Abstraction-Based Control of Unknown Continuous-Space Models with Just Two Trajectories

Behrad Samari   BEHRAD.SAMARI@NEWCASTLE.AC.UK
Mahdieh Zaker   MAHDIEH.ZAKER@NEWCASTLE.AC.UK
Abolfazl Lavaei   ABOLFAZL.LAVAEI@NEWCASTLE.AC.UK
School of Computing, Newcastle University, United Kingdom

Abstract

Finite abstractions (a.k.a. symbolic models) offer an effective scheme for approximating the complex continuous-space systems with simpler models in the *discrete-space* domain. A crucial aspect, however, is to establish a formal relation between the original system and its symbolic model, ensuring that a discrete controller designed for the symbolic model can be effectively implemented as a *hybrid* controller (using an interface map) for the original system. This task becomes even more challenging when the exact mathematical model of the continuous-space system is unknown. To address this, the existing literature mainly employs *scenario-based* data-driven methods, which require collecting a large amount of data from the original system. In this work, we propose a data-driven framework that utilizes only *two input-state trajectories* collected from unknown nonlinear polynomial systems to synthesize a hybrid controller, enabling the desired behavior on the unknown system through the controller derived from its symbolic model. To accomplish this, we employ the concept of alternating simulation functions (ASFs) to quantify the closeness between the state trajectories of the unknown system and its data-driven symbolic model. By satisfying a specific rank condition on the collected data, which intuitively ensures that the unknown system is persistently excited, we directly design an ASF and its corresponding hybrid controller using finite-length data without explicitly identifying the unknown system, while providing correctness guarantees. This is achieved through proposing a data-based sum-of-squares (SOS) optimization program, enabling a systematic approach to the design process. We illustrate the effectiveness of our data-driven approach through a case study.

Keywords: Abstraction-based control, direct data-driven approaches, symbolic models, alternating simulation functions, correctness guarantees

1. Introduction

Over the past two decades, formal methods have garnered significant attention as a promising technique for synthesizing controllers that enforce high-level logic specifications, such as those expressed by linear temporal logic (LTL) formulae (Pnueli, 1977). These methods enable the reliable control of complex dynamical systems while ensuring rigorous mathematical guarantees. However, synthesizing formal controllers for complex dynamical systems, especially those with nonlinear terms, to satisfy high-level logic properties remains inherently challenging, primarily due to the computational complexity associated with *continuous* state and input spaces.

In order to address the encountered computational challenges, a promising approach in the literature is to construct finite abstractions (also referred to as symbolic models), which serve as approximate representations of continuous-space systems where each finite state corresponds to a group of continuous states from the original (concrete) systems. These finite abstractions can then be used as substitutes for the original systems, enabling verification or controller synthesis on the abstract

models, and subsequently, the results can be translated back to the original complex systems. This is achieved by establishing a similarity relationship between the state trajectories of the original system and its finite abstraction using alternating simulation functions (ASFs), ensuring that the original system satisfies the same property as the abstract model (see the work by Calbert et al. (2024) for a thorough classification of simulation relations).

Symbolic models are generally divided into two distinct classes: *sound and complete* abstractions (Tabuada, 2009), with our approach falling into the latter category. Complete abstractions provide both *necessary and sufficient* guarantees, implying that a controller enforcing a desired property on the abstraction exists *if and only if* such a controller exists for the original system. On the contrary, sound abstractions provide only *sufficient* guarantees, meaning the failure to synthesize a controller using a sound abstraction does not necessarily imply the absence of a controller for the original system. Despite the fruitfulness of symbolic model methods in complex systems analysis, the precise mathematical model of the original system is required for the construction of such (sound or complete) abstract models (see *e.g.,* the studies by Tabuada (2009); Zamani et al. (2014); Girard and Pappas (2009); Reissig et al. (2016); Girard et al. (2009)). Nonetheless, in many practical scenarios, the mathematical models are either unknown or challenging to obtain.

Data-Driven Techniques. To address this key difficulty, data-driven techniques have been established in two main categories: *indirect and direct* (Dörfler et al., 2022), to the second of which our approach belongs. Indirect data-driven approaches focus on learning unknown dynamical models via system identification techniques (Hou and Wang, 2013). However, accurately identifying these models is often computationally demanding, especially for unknown complex systems with nonlinear dynamics. Furthermore, even after identifying an approximate model, it is necessary to derive the similarity relation between the identified dynamics and its symbolic abstraction using model-based techniques. Consequently, the inherent complexity arises at two levels: model identification and deriving the required relation. In contrast, direct data-driven approaches evade the system identification process, instead leveraging data *directly* to construct symbolic models and establish similarity relations (Dörfler et al., 2022).

Data-driven techniques (including indirect and direct) have been widely used thus far to construct symbolic models for unknown systems. Existing results include building symbolic models for unknown systems with a guaranteed confidence (Lavaei and Frazzoli, 2022), construction of symbolic models for monotone systems with disturbances (Makdesi et al., 2021), sample-based sequential (Banse et al., 2023a) and learning-based abstract construction (Hashimoto et al., 2022), probabilistic verification of logic specifications for deterministic systems (Coppola et al., 2022), and synthesizing controllers for unknown systems through symbolic models with provable guarantees (Devonport et al., 2021; Coppola et al., 2024; Banse et al., 2023b; Ajeleye et al., 2023). In addition, Lavaei (2023) and Samari et al. (2024b) propose abstraction-based controller synthesis frameworks for interconnected networks, offering probabilistic and deterministic (*i.e.,* confidence 1) correctness guarantees, respectively.

The majority of the aforementioned literature focuses on symbolic model construction and ensuring their correctness using *scenario-based* approaches (Calafiore and Campi, 2006; Campi et al., 2009). While these methods have proven effective for symbolic model construction in unknown systems, they rely on the assumption that the data are *independent and identically distributed (i.i.d.).* Consequently, only a single input-output (or input-state) data pair can be extracted from each trajectory (Calafiore and Campi, 2006), necessitating *multiple independent trajectories*—potentially up to millions in practical scenarios—due to the *exponential sample complexity* inherent in scenario-based techniques when applied to the robust problem (Esfahani et al., 2014). As a result, scenario-based

methods are particularly well-suited for system simulators, where generating numerous independent trajectories is practical.

In contrast, *trajectory-based* approaches (De Persis and Tesi, 2019) do not require i.i.d. data, allowing a single trajectory to be collected over a specified horizon for analysis, provided that a rank condition is satisfied, ensuring the data is sufficiently informative for the analysis. Several studies have adopted a trajectory-based approach over the past few years, including, but not limited to, synthesizing robust model predictive controllers for linear time-invariant systems (Berberich et al., 2020), input-to-state stabilizing controllers (Chen et al., 2024), and safety controllers utilizing control barrier certificates for continuous-time (Nejati et al., 2022) and discrete-time nonlinear polynomial systems (Samari et al., 2024a). Nevertheless, to the best of our knowledge, no existing research has explored finite abstractions using a trajectory-based approach, due to the inherent complexity of establishing similarity relations.

Original Contributions. Driven by the challenges outlined in prior data-driven studies on constructing symbolic models, this work aims to develop a direct data-driven trajectory-based methodology for unknown discrete-time input-affine nonlinear systems with polynomial dynamics. Our scheme focuses on constructing symbolic models and establishing the similarity relation between the original, unknown systems and their symbolic counterparts. The cornerstone of our framework is the employment of alternating simulation functions (ASFs) with the pursuit of capturing the mismatch between the original system and its symbolic model. Within the proposed approach, we start with gathering just two trajectories (*i.e.,* two sets of input-state data) from the unknown system while ensuring a specific rank condition is satisfied (*i.e.,* a required minimum number of data is gathered), which means that the collected data is rich enough for conducting the analysis. We then construct ASFs directly from the finite-length samples while ensuring correctness guarantees. We note that constructing complete abstractions in our work requires the system to possess the incremental input-to-state stability (delta-ISS) property. Finally, we synthesize a controller for the symbolic model and transfer the result back to the original unknown system, leveraging a hybrid interface function. We showcase the efficacy of our approach via a case study. Due to space limitations, the proofs of all statements are provided in the extended version of our paper (Samari et al., 2024c).

2. Problem Formulation
2.1. Notation

The set of real numbers is denoted by $\mathbb{R}$, with $\mathbb{R}_{\geq 0}$ and $\mathbb{R}_{>0}$ representing the sets of non-negative and positive real numbers, respectively. The set of non-negative integers is expressed as $\mathbb{N}$, while $\mathbb{N}_{\geq 1}$ signifies the set of positive integers. The $n \times n$ identity matrix is denoted by $\mathbb{I}_n$, while $\mathbf{0}_n$ refers to the zero vector of dimension n. For any matrix A, its transpose is represented by $A^\top$. The symbol $\star$ indicates the transposed element in the symmetric position of a symmetric matrix. The notation $P \succ 0$ ($P \succeq 0$) indicates that the *symmetric* matrix P is positive definite (positive semi-definite). For a square matrix P, the smallest and largest eigenvalues are denoted by $\lambda_{\min}(P)$ and $\lambda_{\max}(P)$, respectively. The Euclidean norm of a vector $x \in \mathbb{R}^n$ is denoted by $\|x\|$, while for a given matrix A, $\|A\|$ represents its induced 2-norm. The horizontal concatenation of vectors $x_i \in \mathbb{R}^n$ into an $n \times N$ matrix is written as $\begin{bmatrix} x_1 & x_2 & \ldots & x_N \end{bmatrix}$. The union among the sets $\mathsf{X}_i, i \in \{1, \ldots, n_x\}$ is represented as $\cup_i \mathsf{X}_i$. Given sets X and Y, a relation $\mathscr{R} \subseteq X \times Y$ is a subset of the Cartesian product $X \times Y$ that relates $x \in X$ to $y \in Y$ if $(x, y) \in \mathscr{R}$, equivalently denoted by $x \mathscr{R} y$.

2.2. Discrete-Time Control Systems
We commence by describing the discrete-time control systems for which we desire to construct a symbolic model.

Definition 1 (dt-IANSP) *A discrete-time input-affine nonlinear system with polynomial dynamics (dt-IANSP) is described as*

$$\Sigma\colon x^+ = A\mathcal{M}(x) + Bu, \tag{1}$$

where x^+ denotes the state variables at the subsequent time step, specifically defined as $x^+ := x(k+1)$, $k \in \mathbb{N}$. The matrix $A \in \mathbb{R}^{n \times M}$ represents the system matrix, $B \in \mathbb{R}^{n \times m}$ is the control matrix, and $\mathcal{M}(x) \in \mathbb{R}^M$, with $\mathcal{M}(\mathbf{0}_n) = \mathbf{0}_M$, is a vector comprising monomials of the state variables $x \in X$, with $X \subseteq \mathbb{R}^n$ denoting the state set. The control input $u \in U$ is associated with the system, where $U \subseteq \mathbb{R}^m$ represents the control input set. The dt-IANSP in (1) is represented by the tuple $\Sigma = (A, \mathcal{M}, B, X, U)$.

Both matrices A and B are considered *unknown* in our work, aligning with real-world scenarios. While the exact form of the monomial $\mathcal{M}(x)$ is assumed to be unknown, we are provided with either an *upper bound on its maximum degree*—enabling the construction of $\mathcal{M}(x)$ to encompass all possible combinations of states up to that degree—or an extended vector of monomials, referred to as the *extended dictionary*, which includes all monomials relevant to the system dynamics, along with additional terms that may be unnecessary, to ensure thorough representation of unknown systems. We refer to dt-IANSP (1) as a system with a *(partially) unknown* model.

Remark 1 (On $\mathcal{M}(x)$) *Note that knowing either an upper bound or an extended dictionary for $\mathcal{M}(x)$ is not a restrictive assumption. In fact, in numerous practical applications, such as electrical and mechanical systems, the system dynamics are often derived from first principles. However, the precise parameters of the system, namely the matrices A and B, typically remain unknown, as is the case in our work.*

Having defined the dt-IANSP in (1), we now move on to present how symbolic models can be constructed in the following subsection.

2.3. Symbolic Models

Analyzing the dt-IANSP Σ, which evolves in a *continuous-space domain*, presents substantial challenges. To address this, one can approximate the dt-IANSP Σ with a symbolic model defined by *discrete* sets of states and inputs (Pola et al., 2016). The approximation process begins by partitioning the continuous state and input sets into finite segments, denoted as $X = \cup_i X_i$ and $U = \cup_i U_i$, respectively. Representative points $\hat{x}_i \in X_i$ and $\hat{u}_i \in U_i$ are then selected to serve as *discrete* state variables and control inputs. The following definition formally outlines the required procedure for constructing a symbolic model (Pola et al., 2016; Swikir and Zamani, 2019).

Definition 2 (Symbolic Model Construction) *Given a continuous-space dt-IANSP $\Sigma = (A, \mathcal{M}, B, X, U)$, the corresponding symbolic model $\hat{\Sigma}$ is described as*

$$\hat{\Sigma} = (\hat{f}, \hat{X}, \hat{U}), \tag{2}$$

where the discrete state and input sets are constructed as $\hat{X} := \{\hat{x}_i, i = 1, \ldots, n_{\hat{x}}\}$ and $\hat{U} := \{\hat{u}_i, i = 1, \ldots, n_{\hat{u}}\}$, respectively. The transition map $\hat{f} : \hat{X} \times \hat{U} \to \hat{X}$ is defined as

$$\hat{\Sigma}\colon \hat{x}^+ = \hat{f}(\hat{x}, \hat{u}) = \Pi(A\mathcal{M}(\hat{x}) + B\hat{u}), \tag{3}$$

where the quantization map $\Pi : X \to \hat{X}$ assigns representative points $\hat{x} \in \hat{X}$ and $\hat{u} \in \hat{U}$ to any $x \in X$ and $u \in U$ from the corresponding partitions. This quantization map satisfies the inequality

$$\|\Pi(x) - x\| \leq \delta, \quad \forall x \in X, \tag{4}$$

where $\delta := \sup\{\|x - x'\|, x, x' \in X_i, \ i = 1, 2, \ldots, n_{\hat{x}}\}$ is the state discretization parameter.

Since the symbolic models in Definition 2 operate within a discrete domain, algorithmic methods from computer science can be leveraged to synthesize controllers that enforce complex logical properties (Baier and Katoen, 2008), *e.g.,* reach-while-avoid. The key challenge, however, lies in ensuring that the properties of interest illustrated by the symbolic models are accurately transferred to the original systems. To address this, a similarity relation between the state trajectories of the two systems must be established, which is achieved using the concept of *alternating simulation functions*, as detailed in the following subsection.

2.4. Alternating Simulation Functions

We begin by presenting the concept of an alternating[1] simulation function between a dt-IANSP and its symbolic model. The following definition formalizes this concept (Tabuada, 2009; Swikir and Zamani, 2019).

Definition 3 (ASF) *Consider a dt-IANSP $\Sigma = (A, \mathcal{M}, B, X, U)$ and its symbolic model $\hat{\Sigma} = (\hat{f}, \hat{X}, \hat{U})$. A function $\boldsymbol{S} : X \times \hat{X} \to \mathbb{R}_{\geq 0}$ is called an alternating simulation function (ASF) from $\hat{\Sigma}$ to Σ, denoted as $\hat{\Sigma} \preceq_{\boldsymbol{S}} \Sigma$, if the following conditions are satisfied:*

- *For all $x \in X$ and $\hat{x} \in \hat{X}$:*
$$\alpha \|x - \hat{x}\|^2 \leq \boldsymbol{S}(x, \hat{x}), \tag{5a}$$

- *For all $x \in X$, $\hat{x} \in \hat{X}$, and $\hat{u} \in \hat{U}$, there exists $u \in U$ such that:*
$$\boldsymbol{S}\big(A\mathcal{M}(x) + Bu, \Pi(A\mathcal{M}(\hat{x}) + B\hat{u})\big) \leq \gamma \boldsymbol{S}(x, \hat{x}) + \rho \|\hat{u}\| + \psi, \tag{5b}$$

for some $\alpha, \psi \in \mathbb{R}_{>0}$, $\rho \in \mathbb{R}_{\geq 0}$, and $\gamma \in (0, 1)$.

Remark 2 (Interface Function) *Condition (5b) in Definition 3, essentially ensures the existence of a hybrid interface function $u(x, \hat{x}, \hat{u})$ that meets the inequality (5b). This interface function enables the translation of a synthesized controller $\hat{u}$ for the symbolic model $\hat{\Sigma}$ into a corresponding controller u for the original system Σ.*

Remark 3 (Interpretation of Definition 3) *The ASF defines a similarity relationship between state trajectories of the dt-IANSP Σ and its symbolic model. Specifically, if the initial states of Σ and $\hat{\Sigma}$ are sufficiently close, as quantified by $\boldsymbol{S}(x, \hat{x})$ in (5a), their trajectories will remain in proximity as the systems evolve over time, as described by the right-hand side terms of the inequality (5b) (Tabuada, 2009).*

The following theorem leverages an ASF to measure the distance between the state trajectories of a dt-IANSP Σ and its symbolic model $\hat{\Sigma}$, using the notion of ϵ-approximate alternating simulation relations (Tabuada, 2009, Definition 9.6).

Theorem 1 (Closeness Guarantee) *Consider a dt-IANSP $\Sigma = (A, \mathcal{M}, B, X, U)$ and its symbolic model $\hat{\Sigma} = (\hat{f}, \hat{X}, \hat{U})$. Assume that $\boldsymbol{S}$ is an ASF from $\hat{\Sigma}$ to Σ as defined in Definition 3, and there exists $\nu \in \mathbb{R}_{>0}$ such that $\|\hat{u}\| \leq \nu, \forall \hat{u} \in \hat{U}$. Then, a relation $\mathscr{R} \subseteq X \times \hat{X}$, which is defined as (Tabuada, 2009)*
$$\mathscr{R} := \big\{(x, \hat{x}) \in X \times \hat{X} \mid \boldsymbol{S}(x, \hat{x}) \leq \max\big(\bar{\rho}\nu, \bar{\psi}\big)\big\}, \tag{6a}$$

---

1. While simulation functions suffice for analyzing whether the symbolic model replicates the original system's behavior, controller synthesis requires *alternating* simulation functions to ensure that every control input on the symbolic model is realizable by the original system (Tabuada, 2009).

is an ϵ-approximate alternating simulation relation from $\hat{\Sigma}$ to Σ with

$$\epsilon = \Big(\frac{\max\big(\bar{\rho}\nu, \bar{\psi}\big)}{\alpha}\Big)^{\frac{1}{2}}, \quad where \ \ \bar{\rho} = \frac{(1+\eta_2)\eta_3}{(1-\gamma)\eta_1}\rho, \ \ \bar{\psi} = \frac{(1+\eta_2)\eta_3}{(1-\gamma)(\eta_3-1)\eta_1\eta_2}\psi, \qquad (6b)$$

for any $\eta_1, \eta_2 \in (0,1)$ and $\eta_3 \in (1,2)$.

Remark 4 (Special Case $\rho \equiv 0$) *It is worth noting that in the case of $\rho \equiv 0$ (cf. Theorem 2), the result of Theorem 1 is boiled down to the following:*

$$\epsilon = \Big(\frac{\bar{\psi}}{\alpha}\Big)^{\frac{1}{2}}, \quad where \ \ \bar{\psi} = \frac{\psi}{(1-\gamma)\eta_1}, \quad for \ any \ \eta_1 \in (0,1). \qquad (7)$$

Of note is that Theorem 1 ensures the closeness between the state trajectories of a dt-IANSP and its symbolic model, enabling the enforcement of complex properties such as safety, reachability, and reach-while-avoid (Tabuada, 2009). This approach allows properties to be addressed in simplified *discrete-space* domains and refined for the original *continuous-space* systems while maintaining a quantifiable error bound as defined in (6b), leveraging the capabilities of Theorem 1.

2.5. Data-Driven Symbolic Models

While the mathematical model of the dt-IANSP is unavailable, one can still construct the *symbolic model* $\hat{f}(\hat{x}, \hat{u}) = \Pi(A\mathcal{M}(\hat{x}) + B\hat{u})$ using data by initializing the unknown dt-IANSP with a discrete state $\hat{x}$ under a discrete input $\hat{u}$ to compute $A\mathcal{M}(\hat{x}) + B\hat{u}$. With a state discretization parameter δ and by applying the quantization map Π, the symbolic model $\hat{f}(\hat{x}, \hat{u}) = \Pi(A\mathcal{M}(\hat{x}) + B\hat{u})$ is determined as the *closest representative point* to the value of $A\mathcal{M}(\hat{x}) + B\hat{u}$ that satisfies condition (4). This procedure is repeated for all combinations of discrete state variables $\hat{x} \in \hat{X}$ and input variables $\hat{u} \in \hat{U}$, forming the data-driven symbolic model. This approach is fully aligned with the process of constructing symbolic models in model-based scenarios. It is worth noting that this way of complete symbolic model construction introduces an approximation error, represented by ψ in condition (5b) and correspondingly in relation (6), even in the model-based case. We formally derive a closed-form expression for this error using data, based on the discretization parameter δ, in Theorem 2. In particular, if δ approaches zero, ψ also goes to zero, while the cardinality of the discrete set $\hat{X}$ tends to infinity, illustrating that the discrete set increasingly resembles the continuous space.

Notwithstanding the advantages of constructing ASFs to capture the closeness between the trajectories of two systems, the precise model of the system appears in (5b). However, in many practical scenarios, the exact mathematical model of the system is not available, demonstrating the demand to develop a data-driven approach. In this regard, a few *scenario-based* data-driven frameworks have been proposed to construct ASFs (*e.g.,* (Lavaei and Frazzoli, 2022; Devonport et al., 2021; Lavaei, 2023; Samari et al., 2024b)). While promising, all these studies often require gathering a vast number of i.i.d. samples—sometimes reaching millions—which limits their practical applicability in many scenarios. Note that without formally establishing ASFs, the constructed symbolic model would be ineffective, as its results cannot be transferred back to the original system. Motivated by these key challenges, we now introduce the problem we opt to address in this work.

Problem 1 *Consider a dt-IANSP Σ with unknown matrices A and B, and a given upper bound on the maximum degree of $\mathcal{M}(x)$. Develop a direct data-driven method to construct an ASF, ensuring its correctness guarantees by leveraging only two input-state trajectories. Following this, design a hybrid controller to enforce the desired behaviors over the unknown dt-IANSP, using the controller synthesized for its data-driven symbolic model.*

To address Problem 1, we introduce our data-driven framework in the subsequent section.

3. Data-Driven Framework

This section introduces our proposed data-driven approach for constructing an ASF from the data-driven symbolic model $\hat{\Sigma}$ to the unknown dt-IANSP Σ. To achieve this, we first define the ASF in a *quadratic* form as $\boldsymbol{\mathcal{S}}(x, \hat{x}) = (x - \hat{x})^{\top}\mathcal{P}(x - \hat{x})$, where $\mathcal{P} \succ 0$. Next, assuming that all system states and inputs are measurable, we gather input-state data (*i.e.,* observations) over a time horizon $[0, \mathcal{T}]$, where $\mathcal{T} \in \mathbb{N}_{\geq 1}$ denotes the number of collected samples:

$$\boldsymbol{\mathcal{O}} := \begin{bmatrix} x(0) & x(1) & \ldots & x(\mathcal{T} - 1) \end{bmatrix} \in \mathbb{R}^{n \times \mathcal{T}}, \tag{8a}$$

$$\boldsymbol{\mathcal{I}} := \begin{bmatrix} u(0) & u(1) & \ldots & u(\mathcal{T} - 1) \end{bmatrix} \in \mathbb{R}^{m \times \mathcal{T}}, \tag{8b}$$

$$\boldsymbol{\mathcal{O}}^{+} := \begin{bmatrix} x(1) & x(2) & \ldots & x(\mathcal{T}) \end{bmatrix} \in \mathbb{R}^{n \times \mathcal{T}}. \tag{8c}$$

We refer to (8) as a *single input-state trajectory* from the system. To construct an ASF within our framework, we collect another trajectory, this time from $A\mathcal{M}(\hat{x}) + B\hat{u}$ (referred to as the *second trajectory*), similar to (8), with the same number of samples $\mathcal{T}$. This second trajectory is denoted as $\hat{\boldsymbol{\mathcal{O}}}, \hat{\boldsymbol{\mathcal{I}}}$, and $\hat{\boldsymbol{\mathcal{O}}}^{+}$, in which the initial condition $\hat{x}(0)$ can be selected from the discrete state set $\hat{X}$ and the control inputs from the discrete input set $\hat{U}$. Gathering the second trajectory is necessary within our framework as we ultimately desire to obtain the data-based representation of $A\mathcal{M}(x) + Bu - (A\mathcal{M}(\hat{x}) + B\hat{u})$ (cf. Lemma 1) to construct the ASF (cf. Theorem 2).

It is worth noting that $A\mathcal{M}(\hat{x}) + B\hat{u}$, from which the second trajectory is collected, differs from the symbolic model dynamics in (3) since it does not involve the quantization map Π. Specifically, the incremental difference $A\mathcal{M}(x) + Bu - (A\mathcal{M}(\hat{x}) + B\hat{u})$ is required, as this work aims to construct a *complete* abstraction in accordance with condition (4). This relies on the original system satisfying the incremental input-to-state stability (delta-ISS) property (cf. Swikir and Zamani (2019, Definition 4.1)), which is implicitly reflected in condition (5b) with $\gamma \in (0, 1)$. This enables us to construct an ASF from $\hat{\Sigma}$ to Σ ($\hat{\Sigma} \preceq_{\boldsymbol{\mathcal{S}}} \Sigma$) as well as from Σ to $\hat{\Sigma}$ ($\Sigma \preceq_{\boldsymbol{\mathcal{S}}} \hat{\Sigma}$), *i.e.,* thereby establishing an alternating *bisimulation* function between Σ and $\hat{\Sigma}$ ($\Sigma \cong_{\boldsymbol{\mathcal{S}}} \hat{\Sigma}$)—see Theorem 2 and Remark 8. It is worth noting that the delta-ISS nature of complete abstraction implies that the second trajectory can be even collected from any arbitrary states and inputs within the continuous domain.

Having been inspired by (Guo et al., 2021), in the following lemma, we raise the required conditions along with the proposed hybrid interface function to derive the closed-form data-based representation of $A\mathcal{M}(x) + Bu - (A\mathcal{M}(\hat{x}) + B\hat{u})$, which is subsequently utilized in the main results of the paper.

Lemma 1 (Data-based Representation) *Assume $\boldsymbol{\mathcal{G}_1}(x) \in \mathbb{R}^{\mathcal{T} \times n}$ and $\boldsymbol{\mathcal{G}_2}(\hat{x}) \in \mathbb{R}^{\mathcal{T} \times n}$ are arbitrary matrix-valued functions such that*

$$\Upsilon(x) = \mathbb{M}\boldsymbol{\mathcal{G}_1}(x), \qquad \Upsilon(\hat{x}) = \hat{\mathbb{M}}\boldsymbol{\mathcal{G}_2}(\hat{x}), \tag{9}$$

where $\Upsilon(\cdot) \in \mathbb{R}^{M \times n}$ is a matrix-valued function satisfying the following conditions:

$$\mathcal{M}(x) = \Upsilon(x)x, \qquad \mathcal{M}(\hat{x}) = \Upsilon(\hat{x})\hat{x}. \tag{10}$$

Moreover, $\mathbb{M}$ is a full row-rank *matrix constructed from the vector $\mathcal{M}(x)$ and sampled data $\boldsymbol{\mathcal{O}}$ as*

$$\mathbb{M} = \begin{bmatrix} \mathcal{M}(x(0)) & \mathcal{M}(x(1)) & \ldots & \mathcal{M}(x(\mathcal{T} - 1)) \end{bmatrix} \in \mathbb{R}^{M \times \mathcal{T}}, \tag{11}$$

and $\hat{\mathbb{M}}$ is constructed analogously. If the hybrid interface function is designed as

$$u = \boldsymbol{\mathcal{I}}\boldsymbol{\mathcal{G}_1}(x)x - \hat{\boldsymbol{\mathcal{I}}}\boldsymbol{\mathcal{G}_2}(\hat{x})\hat{x} + \hat{u}, \tag{12}$$

then the difference $A\mathcal{M}(x) + Bu - (A\mathcal{M}(\hat{x}) + B\hat{u})$ can be expressed in the data-based form

$$A\mathcal{M}(x) + Bu - (A\mathcal{M}(\hat{x}) + B\hat{u}) = \boldsymbol{\mathcal{O}}^{+}\boldsymbol{\mathcal{G}_1}(x)x - \hat{\boldsymbol{\mathcal{O}}}^{+}\boldsymbol{\mathcal{G}_2}(\hat{x})\hat{x}. \tag{13}$$

7

Remark 5 (Rank Conditions) *If the matrices $\mathbb{M}$ and $\hat{\mathbb{M}}$ possess full-row rank, this ensures the existence of matrices $\mathcal{G}_1(x)$ and $\mathcal{G}_2(\hat{x})$ that satisfy* (9). *To fulfill this condition, it is necessary to collect at least $M + 1$ data points, implying that $\mathcal{T}$ must satisfy $\mathcal{T} \geq M + 1$. Note that this requirement is readily verifiable since matrices $\mathbb{M}$ and $\hat{\mathbb{M}}$ are constructed from the gathered data.*

Remark 6 (Transformation Matrices) *Since $\mathcal{M}(\mathbf{0}_n) = \mathbf{0}_M$, it is evident that, without loss of generality, transformation matrices $\Upsilon(x)$ and $\Upsilon(\hat{x})$ can always be found to satisfy conditions* (10). *This approach facilitates our framework by representing everything in terms of x and $\hat{x}$ instead of $\mathcal{M}(x)$ and $\mathcal{M}(\hat{x})$, aligning with the structure of our ASF $\mathcal{S}(x, \hat{x}) = (x - \hat{x})^\top \mathcal{P}(x - \hat{x})$.*

With the data-based representation of $A\mathcal{M}(x) + Bu - (A\mathcal{M}(\hat{x}) + B\hat{u})$ established in Lemma 1, we now present the following theorem as the main contribution of this work. This theorem enables the design of an ASF from the data-driven symbolic model $\hat{\Sigma}$ to the original dt-IANSP Σ, relying solely on two trajectories from the unknown system.

Theorem 2 (Data-Driven ASFs) *Consider an unknown dt-IANSP Σ, defined in Definition 1, its data-driven symbolic model $\hat{\Sigma}$, defined in Subsection 2.5, and the data-based representation of $A\mathcal{M}(x) + Bu - A\mathcal{M}(\hat{x}) - B\hat{u}$, as in* (13), *proposed in Lemma 1. Now, if there exist matrix-valued functions $\mathcal{Y}_1(x) \in \mathbb{R}^{\mathcal{T} \times n}$ and $\mathcal{Y}_2(\hat{x}) \in \mathbb{R}^{\mathcal{T} \times n}$, along with constant matrices $\Xi \in \mathbb{R}^{n \times n}$ and $\Theta \in \mathbb{R}^{n \times n}$, with $\Xi \succ 0$, satisfying*

$$\mathbb{M}\mathcal{Y}_1(x) = \Upsilon(x)\Xi, \qquad\qquad \hat{\mathbb{M}}\mathcal{Y}_2(\hat{x}) = \Upsilon(\hat{x})\Xi, \qquad\qquad (14a)$$

$$\mathcal{O}^+\mathcal{Y}_1(x) = \Theta, \qquad\qquad \hat{\mathcal{O}}^+\mathcal{Y}_2(\hat{x}) = \Theta, \qquad\qquad (14b)$$

$$\begin{bmatrix} (\frac{1}{1+\mu})\Xi & \Theta \\ \star & \gamma\Xi \end{bmatrix} \succeq 0, \qquad\qquad (14c)$$

for some $\mu \in \mathbb{R}_{>0}$ and $\gamma \in (0, 1)$, then one can conclude that $\mathcal{S}(x, \hat{x}) = (x - \hat{x})^\top \mathcal{P}(x - \hat{x})$, with $\mathcal{P} := \Xi^{-1}$, is an ASF from the data-driven symbolic model $\hat{\Sigma}$ to the unknown dt-IANSP Σ, with $\alpha = \lambda_{\min}(\mathcal{P}), \rho \equiv 0$, and $\psi = (1 + \frac{1}{\mu})\|\sqrt{\mathcal{P}}\|^2\delta^2$. Furthermore, the interface function is designed as $u = \mathcal{I}\mathcal{Y}_1(x)\mathcal{P}x - \hat{\mathcal{I}}\mathcal{Y}_2(\hat{x})\mathcal{P}\hat{x} + \hat{u}$.

Remark 7 (Enforcement of Conditions (14a)**-**(14c)**)** *To efficiently impose the conditions* (14a)*-*(14c)*, one can leverage existing software tools such as* SOSTOOLS *(Prajna et al., 2004) for fulfilling conditions* (14a)*-*(14b) *in conjunction with semi-definite programming solvers like* SeDuMi *(Sturm, 1999) for solving condition* (14c)*.*

Remark 8 (ASF from Σ to $\hat{\Sigma}$) *While the results of Theorem 2 construct an ASF from $\hat{\Sigma}$ to Σ ($\hat{\Sigma} \preceq_\mathcal{S} \Sigma$), one can readily show that the proposed $\mathcal{S}(x, \hat{x})$ is also an ASF from Σ to $\hat{\Sigma}$ ($\Sigma \preceq_\mathcal{S} \hat{\Sigma}$), thereby establishing an alternating* bisimulation *function between Σ and $\hat{\Sigma}$ ($\Sigma \cong_\mathcal{S} \hat{\Sigma}$). In particular, by the definition of $\hat{U}$, for any $u = \mathcal{I}\mathcal{Y}_1(x)\mathcal{P}x - \hat{\mathcal{I}}\mathcal{Y}_2(\hat{x})\mathcal{P}\hat{x} + \tilde{u} \in U$ there always exists $\hat{u} \in \hat{U}$ such that $\|\tilde{u} - \hat{u}\| \leq \beta_1$, with β_1 being the input discretization parameter. If one follows the proof of Theorem 2 with the new $u = \mathcal{I}\mathcal{Y}_1(x)\mathcal{P}x - \hat{\mathcal{I}}\mathcal{Y}_2(\hat{x})\mathcal{P}\hat{x} + \tilde{u}$, and through addition and subtraction of $A\mathcal{M}(\hat{x}) + B\hat{u}$ (instead of $A\mathcal{M}(\hat{x})$), all proof steps remain valid under the satisfaction of conditions* (14a)*-*(14c)*, with the only modification being the parameter ψ, now constructed as $\psi = (1 + \frac{1}{\mu})\|\sqrt{\mathcal{P}}\|^2(\beta_1^2\beta_2^2 + \delta^2)$, with β_2 being an upper bound on the induced 2-norm of B (cf. Swikir and Zamani (2019, Theorem 4.8) for the model-based analysis).*

One can use the data-driven symbolic models to synthesize discrete controllers $\hat{u}$ capable of satisfying complex logic properties via formal methods tools, such as SCOTS (Rungger and Zamani, 2016). Leveraging the ASF, established by Theorem 2, and its corresponding data-driven interface map, expressed as $u = \mathcal{I}\mathcal{Y}_1(x)\mathcal{P}x - \hat{\mathcal{I}}\mathcal{Y}_2(\hat{x})\mathcal{P}\hat{x} + \hat{u}$, the hybrid controller u can be effectively applied to the unknown dt-IANSP Σ. This ensures that the state trajectories of the two systems remain closely aligned, as guaranteed by the results of Theorem 1.
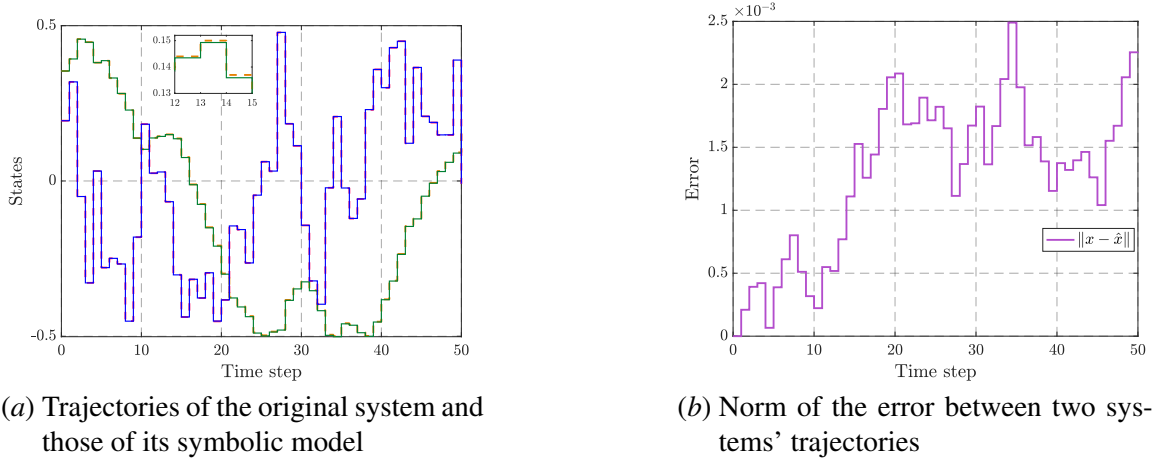
(*a*) Trajectories of the original system and those of its symbolic model

(*b*) Norm of the error between two systems' trajectories

Figure 1: The left-hand side subfigure illustrates the trajectories of x_1 (——), x_2 (——), $\hat{x}_1$ (- - -), and $\hat{x}_2$ (- - -). As shown, it is clear that the safety property is fulfilled. Moreover, the right-hand side subfigure depicts the norm of the error between the trajectories of the original system and those of its symbolic model. As illustrated, the error is always lower than the ϵ reported.
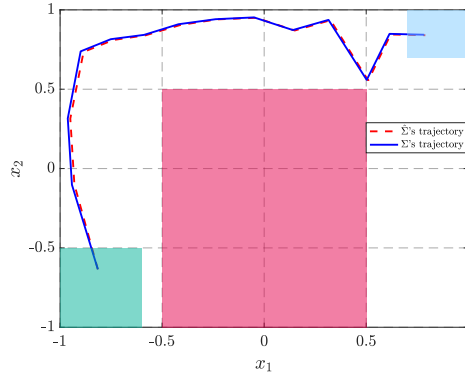

Figure 2: The trajectories must originate from the initial set ▮, reach the target set ▮, while avoiding the obstacle ▮. As shown, the trajectories of the unknown system (——) and its data-driven symbolic model (- - -) closely align, demonstrating compliance with the system's reach-while-avoid property.

4. Simulation Results

In this section, we evaluate the efficacy of our data-driven framework by constructing an ASF and a symbolic model using data, while enforcing two types of specifications over an unknown dt-IANSP: (i) safety and (ii) reach-while-avoid. All simulations were performed in Matlab on a MacBook Pro (Apple M2 Max) with 32GB memory. The details of the system model are provided in Section 4 of the extended version (Samari et al., 2024c).

The main goal is to collect *two input-state trajectories* from the unknown system to design an ASF and a *hybrid* interface map based on the result of Theorem 2. Subsequently, as the safety specification, the data-driven symbolic model is employed to design a discrete controller within the input set $U = [-2.5, 2.5]$, ensuring that the system's states remain within a predefined safe set $X = [-0.5, \ 0.5] \times [-0.5, \ 0.5]$ with correctness guarantees. To do so, under the assumption that the extended dictionary $\mathcal{M}(x) = \begin{bmatrix} x_1 & x_2 & x_1^2 & x_1x_2 & x_2^2 \end{bmatrix}^\top$ is available, we collect two sets of trajectories from the unknown system, as in (8), over the time horizon $\mathcal{T} = 9$. Moreover, we set $\mu = 0.01, \delta = 0.001$, and $\gamma = 0.99$. By solving conditions (14a)-(14c), we design

$$\mathcal{P} = \begin{bmatrix} 5.5818 & 1.6117 \\ 1.6117 & 6.1267 \end{bmatrix}, \quad \Theta = \begin{bmatrix} 0.1837 & -0.0157 \\ -0.0487 & 0.0135 \end{bmatrix},$$

and the hybrid interface map

$$u = -x_1^2 - 1.2495x_1 - 4.9773x_2 + \hat{x}_1^2 + 1.2495\hat{x}_1 + 4.9773\hat{x}_2 + \hat{u}. \tag{15}$$

We note that the parameters μ and γ should be chosen—typically via bisection or trial and error—in a way that yields an acceptably low value of ϵ. Furthermore, based on the designed values, we compute $\psi = 0.0014$ and $\alpha = 4.2197$. Finally, with $\eta_1 = 0.99$, we compute $\bar{\psi} = 0.1414$ and accordingly $\epsilon = 0.1831$ based on Remark 4, which formally quantifies the closeness between the state trajectories of the data-driven symbolic model and the unknown dt-IANSP.

Using the constructed data-driven symbolic model, we proceed to design a controller for the system that ensures the state variables remain within the safe set $X = [-0.5, \ 0.5] \times [-0.5, \ 0.5]$. To achieve this, we first synthesize a discrete controller for the data-driven symbolic model using SCOTS (Rungger and Zamani, 2016). Subsequently, this controller is refined back over the unknown dt-IANSP using the data-driven interface map designed in (15). The closed-loop state trajectories of the unknown system using the hybrid interface map are depicted in Figure 1(a), while the error between the state trajectories of the unknown system and its symbolic model is shown in Figure 1(b). As shown, the synthesized hybrid controller successfully enforces the state variables of the unknown system to remain within the safe set $X = [-0.5, \ 0.5] \times [-0.5, \ 0.5]$.

To further demonstrate the efficacy of our framework, we opt to enforce a *reach-while-avoid* specification over the unknown system without redesigning the interface function (15). This highlights that once the ASF, interface function, and data-driven symbolic model are established, enforcing different specifications on the unknown system becomes readily achievable. The *reach-while-avoid* specification requires the system's position to start within the initial set $[-1, -0.6] \times [-1, -0.5]$, reach the target set $[0.7, 1] \times [0.7, 1]$, while avoiding the obstacle $[-0.5, 0.5] \times [-1, 0.5]$. To achieve this, we utilize the constructed data-driven symbolic model and synthesize a discrete controller employing SCOTS, followed by refining it back over the unknown system leveraging the hybrid interface function (15). Figure 2 provides the simulation result, which shows the practicality of our framework for enforcing such a complex specification. To quantitatively assess data efficiency, we compare our approach with that of (Lavaei and Frazzoli, 2022), which used the *scenario approach* and required $1,100,794$ samples for a different yet 2D nonlinear system. In contrast, our method needs only two input-state trajectories of 9 samples each (18 samples in total), demonstrating a significant advantage in data efficiency.

5. Conclusion

We introduced a direct data-driven abstraction-based methodology that synthesizes a hybrid controller for unknown nonlinear polynomial systems using only *two input-state trajectories*. Our framework ensures that the desired system behavior is achieved by designing a controller derived from a symbolic model. To accomplish this, we utilized ASFs to establish a formal relation between the state trajectories of the unknown system and its symbolic counterpart. Future studies may extend the proposed framework to a broader class of nonlinear systems, potentially accounting for stochasticity in either the system dynamics or the measured data.

References

D. Ajeleye, A. Lavaei, and M. Zamani. Data-driven controller synthesis via finite abstractions with formal guarantees. *IEEE Control Systems Letters*, 7:3453–3458, 2023.

C. Baier and J.-P. Katoen. *Principles of model checking*. MIT press, 2008.

A. Banse, L. Romao, A. Abate, and R. Jungers. Data-driven memory-dependent abstractions of dynamical systems. In *Proceedings of Learning for Dynamics and Control Conference*, pages 891–902. PMLR, 2023a.

A. Banse, L. Romao, A. Abate, and R. M. Jungers. Data-driven abstractions via adaptive refinements and a kantorovich metric. In *Proceedings of the 62nd Conference on Decision and Control (CDC)*, pages 6038–6043. IEEE, 2023b.

J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer. Data-driven model predictive control with stability and robustness guarantees. *IEEE Transactions on Automatic Control*, 66(4):1702–1717, 2020.

G. C. Calafiore and M. C. Campi. The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51(5):742–753, 2006.

J. Calbert, A. Girard, and R. Jungers. Classification of simulation relations for symbolic control. *arXiv:2410.06083*, 2024.

M. C. Campi, S. Garatti, and M. Prandini. The scenario approach for systems and control design. *Annual Reviews in Control*, 33(2):149–157, 2009.

H. Chen, A. Bisoffi, and C. De Persis. Data-driven input-to-state stabilization. *arXiv:2407.06044*, 2024.

R. Coppola, A. Peruffo, and M. Mazo Jr. Data-driven abstractions for verification of deterministic systems. *arXiv:2211.01793*, 2022.

R. Coppola, A. Peruffo, and M. Mazo Jr. Data-driven abstractions for control systems. *arXiv:2402.10668*, 2024.

C. De Persis and P. Tesi. Formulas for data-driven control: Stabilization, optimality, and robustness. *IEEE Transactions on Automatic Control*, 65(3):909–924, 2019.

A. Devonport, A. Saoud, and M. Arcak. Symbolic abstractions from data: A PAC learning approach. In *Proceedings of the 60th Conference on Decision and Control (CDC)*, pages 599–604. IEEE, 2021.

F. Dörfler, J. Coulson, and I. Markovsky. Bridging direct and indirect data-driven control formulations via regularizations and relaxations. *IEEE Transactions on Automatic Control*, 68(2): 883–897, 2022.

P. Mohajerin Esfahani, T. Sutter, and J. Lygeros. Performance bounds for the scenario approach and an extension to a class of non-convex programs. *IEEE Transactions on Automatic Control*, 60(1): 46–58, 2014.

A. Girard and G. J. Pappas. Hierarchical control system design using approximate simulation. *Automatica*, 45(2):566–571, 2009.

A. Girard, G. Pola, and P. Tabuada. Approximately bisimilar symbolic models for incrementally stable switched systems. *IEEE Transactions on Automatic Control*, 55(1):116–126, 2009.

M. Guo, C. De Persis, and P. Tesi. Data-driven stabilization of nonlinear polynomial systems with noisy data. *IEEE Transactions on Automatic Control*, 67(8):4210–4217, 2021.

K. Hashimoto, A. Saoud, M. Kishida, T. Ushio, and D. V. Dimarogonas. Learning-based symbolic abstractions for nonlinear control systems. *Automatica*, 146:110646, 2022.

Z.-S. Hou and Z. Wang. From model-based control to data-driven control: Survey, classification and perspective. *Information Sciences*, 235:3–35, 2013.

A. Lavaei. Symbolic abstractions with guarantees: A data-driven divide-and-conquer strategy. In *Proceedings of the 62nd Conference on Decision and Control (CDC)*, pages 7994–7999. IEEE, 2023.

A. Lavaei and E. Frazzoli. Data-driven synthesis of symbolic abstractions with guaranteed confidence. *IEEE Control Systems Letters*, 7:253–258, 2022.

A. Makdesi, A. Girard, and L. Fribourg. Efficient data-driven abstraction of monotone systems with disturbances. *IFAC-PapersOnLine*, 54(5):49–54, 2021.

A. Nejati, B. Zhong, M. Caccamo, and M. Zamani. Data-driven controller synthesis of unknown nonlinear polynomial systems via control barrier certificates. In *Proceedings of Learning for Dynamics and Control Conference*, pages 763–776. PMLR, 2022.

A. Pnueli. The temporal logic of programs. In *Proceedings of the 18th annual symposium on foundations of computer science*, pages 46–57. IEEE, 1977.

G. Pola, P. Pepe, and M. D. Di Benedetto. Symbolic models for networks of control systems. *IEEE Transactions on Automatic Control*, 61(11):3663–3668, 2016.

S. Prajna, A. Papachristodoulou, P. Seiler, and P. A. Parrilo. SOSTOOLS: Control applications and new developments. In *Proceedings of IEEE International Conference on Robotics and Automation*, pages 315–320, 2004.

G. Reissig, A. Weber, and M. Rungger. Feedback refinement relations for the synthesis of symbolic controllers. *IEEE Transactions on Automatic Control*, 62(4):1781–1796, 2016.

M. Rungger and M. Zamani. SCOTS: a tool for the synthesis of symbolic controllers. In *Proceedings of the 19th international conference on hybrid systems: Computation and control*, pages 99–104, 2016.

B. Samari, O. Akbarzadeh, M. Zaker, and A. Lavaei. From a single trajectory to safety controller synthesis of discrete-time nonlinear polynomial systems. *IEEE Control Systems Letters*, 2024a.

B. Samari, A. Nejati, and A. Lavaei. Data-driven control of large-scale networks with formal guarantees: A small-gain free approach. *arXiv:2411.06743*, 2024b.

B. Samari, M. Zaker, and A. Lavaei. Abstraction-based control of unknown continuous-space models with just two trajectories. *arXiv:2412.03892*, 2024c.

J. F. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11(1-4):625–653, 1999.

A. Swikir and M. Zamani. Compositional synthesis of finite abstractions for networks of systems: A small-gain approach. *Automatica*, 107:551–561, 2019.

P. Tabuada. *Verification and control of hybrid systems: A symbolic approach.* Springer Science & Business Media, 2009.

M. Zamani, P. Mohajerin Esfahani, R. Majumdar, A. Abate, and J. Lygeros. Symbolic control of stochastic systems via approximately bisimilar finite abstractions. *IEEE Transactions on Automatic Control*, 59(12):3135–3150, 2014.