The 2009 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 11-15, 2009 St. Louis, USA

Synthesis of Output Feedback Control for Motion Planning Based on LTL Specifications

Min Wu, Gangfeng Yan, Zhiyun Lin, and Ying Lan

Abstract— **In the paper, we study the motion planning problem of a mobile robot in the plane. The goal is to design output feedback control such that the resulting path of a mobile robot satisfies desired linear temporal logic (LTL) specifications. Our control strategy is divided into a local output feedback control problem and a supervisory control for LTL specifications. For the former one, we design output feedback control laws to ensure that output trajectories either remain in a simplex, or leave the simplex and enter an adjacent simplex in finite time. For the latter, we construct a transition system based on reachability and search for feasible paths that satisfy the LTL specifications. In this way, a piecewise affine output feedback control is obtained to solve the motion planning problem. A simulation result is presented to illustrate our approach.**

I. Introduction

Motion planning and control of robots in complex environments are fundamental problems that have received a lot of attention. The work on this problem can be roughly classified into two categories [1]. The first one focuses on the complexity of environment while ideally assuming the robot has no motion constraints. Examples are [2] and [3] where Voronoi diagrams, potential fields, and navigation functions are used to deal with complex environments. The second one considerers specific dynamic model for robots in an unconstrained state-space. For this, more effort is on addressing the difficulties caused by the robot dynamics using varying approaches such as differential geometric approach [4] and discontinuous control laws [5]. An extensive discussion of the problem can be found in [6].

Traditionally, the path planning problem for mobile robots considers specifications of the form "move from an initial position I to a goal position G while staying within a region R". Such specifications can be formulated as temporal logics [7]. The applicability of temporal logics in robotics was advocated as far back as [8]. More recently, model checking approaches have been used for motion planning in order to satisfy temporal logic specifications. Based on the earlier work of [9], symbolic control approaches are used to compute a partition of the state space, design a path at the discreet level, and refine it using local continuous controller in each domain of the partition. However, this approach has been limited to the case that the system has fully actuated dynamics with full state available [10]–[13].

In this paper, we borrow ideas from formal analysis of hybrid systems and extend the method firstly proposed in [14]. Since many dynamical systems can in a first approximation be described by piecewise affine hybrid systems [15]–[19], we consider a general (piecewise) affine system model for robots. Our approach consists of two main ingredients: a local output feedback control strategy and a supervisory control for LTL specifications. First we design output feedback control law to ensure that output trajectories either leave a simplex and enter an adjacent simplex in finite time, or remain in a simplex. Next, we construct a transition system based on the former results and search for feasible paths that satisfy the LTL specifications. In this way, a piecewise affine output feedback is obtained to solve the problem. Comparing with state feedback, output feedback control has the following advantages. First, we can focus on the robot's position, which is the main concern in most motion planning problems, rather than irrelevant states (e.g. heading angle, velocity, and acceleration). Second, output feedback takes less measurement information and less implementation cost as some states are not easy to obtain.

The paper is organized as follows. In the next section, we formulate the motion planning problem after giving some necessary notations. Then in Section III, a local control strategy and a supervisory control law are proposed. A simulation result is presented in Section IV. The paper ends with concluding remarks and brief discussions of future research in Section V.

II. Preliminaries

In this section, we first present some background materials and then introduce the problem.

A. Polytopes, Triangulation, and Affine Systems

Notations $\mathbb{R}$ and $\mathbb{N}$ are used to represent the set of real numbers and natural numbers, respectively. We use $\mathrm{co}(v_1, \ldots, v_n)$ to denote the *convex hull* of points $v_1, \ldots, v_n$. Let $\mathcal{P}$ be an n-dimensional *polytope* in $\mathbb{R}^n$. It can be written as the intersection of d half spaces, where d is the least number required. That is,

$$\mathcal{P} = \bigcap_{i=1}^{d} \{x \in \mathbb{R}^n | n_i^T x \leq \gamma_i\}, \quad (1)$$

This work is supported by National Natural Science Foundation of China under Grant 60504024, 60874050, and 60875074, Zhejiang Provincial Natural Science Foundation of China under Grant Y106010, Specialized Research Fund for Doctoral Program of Higher Education (SRFDP), China, under Grant 20060335022, and "151 Talent Project" of Zhejiang Province (No. 05-3-1013 and No. 06-2-034).

The authors are with Asus Intelligent Systems Lab, Department of Systems Science and Engineering, Zhejiang University, Hangzhou 310027, P. R. China. Email: linz@zju.edu.cn

978-1-4244-3804-4/09/$25.00 ©2009 IEEE

where n_i is a unit normal vector and γ_i is a constant. The set $\{x \in \mathbb{R}^n | n_i^T x = \gamma_i\}$ is called a *supporting hyperplane* of $\mathcal{P}$. A *facet* of polytope $\mathcal{P}$ is the intersection of $\mathcal{P}$ with one of its supporting hyperplanes, which is of $(n-1)$-dimension. That is,

$$\mathcal{F}_i = \{x \in \mathcal{P} | n_i^T x = \gamma_i\}, \quad i = 1, \ldots, d.$$

Alternatively, the polytope $\mathcal{P}$ can be viewed as the convex hull of its vertices. We denote $\mathrm{vert}(\mathcal{P})$ the set of vertices of $\mathcal{P}$. A *simplex* is an n-dimensional polytope with $n+1$ vertices. We label the vertices of a simplex $v_1, \ldots, v_{n+1}$, and label its facets $\mathcal{F}_1, \ldots, \mathcal{F}_{n+1}$ such that $v_i \notin \mathcal{F}_i$.

A *triangulation* of a polytope E, denoted by $\mathbb{S} = \{\mathcal{S}_1, \mathcal{S}_2, \ldots, \mathcal{S}_N\}$, is defined as follows:
(1) For all $\mathcal{S}_i \in \mathbb{S}$, $\mathcal{S}_i$ is a full dimensional simplex;
(2) For all $\mathcal{S}_i$, $\mathcal{S}_j \in \mathbb{S}$, their intersection is either the convex hull of their common vertices or empty;
(3) $\mathcal{S}_1 \cup \mathcal{S}_2 \cup \cdots \cup \mathcal{S}_N = E$.

An affine control system is described as

$$\begin{cases} \dot{x} &= Ax + a + Bu, \\ y &= Cx + d, \end{cases} \tag{2}$$

where $A \in \mathbb{R}^{n \times n}$, $a \in \mathbb{R}^n$, $B \in \mathbb{R}^{n \times s}$, $C \in \mathbb{R}^{m \times n}$, and $d \in \mathbb{R}^m$. Without loss of generality, we assume $\mathrm{rank}(C) = m$.

Correspondingly, we denote an autonomous affine system with output by

$$\begin{cases} \dot{x} &= Ax + a, \\ y &= Cx + d. \end{cases} \tag{3}$$

Let $x(t, x_0)$ be the state trajectory of (3) starting at x_0. Let $Y(t, y_0)$ denote the set of output trajectories of (3) with any initial state x_0 satisfying $Cx_0 + d = y_0$, i.e.,

$$Y(t, y_0) := \{Cx(t, x_0) + d | x_0 \in \mathcal{P} \text{ and } Cx_0 + d = y_0\}.$$

And $y(t, y_0)$ is used to represent an element of $Y(t, y_0)$. When the initial output y_0 can be ignored, we just use notations $Y(t)$ and $y(t)$.

B. Transition Systems and Linear Temporal Logic

A *transition system* is a tuple $T = (Q, Q_0, \to, \Pi, \models)$, where Q is a set of states, $Q_0 \subset Q$ is a set of initial states, $\to \subset Q \times Q$ is a transition relation, Π is a finite set of atomic propositions, and $\models \subset Q \times \Pi$ is a satisfaction relation.

In the paper, we assume that the transition system is *finite* (that is, the state set Q is finite). For any proposition $\pi \in \Pi$, we define $[\pi] = \{q \in Q | q \models \pi\}$ as the set of states satisfying it. Conversely, for any state $q \in Q$, let $\Pi_q = \{\pi \in \Pi | q \models \pi\}$ ($\Pi_q \in 2^\Pi$) denote the set of all atomic propositions satisfied at q. A *path* of a transition system T starting from q is an infinite sequence $r = r(1)r(2)r(3)...$ with the property that $r(1) = q$, $r(i) \in Q$, and $(r(i), r(i+1)) \in \to$ for all $i \geq 1$. A path $r = r(1)r(2)r(3)\ldots$ defines a *word* $w = w(1)w(2)w(3)\ldots$, where $w(i) = \Pi_{r(i)}$.

A *linear temporal logic* (LTL) formula over a set of atomic proportions $\Pi = \{\pi_1, \ldots, \pi_N\}$ is recursively defined as follows (syntax):
(1) Every atomic proposition π_i is a formula;
(2) If ϕ_1 and ϕ_2 are formulas, then $\phi_1 \vee \phi_2$, $\neg\phi_1$, $\phi_1 \mathcal{U} \phi_2$ are also formulas.

The semantics of LTL formulas are given over words of a transition system T. The satisfaction of a formula ϕ at position $i \in \mathbb{N}$ of a word w, denoted by $w(i) \models \phi$, is defined recursively as follows:
(1) $w(i) \models \pi$, if $\pi \in w(i)$;
(2) $w(i) \not\models \pi$, if $\pi \notin w(i)$;
(3) $w(i) \models \neg\phi$, if $w(i) \not\models \phi$;
(4) $w(i) \models \phi_1 \vee \phi_2$, if $w(i) \models \phi_1$ or $w(i) \models \phi_2$;
(5) $w(i) \models \phi_1 \mathcal{U} \phi_2$, if there exists a $j \geq i$ such that $w(j) \models \phi_2$ and for all k, $(i \leq k < j,)$ we have $w(k) \models \phi_1$.

A word w satisfies an LTL formula ϕ, written as $w \models \phi$, if $w(1) \models \phi$.

The boolean constants $\top$ and $\bot$ are defined as $\top = \pi \vee \neg\pi$ and $\bot = \neg\top$, respectively. Given negation $\neg$ and disjunction $\vee$, we can define conjunction $\wedge$, implication $\Rightarrow$, and equivalence $\Leftrightarrow$. Furthermore, we can also derive additional temporal operators such as eventuality $\diamond\phi = \top\mathcal{U}\phi$ and safety $\Box\phi = \neg\diamond\neg\phi$. Note that our syntax does not contain the common next operator $\bigcirc\phi$.

C. Problem Formulation

In this paper, we consider a robot moving in a polygonal environment E. The environment is not necessary to be convex, but it should be able to be viewed as a collection of polytopes. The dynamics of the robot is given or approximated in the form of (2) where its output is the position. The goal of this paper is to construct an output feedback $u(y)$ so that all output trajectories satisfy an LTL formula. The formula is built from a finite number of atomic propositions, which label areas of interest in the environment such as rooms or obstacles.

Problem 1: Given an LTL temporal logic formula ϕ and a robot with its dynamics defined in (2), construct an output feedback control $u(y)$ so that all output trajectories satisfy the formula ϕ.

Remark 1: Due to physical constrains, we assume that the robot's state (position and velocity) is restricted in a polytope $\mathcal{P}$.

Example 1: Consider a polygonal environment in Fig. 1. It is expected that a robot goes to room R1, R2, R3, R4, and R1 in order, and then remains in R1 thereafter. Meanwhile, it is desired that it shall avoid obstacles O1, ..., and O6. The specification can be interpreted as the following LTL formula

$$\Box\mathrm{E} \wedge \neg\mathrm{O} \wedge \diamond(\mathrm{R1} \wedge \diamond(\mathrm{R2} \wedge \diamond(\mathrm{R3} \wedge \diamond(\mathrm{R4} \wedge \diamond\Box\mathrm{R1})))),$$

where $\mathrm{O} = \mathrm{O1} \vee \mathrm{O2} \vee \cdots \vee \mathrm{O6}$.

III. Motion Planning

A. Local Control Strategies

In this subsection we consider two local control problems and their solutions, which will be used for constructing a finite transition system and verifying LTL specifications.

More specifically, we study the following two subproblems. Consider affine system (2) with its state restricted in

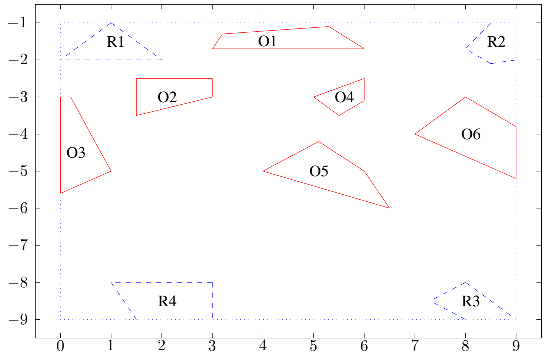

Fig. 1. An environment with obstacles.

an n-dimensional polytope $\mathcal{P}$ and its output restricted in an m-dimensional simplex $\mathcal{S}$.

Problem 2: Find, if possible, an output feedback control $u = Fy + g$, where $F \in \mathbb{R}^{s\times m}$ and $g \in \mathbb{R}^s$, such that all the output trajectories $y(t) \in Y(t)$ starting from $\mathcal{S}$ exit $\mathcal{S}$ in finite time through a specified facet (denoted by $\mathcal{F}_1$).

Another problem is to find output feedback such that the output trajectories remain in $\mathcal{S}$.

Problem 3: Find, if possible, an output feedback control $u = Fy + g$ such that if $y_0 \in \mathcal{S}$, then all $y(t, y_0) \in Y(t, y_0)$ are in $\mathcal{S}$ for $t \in [0, \infty)$.

The results presented in this subsection extend those of [14] and [20] with some modifications. The synthesis of local controllers solving problem 2 and 3 requires some preliminary results.

Lemma 1: [20] Consider two sets of points $\{v_1, \ldots, v_{m+1}\}$ in $\mathbb{R}^m$ and $\{u_1, \ldots, u_{m+1}\}$ in $\mathbb{R}^s$. Suppose that $v_1, \ldots, v_{m+1}$ are affinely independent. Then there exist a unique matrix $F \in \mathbb{R}^{s\times m}$ and a unique vector $g \in \mathbb{R}^s$ such that for each v_i, we have $u_i = Fv_i + g$.

The matrix F and g can be calculated from the following equation

$$[u_1 \ldots u_{n+1}] = [F\ g]\begin{bmatrix} v_1 & v_2 & \ldots & v_{n+1} \\ 1 & 1 & \ldots & 1 \end{bmatrix}.$$

Define a set-valued map $G : \mathbb{R}^m \to \mathbb{R}^m$ with

$$G(y) = \{C\dot{x} | Cx + d = y \text{ and } x \in \mathcal{P}\}.$$

The set $G(y)$ denotes all possible directions at y in the output space. In what follows, for any set $A \subset \mathbb{R}^m$, we use $G(A)$ to denote the set $\{C\dot{x} | Cx + d \in A \text{ and } x \in \mathcal{P}\}$,

Lemma 2: Consider affine system (3) and a compact set $\mathcal{S}$ in $\mathbb{R}^m$. If there is a vector $\xi \in \mathbb{R}^m$ such that $\xi^T h < 0$ for all $h \in G(\mathcal{S})$, then all output trajectories starting from $\mathcal{S}$ leave $\mathcal{S}$ in finite time.

Proof: Suppose by contradiction that there is an output trajectory $y(t, y_0)$ remaining in $\mathcal{S}$. Since $\xi^T h < 0$ for all $h \in G(\mathcal{S})$ and $G(\mathcal{S})$ is compact, it follows that there is an $\epsilon > 0$ such that $\xi^T h < -\epsilon$ for all $h \in G(\mathcal{S})$. Hence, we have $\xi^T \dot{y}(t, y_0) < -\epsilon$ for all t and therefore,the output trajectory $y(t, y_0)$ eventually leaves $\mathcal{S}$, a contradiction. □

A facet $\mathcal{F}_j$ is *restricted* or *blocked* if $n_j^T h \leq 0$ for all $h \in G(\mathcal{F}_j)$. The lemma below states a condition that ensures no output trajectory leaves $\mathcal{S}$ from a restricted facet.

Lemma 3: Let $\mathcal{F}_j$ be a facet of $\mathcal{S}$ with the unit normal vector n_j pointing out of $\mathcal{S}$. For affine system (3), if $n_j^T h \leq 0$ for all $h \in G(\mathcal{F}_j)$, then no output trajectory leaves $\mathcal{S}$ from $\mathcal{F}_j$.

The proof can be obtained similarly as the one in [20].

Next, we prove that the condition in Lemma 3 can be verified by only checking at the vertices of the facet. A few notations are introduced firstly.

For any point $y \in \mathcal{S}$, let

$$R(y) = \{x \in \mathcal{P} | Cx + d = y\}.$$

Notice that $R(y)$ is an $(n - m)$-dimensional polytope and is the set of states whose corresponding output is y.

For a facet $\mathcal{F}_j$ and a vertex $v_i \in \text{vert}(\mathcal{F}_j)$, we define

$$\beta_{ij}^- = \min_{x\in R(v_i)} n_j^T CAx, \ \beta_{ij}^+ = \max_{x\in R(v_i)} n_j^T CAx.$$

Lemma 4: For affine system (3), a facet $\mathcal{F}_j$ is restricted if and only if

$$\beta_{ij}^+ + n_j^T Ca \leq 0, \forall v_i \in \text{vert}(\mathcal{F}_j).$$

Proof: ($\Leftarrow$) Denote the vertices of F_j as $v_1, \ldots, v_k$. For any $y \in F_j$, it can be written as a convex combination of $v_1, \ldots, v_k$, i.e., $y = \alpha_1 v_1 + \cdots + \alpha_k v_k$ for $\alpha_1, \ldots, \alpha_k$ satisfying $\alpha_1 + \cdots + \alpha_k = 1$ and $0 \leq \alpha_i \leq 1$. For any $x \in R(y)$, where $y \in F_j$, there are points $x_1 \in R(v_1), \ldots, x_k \in R(v_k)$ such that x is the same convex combination of $x_1, \ldots, x_k$, i.e., $x = \alpha_1 x_1 + \cdots + \alpha_k x_k$. For $h \in G(y)$, there exists an $x \in R(y)$ satisfying $h = C\dot{x} = C(Ax + a)$. So $n_j^T h = n_j^T [CA(\alpha_1 x_1 + \cdots + \alpha_k x_k) + Ca]$ = $\alpha_1 n_j^T C(Ax_1 + a) + \cdots + \alpha_k n_j^T C(Ax_k + a)$. Every term, $n_j^T C(Ax_i + a)$, $i = 1, \ldots, k$, is less than or equals to zero. Thus, $n_j^T h \leq 0$. So the facet F_j is restricted by the definition.

($\Rightarrow$) Suppose that there is a vertex v_i such that $\beta_{ij}^+ + n_j^T Ca > 0$. It follows that $n_j^T CAx + n_j^T Ca > 0$ for some $x \in R(v_i)$, which means $n_j^T h > 0$ for $h = C\dot{x}$, a contradiction. □

Now, we are ready to present solutions to Problem 2 and Problem 3.

Theorem 1: Problem 2 is solvable if there are vectors $u_1, \ldots, u_{m+1}$ in $\mathbb{R}^s$ such that

$$\beta_{i1}^- + n_1^T (Ca + CBu_i) > 0, \forall v_i \in \text{vert}(S), \quad (4)$$

and for $j = 2, \ldots, m + 1$,

$$\beta_{ij}^+ + n_j^T (Ca + CBu_i) \leq 0, \forall v_i \in \text{vert}(\mathcal{F}_j). \quad (5)$$

Proof: By Lemma 1 a matrix F and a vector g can be constructed uniquely when $u_1, \ldots, u_m$ are solved from (4) and (5). Taking the F and g as output feedback, we have that the closed loop system is still affine. Consider the closed loop system. For $h \in G(v_i)$, $x \in R(v_i)$, we have

$$\begin{aligned} n_1^T h &= n_1^T (CAx + Ca + CBu_i) \\ &> \beta_{i1}^- + n_1^T (Ca + CBu_i) > 0. \end{aligned}$$

So by convex argument and Lemma 2, it follows that all output trajectories will leave the polytope. Furthermore, by Lemma 4, condition (5) ensures that all other facets are restricted. Combining these two facts, all output trajectories will leave $\mathcal{S}$ only via $\mathcal{F}_1$, and the conclusion follows. □

Remark 2: Notice that the solvability of linear inequality problem can be determined by the famous Farkas' Lemma.

Similarly, we have the following sufficient condition for the solvability of Problem 3.

Theorem 2: Problem 3 is solvable if there are vectors $u_1, \ldots, u_{m+1}$ in $\mathbb{R}^s$ such that for $j = 1, \ldots, m+1$,

$$\beta_{ij}^{+} + n_j^T(Ca + CBu_i) \leq 0, \, \forall v_i \in \operatorname{vert}(\mathcal{F}_j). \tag{6}$$

Proof: The conclusion follows since every facet is restricted by (6) and Lemma 4. □

Remark 3: Although, we consider only problems in simplices, there is no difficulty in extending the results to polytopes. In other words, in Problem 2 and 3, $\mathcal{S}$ can be a polytope instead of a simplex.

B. Determining Feasible Paths

We now describe our approach for motion planning. There are three main steps.

Step 1: Construct a transition system.

Consider an arbitrary triangulation $\{\mathcal{S}_1, \ldots, \mathcal{S}_N\}$ for E. We then construct an associated transition system $T = (Q, Q_0, \rightarrow, \Pi, \models)$ as follows:
(1) $Q = \{\mathcal{S}_1, \ldots, \mathcal{S}_N\}$;
(2) $Q_0 = \{\mathcal{S}_i\}$ if $y_0 \in \mathcal{S}_i$;
(3) For two simplices $\mathcal{S}_i$, $\mathcal{S}_j$ with a common facet $\mathcal{F}$, if Problem 2 is solvable for $\mathcal{S}_i$ and $\mathcal{F}$, then let $(\mathcal{S}_i, \mathcal{S}_j) \in \rightarrow$. On the other hand, if Problem 3 is solvable for simplex $\mathcal{S}_i$, then $(\mathcal{S}_i, \mathcal{S}_i) \in \rightarrow$.
(4) $\Pi = \{\pi_0, \pi_1, \ldots, \pi_M\}$, where each proposition π_i, $i = 1, \ldots, M$, denotes a region in E of interest and π_0 denotes the obstacles.
(5) $\mathcal{S}_i \models \pi_j \in \Pi$ if $\mathcal{S}_i \in [\pi_j]$.

Step 2: Determine feasible paths.

For the transition system T we just obtained, determine feasible paths that satisfy the given LTL formula ϕ. This is a well studied problem in model checking and there are many tools [21]. We proceed along the following route:
(1) Construct an automaton (also known as Buchi automaton) for the LTL formula ϕ and denote it by A_ϕ. The automaton has the property that it encodes precisely the paths which satisfy the LTL formula ϕ.
(2) Combine the automaton A_ϕ and the transition system T. The combination operation results in a transition system $A_\phi \times T$ whose paths are paths for both of the automaton A_ϕ and the transition system T.
(3) Find a path starting from a state derived from q_0 in the combined transition system $A_\phi \times T$. Such a path (if exits) can be interpreted as a path in T starting at q_0, which satisfies ϕ.

Step 3: Design output feedback control laws.

Let r be a path of T that satisfies ϕ. Then, for any two successive states $\mathcal{S}_i, \mathcal{S}_j \in r$, design local control $u_{ij} = F_{ij}y + g_{ij}$ according to Theorem 1. For a infinitely repeated state $\mathcal{S}_i \in r$, design local control $u_{ii} = F_{ii}y + g_{ii}$ according to Theorem 2.

In this way, we can obtain a piecewise affine output feedback control law for Problem 1.

Remark 4: It is worth pointing out that our approach for motion planning (Problem 1) is conservative due to the following reasons. First, Theorem 1 and 2 give only sufficient conditions for local reachability and set invariance problem. So it may fail to find a transition relation between two adjacent simplices. Second, the algorithm may fail for an unappropriate triangulation.

IV. Simulation

In this section, we present a simulation to illustrate our results. Consider the problem stated in Example 1. We assume that the dynamics of the robot is given by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix},$$

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix},$$

where the output (y_1, y_2) is the position in the plane, and (x_3, x_4) are its velocity. The robot is constrained in the region $\{0 \leq x_1 \leq 9, \, -9 \leq x_2 \leq -1\}$. Its velocity lies in $\{-30 \leq x_3 \leq 30, \, -30 \leq x_3 \leq 30\}$ due to physical constraints, Thus, the polytope describing the state constraints is given by $\mathcal{P} = \{0 \leq x_1 \leq 9, \, -9 \leq x_2 \leq -1, \, -30 \leq x_3 \leq 30, \, -30 \leq x_3 \leq 30, \, -30 \leq x_4 \leq 30\}$.

Propositions π_1, π_2, π_3, and π_4 are used to denote room R1, R2, R3, and R4, respectively. Proposition π_0 denotes the obstacles O=O1$\cup \cdots \cup$O6 . The LTL specification is then given as

$$\neg\pi_0 \wedge \diamond(\pi_1 \wedge \diamond(\pi_2 \wedge \diamond(\pi_3 \wedge \diamond(\pi_4 \wedge \diamond\Box\pi_1)))).$$

Applying our approach to the specific problem, we obtain a piecewise output feedback control such that resulting paths satisfy the LTL specification we give. Take R for an example, an output feedback control making output trajectories leave R1 enter S1 is $u = F_1 y + g_1$, where

$$F_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \text{ and } g_1 = \begin{bmatrix} 49.4546 \\ -109.4546 \end{bmatrix}.$$

An output feedback making R1 positive invariant is $u = F_2 y + g_2$, where

$$F_2 = \begin{bmatrix} -108.9449 & 63.2723 \\ 18.9449 & -153.2723 \end{bmatrix}$$

and

$$g_2 = \begin{bmatrix} 216.5447 \\ -276.5447 \end{bmatrix}.$$

A simulated trajectory starting at

$$[1.0000, \, -1.6667, \, 0.7094, \, 0.7547]^T \quad \text{(in R1)}$$

is shown in Fig. 2.

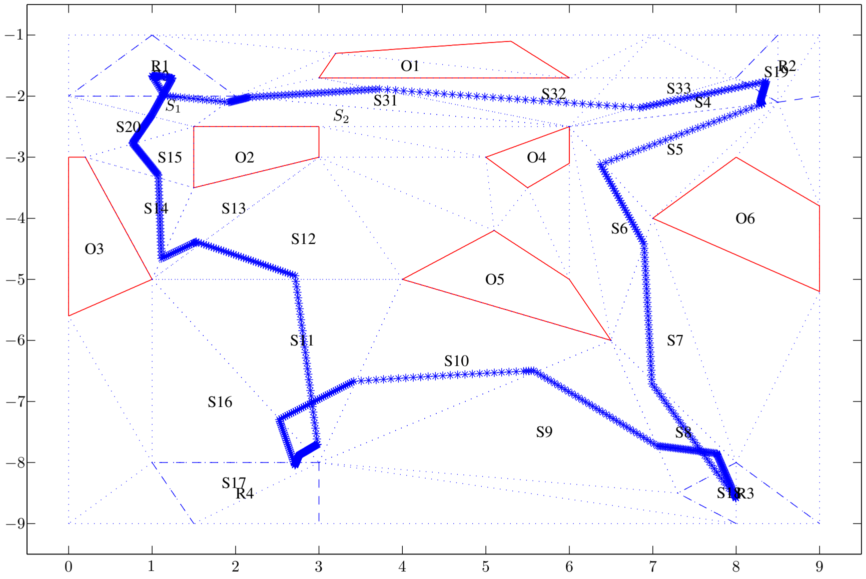

Fig. 2. The output trajectory of the robot.

V. Conclusion

In this paper, we present an algorithmic approach to a LTL motion planning problem. Our technique is based on two main parts, namely, a local output feedback strategy and a supervisory control for LTL specifications. The supervisory control is performed in a transition system, done by model checking. The local control is based on the reachability and invariance problem on simplices. The resulting control law is piecewise affine output feedback and valid on the corresponding simplices. There are several possible extensions for this work. First, one could use hybridization method to approximate a nonlinear system into a piecewise affine system with disturbance. Second, we will investigate the necessary and sufficient condition for local control in both simplices and polytopes.

References

[1] C. Belta, V. Isler, and G. J. Pappas, "Discrete abstractions for robot motion planning and control in polygonal environments," *IEEE Transaction on Robotics*, vol. 21, no. 5, pp. 864–874, 2005.

[2] J. C. Latombe, *Robot Motion Planning*. Norwell, MA: Kluwer, 1991.

[3] D. E. Koditschek, "The control of natural motion in mechanical systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 119, no. 4, pp. 548–551, 1991.

[4] G. A. Lafferriere and H. Sussmann, *Nonholonomic Motion Planning*. Norwell, MA: Kluwer, 1993, ch. A differential geometric approach to motion planning, pp. 235–270.

[5] A. Astolfi, "Discontinuous control of nonholonomic systems," *IEEE Transaction on Automatic Control*, vol. 27, no. 1, pp. 37–45, 1996.

[6] S. Lavalle, *Planning Algorithms*. Cambridge University Press, 2006.

[7] A. E. Emerson, *Handbook of Theoretical Computer Science*. Elsevier Science Publisher, 1990, vol. B, ch. Temporal and modal logic, pp. 997–1072.

[8] M. Antoniotti and B. Mishra, "Discrete event models + temproal logic = supervisory controller: Automatic synthesis of locomotion controllers," in *Proceedings of IEEE international Conference on Robotics and Automation*, Nagoya, Aichi, Japan, 1995.

[9] G. E. Fainekos, H. Kress-Gazit, and G. J. Pappas, "Temporal logic motion planning for mobile robots," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, Barcelona, Spain, 2005, pp. 2032–2037.

[10] C. Belta and L. C. G. J. M. Habets, "Constructing decidable hybrid systems with velocity bounds," in *Proceedings of 43rd IEEE Conference on Decision and Control*, Atlantis, Bahamas, 2004, pp. 467–472.

[11] G. Fainekos, A. Girard, H. Kress-Gazit, and G. J. Pappas, "Temporal logic motion planning for dynamic robots," *Automatica*, vol. 45, no. 2, pp. 343–352, 2009.

[12] G. E. Fainekos, A. Girard, H. Kress-Gazit, and G. J. Pappas, "Temporal logic motion planning for dynamic mobile robots," Department of CIS, University of Pennsylvania, Tech. Rep., 2007.

[13] G. E. Fainekos, S. G. Loizou, and G. J. Pappas, "Translating temporal logic to controller specifications," in *Proceedings of 45th IEEE Conference on Decision and Control*, San Diego, USA, 2006, pp. 13–15.

[14] L. C. G. H. M. Habets and J. H. van Schuppen, "A control problem for affine dynamical systems on a full-dimensional polytope," *Automatica*, vol. 40, pp. 21–35, 2004.

[15] E. D. Sontag, "Interconnected automata and linear systems: a theoretical framework in discrete-time," in *Hybrid Systems III: Verification and Control*. New Brunswick, USA: Springer, 1996, pp. 436–448.

[16] ——, "From linear to nonlinear: some complexity comparisons," in *Proceedings of IEEE Conference on Decision and Control*, New Orleans, USA, 1995, pp. 2916–2920.

[17] ——, "Nonlinear regulation: the piecewise linear approach," *IEEE Transaction on Automatic Control*, vol. 26, no. 2, pp. 346–358, 1981.

[18] A. Girard and S. Martin, "Motion planning for nonlinear systems using hybridizations and robust controllers on simplices," in *Proceedings of the 47th IEEE Conference on Decision and Control*, Cancun, Mexico, 2008.

[19] E. Asarin, T. Dang, and A. Girard, "Hybridization methods for the analysis of non-linear systems," *Acta Informatica*, vol. 43, no. 7, pp. 451–476, 2007.

[20] B. Roszak, "Necessary and sufficient conditions for reachability on a simplex," Master's thesis, University of Toronto, Toronto, Canada, 2005.

[21] M. Huth and M. Ryan, *Logic in Computer Science: Modelling and Reasoning about Systems*, 2nd ed. Cambridge University Press, 2004.