# Unification of Locomotion Pattern Generation and Control Lyapunov Function-Based Quadratic Programs

Kenneth Y. Chao[1], Matthew J. Powell[2], Aaron D. Ames[2] and Pilwon Hur[1]

***Abstract*— This paper presents a novel method of combining real-time walking pattern generation and constrained nonlinear control to achieve robotic walking under Zero-Moment Point (ZMP) and torque constraints. The proposed method leverages the fact that existing solutions to both walking pattern generation and constrained nonlinear control have been independently constructed as Quadratic Programs (QPs) and that these constructions can be related through an equality constraint on the instantaneous acceleration of the center of mass. Specifically, the proposed method solves a single Quadratic Program which incorporates elements from Model Predictive Control (MPC) based center of mass planning methods and from rapidly exponentially stabilizing control Lyapunov function (RES-CLF) methods. The resulting QP-based controller simultaneously solves for a COM trajectory that satisfies ZMP constraints over a future horizon while also producing joint torques consistent with instantaneous acceleration, torque, ZMP and RES-CLF constraints. The method is developed for simulation and experimental study on a seven-link, planar robot.**

## I. Introduction

Numerical optimization plays an important role in the development of numerous walking control approaches as the mathematics used to model bipedal robot control systems are often constrained, nonlinear, high-dimensional and incorporate impulse effects due to collisions between the robot and the ground. The usage of optimization in the control of robot walking can be categorized into "offline" optimizations which solve for walking gaits before the robot is turned on and "online" optimizations which are solved while the robot is walking. Examples of successful usage of offline nonlinear optimization include the efficient design of the Cornell Ranger [6], control output parameterization establishing (hybrid) system stability through Hybrid Zero Dynamics (HZD) and Human-inspired Control [2], [9], and direct state and input trajectory optimization [12].

On the other end of the spectrum, online numerical optimization – in the form of Quadratic Programs – has become increasingly popular in the control of walking robots due to the fact that some QP-based controllers with affine constraints can be solved in real-time [17] and that the structure of a Quadratic Program is well suited to handle a diverse set of problems in robotic walking. For example, in locomotion pattern generation applications, Quadratic Programs can be used to solve Model Predictive Control problems to obtain center of mass trajectories consistent with Zero Moment Point constraints over a future horizon, as in [7], [10], [15], [18]. In this setting, the QP cost function is often setup to minimize the error between future values of the COM and desired reference values. On the other hand, in the context of nonlinear systems, QPs can be naturally coupled with control Lyapunov functions (CLFs) to form an optimal controller guaranteed to stabilize outputs corresponding to walking [3], [8]. In this setting, the quadratic cost function minimizes actuation effort and the constraints encode instantaneous ZMP and torque limits on the full nonlinear system.

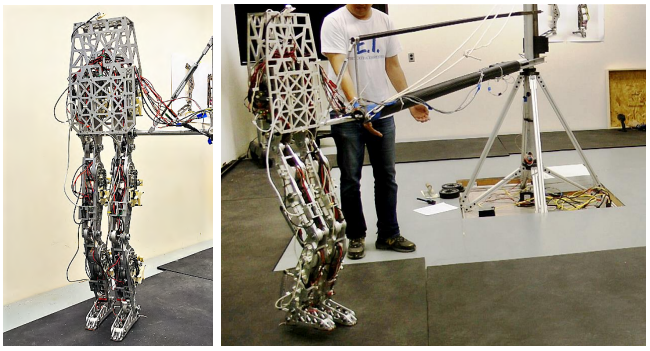

Fig. 1: The human-sized planar bipedal robot: AMBER 3.

Inspired by optimization-based approaches to locomotion, the proposed method combines two QPs: an adaptation of the MPC proposed in [15] for planning center of mass trajectories with the Linear Inverted Pendulum (LIP) model and an adaptation of [4], [5] for locally exponentially stabilizing a control Lyapunov function for the full nonlinear dynamics of the robot. The connection point is an equality constraint imposed on the dynamics of the center of mass which enforces that the instantaneous horizontal COM acceleration is the same in both the nonlinear system and the LIP model. With this bridge in place, the unified QP enjoys the advantages of both QPs: it resolves control actions which locally stabilize nonlinear control system outputs while ensuring that these control actions are consistent with a forward horizon COM plan that satisfies ZMP constraints in the simplified model.

It is important to note that similar combinations of walking pattern generation methods and constrained, local nonlinear control have been proposed before. For example, in [11], the authors propose a similar QP which regulates the ZMP to zero over an infinite horizon using an optimal cost-to-go. In the present paper, however, the proposed controller solves a finite-time horizon MPC problem on the COM trajectory, which allows for both ZMP regulation and the enforcement of constraints on the evolution of the COM.

[1]Kenneth Y. Chao and Pilwon Hur are with the Department of Mechanical Engineering, Texas A&M University, 3123 TAMU, College Station, TX, 77843. {kchao,pilwonhur}@tamu.edu

[2]Matthew J. Powell and Aaron D. Ames are with the Department of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30308. {mpowell35,ames}@gatech.edu

## II. Controlling Robot Locomotion Under ZMP Constraints

The Zero Moment Point (ZMP) is an important concept in the study of balance in robotic [16] and human locomotion [19]. For a legged robot with feet, the condition for dynamic balance (i.e. the robot not tipping over) is that the robot's ZMP lies inside the robot's support polygon. This section describes control methods for walking with ZMP constraints.

### A. ZMP Constraints

As shown in Fig. 2, the ZMP position $x_z$ in the saggital plane can be expressed with the ground reaction normal force $F_z$ and moment $\tau_y$ , e.g. $x_z = -\frac{\tau_y}{F_z}$ in single support. The ZMP constraints for dynamic balance can be described as:

$$a_\square \leq -\tau_y / F_z \leq b_\square \tag{1}$$

where $a_\square \in \{a_{SS}, a_{DS}\}$ and $b_\square \in \{b_{SS}, b_{DS}\}$ encode the largest moment arms of the support polygon in single support or in double support, as shown in Fig. 2. To satisfy *instantaneous* dynamic balance during walking, the inequality (1) on the ground reaction forces (GRFs) needs to be satisfied.

### B. Nonlinear Robot Control System with ZMP Constraints

To achieve walking control with ZMP constraints and minimum control effort, a QP-based nonlinear controller with a Rapidly Exponentially Stabilizing Control Lyapunov Function (RES-CLF) [4] and force-based task [5] is adopted. This controller requires the full constrained dynamics described as the following form:

$$D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \begin{bmatrix} B & J_h^T \end{bmatrix} \begin{bmatrix} u \\ F \end{bmatrix} \triangleq \bar{B}(q)\bar{u} \tag{2}$$

where $q$ is the generalized coordinate, $D(q)$ is inertia matrix, $C(q,\dot{q})$ is Coriolis matrix, $G(q)$ is gravity vector, $J_h$ is Jacobian matrix of the contact constraint $h(q)$, $B$ is the torque distribution matrix, $F$ is the GRF vector ($F = [F_x, F_z, \tau_y]^T$ in the saggital plane) and $u$ is a set of actuator torques. Based on the extended input $\bar{u}$ which includes joint torques and GRFs, instantaneous dynamic balance can be satisfied by solving the following quadratic program:

$$\begin{aligned} \bar{u}^* = \underset{u}{\text{argmin}} \quad & \bar{u}^T H_{CLF} \bar{u} + f_{CLF}^T \bar{u} && \text{(CLF-QP)} \\ \text{s.t.} \quad & \dot{V}_\varepsilon(x) \leq -\varepsilon V_\varepsilon(x) \\ & bF_z \leq \tau_y \leq aF_z \end{aligned} \tag{3}$$

where $x = [q, \dot{q}]^T$, $H_{CLF}$ and $f_{CLF}$ are the quadratic and linear objective function respectively. The first constraint establishes the exponential stability of output tracking where $V_\varepsilon(x)$ is a RES-CLF. The second inequality ensures that the instantaneous ZMP lies within the support polygon.

However, the second inequality (3) is not guaranteed to be solvable, i.e. the robot can enter states for which there is no feasible solution to the ZMP constraints (3). This limitation is one of the primary motivators for combining local QPs with COM trajectory planning methods. In the following section, we show how to pose a Quadratic Program which solves for COM trajectories that satisfies ZMP constraints in the linear inverted pendulum model.

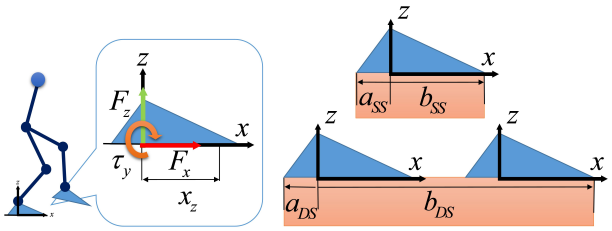


Fig. 2: The ZMP position $x_z$, ground reaction forces, and the corresponding ZMP boundaries $a$ and $b$ in single support (SS) and double support (DS) are shown.

### C. Linear Inverted Pendulum Model for COM Trajectory Generation

To simplify the ZMP tracking problem, one common approach is to generate a COM trajectory with the linear inverted pendulum model which tracks a desired ZMP trajectory. Model Predictive Control (MPC) is one method which has been employed in the literature for pattern generation with the Linear Inverted Pendulum Model (LIP model) [15], [18], [14]. The LIP model assumes a constant center of mass height. The resulting equation of motion forms a simple expression relating the ZMP and the horizontal COM, $x_c$,

$$\ddot{x}_c = \frac{g}{z_0}(x_c - x_z) \triangleq \omega^2(x_c - x_z) \tag{4}$$

where $z_0$ is the constant COM height and $g$ is the gravity constant. To implement MPC with the LIP model in (4), the discretized state space form of LIP can be derived as shown:

$$x_{t+1} = \begin{bmatrix} 1 & \Delta T & 0 \\ \omega^2 \Delta T & 1 & -\omega^2 \Delta T \\ 0 & 0 & 1 \end{bmatrix} x_t + \begin{bmatrix} 0 \\ 0 \\ \Delta T \end{bmatrix} u_t \tag{5}$$

where $x_t = \begin{bmatrix} x_{ct} & \dot{x}_{ct} & x_{zt} \end{bmatrix}^T$, $u_t = \dot{z}_t$, and $\Delta T$ is the sampling time. With a given initial state $x_{t_0}$ and a sequence of control inputs $\bar{U}$ , the predicted sequence of states $\bar{X}$ for the next $N$ time-steps can be expressed as $\bar{X} = \bar{A}\, \bar{X}_{t_0} + \bar{B}\, \bar{U}$ and $\bar{A}$ and $\bar{B}$ can be derived recursively from (5). The predicted states then can be used to formulate an MPC-based quadratic program for COM trajectory generation:

$$\begin{aligned} \bar{U}^* = \underset{\bar{U}}{\text{argmin}} \quad & \bar{U}^T H_p \bar{U} + f_p^T \bar{U} && \text{(MPC-QP)} \\ \text{s.t.} \quad & A_{iq,p} \bar{U} \leq b_{iq,p}, \end{aligned} \tag{6}$$

where $A_{iq,p}$ and $b_{iq,p}$ include constraints on the evolution of the COM and ZMP over an $N$ time-step forward horizon. The advantage of COM generation with MPC is that it can be easily implemented in real-time. However, due to the simplification, there are some potential issues considering the implementation in full nonlinear dynamics, such as the fact that the control sequence $\bar{U}$ may not be feasible, or the generated COM trajectory for the simplified LIP model may not result in a feasible ZMP trajectory. These potential issues motivate a combined control method which takes advantage of the rapid pattern generation capabilities of the MPC-QP and ensures that the actual nonlinear control system satisfies instantaneous balance constraints.

## III. Unification of Local Nonlinear Control and Walking Pattern Generation

In this section, we present the main formulation of the paper: a process for combining the nonlinear CLF-QP with the MPC-QP in a single control framework. Before the unification process is introduced, the setup of the CLF-QP and the MPC-QP, i.e. the construction of objective functions and constraints, will each be explained.

### A. Control Lyapunov Functions

To realize ZMP-based locomotion, a local nonlinear controller in the form of the CLF-QP is used for tracking a set of control objectives [5]. For the controller construction, the rigid body equations of motion (2) can be expressed in the following general nonlinear control system form:

$$\dot{x} = f(x) + g(x)\bar{u}, \tag{7}$$

where $x = [q, \dot{q}]^T$. Input/output linearization [13] can be used to drive a set of control outputs $y(q) \triangleq y_a(q) - y_d(t)$ toward zero (where $y_a(q)$ are actual outputs, $y_d(t)$ are time-based desired outputs). Here, the input/output relation for the (relative degree two) outputs $y(q)$ is

$$\ddot{y} = L_f^2 y(x) + L_f L_g y(x)\bar{u} + \ddot{y}_d \triangleq L_f + \bar{A}\bar{u} + \ddot{y}_d, \tag{8}$$

where "L" is the Lie derivative operator and $\bar{A}$ denotes the decoupling matrix. Given a desired output dynamics $\ddot{y} = \mu$, a corresponding vector of joint torques and GRFs, $\bar{u}$, can be obtained through (8). In standard input/output linearization, this requires the inversion of $\bar{A}$, however, as mentioned in [5], (8) can be resolved implicitly via quadratic programming.

The goal in the design of $\mu$ is to drive $y \to 0$. This motivates consideration of a linearized system with coordinates $\eta = [y, \dot{y}]^T$ which can be expressed as $\dot{\eta} = F\eta + G\mu$. To exponentially stabilize $\eta$ to zero with a convergence rate $\varepsilon > 0$, $\mu$ is designed to satisfy the following condition:

$$\dot{V}_\varepsilon(\eta) = L_f V_\varepsilon(\eta) + L_g V_\varepsilon(\eta)\mu \leq -\varepsilon V_\varepsilon(\eta), \tag{9}$$

where $V_\varepsilon(\eta) = \eta^T P_\varepsilon \eta$ is a RES-CLF, $P_\varepsilon$ is obtained by solving equation (47) in [3], and $L_f V_\varepsilon(\eta) = \eta^T(F^T P_\varepsilon + P_\varepsilon F)\eta$, $L_g V_\varepsilon(\eta) = 2\eta^T P_\varepsilon G$. A CLF-based Quadratic Program (CLF-QP) of the form (3) is implemented to find the minimum control input $\mu$ that guarantees Lyapunov stability through the satisfaction of (9) and additional constraints.

### B. CLF-QP Setup

This section describes the construction of the specific constraints and cost function considered for the local nonlinear CLF-QP of interest in this paper; for more detail, see [5].

*1) CLF-QP Constraints:* The final form of the set of constraints to be used in the proposed CLF-QP variant is

$$A_{iq,CLF}\bar{u} \leq b_{iq,CLF}, \tag{10}$$

$$A_{eq,CLF}\bar{u} = b_{eq,CLF}, \tag{11}$$

where $A_{iq,CLF}$ and $A_{eq,CLF}$ are matrices and $b_{iq,CLF}$ and $b_{eq,CLF}$ are vectors of appropriate dimension. The subscripts $iq$ and $eq$ denote inequality and equality, respectively.

As discussed in the previous section, ZMP constraints (1) are included in the (CLF-QP) optimization to ensure that the robot maintains dynamic balance. An additional constraint is included to ensure that the normal force applied to support foot is positive, i.e. $F_z \geq 0$. Note that the ZMP constraints (1) and the normal force constraint can be written as inequality constraints on $\bar{u}$ using the equations of motion (2), and thus be included in (10). Actuator saturation limits are likewise incorporated in (10) via the inequalities $-u_{max} \leq u \leq u_{max}$, where $u_{max}$ is a vector of maximum allowable torques. Finally, a CLF constraint is to used to drive the control objectives $\eta \to 0$. However, as aggressive control objectives and conservative torque limits can lead to infeasible systems of inequalities, the CLF constraint is relaxed [5] by $\delta > 0$, resulting in

$$\dot{V}_\varepsilon(\eta) = L_f V_\varepsilon(\eta) + L_g V_\varepsilon(\eta)\mu \leq -\varepsilon V_\varepsilon(\eta) + \delta. \tag{12}$$

The relaxation $\delta$ will be minimized in the cost function of the corresponding CLF-QP. The CLF constraint (12) together with the ZMP, normal force and torque constraints comprise the inequality constraints (10).

The equality constraints (11) enforce holonomic constraints $h(q) = 0$ corresponding to contact(s) between the robot and the ground through a constraint on the acceleration

$$\ddot{h}(q, \dot{q}, \bar{u}) = 0. \tag{13}$$

The vector $h(q)$ includes the horizontal and vertical components of one foot in the single support phase, and both feet in the double support phase.

*2) CLF-QP Cost Function:* The CLF-QP cost function is designed to balance the minimization of the control $\mu$ and the relaxation $\delta$ to the CLF constraint in (12)

$$\underset{(\bar{u},\delta)}{\operatorname{argmin}} \quad p\delta^2 + \bar{u}^T \bar{A}^T \bar{A}\bar{u} + 2L_f^T \bar{A}\bar{u} \tag{14}$$

where $p > 0$ is a weighting factor. Note that (14) encodes the goal of minimizing $\mu^T\mu$ through (8). The CLF-QP cost function (14) and constraints (10)–(11) are used in conjunction with elements of a walking pattern generation QP to form the unified QP described in Section III-E.

### C. Walking Control Objectives

This section describes the choice of control objectives, i.e., $y_a(q)$ and $y_d(t)$, for achieving ZMP-based walking in the nonlinear system (2). To reduce the differences between the LIP and the full nonlinear dynamic model, the height of the COM is regulated to a constant $z_0 > 0$, and the desired torso angle with respect to inertial frame is set to zero. In the single support phase, the desired orientation of the swing foot is set to zero (to ensure that the foot lands flat on the ground), and the desired horizontal and vertical components of the swing foot are smooth time-based polynomial functions with zero boundary velocities and accelerations. Finally, note that

$$\ddot{x}_c = L_f^2 x_c + L_f L_g x_c \bar{u}, \tag{15}$$

is the actual acceleration of the center of mass in the nonlinear system. To achieve forward walking, an equality

constraint will be enforced on $\bar{u}$ to achieve a desired acceleration of the center of mass, i.e. $L_f^2 x_c + L_f L_g x_c \bar{u} = \ddot{x}_c^d$. The value of $\ddot{x}_c^d$ will be determined through use of the LIP model for walking pattern generation, as described in the next section and (15) will subsequently be used as a bridge between the nonlinear robot dynamics and the LIP model.

### D. LIP Model Predictive Control Setup

The CLF-QP – described by the constraints (10) and (11) and cost function (14) – provides a method of locally stabilizing control objectives in the full nonlinear robot dynamics while also ensuring the instantaneous ZMP constraints are satisfied. However, under the action of the CLF-QP alone, the robot can enter states for which there is no feasible solution to the ZMP constraints. This motivates the combination elements of the local nonlinear CLF-QP with elements of a Model Predictive Control QP for producing feasible ZMP trajectories over a forward horizon. The following sections describe the construction of the specific MPC-QP considered.

*1) General MPC Setup:* The MPC-QP solves a receding horizon problem using the discrete-time, LIP dynamics (5) with time-step $\Delta T$. The target walking behavior consists of alternative phases of single and double support. The target duration of the single and double support phases are $T_{SS}$ and $T_{DS}$ seconds, respectively. The number of discrete points in the plan is fixed to be $N = (T_{SS} + T_{DS})/\Delta T$. Similar to $\bar{X}$ in (5), the predicted evolution of the ZMP, $x_{zt}$, COM, $x_{ct}$, and COM velocity, $\dot{x}_{ct}$, for the next N discrete points can be expressed as:

$$\begin{aligned} \bar{X}_z &= \bar{A}_{zmp}\bar{X}_{t_0} + \bar{B}_{zmp}\bar{U} \\ \bar{X}_c &= \bar{A}_{com}\bar{X}_{t_0} + \bar{B}_{com}\bar{U} \\ \dot{\bar{X}}_c &= \bar{A}_{comV}\bar{X}_{t_0} + \bar{B}_{comV}\bar{U} \end{aligned} \tag{16}$$

where $\bar{A}_{zmp}$, $\bar{B}_{zmp}$, $\bar{A}_{com}$, $\bar{B}_{com}$, $\bar{A}_{comV}$ and $\bar{B}_{comV}$ also can be derived recursively from (5). As these expressions are affine in $\bar{U}$, constraints on the evolution of ZMP and COM can be expressed as constraints on $\bar{U}$.

*2) MPC Horizon Computation:* As mentioned previously, the COM trajectory planner will implement a receding horizon. The model predictive control problem will solve 2 phases into the future. In general, this means the problem will have 3 domains: one for the completion of the current phase (with $N_1$ discrete points), one for the entire duration of the next phase (with $N_2$ discrete points) and one for the remainder (with $N_3$ discrete points), where $N_1 + N_2 + N_3 = N$. As the target walking consists of alternating phases of single and double support, the values $N_1$, $N_2$ and $N_3$ will change depending on the current phase. Specifically, at a point in time $t$ during a single support phase, the number of discrete points in first domain is $N_{1,SS} = (T_{SS} - t)/\Delta T$, the number of discrete points for next domain is $N_{2,DS} = (T_{DS})/\Delta T$ and the third and final domain's discrete point number is $N_{3,SS} = t/\Delta T$. Similarly, at a point in time $t$ during a double support phase, the numbers of discrete points for the three domains are $N_{1,DS} = (T_{DS} - t)/\Delta T$, $N_{2,SS} = (T_{SS})/\Delta T$ and $N_{3,DS} = t/\Delta T$ respectively.

*3) MPC Constraints:* An equality constraint is imposed on the MPC-QP to enforce that the center of mass reaches the position $x_c^{goal}$ at the end of the trajectory with the terminal velocity $\dot{x}_c^{goal}$. Additionally, inequality constraints are imposed on the resultant ZMP trajectories to ensure that the ZMP lies within the support polygon throughout the duration of the plan.

$$\begin{aligned} A_{eq,p}\bar{U} &= b_{eq,p} \\ A_{iq,p}\bar{U} &\leq b_{iq,p} \end{aligned} \tag{17}$$

where

$$\begin{aligned} A_{eq,p} &= \begin{bmatrix} [0_{N-1}, 1]\bar{B}_{com} \\ [0_{N-1}, 1]\bar{B}_{comV} \end{bmatrix} \\ b_{eq,p} &= \begin{bmatrix} x_c^{goal} - [0_{N-1}, 1]\bar{A}_{com}\bar{X}_{t_0} \\ \dot{x}_c^{goal} - [0_{N-1}, 1]\bar{A}_{comV}\bar{X}_{t_0} \end{bmatrix} \\ A_{iq,p} &= [\bar{B}_{zmp}, -\bar{B}_{zmp}]^T \\ b_{iq,p} &= [\bar{b} - \bar{A}_{zmp}\bar{X}_{t_0}, -\bar{a} + \bar{A}_{zmp}\bar{X}_{t_0}]^T \end{aligned} \tag{18}$$

In (18), $\bar{a}$ and $\bar{b}$ both include three ZMP boundary sequences, which are determined by the three domains in the horizon, with the corresponding support phase (single support or double support) and step length ($N_1$, $N_2$, or $N_3$). For example, the frontal ZMP boundary of the horizon $\bar{a}$ will be $[\bar{a}_{SS}, \bar{a}_{DS}, \bar{a}_{SS} + 0.5L_{step}]^T$ if it is in single support, and $\bar{a}$ will be $[\bar{a}_{DS}, \bar{a}_{SS} + 0.5L_{step}, \bar{a}_{SS} + 0.5L_{step}]^T$ if it is in double support, where $L_{step}$ is the step length.

*4) MPC Cost Function:* The cost function balances the goals of minimizing control effort, achieving ZMP trajectory tracking, and driving the COM position to the desired location for next stepping. This formulation is similar to the one used in [18]. The sequence of control inputs $\bar{U}$ then can be derived by solving the following optimization problem:

$$\begin{aligned} \underset{\bar{U}^*}{\text{argmin}} \quad & \omega_1 \bar{U}^T \bar{U} + \omega_2 |\bar{X}_z - \bar{X}_z^{goal}|^2 && (19) \\ \text{s.t.} \quad & \bar{a} \leq \bar{X}_z \leq \bar{b} && \text{(ZMP)} \\ & x_{c_{t_0+N}} = x_c^{goal} && \text{(COM)} \\ & \dot{x}_{c_{t_0+N}} = \dot{x}_c^{goal} && \text{(COM Vel.)} \end{aligned}$$

where $\omega_1$ and $\omega_2$ are weighting factors, $\bar{X}_z^{goal}$ is the desired ZMP trajectory, $x_c^{goal}$ and $\dot{x}_c^{goal}$ are the desired COM terminal location and velocity at $t = t_0 + N$ respectively, $\bar{a}$ and $\bar{b}$ are ZMP boundary vectors of the horizon. Using the equation in (16), the cost function in (19) can be expressed as follows:

$$\begin{aligned} \underset{\bar{U}^*}{\text{argmin}} \quad & \frac{1}{2}\bar{U}^T H_p \bar{U} + {f_p}^T \bar{U} && \text{(MPC-QP)} \\ \text{s.t.} \quad & A_{eq,p}\bar{U} = b_{eq,p} \\ & A_{iq,p}\bar{U} \leq b_{iq,p} \end{aligned} \tag{20}$$

where

$$\begin{aligned} H_p &= 2\omega_1 I + 2\omega_2 \bar{B}_{zmp}^T \bar{B}_{zmp} \\ f_p &= 2\omega_2 [\bar{A}_{zmp}\bar{X}_{t_0} - \bar{X}_z^{goal}]^T \bar{B}_{zmp} \end{aligned} \tag{21}$$

Note that the desired ZMP sequence $\bar{X}_z^{goal}$ and the desired terminal COM position $x_c^{goal}$ are calculated based on a horizon which changes over time.

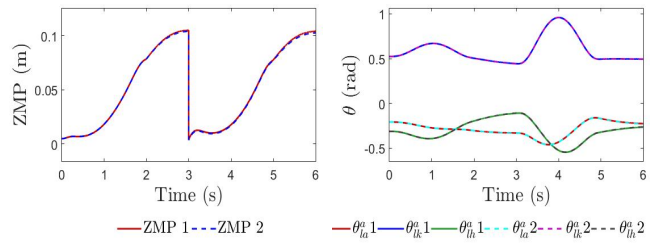


Fig. 3: A comparison of ZMP trajectories (left) and joint tracking profiles (right) from two different simulations of the proposed method: in simulation (1) the unified QP with terminal constraints on the COM is used and in simulation (2) the terminal constraints are not used.

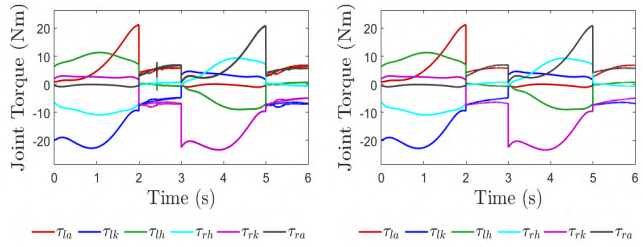


Fig. 4: Joint torques from simulation of the proposed unified RES-CLF QP with (left) and without (right) terminal constraints on the COM.

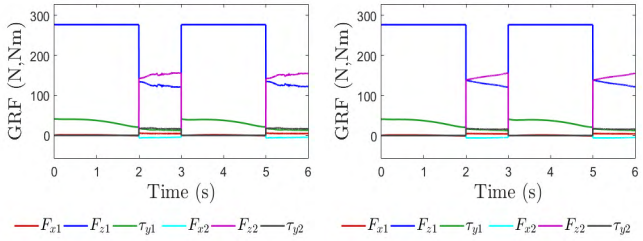


Fig. 5: Ground reaction forces from simulation using the proposed unified RES-CLF QP with (left) and without (right) terminal constraints on the COM.

### *E. Main Result: Unified QP Combining Pattern Generation and ZMP-based Walking Control*

Using the building blocks of the quadratic programs for pattern generation and ZMP-based locomotion with RES-CLF QP, the proposed controller synthesizes all elements into a unified quadratic program:

$$\underset{\bar{u}^*,\bar{U}^*,\delta^*}{\operatorname{argmin}} \frac{1}{2} \begin{bmatrix} \bar{u} \\ \bar{U} \\ \delta \end{bmatrix}^T \begin{bmatrix} H_{CLF} & 0 & 0 \\ 0 & H_p & 0 \\ 0 & 0 & p \end{bmatrix} \begin{bmatrix} \bar{u} \\ \bar{U} \\ \delta \end{bmatrix} + \begin{bmatrix} f_{CLF} \\ f_p \\ 0 \end{bmatrix}^T \begin{bmatrix} \bar{u} \\ \bar{U} \\ \delta \end{bmatrix} \tag{22}$$

$$\text{s.t.} \quad \begin{bmatrix} A_{eq,CLF} & 0 & 0 \\ 0 & A_{eq,p} & 0 \end{bmatrix} \begin{bmatrix} \bar{u} \\ \bar{U} \\ \delta \end{bmatrix} = \begin{bmatrix} b_{eq,CLF} \\ b_{eq,p} \end{bmatrix}$$

$$\begin{bmatrix} A_{iq,CLF} & 0 & -1 \\ 0 & A_{iq,p} & 0 \end{bmatrix} \begin{bmatrix} \bar{u} \\ \bar{U} \\ \delta \end{bmatrix} \leq \begin{bmatrix} b_{iq,CLF} \\ b_{iq,p} \end{bmatrix}$$

$$(L_f^2 x_c + L_f L_g x_c \bar{u}) \frac{z_0}{g} - x_c = -x_z$$

In (22), the LIP model equation in (4) is adopted as the QP synthesis constraint. The first term on left hand side is the COM acceleration expressed using input/output relation with full dynamics in (8). By solving the quadratic program above for each time step, the instantaneous torque input $\bar{u}$ for ZMP-based locomotion considering both output tracking and COM planning on-the-fly then can be derived.

## IV. Simulation Results

The unified controller was implemented in simulation on the model of AMBER 3[1], which is a human-sized, planar, and fully actuated bipedal robot (Fig. 1). Using the walking parameters listed in TABLE I, the proposed unified controller was implemented in MATLAB, where the unified QP combining the nonlinear CLF-QP with the MPC-QP is solved at every time step.

[1]AMBER 3 was built in AMBER Lab led by Dr. Aaron Ames at Texas A&M University. Since July 2015, AMBER Lab has moved to Georgia Tech, and AMBER 3 has been maintained and operated in HUR Group led by Dr. Pilwon Hur at Texas A&M University.

Compared to the controller solving MPC-QP and CLF-QP in sequence, several adjustments of the unified QP controller has to be made to ensure that the system would not be over-constrained. The first and most important change was to remove $x_c$ from the output vector in the unified framework. Since the resolved control input would also minimize the cost function in unified pattern generation, this soft ZMP tracking could implicitly provide larger flexibility for integration with other tasks than a strict ZMP tracking. Second, the COM terminal constraints which were originally for MPC stability were removed in the unified framework, since it would make the system over-constrained and the control input would lose continuity and cause chattering (as shown in Fig. 4 and Fig. 5), although the derived joint angle trajectories and ZMP patterns (Fig. 3) are similar. Last but not least, a impact map is adopted right after the single support phase in simulation, which will cause a discrete jump on COM velocty. The direct feedback of this COM velocity for updating $x_{t_0}$ in real-time COM planning will easily cause the resolved $x_c$ to diverge due to the large postimpact COM velocity. As a result, the feedback of postimpact COM velocity is assigned as zero to enforce the COM planned as free of impact, where the real impact effect in full dynamics is suppressed by the nonlinear controller as a perturbation.

TABLE I: Important Simulation Parameters

| Parameter | Value | Parameter | Value |
|---|---|---|---|
| $T_{SS}$ | 2 s | $T_{DS}$ | 1 s |
| MPC sampling time $\Delta T$ | 0.1 s | Length of MPC horizon | 3 s |
| $L_{step}$ | 10 cm | Stride Height | 5 cm |

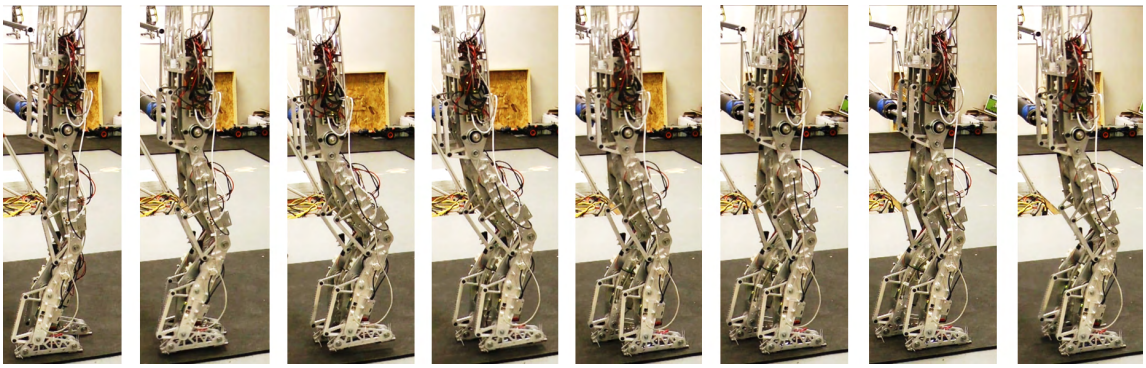

Fig. 6: The walking tiles of a half gait cycle from a trajectory tracking experiment in which AMBER 3 took 383 steps without falling using trajectories produced by the proposed method.

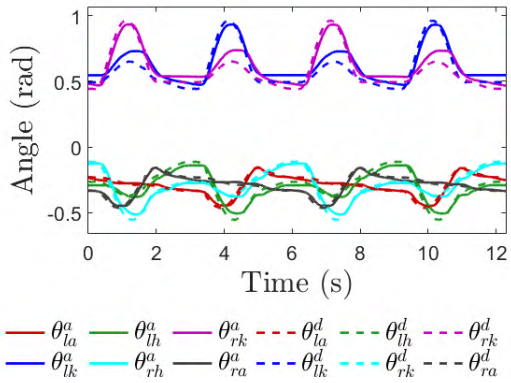


Fig. 7: The joint tracking results from a trajectory tracking experiment in which AMBER 3 took 383 steps without falling using trajectories produced by the proposed method.

## V. Experimental Results and Future Work

The experiment was implemented in LabVIEW with C++ on AMBER 3. The walking motion is generated from simulation by using the proposed unified QP controller. Using this setup, AMBER 3 walked for two laps (about 383 steps); results from the corresponding experiments are shown in Fig. 6 and Fig. 7. Using nonlinear constrained optimization, the torque command was converted into position and velocity commands. Although real-time implementation of the quadratic program is still in progress, the current joint trajectory tracking and experiment video [1] are quite similar to the walking motion displayed in simulation.

Future work entails completing a real-time implementation of the unified QP controller in C++. Robustness tests such as walking with disturbances, push recovery, or walking through uneven terrain are planned to be conducted.

## Acknowledgments

The authors thank Eric Ambrose, Wen-Loong Ma, Aakar Mehra, Victor Paredes, Michael Zeagler and other members in AMBER Lab for their assistance in the hardware implementation on AMBER 3.

## References

[1] `https://youtu.be/KNUo4AH2fv0`.
[2] A. D. Ames. Human-inspired control of bipedal walking robots. *IEEE Transactions on Automatic Control*, 59(5):1115–1130, May 2014.
[3] A. D. Ames, K. Galloway, and J. W. Grizzle. Control lyapunov functions and hybrid zero dynamics. In *IEEE Conference on Decision and Control*, pages 6837–6842, Dec 2012.
[4] A. D. Ames, K. Galloway, K. Sreenath, and J. W. Grizzle. Rapidly exponentially stabilizing control lyapunov functions and hybrid zero dynamics. *IEEE Transactions on Automatic Control*, 2014.
[5] A. D. Ames and M. J. Powell. Towards the unification of locomotion and manipulation through control Lyapunov functions and quadratic programs. In *Control of Cyber-Physical Systems*. Springer.
[6] P. A. Bhounsule, J. Cortell, A. Grewal, B. Hendriksen, J. G. D. Karssen, C. Paul, and A. Ruina. Low-bandwidth reflex-based control for lower power walking: 65 km on a single battery charge. *The International Journal of Robotics Research*, 2014.
[7] S. Faraji, S. Pouya, C. Atkeson, and A. Ijspeert. Versatile and robust 3d walking with a simulated humanoid robot (atlas): A model predictive control approach. In *IEEE International Conference on Robotics and Automation*, pages 1943–1950, May 2014.
[8] 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*, 3:323–332, 2015.
[9] J. W. Grizzle and E. R. Westervelt. Hybrid zero dynamics of planar bipedal walking. In *Analysis and Design of Nonlinear Control Systems*, pages 223–237. Springer, 2008.
[10] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *IEEE International Conference on Robotics and Automation*, volume 2, pages 1620–1626, Sept 2003.
[11] S. Kuindersma, F. Permenter, and R. Tedrake. An efficiently solvable quadratic program for stabilizing dynamic locomotion. *CoRR*, 2013.
[12] M. Posa and R. Tedrake. Direct trajectory optimization of rigid body dynamical systems through contact. In *Algorithmic Foundations of Robotics X*, volume 86 of *Springer Tracts in Advanced Robotics*, pages 527–542. Springer, 2013.
[13] S. S. Sastry. *Nonlinear Systems: Analysis, Stability and Control*. Springer, New York, 1999.
[14] B. Stephens. *Push Recovery Control for Force-Controlled Humanoid Robots*. PhD thesis, Carnegie Mellon University, 2011.
[15] B. Stephens and C. Atkeson. Push recovery by stepping for humanoid robots with force controlled joints. In *IEEE-RAS International Conference on Humanoid Robots*, pages 52–59, Dec 2010.
[16] M. Vukobratović and B. Borovac. Zero-moment point – Thirty five years of its life. *International Journal of Humanoid Robotics*, 01(01):157–173, 2004.
[17] Y. Wang and S. Boyd. Fast evaluation of quadratic control-lyapunov policy. *IEEE Transactions on Control Systems Technology*, 2011.
[18] P. B. Wieber. Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. *International Conference on Humanoid Robots*, pages 137–142, 2006.
[19] D. A. Winter. *Biomechanics and Motor Control of Human Movement*. Wiley, 2009.