Feedback Design for Multi-contact Push Recovery via LMI Approximation of the Piecewise-Affine Quadratic Regulator

Weiqiao Han and Russ Tedrake

Abstract—We consider the problem of stabilizing a robot that has to make and break multiple contacts with the environment. We approximate the dynamics of this hybrid system as a discrete-time Piecewise Affine (PWA) system. We propose novel techniques for the design of stabilizing controllers for such PWA systems. The Lyapunov stability conditions are translated into Linear Matrix Inequalities (LMIs). A Piecewise Quadratic (PWQ) Lyapunov function together with a Piecewise Linear (PL) feedback controller can be obtained by Semidefinite Programming (SDP). We show that we can embed a quadratic objective in the SDP, designing a Piecewise-Affine Quadratic Regulator (PWAQR) controller. Moreover, we observe that our formulation restricted to the linear system case appears to always produce exactly the unique stabilizing solution to the Discrete Algebraic Riccati Equation (DARE). In addition, we extend the search from the PL controller to the PWA controller via Bilinear Matrix Inequalities (BMIs). Finally, we demonstrate and evaluate our methods on a few PWA systems, including a simplified humanoid robot model.

I. INTRODUCTION

Local stabilization of a fixed point or a trajectory of a nonlinear system, such as a humanoid robot, is easily achieved by means of linearizing the dynamics and designing a Linear Quadratic Regulator (LQR) controller [1]. However, many critical tasks, such as recovery from a large external push, require a humanoid robot to make and break contact with its environment at multiple locations. For the purpose of such tasks, the humanoid robot is best modeled as a hybrid dynamic system. Unfortunately, there is a surprising lack of principled design techniques for such systems.

Despite this lack of generally applicable techniques, humanoid push recovery has been studied extensively in recent years. Strategies based on zero moment point (ZMP) are often used to balance the biped robot [2], [3]. In [4], three strategies were proposed for large disturbance recovery, including center of pressure (CoP) balancing, centroidal moment point (CMP) balancing, and stepping. N-step captures, the ability of a legged system to come to a stop without falling by taking N or fewer steps, was studied recently [5]. While previous work on humanoid push recovery has only looked at foot contacts, we are considering recovery strategies where the robot can also reach out with its hand and push on the surrounding environment.

To address these issues, we propose to globally approximate the nonlinear hybrid system by a time-stepping Piecewise Affine (PWA) system. Such a system can be obtained by performing a first order Taylor expansion of the nonlinear dynamics at many different points. PWA systems for short are defined by partitioning the state-input space into polyhedral regions and associating with each region a different affine state update equation [6]. We adopt the terminology in the hybrid system literature and call each region a “mode”. Approximating a nonlinear system by a PWA system is actually ubiquitous, and in theory any general nonlinear system can be globally approximated by a PWA system with arbitrary accuracy [7]. We are interested in stabilizing the resulting PWA system.

A state-of-the-art planning technique that can be used to tackle the problem of stabilizing PWA systems is based on explicit Model Predictive Control (MPC) and Mixed Integer Quadratic Programming (MIQP) [8]. This approach enumerates all possible mode switch sequences over a certain number of time steps. For example, {mode 2 at time 1, mode 2 at time 2, mode 1 at time 3, ..., mode 1 at time T} is one such mode switch sequence. For each mode switch sequence the approach solves a multi-parametric QP, and by comparing the cost of all such solutions it finds the optimal solution. The optimal controller is known to be piecewise affine and the cost-to-go function piecewise quadratic [9]. This approach performs only a few time steps of look-ahead and scales badly with the number of time steps. Different from the previous approach, we mainly seek a natural analogue of the LQR control design procedure for the PWA system, which shall be more scalable.

The study of the stability and stabilization of PWA systems goes back decades ago. Hassibi et al. studied the stabilization and control of PWA system via a Linear Matrix Inequality (LMI) approach [10]. Solving an LMI amounts
to solving a feasibility Semidefinite Program (SDP). A Piecewise Quadratic (PWQ) Lyapunov function was used for the stability proof. An ellipsoidal approximation to the state cells was used to reduce the conservativeness of the LMI. However, they considered continuous-time systems instead of discrete-time systems. Özkan et al. studied MPC for discrete-time PWA systems [11]. To reduce the complexity, they assume the mode switch sequence is known. For controller synthesis, both Hassibi et al. and Özkan et al. use a common quadratic Lyapunov function instead of a PWQ Lyapunov function, which is conservative.

In this paper, we study the Piecewise Affine Quadratic Regulator (PWAQR) problem, the controller design procedure where the system dynamics is PWA and the cost is quadratic. We adopt an LMI approach, similar to [10], [11]. We solve an SDP to get a PWQ Lyapunov function, a Piecewise Linear (PL) feedback controller, and an upper bound on the cost of any trajectory. However, the SDP depends on the initial state of the system. In order to be general, we consider a few variants of the SDP. We observe that one of the variants is more natural than the others in that when it is applied to the linear system, it appears to always produce the unique stabilizing solution to the Discrete Algebraic Riccati Equation (DARE), hence making it a true generalization of LQR. Furthermore, we extend our search from the PL controller to the PWA controller by means of a PWQ Lyapunov function with linear and constant terms. In contrast to the case of searching for a PL controller, searching for a PWA controller requires a formulation involving Bilinear Matrix Inequalities (BMI), a much harder problem. Finally, we demonstrate and evaluate our methods on a few PWA systems, including a simplified humanoid model (Figure 1).

While the controller synthesis for PWA systems (Section II) is standard, our main contribution is deriving the PWAQR design rule (Section III), extending it from designing PL controllers to designing PWA controllers (Section IV), and applying the rule to humanoid push recovery where the robot makes and breaks multiple contacts with the environment (Section V). Compared with [4], [5], our approach applied to humanoid push recovery has the following advantages:
1. works on non-flat terrain;
2. does not assume that the CoM moves on a plane;
3. can incorporate richer swing leg dynamics;
4. can handle a multi-contact scenario.

A. The Piecewise Affine System

Discrete-time Piecewise Affine (PWA) systems are described by the state-space equations:
\[ x_{k+1} = A_i x_k + B_i u_k + a_i, \text{ for } x_k \in X_i \subseteq \mathbb{X} \]  

where the state set \( \mathbb{X} \subseteq \mathbb{R}^n \) is a polyhedron containing the origin, \( \{X_i\}_{i=1}^s \) is a polyhedron partition of \( \mathbb{X} \), and \( u_k \in \mathbb{R}^m \) is the control input. \( I = \{1, \ldots, s\} \) is the set of indices of the state space cells. \( I_0 \subseteq I \) is the set of indices of the state space cells that contain the origin (There can be cases where the origin is on the boundaries of several cells.). Denote \( I_1 = I \setminus I_0 \). \( \mathcal{S} = \{(i,j) \in I \times I : \exists \ x_k \in X_i, x_{k+1} \in X_j\} \) is the set of all ordered pairs (\( i,j \)) of indices, denoting the possible switches from cell \( i \) to cell \( j \).

Assume \( X_i = \{x \mid E_i x \geq e_i\} \). For later use, it is convenient to outer approximate each cell \( X_i \) with a union of ellipsoids \( \mathcal{E}_{ip} = \{x \mid ||F_{ip}x + f_{ip}|| \leq 1\} \), \( p = 1, \ldots, n_i \):
\[ X_i \subseteq \bigcup_{p=1}^{n_i} \mathcal{E}_{ip}. \]

We assume \( x = 0 \) is an equilibrium of the system (1), and \( a_i = 0 \) for all \( i \in I_0 \).

B. State Feedback Synthesis

We consider the synthesis of a PL state feedback
\[ u_k = K_i x_k, \forall x_k \in X_i, \]  

for the PWA system (1) that stabilizes the origin, certified by a PWQ Lyapunov function
\[ V(x) = x^\top P_i x, P_i > 0, \forall x \in X_i. \]  

By Lyapunov theory, a sufficient condition for stability is that
\[ \Delta V(x_{k+1}, x_k) = V(x_{k+1}) - V(x_k) < 0 \]  

for any \( x_k \in \mathbb{X} \).

In the following, to simplify the notations, we denote \( A_i + B_i K_i \), the closed-loop state matrix, by \( A_{cl,i} \).

Since
\[ \Delta V(x_{k+1}, x_k) = (A_{cl,i} x_k + a_i)^\top P_j (A_{cl,i} x_k + a_i) - x_k^\top P_i x_k, \]
for \( x_k \in X_i \), the condition (4) is equivalent to looking for the matrices \( P_i > 0 \) and \( K_i, \forall i \in I \) such that
\[ \begin{bmatrix} x_k^\top & 1 \end{bmatrix} \begin{bmatrix} A_{cl,i}^\top P_j A_{cl,i} - P_i & A_{cl,i}^\top P_j a_i \\ a_i^\top P_j A_{cl,i} & a_i^\top P_j a_i \end{bmatrix} \begin{bmatrix} x_k \\ 1 \end{bmatrix} < 0, \forall x_k \in X_i, \forall (i,j) \in \mathcal{S}. \]  

A sufficient condition for (5) to hold is
\[ \begin{bmatrix} A_{cl,i}^\top P_j A_{cl,i} - P_i & a_i^\top P_j a_i \\ a_i^\top P_j A_{cl,i} & a_i^\top P_j a_i \end{bmatrix} < 0, \forall (i,j) \in \mathcal{S}. \]  

By Schur complement, (6) is equivalent to
\[ \begin{bmatrix} P_i & 0 & 0 \\ 0 & a_i & 0 \\ a_i & P_j^{-1} \end{bmatrix} > 0, \forall (i,j) \in \mathcal{S}. \]  

From now on, for symmetric matrices, we sometimes omit the symmetric halves and simply use stars * to represent their entries. Introducing variables \( W_i = P_i^{-1}, Y_i = K_i W_i \), and
multiplying both sides of (7) by \( \begin{bmatrix} W_i & 0 \\ 0 & I \end{bmatrix} \), (7) is equivalent to
\[
\begin{bmatrix}
W_i & * & * \\
0 & 0 & * \\
A_iW_i + B_iY_i & a_i & W_f
\end{bmatrix} > 0, \forall (i, j) \in S,
\] (8)
which is an LMI in \((W_i, Y)\).

However, since the inequality in (8) is strict but the matrix on the left-hand side has a principal minor equal to 0, (8) has no solution. We will see later that the PWAQR design procedure resolves this issue automatically. Now we look for a better sufficient condition for (5) that can be turned into a feasible LMI.

The sufficient conditions (6) is conservative, because it requires the inequality in (5) to hold for all \( x_k \in \mathbb{R}^n \), while we only need it hold for all \( x_k \in \mathcal{X}_i \). A less conservative sufficient condition can be obtained by outer ellipsoid approximation to the state cells \( \mathcal{X}_i \subseteq \bigcup \mathcal{E}_{ip} \). We want the inequality in (5) holds for all \( x_k \in \bigcup \mathcal{E}_{ip} \), i.e., for all \( x_k \) satisfying
\[
\begin{bmatrix}
x_k^T \\
1
\end{bmatrix}
\begin{bmatrix}
P_{ip}^T & f_{ip}^T & f_{ip}^T & f_{ip}^T
f_{ip} & f_{ip} & f_{ip} & f_{ip} - 1
\end{bmatrix}
\begin{bmatrix}
x_k \\
1
\end{bmatrix} < 0, 1 \leq p \leq n_i.
\] By S-procedure, a sufficient condition for (5) becomes
\[
\begin{bmatrix}
A_{ip}^T P_j A_{cl,i} - P_i & A_{ip}^T P_j a_i \\
a_i P_j A_{cl,i} & a_i P_j a_i
\end{bmatrix}

- \lambda_{ip} \begin{bmatrix}
P_{ip}^T & f_{ip}^T & f_{ip}^T & f_{ip}^T \\
f_{ip} & f_{ip} & f_{ip} & f_{ip} - 1
\end{bmatrix} < 0,
\] (9)
\[
\lambda_{ip} > 0, 1 \leq p \leq n_i, (i, j) \in S.
\]
By some algebraic manipulations, including Schur complement, similar matrix transformations, and the identities:

\[
(I - E^T E)^{-1} = I + E^T (I - EE^T)^{-1} E
\]
\[
E(I - E^T E)^{-1} = (I - EE^T)^{-1} E
\]
where \( E \) can be a matrix of any size, (4) is equivalent to
\[
\begin{bmatrix}
W_i & * & * \\
A_iW_i + B_iY_i & W_f + \beta_{ip} a_i a_i^T & \beta_{ip} f_{ip} f_{ip} - I
\end{bmatrix} > 0,
\] (10)
where \( \beta_{ip} = \lambda_{ip}^{-1} \).

Notice that (10) requires \( f_{ip} f_{ip}^T - I > 0 \), which does not hold for \( i \in I_0 \). For \( i \in I_0 \), we simply require
\[
A_{ip}^T P_j A_{cl,i} - P_i < 0,
\] which is equivalent to
\[
\begin{bmatrix}
W_i & * & * \\
A_iW_i + B_iY_i & W_f
\end{bmatrix} > 0, (i, j) \in S,
\] (11)

As pointed out in [10], [11], the polyhedron description of the state cell cannot be incorporated into the LMI. That is why we have to approximate the polyhedron by the ellipsoids.

In summary, a PL state feedback controller that stabilizes the origin can be obtained by solving the feasibility SDP:

\[
\begin{align*}
& \text{find } W_i, Y, \beta \\
& \text{subject to } W_i > 0, i \in I, \\
& \text{(11)} \quad \text{if } i \in I_0, \\
& \text{(10)} \quad \text{if } i \in I_1,
\end{align*}
\]
and the state feedback matrix \( K_i = Y_i W_i^{-1}, i \in I \).

III. PWAQR: PL CONTROLLER

In this section, we derive the PWAQR design procedure, i.e., the controller design procedure where the system dynamics is PWA and the cost function is quadratic. The controller in consideration is still PL as in (2) and the PWQ Lyapunov function is still of the form (3).

A. Quadratic Objective and Its Upper Bound

We consider a quadratic objective (cost function) for the controller synthesis of the PWA system. Define the cost matrices \( Q_i \geq 0, R_i > 0 \) for the cell \( \mathcal{X}_i, i \in I \). The quadratic cost function is
\[
\sum_{k=0}^{\infty} x_k^T Q_i(x_k) + u_k^T R_i(x_k) u_k,
\] (12)
where \( i(k) \in I \) is the index such that \( [x_k^T, u_k^T]^T \in \mathcal{X}_i(k) \).

Lemma 1. If there are matrices \( P_i > 0 \) and \( K_{i}, i \in I \), satisfying
\[
\Delta V(x_{k+1}, x_k) + x_k^T (Q_i(k) + K_{i}^T R_i(k) K_{i}) x_k \leq 0, \forall x_k,
\] (13)
then the PL controller (2) stabilizes the origin asymptotically, and the PWQ Lyapunov function \( V(x) = x^T P_i x \) proves the bound
\[
\sum_{k=0}^{\infty} x_k^T Q_i(k) x_k + u_k^T R_i(x_k) u_k \leq x_0^T P_i(x_0) x_0.
\] (14)

\[\square\]

Proof: Since \( \Delta V(x_{k+1}, x_k) < 0 \) for all \( x_k \neq 0 \), the controller asymptotically stabilizes the origin.

Since \( x_k^T (Q_i(k) + K_{i}^T R_i(k) K_{i}) x_k \leq -\Delta V(x_{k+1}, x_k) \). Summing over the trajectory \( \{x_k\}_{k=0}^{\infty} \),
\[
\sum_{k=0}^{\infty} x_k^T Q_i(k) x_k + u_k^T R_i(x_k) u_k
\]
\[
= \sum_{k=0}^{\infty} x_k^T (Q_i(k) + K_{i}^T R_i(k) K_{i}) x_k
\]
\[
\leq \sum_{k=0}^{\infty} -\Delta V(x_{k+1}, x_k)
\]
\[
= \sum_{k=0}^{\infty} -(V(x_{k+1}) - V(x_k))
\]
\[
= V(x_0).
\]
In the last step, the series $\sum_{k=0}^{\infty} -(V(x_{k+1}) - V(x_k))$ converges to $V(x_0)$, because the partial sum $\sum_{k=0}^{K} -(V(x_{k+1}) - V(x_k)) = V(0) - V(K+1)$ and $V(K+1) \to 0$ as $K \to \infty$ by asymptotic stability. 

Since $V(x_{k+1}, x_k) = (A_{cl,i} x_k + a_i)\top P_j (A_{cl,i} x_k + a_i) - x_k \top P_{ij} x_k$, for $x_k \in X_i$, 

$$\begin{bmatrix} x_k \\ 1 \end{bmatrix} \begin{bmatrix} A_{cl,i} P_j A_{cl,i} - P_i + Q_i + K_i R_i K_i & 0 \\ 0 & a_i \top P_j a_i \end{bmatrix} \begin{bmatrix} x_k \\ 1 \end{bmatrix} \leq 0, \quad \forall x_k \in X_i, \forall (i,j) \in S.$$ 

A sufficient condition for (15) is 

$$\begin{bmatrix} A_{cl,i} P_j A_{cl,i} - P_i + Q_i + K_i R_i K_i & 0 \\ 0 & a_i \top P_j a_i \end{bmatrix} \leq 0, \quad \forall (i,j) \in S.$$ 

By Schur complement, (16) is equivalent to 

$$\begin{bmatrix} W_i & * & * & * & * \\ 0 & 0 & * & * & * \\ A_i W_i + B_i Y_i & a_i & W_j & * & * \\ Q_i^{1/2} W_i & 0 & 0 & I & * \\ R_i^{1/2} Y_i & 0 & 0 & 0 & I \end{bmatrix} \geq 0, \forall (i,j) \in S,$$ 

Here we have non-strict inequality. The previous issue raised by the strict inequality is automatically resolved. Moreover, even if $i \in I_0$ and $a_i = 0$, we can still use (17).

We want to minimize the upper bound (14) on the cost function. This leads to the following lemma.

**Lemma 2.** Let $V(x) = x\top P x, \forall x \in X_i$, A stable PL state feedback that asymptotically stabilizes the origin with initial state $x_0$ can be found solving the SDP (18) for $\gamma$, $W_i$ and $Y_i$. $K_i$ is then given by $K_i = Y_i W_i^{-1}$. The cost of any trajectory $(x_k)_{k=0}^{\infty}$ is bounded by $\gamma$.

$$\min_{\gamma, W_i, Y_i} \gamma$$

subject to $\begin{bmatrix} \gamma \\ x_0 \top W_i(0) \end{bmatrix} \geq 0, W_i > 0$, and (17).

Notice that the SDP (18) depends on the initial state $x_0$. This is not in the spirit of LQR for linear systems, and would necessitate solving SDPs online, which is impractical. We want an SDP that is independent of the initial state. This will be discussed in detail later.

**B. Ellipsoid Approximation**

As before, using outer ellipsoid approximation, a sufficient condition for (13) is 

$$\begin{bmatrix} W_i & * & * & * & * \\ A_i W_i + B_i Y_i & W_j & * & * & * \\ Q_i^{1/2} W_i & 0 & I & * & * \\ R_i^{1/2} Y_i & 0 & 0 & I & * \end{bmatrix} \geq 0, \forall i \in I_0, (i,j) \in S,$$ 

subject to $A_i W_i + B_i Y_i \geq 0, (i,j) \in S$.

$$\min_{\gamma, W_i, Y_i} \gamma$$

subject to $A_i W_i + B_i Y_i \geq 0, (i,j) \in S$.

$$\begin{bmatrix} W_i & * & * & * & * \\ A_i W_i + B_i Y_i & W_j & * & * & * \\ Q_i^{1/2} W_i & 0 & I & * & * \\ R_i^{1/2} Y_i & 0 & 0 & I & * \end{bmatrix} \geq 0, \forall i \in I_0, (i,j) \in S.$$

### C. Variants of the Objective

As mentioned earlier, the SDP (18) depends on the initial state $x_0$. Now we discuss some possible variants of the objective that are independent of the initial state $x_0$.

Since we want to minimize the quantity $x_0 \top P_i(0)x_0$ for generic $x_0$, it is natural to minimize $\text{trace}(P_i(0))$, or more generally, minimize $\text{trace}(\sum_i P_i)$. Since $P = W^{-1}$, it is natural to maximize $\text{trace}(\sum_i W_i)$. If we don’t care about the upper bound on the cost and only want to find a feasible controller, we can simply solve a feasibility problem. Another possible objective arises if we let $P_i = \gamma_i W_i^{-1}, \forall i$, and minimize $x_0 \top P_i(0)x_0$ as in [11]. The inequality $\gamma_i \geq x_0 \top P_i(0)x_0$ becomes the LMI

$$\begin{bmatrix} 1 \\ x_0 \top W_i(0) \end{bmatrix} \geq 0$$ 

This still depends on the initial state. We summarize the variants of the objective below.

1) **SDP1**

   SDP (18).

2) **SDP2**

   maximize $\text{trace}(\sum_{i=1}^{s} W_i)$

   subject to $W_i > 0$, and (17).

3) **SDP3**

   minimize $0$

   subject to $W_i > 0$, and (17).

4) **SDP4**

   minimize $\gamma$
Let $Q$ be a linear system. This suggests that our choice of the objective is more natural. However, we have not yet been able to prove it, so we leave it as a conjecture.

**Conjecture 1.** Let $x_{k+1} = Ax_k + Bu_k$ be a linear system. Let $Q \geq 0, R > 0$ be cost matrices. Assume that $(A, B)$ is a stabilizable pair, and that $(A, C)$ is a detectable pair, where $C$ is a full-rank factorization of $Q$ (i.e., $C^T C = Q$ and rank$(C) =$ rank$(Q)$).

Suppose $W$ and $Y$ is a pair of solutions to the SDP \(23\). Let $P_1 = W^{-1}$ and $K_1 = Y W^{-1}$.

Suppose $P_2$ is the unique stabilizing solution to the Discrete Algebraic Riccati Equation

$$X = A^T X A - (A^T X B) (R + B^T X B)^{-1} (B^T X A) + Q,$$

and the optimal gain matrix is

$$K_2 = -(R + B^T P_2 B)^{-1} B^T P_2 A.$$

Then

$$P_1 = P_2, K_1 = K_2.$$ 

There are also ellipsoid approximation counterparts for these SDP’s, which we do not explicitly write down here. We denote the ellipsoid approximation counterparts of the SDP’s by SDP1*, SDP2*, SDP3*, and SDP4*, correspondingly.

### IV. PWAQR: PWA CONTROLLER

So far, we have considered PL controllers in the PWAQR design. In this section, we generalize the previous results to the synthesis of a PWA controller

$$u_k = K_i x_k + b_i, \forall x_k \in X_i,$$

for the PWA system (1) that stabilizes the origin by means of a PWQ Lyapunov function of the full form

$$V(x) = x^T P_i x + 2 q_i^T x + r_i, \forall x_k \in X_i.$$  \(27\)

The analogue of (13) is

$$\Delta V(x_{k+1}, x_k) = x_k^T Q(i) x_k + (K_i x_k + b_i(i))^T R(i)(K_i x_k + b_i(i)) \leq 0, \forall x_k.$$  \(28\)

Denote $a_{cl,i} = B_i b_i + a_i$. Then $x_{k+1} = A_i x_k + B_i u_k + a_i = A_{cl,i} x_k + a_{cl,i}$.

We obtain BMIs in the variables $(W, q, r, Y, b)$ for controller synthesis

$$\begin{bmatrix}
W_i \\
q_j^T A_{cl,i} W_i + q_i^T W_i \\
-2 q_j^T a_{cl,i} + r_i - r_j \\
A_{cl,i} W_i \\
Q \frac{1}{2}W_i \\
R \frac{1}{2} Y_i \\
n_{cl,i} W_j \\
a_{cl,i} W_j \\
W_j \\
2b_i \\
0 \\
0 \\
0 \\
I \\
I
\end{bmatrix} \geq 0, (i, j) \in S,$$

(notice that $A_{cl,i} W_i = A_i W_i + B_i Y_i$)

$W_i > 0, i = 1, \ldots, s,$

$E_i W_i q_i + e_i > 0, i \in I_i,$

$q_i = 0, r_i = 0, i \in I_0.$  \(29\)

The first inequality in \(29\) ensures that $V$ decreases along all state trajectories. The local minima of the PWQ Lyapunov function candidate $V$ are in the set

$$Q = \{ -W_1 q_1, \ldots, -W_s q_s \}.$$  

In order to stabilize the system to the origin, we require $-W_i q_i, i \in I_i$ to be outside the region $X_i$, and $-W_i q_i, i \in I_0$ to be the origin. The third and the fourth inequalities in \(29\) guarantee this property. Together with the second inequality, they ensure that $V$ is positive.

Solving the BMIs is NP-hard [12]. Nevertheless, there exist fast heuristic methods for solving them, for example, based on SDP [13]. However, the approach in [13] does not guarantee to find the global minimum.

### V. EXPERIMENT

#### A. Cart-Pole Balance Control

![Cart-Pole System](cart-pole.png)

Fig. 2. The cart-pole system.

We consider the problem of balancing the cart-pole system as shown in Figure 2. We want to show that SDP2 produces the LQR controller for the linearized system around the fixed point. Let $q = [x, \theta]^T$, $x = [q^T, \dot{q}^T]^T$ and $u = f$. We are interested in balancing the simple pendulum around its unstable fixed point $x^* = [0, \pi, 0, 0]^T$ using only horizontal force on the cart [14]. We assume there is no friction or air resistance.

The manipulator equation is

$$H(q) \ddot{q} + C(q, \dot{q}) \dot{q} + G(q) = B u$$

where

$$H(q) = \begin{bmatrix} m_c + m_p & m_p l \cos \theta \\ m_p l \cos \theta & m_p l^2 \end{bmatrix}, \quad G(q) = \begin{bmatrix} 0 \\ m_p g \sin \theta \end{bmatrix},$$

$$C(q, \dot{q}) = \begin{bmatrix} 0 & -m_p l \dot{\theta} \sin \theta \\ 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 1 \\ 0 \end{bmatrix}.$$  

Linearizing around the fixed point $(x^*, u^*) = ((0, \pi, 0, 0)^T, 0)$ using Taylor expansion, we have

$$\ddot{x} = f(x, u) = f(x^*, u^*) + \left[ \frac{\partial f}{\partial x} \right]_{x=x^*, u=u^*} (x - x^*)$$
\[ f(x, u) = \begin{bmatrix} H^{-1}(q)[B(q)u - C(q, \dot{q})\dot{q} - G(q)] \\ \dot{q} \end{bmatrix}, \]

\[ f(x^*, u^*) = [0, 0]^T, \]

\[ A_{lin} = \begin{bmatrix} -H^{-1} \frac{\partial G}{\partial q} + \sum_j H^{-1} \frac{\partial B_j}{\partial q} u_j & -H^{-1}C \end{bmatrix}_{x = x^*}, u = u^*, \]

\[ B_{lin} = \begin{bmatrix} 0 \\ H^{-1}B \end{bmatrix}_{x = x^*}, u = u^*. \]

where

\[ x_{t+1} = Ax_t + Bu_t, \]

where

\[ A = \begin{bmatrix} 1 & 0 & 0.05 & 0 \\ 0 & 1 & 0 & 0.05 \\ 0 & 0.0491 & 1 & 0 \\ 0 & 1.0791 & 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0.005 \\ 0.01 \end{bmatrix}. \]

Choose the cost matrices \( Q = I, R = I \). Our SDP2 produces exactly the LQR optimal state feedback gain

\[ K = \begin{bmatrix} 0.8027 \\ -214.6740 \\ 4.2763 \\ -46.4030 \end{bmatrix}. \]

This again strengthens our belief that the Conjecture 1 is true.

**Table 1**

<table>
<thead>
<tr>
<th></th>
<th>Explanation</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( m_c )</td>
<td>cart mass</td>
<td>10</td>
</tr>
<tr>
<td>( m_p )</td>
<td>pole mass</td>
<td>1</td>
</tr>
<tr>
<td>( l )</td>
<td>pole length</td>
<td>0.5</td>
</tr>
<tr>
<td>( g )</td>
<td>gravitational acceleration</td>
<td>9.81</td>
</tr>
<tr>
<td>( \Delta t )</td>
<td>discretization time interval</td>
<td>0.05</td>
</tr>
</tbody>
</table>

The discrete-time system dynamics is \( x_{t+1} = Ax_t + Bu_t \), where

\[ x = x^* \Rightarrow Qx + Bu = 0 \]

We choose the physical parameters of the cart-pole system as described in the Table I.

We next consider the 4-cell PWA system as described in [15]. We use this example to evaluate the controllers returned by SDP1, . . . , SDP4. The dynamics of the system is

\[ x_{k+1} = \begin{bmatrix} -0.04 & -0.461 \\ -0.139 & 0.341 \\ 0.936 & 0.323 \\ 0.788 & -0.049 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k, \quad E_1 x_k \geq 0 \]

\[ x_{k+1} = \begin{bmatrix} -0.857 & 0.185 \\ 0.491 & 0.62 \\ -0.022 & 0.644 \\ 0.758 & 0.271 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k, \quad E_2 x_k \geq 0 \]

\[ x_{k+1} = \begin{bmatrix} -0.461 & -0.04 \\ -0.341 & 0.139 \\ 0.323 & 0.936 \\ -0.049 & 0.788 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k, \quad E_3 x_k \geq 0 \]

\[ x_{k+1} = \begin{bmatrix} -0.04 & -0.461 \\ -0.139 & 0.341 \\ 0.936 & 0.323 \\ 0.788 & -0.049 \end{bmatrix} x_k + \begin{bmatrix} 1 \\ 0 \end{bmatrix} u_k, \quad E_4 x_k \geq 0 \]

where the partitioning corresponds to the four quadrants of the two dimensional \( x \) plane, i.e.,

\[ E_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad E_2 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}, \quad E_3 = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \quad E_4 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}. \]

The vector field of the system in the two-dimensional \( x \) plane is plotted in Figure 5 (Note that the vector field plot in [15] is incorrect.)

The PWQ Lyapunov functions of the SDP2 solution. Top left: \( Q = I, R = I \). Top right: \( Q = I, R = 10I \). Bottom left: \( Q = I, R = 20I \). Bottom right: \( Q = 10I, R = I \). The vertical axes have different scales.
The upper bound $\gamma$ by the red point. The rectangles surrounding the limb points pictured by the four cyan points, and a center of mass, depicted a simplified 2-dimensional model for the Valkyrie bipedal robot.

C. Simplified Humanoid Model

Finally, we consider the “box Valkyrie model” (Figure 8), a simplified 2-dimensional model for the Valkyrie bipedal robot. It has four massless, velocity-controlled limbs, depicted by the four cyan points, and a center of mass, depicted by the red point. The rectangles surrounding the limb points and the center of mass are just for better visual effects. It was called "box Valkyrie" because it was situated in a box, not because the limbs needed to be visualized as boxes. Two feet are on the floor. The arrows pointing upwards are the normal forces exerted by the floor. There are two walls at $x = -0.5$ and $x = 0.5$, respectively.

The goal is to keep the center of mass at the origin. The center of mass is controlled by the contact forces, which are exerted upon those limbs that are in contact with the environment. Different from previous work on humanoid push recovery, we consider the recovery strategies where the robot can reach out with its hand and push on the surrounding environment.

Notice that in Figure 8, there are two points at each foot. This is the contact model we are going to explain. When a limb $p$ is in contact with the wall or the floor,
we model it using two points: a non-penetrating point $p_1$ staying on the contact surface, and a penetrating point $p_2$ penetrating the contact surface, as shown in Figure 9. Both $p_1$ and $p_2$ are velocity controlled. The normal force $F_N$ is proportional to the vertical displacement between $p_1$ and $p_2$, and the frictional force $F_f$ is proportional to the horizontal displacement between $p_1$ and $p_2$, i.e., $F_N = -k \Delta z$ and $F_f = -k \Delta x$. The blue dashed lines in the Figure 9 are the boundary lines of the friction cone. We keep $p_2$ inside the "reflected friction cone" so that the frictional force lies in the friction cone. Once $p_1$ or $p_2$ goes outside the contact surface, we recombine them into one point $p$.

The box Valkyrie model has 20 states and 8 control inputs. Since the controller is linear in any specific mode, we cannot expect the controller to lift a foot and put it down somewhere else. So we simply put the feet at their equilibrium points in the initial state. The left hand may touch the left wall, and the right hand may touch the right wall. So the model is a piecewise affine system with four modes: no hands in contact with the walls, the left hand in contact with the left wall, the right hand in contact with the right wall, or two hands both in contact with the corresponding walls. The state-of-the-art MIQP approach would have two binary variables for each time step, one indicating if the right hand is touching the wall, and the other indicating if the left hand is touching the wall. It would have 4 possible modes at every time step, and the number of mode switch sequences grows exponentially with the number of time steps. Using our approach, we only need to think about the set $S$, which is polynomial (quadratic) in the number of modes. We use SDP2 or SDP2* to solve for the controller with $S = I \times I$. The sequence of poses in Figure 7 shows that when the robot body is pushed to the right and one hand is in contact with the wall, the controller stabilizes the center of mass in approximately 30 time steps.

VI. CONCLUSION AND FUTURE WORK

We have derived the PWAQR controller design rule, independent of the initial system state. We applied this method to a PWA approximation of a nonlinear hybrid system representing a humanoid robot’s centroidal dynamics in the plane. As oppose to the MPC and the MIQP approaches, our method does not require enumerating the mode switch sequences, and hence scales better. We have also generalized our result to the design of PWA controllers.

There are some limitations. (i) For the humanoid model, we do not take the centroidal angular momentum into account. (ii) We cannot incorporate the PWQ Lyapunov function of the full form (27) into the LMIs for the controller synthesis. So we have to express the Lyapunov stability conditions as BMIs, solving which is NP-hard. Also, the domain information, either the polyhedron or the ellipsoid approximation, cannot be incorporated into the BMIs.

Since the PWAQR naturally does not have any constraints on the control input, we impose the constraints on the control inputs by limiting the state of the system. For example, in the box Valkyrie model, we keep the position of the penetrating point $p_2$ inside the “reflected friction cone” so that the frictional force always lies inside the friction cone.

In the future, another possible way to try is to incorporate the force constraints into the system dynamics as in [16]. It is also interesting to prove Conjecture 1.

ACKNOWLEDGMENT

This work was supported by the NASA Award NNX16A4C9A. The views, opinions and positions expressed by the authors are theirs alone, and do not necessarily reflect the views, opinions or positions of NASA.

REFERENCES