Model Predictive Control for Continuous-time Piecewise Affine Systems Based on Mode Controllability

Yuichi Tazaki and Jun-ichi Imura

***Abstract*— This paper presents an approximate semi-offline approach to model predictive control of continuous-time piecewise affine (CTPWA) systems. The proposed method computes in the offline phase a controllable set and a minimum transition cost associated with each discrete state sequence. Then at each sampling time in the online phase, the controller determines the optimal sequence of intermediate target states over a prediction horizon, taking advantage of the information precomputed off line. By effectively distributing the computation over the offline phase and the online phase, real-time control is achieved, striking a good balance between the amount of precomputed data and the online computation time. The proposed method has been applied to a simple CTPWA system, obtained by piecewise linear approximation of a pendulum.**

***Index Terms*— Model Predictive Control, Continuous-time Piecewise Affine Systems**

I. Introduction

Hybrid systems are a class of systems involving both continuous dynamics and discrete events, and therefore capable of modeling a wide range of real world plants. This paper studies model predictive control of a subclass of hybrid systems called continuous-time piecewise affine systems. In model predictive control (MPC), a finite-horizon optimal control problem is solved at each sampling time, and the obtained input is applied until the next sampling time. This procedure is repeated from the next sampling time. Various control objectives can be specified by means of cost functions and it is also possible to handle state and input constraints in an explicit manner.

Recently, there has been growing research interest in applying MPC to a class of systems that exhibits relatively rapid motion, such as robots: to this aim, reducing real-time computational burden is a critical issue. The offline approach, which executes the major portion of the required computation in advance, is effective for this purpose [1][2]. Most of the existing offline approaches, however, suffer from the rapid growth of the size of the precomputed information. We would like to point out that one can find a balance between the size of precomputed information and the online computation time if the computation is distributed over the offline and the online phase, instead of being processed entirely in advance.

Imura [3] discussed the optimal control problem of sampled-data piecewise affine (SDPWA) systems. In these systems, the dynamics of the continous state during each sampling time interval is discribed by a continuous-time affine system, while the transition of the mode (the discrete state) is determined at each sampling time. It was shown, that the optimal control problem of SDPWA systems is reduced to an optimization over sequences of discrete states during each sampling time interval, together with sequences of continuous states to be passed at each sampling time. In physical systems, however, a discrete state such as a contact configuration between two rigid bodies, may change not only at prescribed sampling times but at any point of time according to the continuous state and the input of the system. This means that if such a continuous-time system is modeled as a discrete-time system or a sampled-data system, the discrete state (and therefore the dynamics) of the actual system and the one of the model may differ at some points in time. In order to avoid such phenomena, explicit consideration for holding the discrete state constant over the sampling time intervals should be included in the controller design.

In light of the above background, this paper proposes an approximate semi-offline implimentation of MPC for continuous-time piecewise affine (CTPWA) systems, which explicitly considers the constant mode condition during sampling time intervals. The method is called semi-offline, since the computation is effectively distributed over the offline phase and the online phase.

The rest of this paper is organized as follows. In Section 2, we formalize the finite-time optimal control problem of CTPWA systems and discuss basic control strategies. In Section 3, the notion of the controllability of a mode sequence is introduced. Then, in Section 4, we present an offline algorithm for computing the controllable set and the minimum transition cost associated with each of the mode sequences, and an online algorithm which performs MPC taking advantage of information precomputed offline. In Section 5, the proposed method is applied to a simple CTPWA system and numerical results are presented. Section 6 concludes the paper with several notes for future research.

II. Problem Formalization and Basic Strategies

In this section, we formalize the finite-horizon optimal control problem of CTPWA systems. First, consider the continuous-time piecewise affine (CTPWA) system

$$\Sigma : \quad \begin{array}{l} \dot{\boldsymbol{x}}(t) = A_{I(t)}\boldsymbol{x}(t) + a_{I(t)} + B_{I(t)}\boldsymbol{u}(t) \\ \text{if } \begin{bmatrix} \boldsymbol{x}(t) \\ \boldsymbol{u}(t) \end{bmatrix} \in \mathcal{S}_{I(t)} \end{array} \tag{1}$$

Y. Tazaki is with the Department of Mechanical and Environmental Informatics, Tokyo Institute of Technology, Tokyo, Japan tazaki@cyb.mei.titech.ac.jp

J. Imura is with the Department of Mechanical and Environmental Informatics, Tokyo Institute of Technology, Tokyo, Japan imura@mei.titech.ac.jp

where $\boldsymbol{x} \in \Re^n$ is the continuous state, $\boldsymbol{u} \in \Re^m$ is the continuous input, and $I \in \mathcal{M}(= \{1, 2, \ldots, M\})$ is the mode (the discrete state). Furthermore, $A_I \in \Re^{n\times n}$ and $B_I \in \Re^{n\times m}$ are constant matrices, and $a_I \in \Re^n$ is a constant vector. A subregion of $(\boldsymbol{x}, \boldsymbol{u})$ assigned to the mode I is given by the closed polyhedral cell

$$\mathcal{S}_I = \left\{ \begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{u}\end{bmatrix} \in \Re^{n+m} \;\middle|\; C_I \begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{u}\end{bmatrix} \le d_I \right\} \tag{2}$$

where $C_I \in \Re^{p_I \times (n+m)}$, $d_I \in \Re^{p_I}$ and the vector inequality $\boldsymbol{x} \le \boldsymbol{0}$ indicates $x_i \le 0$ for each element. It is assumed that each $\mathcal{S}_I$ has a pairwise disjoint interior but a common boundary, that is, $\operatorname{int}\mathcal{S}_I \cap \operatorname{int}\mathcal{S}_J = \emptyset$ and $\mathcal{S}_I \cap \mathcal{S}_J$ (if it exists) is included in the boundary of $\mathcal{S}_I(\mathcal{S}_J)$, where $\operatorname{int}\mathcal{S}$ expresses the interior of $\mathcal{S}$. In this paper, the well-posedness of the system is assumed to be guaranteed for brevity. Refer to [4][5][6] for a more detailed discussion. We further assume that the affine system assigned to each mode is controllable. For the above system, we consider the following finite horizon optimal control problem.

[Problem 1] Suppose the initial state $\boldsymbol{x}_s$, the target state $\boldsymbol{x}_f$, the positive definite matrix $R \in \Re^{m\times m}$, and the prediction horizon $T > 0$ are given. Then, for system (1), find an input sequence $\boldsymbol{u}$ that minimizes the cost function

$$J(\boldsymbol{u}) = \int_{t_0}^{t_0+T} \boldsymbol{u}(\tau)'\, R\, \boldsymbol{u}(\tau)\, d\tau \tag{3}$$

while satisfying $\boldsymbol{x}(t_0) = \boldsymbol{x}_s$ and $\boldsymbol{x}(t_0 + T) = \boldsymbol{x}_f$.

This paper applies the method for the optimal control problem of SDPWA systems proposed in [3] to CTPWA systems with some modifications. The method decomposes the whole controller design process into the following three steps.

Step 1 Determine the sequence of modes $\{I_0, I_1, \ldots, I_{N-1}\}$ where $I(t) = I_k \ {}^\forall t \in [t_k, t_{k+1}]$.

Step 2 Determine the sequence of waypoints (intermediate target states) $\{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ where $\boldsymbol{x}_0 = \boldsymbol{x}_s$, $\boldsymbol{x}_N = \boldsymbol{x}_f$, and $\boldsymbol{x}(t_k) = \boldsymbol{x}_k$.

Step 3 Generate a continuous-time input sequence $\boldsymbol{u}$, which drives the state from $\boldsymbol{x}_k$ to $\boldsymbol{x}_{k+1}$ through the region $\mathcal{S}_{I_k}$ for each $k = 0, 1, \ldots, N-1$.

Here, the sequence $\{t_k\}$ on the time axis is given by $t_k = kh$ where h is a positive constant. The symbol N refers to the number of prediction steps and is given by $N = T/h$.

Fig. 1 illustrates this strategy applied to a CTPWA system with 2 states, 1 input, and 4 modes. The state, the input, and the modes are denoted by $\boldsymbol{x} = (x_1, x_2)$, u, and $I \in \mathcal{M} = \{1, 2, 3, 4\}$, respectively. Each mode is assigned a cuboid region in the (x_1, x_2, u) space. In this example, four waypoints ($\boldsymbol{x}_0$, $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_3$) are placed on the state space. The waypoint $\boldsymbol{x}_1$ is placed on the boundary of mode 1 and mode 2, and $\boldsymbol{x}_2$ is placed on the boundary of mode 1 and mode 4. The mode between each subsequent pair of these waypoints is $I_0 = 2, I_1 = 1$ and $I_2 = 4$, respectively. Arrowed curves with solid lines express the trajectory of

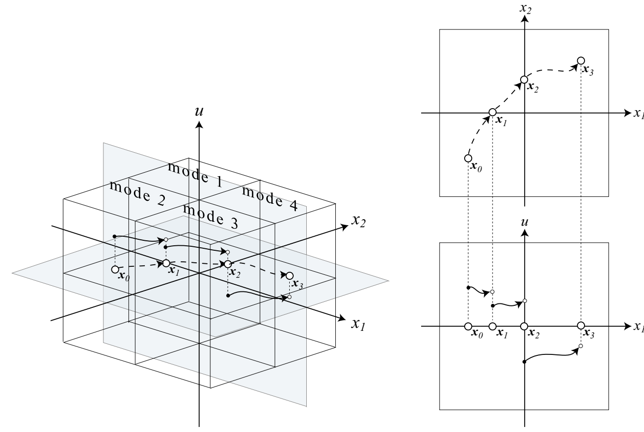


Fig. 1. Illustration of the control strategy applied to a simple CTPWA system. (a) Entire view of the (x_1, x_2, u) space. (b) Projected onto (x_1, x_2) plane (upper) and (x_1, u) plane (lower).

(x_1, x_2, u), while arrowed curves with dashed lines express the same trajectory projected onto the (x_1, x_2) plane.

It should be mentioned, that for continuous-time PWA systems, restricting the mode transitions to lie one a set of equally distributed points in time is unnecessary, and doing so could lead to merely a suboptimal solution. This time, however, we accept this artificial constraint in order to keep the problem tractable.

It should also be considered, that the pair of state and input must stay within the region of the specified mode during each time interval; that is, the following condition must hold.

$$\begin{aligned} &{}^\forall k = 0, 1, \ldots, N-1\,, \\ &C_{I_k} \begin{bmatrix}\boldsymbol{x}(t)\\ \boldsymbol{u}(t)\end{bmatrix} \le d_{I_k} \ {}^\forall t \in [t_k, t_{k+1}]. \end{aligned} \tag{4}$$

Now, let us for now assume that condition (4) holds, though it will be explicitly considered later. Then, for each time interval $[t_k, t_{k+1}]$, the plant is expressed as a time-invariant affine system corresponding to the mode I_k; $\dot{\boldsymbol{x}} = A_{I_k}\boldsymbol{x} + a_{I_k} + B_{I_k}\boldsymbol{u}$. This enables Step 3 to be formalized as a fixed terminal optimal control problem of an affine system as below.

[Problem 2] For an affine system $\dot{\boldsymbol{x}} = A_{I_k}\boldsymbol{x} + a_{I_k} + B_{I_k}\boldsymbol{u}$, find $\boldsymbol{u}$ that minimizes the cost function

$$J(\boldsymbol{u}) = \int_{t_k}^{t_{k+1}} \boldsymbol{u}(\tau)'\, R\, \boldsymbol{u}(\tau)\, d\tau \tag{5}$$

while satisfying $\boldsymbol{x}(t_k) = \boldsymbol{x}_k$, $\boldsymbol{x}(t_{k+1}) = \boldsymbol{x}_{k+1}$.

It is well known that the solution of Problem 2 is given in an explicit form as follows.

$$\boldsymbol{u}^*_{I_k}(t\,;\boldsymbol{x}_k,\boldsymbol{x}_{k+1}) = -R^{-1}B'_{I_k}e^{A'_{I_k}(h-t)}W_{I_k}(h)^{-1}\eta_{I_k}, \tag{6}$$

$$\boldsymbol{x}^*_{I_k}(t\,;\boldsymbol{x}_k,\boldsymbol{x}_{k+1}) = e^{A_{I_k}(t-h)}\left\{W_{I_k}(h-t)W_{I_k}(h)^{-1}\eta_{I_k} - \int_0^{h-t} e^{A_{I_k}\tau}d\tau\, a_{I_k} + \boldsymbol{x}_f\right\}, \tag{7}$$

$$J^*_{I_k}(\boldsymbol{x}_k,\boldsymbol{x}_{k+1}) = \eta'_{I_k}W_{I_k}(h)^{-1}\eta_{I_k}. \tag{8}$$

Here, $W_{I_k}(t) = \int_0^t e^{A_{I_k}\tau}B_{I_k}R^{-1}B'_{I_k}e^{A'_{I_k}\tau}d\tau$ and $\eta_{I_k} = e^{A_{I_k}h}\boldsymbol{x}_k - \boldsymbol{x}_{k+1} + \int_0^h e^{A_{I_k}\tau}d\tau\, a_{I_k}$. Rearranging (6) and (7) yields

$$\boldsymbol{u}^*_{I_k}(t\,;\boldsymbol{x}_k,\boldsymbol{x}_{k+1}) = E_{I_k}(t)\,\boldsymbol{x}_k + F_{I_k}(t)\,\boldsymbol{x}_{k+1} + \zeta_{I_k}(t), \tag{9}$$

$$\boldsymbol{x}^*_{I_k}(t\,;\boldsymbol{x}_k,\boldsymbol{x}_{k+1}) = G_{I_k}(t)\,\boldsymbol{x}_k + H_{I_k}(t)\,\boldsymbol{x}_{k+1} + \xi_{I_k}(t) \tag{10}$$

where $E_{I_k}(t), F_{I_k}(t), G_{I_k}(t)$, and $H_{I_k}(t)$ are time varying matrices, and $\zeta_{I_k}(t), \xi_{I_k}(t)$ are time-varying vectors. As shown above, so far as the mode invariance condition (4) holds, the solution to Problem 2 (and hence, to Step 3) is given by the explicit function of waypoints $\boldsymbol{x}_k$, $\boldsymbol{x}_{k+1}$, mode I_k, and time t.

We now turn our attention to considering condition (4) in the controller design. The most straightforward way of doing this is by treating the condition in Step 3, that is, adding (4) into Problem 2, as a constraint. In this way, however, Problem 2 is no longer expected to be given an explicit solution like (6) and (7). In light of this observation, we give (6) and (7) as a solution to Step 3, and keeping this in mind, guarantee condition (4) to hold in Step 1 and Step 2. First, substitute (9) and (10) into (4). Then we have

$$C_{I_k}\begin{bmatrix} E_{I_k}(t) & F_{I_k}(t) \\ G_{I_k}(t) & H_{I_k}(t)\end{bmatrix}\begin{bmatrix}\boldsymbol{x}_k \\ \boldsymbol{x}_{k+1}\end{bmatrix} \le d_{I_k} - C_{I_k}\begin{bmatrix}\zeta_{I_k}(t) \\ \xi_{I_k}(t)\end{bmatrix}. \tag{11}$$

This recasts condition (4) into a constraint with respect to waypoints $\boldsymbol{x}_k$, $\boldsymbol{x}_{k+1}$, and mode I_k. Since condition (11) has nonlinear terms on time t, we approximate (11) by a time-invariant linear inequality by means of discretization. Introducing a sequence of time points $t_{ki} = t_k + (h/N_d)\,i \ \ (i = 0, 1, \ldots, N_d)$ over the interval $[t_k, t_{k+1}]$ and evaluating (11) at each of these points yields

$$\bar{C}_{I_k}\begin{bmatrix}\boldsymbol{x}_k \\ \boldsymbol{x}_{k+1}\end{bmatrix} \le \bar{d}_{I_k}, \tag{12}$$

$$\bar{C}_{I_k} = \begin{bmatrix} C_{I_k}\begin{bmatrix} E_{I_k}(t_k) & F_{I_k}(t_k) \\ G_{I_k}(t_k) & H_{I_k}(t_k)\end{bmatrix} \\ C_{I_k}\begin{bmatrix} E_{I_k}(t_{k1}) & F_{I_k}(t_{k1}) \\ G_{I_k}(t_{k1}) & H_{I_k}(t_{k1})\end{bmatrix} \\ \vdots \\ C_{I_k}\begin{bmatrix} E_{I_k}(t_{k+1}) & F_{I_k}(t_{k+1}) \\ G_{I_k}(t_{k+1}) & H_{I_k}(t_{k+1})\end{bmatrix}\end{bmatrix},$$

$$\bar{d}_{I_k} = \begin{bmatrix} d_{I_k} - C_{I_k}\begin{bmatrix}\zeta_{I_k}(t_k) \\ \xi_{I_k}(t_k)\end{bmatrix} \\ d_{I_k} - C_{I_k}\begin{bmatrix}\zeta_{I_k}(t_{k1}) \\ \xi_{I_k}(t_{k1})\end{bmatrix} \\ \vdots \\ d_{I_k} - C_{I_k}\begin{bmatrix}\zeta_{I_k}(t_{k+1}) \\ \xi_{I_k}(t_{k+1})\end{bmatrix}\end{bmatrix}$$

where N_d is a positive constant. Although (12) is merely a necessary condition to (11), it can be made arbitarily close to (11) by increasing N_d.

Based on the above strategy, Problem 1 is transformed into an approximate optimization problem of mode sequences and waypoint sequences formalized as below.

[Problem 3] Suppose the initial state $\boldsymbol{x}_s$, the target state $\boldsymbol{x}_f$, and the number of prediction steps N are given. Then, for system (1), find a mode sequence $\mathcal{I} = \{I_0, I_1, \ldots, I_{N-1}\}$ and a waypoint sequence $\mathcal{X} = \{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ minimizing the cost function

$$J(\mathcal{I},\mathcal{X}) = \sum_{k=0}^{N-1} J^*_{I_k}(\boldsymbol{x}_k,\boldsymbol{x}_{k+1}) \tag{13}$$

while satisfying

$$\boldsymbol{x}_0 = \boldsymbol{x}_s,\ \boldsymbol{x}_N = \boldsymbol{x}_f, \tag{14}$$

$$\bar{C}_{I_k}\begin{bmatrix}\boldsymbol{x}_k \\ \boldsymbol{x}_{k+1}\end{bmatrix} \le \bar{d}_{I_k} \quad (k = 0, 1, \ldots, N-1). \tag{15}$$

Notice that Problem 3 requires the optimization of mode sequence, while at the same time optimizing the waypoint sequence for a given mode sequence. The waypoint optimization subproblem is a quadratic programming problem. Let us define this subproblem for future reference as below.

[Problem 4] Suppose the system (1), $\boldsymbol{x}_s$, $\boldsymbol{x}_f$, and the mode sequence $\mathcal{I} \in \mathcal{M}^N$ are given. Then, find the optimal waypoint sequence $\mathcal{X} = \{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ minimizing the cost function (13) while satisfying (14) (15).

The model predictive controller should solve Problem 3 at each sampling time t with $t_0 = t$ and $\boldsymbol{x}_s = \boldsymbol{x}(t)$. Here, we assume for simplicity that the sampling period of MPC is equal to the partition length h , although these may differ in general. Problem 3 should be solved in a shorter time than the sampling period in order to perform the MPC. With this requirement and keeping the discussion given in the introduction in mind, the method proposed in this paper precomputes the controllable set and the lower bound of the transition cost for every N-step mode sequence. The notion of controllability associated with a mode sequence will be defined at the beginning of the next section. Moreover, the minimum transition cost is given as a solution to the relaxed version of Problem 4 as below.

[Problem 4'] For system (1) and a given mode sequence $\mathcal{I} \in \mathcal{M}^N$, find the optimal waypoint sequence $\mathcal{X} =$

$\{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ minimizing cost function (13) while satisfying (15).

The only difference between Problem 4 and Problem 4' is that constraint (14) has been removed. Thus the minimum cost of Problem 4' provides a good lower bound to Problem 4, and moreover, Problem 4' is suitable for precomputation since it is independent of specific $\boldsymbol{x}_s$ and $\boldsymbol{x}_f$.

III. Controllable Sets of Mode Sequences

In this section, we introduce the notion of controllability associated with a mode sequence and derive some useful properties. To begin with, let us denote the set of pairs of waypoints $(\boldsymbol{x}_0, \boldsymbol{x}_1)$ subject to (12) for a given mode I as below.

$$\mathcal{F}(I) := \left\{ \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{x}_1 \end{bmatrix} \in \Re^{2n} \;\middle|\; \bar{C}_I \begin{bmatrix} \boldsymbol{x}_0 \\ \boldsymbol{x}_1 \end{bmatrix} \le \bar{d}_I \right\}. \tag{16}$$

Similarly, denote the set of waypoint sequences $\mathcal{X} = \{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ satisfying (12) according to a given mode sequence $\mathcal{I} = \{I_0, I_1, \ldots, I_{N-1}\}$ as follows.

$$\mathcal{F}(\mathcal{I}) := \Big\{ \mathcal{X} = (\boldsymbol{x}_0' \, \boldsymbol{x}_1' \, \ldots \, \boldsymbol{x}_N')' \in \Re^{(N+1)n} \;\Big|\; \begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{x}_{k+1} \end{bmatrix} \in \mathcal{F}(I_k) \;\; (k = 0, 1, \ldots, N-1) \Big\}. \tag{17}$$

Next, consider the projection of $\mathcal{F}(\mathcal{I})$ onto the space of pairs of the first and the last waypoints:

$$\mathcal{E}(\mathcal{I}) := \pi^{0,N}(\mathcal{F}(\mathcal{I})). \tag{18}$$

Here, the function $\pi^{0,N}(\mathcal{X})$ projects the set of N-step waypoint sequences $\mathcal{X}$ onto the space of pairs of the first and the last waypoints. That is,

$$\pi^{0,N}(\mathcal{X}) := \Big\{ [\boldsymbol{x}_0' \, \boldsymbol{x}_N']' \in \Re^{2n} \;|\; {}^{\exists}\boldsymbol{x}_i \in \Re^n \; (i = 1, 2, \ldots, N-1) \text{ s.t. } [\boldsymbol{x}_0' \, \boldsymbol{x}_1' \, \ldots \, \boldsymbol{x}_N']' \in \mathcal{X} \subset \Re^{(N+1)n} \Big\}. \tag{19}$$

Clearly, in the case of 1 step, the set $\mathcal{E}(I)$ is equivalent to $\mathcal{F}(I)$. By definition, if $(\boldsymbol{x}_0, \boldsymbol{x}_N) \in \mathcal{E}(\mathcal{I})$ holds for a mode sequence $\mathcal{I} = \{I_0, I_1, \ldots, I_{N-1}\}$, then it follows that there exists an intermediate waypoint sequence $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_{N-1}$ that satisfies (12). This leads us to define the controllability of mode sequences as follows.

[Definition 1] A mode sequence $\mathcal{I} \in \mathcal{M}^N$ is controllable iff $\mathcal{E}(\mathcal{I}) \neq \emptyset$. Moreover, if a pair $(\boldsymbol{x}_s, \boldsymbol{x}_f)$ satisfies $(\boldsymbol{x}_s, \boldsymbol{x}_f) \in \mathcal{E}(\mathcal{I})$, then the pair $(\boldsymbol{x}_s, \boldsymbol{x}_f)$ is said to be controllable with respect to $\mathcal{I}$.

In the sense of Definition 1, we call $\mathcal{E}(\mathcal{I})$ the controllable set of $\mathcal{I}$. Now, let us define the following set operation.

[Definition 2] For a set $\mathcal{E}_0 \subset \Re^{2n}$ and $\mathcal{E}_1 \subset \Re^{2n}$,

$$\mathcal{E}_0 \cdot \mathcal{E}_1 := \pi^{0,2}((\mathcal{E}_0 \times \Re^n) \cap (\Re^n \times \mathcal{E}_1)). \tag{20}$$

Then, the controllable set of a mode sequence $\mathcal{I}$ and the controllable sets of its subsequences $\mathcal{I}_0, \mathcal{I}_1$ are related in terms of (20). That is,

$$\mathcal{E}(\mathcal{I}) = \mathcal{E}(\mathcal{I}_0) \cdot \mathcal{E}(\mathcal{I}_1) \tag{21}$$

where $\mathcal{I} = \{\mathcal{I}_0, \mathcal{I}_1\}$. Moreover, for a mode sequence $\mathcal{I} = \{I_0, I_1, \ldots, I_{N-1}\}$, the following relation holds.

$$\mathcal{E}(\mathcal{I}) = \mathcal{E}(I_0) \cdot \mathcal{E}(I_1) \cdot \, \ldots \, \cdot \mathcal{E}(I_{N-1}). \tag{22}$$

From (16) and (17), $\mathcal{F}(\mathcal{I})$ is given by a linear inequality on $\Re^{(N+1)n}$. It follows that $\mathcal{E}(\mathcal{I})$ is given by a linear inequality on $\Re^{2n}$: we denote this by

$$\mathcal{E}(\mathcal{I}) = \langle H_{\mathcal{I}}, K_{\mathcal{I}} \rangle \,. \tag{23}$$

Here, $\langle H, K \rangle$ is defined as

$$\langle H, K \rangle := \left\{ \; \boldsymbol{z} \in \Re^s \;\middle|\; H\boldsymbol{z} \le K \; \right\} \tag{24}$$

where s is the number of columns of H. It follows that once $H_{\mathcal{I}}$ and $K_{\mathcal{I}}$ are known, one can test the controllability of a pair $(\boldsymbol{x}_s, \boldsymbol{x}_f)$ with respect to $\mathcal{I}$ by simply evaluating $H_{\mathcal{I}}(\boldsymbol{x}_s' \, \boldsymbol{x}_f')' \le K_{\mathcal{I}}$.

For numerically computing $H_{\mathcal{I}}$ and $K_{\mathcal{I}}$, this paper employs a projection algorithm proposed in [7] and [8]. This algorithm takes a linear inequality representation $\langle H, K \rangle$ as an input, and outputs the linear inequality representation $\langle H_d, K_d \rangle$ of the projection $\{\boldsymbol{x} \in \Re^d \,|\, {}^{\exists}\boldsymbol{y} \in \Re^{s-d} \text{ s.t. } H(\boldsymbol{x}' \, \boldsymbol{y}')' \le K\}$. Let us denote this by $\langle H_d, K_d \rangle = projection(\langle H, K \rangle, d)$. This projection algorithm repeatedly finds an extreme point of the projection by solving a linear programming problem on $\Re^s$, until every extreme point is found. This indicates that directly projecting $\mathcal{F}(\mathcal{I})$ to $\mathcal{E}(\mathcal{I})$ is likely to become prohibitively slow as N grows. To overcome this problem, (21) and (22) provide a way to reduce the projection of $\Re^{(N+1)n}$ to $\Re^{2n}$ into the projection of $\Re^{3n}$ to $\Re^{2n}$. More precisely, if the controllable sets of two mode sequences $\mathcal{I}_0$ and $\mathcal{I}_1$ are given, the controllable set of $\mathcal{I} = \{\mathcal{I}_0, \mathcal{I}_1\}$ is obtained by means of (21). Furthermore, notice that (21) is rewritten in terms of linear inequality representations as

$$\begin{aligned} \langle H, K \rangle &= \langle [H_{01} \, H_{02}] \,, K_0 \rangle \cdot \langle [H_{11} \, H_{12}] \,, K_1 \rangle \\ &= projection\left(\left\langle \begin{bmatrix} H_{01} & O & H_{02} \\ O & H_{12} & H_{11} \end{bmatrix}, \begin{bmatrix} K_0 \\ K_1 \end{bmatrix} \right\rangle, 2n \right) \end{aligned} \tag{25}$$

where $\mathcal{E}(\mathcal{I}) = \langle H, K \rangle$, $\mathcal{E}(\mathcal{I}_0) = \langle [H_{01} \, H_{02}], K_0 \rangle$, and $\mathcal{E}(\mathcal{I}_1) = \langle [H_{11} \, H_{12}], K_1 \rangle$. With this in place, an algorithm which computes the controllable sets of mode sequences, by extending the length of mode sequences step by step starting from 1 step, will be presented in the next section.

IV. Algorithms of Semi-offline MPC

In this section, we construct a model predictive control law based on the discussions of the preceding sections. The outline of the method is as follows: In the offline phase, for every mode sequence with length from 1 to N, compute the linear inquality representation of the controllable set and the minimum transition cost. Then at each sampling time t

in the online phase,

i) enumerate all N-step mode sequences controllable to the state pair $(\boldsymbol{x}(t), \boldsymbol{x}_f)$.
ii) determine the optimal waypoint sequence by solving Problem 4 for each of the enumerated mode sequences.
iii) generate a continuous-time input $\boldsymbol{u}^*(\boldsymbol{x}(t), \boldsymbol{x}_1)$ where $\boldsymbol{x}_1$ is the first element of the optimal waypoint sequence obtained in ii).

We call the method "semi-offline" since the optimization of waypoints is left to be processed on line.

The algorithm given below computes the controllable sets and the minimum transition costs of mode sequences offline.

[Algorithm 1]

Inputs: Σ, R, h, N

Outputs: Θ_k $(k = 1, 2, \ldots, N)$: set of controllable k-step mode sequences.
$\langle H_{\mathcal{I}}, K_{\mathcal{I}} \rangle$: linear inequality representation of the controllable set associated with the mode sequence $\mathcal{I}$.
$J(\mathcal{I})$: minimum transition cost of the mode sequence $\mathcal{I}$.

Compute controllable sets of mode sequences.

1: $\Theta_1 \leftarrow \emptyset$.
2: for each $I \in \mathcal{M}$
3: if $\mathcal{F}(I) \neq \emptyset$ then
4: add I into Θ_1.
5: $(H_{\{I\}}, K_{\{I\}}) \leftarrow (\bar{C}_I, \bar{d}_I)$.
6: end
7: end
8: for each $k \in [2, N]$
9: $\Theta_k \leftarrow \emptyset$.
10: for each $\mathcal{I} \in \Theta_{k-1}$
11: for each $J \in \mathcal{M}$
12: $\mathcal{J} := \{\mathcal{I}, J\}$.
13: $\langle H_{\mathcal{J}}, K_{\mathcal{J}} \rangle \leftarrow \langle H_{\mathcal{I}}, K_{\mathcal{I}} \rangle \cdot \langle H_{\{J\}}, K_{\{J\}} \rangle$ (25).
14: If $\langle H_{\mathcal{J}}, K_{\mathcal{J}} \rangle \neq \emptyset$, then add $\mathcal{J}$ into Θ_k.
15: end
16: end
17: end

Compute minimum transition cost of mode sequences.

18: for each $\mathcal{I} \in \Theta_k$ $(k \in [1, N])$
19: Solve Problem 4' for $\mathcal{I}$ then store the minimum cost to $J(\mathcal{I})$.
20: end

The symbol $\leftarrow$ refers to substitution from the right hand side to the left hand side.

The following algorithm implements the model predictive controller utilizing the output of Algorithm 1.

[Algorithm 2]

Inputs: $\boldsymbol{x}_s$, $\boldsymbol{x}_f$, h, N

TABLE I
COMPARISON OF THE NUMBER OF MODE SEQUENCES.

steps	1	2	3	4	5	6
controllable mode sequences	7	19	46	107	247	583
possible mode sequences	7	19	53	149	421	1193

0: $t \leftarrow t_0$, $\boldsymbol{x}(t_0) \leftarrow \boldsymbol{x}_s$.
1: $\Theta \leftarrow \emptyset$.
2: for each $\mathcal{I} \in \Theta_N$
3: if $H_{\mathcal{I}}(\boldsymbol{x}(t)'\, \boldsymbol{x}_f')' \leq K_{\mathcal{I}}$ then add $\mathcal{I}$ into Θ.
4: end
5: Sort Θ in ascending order by using $J(\mathcal{I})$ as a comparison key.
6: $J^* \leftarrow \infty$.
7: for each $\mathcal{I} \in \Theta$
8: if $J(\mathcal{I}) < J^*$ then
9: solve Problem 4 for $\boldsymbol{x}(t)$, $\boldsymbol{x}_f$, and $\mathcal{I}$. Denote the obtained minimum cost by J.
10: if $J < J^*$ then $J^* \leftarrow J$.
11: end
12: end
13: Denote by $\{I_0, I_1, \ldots, I_{N-1}\}$ and $\{\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ the mode sequence and the waypoint sequence that give the optimal cost J^*. Generate continuous time input $\boldsymbol{u}^*_{I_0}(\boldsymbol{x}(t), \boldsymbol{x}_1)$ by (6) and apply it to the plant during the sampling time interval h.
14: $t \leftarrow t + h$.
15: Go to Step 1.

V. Numerical Examples

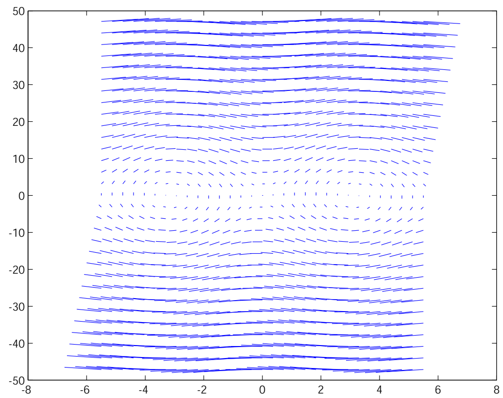
This section shows numerical examples. The CTPWA system shown here is obtained by a piecewise linear approximation of a pendulum. More precisely, the original nonlinear state equation of the pendulum is discribed as

$$\dot{\boldsymbol{x}} = \begin{bmatrix} \dot{q} \\ \frac{g}{l}\sin q \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{ml^2} \end{bmatrix} u \quad \boldsymbol{x} = \begin{bmatrix} q \\ \dot{q} \end{bmatrix} \tag{26}$$

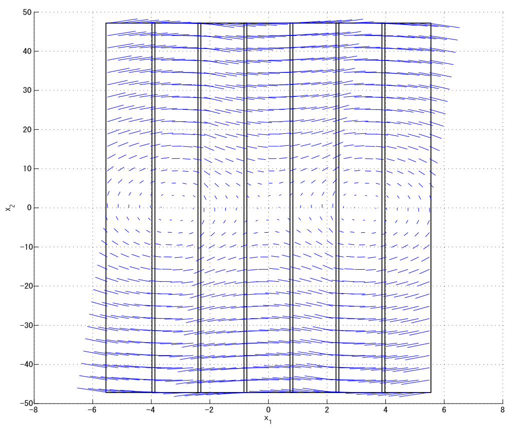
where q [rad] is the joint angle, $\dot{q}$ [rad/s] is the joint velocity, and u [Nm] is the joint torque. Moreover, $m = 0.1$[kg] is the point mass attached to the tip of the arm, $l = 0.3$[m] is the length of the arm, and $g = 9.8$[m/s^2] is the gravitational acceleration. For this system, consider a region given by $\{(q, \dot{q}, u) \in \Re^3 \mid q \in [-(7/4)\pi, (7/4)\pi] \wedge \dot{q} \in [-15\pi, 15\pi] \wedge u \in [-0.8mgl, 0.8mgl]\}$ divided it into 7 subregions with equal widths along the q-axis. Then, assign a time-invariant affine system to each subregion, which is obtained by calculating the taylor series expansion of the right hand side of (26) at the center of the subregion and ignoring the nonlinear terms:

$$\dot{\boldsymbol{x}} = \begin{bmatrix} 0 & 1 \\ \frac{g}{l}\cos\bar{q} & 0 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 0 \\ \frac{g}{l}(\sin\bar{q} - \bar{q}\cos\bar{q}) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{ml^2} \end{bmatrix} u \tag{27}$$

where $\bar{q}$ refers to the value of q at the center of each subregion. Fig. 2 shows the phase map of the original pendulum system and that of the CTPWA system.



(a)



(b)

Fig. 2. phase map of (a) the original pendulum system and (b) the piecewise linearized system.

First, Algorithm 1 is tested for the above CTPWA system. Here, the sampling period h is given by 0.3[s] and the number of prediction steps N is given by 6. The total amount of time required for this computation is 1468[s] with Intel Xeon Processor 2GHz with 2GBytes of memory and the output size is 9.85MBytes. Table I shows the comparison between the number of controllable mode sequences and the number of all possible mode sequences.

Fig. 3 shows the result of model predictive control (Algorithm 2) performed on (27) for 10 steps by numerical simulation. The initial state $\boldsymbol{x}_s$ and the target state $\boldsymbol{x}_f$ is set to $[\pi, 0]$ and $[0, 0]$, respectively. Each row of Fig. 3 shows, from the upper most row, the trajectory of $q(t)$, $\dot{q}(t)$, $u(t)$, and $I(t)$, respectively. It is observed in the figure that the pendulum is swung in the opposite direction at the beginning, in order to gain enough velocity for reaching the target state under input limitation. The average time required in each sampling time for processing Step 1 through Step 6 of Algorithm 2 is 116[ms]: the computation time is treated as 0, however, in the numerical simulation. It should be noted that the precomputed information is not restricted to a specific

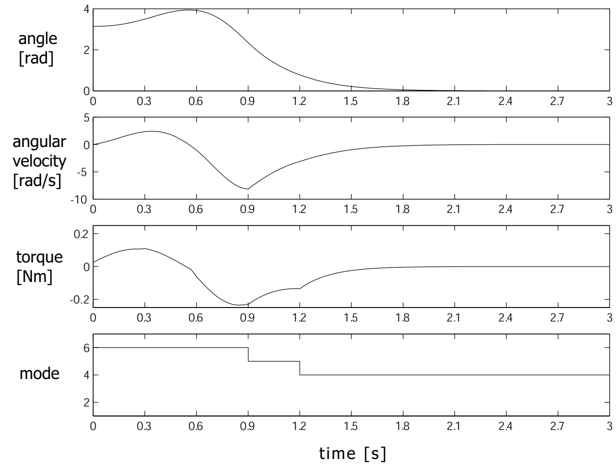


Fig. 3. trajectory of the state, the input, the mode.

initial state or target state, but is able to handle arbitrary pairs of initial and target states, as long as the pair is controllable with respect to at least one of the N-step mode sequences.

VI. Conclusion

This paper has presented semi-offline model predictive control for continuous-time piecewise affine systems. The proposed method computes controllable sets and minimum transition costs for mode sequences in the offline phase, then at each sampling time in the online phase, determines the optimal waypoint sequence by solving a quadratic programming problem for each controllable mode sequence. By distributing the computation over the offline phase and the online phase, real-time control is achieved while saving the size and the complexity of the precomputed structure in an acceptable level. The proposed method has been applied to a simple CTPWA system, obtained by piecewise linear approximation of a pendulum, to demonstrate the effectiveness of the proposed approach.

References

[1] A. Bemporad, M. Morari, V. Dua and E.N. Pistikopoulos: The Explicit Linear Quadratic Regulator for Constrained Systems, Automatica, 38-1, 3/20(2002)

[2] A. Bemporad, F. Borrelli and M. Morari: Piecewise Linear Optimal Controllers for Hybrid Systems, Proc. of the 2000 American Control Conference, 1190/1194(2000)

[3] J.Imura : Optimal control of sampled-data piecewise affine systems, Automatica, Vol.40, No.4, pp.661-669, 2004.

[4] J. Imura and A.J. van der Schaft. Characterization of well-posedness of piecewise linear systems. IEEE Trans. on Automatic Control, Vol.45, No.9, 1600/1619(2000).

[5] J. Imura. Classification and stabilizability analysis of bimodal piecewise affine systems. Int. J. Robust and Nonlinear Control, Vol.12, No.10, 897/926(2002).

[6] J. Imura. Well-posedness analysis of switch-driven hybrid systems. IEEE Trans. on Automatic Control Vol.48, No.11, 1926-1935(2003).

[7] S. Azuma and J. Imura. A Probabilistic Approach to Controllability/Reachability Analysis of Hybrid Systems. Proceedings of the 43rd IEEE Conference on Decision and Control (CDC04), Bahamas, December 14-17, pp. 485-490 (2004)

[8] Multi-Parametric Toolbox for MATLAB. http://control.ee.ethz.ch/˜mpt/