CONTROL BARRIER FUNCTION BASED QUADRATIC PROGRAMS WITH

APPLICATION TO BIPEDAL ROBOTIC WALKING


A Thesis

by

SHAO-CHEN HSU


Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE


| | |
|---|---|
| Chair of Committee, | Aaron D. Ames |
| Committee Members, | Wong-Jon Kim |
| | Reza Langari |
| Head of Department, | Andreas A. Polycarpou |


December 2014


Major Subject: Department of Mechanical Engineering

ABSTRACT

This thesis presents a methodology for the development of control barrier functions (CBFs) through a backstepping inspired approach. Given a set defined as the superlevel set of a function, $h$, the main result is a constructive means for generating control barrier functions that guarantee forward invariance of this set. In particular, if the function defining the set has relative degree $n$, an iterative methodology utilizing higher order derivatives of $h$ provably results in a control barrier function that can be explicitly derived. To demonstrate these formal results, they are applied in the context of bipedal robotic walking. Physical constraints, e.g., joint limits, are represented by control barrier functions and unified with control objectives expressed through control Lyapunov functions (CLFs) via quadratic program (QP) based controllers. The end result is the generation of stable walking satisfying physical realizability constraints for a model of the bipedal robot AMBER2.

DEDICATION

To my parents, who always encourage me and give the best support so that I can feel free to study in the United States.

# ACKNOWLEDGEMENTS

TABLE OF CONTENTS

LIST OF FIGURES

FIGURE                                                                    Page

# LIST OF TABLES

TABLE                                                                    Page

# 1. INTRODUCTION

Humans can perform many difficult dynamic behaviors with ease, including: crawling, climbing and — of special focus in this thesis — walking. Bipedal robots provide simplified prototypes of human systems and, as such, provide a means in which to understand human behaviors from a formal perspective. Current state of the art in robotic walking technology is still inferior to the performance of the human walking behaviors. Regardless, robotic walking has progressed by leaps and bounds in the last couple of decades. For the purpose of conserving energy, passive walking robots have been designed in a very efficient fashion [10, 27, 28]. Without any torque input, passive walking robots can save the energy from previous step and walk forward. Another model that represents energy efficient walking, called the SLIP model, has also been studied [32, 17], in which the robots can save energy by springing into the next forward step. On the other hand, Zero Moment Point (ZMP) is the most popular approach in bipedal robotics [19, 35, 36]. However, ZMP criterion is typically applied only to the case of flat-foot walking, and it does not guarantee stable dynamic walking [37]. Another approach to walking robot has been studied through Hybrid Zero Dynamics (HZD) [37, 16, 38, 2], which creates low-dimensional representations and renders highly dynamic locomotion through the use of nonlinear control methods.

At the core of performing robotic walking is the ability to satisfy structural and physical constraints while simultaneously realizing dynamics based control objective. Realizing this balance between safety constraints and control objectives in the context of dynamic behaviors has yet to be fully realized on robotic systems. A core issue preventing this is the unification of safety and control objectives in a single unified

framework—one that can be solved online in real-time, i.e., does not require *a priori* optimization, while still yielding formal guarantees of correctness. The goal of this thesis is to present a methodology for realizing physical constraints on robotic systems through control barrier functions, and balancing these constraints through control objectives represented as control Lyapunov functions expressed through a unified quadratic program based control methodology. The application of these ideas to robotic walking will demonstrate their affectiveness in ensuring physical constraints during dynamic behaviors.

Control Lyapunov functions (CLFs), which were mainly pioneered by Artstein and Sontag [8, 33], have been widely used in nonlinear control [24, 11, 12]. The concept of control Lyapunov functions is to design a set of controllers, which ensures that the derivative of a Lyapunov function is negative. Given a nonlinear system as an example,

$$\dot{x} = f(x, u), \quad \text{where} \quad f(0, u) = 0$$

for all $x \neq 0$, if there exists $u$ such that

$$\dot{V}(x, u) < 0,$$

where $V(x)$ is a positive definite function, then $V(x)$ is a control Lyapunov function. By utilizing $u$, the system converges to the equilibrium point $x = 0$. In the field of bipedal robotic locomotion, the method has been applied both in simulations and experiments [5, 3, 13].

Barrier functions, which are firstly utilized in numerical optimization methods [9, 40], are continuous functions whose values approach infinity when the state ap-

proaches the boundary of a set. For instance, given a set $C$, $B(x)$ is a barrier function, if $B(x) \to \infty$ as $x \to \partial C$. The concept recently has been related to control Lyapunov functions for the purposes of constructing nonlinear controllers [39]. In particular, Lyapunov-like barrier functions have been established; that is, employing derivatives of barrier functions guarantees the invariance of set $C$, e.g. $\dot{B}(x) < 0$ [30, 34, 31]. With a view towards expanding the class of control inputs that imply set invariance, recent work has focused on a new class of barrier functions that ensure set invariance while yielding a larger set of control inputs [4]. In particular, if $\mathcal{C}$ is the superlevel set of a function $h(x)$, there is the corresponding barrier function candidate:

$$B(x) = -\log\left(\frac{h(x)}{1 + h(x)}\right) \tag{1.1}$$

which is a valid barrier function if it satisfies the condition:

$$\dot{B}(x, u) < \frac{\gamma}{B(x)} \tag{1.2}$$

for $\gamma \geq 0$. Importantly, this allows for $B(x)$ to grow when it is far from the boundary of the set $C$ while still provably yielding set invariance [4]. Similar to CLFs, a control law can be designed such that the control barrier function (CBF) condition (1.2) is satisfied. Comparing with CLFs, which drive the system to a fixed point, barrier functions prevent the system from leaving a "safe" set. When formulating control barrier functions, it is important to determine if there exists a control law. For instance, in the case of a set defined as the superlevel set of a function of relative degree 2 [18], the control inputs do not appear in the derivative of the barrier function. This naturally points to the use of backstepping methods, first developed by Kokotović [21, 22, 25], in the context of constructing control barrier functions.

The concept of the backstepping method is to design a new function as a "virtual" input which stabilizes the subsystem and steps back the virtual input that stabilizes the next subsystem until the "true" control input is reached.

In this research, a quadratic program (QP) is formulated based on CLFs and CBFs. While CLFs drive actual outputs to desired outputs, CBFs keep outputs in a "safe" set. Thus, a conflict between the CLFs and the CBFs happens if the desired outputs are out of the set. This issue can be addressed by considering the CLFs as soft constraints, which results in the CBF constraints being satisfied. By relaxing the CLF constraints, they will be violated when approaching the boundary of the set; otherwise, the CLF constraints will be satisfied in the set. The end result is the unification of control objectives (formulated as CLFs) together with safety constraints (formulated as CBFs).

The goal of this research is to develop control barrier functions via methods motivated by Lyapunov backstepping. Ultimately, these methods will be applied to a 7-link bipedal robot, AMBER2. The following list enumerates the specific objectives of this research:

- Prove that CBFs can be constructed through methods motivated by Lyapunov backstepping so as to achieve forward invariance of sets defined as the superlevel set of functions.

- Formulate a QP based on CLFs and CBFs and apply the method to a simple nonlinear system example to show the improved performance of the method versus existing approaches.

- Formulate CBFs for bipedal robots so as to capture essential physical constraints in the system.

- Apply the resulting CBFs to AMBER2 in simulation so as to achieve a stable walking gait.

To summarize, the main result of this thesis is a novel methodology for developing control barrier functions with direction application to bipedal robotic walking.

The following contents are presented in the thesis. Section 1 briefly introduces a concept of walking bipedal robot, control Lyapunov functions, control barrier functions, and quadratic programs, together with a brief literature review. Section 2 gives a brief overview of control Lyapunov functions, control barrier functions and quadratic programs, and proves that a control barrier function designed through backstepping methods is a true control barrier functions that renders the corresponding set forward invariant; a simple example is presented that demonstrates the CLF-CBF-QP formulation. Section 3 describes the model of a 7-link bipedal robot, AMBER2, as a hybrid system (following from the fact that it displays both continuous and discrete dynamics). Section 4 introduces Human-Inspired Control through CLF-CBF-QP, and explicitly designs both CLFs and CBFs. Section 5 presents simulation results for the resulting stable walking gait achieved along with a discussion on the performance of the simulation. Finally, a conclusion based on the method and results is included in Section 6.

# 2. CONTROL LYAPUNOV FUNCTIONS, CONTROL BARRIER FUNCTION, AND QUADRATIC PROGRAMS

This chapter presents a brief overview of rapidly exponentially stabilizing control Lyapunov functions and control barrier functions for a nonlinear system, and a quadratic program is introduced to unify control Lyapunov functions and control barrier functions. More details can be found in [4] and [3].

Beside, This chapter develops and presents the main formal results of this research: a backstepping inspired methodology for constructing control barrier functions (CBFs). We begin by introducing the form of barrier functions considered in the thesis, as introduced in [4], defined for a set $\mathcal{C}$ that is the super level set of a function of the form: $z(x) = h(x) - k(x)$. Motivated by the use of backstepping in generating Lyapunov functions [34, 21], we assume that $h$ has relative degree $n$ and utilize the higher order derivatives of $h$ and $k$ to iteratively construct valid control barrier functions. The end product of this procedure yields the main result: a formal guarantee that the resulting control barrier function is valid, i.e., that control inputs exist that satisfy the barrier function condition (1.2).

## 2.1 Control Lyapunov Functions

Consider an affine nonlinear control system as follows:

$$\dot{x} = f(x) + g(x)u, \tag{2.1}$$

$$\dot{z} = q(x),$$

where $x \in \mathbb{R}^n$ and $u \in U = \mathbb{R}^m$ with $f$ and $g$ assumed to be locally Lipschitz. In addition, assume that $f(0) = 0$, resulting in an invariant surface $Z$ defined by

$x = 0$. To achieve exponential stability of $x$ to 0 we utilize a special class of control Lyapunov functions [3]:

**Definition 1.** *For the system* (2.1), *a one-parameter family of continuously differentiable function* $V_\varepsilon : X \to \mathbb{R}$ *is an **rapidly exponentially stabilizing control Lyapunov function (RES-CLF)** if there exists positive constants* $c_1, c_2, c_3 > 0$ *such that for all* $0 < \varepsilon < 1$ *and for all* $x \in X \times Z$

$$c_1 \|x\|^2 \leq V_\varepsilon(x) \leq \frac{c_2}{\varepsilon} \|x\|^2, \tag{2.2}$$

$$\inf_{u \in U} \left[ L_f V_\varepsilon(x) + L_g V_\varepsilon(x) u + \frac{c_3}{\varepsilon} V_\varepsilon(x) \right] \leq 0. \tag{2.3}$$

Given a RES-CLF, we define the set

$$K_\varepsilon(x) =$$
$$\{u \in U : L_f V_\varepsilon(x) + L_g V_\varepsilon(x) u + \frac{c_3}{\varepsilon} V_\varepsilon(x) \leq 0\}, \tag{2.4}$$

for which it follows that for any locally Lipschitz continuous feedback control law $u(x)$ such that $u(x) \in K_c(x)$, the solutions to the system (2.1) satisfy

$$\|x(t)\| \leq \frac{1}{\varepsilon} \sqrt{\frac{c_2}{c_1}} e^{-\frac{c_3}{2\varepsilon} t} \|x(0)\|, \tag{2.5}$$

implying that every $u(x) \in K_c(x)$ exponentially stabilizes the system (2.1) to the zero dynamics, $Z$. In addition, the control value of minimum norm can be determined as follow:

$$m(x) = \operatorname{argmin}\{\|u\| : u \in K_c(x)\}, \tag{2.6}$$

which is known as the *min-norm* controller [12]. In particular, (2.6) can be presented as a quadratic program (QP):

$$m(x) = \operatorname*{argmin}_{u \in U} u^T u \tag{2.7}$$

$$s.t. \quad \psi_0(x) + \psi_1^T(x)u \le 0,$$

where

$$\psi_0(x) = L_f V(x) + \frac{c_3}{\varepsilon} V(x), \tag{2.8}$$

$$\psi_1(x) = L_g V(x)^T. \tag{2.9}$$

Moreover, the solution to the QP (2.7) can be stated in closed form as:

$$m(x) = \begin{cases} -\frac{\psi_0(x)\psi_1(x)}{\psi_1(x)^T \psi_1(x)} & \text{if} \quad \psi_0(x) > 0 \\ 0 & \text{if} \quad \psi_0(x) \le 0 \end{cases}, \tag{2.10}$$

The QP-based control method (2.7) has been applied to robotic locomotion and manipulation [5], experimental robotic walking [3, 13], and adaptive cruise control [4].

## 2.2 Control Barrier Functions

Considering the affine nonlinear control system (2.1) and given a set $\mathcal{C} \subset \mathbb{R}^n$, we determine conditions on functions $B : \mathcal{C} \to \mathbb{R}$ such that solutions to (2.1), with initial condition in $\mathcal{C}$, remain in $\mathcal{C}$ for all time. First, we note that since (2.1) is assumed to be locally Lipschitz, for any initial condition $x_0 \in \mathbb{R}^n$ there exists a maximum time interval $I(x_0) = [0, \tau_{\max})$ such that $x(t)$ is the unique solution to (2.1) on $I(x_0)$; in the case when $f$ is forward complete, $\tau_{\max} = \infty$. The set $\mathcal{C}$ is *forward invariant* if for

every $x \in \mathcal{C}$, $x(t) \in \mathcal{C}$ for all $t \in I(x)$.

For simplicity, suppose that

$$\mathcal{C} = \{x \in \mathbb{R}^n : h(x) \geq k(x)\}, \tag{2.11}$$

$$\partial\mathcal{C} = \{x \in \mathbb{R}^n : h(x) = k(x)\}, \tag{2.12}$$

$$\text{Int}(\mathcal{C}) = \{x \in \mathbb{R}^n : h(x) > k(x)\}, \tag{2.13}$$

where $h(x) : \mathbb{R}^n \to \mathbb{R}$ represents a safety constraint, and $k(x) : \mathbb{R}^n \to \mathbb{R}$ is a boundary that $h(x)$ cannot violate. In addition, we can define $z_1(x) = h(x) - k(x)$, from which it follows that

$$\mathcal{C} = \{x \in \mathbb{R}^n : z_1(x) \geq 0\}, \tag{2.14}$$

$$\partial\mathcal{C} = \{x \in \mathbb{R}^n : z_1(x) = 0\}, \tag{2.15}$$

$$\text{Int}(\mathcal{C}) = \{x \in \mathbb{R}^n : z_1(x) > 0\}. \tag{2.16}$$

It is noted that (2.14)-(2.16) are identical to (2.11)-(2.13).

To ensure that $x$ stays in the set $\mathcal{C}$, we have the following definition:

**Definition 2.** *Let $\mathcal{C} \subset \mathbb{R}^n$ be defined by (2.11)-(2.13) for a continuously differentiable function $h : \mathbb{R}^n \to \mathbb{R}$, then a function $B : \mathcal{C} \to \mathbb{R}$ is a **control barrier function (CBF)** if there exists class $\mathcal{K}$ functions $\alpha_1, \alpha_2$ and $0 < \gamma$ such that*

$$\frac{1}{\alpha_1(z_1(x))} \leq B(x) \leq \frac{1}{\alpha_2(z_1(x))}, \tag{2.17}$$

$$\inf_{u \in U}[L_f B(z) + L_g B(x)u - \frac{\gamma}{B(x)}] \leq 0. \tag{2.18}$$

Using the definition of a CBF, we can define a set

$$K_B(x) =$$

$$\{u \in U : L_f B(x) + L_g B(x)u - \frac{\gamma}{B(x)} \leq 0\}, \tag{2.19}$$

which yields the following result from [4]:

**Theorem 1.** *Given a set* $\mathcal{C} \subset \mathbb{R}^n$ *defined by* (2.14)-(2.16) *with associated barrier function B, any Lipschitz continuous controller* $u(x) \in K_B(x)$ *for the system* (2.1) *renders the set* $\mathcal{C}$ *forward invariant.*

## 2.3 Combining CLFs and CBFs via QPs

To unify CLFs and CBFs, we can combine (2.7) and (2.19) into a QP as follows:

$$u^*(x) = \underset{\mathbf{u}=\begin{bmatrix} u \\ \delta \end{bmatrix} \in \mathbb{R}^{m+1}}{\operatorname{argmin}} \quad \frac{1}{2}\mathbf{u}^T H(x)\mathbf{u} + F(x)^T \mathbf{u} \tag{2.20}$$

$$s.t. \quad \psi_0(x) + \psi_1^T(x)u \leq \delta, \tag{CLF}$$

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

where $H(x) \in \mathbb{R}^{m+1 \times m+1}$ and $F(x) \in \mathbb{R}^{m+1}$ are arbitrary cost functions that can be chosen based upon desired (state based) weighting of the control inputs, and $\delta \in \mathbb{R}$ is a violation of the CLF constraint, which can be chosen to guarantee a feasible solution to the QP.

The solution to the CLF-CBF-QP is a control $u$ which forces the control objective to be achieved by using CLF constraint while rendering the set $\mathcal{C}$ forward invariant via the CBF constraint. Therefore, following from [29, 4] it follows that:

**Theorem 2.** *Given a set $\mathcal{C} \subseteq \mathbb{R}^n$ defined by (2.11)-(2.13) with $B$ an associated control barrier function, for any positive definite $H(x)$, the control law $u^*(x)$ obtained by solving the QP (2.20) is Lipschitz continuous and renders the set $\mathcal{C}$ forward invariant.*

In practice, if $x \in \text{Int}(\mathcal{C})$ are far away from the boundary, $\partial\mathcal{C}$, the control objective will be achieved exponentially; otherwise, it will be violated depending on how $x$ close to $\partial\mathcal{C}$.

### 2.4 CBFs with Backstepping Method

Considering the QP (2.20), it may not be possible to solve the constraint associated with the control barrier function due to disappearance of the input $u$, i.e. $L_g B = 0$. This happens when $h(x)$ has relative degree greater than one. Therefore, motivated by [34], we introduce backstepping method to CBFs so that the inputs can be explicit. In the following of this section, we prove that CBFs with backstepping method render the set $\mathcal{C}$ forward invariant. The concept is to show a CBF with backstepping method is also a CBF and Theorem 2 applies.

To make inputs explicit, $h(x)$ has to be developed with dynamic extension. Suppose $h(x)$ is an output required to be greater than a function $k(x)$, and has relative degree 2 as illustrated by the following relationship:

$$\phi_1(x) = h(x), \tag{2.21}$$

$$\dot{\phi}_1(x) = \phi_2(x), \tag{2.22}$$

$$\dot{\phi}_2(x, \dot{x}) = L_f L_f h(x) + L_g L_f h(x)u, \tag{2.23}$$

and a set $\mathcal{C}$ is defined by (2.14) to (2.16). Therefore, we can pick a *control barrier*

*function*, $B_1(x)$, so that $B_1(x)$ has the inequalities:

$$\frac{1}{\alpha_{1,1}(z_1(x))} \leq B_1(x) \leq \frac{1}{\alpha_{1,2}(z_1(x))}, \tag{2.24}$$

$$\dot{B}_1(x) \leq \frac{\gamma}{B_1(x)}, \tag{2.25}$$

where $\alpha_{1,1}$ and $\alpha_{1,2}$ are class $\mathcal{K}$ functions, and $z_1(x) = h(x) - k(x)$. In addition, we

let

$$z_2(x) = \phi_2(x) - \xi,$$

where $\xi$ is a stabilizing function we have to design. The time derivative of $B_1(x)$ is

thus given by:

$$\begin{aligned}
\dot{B}_1 &= \frac{dB_1(x)}{dt} \\
&= \frac{dB_1(x)}{dz_1} \dot{z}_1 \\
&= \frac{dB_1(x)}{dz_1} [\phi_2 - \dot{k}(x)] \\
&= \frac{dB_1(x)}{dz_1} [z_2 + \xi - \dot{k}(x)].
\end{aligned}$$

Picking $\xi = \dot{k}(x)$ results in:

$$\dot{B}_1 = \frac{dB_1(x)}{dz_1} z_2. \tag{2.26}$$

Following the backingstepping method [20], we can define another candidate barrier

function with $z_2$

$$B_2(x) = B_1(x) + \frac{1}{2} z_2(x)^2, \tag{2.27}$$

which gives the main result.

**Theorem 3.** *Given a set $\mathcal{C} \subset \mathbb{R}^n$ defined by (2.11) to (2.13), if $h(x)$ defined by (2.21) to (2.23) has relative degree 2, there exists a Lipschitz continuous controller $u(x) \in K_B(x)$, with the control barrier function is defined by (2.27), such that the set $\mathcal{C}$ is forward invariant.*

*Proof.* Reformulating (2.24), we have

$$\alpha_{1,1}^{-1}\left(\frac{1}{B_1(x)}\right) \leq z_1 \leq \alpha_{1,2}^{-1}\left(\frac{1}{B_1(x)}\right), \tag{2.28}$$

where $\alpha^{-1}$ represents the inverse of $\alpha$. Moreover, substituting the reciprocal of (2.24) into (2.28) yields

$$\alpha_{1,1}^{*}(z_1) \leq z_1 \leq \alpha_{1,2}^{*}(z_1,),$$

where

$$\alpha_{1,1}^{*}(z_1) =: \alpha_{1,1}^{-1}\left(\alpha_{1,2}(z_1)\right),$$

$$\alpha_{1,2}^{*}(z_1) =: \alpha_{1,2}^{-1}\left(\alpha_{1,1}(z_1)\right).$$

Since the time derivative of $z_1$ is $z_2$, we can write

$$\dot{\alpha}_{1,1}^{*}(z_1) \leq z_2 \leq \dot{\alpha}_{1,2}^{*}(z_1). \tag{2.29}$$

It is noted that $\alpha_{1,1}^{*}(z_1)$ and $\alpha_{1,2}^{*}(z_1)$ are class $\mathcal{K}$ functions [20]. It is also noted that $\dot{\alpha}_{1,1}^{*}$ and $\dot{\alpha}_{1,2}^{*}$ are greater than zero because class $\mathcal{K}$ functions must be strictly

increasing. Utilizing (2.24), (2.27) and (2.29) yields:

$$\frac{1}{\alpha_{2,1}(z_1)} \leq B_2(x) \leq \frac{1}{\alpha_{2,2}(z_1)},$$

where

$$\alpha_{2,1}(z_1) = \frac{\alpha_{1,1}(z_1)}{1 + \alpha_{1,1}(z_1)\frac{1}{2}(\dot{\alpha}_{1,1}^*(z_1))^2},$$

$$\alpha_{2,2}(z_1) = \frac{\alpha_{1,2}(z_1)}{1 + \alpha_{1,2}(z_1)\frac{1}{2}(\dot{\alpha}_{1,2}^*(z_1))^2}.$$

Since $\alpha_{2,1}(0) = \alpha_{2,2}(0) = 0$, $\alpha_{2,1}(z_1) > 0$ and $\alpha_{2,2}(z_1) > 0$ for every $z_1 \neq 0$, $\alpha_{2,1}$ and $\alpha_{2,2}$ are positive definite functions. Two new bounds are thus established according to Lemma 4.3 in [20]

$$\alpha_{2,1}^-(z_1) \leq \alpha_{2,1}(z_1) \leq \alpha_{2,1}^+(z_1), \tag{2.30}$$

$$\alpha_{2,2}^-(z_1) \leq \alpha_{2,2}(z_1) \leq \alpha_{2,2}^+(z_1), \tag{2.31}$$

where $\alpha_{2,1}^-(z_1)$, $\alpha_{2,1}^+(z_1)$, $\alpha_{2,2}^-(z_1)$, and $\alpha_{2,2}^+(z_1)$ are Class $\mathcal{K}$ functions. Letting $\alpha_{2,1}^*(z_1) = \alpha_{2,1}^+(z_1)$ and $\alpha_{2,2}^*(z_1) = \alpha_{2,2}^-(z_1)$, it follows:

$$\frac{1}{\alpha_{2,1}^*(z_1)} \leq B_2(x) \leq \frac{1}{\alpha_{2,2}^*(z_1)}, \tag{2.32}$$

On the other hand, take the time derivative of $B_2$, which becomes

$$\dot{B}_2 = \dot{B}_1 + z_2 \dot{z}_2$$
$$= \frac{dB_1(x)}{dz_1} z_2 + z_2(L_f L_f h(x) + L_g L_f h(x)u - \dot{\xi}),$$

14

where $\dot{\xi} = \ddot{k}(x)$. Therefore, we have the following condition

$$\frac{dB_1(x)}{dz_1}z_2 + z_2(L_fL_fh(x) + L_gL_fh(x)u - \dot{\xi}) \leq \frac{\gamma}{B_2(x)}. \qquad (2.33)$$

Rearranging (2.33) to obtain the control law,

$$u \leq \frac{1}{z_2L_gL_fh(x)}\left(\frac{\gamma}{B_2(x)} - \frac{dB_1(x)}{dz_1}z_2\right) - \frac{L_fL_fh(x) - \dot{\xi}}{L_gL_fh(x)}. \qquad (2.34)$$

$L_gL_fh(x)$ is non-singular because $h(x)$ has relative degree two by assumption, so the input $u$ is guaranteed to be obtained. According to Definition 2, $B_2$ is a *control barrier function* because (2.32) and (2.33) hold. Finally, based on Corollary 1 in [4], any Lipschitz continuous control input $u$ satisfying (2.33) renders the set $\mathcal{C}$ forward invariant. $\qquad \square$

Having established Theorem 3, the CBF can be extended to a more general case. Suppose $h(x)$ is an output which desires to be greater than a boundary $k(x)$ and has relative degree $n$ as following relationship:

$$\begin{aligned}
\phi_1(x) &= h(x), \\
\dot{\phi}_1(x) &= \phi_2(x), \\
\dot{\phi}_2(x) &= \phi_3(x), \\
&\vdots \\
\dot{\phi}_n(x, \dot{x}) &= L_f^n h(x) + L_g L_f^{n-1} h(x)u,
\end{aligned} \qquad (2.35)$$

where $n \geq 2$. Following the backstepping methodology, and motivated by the control barrier function considered in (2.27), define the following control barrier function

candidate:

$$B_n(x) = B_1(x) + \frac{1}{2}\sum_{i=2}^{n} z_i^2, \tag{2.36}$$

where

$$z_1 = h(x) - k(x),$$

$$z_i = \phi_i - \xi_{i-1} \quad \text{for} \quad i \geq 2,$$

and the stabilizing functions given by:

$$\xi_1 = \dot{k}(x),$$

$$\xi_2 = -\frac{dB_1(x)}{dz_1} + \dot{\xi}_1,$$

$$\xi_i = -z_{i-1} + \dot{\xi}_{i-1} \quad \text{for} \quad i \geq 3.$$

The derivative of $B_n(x)$ thus can be represented as

$$\dot{B}_n(x) = z_{n-1}z_n + z_n\left(L_f^n h(x) + L_g L_f^{n-1} h(x)u - \dot{\xi}_{n-1}\right),$$

so the control law can be determined through the inequality:

$$u \leq \frac{1}{z_n L_g L_f^{n-1} h(x)}\left(\frac{\gamma}{B_n(x)} - z_{n-1}z_n\right) - \frac{L_f^n h(x) - \dot{\xi}_{n-1}}{L_g L_f^{n-1} h(x)}, \tag{2.37}$$

where $L_g L_f^{n-1} h(x)$ is non-singular because $h(x)$ has relative degree $n$ by assumption, and $u$ is guaranteed to be obtained.

Since $B_2(x) = B_1(x) + \frac{1}{2}z_2^2$ is a CBF by the proof of Theorem 3, it can similarly be shown that $B_3(x) = B_2(x) + \frac{1}{2}z_3^2$ is also a CBF with the same arguments. Similarly,

by induction, it follows that for a relative degree $n$ output $h(x)$, the function $B_n(x)$ defined by (2.36), is also a CBF. this is summarized in the following theorem:

**Theorem 4.** *Given a set $\mathcal{C} \subset \mathbb{R}^n$ defined by (2.11) to (2.13), if $h(x)$ defined by (2.35) has relative degree $n$, there exists a Lipschitz continuous controller $u(x) \in K_B(x)$, with the control barrier function is defined by (2.36), such that the set $\mathcal{C}$ is forward invariant.*

## 2.5 Example

To show the performance of the CLF-CBF-QP, in this section, we apply the method to a simple nonlinear system and choose constrained outputs, which have relative degree 2.

Consider a nonlinear system:

$$\dot{x}_1 = x_1^2 + x_2,$$

$$\dot{x}_2 = x_1 x_2 + x_1 + (1 + x_1^2)u,$$

where $u$ is the control law we have to design. The goal is for $x_1$ to track the desired trajectory, $y_d = 0.5 \sin t$, and subject to $|x_1| \le 0.4$. Hence, a CLF can be applied to achieve tracking of $y_d$ and CBFs can be used to satisfy the constraints on $x_1$. As $x_1$ has relative degree 2, the backstepping method outlined in the previous section will need to be applied.

### 2.5.1 CLFs

Let $z_1 = x_1 - y_d$ and $z_2 = x_2 - \xi_c$, where $\xi_c$ will be designed, define a Lyapunov function candidate

$$V = \frac{1}{2}z_1^2,$$

17

and take derivative of $V$

$$\dot{V} = z_1 \dot{z}_1$$
$$= z_1(\dot{x}_1 - \dot{y}_d)$$
$$= z_1(x_1^2 + x_2 - \dot{y}_d)$$
$$= z_1(x_1^2 + z_2 + \xi_c - \dot{y}_d).$$

Following the backstepping method, we choose $\xi_c = -x_1^2 + \dot{y}_d - k_c z_1$, where $k_c$ is a positive constant. It follows that $\dot{V} = -k_c z_1^2 + z_1 z_2$, so we define an additional candidate Lyapunov function as

$$V^* = \frac{1}{2}z_1^2 + \frac{1}{2}z_2^2,$$

and we can take the derivative again:

$$\dot{V}^* = -k_c z_1^2 + z_1 z_2 + z_2 \dot{z}_2$$
$$= -k_c z_1^2 + z_1 z_2 + z_2(x_1 x_2 + x_1 + (1 + x_1^2)u - \dot{\xi}_c).$$

Hence, the CLF constraint can be stated as

$$\dot{V}^* \leq -\epsilon_c V^* + \delta_c, \tag{2.38}$$

where $\epsilon_c$ is a positive constant, and $\delta_c$ is the relaxation so that CBFs have priority to be satisfied.

To constrain $|x_1| \leq 0.4$, we can formulate 2 constraints, $x_1 \geq -0.4$ and $x_1 \leq 0.4$, so we let $z_{l,1} = x_1 + 0.4$, $z_{u,1} = -x_1 + 0.4$, $z_{l,2} = x_2 - \xi_l$, and $z_{u,2} = x_2 - \xi_u$. Picking two control barrier functions

$$B_l^* = -\log \frac{z_{l,1}}{1 + z_{l,1}} + \frac{1}{2}z_{l,2}^2,$$

$$B_u^* = -\log \frac{z_{u,1}}{1 + z_{u,1}} + \frac{1}{2}z_{u,2}^2,$$

and designing $\xi_l = \xi_u = -x_1^2$ results in

$$\dot{B}_l^* = -\frac{z_{l,2}}{z_{l,1} + z_{l,1}^2} + z_{l,2}(x_1 x_2 + x_1 + (1 + x_1^2)u - \dot{\xi}_l),$$

$$\dot{B}_u^* = \frac{z_{u,2}}{z_{u,1} + z_{u,1}^2} + z_{u,2}(x_1 x_2 + x_1 + (1 + x_1^2)u - \dot{\xi}_u).$$

Therefore, the CBF constraints can be stated as

$$\dot{B}_l^* \leq \frac{\gamma_l}{B_l^*}, \tag{2.39}$$

$$\dot{B}_u^* \leq \frac{\gamma_u}{B_u^*}. \tag{2.40}$$

Given the CLF constraint (2.38) and CBF constraint (2.39) and (2.40), A QP can be formulated by:

$$u^*(x_1, x_2) = \operatorname*{argmin}_{u, \delta_c} u^T u + p_{CLF}\delta_c$$

$$\text{s.t.} \quad A_{CLF}u \leq b_{CLF},$$

$$A_{CBF}u \leq b_{CBF},$$

where

$$A_{CLF} = z_2(1 + x_1^2),$$

$$b_{CLF} = \delta_c + k_c z_1^2 - z_1 z_2 - z_2(x_1 x_2 + x_1 - \dot{\xi}_c) + \epsilon_c V,$$

$$A_{CBF} = \begin{bmatrix} z_{l,2}(1 + x_1^2) \\ z_{u,2}(1 + x_1^2) \end{bmatrix},$$

$$b_{CBF} = \begin{bmatrix} \frac{z_{l,2}}{z_{l,1}+z_{l,1}^2} - z_{l,2}(x_1 x_2 + x_1 - \dot{\xi}_l + \frac{\gamma_l}{B_l^*}) \\ -\frac{z_{u,2}}{z_{u,1}+z_{u,1}^2} - z_{u,2}(x_1 x_2 + x_1 - \dot{\xi}_u + \frac{\gamma_u}{B_u^*}) \end{bmatrix}.$$

By choosing an appropriate penalty, $p_{CLF}$, the control value can be guaranteed to be obtained through solving the QP.

### 2.5.4  Results

Table 2.1 shows the parameters used in the simulation, and Fig 2.1 shows the simulation results with the initial condition $(x_1, x_2) = (0, 0)$. As seen in Figure 2.1, the output $x_1$ is convergent to the desired output but constrained between -0.4 and 0.4.

| Parameter | Value | Unit |
|:---:|:---:|:---:|
| $p_{CLF}$ | $10^5$ | Unitless |
| $k_c$ | 10 | Unitless |
| $\epsilon_c$ | 2 | Unitless |
| $\gamma_l$ | 1 | Unitless |
| $\gamma_u$ | 1 | Unitless |

Table 2.1: The parameters used in the simulation.
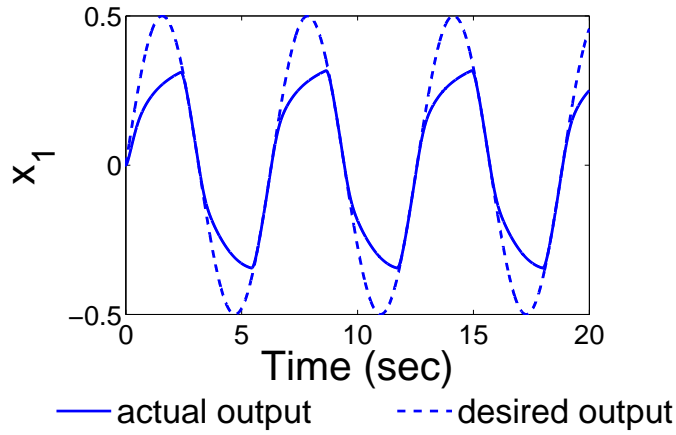


(a) Outputs
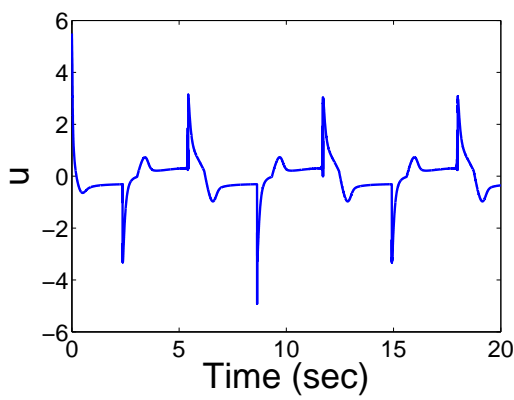


(b) Torque



(c) Barriers

Figure 2.1: Simulation results from a simple nonlinear system example including the output, input, and barrier functions.

# 3. BIPEDAL ROBOT MODEL

The main goal of this research is to apply control barrier functions to the control of bipedal robots. In this chapter, we therefore review the preliminaries in modeling that will set the stage for the introduction of control barrier functions in the context of robotic walking. We introduce a mathematical model of a 7-link bipedal robot 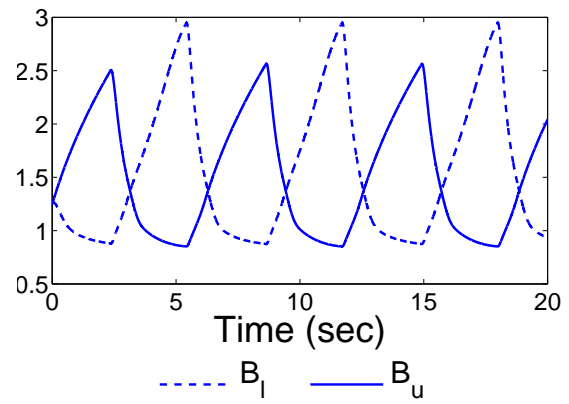AMBER2 [26, 41], which is a fully actuated system because there are 6 actuators, shown in Figure 3.1. In addition, We consider the AMBER2 as a hybrid system, which contains continuous dynamics and discrete dynamics. The discrete dynamics shows up only when an impact happens [14], and the states will be reset through a reset map.

## 3.1 Hybrid System

A walking robot can be modeled as a hybrid control system which is a tuple [7, 1],

$$\mathscr{HC} = (\mathcal{D}, U, S, \Delta, f, g), \tag{3.1}$$

where

- $\mathcal{D}$ is the domain with $\mathcal{D} \subseteq R^n$ a smooth submanifold of the state space $\mathbb{R}^n$,

- $U \subseteq \mathbb{R}^m$ is the set of admissible controls,

- $S \subset \mathcal{D}$ is a proper subset of $\mathcal{D}$ called the guard or switching surface,

- $\Delta{:}S \to X$ is a smooth map called the reset map,

- $(f, g)$ is a control system on $\mathcal{D}$, i.e., in coordinates: $\dot{x} = f(x) + g(x)u$

22
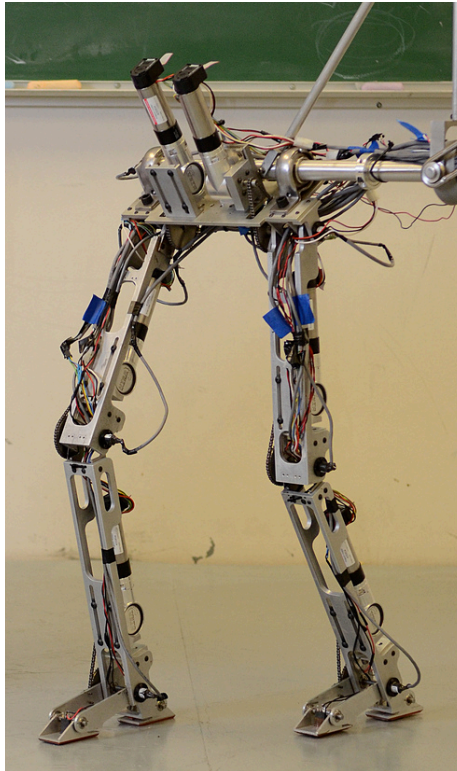
Figure 3.1: The bipedal robot AMBER2 that serves as the basis for simulation results demonstrating the formal results presented.
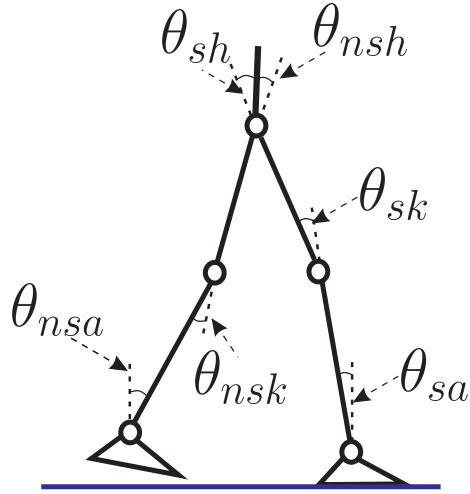
Figure 3.2: The configuration space of AMBER2 [26].

For AMBER2, which has 7 links (2 feet, 2 calves, 2 thighs and a torso.), the configuration space $\mathcal{Q}$ can be defined by:

$$q = (\theta_{sa}, \theta_{sk}, \theta_{sh}, \theta_{nsh}, \theta_{nsk}, \theta_{nsa}), \tag{3.2}$$

where $\theta_{sa}, \theta_{sk}, \theta_{sh}, \theta_{nsh}, \theta_{nsk},$ and $\theta_{nsa}$ represent the joint angles of stance ankle, stance knee, stance hip, non-stance hip, non-stance knee, and non-stance ankle. The configuration space is shown in Figure 3.2.

According to the statement above, AMBER2 can be modeled as a hybrid control system, $\mathscr{HC}$. $\mathcal{D} \subset T\mathcal{Q}$ is the domain given by the constraint $h_R(\theta) \geq 0$, where $h_R$ is the height of the non-stance foot, $U \subset \mathbb{R}^6$ is the set of admissible controls, $S \subset \mathcal{D}$ is the guard, and $\Delta$ is the reset map which changes the velocity at the impact.

## 3.2   Continuous Dynamics

Given the configuration $q$, the mass, length and inertia properties of each link of the robot, the Lagrangian, $L : TQ \to \mathbb{R}$, can be computed in the form of the kinetic minus potential energy as follows:

$$L(q, \dot{q}) = \frac{1}{2} \dot{q}^T D(q) \dot{q} - v(q),$$

which yields the equations of motion for AMBER2 in the form of a set of first order ODEs:

$$D(q)\ddot{q} + H(q, \dot{q}) = B(q)u, \tag{3.3}$$

where $D$ is the inertia matrix, $H$ is a vector containing the Coriolis and gravity terms, and $B \in \mathbb{R}^{6 \times 6}$ is the actuation matrix which determines the way in which the torque inputs, $u \in \mathbb{R}^6$ actuate the system. Moreover, the equations of motion can be converted to the affine control system $(f, g)$:

$$f(q, \dot{q}) = \begin{bmatrix} \dot{q} \\ -D^{-1}(q)H(q, \dot{q}) \end{bmatrix}, \quad g(q) = \begin{bmatrix} \mathbf{0} \\ D^{-1}(q)B(q) \end{bmatrix},$$

where $\mathbf{0} \in \mathbb{R}^{6 \times 6}$ is a matrix of zeros.

## 3.3   Domain and Guard

The allowable configurations, i.e. the domain of the AMBER2 model, are ones in which the height of the non-stance foot is on-or-above the ground. In particular, the domain $\mathcal{D}$ can be defined by:

$$\mathcal{D} = \{(q, \dot{q}) \in TQ : h_R(q) \geq 0\} \tag{3.4}$$

The guard is the boundary of the domain with another constraint, which requires that the non-stance foot velocity to be negative. Therefore, the guard can be defined by:

$$S = \{(q, \dot{q}) \in T\mathcal{Q} : h_R(q) = 0 \text{ and } dh_R(q)\dot{q} < 0\}, \tag{3.5}$$

where $dh_R(q)$ is the Jacobian of $h_R(q)$ at $q$.

## 3.4 Discrete Dynamics

The discrete dynamics of AMBER2 describes the change of the states, i.e. the angles and angular velocities, after the non-stance foot impacts the ground. Hence, the reset map $\Delta$ is defined by:

$$\Delta : S \rightarrow \mathcal{D}, \quad \Delta(q, \dot{q}) = \begin{bmatrix} \Delta_q q \\ \Delta_{\dot{q}}(q)\dot{q} \end{bmatrix}, \tag{3.6}$$

where $\Delta_q$ is a relabeling matrix which switches the stance and non-stance leg at impact, e.g. non-stance foot to stance foot. $\Delta_{\dot{q}}$ determines the change in velocity at impact; more detail about the computation can be found in [7] and[15].

# 4. HUMAN-INSPIRED CONTROL

Human-inspired control [7, 1, 23] is the methodology used to achieve robotic walking on bipedal robots. By observing human walking data [6, 7], Canonical Walking Function (CWF), which is the specific function of time and joint angle, is picked as the reference behavior, i.e., the actual output. Based on the CWFs, human-inspired outputs can be determined, which in turn yield outputs (or virtual constraints) that can be driven to zero through the use of control Lyapunov functions (CLFs).

In order to achieve human-like walking, human walking data should be considered first. By observing the human output data, which is shown in Fig. 4.1, hip position is a linear function of time $,\delta p_{hip}^H(t) = v_{hip}t$, and other outputs can be considered as the solution of a mass-spring-damper system. Therefore, the canonical walking functions can be defined by:

$$y_H(t, \alpha) = e^{-\alpha_4 t}(\alpha_1 \cos(\alpha_2 t) + \alpha_3 \sin(\alpha_2 t)) + \alpha_5, \tag{4.1}$$

where $\alpha = (\alpha_1, \alpha_2, \alpha_3, \alpha_4, \alpha_5)$, so by picking up different $\alpha$, each CWF can fit the human data very well.

According to the human outputs produced by CWFs, which are desired outputs, the actual outputs for AMBER2 are defined, so that AMBER2 has the same walking behavior as a human being. With the goal of controlling the velocity of AMBER2, the actual and desired relative degree 1 outputs can be defined by:

$$y_{a,1}(q, \dot{q}) = \delta \dot{p}_{hip}(q, \dot{q}) = d\delta p_{hip}(q)\dot{q}, \ y_{d,1} = v_{hip} \tag{4.2}$$
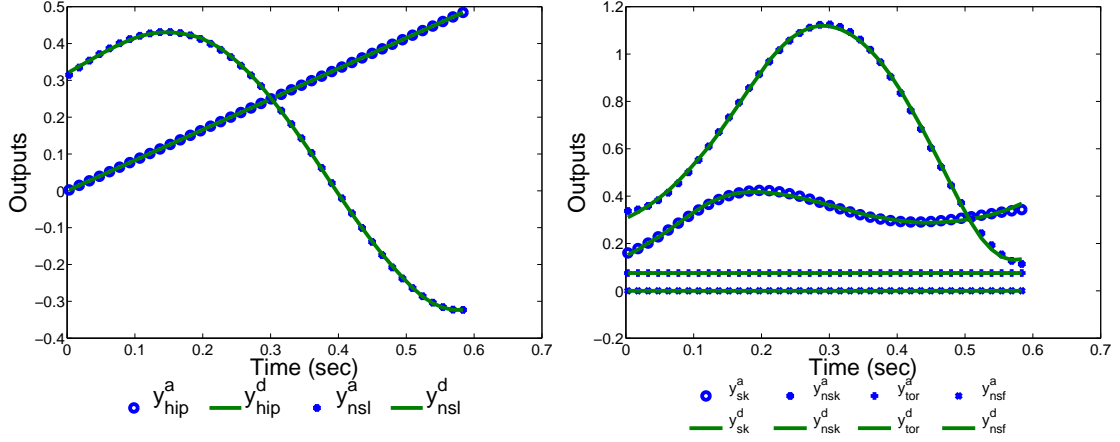
Figure 4.1: The human output data and the canonical walking function fits for each subjects.

where $\delta p_{hip}(q)$ is the linearized position of the hip, which is given by:

$$\delta p_{hip}(q) = L_c(-\theta_{sa}) + L_t(-\theta_{sa} - \theta_{sk}), \tag{4.3}$$

where $L_c$ is the length of the calf and $L_t$ is the length of the thigh. Moreover, the actual linearized relative degree 2 outputs and desired relative degree 2 outputs also can be defined by:

$$
y_{a,2}(q) = \begin{bmatrix} \theta_{sk} \\ \theta_{nsk} \\ \delta m_{nsl}(q) \\ \theta_{tor}(q) \\ \theta_{nsf}(q) \end{bmatrix}, \; y_{d,2}(t,\alpha) = \begin{bmatrix} y_H(t,\alpha_{sk}) \\ y_H(t,\alpha_{nsk}) \\ y_H(t,\alpha_{nsl}) \\ y_H(t,\alpha_{tor}) \\ y_H(t,\alpha_{nsf}) \end{bmatrix}, \tag{4.4}
$$

where $\alpha = \{v_{hip}, \alpha_{sk}, \alpha_{nsk}, \alpha_{nsl}, \alpha_{tor}, \alpha_{nsf}\} \in \mathbb{R}^{26}$, and $\delta m_{nsl}(q)$ is the linearization of

28
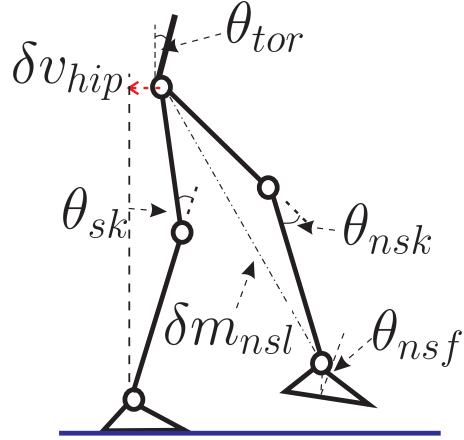
Figure 4.2: The outputs of AMBER2 [26].

the slope of the non-stance leg:

$$\delta m_{nsl}(q) = -\theta_{sf} - \theta_{sk} - \theta_{sh} + \theta_{nsh} + \frac{L_c}{L_c + L_t}\theta_{nsk}, \tag{4.5}$$

$\theta_{tor}(q)$ is the angle of the torso from vertical:

$$\theta_{tor}(q) = \theta_{sf} + \theta_{sk} + \theta_{sh}, \tag{4.6}$$

$\theta_{nsf}$ is the angle of the non-stance foot from vertical:

$$\theta_{nsf} = \theta_{sf} + \theta_{sk} + \theta_{sh} - \theta_{nsh} - \theta_{nsk} - \theta_{nsf}. \tag{4.7}$$

Fig. 4.2 shows the detail of the outputs of AMBER2.

In order to simplify the complexity of the walking, the desired angle of the non-stance foot from vertical will be considered as zero; that is, the flat foot surface will be always parallel with the ground. In particular, $\alpha_{nsf} = \mathbf{0}$, which results $y_H(t, \alpha_{nsf}) = 0$.

In order to create an autonomous controller, which is more robust than a time-based controller, the state-based parameterization of time can be defined by:

$$\tau(q) = \frac{\delta p_{hip}(q) - \delta p_{hip}(q^+)}{v_{hip}},$$ (4.8)

where $\delta p_{hip}(q^+)$ is the linearized hip position at the beginning of a step. Therefore, the time base can be converted to state base, which means $t$ can be converted to $\tau(q)$, which renders the desired outputs as $y_{d,2}(t, \alpha) = y_{d,2}(\tau(q), \alpha)$.

Based on the outputs defined above, the human-inspired outputs is defined by:

$$y_1(q, \dot{q}, v_{hip}) = y_{a,1}(q, \dot{q}) - v_{hip},$$ (4.9)

$$y_2(q, \alpha) = y_{a,2}(q) - y_{d,2}(\tau(q), \alpha),$$ (4.10)

which means the differences between actual and desired outputs. It is important to note that the parameters $\alpha$ of $y_{d,2}$ are typically chosen through nonlinear optimization methods to yield *hybrid zero dynamics* and, thereby, guarantee a stable walking gait [15]. In this thesis, we will instead choose $\alpha$ to be parameters obtained directly from human data [1] and utilize control barrier functions to achieve robotic walking.

## 4.1 Control Lyapunov Functions and Quadratic Programs

With the goal of driving $y_1 \to 0$ and $y_2 \to 0$, utilizing the method from [3], we differentiate the relative degree 1 output once and relative degree 2 output twice:

$$\begin{bmatrix} \dot{y}_1 \\ \ddot{y}_2 \end{bmatrix} = \underbrace{\begin{bmatrix} L_f y_1(q, \dot{q}) \\ L_f^2 y_2(q, \dot{q}) \end{bmatrix}}_{L_f} + \underbrace{\begin{bmatrix} L_g y_1(q, \dot{q}) \\ L_g L_f y_2(q, \dot{q}) \end{bmatrix}}_{A} u,$$ (4.11)

and pick

$$u = A^{-1}(-L_f + \mu), \tag{4.12}$$

for some $\mu \in \mathbb{R}^6$. Combining (4.11) and (4.12) results

$$\begin{bmatrix} \dot{y}_1 \\ \ddot{y}_2 \end{bmatrix} = \mu. \tag{4.13}$$

By defining $\eta = (y_1, y_2, \dot{y}_2) \in \mathbb{R}^{11}$, (4.13) can be converted to a linear control system:

$$\dot{\eta} = \underbrace{\begin{bmatrix} 0 & 0 \\ 0 & I \\ 0 & 0 \end{bmatrix}}_{F} \eta + \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 0 \\ 0 & I \end{bmatrix}}_{G} \mu. \tag{4.14}$$

Therefore, we can formulate the continuous time algebraic Riccati equation(CARE):

$$F^T P + PF - PGG^T P + Q = 0, \tag{4.15}$$

where $Q = Q^T > 0$ and $P = P^T > 0$ is the solution, then we can use $P$ to construct an rapidly exponentially stabilizing control Lyapunov function (RES-CLF) that can be used to stabilize the dynamics (4.14). More detail can be found in [3].

By defining $V(\eta) = \eta^T P \eta$, it follows that

$$\dot{V}(\eta) = L_f V(\eta) + L_g V(\eta)\mu, \tag{4.16}$$

where

$$L_f V(\eta) = \eta^T (F^T P + PF)\eta,$$
$$L_g V(\eta) = 2\eta^T PG. \tag{4.17}$$

To stabilize $\eta$ to zero, we have to find $\mu$ such that:

$$\dot{V}(\eta) \leq -\frac{\gamma}{\varepsilon}V(\eta), \tag{4.18}$$

or in another form:

$$L_fV(\eta) + L_gV(\eta)\mu \leq -\frac{\gamma}{\varepsilon}V(\eta), \tag{4.19}$$

for some $\gamma > 0$. Therefore, a quadratic program(QP) can be formulated to search the optimal $\mu$:

$$m(\eta) = \underset{\mu \in \mathbb{R}^6}{argmin} \quad \mu^T\mu \tag{4.20}$$

$$\text{s.t.} \quad \psi_0(\eta) + \psi_1^T(\eta)\mu \leq 0, \tag{CLF}$$

where

$$\psi_0(\eta) = L_fV(\eta) + \frac{\gamma}{\varepsilon}V(\eta),$$
$$\psi_1(\eta) = L_gV(\eta)^T. \tag{4.21}$$

It is important to note that the solution of the QP is basically the same as the min-norm controller [12], which can be presented in close form by:

$$m(\eta) = \begin{cases} -\frac{\psi_0(\eta)\psi_1(\eta)}{\psi_1(\eta)^T\psi_1(\eta)} & \text{if } \psi_0(\eta) > 0 \\ 0 & \text{if } \psi_0(\eta) \leq 0 \end{cases}. \tag{4.22}$$

Therefore, the control low based on the QP is obtained:

$$u(q, \dot{q}) = A^{-1}(q, \dot{q})(-L_f(q, \dot{q}) + m(q, \dot{q})), \tag{4.23}$$

where $m(q, \dot{q})$ can be expressed in terms of $(q, \dot{q})$ because $\eta$ is a function of $(q, \dot{q})$. To

make $u$ explicit, we can combine (4.20) with (4.12), and it follows that

$$m(q, \dot{q}) = \underset{u \in \mathbb{R}^6}{argmin} \quad u^T A^T A u + 2 L_f^T A u \tag{4.24}$$

$$\text{s.t.} \quad \psi_0(q, \dot{q}) + \psi_1^T(q, \dot{q})(Au + L_f) \leq 0. \tag{CLF}$$

The most important advantage of making $u$ explicit in the QP is that any constraints in terms of $u$ can be included directly, such as torque bounds. To obtain a feasible solution from CLF-QP with torque bounds, we have to introduce relaxation for CLF, and it follows that

$$\underset{(\delta, u) \in \mathbb{R}^7}{argmin} \quad p\delta^2 + u^T A^T A u + 2 L_f^T A u \tag{4.25}$$

$$\text{s.t.} \quad \psi_0(q, \dot{q}) + \psi_1^T(q, \dot{q})(Au + L_f) \leq \delta, \tag{CLF}$$

$$u \leq u_{max}, \tag{Max Torque}$$

$$-u \leq u_{max}, \tag{Min Torque}$$

where $p > 0$ is a large number that is chosen to penalize violation ($\delta$) of the CLF constraint, and $u_{max} \in \mathbb{R}^6$ are maximum torques.

Another issue that has to be concerned is the reaction force from the ground, so contact forces, $F \in \mathbb{R}^3$, should be determined. First of all, holonomic constraints are given: $h(q) = \mathbf{0} \in \mathbb{R}^3$ and we can obtain the Jacobian $J_h(q) = \frac{\partial h(q)}{\partial q}$, then the contact forces can be added to dynamics (3.3) and it follows that

$$D(q)\ddot{q} + H(q, \dot{q}) = B(q)u + J_h^T F, \tag{4.26}$$

where $J_h^T F$ projects the contact wrench into joint-space coordinates. To make sure

$F$ are always valid, they have to satisfy following constraints [5]

$$D(q)\ddot{q} + H(q, \dot{q}) = B(q)u + J_h^T F, \tag{4.27}$$

$$\dot{J}_h\dot{q} + J_h\ddot{q} = 0, \tag{4.28}$$

$$-l_h F^{fz} < F^{my} < l_t F^{fz}, \tag{4.29}$$

$$F^{fx} < \mu_k F^{fz}, \tag{4.30}$$

where $\mu_k$ is the coeficient of static friction between AMBER2 and the ground, $l_t$ is the length of the toe, $l_h$ is the length of the heel, $F = (F^{fx}, F^{my}, F^{fz})$ are contact forces on the stance foot, $F^{fx}$ is the horizontal ground reaction force, $F^{fz}$ is the vertical ground reaction force and $F^{my}$ is the ground reaction torque. (4.29) is the constraint named zero moment point (ZMP), which ensures that the stance foot does not rotate during walking so that 2-domain walking can be maintained. (4.30) ensures the stance foot does not slip. Combining $F$ with $u$, we introduce $\bar{u}$ and $\bar{B}(q)$, so (4.26) can be rewritten as:

$$D(q)\ddot{q} + H(q, \dot{q}) = \underbrace{\begin{bmatrix} B & J_h^T \end{bmatrix}}_{\bar{B}} \underbrace{\begin{bmatrix} u \\ F \end{bmatrix}}_{\bar{u}}. \tag{4.31}$$

Moreover, we can substitute (4.28) into (4.31) and add contact constraints into (4.25),

then the QP can be rewritten as

$$\underset{(\delta,\bar{u})\in\mathbb{R}^{10}}{argmin} \quad p\delta^2 + \bar{u}^T\bar{A}^T\bar{A}\bar{u} + 2L_f^T\bar{A}\bar{u} \tag{4.32}$$

$$\text{s.t.} \quad \psi_0(q,\dot{q}) + \psi_1^T(q,\dot{q})(\bar{A}\bar{u} + L_f) \leq \delta, \tag{CLF}$$

$$\dot{J}_h\dot{q} + J_h D(q)^{-1}(\bar{B}\bar{u} - H(q,\dot{q})) = 0, \tag{Constrained Dynamics}$$

$$u \leq u_{max}, \tag{Max Torque}$$

$$-u \leq u_{max}, \tag{Min Torque}$$

$$-l_h F^{fz} < F^{my} < l_t F^{fz}, \tag{ZMP}$$

$$|F^{fx}| < \mu_k F^{fz}, \tag{Friction}$$

where $\bar{A}$ is determined by $\bar{g} = \begin{bmatrix} 0 \\ D^{-1}(q)\bar{B}(q) \end{bmatrix}$. It is important to note that the solution to the QP (4.32) guarantees that the relative degree 1 output and relative degree two outputs converge exponentially while contact forces satisfy the contact constraints.

Since it is hard to find a feasible solution in some cases, we have to separate CLF into three parts [29], i.e. $\psi_1$ for the relative degree 1 output, $\psi_3$ for the non-stance foot, and $\psi_2$ for the rest of the relative degree 2 outputs. By doing so, three CLFs have relaxation separately depending on their own needs. Therefore, (4.32) can be

modified as

$$\underset{(\delta,\bar{u})\in\mathbb{R}^{11}}{argmin} \quad \delta^T p \delta + \bar{u}^T \bar{A}^T \bar{A} \bar{u} + 2L_f^T \bar{A} \bar{u} \tag{4.33}$$

$$\text{s.t.} \quad \psi_{1,0}(q,\dot{q}) + \psi_{1,1}^T(q,\dot{q})(\bar{A}\bar{u} + L_f) \leq \delta_1, \tag{CLF1}$$

$$\psi_{2,0}(q,\dot{q}) + \psi_{2,1}^T(q,\dot{q})(\bar{A}\bar{u} + L_f) \leq \delta_2, \tag{CLF2}$$

$$\psi_{3,0}(q,\dot{q}) + \psi_{3,1}^T(q,\dot{q})(\bar{A}\bar{u} + L_f) \leq \delta_3, \tag{CLF3}$$

$$\dot{J}_h \dot{q} + J_h D(q)^{-1}(\bar{B}\bar{u} - H(q,\dot{q})) = 0, \tag{Constrained Dynamics}$$

$$u \leq u_{max}, \tag{Max Torque}$$

$$-u \leq u_{max}, \tag{Min Torque}$$

$$-l_h F^{fz} < F^{my} < l_t F^{fz}, \tag{ZMP}$$

$$|F^{fx}| < \mu_k F^{fz}, \tag{Friction}$$

where $p = \begin{bmatrix} p_1 & 0 & 0 \\ 0 & p_2 & 0 \\ 0 & 0 & p_3 \end{bmatrix}$ and $\delta = \begin{bmatrix} \delta_1 \\ \delta_2 \\ \delta_3 \end{bmatrix}$.

## 4.2    Control Barrier Functions and Quadratic Programs

Since it is essential to satisfy certain physical constraints, control barrier functions will be utilized to enforce constraints of this form, including: avoiding foot scuffing, higher foot clearance, or other specific physical condition we desire. To do so, we constrain the human-inspired outputs, which include relative degree 1 and relative degree 2 outputs, i.e. hip velocity, non-stance foot height, stance knee, and non-stance slope. In this research, we will formulate a set of control barrier functions with the backstepping method developed and add them as additional constraints to a CLF-QP, so that a stable walking is obtained.

### 4.2.1  Hip velocity

Since CLF1 has relaxation parameter, the hip velocity will not converge to the desired velocity and it even increases to an unreasonable value in some cases. To deal with it, a hip velocity constraint is considered. In particular, we constrain

$$y_{a,1}(q, \dot{q}) \leq V_{max}.$$

Therefore, we can let $z_{vmax} = V_{max} - y_{a,1}(q, \dot{q})$, and formulate a control barrier function

$$B_{vmax} = -\log \frac{z_{vmax}}{1 + z_{vmax}},$$

so the constraint can be presented as

$$\dot{B}_{vmax} \leq \frac{\gamma_{vmax}}{B_{vmax}}, \tag{CBF1}$$

where $\gamma_{vmax}$ is a positive constant.

### 4.2.2  Non-stance foot height boundary

In general, the non-stance foot height is based on the desired outputs, but sometimes the foot height is not high enough to cross an obstacle. Moreover, the angle of the foot strike is too small, so that foot scuffing may occur. To deal with this problem, we can constrain the foot height in a reasonable region by using control barrier functions. However, the foot height has relative degree 2, which can be presented as

a nonlinear system of the form

$$x_{h,1} = h(q), \qquad (4.34)$$

$$\dot{x}_{h,1} = x_{h,2},$$

$$\dot{x}_{h,2} = L_f L_f x_{h,1} + L_g L_f x_{h,1} u,$$

so the backstepping method should be applied, which is similar to the example in Section 2. Before formulating the CBFs, we should set up two state-based functions to build up two boundaries, i.e. $f_l : \mathbb{R} \to \mathbb{R}$ and $f_u : \mathbb{R} \to \mathbb{R}$, which represents lower boundary function and upper boundary function respectively. Here we choose

$$f_l(x_{foot}) = a_l x_{foot}^3 + b_l x_{foot}^2 + c_l x_{foot} + d_l,$$

$$f_u(x_{foot}) = a_u x_{foot}^2 + b_u x_{foot} + c_u,$$

where $a_l$, $b_l$, $c_l$, $d_l$, $a_u$, $b_u$ and $c_u$ are the parameters dependent on the x-position of the non-stance foot before and after the impact, and the desired maximum values of the functions. Figure 4.3 depicts the shapes of two boundaries, which enforce the actual foot height to the ground eventually. Based on the boundary functions above, two CBFs can be set up. Let $z_{fl,1} = h(q) - f_l$, $z_{fu,1} = f_u - h(q)$, $z_{fl,2} = x_{h,2} - \xi_{fl}$, and $z_{fu,2} = x_{h,2} - \xi_{fu}$, we can pick up two control barrier functions

$$B_{fl} = -\log \frac{z_{fl,1}}{1 + z_{fl,1}}, \quad B_{fu} = -\log \frac{z_{fu,1}}{1 + z_{fu,1}},$$
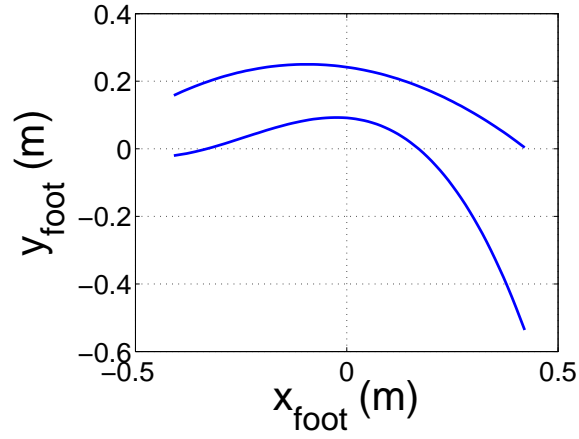
Figure 4.3: Upper and Lower boundaries enforcing $y_{foot}$ to zero in the end.

and take derivatives

$$
\begin{aligned}
\dot{B}_{fl} &= -\frac{\dot{z}_{fl,1}}{z_{fl,1} + z_{fl,1}^2} \\
&= -\frac{x_{h,2} + \xi_{fu} - \dot{f}_l}{z_{fl,1} + z_{fl,1}^2}, \\
\dot{B}_{fu} &= -\frac{\dot{z}_{fu,1}}{z_{fu,1} + z_{fu,1}^2} \\
&= -\frac{\dot{f}_u - x_{h,2} - \xi_{fu}}{z_{fu,1} + z_{fu,1}^2}.
\end{aligned}
$$

Designing $\xi_{fl}$ and $\xi_{fu}$ as $\xi_{fl} = \dot{f}_l$ and $\xi_{fl} = \dot{f}_l$ results

$$
\begin{aligned}
\dot{B}_{fl} &= -\frac{x_{h,2}}{z_{fl,1} + z_{fl,1}^2}, \\
\dot{B}_{fu} &= \frac{x_{h,2}}{z_{fu,1} + z_{fu,1}^2}.
\end{aligned}
$$

39

Next, we add quadratic functions to $B_{fl}$ and $B_{fu}$:

$$B_{fl}^* = -\log \frac{z_{fl,1}}{1 + z_{fl,1}} + \frac{1}{2} z_{fl,2}^2, \quad B_{fu}^* = -\log \frac{z_{fu,1}}{1 + z_{fu,1}} + \frac{1}{2} z_{fu,2}^2,$$

and take derivatives again

$$\begin{aligned}
\dot{B}_{fl}^* &= -\frac{x_{h,2}}{z_{fl,1} + z_{fl,1}^2} + z_{fl,2} \dot{z}_{fl,2} \\
&= -\frac{x_{h,2}}{z_{fl,1} + z_{fl,1}^2} + z_{fl,2}(L_f L_f x_{h,1} + L_g L_f x_{h,1} u - \dot{\xi}_{fl}), \\
\dot{B}_{fu}^* &= \frac{x_{h,2}}{z_{fu,1} + z_{fu,1}^2} + z_{fu,2} \dot{z}_{fu,2} \\
&= \frac{x_{h,2}}{z_{fu,1} + z_{fu,1}^2} + z_{fu,2}(L_f L_f x_{h,1} + L_g L_f x_{h,1} u - \dot{\xi}_{fu}),
\end{aligned}$$

so the control effort is explicit and the CBF constraints are given by

$$\dot{B}_{fl}^* \leq \frac{\gamma_{fl}}{B_{fl}^*} \tag{CBF2}$$

$$\dot{B}_{fu}^* \leq \frac{\gamma_{fu}}{B_{fu}^*}, \tag{CBF3}$$

where $\gamma_{fl}$ and $\gamma_{fu}$ are positive constants.

### 4.2.3 Stance knee angle

In some cases, the stance knee angle may be lower than 0 degrees, which means that the knee angle violates physical realizability conditions, to satisfy other constraints in the QP. Hence, a constraint that keeps the angle to be larger than 0 degrees will be considered. Similar to (4.34), the stance knee can be presented as a

form

$$x_{sk,1} = \theta_{sk}, \tag{4.35}$$

$$\dot{x}_{sk,1} = x_{sk,2},$$

$$\dot{x}_{sk,2} = L_f L_f x_{sk,1} + L_g L_f x_{sk,1} u,$$

so we can follow the same procedure with previous subsection. Letting $z_{sk,1} = x_{sk,1}$ and $z_{sk,2} = x_{sk,2} - \xi_{sk}$, we can pick up a control barrier function

$$B_{sk}^* = -\log \frac{z_{sk,1}}{1 + z_{sk,1}} + \frac{1}{2} z_{sk,2}^2$$

and design $\xi_{sk} = 0$. Taking a derivative of $B_{sk}^*$ results

$$\dot{B}_{sk}^* = -\frac{z_{sk,2}}{z_{sk,1} + z_{sks,1}^2} + z_{sk,2}(L_f L_f x_{sk,1} + L_g L_f x_{sk,1} u),$$

so the constraint is given by

$$\dot{B}_{sk}^* \leq \frac{\gamma_{sk}}{B_{sk}^*}, \tag{CBF4}$$

where $\gamma_{sk}$ is a positive constant.

### 4.2.4   Non-stance slope

Since CLF1 and CLF2 have relaxation, the actual outputs cannot track the desired outputs, which may affect the non-stance foot not moving forward. Hence, non-stance slope constraint is introduced into the QP, so that the slope can stay in a meaningful region and the non-stance foot can keep moving forward. Similarly, the

41

non-stance slope has the following property

$$x_{nsl,1} = \delta m_{nsl}, \tag{4.36}$$

$$\dot{x}_{nsl,1} = x_{nsl,2},$$

$$\dot{x}_{nsl,2} = L_f L_f x_{nsl,1} + L_g L_f x_{nsl,1} u,$$

and should be constrained under the desired slope $y_H(t, \alpha_{nsl})$. In particular, following condition should be satisfied

$$\delta m_{nsl} < y_H(t, \alpha_{nsl}) + k_{nsl} + \frac{1}{1 + c_{nsl}t},$$

where $k_{nsl}$ is a constant that offsets the boundary, $\frac{1}{1+ct}$ is the function which guarantee that $m_{nsl}$ can be smaller than $y_H(t, \alpha_{nsl}) + k_{nsl}$ after the impact. Therefore, we can let $z_{nsl,1} = \frac{1}{1+c_{nsl}t} + k_{nsl} + y_H(t, \alpha_{nsl}) - \delta m_{nsl}$ and $z_{nsl,2} = x_{nsl,2} - \xi_{nsl}$, pick up a CBF

$$B^*_{nsl} = -\log \frac{z_{nsl,1}}{1 + z_{nsl,1}} + \frac{1}{2} z^2_{nsl,2},$$

design $\xi_{nsl} = \dot{y}_H(t, \alpha_{nsl}) - \frac{c_{nsl}}{(1 + c_{nsl}t)^2}$, and take a derivative of $B^*_{nsl}$. It follows

$$\dot{B}^*_{nsl} = \frac{z_{nsl,2}}{z_{nsl,1} + z^2_{nsl,1}} + z_{nsl,2}(L_f L_f x_{nsl,1} + L_g L_f x_{nsl,1} u) - \dot{\xi}_{nsl},$$

so the constraint is given by:

$$\dot{B}^*_{nsl} \leq \frac{\gamma_{nsl}}{B^*_{nsl}}, \tag{CBF5}$$

where $\gamma_{nsl}$ is a positive constant.

### 4.2.5 CLF-CBF-QP

In order to modify the robotic walking behavior given by the QP (4.33), the CLF-CBF-QP is given by

$$\underset{(\delta,\bar{u})\mathbb{R}^{11}}{argmin} \quad \delta^T p \delta + \bar{u}^T \bar{A}^T \bar{A} \bar{u} + 2 L_f^T \bar{A} \bar{u} \tag{4.37}$$

$$\text{s.t.} \quad \psi_{1,0}(q,\dot{q}) + \psi_{1,1}^T(q,\dot{q})(\bar{A}\bar{u} + L_f) \leq \delta_1, \tag{CLF1}$$

$$\psi_{2,0}(q,\dot{q}) + \psi_{2,1}^T(q,\dot{q})(\bar{A}\bar{u} + L_f) \leq \delta_2, \tag{CLF2}$$

$$\psi_{3,0}(q,\dot{q}) + \psi_{3,1}^T(q,\dot{q})(\bar{A}\bar{u} + L_f) \leq \delta_3, \tag{CLF3}$$

$$\dot{J}_h \dot{q} + J_h D(q)^{-1}(\bar{B}\bar{u} - H(q,\dot{q})) = 0, \tag{Constrained Dynamics}$$

$$u \leq u_{max}, \tag{Max Torque}$$

$$-u \leq u_{max}, \tag{Min Torque}$$

$$-l_h F^{fz} < F^{my} < l_t F^{fz}, \tag{ZMP}$$

$$|F^{fx}| < \mu_k F^{fz}, \tag{Friction}$$

$$\text{(CBF1)-(CBF5)}.$$

However, it is not guaranteed to obtain a feasible solution due to the conflict between these physical conditions, so some parameters and constraints should be tuned to feasibly solve (4.37), e.g. $\delta_1$, $\delta_2$, $\delta_3$ , the desired foot height, the desired foot length, and so on. Section 5 will show a stable walking gait through (4.37) with the specific parameters we tested.

# 5. RESULTS AND DISCUSSIONS

This chapter shows the results of 7-link bipedal robotic walking through control Lyapunov functions, control barrier functions, and quadratic program (CLF-CBF-QP), and the stability of the walking is also shown by using Poincare map. In particular, we being with parameters $v_{hip}$ and $\alpha$ of the the human-inspired outputs (4.9) and (4.10) obtained by directly fitting the desired outputs to human data (see [7, 1, 26] for a more complete discussion). This is in contrast to the methods presented in [1], since we do not perform an *a priori* optimization to obtained parameters that guarantee (partial) hybrid zero dynamics [15]. If the control law (4.33) obtained from these outputs is simulated directly with the parameters $v_{hip}$ and $\alpha$ obtained by fitting human data, the robot would stumble and fall. Yet, through the addition of the control barrier functions (CBF1)-(CBF5), the biped displays a stable walking gait (with the proper choice of parameters of the barrier functions); this points to the importance of enforcing physical constraints in the synthesis of robotic walking gaits. It is noted that to obtain a stable walking gait on the model of AMBER2, the parameters in Table 5.1 were chosen.

The initial condition, $(q_0, \dot{q}_0)$, was chosen by assuming that human-inspired outputs are zeros and the non-stance foot is flat on the ground in the beginning. In particular, $q_0$ can be found by solving the inverse kinematics problem outlined in [1]:

$$q_0 = \theta \quad \text{s.t.} \quad \begin{bmatrix} y_2(\Delta\theta) \\ h_R(\theta) \\ \theta_{nsf} \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 0 \\ 0 \end{bmatrix}.$$

| Parameter | Value | Unit |
| --- | --- | --- |
| $p_1$ | 1 | unitless |
| $p_2$ | $10^{-5}$ | unitless |
| $p_3$ | $10^{10}$ | unitless |
| $\mu_k$ | 1000 | unitless |
| $a_l$ | -2.8885 | unitless |
| $b_l$ | -2.0900 | unitless |
| $c_l$ | -0.0991 | unitless |
| $d_l$ | 0.0914 | unitless |
| $a_u$ | -0.9272 | unitless |
| $b_u$ | -0.1748 | unitless |
| $c_u$ | 0.2418 | unitless |
| $c_{nsl}$ | 50 | unitless |
| $k_{nsl}$ | 0.1 | unitless |
| $\gamma_{vmax}$ | 8 | unitless |
| $\gamma_{fl}$ | 30 | unitless |
| $\gamma_{fu}$ | 20 | unitless |
| $\gamma_{sk}$ | 1 | unitless |
| $\gamma_{nsl}$ | 10 | unitless |

Table 5.1: The parameters of the walking which are specifically tuned.

In addition, $\dot{q}_0$ can be determined by

$$\dot{q}_0 = \dot{\theta} \quad \text{s.t.} \quad \dot{\theta} = Y^{-1}(\theta) \begin{bmatrix} v \\ \mathbf{0} \end{bmatrix},$$

where $Y(\theta) = \begin{bmatrix} dy_1^H(\theta) \\ dy_2(\theta) \end{bmatrix}$ is invertible [1]. The initial condition obtained is

$$q_0 = \{0.2707, 0.1569, -0.3525, 0.2056, 0.3370, -0.4675\},$$

$$\dot{q}_0 = \{-0.7807, -0.1932, 0.2514, -0.2094, 0.2446, -1.3356\}.$$

Figure 5.1 and 5.2 show the angles of each output and the torques on each joint, Figure 5.3 shows that the outputs are constrained inside the boundaries we designed via the CBF constraints, and Figure 5.5 shows the gait tiles for 1-step walking; importantly, the actual outputs do not converge to the desired outputs by design, i.e., the use of control barrier functions prevent exact convergence since they enforce physical constraints that dominate the control law. Yet a stable walking gait is still achieved, as evidenced by Figure 5.4 showing the phase plot for 20 steps implies that the walking converges to a limit cycle. The stability can be determined by the Poincare map as well, and the maximum value of eigenvalue is 0.4432 (and hence smaller than 1) indicating stability. It is noted that the torque bounds cannot be added into the QP, since the whole combined constraints are coupled and too strict. On the other words, if torque bounds are added, some constraints will be violated and the solution will be infeasible. Hence, to achieve the physical limitation of the torques from motors, we have to tune the parameters from the QP to get both the reasonable walking gait and torques.
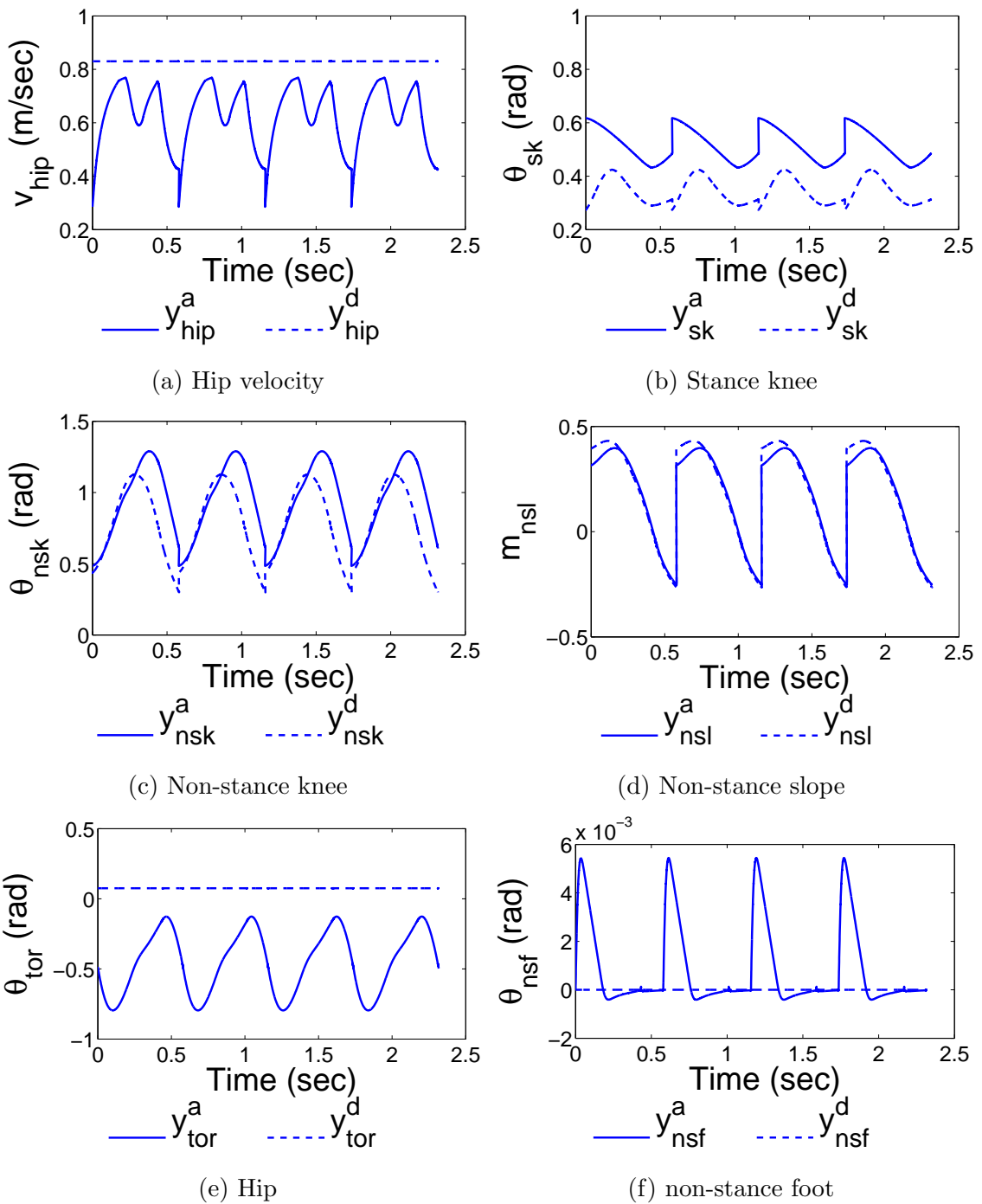
(a) Hip velocity

(b) Stance knee

(c) Non-stance knee

(d) Non-stance slope

(e) Hip

(f) non-stance foot

Figure 5.1: Desired (dot lines) and actual (solid lines) outputs during stable periodic walking.

(a) Stance ankle

(b) Stance knee

(c) Stance torso

(d) Non-stance torso

(e) Non-stance knee

(f) Non-stance ankle
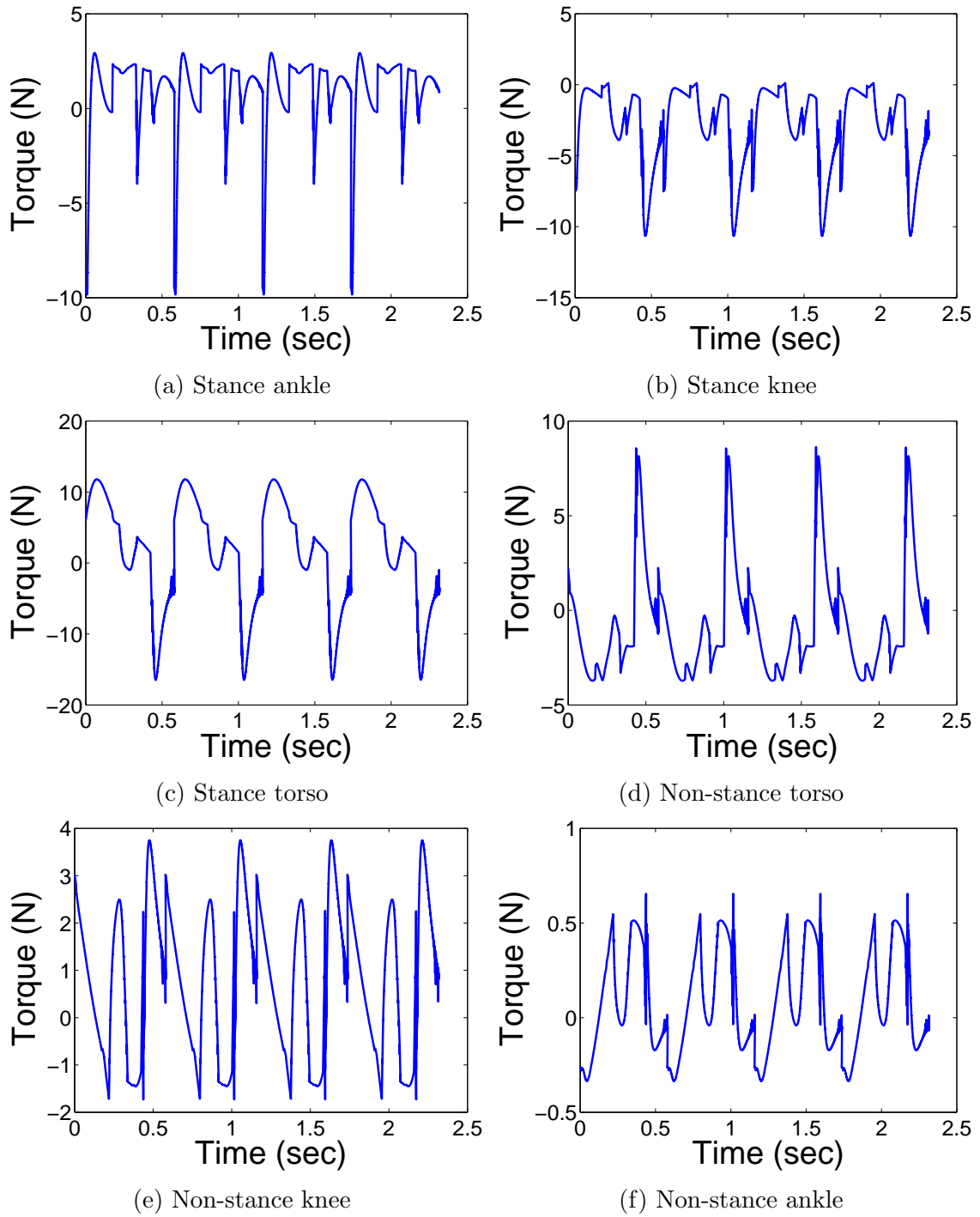
Figure 5.2: Torques on each joint during stable periodic walking.

(a) Hip velocity

(b) Non-stance foot height

(c) Stance torso

(d) Non-stance slope
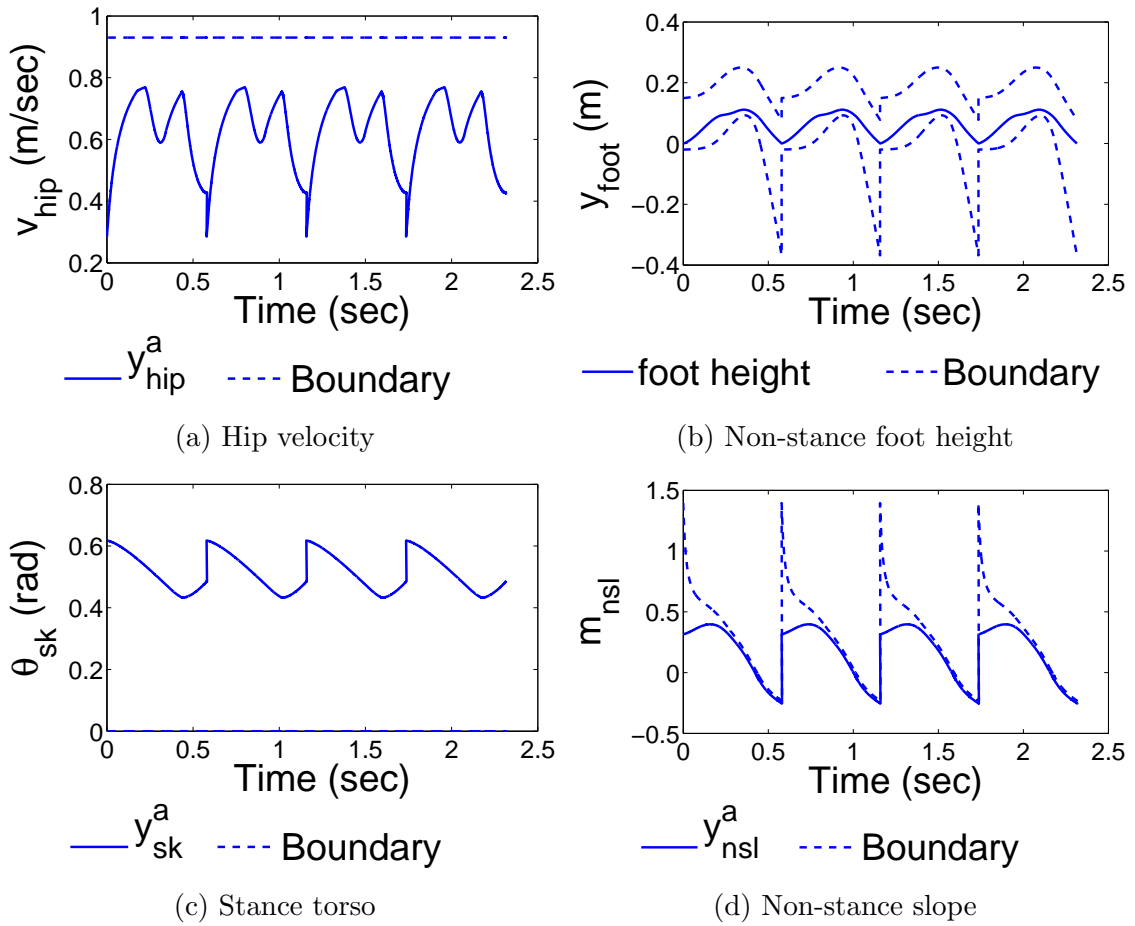
Figure 5.3: Outputs constrained onside the boundaries we desired.

49

(a) Stance ankle

(b) Stance knee

(c) Stance torso

(d) Non-stance torso

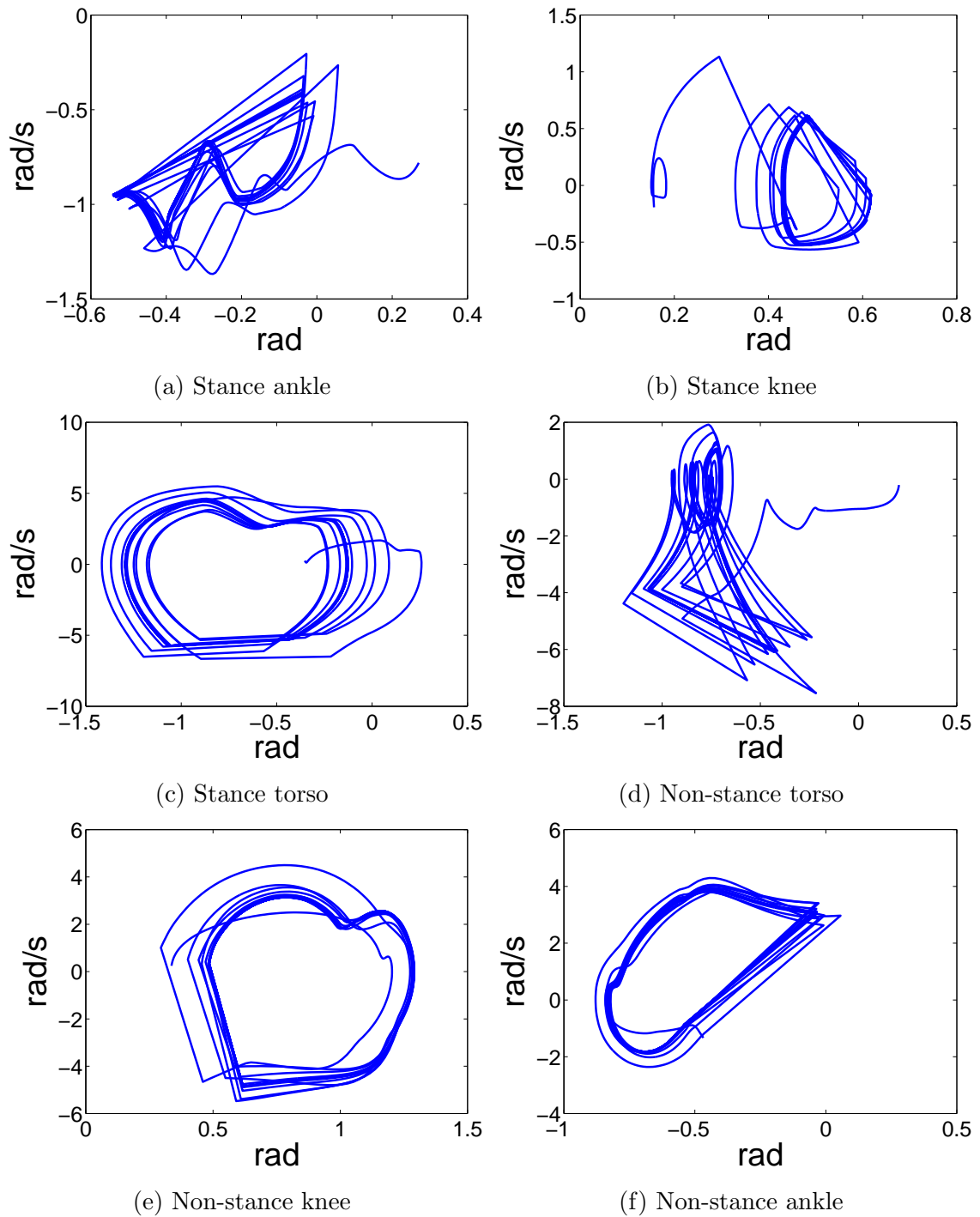(e) Non-stance knee

(f) Non-stance ankle

Figure 5.4: Phase portraits for each joint over 20 steps when started from an initial condition away from the fixed point; convergence to a stable periodic orbit can be seen.
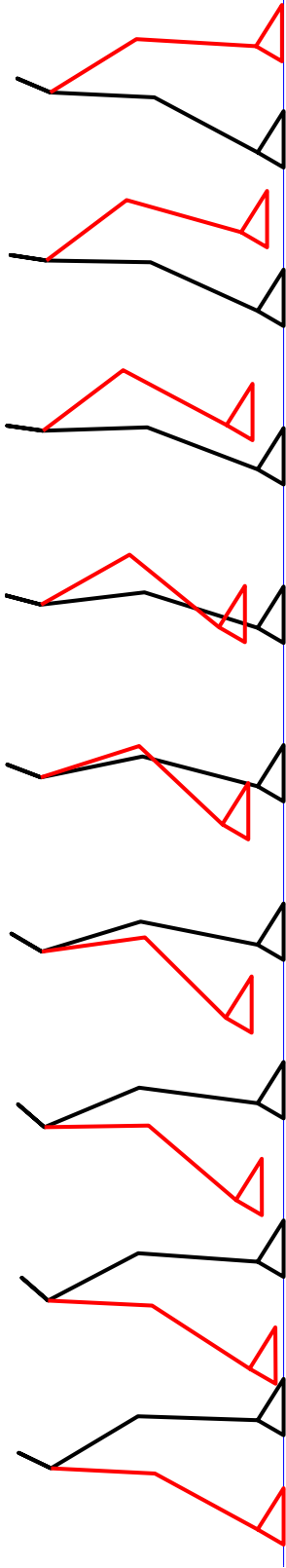
Figure 5.5: Gait tiles for one step of a stable walking gait.

# 6. CONCLUSION

This thesis presented a novel method for constructing control barrier functions through a backstepping inspired approach. In particular, we began by introducing a type of control barrier function that gives the maximum control authority (by allowing $B$ to grow aware from the boundary of the set $\mathcal{C}$); this allowed for the unification of safety constraints and control objectives through CLF-CBF based QPs. Yet the existence of control barrier functions of this form are not guaranteed to exist, i.e., there may not be control inputs that satisfied the required derivative conditions on the CBF. This motivated the main result of this thesis: formal guarantees on the existence of CBFs under assumptions on the relative degree of the function defining $\mathcal{C}$. To demonstrate the usefulness of these results, they were applied to a simple nonlinear and system bipedal robotic walking –which is the main objective of this research. Physical constraints that the robot must satisfy while locomoting were encoded as CBFs and combined with control objectives and torque/force constraints through a single QP based control law. The end result was stable walking in simulation.

## REFERENCES

[1] A. D. Ames. Human-inspired control of bipedal walking robots. *Automatic Control, IEEE Transactions on*, 59(5):1115–1130, 2014.

[2] A. D. Ames, E. Cousineau, and M. Powell. Dynamically stable bipedal robotic walking with nao via human-inspired hybrid zero dynamics. *In Hybrid Systems: Computation and Control*, 2012.

[3] A. D. Ames, K. Galloway, K. Sreenath, and J. W. Grizzle. Rapidly exponentially stabilizing control lyapunov functions and hybrid zero dynamics. *Automatic Control, IEEE Transactions on*, 59(4):876–891, 2014.

[4] A. D. Ames, J. W. Grizzle, and P. Tabuada. Control barrier function based quadratic programs with application to adaptive cruise control. In *IEEE Conference on Decision and Control (CDC)*, Los Angeles, USA, 2014.

[5] A. D. Ames and M. Powell. Towards the unification of locomotion and manipulation through control lyapunov functions and quadratic programs. In *Control of Cyber-Physical Systems*, pages 219–240. Springer, 2013.

[6] A. D. Ames, R. Vasudevan, and R. Bajcsy. Human-data based cost of bipedal robotic walking. In *Proceedings of the 14th International Conference on Hybrid Systems: Computation and Control*, pages 153–162, Berlin, Germany, 2011. ACM.

[7] Aaron D Ames. First steps toward automatically generating bipedal robotic walking from human data. In *Robot Motion and Control 2011*, pages 89–116. Springer, 2012.

[8] Z. Artstein. Stabilization with relaxed controls. *Nonlinear Analysis: Theory, Methods & Applications*, 7(11):1163–1173, 1983.

[9] S. Boyd and L. Vandenberghe. *Convex optimization*. Cambridge University Press, 2009.

[10] S. Collins, A. Ruina, R. Tedrake, and M. Wisse. Efficient bipedal robots based on passive-dynamic walkers. *Science*, 307(5712):1082–1085, 2005.

[11] N. H. El-Farra, P. Mhaskar, and P. D. Christofides. Output feedback control of switched nonlinear systems using multiple lyapunov functions. *Systems & Control Letters*, 54(12):1163–1182, 2005.

[12] R. A. Freeman and P. V. Kokotović. *Robust nonlinear control design: state-space and Lyapunov techniques*. Springer, 2008.

[13] K. Galloway, A. D. Sreenath, K.and Ames, and J. W. Grizzle. Torque saturation in bipedal robotic walking through control lyapunov function based quadratic programs. *arXiv:1302.7314*, 2013.

[14] J. W. Grizzle, G. Abba, and F. Plestan. Asymptotically stable walking for biped robots: Analysis via systems with impulse effects. *Automatic Control, IEEE Transactions on*, 46(1):51–64, 2001.

[15] J. W. Grizzle, C. Chevallereau, A. D. Ames, and R. W. Sinnet. 3D bipedal robotic walking: models, feedback control, and open problems. In *IFAC Symposium on Nonlinear Control Systems*, Bologna, Italy, September 2010.

[16] J. W. Grizzle, C. Chevallereau, and C.-L. Shih. HZD-based control of a five-link underactuated 3D bipedal robot. In *IEEE Conference on Decision and Control*, Cancún, Mexico, December 2008.

[17] A. Hereid, S. Kolathaya, M. S. Jones, J. Van Why, J. W. Hurst, and A. D. Ames. Dynamic multi-domain bipedal walking with atrias through slip based human-inspired control. In *Proceedings of the 17th International Conference on Hybrid Systems: Computation and Control*, pages 263–272, Berlin, Germany, 2014. ACM.

[18] A. Isidori. *Nonlinear control systems*, volume 1. Springer, 1995.

[19] S. Kajita, M. Morisawa, K. Harada, K. Kaneko, F. Kanehiro, K. Fujiwara, and H. Hirukawa. Biped walking pattern generator allowing auxiliary zmp control. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 2993–2999, Beijing, China, 2006. IEEE.

[20] H. K Khalil. *Nonlinear systems*, volume 3. Prentice hall Upper Saddle River, 2002.

[21] P. V. Kokotović. The joy of feedback: nonlinear and adaptive. *IEEE Control Systems Magazine*, 12(3):7–17, 1992.

[22] P. V. Kokotović and M. Arcak. Constructive nonlinear control: a historical perspective. *Automatica*, 37(5):637–662, 2001.

[23] S. Kolathaya and A. D. Ames. Achieving bipedal locomotion on rough terrain through human-inspired control. In *Safety, Security, and Rescue Robotics (SSRR), 2012 IEEE International Symposium on*, pages 1–6, Toyako-cho, Japan, 2012. IEEE.

[24] M. Krstic, I. Kanellakopoulos, and P. V. Kokotović. *Nonlinear and adaptive control design*. Wiley, 1995.

[25] M. KrstiC, J. Sun, and P. V. Kokotović. Robust control of nonlinear systems with input unmodeled dynamics. *Automatic Control, IEEE Transactions on*,

41(6):913–920, 1996.

[26] W. Ma, H. Zhao, S. Kolathaya, and A. D. Ames. Human-inspired walking via unified pd and impedance control. In *2014 IEEE Conference on Robotics and Automation*, Hong Kong, China, 2014.

[27] T. McGeer. Passive dynamic walking. *The International Journal of Robotics Research*, 9(2):62–82, 1990.

[28] T. McGeer. Passive walking with knees. In *Robotics and Automation, 1990. Proceedings., 1990 IEEE International Conference on*, pages 1640–1645, Cincinnati, USA, 1990. IEEE.

[29] B. Morris, M. J. Powell, and A. D. Ames. Sufficient conditions for the lipschitz continuity of qp-based multi-objective control of humanoid robots. In *Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on*, pages 2920–2926, Florence, Italy, 2013. IEEE.

[30] K. B. Ngo, R. Mahony, and Z.P. Jiang. Integrator backstepping using barrier functions for systems with multiple state constraints. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pages 8306–8312, Seville, Spain, 2005. IEEE.

[31] S. Prajna and A. Jadbabaie. Safety verification of hybrid systems using barrier certificates. In *Hybrid Systems: Computation and Control*, pages 477–492. Springer, 2004.

[32] W. J. Schwind. *Spring loaded inverted pendulum running: a plant model*. University of Michigan, 1998.

[33] E. D. Sontag. A lyapunov-like characterization of asymptotic controllability. *SIAM Journal on Control and Optimization*, 21(3):462–471, 1983.

[34] K. P. Tee, S. Sam Ge, and E. H. Tay. Barrier lyapunov functions for the control of output-constrained nonlinear systems. *Automatica*, 45(4):918–927, 2009.

[35] M. Vukobratović and B. Borovac. Zero-moment point—thirty-five years of its life. *Intl. J. of Humanoid Robotics*, 1(1):157–173, March 2005.

[36] M. Vukobratović, B. Borovac, D. Surla, and D. Stokic. *Biped Locomotion*. Springer-Verlag, Berlin, March 1990.

[37] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris. *Feedback Control of Dynamic Bipedal Robot Locomotion*. CRC Press, Boca Raton, June 2007.

[38] Eric R Westervelt, Jessy W Grizzle, and Daniel E Koditschek. Hybrid zero dynamics of planar biped walkers. *Automatic Control, IEEE Transactions on*, 48(1):42–56, 2003.

[39] P. Wieland and F. Allgöwer. Constructive safety using control barrier functions. In *Proceedings of the 7th IFAC Symposium on Nonlinear Control Systems*, pages 462–467, Pretoria, South Africa, 2007.

[40] S. J. Wright and J. Nocedal. *Numerical optimization*, volume 2. Springer New York, 1999.

[41] H Zhao, W Ma, Michael B Zeagler, and Aaron D Ames. Human-inspired multi-contact locomotion with AMBER2. In *International Conference on Cyber-Physical Systems*, Berlin, Germany, 2014.