

Controller synthesis against omega-regular specifications: A funnel-based control approach

Pushpak Jagtap¹  | Dimos V. Dimarogonas²

¹Robert Bosch Center for Cyber Physical Systems, Indian Institute of Science, Bengaluru, India

²Division of Decision and Control Systems, KTH Royal Institute of technology, Stockholm, Sweden

Correspondence

Pushpak Jagtap, Robert Bosch Center for Cyber-Physical Systems, Indian Institute of Science, Bengaluru, India.

Email: pushpak@iisc.ac.in

Funding information

Knut and Alice Wallenberg Foundation; Swedish Foundation for Strategic Research (SSF); Swedish Research Council (VR); ERC CoG LEAFHOUND project; SERB Start-up Research Grant; ARTPARK; Siemens; Nokia

Abstract

The paper focuses on the problem of formal synthesis of controllers for control-affine nonlinear systems against complex properties. Our goal is to design a closed-form control policy that guarantees the satisfaction of complex properties that are expressed using (ω)-regular languages and equivalently recognized by nondeterministic Büchi automata (NBA). We propose leveraging a funnel-based control approach to provide a closed-form solution to the problem. Our approach decomposes the specification represented by NBA into a sequence of reachability problems, which we solve using a funnel-based control approach. Controllers associated with each reachability problem are then combined to design a hybrid control policy enforcing the desired (ω)-regular property. We demonstrate the effectiveness of the proposed results on room temperature control and mobile robot motion control case studies.

KEYWORDS

formal controller synthesis, funnel-based control, omega-regular specifications

1 | INTRODUCTION

The formal synthesis of controllers for dynamical systems against complex logic specifications has gained remarkable attention in recent years. These specifications are usually expressed using temporal logic formulae¹ or (in)finite strings over automata.² In the literature, abstraction-based approaches³ are popular for solving such synthesis problems. However, since the abstraction-based approaches usually require discretization of the state and input sets of physical systems, the synthesis problem becomes very intractable for control systems with higher state-space dimensions. To address this scalability issue, state-space discretization-free abstractions^{4,5} and compositional abstraction-based techniques^{6,7} have been proposed under suitable assumptions on the system dynamics (e.g., Lipschitz continuity or incremental input-to-state stability).

The discretization-free approaches based on control barrier functions (CBF) have shown a potential to solve the formal synthesis problem for various classes of complex logic specifications. The results in References 8–10 search for parametric CBFs to synthesize controllers to enforce specifications given by deterministic forms of Automata. However, there are several limitations over the applicability of these results; for example, (i) the techniques rely on a parametric form of CBF (such as polynomial) that will solve the problem, which is very difficult to determine for a given system and (ii) the techniques rely on a numerical search of the unknown parameters by using sum-of-squares or counter-example-guided inductive synthesis approaches which are computationally heavy and most of the time suffer from numerical errors. There are a few works that use another version of the CBF methodology for temporal-logic motion planning,^{11,12} and which rely on the online computation of the controller using optimization tools. Nevertheless, none of these approaches provide

closed-form controllers and rely on computational techniques that are very often numerically intractable and without feasibility guarantees.

To address these issues, this work proposes the use of a funnel-based control approach¹³ to provide a closed-form hybrid control policy that enforces complex specifications recognized by (ω)-regular languages² (or equivalently provided by nondeterministic Büchi automata [NBA]¹⁴). There have been few attempts to utilize funnel-based control approaches for enforcing a class of complex specifications. For example, the authors in References 15 and 16 provide results enforcing a fragment of signal temporal logic specifications, and the work in Reference 17 presents results enforcing a metric interval temporal logic for cooperative manipulation by discretizing state-space. However, to the best of our knowledge, this is the first paper to provide closed-form solutions enforcing specifications that are recognized by (ω)-regular languages. In particular, first, we provide funnel-based closed-form controllers that solve simple reachability problems for control-affine nonlinear systems. Then, we decompose given specifications as a sequence of reachability problems generated with the help of NBA recognizing the given (ω)-regular property. Finally, we combine the obtained controllers corresponding to each reachability task to design a hybrid control policy that enforces the given (ω)-regular property.

In the following, we summarize the key contributions of this work.

- Unlike the abstraction-based approaches,^{3,4} reachable set computations-based approaches,¹⁸ and the CBF-based discretization-free approaches,^{8,9} we provide a closed-form solution that does not require any numerical computations to design a controller enforcing complex logic specifications.
- The proposed result can handle a general class of complex specifications that are recognized by (ω)-regular languages (which is capable of capturing the full class of linear temporal logic [LTL] specifications).
- The results in References 8 and 9 work only with deterministic forms of automata and need to compute the complement of automata representing original specifications. In contrast, the proposed results work directly with NBA that recognizes ω -regular languages.

The remainder of this paper is structured as follows. In Section 2, we introduce system dynamics and the complex specifications given by the ω -regular property. Then, we formally define the problem considered in this paper. Section 3 provides a result on solving a reachability specification by utilizing a funnel-based control approach. In Section 4, we discuss the design of a closed-form hybrid control policy by utilizing the results obtained in Section 3 to enforce the ω -regular property. Section 6 demonstrates the effectiveness of the results on two case studies: (i) temperature control in a room and (ii) mobile robot motion control under complex tasks. Finally, Section 7 concludes the paper.

2 | PRELIMINARIES AND PROBLEM STATEMENT

2.1 | Notations

The symbols \mathbb{N} , \mathbb{N}_0 , \mathbb{Z} , \mathbb{R} , \mathbb{R}^+ , and \mathbb{R}_0^+ denote the set of natural, nonnegative integer, integer, real, positive real, and non-negative real numbers, respectively. We use $\mathbb{R}^{n \times m}$ to denote a vector space of real matrices with n rows and m columns. We use $\|\cdot\|$ to represent the Euclidean norm. For $a, b \in \mathbb{R}$ and $a < b$, we use (a, b) to represent open interval in \mathbb{R} . For $a, b \in \mathbb{N}$ and $a \leq b$, we use $[a; b]$ to denote close interval in \mathbb{N} . We use I_n and $0_{n \times m}$ to denote identity matrix in $\mathbb{R}^{n \times n}$ and zero matrix in $\mathbb{R}^{n \times m}$, respectively. A diagonal matrix in $\mathbb{R}^{n \times n}$ with diagonal entries d_1, \dots, d_n is denoted by $\text{diag}\{d_1, \dots, d_n\}$. Given a matrix $M \in \mathbb{R}^{n \times m}$, M^T represents transpose of matrix M . Given a matrix $P \in \mathbb{R}^{n \times n}$, $\text{Tr}(P)$ represents trace of matrix P . Given a set A , we use $|A|$ to represent the cardinality of the set A . $x_i, i \in [1; n]$ denotes i th element of vector $x \in \mathbb{R}^n$. Consider N sets $A_i, i \in \{1, \dots, N\}$, the Cartesian product of the sets is given by $A = \prod_{i \in \{1, \dots, N\}} A_i := \{(a_1, \dots, a_N) | a_i \in A_i, i \in \{1, \dots, N\}\}$. Consider a set $X_a \subset \mathbb{R}^n$, its projection on i th dimension, where $i \in [1; n]$, is given by an interval $[\underline{X}_{ai}, \overline{X}_{ai}] \subset \mathbb{R}$, where $\underline{X}_{ai} := \min\{x_i \in \mathbb{R} | [x_1, x_2, \dots, x_n] \in X_a\}$ and $\overline{X}_{ai} := \max\{x_i \in \mathbb{R} | [x_1, x_2, \dots, x_n] \in X_a\}$; and $\text{Int}(X_a)$ represents the interior of set X_a . We denote the empty set by \emptyset . Given a set S , the notation $|S|$ denotes the cardinality of S ; S^* and S^ω denote the set of all finite and infinite strings over S , respectively. Given sets U and $S \subset U$, the complement of S with respect to U is defined as $U \setminus S = \{x : x \in U, x \notin S\}$.

2.2 | System description

In this work, we consider control-affine systems S given by ordinary differential equations of the form

$$S : \dot{x} = f(x) + g(x)u, \quad (1)$$

where $x(t) = [x_1(t), x_2(t), \dots, x_n(t)]^T \in X \subseteq \mathbb{R}^n$ and $u(t) \in \mathbb{R}^m$ are state and input vectors, respectively. We assume that $g(x(t))g(x(t))^T$ is positive definite for all $x(t) \in X$. We use x_{x_0u} to denote a trajectory of S starting from initial state $x_0 \in X$ under a control signal u .

2.3 | Class of specifications

The main goal in this work is to synthesize controllers for systems (1), ensuring the satisfaction of ω -regular properties.² Such specifications can be expressed by ω -automata that can recognize infinite words, such as NBA,¹⁴ deterministic Rabin automata,¹⁹ deterministic Streett automata,²⁰ parity automata or Muller automata.²¹ While the above-mentioned automata have different acceptance conditions, they have the same expressive power and all of them recognize ω -regular languages. Due to the straightforward accepting condition and its well-established formalism, in this work, we use NBA to describe ω -regular properties, which are formally defined next.

Definition 1. A NBA is a tuple $\mathcal{A} = (Q, Q_0, \Sigma, \delta, F)$, where Q is a finite set of states, $Q_0 \subseteq Q$ is a set of initial states, Σ is a finite set of alphabets, $\delta : Q \times \Sigma \rightarrow 2^Q$ is a transition function, and $F \subseteq Q$ is a set of accepting states.

We use notation $q \xrightarrow{\sigma} q'$ to denote transition $(q, \sigma, q') \in \delta$ in an NBA \mathcal{A} . Consider an infinite state-run $q = (q_0, q_1, \dots) \in Q^\omega$, an infinite word (a.k.a. trace) $\sigma = (\sigma_0, \sigma_1, \dots) \in \Sigma^\omega$ such that $q_0 \in Q_0$, $q_i \xrightarrow{\sigma_i} q_{i+1}$ for all $i \in \mathbb{N}_0$, and let $\text{Inf}(q)$ be the set of states that occur infinitely many times in q . An infinite word $\sigma = (\sigma_0, \sigma_1, \dots) \in \Sigma^\omega$ is accepted by NBA \mathcal{A} if there exists an infinite state run q corresponding to σ such that $\text{Inf}(q) \cap F \neq \emptyset$. The set of words accepted by \mathcal{A} is called the accepting language of \mathcal{A} and is denoted by $\mathcal{L}(\mathcal{A})$.

We consider specifications expressed by accepting languages of NBA \mathcal{A} when input symbols are defined over a set of atomic propositions Π as the alphabet, that is, $\Sigma = 2^\Pi$. We should highlight that the full class of LTL properties can be represented using NBA and the conversion can be done by the existing tools such as SPOT²² and LTL2BA.²³

Remark 1. The approach proposed in this work can also be used for ω -regular languages represented using deterministic Streett automata and regular languages represented by (non)deterministic finite automata.²⁴ From a temporal logic perspective, the full class of LTL can be represented by deterministic Streett automata (one can use tool `ltl2dstar`²⁵ for conversions) and LTL over finite traces (LTL_f)²⁶ can be represented using deterministic finite automata (one can utilize toolbox `MONA`²⁷ for conversions).

2.4 | Satisfaction of specification by systems S

A given system S in (1) is connected to the specification given by the accepting language of an NBA \mathcal{A} defined over the set of atomic propositions Π , with the help of a labeling function $L : X \rightarrow \Pi$ as described in the next definition which is adapted from Reference 28, Def. 2.

Definition 2. For a system S in (1) and a labeling function $L : X \rightarrow \Pi$, an infinite sequence $\sigma(x_{x_0u}) = (\sigma_0, \sigma_1, \dots) \in \Pi^\omega$ is an infinite trace of the trajectory x_{x_0u} of S if there exists an associated timing sequence t_0, t_1, \dots such that $t_0 = 0$, $t_r \rightarrow \infty$ as $r \rightarrow \infty$, and for all $j \in \mathbb{N}$, $t_j \in \mathbb{R}_0^+$, and the following conditions hold

- $t_j < t_{j+1}$;
- $x_{x_0u}(t_j) \in L^{-1}(\sigma_j)$;
- If $\sigma_j \neq \sigma_{j+1}$, then for some $t'_j \in [t_j, t_{j+1}]$, $x_{x_0u}(t) \in L^{-1}(\sigma_j)$ for all $t \in (t_j, t'_j)$; $x_{x_0u}(t) \in L^{-1}(\sigma_{j+1})$ for all $t \in (t'_j, t_{j+1})$; and either $x_{x_0u}(t'_j) \in L^{-1}(\sigma_j)$ or $x_{x_0u}(t'_j) \in L^{-1}(\sigma_{j+1})$.

Next, we define the satisfaction of specification given by the language of an NBA \mathcal{A} .

Definition 3. Consider a system S in (1), a specification given by the accepting language of an NBA \mathcal{A} , and $\sigma(x_{x_0u})$ (i.e., an infinite trace of trajectory x_{x_0u}) as in Definition 2. We say that the trajectory of S starting from initial state $x_0 \in X$ under input signal u satisfies specification given by \mathcal{A} , denoted by $\sigma(x_{x_0u}) \models \mathcal{A}$, if $\sigma(x_{x_0u}) \in \mathcal{L}(\mathcal{A})$.

2.5 | Problem definition

The main controller synthesis problem considered in this work is formally defined next.

Problem 1. Consider a system S in (1), a specification given by the accepting language of an NBA $\mathcal{A} = (Q, Q_0, \Pi, \delta, F)$ over a set of atomic propositions $\Pi = \{p_0, p_1, \dots, p_M\}$, and a labeling function $L : X \rightarrow \Pi$. We aim at computing a closed-form hybrid controller u such that $\sigma(x_{x_0u}) \models \mathcal{A}$ for all $x_0 \in L^{-1}(p_i)$ and some $i \in [1; M]$ (i.e., we will have different controllers based on different starting regions associated with each initial atomic propositions).

To solve the aforementioned problem, we first utilize a funnel-based control approach which is discussed in the next section to solve reachability specification.

3 | REACHABILITY USING FUNNEL-BASED CONTROL

In this section, we propose the use of a funnel-based control approach¹³ to solve the reachability problem, which will later serve as a key element to solve Problem 1. Consider a funnel representing time-varying bounds for the trajectory $x_i, i \in [1; n]$ given as follows

$$-c_i \rho_i(t) < x_i(t) - \eta_i < d_i \rho_i(t), \quad (2)$$

for all $t \in \mathbb{R}_0^+$, where $\rho_i : \mathbb{R}_0^+ \rightarrow \mathbb{R}^+$, $i \in [1; n]$ are positive, smooth, and strictly decreasing funnel functions, $c_i, d_i \in \mathbb{R}_0^+$ and $\eta_i \in \mathbb{R}$ are some constants. In this work, we consider the following form of funnel function

$$\rho_i(t) = \rho_{i0} e^{-\epsilon_i t} + \rho_{i\infty}, \quad (3)$$

where $\rho_{i0}, \rho_{i\infty}, \epsilon_i \in \mathbb{R}^+$ are positive constants and $\rho_{i\infty} = \lim_{t \rightarrow \infty} \rho_i(t)$. Now, by normalizing $x_i(t) - \eta_i$ with respect to the performance function $\rho_i(t)$, we define the modulating error as $\hat{x}_i(t) := \frac{x_i(t) - \eta_i}{\rho_i(t)}$ and the corresponding performance region $\hat{D}_i := \{\hat{x}_i | \hat{x}_i \in (-c_i, d_i)\}$. Then, the modulated error is transformed through a transformation function $T_i : \hat{D}_i \rightarrow \mathbb{R}$ such that $T_i(0) = 0$ and is chosen as

$$T_i(\hat{x}_i) = \ln\left(\frac{d_i(c_i + \hat{x}_i)}{c_i(d_i - \hat{x}_i)}\right). \quad (4)$$

The transformed error is then defined as $\xi_i(x_i(t), \rho_i(t)) := T_i(\hat{x}_i)$. It can be verified that if the transformed error is bounded, then the modulated error \hat{x}_i is constrained within the region \hat{D}_i . This also implies that $x_i(t) - \eta_i$ evolves within the bounds given in (2). Differentiating ξ_i with respect to time, we obtain transformed error dynamics for i th dimension as

$$\dot{\xi}_i = \phi_i(\hat{x}_i, t)[\dot{x}_i + \alpha_i(t)(x_i - \eta_i)], \quad (5)$$

where $\phi_i(\hat{x}_i, t) := \frac{1}{\rho_i(t)} \frac{\partial T_i(\hat{x}_i)}{\partial \hat{x}_i} > 0$ for all $\hat{x}_i \in (-c_i, d_i)$ and $\alpha_i(t) := -\frac{\dot{\rho}_i(t)}{\rho_i(t)} > 0$ for all $t \in \mathbb{R}_0^+$ are the normalized Jacobian of the transformation function T_i and the normalized derivative of the performance function ρ_i , respectively. Now, by stacking all the transformed error dynamics, one gets

$$\dot{\xi} = \Phi_t(\dot{x} + \alpha_t(x - \eta)), \quad (6)$$

where $\xi = [\xi_1, \dots, \xi_n]^T$, $\Phi_t = \text{diag}\{\phi_1(\hat{x}_1, t), \dots, \phi_n(\hat{x}_n, t)\}$, $\alpha_t = \text{diag}\{\alpha_1(t), \dots, \alpha_n(t)\}$, and $\eta = [\eta_1, \dots, \eta_n]$. In the next theorem, we provide the result for enforcing reachability specification by utilizing the funnel approach.

Theorem 1. Consider system S in (1), any arbitrary sets $X_a, X_b \subset X$, $\Xi_i := [\underline{X}_{ai}, \bar{X}_{ai}] \cap [\underline{X}_{bi}, \bar{X}_{bi}]$, $\underline{X}_i := \min\{\underline{X}_{ai}, \underline{X}_{bi}\}$, $\bar{X}_i := \max\{\bar{X}_{ai}, \bar{X}_{bi}\}$, an arbitrarily chosen state $\eta = [\eta_1, \eta_2, \dots, \eta_n]^T \in \text{Int}(X_b)$ satisfying

$$\eta_i \in \begin{cases} \Xi_i & \text{if } \Xi_i \neq \emptyset \\ [\underline{X}_{bi}, \bar{X}_{bi}] & \text{if } \Xi_i = \emptyset, \end{cases}$$

$i \in [1; n]$, and funnel function (2) with $\epsilon_i \in \mathbb{R}^+$,

$$\rho_{i0} = \begin{cases} \max\{|\eta_i - \underline{X}_{ai}|, |\eta_i - \bar{X}_{ai}|\} & \text{if } \Xi_i \neq \emptyset \\ \max\{|\eta_i - \underline{X}_i|, |\eta_i - \bar{X}_i|\} & \text{if } \Xi_i = \emptyset, \end{cases}$$

constants c_i, d_i as follows:

$$\begin{aligned} c_i &= \frac{|\eta_i - \underline{X}_{ai}|}{\rho_{i0}}, d_i = 1, & \text{if } \Xi_i \neq \emptyset \text{ and } |\eta_i - \underline{X}_{ai}| \leq |\eta_i - \bar{X}_{ai}| \\ c_i &= 1, d_i = \frac{|\eta_i - \bar{X}_{ai}|}{\rho_{i0}}, & \text{if } \Xi_i \neq \emptyset \text{ and } |\eta_i - \underline{X}_{ai}| > |\eta_i - \bar{X}_{ai}| \\ c_i &= \frac{|\eta_i - \underline{X}_i|}{\rho_{i0}}, d_i = 1, & \text{if } \Xi_i = \emptyset \text{ and } |\eta_i - \underline{X}_i| \leq |\eta_i - \bar{X}_i| \\ c_i &= 1, d_i = \frac{|\eta_i - \bar{X}_i|}{\rho_{i0}}, & \text{if } \Xi_i = \emptyset \text{ and } |\eta_i - \underline{X}_i| > |\eta_i - \bar{X}_i|, \end{aligned}$$

and $\rho_{i\infty}$ is such that $\prod_{i \in [1; n]} \eta_i + [-c_i \rho_{i\infty}, d_i \rho_{i\infty}] \subset X_b$.

Then, under time-varying control law:

$$u(x, \rho) = -g(x)^T (g(x)g(x)^T)^{-1} (f(x) + \xi(x, \rho) + \bar{e}(x - \eta)), \tag{7}$$

where

$$\xi(x, \rho) = [\xi_1(x_1, \rho_1), \dots, \xi_n(x_n, \rho_n)]^T := \left[\ln \left(\frac{d_1 \left(c_1 + \frac{x_1 - \eta_1}{\rho_1} \right)}{c_1 \left(d_1 - \frac{x_1 - \eta_1}{\rho_1} \right)} \right), \dots, \ln \left(\frac{d_n \left(c_n + \frac{x_n - \eta_n}{\rho_n} \right)}{c_n \left(d_n - \frac{x_n - \eta_n}{\rho_n} \right)} \right) \right]^T$$

is a transformation error in (4) as discussed in Section 3 and $\bar{e} := \max_{i \in [1; n]} \epsilon_i$, one can ensure that $\exists t \in \mathbb{R}_0^+$ such that $x_{x_0, u}(t) \cap X_b \neq \emptyset$ for all $x_0 \in X_a$. In other words, the trajectory starting for any initial point in X_a will reach X_b in a finite time under the control law (7).

Proof. Consider Lyapunov like function $V = \frac{1}{2} \xi^T \xi$ and

$$\begin{aligned} \dot{V} &= \xi^T \Phi_t (f(x) + g(x)u + \alpha_t(x - \eta)) \\ &= \xi^T \Phi_t (f(x) - g(x)g(x)^T (g(x)g(x)^T)^{-1} (f(x) + \xi + \bar{e}(x - \eta)) + \alpha_t(x - \eta)). \end{aligned}$$

By following the facts that Φ_t and α_t are positive definite matrices, $\alpha_t < \bar{e} := \max_{i \in [1; n]} \epsilon_i$ and $\xi^T(x - \eta) \geq 0$ (this is due to $\xi_i(\hat{x}_i)$ is strictly increasing and $\xi_i(0) = 0$), one obtains $\dot{V} \leq -\xi^T \Phi_t \xi$. This implies that $\xi(t)$ is bounded for all $t \in \mathbb{R}_0^+$ and hence we guarantee (2) that is $-c_i \rho_i(t) + \eta_i < x_i(t) < d_i \rho_i(t) + \eta_i$. From the choice of $\eta \in \text{Int}(X_b)$ and constants $\rho_{i0}, \rho_{i\infty}, c_i, d_i, \eta_i$ for all $i \in [1; n]$, one can readily ensure that $X_a \subseteq \prod_{i \in [1; n]} [-c_i \rho_i(0^+) + \eta_i, -d_i \rho_i(0) + \eta_i]$ and as $\lim_{t \rightarrow \infty} \prod_{i \in [1; n]} [-c_i \rho_i(t) + \eta_i, -d_i \rho_i(t) + \eta_i] = \prod_{i \in [1; n]} \eta_i + [-c_i \rho_{i\infty}, d_i \rho_{i\infty}] \subset X_b$. This implies that there exist $t \in \mathbb{R}_0^+$ such that $x_{x_0, u}(t) \cap X_b \neq \emptyset$ for all $x_0 \in X_a$. ■

Remark 2. It is worth emphasizing that the sets X_a and X_b considered in Theorem 1 are arbitrary sets. The projection intervals of those arbitrary sets on the i th dimensions are used to design the corresponding funnel parameters.

4 | EXTENSION TO COMPLEX SPECIFICATIONS

In this section, we first provide an approach to decompose specifications given by the language of an NBA into a sequence of reachability tasks. Then, by utilizing the result of Theorem 1, we provide a construction of a hybrid control policy solving Problem 1.

4.1 | Sequential reachability decomposition

Consider an NBA $\mathcal{A} = (Q, Q_0, \Pi, \delta, F)$ corresponding to an ω -regular language expressing the properties of interest for the system S .

For a given accepting state-run q of \mathcal{A} , we denote the corresponding infinite words by $\sigma(q) \subseteq \Pi^\omega$. We also use a similar notation to denote finite words corresponding to finite state runs (i.e., $\sigma(\bar{q}) \in \Pi^n$ for $\bar{q} \in Q^{n+1}$, $n \in \mathbb{N}$). It is known that there exists a word $\sigma \in \Pi^\omega$ accepted by \mathcal{A} if there exists a state run of \mathcal{A} of the form $q = (q_0^r, q_1^r, \dots, q_{m_r}^r, (q_0^s, q_1^s, \dots, q_{m_s}^s)^\omega) \in Q^\omega$, where $m_r, m_s \in \mathbb{N}$, $q_0^r \in Q_0$ and $q_0^s \in F$.²⁴ Note that the considered structure of accepting run q captures a subset of all accepting runs. Let \bar{q} be a finite state run fragment of an accepting run q constructed by considering infinite sequence $(q_0^s, q_1^s, \dots, q_{m_s}^s)$ only once and is given by $\bar{q} = (q_0^r, q_1^r, \dots, q_{m_r}^r, q_0^s, q_1^s, \dots, q_{m_s}^s, q_0^s, q_1^s) \in Q^*$.

Let \mathcal{R} be the set of all such finite state-run fragments, excluding self-loops and repeated edges,

$$\mathcal{R} := \{\bar{q} = (q_0^r, q_1^r, \dots, q_{m_r}^r, q_0^s, q_1^s, \dots, q_{m_s}^s, q_0^s, q_1^s) \mid q_0^r \in Q_0, q_0^s \in F, q_i^r \neq q_{i+1}^r, \forall i < m_r, \text{ and } q_j^s \neq q_{j+1}^s, \forall j < m_s\}. \quad (8)$$

For each $p \in \Pi$, we define a set \mathcal{R}^p as

$$\mathcal{R}^p := \{\bar{q} = (q_0^r, q_1^r, \dots, q_{m_r}^r, q_0^s, q_1^s, \dots, q_{m_s}^s, q_0^s, q_1^s) \in \mathcal{R} \mid \sigma(q_0^r, q_1^r) = p\}. \quad (9)$$

The above notation with superscript p makes partitions of set \mathcal{R} according to the initial atomic proposition of finite state-run fragments in (8). In particular, it will help in defining hybrid controllers corresponding to each initial state set $L^{-1}(p_i)$, $i \in \{1, 2, \dots, M\}$. Decomposition into sequential reachability is performed as follows. For any $\bar{q} = (q_0, q_1, \dots, q_{m_r+m_s+3}) \in \mathcal{R}^p$, we define $\mathcal{P}^p(\bar{q})$ as a set of all state runs of length 3 (referred to as triplets),

$$\mathcal{P}^p(\bar{q}) := \{(q_i, q_{i+1}, q_{i+2},) \mid 0 \leq i \leq m_r + m_s + 1\}. \quad (10)$$

For a better understanding, the decomposition into sequential reachability is demonstrated below with the following example.

Example 1. Consider an NBA \mathcal{A} as shown in Figure 1, where $Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6\}$, $Q_0 = \{q_0\}$, $\Pi = \{p_0, p_1, p_2, p_3\}$, and $F = \{q_3\}$.

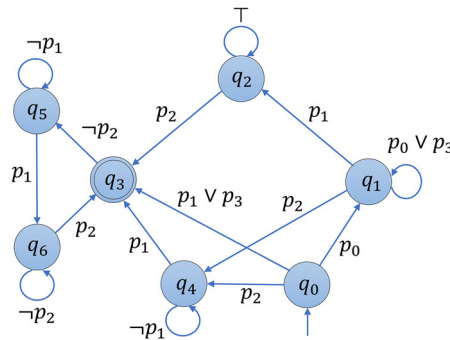


FIGURE 1 Nondeterministic Büchi automata \mathcal{A} used in Example 1.

The set of accepting state runs is $\{(q_0, q_1^*, q_2^*, (q_3, q_5^*, q_6^*)^\omega), (q_0, q_1^*, q_4^*, (q_3, q_5^*, q_6^*)^\omega), (q_0, q_1^*, q_4^*, (q_3, q_5^*, q_6^*)^\omega), (q_0, (q_3, q_5^*, q_6^*)^\omega)\}$. The set of finite state-run fragments \mathcal{R} in (8) is obtained as follow:

$$\mathcal{R} = \{(q_0, q_1, q_2, q_3, q_5, q_6, q_3, q_5), (q_0, q_1, q_4, q_3, q_5, q_6, q_3, q_5), (q_0, q_4, q_3, q_5, q_6, q_3, q_5), (q_0, q_3, q_5, q_6, q_3, q_5)\}.$$

The sets \mathcal{R}^p for $p \in \Pi$ are as follows:

$$\begin{aligned} \mathcal{R}^{p_0} &= \{(q_0, q_1, q_2, q_3, q_5, q_6, q_3, q_5), (q_0, q_1, q_4, q_3, q_5, q_6, q_3, q_5)\}, \mathcal{R}^{p_1} = \{(q_0, q_3, q_5, q_6, q_3, q_5)\}, \\ \mathcal{R}^{p_2} &= \{(q_0, q_4, q_3, q_5, q_6, q_3, q_5)\}, \mathcal{R}^{p_3} = \{(q_0, q_3, q_5, q_6, q_3, q_5)\}. \end{aligned}$$

The sets $\mathcal{P}^p(\bar{q})$ for $\bar{q} \in \mathcal{R}^p$ are as follows:

$$\begin{aligned} \mathcal{P}^{p_0}(q_0, q_1, q_2, q_3, q_5, q_6, q_3, q_5) &= \{(q_0, q_1, q_2), (q_1, q_2, q_3), (q_2, q_3, q_5), (q_3, q_5, q_6), (q_5, q_6, q_3), (q_6, q_3, q_5)\}, \\ \mathcal{P}^{p_0}(q_0, q_1, q_4, q_3, q_5, q_6, q_3, q_5) &= \{(q_0, q_1, q_4), (q_1, q_4, q_3), (q_4, q_3, q_5), (q_3, q_5, q_6), (q_5, q_6, q_3), (q_6, q_3, q_5)\}, \\ \mathcal{P}^{p_2}(q_0, q_4, q_3, q_5, q_6, q_3, q_5) &= \{(q_0, q_4, q_3), (q_4, q_3, q_5), (q_3, q_5, q_6), (q_5, q_6, q_3), (q_6, q_3, q_5)\}, \\ \mathcal{P}^{p_1}(q_0, q_3, q_5, q_6, q_3, q_5) &= \mathcal{P}^{p_3}(q_0, q_3, q_5, q_6, q_3, q_5) = \{(q_0, q_3, q_5), (q_3, q_5, q_6), (q_5, q_6, q_3), (q_6, q_3, q_5)\}. \end{aligned}$$

4.2 | Hybrid control policy

Each triplet $v = (q, q', q'') \in \mathcal{P}^p(\bar{q})$, $p \in \Pi$ can be viewed as reach while avoid specification. In particular, the system trajectory should reach region $L^{-1}(\sigma(q', q''))$ while avoiding region $X \setminus L^{-1}(\sigma(q', q''))$ starting from some initial region $L^{-1}(\sigma(q, q'))$. Next, we raise an Assumption 1 on the locations of these sets under which we can use the proposed approach to solve Problem 1.

Assumption 1. We assume that there exist an accepting state-run q such that for all $v = (q, q', q'') \in \mathcal{P}^p(\bar{q})$, $p \in \Pi$, either of the following hold:

- $X \setminus L^{-1}(\sigma(q', q'')) = L^{-1}(\sigma(q', q''))$ or
- $X \setminus L^{-1}(\sigma(q', q'')) \cap \prod_{i \in [1, n]} [-c_i \rho_i(0) + \eta_i, -d_i \rho_i(0) + \eta_i] = \emptyset$,

where $\rho_i(0)$, c_i , d_i , and η_i are as defined in Theorem 1 with $X_a = L^{-1}(\sigma(q, q'))$ and $X_b = L^{-1}(\sigma(q', q''))$.

Remark 3. Assumption 1 says that either the avoid region should be the complement of the reach region or the intersection of the avoid region with the region given by the funnel should be empty. In general, NBA have more than one accepting state-run, so the possibility of having an accepting state-run that does not satisfy Assumption 1 is much smaller. Even in the unlikely case that there is no accepting state-run satisfying the above assumption, one can introduce dummy atomic propositions that ensure that the assumption is satisfied with the newly constructed NBA without affecting the original specification. One can readily observe it for the two-dimensional case. However, providing a generalized and automated algorithm is challenging and will be considered as a future research direction. Another advantage of Assumption 1 is to filter out practically infeasible accepting state runs (e.g., those that do not have continuity). For better understanding, consider the temperature control case study (given in Section 6.1) where a triplet (q_1, q_3, q_1) in Figure 3 is associated with a reachability task defined as “reach $L^{-1}(p_2)$ from $L^{-1}(p_1)$ without entering any other region.” Then, one can readily see from Figure 4 that the task is not feasible.

Next, we provide a Lemma that correlates a particular segment in specification automata NBA \mathcal{A} (referred to as triplet) with the result of Theorem 1.

Lemma 1. For a triplet $v = (q, q', q'') \in \mathcal{P}^p(\bar{q})$, where $\bar{q} \in \mathcal{R}^p$ for some $p \in \Pi$ satisfying Assumption 1, if we use control policy $u_v(x, \rho_v)$ as in (7) in Theorem 1 with $X_a = L^{-1}(\sigma(q, q'))$ and $X_b = L^{-1}(\sigma(q', q''))$, then there exist $t \in \mathbb{R}_0^+$ such that the trajectory ξ_{x_0, u_v} of S starting from any initial state $x_0 \in X_a$ under policy u_v satisfies $x_{x_0, u_v} \cap X_b \neq \emptyset$.

Proof. The proof follows similar to that of Theorem 1. ■

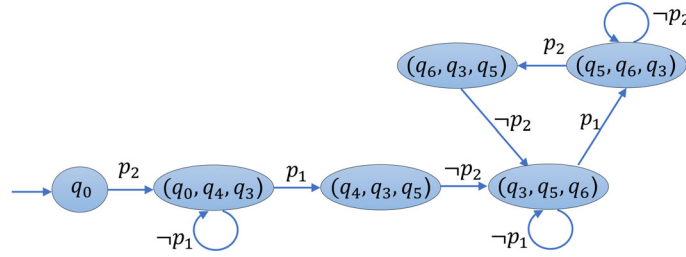


FIGURE 2 Switching mechanism given by state transition system.

Given an NBA $\mathcal{A} = (Q, Q_0, \Pi, \delta, F)$ expressing the properties of interest, an accepting run q , and corresponding finite state-run fragment \bar{q} satisfying Assumption 1, next we define a finite state transition system which provides a switching mechanism for a hybrid control policy. The switching mechanism is given by a finite state transition system $\mathfrak{S} = (Q_s, Q_{0s}, \Pi, \delta_s)$, where $Q_{0s} = Q_0$, $Q_s = Q_{0s} \cup \mathcal{P}^p(\bar{q})$, and transition relation $(q_s, \sigma, q'_s) \in \delta_s$ (also denoted by $q_s \xrightarrow{\sigma} q'_s$) is defined as:

- for all $q_s = q_0 \in Q_{0s}$,
 - $q_0 \xrightarrow{\sigma(q_0, q')} (q_0, q', q'')$, where $q_0 \xrightarrow{\sigma(q_0, q')} q'$;
- for all $q_s = (q, q', q'') \in Q_s \setminus Q_{0s}$,
 - $(q, q', q'') \xrightarrow{\sigma(q', q'')} (q', q'', q''')$, such that $q, q', q'', q''' \in Q$, $q' \xrightarrow{\sigma(q', q'')} q''$, $q'' \xrightarrow{\sigma(q'', q''')} q'''$.

Example 2. For a run $\bar{q} = (q_0, q_4, q_3, q_5, q_6, q_3, q_5) \in \mathcal{R}^{p_2}$ in an NBA \mathcal{A} as shown in Figure 1, the finite state transition system corresponding to the switching mechanism is illustrated in Figure 2.

The hybrid controller defined over the augmented state-space $X \times Q_s$ that is a candidate for solving Problem 1 is given by

$$\bar{\mathbf{u}}(x(t), q_s) := u_{q'_s}(x(t), \rho_{q'_s}(t - t_{q_s})), \quad \forall (q_s, L(x), q'_s) \in \delta_s, \quad \forall t \in \mathbb{R}_0^+, \quad (11)$$

where $t_{q_s} \in \mathbb{R}_0^+$ is the time instance defined as $t_{q_s} := \min\{t \in \mathbb{R}_0^+ \mid x_{x_0, \bar{\mathbf{u}}}(t) \in \sigma(q', q'') \text{ for } q_s = (q, q', q'')\}$.

The next theorem shows that under the proposed hybrid controller in (11), one can ensure the satisfaction of the specification given by the language of an NBA.

Theorem 2. Consider a system S (1) and an NBA \mathcal{A} representing a specification with a finite state-run fragment $\bar{q} \in \mathcal{R}^p$ for some $p \in \Pi$ corresponding to the accepting run q satisfying Assumption 1. Then the state trajectory $x_{x_0, \bar{\mathbf{u}}}$ of S starting from any initial state $x_0 \in L^{-1}(p)$ under the hybrid controller $\bar{\mathbf{u}}$ satisfies the language of NBA \mathcal{A} , that is, $\sigma(x_{x_0, \bar{\mathbf{u}}}) \models \mathcal{A}$.

Proof. Consider $p \in \Pi$ and an accepting state run $q = (q_0^r, q_1^r, \dots, q_m^r, (q_0^s, q_1^s, \dots, q_m^s)^\omega) \in Q^\omega$ in \mathcal{A} with $\sigma(q_0^r, q_1^r) = p$. Let the corresponding finite state-run be $\bar{q} \in \mathcal{R}^p$ as defined in Section 4.1 satisfying Assumption 1. If we utilize controller (11) that provides a switching strategy among the reachability control laws $u_\nu(x)$ among $\nu = (q, q', q'') \in \mathcal{P}^p(\bar{q})$ as given in Lemma 1, one can conclude $\sigma(q) \in \mathcal{L}(\mathcal{A})$. By utilizing the definition of labeling function L , this implies that the state run $x_{x_0, \bar{\mathbf{u}}}$ of S starting from any initial state $x_0 \in L^{-1}(p)$ under policy $\bar{\mathbf{u}}$ given in (11) satisfies $\sigma(x_{x_0, \bar{\mathbf{u}}}) \in \mathcal{L}(\mathcal{A})$. This concludes the proof. ■

5 | COMPUTATIONAL COMPLEXITY

Although the controller in (11) constitutes a closed-form solution for a given state-run $\bar{q} \in \mathcal{R}$ satisfying Assumption 1 there is some complexity that lies in establishing the set \mathcal{R} . For a given NBA $\mathcal{A} = (Q, Q_0, \Sigma, \delta, F)$ and a run of

the form $\bar{q} = (q_0^r, q_1^r, \dots, q_{m_r}^r, q_0^s, q_1^s, \dots, q_{m_s}^s, q_0^s, q_1^s) \in \mathcal{R}$ such that $q_i^r \in Q_0, q_0^s \in F, q_i^r \neq q_{i+1}^r, \forall i < m_r$, and $q_j^s \neq q_{j+1}^s, \forall j < m_s$, the worst-case complexity for the construction of set \mathcal{R} can be expressed as $|Q_0|(|Q| - 1)^{|\delta|-1}$ for constructing $q_0^r, q_1^r, \dots, q_{m_r}^r, q_0^s$ part of \bar{q} and $(|Q| - 1)^{|\delta|-1}$ for constructing $q_0^s, q_1^s, \dots, q_{m_s}^s, q_0^s, q_1^s$ part of \bar{q} , where $|Q|$ and $|\delta|$ are the cardinality of the states and transitions (excluding self-loops) in NBA, respectively.²⁸ However, in practice, the complexity is considerably reduced as generating a complete set \mathcal{R} is unnecessary. The search process can be terminated upon encountering a run that fulfils Assumption 1.

6 | CASE STUDIES

In this section, we consider two case studies to demonstrate the effectiveness of our results.

6.1 | Temperature control of a room

We consider the evolution of a room temperature given by

$$\dot{T} = \alpha_e(T_e - T) + \alpha_H(T_h - T)u, \quad (12)$$

where $T(t)$ denotes the temperature of the room, $u(t)$ represents control input, $T_h = 55^\circ\text{C}$ is the heater temperature, $T_e = 15^\circ\text{C}$ is the ambient temperature, and $\alpha_e = 8 \times 10^{-3}$ and $\alpha_H = 3.6 \times 10^{-3}$ are heat exchange coefficients. All the parameters are adopted from Reference 5.

The state set of the system is $X = [10, 30]$. We consider regions of interest $X_0 = [10, 15], X_1 = [18, 20], X_2 = [25, 30]$, and $X_3 = X \setminus (X_0 \cup X_1 \cup X_2)$. The set of atomic propositions is given by $\Pi = \{p_0, p_1, p_2, p_3\}$ with labeling function $L(x_i) = p_i$ for all $x_i \in X_i, i \in \{0, 1, 2, 3\}$. The objective is to compute a hybrid control policy ensuring the satisfaction of the specification given by LTL formula $\varphi = p_0 \wedge (\Box \Diamond p_1 \wedge \Box \Diamond p_2)$ or equivalently by the accepting language of the NBA \mathcal{A} in Figure 3. Note that \Box and \Diamond are temporal operators in LTL specification representing always and eventually operators, see Reference 24 for more details. From Figure 3, one can readily identify that there are four possible structures of accepting state runs: $(q_0, q_1^*, (q_3, q_1^*)^\omega)$, $(q_0, q_2^*, q_1^*(q_3, q_1^*)^\omega)$, $(q_0, q_1^*, (q_3, q_2^*, q_1^*)^\omega)$, and $(q_0, q_2^*, q_1^*(q_3, q_2^*, q_1^*)^\omega)$ with corresponding sets

$$\begin{aligned} \mathcal{R}^{p_0} &= \{(q_0, q_1, q_3, q_1, q_3, q_1), (q_0, q_2, q_1, q_3, q_1, q_3, q_1), (q_0, q_1, q_3, q_2, q_1, q_3, q_2), (q_0, q_2, q_1, q_3, q_2, q_1, q_3, q_2)\}, \\ \mathcal{R}^{p_1} = \mathcal{R}^{p_2} = \mathcal{R}^{p_3} &= \emptyset, \mathcal{P}^{p_0}(q_0, q_1, q_3, q_1, q_3, q_1) = \{(q_0, q_1, q_3), (q_1, q_3, q_1), (q_3, q_1, q_3), (q_1, q_3, q_1)\}, \\ \mathcal{P}^{p_0}(q_0, q_2, q_1, q_3, q_2, q_1, q_3, q_2) &= \{(q_0, q_2, q_1), (q_2, q_1, q_3), (q_1, q_3, q_2), (q_3, q_2, q_1), (q_2, q_1, q_3), (q_1, q_3, q_2)\}, \\ \mathcal{P}^{p_0}(q_0, q_2, q_1, q_3, q_1, q_3, q_1) &= \{(q_0, q_2, q_1), (q_2, q_1, q_3), (q_1, q_3, q_1), (q_3, q_1, q_3), (q_1, q_3, q_1)\}, \\ \text{and } \mathcal{P}^{p_0}(q_0, q_1, q_3, q_2, q_1, q_3, q_2) &= \{(q_0, q_1, q_3), (q_1, q_3, q_2), (q_3, q_2, q_1), (q_2, q_1, q_3), (q_1, q_3, q_2)\}. \end{aligned}$$

One can readily observe that (q_1, q_3, q_1) in $\mathcal{P}^{p_0}(q_0, q_2, q_1, q_3, q_1, q_3, q_1)$ and $\mathcal{P}^{p_0}(q_0, q_1, q_3, q_1, q_3, q_1)$ does not satisfy Assumption 1. Therefore, we choose $\mathcal{P}^{p_0}(q_0, q_1, q_3, q_2, q_1, q_3, q_2)$ to design a hybrid controller in (11). Figure 4 shows a

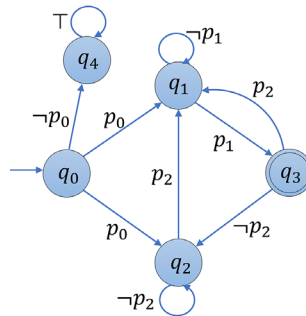


FIGURE 3 An nondeterministic Büchi automata representing specification for the room temperature control example.

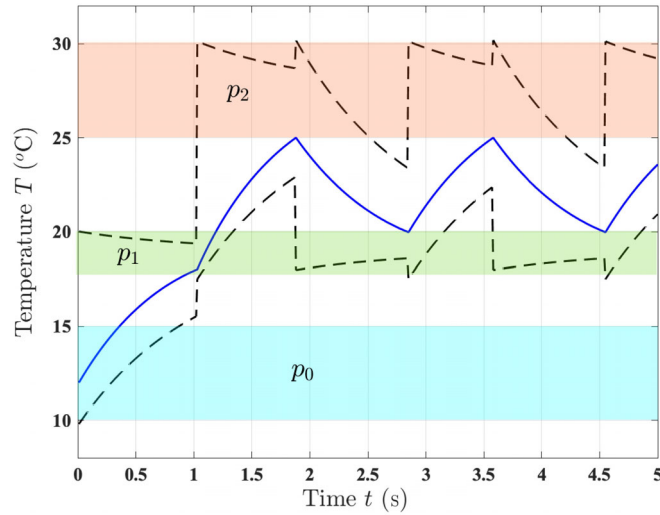


FIGURE 4 A trajectory of the system (12) under hybrid control policy (11). The black dotted lines show funnels in which the trajectory is evolving.

trajectory of the system (12) under a proposed hybrid control policy (11). The black dotted lines show the funnel constructed to design the controller as discussed in Theorem 1. One can readily see that the trajectory is evolving inside the bounds given by constructed funnel functions and, hence, satisfying the desired specification.

6.2 | Mobile robot motion control

For the second case study, we consider a three-wheeled omnidirectional mobile robot²⁹ given by differential equations as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{bmatrix} = \begin{bmatrix} \cos(x_3) & -\sin(x_3) & 0 \\ \sin(x_3) & \cos(x_3) & 0 \\ 0 & 0 & 1 \end{bmatrix} \underbrace{\begin{bmatrix} 0 & -1 & L \\ \cos \frac{\pi}{6} & \sin \frac{\pi}{6} & L \\ -\cos \frac{\pi}{6} & \sin \frac{\pi}{6} & L \end{bmatrix}^{-1}}_B Ru,$$

where two states x_1 and x_2 indicates robot's position, and x_3 indicates robot's orientation with respect to the x_1 -axis. $R = 0.02$ is the wheel radius, B describes geometric constraints with $L = 0.2$ (radius of the robot body), and $u(t) \in \mathbb{R}^3$ are control inputs.

The state set of the system is $X = [0,100] \times [0,100] \times [0, 2\pi]$. We consider regions of interest $X_0 = [40, 70] \times [40, 70] \times [0, 2\pi]$, $X_1 = [10, 35] \times [10, 35] \times [0, 2\pi]$, $X_2 = [10, 30] \times [70, 90] \times [0, 2\pi]$, $X_3 = [70, 90] \times [10, 30] \times [0, 2\pi]$, and $X_4 = X \setminus (X_0 \cup X_1 \cup X_2 \cup X_3)$ (see Figure 6). The set of atomic propositions is given by $\Pi = \{p_0, p_1, p_2, p_3, p_4\}$ with labeling function $L(x_i) = p_i$ for all $x_i \in X_i, i \in \{0, 1, 2, 3, 4\}$. The objective is to compute a hybrid control policy ensuring the satisfaction of the specification given by LTL formula $\varphi = \square \neg p_0 \wedge p_3 \wedge \square \diamond p_2 \wedge \diamond p_1$ or equivalently by the accepting language of the NBA \mathcal{A} in Figure 5. From Figure 5, one can readily identify that there are three possible structures of accepting state runs: (q_0, q_1^+, q_2^ω) , $(q_0, q_1^+(q_2^+, q_3^*)^\omega)$, and $(q_0, q_1^+, q_2^+, q_3^+, q_2^\omega)$ with corresponding sets $\mathcal{R}^{p_3} = \{(q_0, q_1, q_2, q_2), (q_0, q_1, q_2, q_3, q_2, q_3), (q_0, q_1, q_2, q_3, q_2, q_2)\}$, $\mathcal{R}^{p_0} = \mathcal{R}^{p_1} = \mathcal{R}^{p_2} = \mathcal{R}^{p_4} = \emptyset$; $\mathcal{P}^{p_3}(q_0, q_1, q_2, q_2) = \{(q_0, q_1, q_2), (q_1, q_2, q_2)\}$, $\mathcal{P}^{p_3}(q_0, q_1, q_2, q_3, q_2, q_3) = \{(q_0, q_1, q_2), (q_1, q_2, q_3), (q_2, q_3, q_2), (q_3, q_2, q_3)\}$, $\mathcal{P}^{p_3}(q_0, q_1, q_2, q_3, q_2, q_2) = \{(q_0, q_1, q_2), (q_1, q_2, q_3), (q_2, q_3, q_2), (q_3, q_2, q_2)\}$. One can readily observe that $(q_1, q_2, q_2) \in \mathcal{P}^{p_3}(q_0, q_1, q_2, q_2)$ do not satisfy the Assumption 1. Thus, we choose $\mathcal{P}^{p_3}(q_0, q_1, q_2, q_3, q_2, q_3)$ to design a hybrid controller in (11). Figure 6 shows a trajectory of the systems (12) under a proposed hybrid control policy (11). One can readily see that the trajectory is satisfying the specification given in Figure 5.

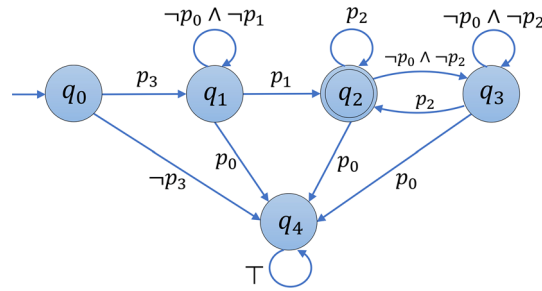


FIGURE 5 A nondeterministic Büchi automata representing specification for the mobile robot example.

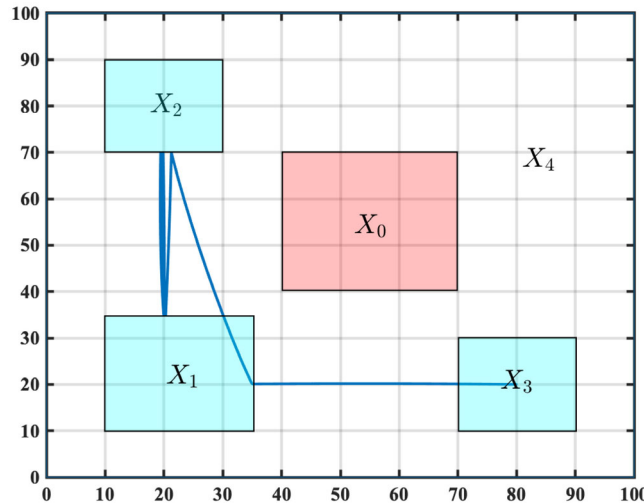


FIGURE 6 Simulation under hybrid control policy (11).

6.2.1 | Comparison with existing methodologies

In the realm of solving synthesis problems for specifications given by NBA, two prevalent methodologies in the literature are symbolic control techniques³ and CBF-based techniques.^{8,9} These approaches rely heavily on algorithmic and optimization-driven methods, which often result in substantial computational demands. In contrast, the proposed approach offers a closed-form state feedback solution, thus, eliminating the need for intensive computations. It is important to acknowledge, however, that the proposed approach is specifically applicable to a limited class of systems, as defined in (1), and assumes unbounded input. Furthermore, in terms of the nature of the solution, symbolic control approaches typically compute a maximal permissible controller, encompassing a set of all possible controllers that enforce a given specification. In contrast, the proposed approach provides a solution from an infinite set of possibilities that can satisfy the desired specifications.

7 | DISCUSSION AND CONCLUSION

In this paper, we proposed a discretization-free approach for the formal synthesis of controllers for a class of control-affine systems to enforce complex properties expressed by ω -regular languages (or by the language of an NBA). The approach utilizes a funnel-based control approach to provide a closed-form solution to solve the problem. However, there are notable limitations in our proposed approach that need further investigation: (i) The presented approach is not suitable for under-actuated systems and general nonaffine nonlinear control systems, (ii) While the approach is theoretically sound and can generate a closed-form hybrid control policy when there exists an accepting run satisfying Assumption 1, there is room to relax Assumption 1, and (iii) The current approach assumes unbounded control inputs, which may not be a good

setting for real-world scenarios. As part of our future research direction, we aim to address these challenges and extend the applicability of our results, ultimately making our approach more versatile and broadly useful.

ACKNOWLEDGMENTS

This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation, the Swedish Foundation for Strategic Research (SSF), the Swedish Research Council (VR), the ERC CoG LEAFHOUND project, the SERB Start-up Research Grant, ARTPARK, and the CSR Grants by Siemens and Nokia.

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

DATA AVAILABILITY STATEMENT

Data sharing is not applicable to this article as no new data were created or analyzed in this study.

ORCID

Pushpak Jagtap  <https://orcid.org/0000-0002-5452-8850>

REFERENCES

1. Pnueli A. The temporal logic of programs. Paper presented at: 18th Annual Symposium on Foundations of Computer Science (SFCS 1977), Providence, RI. 1977:46-57.
2. Thomas W. Automata on infinite objects. In: Van Leeuwen J, ed. *Formal Models and Semantics*. Elsevier; 1990:133-191.
3. Tabuada P. *Verification and Control of Hybrid Systems: A Symbolic Approach*. Springer Science & Business Media; 2009.
4. Girard A. Approximately bisimilar abstractions of incrementally stable finite or infinite dimensional systems. Paper presented at: 53rd IEEE Conference on Decision and Control, Los Angeles, CA. 2014:824-829.
5. Jagtap P, Zamani M. QUEST: a tool for state-space quantization-free synthesis of symbolic controllers. In: Bertrand N, Bortolussi L, eds. *International Conference on Quantitative Evaluation of Systems*. Springer; 2017:309-313.
6. Rungger M, Zamani M. Compositional construction of approximate abstractions. *Proceedings of the 18th International Conference on Hybrid Systems: Computation and Control*. ACM; 2015:68-77.
7. Saoud A, Jagtap P, Zamani M, Girard A. Compositional abstraction-based synthesis for interconnected systems: an approximate composition approach. *IEEE Trans Control Netw Syst*. 2021;8(2):702-712.
8. Jagtap P. *Formal Synthesis of Controllers for Complex Dynamical Systems: State-Space Discretization-Free Approaches*. PhD Thesis. Universitätsbibliothek der TU München. 2020.
9. Anand M, Lavaei A, Zamani M. Compositional synthesis of control barrier certificates for networks of stochastic systems against ω -regular specifications. *Nonlinear Anal Hybrid Syst*. 2024;51:101427.
10. Ramasubramanian B, Niu L, Clark A, Bushnell L, Poovendran R. Linear temporal logic satisfaction in adversarial environments using secure control barrier certificates. In: Alpcan T, Vorobeychik Y, Baras J. S, Dán G, eds. *International Conference on Decision and Game Theory for Security*. Springer; 2019:385-403.
11. He B, Lee J, Topcu U, Sentis L. BP-RRT: barrier pair synthesis for temporal logic motion planning. Paper presented at: 2020 59th IEEE Conference on Decision and Control (CDC), Jeju, South Korea. 2020:1404-1409.
12. Nilsson P, Ames AD. Barrier functions: bridging the gap between planning from specifications and safety-critical control. Paper presented at: 2018 IEEE Conference on Decision and Control (CDC). 2018:765-772.
13. Bechlioulis CP, Rovithakis GA. A low-complexity global approximation-free control scheme with prescribed performance for unknown pure feedback systems. *Automatica*. 2014;50(4):1217-1226.
14. Büchi JR. On a decision method in restricted second order arithmetic. In: Nagel E, Suppes P, Tarski A, eds. *The Collected Works of J. Richard Büchi*. Springer; 1966:425-435.
15. Lindemann L, Verginis CK, Dimarogonas DV. Prescribed performance control for signal temporal logic specifications. Paper presented at: 2017 IEEE 56th Annual Conference on Decision and Control (CDC), Melbourne, VIC, Australia. 2017:2997-3002.
16. Lindemann L, Dimarogonas DV. Feedback control strategies for multi-agent systems under a fragment of signal temporal logic tasks. *Automatica*. 2019;106:284-293.
17. Verginis CK, Dimarogonas DV. Timed abstractions for distributed cooperative manipulation. *Auton Robots*. 2018;42(4):781-799.
18. Schürmann B, Althoff M. Optimizing sets of solutions for controlling constrained nonlinear systems. *IEEE Trans Automat Contr*. 2020;66(3):981-994.
19. Rabin MO. Decidability of second-order theories and automata on infinite trees. *Trans Am Math Soc*. 1969;141:1-35.
20. Streett RS. Propositional dynamic logic of looping and converse is elementarily decidable. *Inform Control*. 1982;54(1-2):121-141.
21. Muller DE. Infinite sequences and finite machines. *Proceedings of the Fourth Annual Symposium on Switching Circuit Theory and Logical Design (SWCT 1963)*. IEEE Computer Society; 1963:3-16.

22. Duret-Lutz A, Lewkowicz A, Fauchille A, Michaud T, Renault E, Xu L. Spot 2.0—a framework for LTL and omega-automata manipulation. In: Artho C, Legay A, Peled D, eds. *International Symposium on Automated Technology for Verification and Analysis*. Springer; 2016:122-129.
23. Gastin P, Oddoux D. Fast LTL to Büchi automata translation. In: Berry G, Comon H, Finkel A, eds. *International Conference on Computer Aided Verification*. Springer; 2001:53-65.
24. Baier C, Katoen JP, Larsen KG. *Principles of Model Checking*. MIT Press; 2008.
25. Klein J. ltl2dstar-LTL to deterministic Streett and Rabin automata. 2007.
26. De Giacomo G, Vardi MY. Linear temporal logic and linear dynamic logic on finite traces. In: Rossi F, ed. *IJCAI'13 Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*. Association for Computing Machinery; 2013:854-860.
27. Henriksen JG, Jensen J, Jørgensen M, et al. Mona: monadic second-order logic in practice. In: Brinksma E, Cleavel WR, Larsen KG, Margaria T, Steffen B, eds. *International Workshop on Tools and Algorithms for the Construction and Analysis of Systems*. Springer; 1995:89-110.
28. Wongpiromsarn T, Topcu U, Lamperski A. Automata theory meets barrier certificates: temporal logic verification of nonlinear systems. *IEEE Trans Automat Contr*. 2015;61(11):3344-3355.
29. Liu Y, Zhu JJ, Williams RL II, Wu J. Omni-directional mobile robot controller based on trajectory linearization. *Robot Autonom Syst*. 2008;56(5):461-479.

How to cite this article: Jagtap P, Dimarogonas DV. Controller synthesis against omega-regular specifications: A funnel-based control approach. *Int J Robust Nonlinear Control*. 2024;34(11):7161-7173. doi: 10.1002/rnc.7339