18], and construct a relative degree one barrier constraint as follows,

$$h_{CBF}(x) := \gamma_b g_b(x) + \dot{g}_b(x) \ge 0. \tag{20}$$

Now, if we have an initial condition x_0 satisfying $g_b(x_0) \ge 0$, and $\gamma_b > 0$, such that $h_{CBF}(x_0) \ge 0$, then $h_{CBF}(x) \ge 0 \implies g_b(x) \ge 0$. This is true since in the extreme case when $g_b(x) = 0$, $h_{CBF}(x) \ge 0 \implies \dot{g}_b(x) \ge 0$, which ensures the constraint $g_b(x) \ge 0$ is not violated. In other words, guaranteeing the barrier constraint (20) guarantees $g_b(x) \ge 0$. Based on this modification, we now can apply the CBF-CLF-QP based controller (18) with the barrier constraint $h(x) = h_{CBF}(x)$.

III. Adverse Effects of Uncertainty in Dynamics on the CBF-CLF-QP controller

The optimization-based control approaches presented in Section II have several interesting properties. Firstly, they provide a guarantee on the exponential stability of the system through Lyapunov functions, they are optimal with respect to some cost function and result in the minimum control effort, provide means of balancing conflicting requirements between performance and input / state-based constraints, and finally provide guarantees on enforcing safety-critical constraints through Barrier functions.

However, a primary disadvantage of these controllers is that they require an accurate dynamical model of the system. Uncertainty in the model can cause poor quality of control leading to tracking errors, and could potentially lead to instability [13]. Moreover uncertainty in the model also makes enforcing input and state constraints on the true system hard. Furthermore, uncertainty could potentially lead to violation of the safety-critical constraints [19]. In this section, we will explore the effect of uncertainty on the CLF-QP controller, input / state constraints, and safety constraints enforced by the CBF-QP controller.

A. Effect of uncertainty on CLF-QP

We begin by considering uncertainty in the dynamics and assume that the vector fields, $f(x), g(x)$ of the true dynamics (1), are unknown. Instead, we have to design our controller based on the nominal vector fields $\tilde{f}(x), \tilde{g}(x)$. Then, the pre-control law (3) get's reformulated as

$$u(x) = u^*(x) + (L_{\tilde{g}} L_{\tilde{f}} y(x))^{-1} \mu, \tag{21}$$

with

$$u^*(x) := -(L_{\tilde{g}} L_{\tilde{f}} y(x))^{-1} L_{\tilde{f}}^2 y(x), \tag{22}$$

where we have used the known nominal model rather than the unknown true dynamics. Substituting $u(x)$ from (21) into (2), the input-output linearized system then becomes

$$\ddot{y} = \mu + \Delta_1 + \Delta_2 \mu, \tag{23}$$

where

$$\begin{aligned} \Delta_1 &= L_f^2 y(x) - L_g L_f y(x) (L_{\tilde{g}} L_{\tilde{f}} y(x))^{-1} L_{\tilde{f}}^2 y(x),\\ \Delta_2 &= L_g L_f y(x) (L_{\tilde{g}} L_{\tilde{f}} y(x))^{-1} - I. \end{aligned} \tag{24}$$

Remark 1: In the definitions of Δ_1, Δ_2, note that when there is no model uncertainty, i.e., $\tilde{f} = f, \tilde{g} = g$, then $\Delta_1 = \Delta_2 = 0$.

Using F and G from (6) and defining,

$$\Delta H := \begin{bmatrix} O \\ \Delta_1 \end{bmatrix}, \qquad \Delta G := \begin{bmatrix} O \\ \Delta_2 \end{bmatrix}, \tag{25}$$

the closed-loop system takes the form

$$\dot{\eta} = F\eta + (G + \Delta G)\mu + \Delta H. \tag{26}$$

For $\Delta H \ne 0$, the closed-loop system does not have an equilibrium, and for $\Delta G \ne 0$, the controller could potentially destabilize the system. This raises the question of whether it's possible for controllers to account for this model uncertainty, and if so, how do we design such a controller.

B. Effect of uncertainty on constraints

The input / state constraints in (12) depend on the model explicitly, and can be rewritten explicitly as,

$$A_c^u(x, f, g) u \le b_c^u(x, f, g). \tag{27}$$

If a controller naively enforces these constraints using the nominal model available to the controller, the controller will enforce the constraint

$$A_c^u(x, \tilde{f}, \tilde{g}) u \le b_c^u(x, \tilde{f}, \tilde{g}). \tag{28}$$

On the true model, this is an invalid constraint and provides no guarantees on enforcing the original constraint (27).

To clearly see how the true constraint would depend on both models, we substitute u from (21) into (27) to obtain,

$$\psi_0^c(x, f, g, \tilde{f}, \tilde{g}) + \psi_1^c(x, f, g, \tilde{f}, \tilde{g})\mu \le 0. \tag{29}$$

where,

$$\psi_0^c(x,f,g,\tilde{f},\tilde{g}) := -\, b_c^u(x,f,g) - A_c^u(x,f,g)(L_{\tilde{g}} L_{\tilde{f}} y)^{-1} L_{\tilde{f}}^2 y,$$
$$\psi_1^c(x,f,g,\tilde{f},\tilde{g}) := A_c^u(x,f,g)(L_{\tilde{g}} L_{\tilde{f}} y)^{-1}. \tag{30}$$

The true constraint to be enforced now becomes (29). As can be seen from (30), it's challenging to enforce this constraint without requiring knowledge of the true model in addition to the nominal model.

Remark 2: It must be noted that certain constraints do not depend on the model at all. In such cases, model uncertainty doesnt affect the constraint. One example of such a constraint is a pure input constraint, such as $u(x) \leq u_{max}$. Expressing this constraint in the form of (27) results in $A_c^u = I, b_c^u = u_{max}$, which clearly is not dependent on the model.

C. Effect of Uncertainty on CBF

Having seen the effect of uncertainty on constraints, we will now see the effect of uncertainty on control Barrier functions. Our formulation will proceed in a parallel manner as presented for the constraints. We start with the time-derivative of the Barrier function in (17) and note that the constraint we need to enforce is

$$L_f B(x) + L_g B(x) u - \frac{\gamma}{B(x)} \leq 0, \tag{31}$$

where $L_f B(x), L_g B(x)$ depends on the true model of the system. As seen in the case of constraints, naively enforcing this barrier constraint using the nominal model results in,

$$L_{\tilde{f}} B(x) + L_{\tilde{g}} B(x) u - \frac{\gamma}{B(x)} \leq 0. \tag{32}$$

Clearly this constraint is different from the previous one. In fact, as analyzed in [19], this results in violation of the safety-critical constraint established by the Barrier function.

To clearly see how the Barrier constraint depends on the true and nominal models, we substitute u from (21) into (31) to obtain,

$$\psi_0^b(x,f,g,\tilde{f},\tilde{g}) + \psi_1^b(x,f,g,\tilde{f},\tilde{g})\mu \leq 0, \tag{33}$$

where,

$$\psi_1^b(x,f,g,\tilde{f},\tilde{g}) := L_f B(x)(L_{\tilde{g}} L_{\tilde{f}} y(x))^{-1},$$
$$\psi_0^b(x,f,g,\tilde{f},\tilde{g}) := L_f B(x) - \frac{\gamma}{B(x)} - L_g B(x)(L_{\tilde{g}} L_{\tilde{f}} y(x))^{-1} L_{\tilde{f}}^2 y(x). \tag{34}$$

The correct barrier constraint to be enforced becomes (33), however, it's challenging to enforce this barrier constraint without knowledge of the true model in addition to the nominal model of the system.

Remark 3: For the I/O linearized system, (5), F, G are linear, time-invariant, and state-independent. For the CLF, we can therefore evaluate the uncertainty based on the difference with a static nominal model to obtain (26). However, because the relation with the control input μ for state / input based constraints ($\psi_0^c(x), \psi_1^c(x)$) in (29)) and CBF constraint ($\psi_0^b(x), \psi_1^b(x)$) in (33)) are not linear and invariant (those functions are dependent on the system state x), the same approach to explore model uncertainty could encounter difficulties. In order to address this issue, we will introduce an alternative design termed "Virtual Input-Output Linearization" that will help us to develop a novel controller in the next section called Robust CBF-CLF-QP with Robust Constraints.

IV. Proposed Robust Control based Quadratic Programs

A. Robust CLF-QP

Having discussed the effect of model uncertainty on the control Lyapunov function based controllers in Section III, we now present a robust controller, developed in [13], that can guarantee tracking and stability in the presence of bounded model uncertainty. As we will see, we will extend this controller to robustify input / state constraints as well as safety constraints encoded in CBFs.

For the following sections, we will abuse notation and redefine $\bar{f}, \bar{g}$ from (6) as

$$\bar{f} = F\eta + \Delta H, \quad \bar{g} = G + \Delta G. \tag{35}$$

We start with the closed-loop model with uncertainty as developed in (26). With the CLF defined in (9), we then have:

$$\dot{V}_\varepsilon = L_{\bar{f}} V_\varepsilon(\eta, z) + L_{\bar{g}} V_\varepsilon(\eta, z)\mu, \tag{36}$$

where,

$$L_{\bar{f}} V_\varepsilon(\eta, z) = \eta^T(F^T P_\varepsilon + P_\varepsilon F)\eta + 2\eta^T P_\varepsilon \Delta H,$$
$$L_{\bar{g}} V_\varepsilon(\eta, z) = 2\eta^T P_\varepsilon (G + \Delta G). \tag{37}$$

The RES condition (8) then becomes:

$$\dot{V}_\varepsilon(\eta, \Delta G, \Delta H, \mu) + \frac{c_3}{\varepsilon} V_\varepsilon \leq 0. \tag{38}$$

Satisfying this inequality for all unknown $\Delta H, \Delta G$ defined in (25) is generally not possible. To address this, we assume the uncertainty is bounded as follows

$$\|\Delta H\| \leq \Delta H_{max}, \qquad \|\Delta G\| \leq \Delta G_{max}, \tag{39}$$

where the first norm is a vector norm, while the second norm is a matrix norm.

The goal for the robust control design is to find μ that satisfies the RES condition (8), evaluated through the given bounds of uncertainty in (39). With the bounded uncertainty assumption, the RES condition (8) will hold if the following inequalities hold:

$$\psi_{0,\varepsilon}^{max} + \psi_{1,\varepsilon}^{p} \mu \leq 0,$$
$$\psi_{0,\varepsilon}^{max} + \psi_{1,\varepsilon}^{n} \mu \leq 0, \tag{40}$$

where,

$$\psi_{0,\varepsilon}^{max} := \max\left(\psi_{0,\varepsilon}^{n}, \psi_{0,\varepsilon}^{p}\right),$$
$$\psi_{0,\varepsilon}^{p} := \eta^T(F^T P_\varepsilon + P_\varepsilon F)\eta + 2\eta^T P_\varepsilon \boxed{\begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix}} \Delta H_{max} + \frac{c_3}{\varepsilon} V_\varepsilon,$$
$$\psi_{0,\varepsilon}^{n} := \eta^T(F^T P_\varepsilon + P_\varepsilon F)\eta - 2\eta^T P_\varepsilon \boxed{\begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix}} \Delta H_{max} + \frac{c_3}{\varepsilon} V_\varepsilon,$$
$$\psi_{1,\varepsilon}^{p} := 2\eta^T P_\varepsilon G(1 + \Delta G_{max}),$$
$$\psi_{1,\varepsilon}^{n} := 2\eta^T P_\varepsilon G(1 - \Delta G_{max}). \tag{41}$$

This robust control problem can be solved through the robust CLF-QP presented in [13]:

$$\begin{aligned} \mu^* = & \underset{\mu, d_1, d_2}{\text{argmin}} && \mu^T \mu + p_1 d_1^2 + p_2 d_2^2 && (42) \\ & \text{s.t.} && \psi_{0,\varepsilon}^{max}(\eta, z) + \psi_{1,\varepsilon}^{p}(\eta, z)\ \mu \leq d_1, \\ &&& \psi_{0,\varepsilon}^{max}(\eta, z) + \psi_{1,\varepsilon}^{n}(\eta, z)\ \mu \leq d_2, \\ &&& \psi_0^c(x) + \psi_1^c(x)\mu \leq 0. && \textbf{(Constraints)} \end{aligned}$$

B. Virtual Input-Ouput Linearization

Our development of the robust CLF-QP controller does not enable us to directly enforce constraints in the presence of model uncertainty, preventing us from robustifying input and state constraints as well as safety constraints that are encoded in the form of CBFs. On the other hand, robustifying the CLF condition was easier. This is because in the presence of model uncertainty, the I/O linearization (2) with the pre-control (3) results in (23). This form of (23) was essential for "robustifying" the CLF-QP condition (see Section IV-A). So, the question then is how to extend this to CBF and input / state constraints. As mentioned in Remark 3, since the effect of uncertainty for CBF constraints with respect to control input μ of the I/O linearization (3) is not static, it causes difficulties in evaluating effects of uncertainty on the CBF constraints.

One idea to address this is to get the CBF condition to the same form of the CLF condition through another I/O linearization for the CBF (17). This will robustify CBFs and enable creating a unified robust CBF-CLF-QP. However this idea is not directly feasible since $L_g B$ is a vector and obviously not invertible. One approach that can potentially work for this is to use MIMO non-square input-output linearization [9].

Here, we solve this problem through optimization by introducing a notion of Virtual Input-Output Linearization (VIOL), wherein an invertible decoupling matrix is not required. We will explain this method for CBF constraints (a similar formulation holds for input and state constraints). We begin with the CBF $B(x)$ and define a virtual control input μ_b such that (17) can be written as,

$$\dot{B}(x, \mu) = A_b(x)\mu + b_b(x) =: \mu_b(x, \mu), \tag{43}$$

where

$$\begin{aligned} A_b(x) &:= L_g B(x) L_g L_f^{-1} y(x), \\ b_b(x) &:= L_f B(x) + L_g B(x) u^*(x), \end{aligned} \tag{44}$$

with $u^*(x)$ as defined in (4). The CBF condition (17) can then be written as

$$\mu_b(x, u) \leq \frac{\gamma}{B(x)}, \tag{45}$$

which can be rewritten as,

$$\psi_0^{bv}(x) + \psi_1^{bv}(x)\mu_b \leq 0, \tag{46}$$

where

$$\psi_0^{bv} := -\frac{\gamma}{B(x)}, \qquad \psi_1^{bv} := 1. \tag{47}$$

The superscript "bv" stands for Barrier function based VIOL.

We can then reformulate the CBF-CLF-QP from (18) such that the QP computes μ_b so as to simultaneously satisfy both (43) and (46):

$$\begin{aligned} \mu^* = & \underset{\mu, \mu_b, d}{\text{argmin}} && \mu^T \mu + p d^2 && (48) \\ & \text{s.t.} && \psi_{0,\varepsilon}(\eta) + \psi_{1,\varepsilon}(\eta)\ \mu \leq d && \textbf{(CLF)} \\ &&& \psi_0^{bv}(x) + \psi_1^{bv}(x)\mu_b \leq 0 && \textbf{(CBF)} \\ &&& A_b(x)\mu + b_b(x) = \mu_b && \textbf{(VIOL)} \\ &&& \psi_0^c(x) + \psi_1^c(x)\mu \leq 0. && \textbf{(Constraints)} \end{aligned}$$

Here, the equality constraint of the Virtual Input-Ouput Linearization (VIOL) is as presented in (43). The solutions of two CBF-CLF-QP controllers (18) and (48) are exactly the same. However, the VIOL in the CBF-CLF-QP opens up a systematic way to design a robust CBF-CLF-QP controller that will be introduced next. Note that this method also enables creating "exponential" Barrier functions that enforce safety constraints with arbitrary high relative-degree [15].

C. Robust CBF-CLF-QP with Robust Constraints

By using VIOL, the CBF now takes the similar form of a linear system, $\dot{B}(x, \mu_b) = \mu_b$, and therefore the effect of uncertainty can be easily extended by using the same approach as with the robust CLF-QP to obtain,

$$\dot{B}(x, \Delta_1^b, \Delta_2^b, \mu_b) = \mu_b + \Delta_1^b + \Delta_2^b \mu_b, \tag{49}$$

where Δ_1^b, Δ_2^b are functions of both the true and nominal system models. The CBF condition (46) then becomes,

$$\psi_0^{bv}(x) + \psi_1^{bv}(x)(\mu_b + \Delta_1^b + \Delta_2^b \mu_b) \leq 0. \tag{50}$$

Because we developed the CLF and CBF to have a similar form for the I/O linearized system, we now have a systematic way to develop the Robust CBF-CLF-QP.

We will again assume that our model uncertainty is bounded, i.e.,

$$||\Delta_1^b|| \leq \Delta_{1,max}^b, ||\Delta_2^b|| \leq \Delta_{2,max}^b. \tag{51}$$

Then, similarly as with the Robust CLF-QP, we now define:

$$\psi_{0,bv}^{max} := max(\psi_{0,p}^{bv}, \psi_{0,n}^{bv}), \tag{52}$$

$$\psi_{0,bv}^{p} := \psi_0^{bv} + \psi_1^{bv} \Delta_{1,max}^b, \tag{53}$$

$$\psi_{0,bv}^{n} := \psi_0^{bv} - \psi_1^{bv} \Delta_{1,max}^b, \tag{54}$$

$$\psi_{1,bv}^{p} := \psi_1^{bv}(1 + \Delta_{2,max}^b), \tag{55}$$

$$\psi_{1,bv}^{n} := \psi_1^{bv}(1 - \Delta_{2,max}^b). \tag{56}$$

The robust version of the CBF constraints will then become,

$$\psi_{0,bv}^{max}(x) + \psi_{1,bv}^{p}(x)\mu_b \leq 0, \tag{57}$$

$$\psi_{0,bv}^{max}(x) + \psi_{1,bv}^{n}(x)\mu_b \leq 0 \tag{58}$$

The same process can be applied to robustify input and state constraints as well. We finally unify the the robust CLF for stability under model uncertainty, robust CBF for safety enforcement under model uncertainty, and the robust constraints to obtain the following unified robust controller.

Robust CBF-CLF-QP with Robust Constraints:

$$\underset{\mu,d_1,d_2}{\text{argmin}} \quad \mu^T\mu + p_1 d_1^2 + p_2 d_2^2 \tag{59}$$

$$\text{s.t.} \quad \psi_{0,\varepsilon}^{max}(\eta) + \psi_{1,\varepsilon}^{p}(\eta)\ \mu \le d_1, \qquad \textbf{(Robust CLF)}$$
$$\psi_{0,\varepsilon}^{max}(\eta) + \psi_{1,\varepsilon}^{n}(\eta)\ \mu \le d_2,$$

$$\psi_{0,bv}^{max}(x) + \psi_{1,bv}^{p}(x)\ \mu_b \le 0, \qquad \textbf{(Robust CBF)}$$
$$\psi_{0,bv}^{max}(x) + \psi_{1,bv}^{n}(x)\ \mu_b \le 0,$$

(Robust Constraints)
$$\psi_{0,cv}^{max}(x) + \psi_{1,cv}^{p}(x)\ \mu_c \le 0,$$
$$\psi_{0,cv}^{max}(x) + \psi_{1,cv}^{n}(x)\ \mu_c \le 0,$$

$$A_b(x)\mu + b_b(x) = \mu_b, \qquad \textbf{(VIOL)}$$
$$A_c(x)\mu + b_c(x) = \mu_c.$$

Remark 4: Note that $\psi_{0,\varepsilon}^{max}, \psi_{0,bv}^{max}, \psi_{0,cv}^{max}$ are defined as the maximum of two continuous functions, and are thus continous. This then results in continuously varying constraints that are enforced point-wise in time, resulting in Lipschitz continuous control inputs [12].

V. Application: Dynamic Bipedal Walking while Carrying Unknown Load, subject to Torque Constraints, Contact Force Constraints, and Foot-Step Location Constraints

To demonstrate the effectiveness of the proposed robust CBF-CLF-QP controller, we will conduct numerical simulations on the model of RABBIT, [4], a planar five-link bipedal robot with a torso and two legs with revolute knees that terminate in point feet. RABBIT weighs 32 kg, has four brushless DC actuators with harmonic drives to control the hip and knee angles, and is connected to a rotating boom which constrains the robot to walk in a circle, approximating planar motion in the sagittal plane. The dynamical model of RABBIT is nonlinear and hybrid, comprising of a continuous-time underactuated stance phase and a discrete-time impact map.

We validate the performance of our proposed robust controller through dynamic bipedal walking on RABBIT, subject to model uncertainty while simultaneously requiring enforcement of input constraints, state constraints, and safety-critical constraints. Model uncertainty appears in the form of an unknown heavy load added to the torso of RABBIT.

In bipedal robotic walking, input constraints arise as constraints on motor torques and state constraints could arise as ground contact force constraints, which appear both as a unilateral constraint on the vertical component of the force at the stance foot and as a friction cone constraint. If $F(x)$ and $N(x)$ are state-dependent friction force and vertical contact

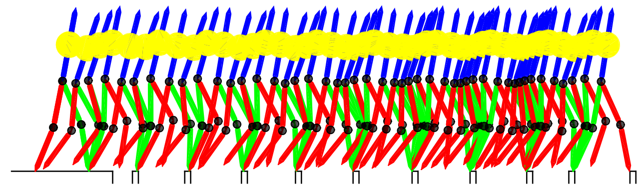


Fig. 1: Dynamic bipedal walking while carrying unknown load, subject to torque saturation constraints (input constraints), contact force constraints (state constraints), and foot-step location constraints (safety constraints). Two simulations of the Robust CBF-CLF-QP with Robust Constraints controller for walking over 10 discrete foot holds is shown, subject to model uncertainty of 15 Kg (47 %). Simulation video: `http://youtu.be/tT0xE1XlyDI`

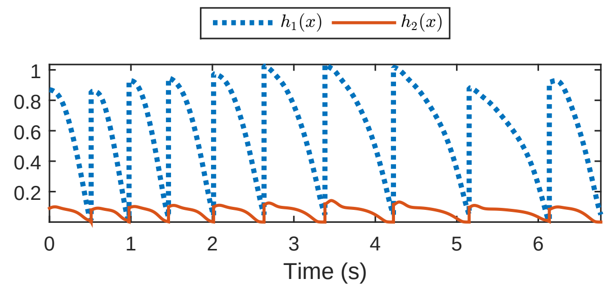


Fig. 2: Dynamic walking of bipedal robot while carrying unknown load of 15 Kg (47 %). The CBF constraints, $h_1(x) \ge 0$ and $h_2(x) \ge 0$ defined in [14], guarantee precise foot-step locations. The figure depicts data for 10 steps of walking. As can be clearly seen, the constraints are strictly enforced despite the large model uncertainty.

force between the stance foot and the ground, then, in order to avoid slipping during walking, we need to guarantee:

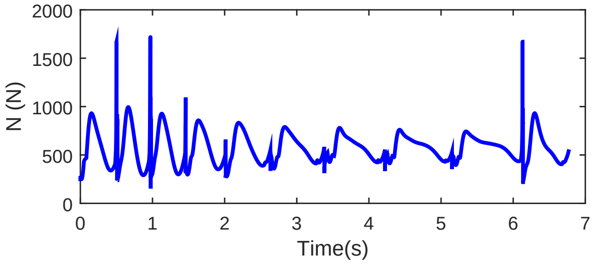
$$N(x) \ge \delta_N > 0, \tag{60}$$
$$\left|\frac{F(x)}{N(x)}\right| \le k_f, \tag{61}$$

where δ_N is a positive threshold for the vertical contact force, and k_f is the friction coefficient.

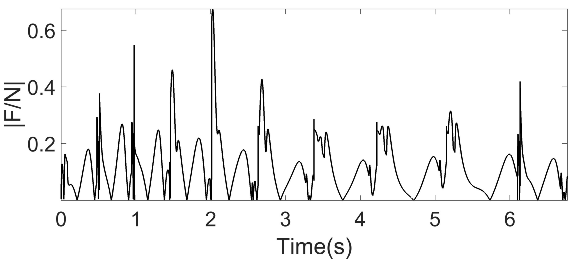
Note that, contact force constraints are very important for the problem of robotics walking. Any violation of these constraints will result in the leg slipping and the robot potentially falling. Although a nominal walking gait is usually designed to respect these constraints, however, we cannot guarantee these constraints under transients or under model uncertainty.

We also consider safety constraints in the form of precise foot placement constraints that need to be critically enforced to safely walk over a terrain of discrete footholds. In prior work [14], these foot placement constraints were formulated as safety constraints through CBFs and enforced assuming perfect knowledge of the system model. We consider these CBF constraints here to demonstrate walking over discrete footholds while subject to model uncertainty.

We consider the above constraints - torque constraints of 150 Nm as per the motor specifications, ground contact



(a) Vertical Contact Force: $N(x) > \delta_N, (\delta_N = 0.1mg)$.



(b) Friction Constraint: $|F/N| \leq k_f$, ($k_f = 0.8$)

Fig. 3: Dynamic walking of bipedal robot while carrying unknown load of 15 Kg (47 %). (a) Vertical contact force constraint and (b) friction constraint are shown for 10 steps walking. As is evident, both constraints are strictly enforced despite the large model uncertainty.

constraints with $\delta_N = 0.1mg$, $k_f = 0.8$, and precise footstep constraints. We ran 100 random simulations, where for each simulation, the unknown load was choosen randomly between 5-15 Kg, and 10 discrete footholds were randomly generated with step lengths between 0.35-0.55 m (the nominal walking gait has a step length of 0.45 m). A run was marked as a failure if either (a) foot placement constraints were violated, or (b) contact force constraints were violated during the simulation. The following two controllers were evaluated on each run: (A) CBF-CLF-QP with Constraints; (B) Robust CBF-CLF-QP with Robust Constraints.

The same set of random parameters was tested on the two controllers. While the nominal CBF-CLF-QP succeeded in only 2% of the tests, the Robust CBF-CLF-QP controller was successful for 98% tests. This result not only strengthens the effectiveness of the proposed controller, but it also emphasizes the importance of considering robust control especially for safety constraints, where even a small model uncertainty can cause violation of such safety critical constraints. Figures 1, 2, 3 illustrate one of the runs where the maximum load of 15 Kg (47% of robot mass) was considered. Stick figure plots, CBF constraints, vertical contact force, and friction constraint plots are shown. Note that, the simulations were artificially limited to 10 steps, to enable fast execution of 100 runs for each controller. Simulations for larger number of steps were also successful as well, but are not presented here due to space constraints.

VI. Conclusion

We have presented a novel method of optimal robust control through quadratic programs for nonlinear hybrid systems to handle stability while enforcing input constraints, state dependent constraints, as well as safety-critical constraints, all in the presence of high level of model uncertainty. The controller was evaluated on a model of RABBIT with a large model uncertainty, in the form of an unknown load on the torso of up to 15 Kg (47% of the robot mass), to achieve dynamic bipedal walking while simultaneously subject to enforcing strict torque saturation, contact force, and precise footstep placements constraints.

References

[1] A. D. Ames, J. Grizzle, and P. Tabuada, “Control barrier function based quadratic programs with application to adaptive cruise control,” in *IEEE Conference on Decision and Control*, 2014.

[2] A. D. Ames, K. Galloway, and J. W. Grizzle, “Control Lyapunov Functions and Hybrid Zero Dynamics,” *Proc. 51st IEEE Conf. Decision and Control*, pp. 6837–6842, 2012.

[3] A. D. Ames, K. Galloway, J. W. Grizzle, and K. Sreenath, “Rapidly Exponentially Stabilizing Control Lyapunov Functions and Hybrid Zero Dynamics,” *IEEE Trans. Automatic Control*, vol. 59, no. 4, pp. 876–891, 2014.

[4] C. Chevallereau, G. Abba, Y. Aoustin, F. Plestan, E. R. Westervelt, C. Canudas-de-Wit, and J. W. Grizzle, “RABBIT: A testbed for advanced control theory,” *IEEE Control Systems Magazine*, vol. 23, no. 5, pp. 57–79, October 2003.

[5] C. Edwards and S. Spurgeon, *Sliding Mode Control: Theory And Applications*. CRC Press, 1998.

[6] R. A. Freeman and P. V. Kokotović, *Robust Nonlinear Control Design*. Birkhäuser, 1996.

[7] K. Galloway, K. Sreenath, A. D. Ames, and J. W. Grizzle, “Torque saturation in bipedal robotic walking through control lyapunov function based quadratic programs,” *IEEE Access*, vol. PP, no. 99, p. 1, April 2015.

[8] S.-C. Hsu, X. Xu, and A. D. Ames, “Control barrier function based quadratic programs with application to bipedal robotic walking,” in *American Control Conference*, 2015.

[9] S. Kolavennu, S. Palanki, and J. Cockburn, “Nonlinear control of nonsquare multivariable systems,” *Chemical Engineering Science*, vol. 56, pp. 2103–2110, 2001.

[10] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal Control*. John Wiley & Sons, 2012.

[11] A. Mehra, W.-L. Ma, F. Berg, P. Tabuada, J. W. Grizzle, and A. D. Ames, “Adaptive cruise control: Experimental validation of advanced controllers on scale-model cars,” in *American Control Conference*, 2015.

[12] B. Morris, M. J. Powell, and A. D. Ames, “Sufficient conditions for the Lipschitz continuity of QP-based multi-objective control of humanoid robots,” *IEEE Conference on Decision and Control*, pp. 2920–2926, Dec. 2013.

[13] Q. Nguyen and K. Sreenath, “Optimal robust control for bipedal robots through control lyapunov function based quadratic programs,” in *Robotics: Science and Systems*, 2015.

[14] ——, “Safety-critical control for dynamical bipedal walking with precise footstep placement,” in *The IFAC Conference on Analysis and Design of Hybrid Systems*, vol. 48, no. 27, Atlanta, GA, October 2015, pp. 147–154.

[15] ——, “Exponential control barrier functions for enforcing high relative-degree safety-critical constraints,” in *American Control Conference*, Boston, MA, to appear, 2016.

[16] E. D. Sontag, “On the input-to-state stability property,” *European Journal of Control*, vol. 1, pp. 24–36, 1995.

[17] L. Vu, D. Chatterjee, and D. Liberzon, “Input-to-state stability of switched systems and switching adaptive control,” *Automatica*, vol. 43, pp. 639–646, 2007.

[18] G. Wu and K. Sreenath, “Safety-critical and constrained geometric control synthesis using control lyapunov and control barrier functions for systems evolving on manifolds,” in *American Control Conference*, 2015.

[19] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames, “Robustness of control barrier functions for safety critical control,” in *The IFAC Conference on Analysis and Design of Hybrid Systems*, To appear, 2015.