Data-driven memory-dependent abstractions of dynamical systems

Adrien Banse[1] ADRIEN.BANSE@UCLOUVAIN.BE
Licio Romao[2] LICIO.ROMAO@CS.OX.AC.UK
Alessandro Abate[2] ALESSANDRO.ABATE@CS.OX.AC.UK
Raphael M. Jungers[1,2] RAPHAEL.JUNGERS@UCLOUVAIN.BE
[1]*ICTEAM, UCLouvain, Louvain-la-Neuve, Belgium*
[2]*Department of Computer Science, Oxford University, Oxford, United Kingdom*

Editors: N. Matni, M. Morari, G. J. Pappas

Abstract

We propose a sample-based, sequential method to abstract a (potentially black-box) dynamical system with a sequence of memory-dependent Markov chains of increasing size. We show that this approximation alleviates a correlation bias that has been observed in sample-based abstractions. We further propose a methodology to detect on the fly the memory length resulting in an abstraction with sufficient accuracy. We prove that, under reasonable assumptions, the method converges to a sound abstraction in some precise sense, and showcase it on two case studies.

Keywords: dynamical models, switched systems, finite abstractions, memory, ergodicity, observability

1. Introduction

Safety-critical applications, such as autonomous vehicles, traffic control, and space systems, require the control designer to enforce rich temporal properties on trajectories of complex models (Lee and Seshia, 2011). A renown approach to address this goal relies on abstractions (Baier and Katoen, 2008), whereby a finite-state machine (also known as ”symbolic model”) approximates the behaviour of the original (a.k.a. ”concrete”) system that, instead, evolves in a continuous (or even hybrid) state space. Formal verification and correct-by-design synthesis frameworks have been developed by defining mathematical relationships between the finite-state machine and the original dynamics (van der Schaft, 2004; Tabuada, 2009; Reissig et al., 2017; Majumdar et al., 2020b).

Despite the success of abstraction methods, most of the existing techniques rely on full knowledge of the underlying dynamical system (Zamani et al., 2018; Mufid et al., 2019; Majumdar et al., 2020a). This may hamper applicability of these methods when the model is too complex or when it cannot be fully built. For this reason, data-driven methods are gaining popularity (Laurenti et al., 2021; Salamati et al., 2021; Rossa et al., 2021; Wang and Jungers, 2021; Kazemi et al., 2022; Coppola et al., 2022; Badings et al., 2022). In order to generate data-driven abstractions, a common approach consists in sampling the initial condition and observing trajectories of a fixed length that unfold from the sampled points, as in Devonport et al. (2021). Alternative approaches consist in combining backward reachable-set computations and scenario optimization to generate, with a given confidence level, an abstract interval Markov chain (Badings et al., 2022), or in representing noisy dynamics with non-deterministic/probabilistic abstractions (Lahijanian et al., 2015).

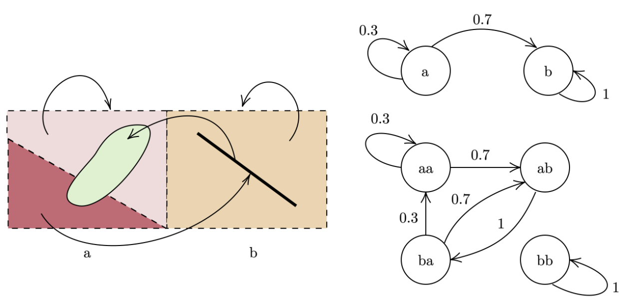

Figure 1: (Left) Pictorial representation of a discrete-time dynamical system. The state-space is partitioned into two cells (labelled a and b) and allowable transitions are indicated by the arrows. (Right) Illustration of two possible abstractions, a memory-1 and a memory-2 one.

Building Markov chain abstractions of dynamical systems must be made with care, as this process may introduce properties in the abstraction that were not present in the original dynamics. As an example of this phenomenon, consider the pictorial discrete-time dynamical system in Figure 1 and a partition of its state space into two cells corresponding to labels a and b. All initial states from the light-red region of a are mapped into the same region, as depicted by the self-loop, and all states in the dark-red region are mapped into a measure-zero subset of b – represented by the black line segment contained in b. Initial states at the yellow region of b are mapped into the same region, and points in the line segment are mapped back into partition a.

On the top-right corner of Figure 1 we illustrate an abstraction obtained by sampling initial conditions from a known distribution and using the frequencies of the different transitions to compute the probabilities shown on the edges; notice that nodes of this model are in one-to-one correspondence with elements in the partition. Using the obtained abstraction to infer transitions of our dynamics leads to erroneous conclusions. First, observe that words abb or $aabb$ may happen with non-zero probability in the top-right model of Figure 1 but are, in fact, not valid trajectories of the original dynamics since each ab must necessarily be followed by an a. We call these words *spurious.* Notice also that the same model does not represent all allowable words. To see this, observe that word aba is not allowed in the abstraction, despite it being a valid word in the original dynamics. We call such words *missing*.

In this paper, we propose a new, sequential approach to build abstractions, where the uncertainty raising from the abstraction step is quantified probabilistically. Such an approach entails turning *epistemic uncertainty* about the dynamics into *aleatoric uncertainty* represented by transition probabilities of the Markov chain, a feature we believe to be unique to our strategy, as far as abstraction of dynamical systems is concerned. By handling abstract probabilistic models, we can analyse the convergence of the probabilistic behaviours. As the abstraction precision increases, we can heuristically estimate the error associated to our models.

Consider now the abstraction illustrated in the bottom-right of Figure 1. The states of this alternative model contains information about one-step transitions, i.e., word ab represents knowledge that we are currently at some state in partition b, which was previously in partition a. Due to richer states, our abstraction can now capture all possible words associated with the dynamics and, as opposed to the memory-1 model, does not possess spurious words. Hence, increasing memory is beneficial to representing dynamical systems.

We prove below that, under some reasonable assumptions, our abstraction procedure converges to the original system in a sense to be described in the sequel. We show on numerical examples that the technique works well even when the assumptions are not satisfied.

The idea of adding memory to produce richer abstractions has been largely explored in different fields of mathematics, engineering, and computer science (see, e.g. Belta et al. (2017); McCallum (1996); Schmuck and Raisch (2014); Frezzatto et al. (2016)). In particular, Coppola et al. (2022) recently proposed it in a non-probabilistic setting. We show here that adding memory is crucial for the construction of probabilistic data-driven abstractions.

This paper is organised as follows. In Section 2, we introduce the setting and describe our data-driven abstraction procedure. In Section 3, we first prove our theoretical results. Numerical experiments are presented in Section 4. We then briefly conclude in Section 5.

2. Definitions and methodology

2.1. Definitions

Consider the discrete-time stochastic system given by

$$S(\nu) = \begin{cases} x_{k+1} \sim T(\cdot|x_k) \\ y_k = H(x_k) \\ x_0 \sim \nu, \end{cases} \tag{1}$$

where $x_k \in X \subset \mathbb{R}^n$ is the state variable, ν is an initial distribution from which the initial state is sampled, $T(\cdot|\cdot) : X \to \mathcal{P}(X) : x_k \mapsto T(\cdot|x_k)$ is a mapping from X to the set of probability measures on X, and $H : X \to \mathcal{A}$ is the *output map*, where $\mathcal{A}$ is a finite set (also referred to as the *output alphabet*). If the stochastic kernel T maps to Dirac distributions, and the initial distribution ν is also a Dirac distribution we say that the system is *deterministic*, and we rewrite the first line of 1 as $x_{k+1} = F(x_k)$ for the sake of clarity.

Assumption A (Measurability) *The mapping $T : X \to \mathcal{P}(X)$ is such that, for any $A \subset X$ the function $g(x) = \int_A T(d\xi|x)$ is measurable and integrable with respect to any measure on X, that is, the integral $\int_A g(\xi)\mu(d\xi)$ is properly defined for any measure $\mu \in \mathcal{P}(X)$.*

Assumption A is a standard technical requirement that enables one to assign probabilities to (sets of) trajectories[1] generated by the stochastic dynamical system (1). The semantics of the dynamical system are denoted as follows: given an initial state $x_0 \sim \nu$, at any time index k, the next state x_{k+1} is defined by sampling according to the probability measure defined by the mapping T, conditional on the current state x_k. Such semantics are known as *stochastic hybrid systems* (Abate et al., 2008).

The output map H induces a partition on the state space as follows. Let $\mathcal{A} = \{y_1, \ldots, y_M\}$ and consider the equivalent relation on $\mathbb{R}^n$ given by $x \approx x'$ if and only if $H(x) = H(x')$. Denote the equivalence classes associated with each element of $\mathcal{A}$ by $[y_j]$, $j = 1, \ldots, M$, i.e., $[y_j] = \{x \in \mathbb{R}^n : H(x) = y_j\}$.

Definition 1 (Probabilities on the states) *For any $k \in \mathbb{N}$, consider the set $A = A_0 \times A_1 \times \ldots \times A_L$, where $A_j \subset \mathbb{R}^n$ for all $j \leq k$, and a probability measure $\mu \in \mathcal{P}(\mathbb{R}^n)$ be given. The dynamical system (1) induces a measure on $\prod_{i=1}^{L+1} \mathbb{R}^n$ that is given by*

$$\mathbb{Q}_L(A) = \int_{A_0} \cdots \int_{A_L} \prod_{j=1}^{L} T(dx_j|x_{j-1})\mu(dx_0).$$

1. For a complete measure-theoretic description of system (1) we refer the reader to Salamon (2016); Tao (2011) and Chapters 2-7 of Rudin (1970).

Definition 2 (Probabilities on the equivalence classes) *For any $L \in \mathbb{N}$, let $\mathbb{Q}_L$ be defined as in Definition 1 and let $y = (y_0, y_1, \dots, y_L)$, where $y_j \in \mathcal{A}$, $j = 0, \dots, L$, be the output of the dynamical system given in (1). Let $A = [y_0] \times [y_1] \times \dots \times [y_L]$ be the set associated with the word y. Then, the probability of such a word is given by $\mathbb{Q}_L(A)$.*

The measures in definitions 1 and 2 are well defined thanks to Assumption A, which ensures that all the nested integrals are well-defined. Next, we formally introduce the set of trajectories that can be observed with probability larger than zero for the system (1), which is also referred to as the behaviour of (1) (see Polderman and Willems (1997) for more details).

Definition 3 (Behaviour) *Consider the dynamical system in (1), let $\mathcal{A}^\star$ be the countable Cartesian product of $\mathcal{A}$ and let $\mathcal{A}^L$ be the L-fold Cartesian product of $\mathcal{A}$. Then we have that:*

- *The* behaviour *of system (1), denoted by $\mathcal{B}(S)$, is the subset of $\mathcal{A}^\star$ defined as $\mathcal{B}(S) = \{y \in \mathcal{A}^\star : \mathbb{Q}(y) > 0\}$, where $\mathbb{Q}$ is the unique measure induced by (1) in the space[2] $\mathcal{A}^\star$.*
- *The L-th step behaviour of the dynamical system (1), denoted by $\mathcal{B}_L(S)$, is a subset of $\mathcal{A}^{L+1}$ defined as $\mathcal{B}_L(S) = \{y = (y_0, \dots, y_L) \in \mathcal{A}^{L+1} : \mathbb{Q}_L(y) > 0\}$.*

The behaviours $\mathcal{B}(S)$ and $\mathcal{B}_L(S)$ are naturally equipped with the probability measures $\mathbb{Q}_{\mathcal{B}(S)}$ and $\mathbb{Q}_{\mathcal{B}_L(S)}$, as per their definition.

In this paper, we will compare the behaviour of a dynamical system to the behaviour of a discrete model that will be described in Section 2.2. Therefore, in addition to the concepts, we provide a notion of measure between two behaviours.

Definition 4 *Given two dynamical systems S_1 and S_2 with the same set of outputs $\mathcal{A}$ as in (1), and a horizon $h \in \mathbb{N}$, we define the measure $d_h(S_1, S_2)$ as*

$$d_h(S_1, S_2) := \mathbb{Q}_{\mathcal{B}_h(S_1)}(\mathcal{B}_h(S_1) \setminus \mathcal{B}_h(S_2)) + \mathbb{Q}_{\mathcal{B}_h(S_2)}(\mathcal{B}_h(S_2) \setminus \mathcal{B}_h(S_1)).$$

For a given horizon h, the measure d_h is the sum of the measure of h-step trajectories that can only be observed for S_1, and the measure of h-step trajectories that can only be observed for S_2.

2.2. Memory-based Markov chains

Inspired by the discussion about the behaviour of the dynamical system depicted in Figure 1, in this section we formalise the syntax and semantics of a memory-based Markov model, which we employ as a template for the abstractions of the given dynamical system.

Definition 5 (Memory-ℓ Markov model) *Let $\ell \in \mathbb{N}$ be a natural number and $\mathcal{A}$ be a finite alphabet. A memory-ℓ Markov model is the 4-tuple $\Sigma_\ell := (\mathcal{S}_\ell, P_\ell, \nu_\ell, H_\ell)$, where $\mathcal{S}_\ell$ is a subset of $\mathcal{A}^\ell$, P_ℓ is the associated stochastic transition matrix, ν_ℓ is the initial state probability, and $H_\ell : \mathcal{S}_\ell \mapsto \mathcal{A}$ is the output (or labelling) map defined as $H_\ell((y_0, \dots, y_{\ell-1})) = y_{\ell-1}$, that is, it is the projection onto the last coordinate of elements of $\mathcal{S}_\ell$. The semantics of the model is a follows: a path $(y^{(0)}, \dots, y^{(L)})$, where each $y^{(j)} \in \mathcal{A}^\ell$, $j = 0, \dots, L$, is an admissible path of size $L + 1$ of a memory-ℓ Markov model $(\mathcal{S}_\ell, P_\ell, \nu_\ell, H_\ell)$ if the following three conditions hold:*

2. This construction can be made rigorous using adequate measure-theoretic results that we omit for brevity, however see Tao (2011) for more details.

1. *Each $y^{(j)}$ is an element of $\mathcal{S}_\ell$ and $y^{(0)}$ is sampled from ν_ℓ.*

2. *For all $j = 0, \dots, L-1$, each $y^{(j+1)}$ is obtained from $y^{(j)}$ by shifting its entries to the left, removing the first element, and inserting an element of $\mathcal{A}$ into the last, empty entry.*

3. *For all $j = 0, \dots, L-1$, we have that $P_\ell(y^{(j+1)} \mid y^{(j)}) > 0$, that is, there is a non-zero probability of transitioning from $y^{(j)}$ to $y^{(j+1)}$.*

Similarly as in Definition 3, we denote by $\mathcal{B}(\Sigma_\ell)$ the behaviour of a memory-ℓ Markov model, which is the collection of all possible outputs that can be observed by running trajectories of the model according to its semantics. The probabilities $\mathbb{Q}_{\mathcal{B}(\Sigma_\ell)}$ and $\mathbb{Q}_{\mathcal{B}_h(\Sigma_\ell)}$ are respectively the unique measures on $\mathcal{B}(\Sigma_\ell)$ and $\mathcal{B}_h(\Sigma_\ell)$ defined by the transition probability and the initial distribution on words. Details are omitted for brevity. An example of a memory-2 Markov model as explained above is depicted on the bottom-right corner in Figure 1, where $y^{(j)}$ belong to $\{aa, ab, ba, bb\}$.

In this work, we compute the measure defined in Definition 4 on two Markov models Σ_{ℓ_1} and Σ_{ℓ_2} where one is a refinement of the other, it is $d_h(\Sigma_{\ell_1}, \Sigma_{\ell_2})$, where $h > \ell_1 > \ell_2$. In that case, the symmetric measure $d_h(\Sigma_{\ell_1}, \Sigma_{\ell_2})$ evaluates the measure of spurious and missing trajectories as defined in Section 1.

2.3. Construction and refinement of probabilistic abstractions

In this subsection we explain in detail our methodology that provides, at every step, an abstract model in the form of a memory-ℓ Markov model, obtained by recording the last ℓ observations. Increasing memory of the Markov representation then leads to more precise abstractions of the original dynamics. Our technique, which is summarized in Algorithm 1, provides a memory-ℓ abstraction for the dynamics in (1). It computes the probability P_ℓ by sampling long trajectories of length $L > \ell$, of the dynamics in (1). The entries of P_ℓ are estimated using the empirical probabilities, i.e., we let

$$P_\ell(y^{(2)} \mid y^{(1)}) = N_{y_0 y_1 \dots y_{\ell-1} y_\ell} \Big/ N_{y^{(1)}}, \tag{2}$$

where $y^{(1)} = y_0 \dots y_{\ell-1}, y^{(2)} = y_1 \dots y_\ell \in \mathcal{A}^\ell$. The symbol N_y, where $y \in \mathcal{A}^\ell$ for some $\ell \in \mathbb{N}$, represents the number of times the word y appears in a word of size $L > \ell$. The procedure consists thus in a Monte-Carlo sampling procedure, where one samples trajectories of length $\ell + 1$ of the form $y_0 y_1 \dots y_{\ell-1} y_\ell$, and increments a counter between $y^{(1)} = y_0 \dots y_{\ell-1}$ and $y^{(2)} = y_1 \dots y_\ell$. For example, if one samples aab, a counter between aa and ab is incremented. Additionally, following the same idea, the initial state distribution for the memory-ℓ Markov model is defined for all $y \in \mathcal{A}^\ell$ by

$$\nu_\ell(y) = N'_y \,/\, N', \tag{3}$$

where N'_y is the number of times the word y appears as the ℓ-long prefix of a L-long sample, and N' is the total number of sampled trajectories of length L.

In our results below, for the sake of clarity, we assume that we know exactly the conditional probabilities defined above. In practice, one would resort to finite sampling, and thereby would imply an estimation error. There are techniques in order to bound this error as, for instance, in Coppola et al. (2022). However, the study of the impact of the sampling error, while certainly of practical importance, is not the focus of the present paper, and we leave it for further work. We formalise this in the next assumption.

Assumption B *For any memory-ℓ Markov model $\Sigma_\ell = (\mathcal{S}_\ell, P_\ell, \nu_\ell, H_\ell)$, we assume that the transition probability P_ℓ and the initial distribution ν_ℓ are known exactly.*

Assumption B also implies that $\mathcal{B}(\Sigma_\ell)$ contains no missing words, that is $\mathcal{B}(\Sigma) \setminus \mathcal{B}(\Sigma_\ell) = \emptyset$, which implies that, for all h, the expression of $d_h(\Sigma_\ell, \Sigma)$ as presented in Definition 4 contains at most one term.

An important feature of our approach is the fact that, irrespective of the memory of the model, the resulting Markov chain is only an approximation of the true dynamics. The reason for this relates to our discussion in the introduction of the paper: the original dynamics may require infinite memory to be represented without errors, and we are instead using a finite memory model, which naturally results in approximation errors. Despite this, we will show that under some hypotheses, successive refinements allow to better approximate the behaviour of a dynamical system. This claim will also be supported by the numerical examples to be presented later.

Algorithm 1 Constructing a memory-ℓ Markov model

1. Fix a horizon $h > 1$, a number of samples $N' \gg 1$, and a sampling length $L \gg h$.
2. Sample N' initial conditions according to initial distribution and simulate N' trajectories of length L.
3. Construct memory-h Markov model (see Def. 5) from samples as described above.
4. Fix $i = 1$, construct the memory-i Markov model from samples, as described above.
5. Compute $d_h(\Sigma_i, \Sigma_h)$ as a proxy of distance between i-memory models and h-step behaviour.
6. If $d_h(\Sigma_i, \Sigma_h)$ is smaller than a given threshold, output memory-i model as final model. If not, $i := i + 1$, and return to item 4.

3. Technical Results

We first state the following elementary proposition.

Proposition 6 *Consider a dynamical system Σ as in (1). For any horizon $h \in \mathbb{N}$, consider a Markov model approximation Σ_h as in Definition 5. It holds that $d_h(\Sigma_h, \Sigma) = 0$, where d_h is defined as in Definition 4.*

Proof By Assumption B and by definition, $\mathcal{B}_h(\Sigma_h)$ contains neither spurious or missing trajectories, that is $\mathcal{B}_h(\Sigma_h) = \mathcal{B}_h(\Sigma)$. Therefore $\mathcal{B}_h(\Sigma_h) = \mathcal{B}_h(\Sigma) \setminus \mathcal{B}_h(\Sigma_h) = \mathcal{B}_h(\Sigma_h) \setminus \mathcal{B}_h(\Sigma) = \emptyset$, which proves the claim. ■

We now present our main result, which provides a justification for the procedure described in Subsection 2.3 and shows that it converges (in some sense) to a correct description of the infinite behaviour of the concrete system. The result leverages two important notions in dynamical systems theory: *observability* and *ergodicity*. In the result, we restrict our analysis to *deterministic* systems, and leave the derivation of a similar result for stochastic systems to future work.

Definition 7 *A deterministic system as in* (1) *is* observable *if for any two trajectories* $x_0x_1\dots$ *and* $x'_0x'_1\dots$ *such that, for all* $i \geq 0$, $H(x_i) = H(x'_i)$, *one has that* $\lim_{l\to\infty} \|x_l - x'_l\| = 0$.

Theorem 8 *Consider a deterministic and observable dynamical system* S *as in* (1)*, and the procedure illustrated in Section 2.3 that generates models* Σ_ℓ, $\ell \in \mathbb{N}$*, as an approximation of the original dynamics. Suppose that Assumption B holds and that the transition function* F *is continuous. Then there exists a sequence* $(\epsilon_\ell)_{\ell\in\mathbb{N}}$ *with* $\lim_{\ell\to\infty}\epsilon_\ell = 0$ *such that the perturbed system*

$$S_{\epsilon_\ell}(\nu) := \begin{cases} x_{k+1} = F(x_k + w_k) \\ y_k = H(x_k) \\ x_0 \sim \nu \end{cases}, \tag{4}$$

where w_k *is a noise sampled from some distribution* $W(x_k, k)$ *for which* $\|w_k\| \leq \epsilon_\ell$*, has the same behaviour as the corresponding Markov model, that is* $\mathcal{B}(S_{\epsilon_\ell}) = \mathcal{B}(\Sigma_\ell)$.

Proof Our proof relies on the implicit existence of an abstraction of the concrete system S. In this abstraction, the abstract states correspond to the equivalence classes

$$[y_0\dots y_{\ell-1}] := \{x : \exists x_0 : F^{\ell-1}(x_0) = x,\ H(F^i(x_0)) = y_i :\ i = 0,\dots,\ell-1\},$$

where F^i denotes the i-th *functional power*[3]. Let $\epsilon(\ell)$ be the diameter of the largest cell of the memory-ℓ Markov model, that is, $\epsilon(\ell) = \max_{y_0,\dots,y_{\ell-1}}\{\mathrm{diam}([y_0\dots y_{\ell-1}])\}$. By observability of F and compactness of X, the maximal diameter $\epsilon(\ell)$ tends to zero. Moreover, it is well known that since F is continuous on the compact X, it admits an invariant measure μ (see (Viana and Oliveira, 2016, Theorem 2.1)). We now prove that, by Birkhoff's theorem (Viana and Oliveira, 2016, Theorem 3.2.3), and assuming perfect sampling by Assumption B, the probability on edge $([y_0\dots y_{\ell-1}], [y_1\dots y_\ell])$ in model Σ_ℓ is equal to

$$\mathbb{P}_\mu(x_{k+1} \in [y_1\dots y_\ell] \,|\, x_k \in [y_0\dots y_{\ell-1}]) := \frac{\mu(\{x \in [y_0,\dots,y_{\ell-1}] : H(F(x)) = y_\ell\})}{\mu([y_0,\dots,y_{\ell-1}])}.$$

Indeed, denoting the indicator function

$$\chi_{[y_0\dots y_\ell]}(x) := \begin{cases} 1 & \text{if } \exists x_0 : x = F^\ell(x_0) \text{ and } H(x_0F(x_0)\dots F^\ell(x_0)) = y_0\dots y_\ell, \\ 0 & \text{otherwise,} \end{cases}$$

and applying Birkhoff's theorem, we have that

$$N_{y_0\dots y_\ell}/N_{y_0\dots y_{\ell-1}} = \int_X \chi_{[y_0\dots y_\ell]}(x)d\mu \Big/ \int_X \chi_{[y_0\dots y_{\ell-1}]}(x)d\mu \tag{5}$$

$$= \int_{x\in[y_0\dots y_{\ell-1}]:H(F(x))=y_\ell} d\mu \Big/ \int_{x\in[y_0\dots y_{\ell-1}]} d\mu \tag{6}$$

$$= \mathbb{P}_\mu(F(x) \in [y_1\dots y_\ell] \,|\, x \in [y_0\dots y_{\ell-1}]). \tag{7}$$

3. For $i = 0$, $f^0 = \mathrm{id}$, the *identity function*, and for $i > 0$, the i-th functional power of some function f is defined inductively as $f^i = f \circ f^{i-1} = f^{i-1} \circ f$.

Equation (5) above follows from the application of Birkhoff's theorem (twice), Equation (6) follows from the invariance of the measure μ, and Equation (7) is the definition of conditional probability.

Now, we claim that we can modify the probabilities $\mathbb{P}_k$, the probability to sample x_k, such that the concrete system behaves as our model. We will first prove the latter for $k = 0$, and then iterate the same argument for times $k > 1$. Consider any probability distribution $\mathbb{P}_0 = \nu$, we show that one can build a probability distribution $\mathbb{P}'_0$ such that their densities $\mathbb{P}'_0([y_0 \dots y_{\ell-1}]) = \mathbb{P}_0([y_0 \dots y_{\ell-1}])$, and such that $\mathbb{P}'_0(x \,|\, x \in [y_0 \dots y_{\ell-1}]) = \mathbb{P}_\mu(x \,|\, x \in [y_0 \dots y_{\ell-1}])$. This new distribution is defined as follows:

$$\mathbb{P}'_0(A) = \mu(A)\frac{\mathbb{P}_0([y_0 \dots y_{\ell-1}])}{\mu([y_0 \dots y_{\ell-1}])},$$

for all $A \in B(X)$, where $B(X)$ is the Borel algebra of X. Moreover, since $\mathbb{P}'_0([y_0 \dots y_{\ell-1}]) = \mathbb{P}_0([y_0 \dots y_{\ell-1}])$, one can express $x' \sim \mathbb{P}'_0$ as $x' = x + w$, where $x \sim \mathbb{P}_0$ and $w \sim W(x, 0)$, and $W(x, 0)$ has support of diameter $\epsilon(\ell)$ (because $W(x, 0)$ perturbs $\mathbb{P}_0$ in the cell to which x belongs). Now, the push-forward measure $\mathbb{P}_1$, defined as $\mathbb{P}_1(A) := \mathbb{P}'_0(F^{-1}(A))$ for all $A \in B(X)$, will not, in general, be equal to μ. However, we can reiterate the construction above and provide a perturbation $\mathbb{P}'_1$ such that $\mathbb{P}'_1([y_0 \dots y_{\ell-1}]) = \mathbb{P}_1([y_0 \dots y_{\ell-1}])$, and such that $\mathbb{P}'_1(x \,|\, x \in [y_0 \dots y_{\ell-1}]) = \mathbb{P}_\mu(x \,|\, x \in [y_0 \dots y_{\ell-1}])$. Again, $\mathbb{P}'_1$ can be achieved by a perturbation $w \sim W(x, 1)$ such that $w < \epsilon(\ell)$, and the proof is concluded by induction. ■

4. Experiments

For a fixed dynamical system S, experiments are set up as follows. For successive values of ℓ, we compute the associated memory-ℓ Markov model $\Sigma_\ell = (\mathcal{S}_\ell, P_\ell, \nu_\ell, H_\ell)$, as explained in subsection 2.3. We also fix a horizon $h > \ell$, for which we compute the corresponding memory-h Markov model $\Sigma_h = (\mathcal{S}_h, P_h, \nu_h, H_h)$. First, for each memory-ℓ model, we compute their measure as defined in Definition 4 with respect to the memory-h model, that is, $d_h(\Sigma_\ell, \Sigma_h)$. This measure is a probabilistic representation of the quality of the memory-ℓ model with respect to $\mathcal{B}_h(S)$, the h-step behaviour of the true system, which we use as a proxy for $\mathcal{B}(S)$. Second, for each pair of memory-ℓ and memory-$(\ell + 1)$ models, we compute their metric with respect to the same horizon h, namely $d_h(\Sigma_\ell, \Sigma_{\ell+1})$. This second measure can be effectively computed in practice, and this distance between models ℓ and $\ell + 1$ allows us to estimate how close our approximations are to convergence. In our experiments, we then verify this by comparing $\mathcal{B}_h(\Sigma_\ell)$ with $\mathcal{B}_h(S)$ (which might not be available in practical applications).

We begin by considering the system generating *Sturmian words* (Fogg et al., 2002).

Example 1 (Deterministic dynamical system) *A* sturmian system *is a deterministic system defined on the state-space* $[0, 2\pi) \subset \mathbb{R}$ *where the next state is defined as*

$$x_{k+1} = F(x_k) = x_k + \theta \mod 2\pi, \tag{8}$$

for some irrational angle θ *and where the output is* $y_k = H(x_k)$*, where* $H(x) = 0$ *if* $x \in [0, \theta)$ *and* $H(x) = 1$ *otherwise. An illustration of the Sturmian dynamics is provided in Figure 2. In the formalism introduced in* (1)*, the alphabet is* $\mathcal{A} = \{0, 1\}$.

We also consider a system of different nature, namely endowed with switching and stochastic behaviour, which has been studied in Dettmann et al. (2020); Stanford and Urbano (1994).

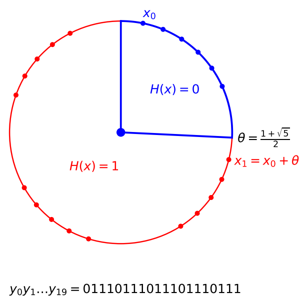

Figure 2: Illustration of the Sturmian dynamical system (see Example 1). The initial state $x_0 \in [0, 2\pi)$ lies in $[0, \theta)$. Therefore, it has an output $y_0 = H(x_0) = 0$. The next state $x_1 = x_0 + \theta$ leaves $[0, \theta)$ and lies in $[\theta, 2\pi)$. Therefore it has an output $y_1 = H(x_1) = 1$. The first 20 states and associated outputs are shown.

Example 2 (Stochastic switched system) *Consider a switched system with two modes defined on the state-space $\mathbb{R}^2$, where the next state is defined as $x_{k+1} = F_{\sigma_k}(x_k)$, where $\sigma_k \in \{1, 2\}$ and the maps $F_i : \mathbb{R}^2 \to \mathbb{R}^2$ are linear maps $F_i(x) = A_i x$ for two matrices $A_1, A_2 \in \mathbb{R}^{2\times 2}$ defined as*

$$A_1 = \begin{pmatrix} \cos(\pi/6) & \sin(\pi/6) \\ -\sin(\pi/6) & \cos(\pi/6) \end{pmatrix} \quad \textit{and} \quad A_2 = \begin{pmatrix} 1.02 & 0 \\ 0 & 1/2 \end{pmatrix}.$$

Suppose in addition that, at each time step, there is a fair probability (equal to $1/2$) to switch to either mode. In the formalism introduced in (1), the stochastic kernel $T(\cdot|x_k)$ is defined as[4]

$$T(\cdot|\cdot) : \mathbb{R}^2 \to \mathcal{P}(\mathbb{R}^2) : x_k \mapsto T(\cdot|x_k) = \frac{1}{2}\delta_{\{F_1(x_k)\}} + \frac{1}{2}\delta_{\{F_2(x_k)\}}. \tag{9}$$

It is not clear whether it is possible to obtain a bi-simulation with classical refinement techniques, and thus we wish to obtain a non-trivial abstraction thanks to the data-driven approach explained in Section 2.3. For this reason, we propose a first rough partition of the state space. The alphabet $\mathcal{A} = \{0, 1, \ldots, 8\}$ and the output function H define a partitioning of the state-space as illustrated in the right part of Figure 3. Together with the output, three trajectories of length 20 are represented in the left of Figure 3.

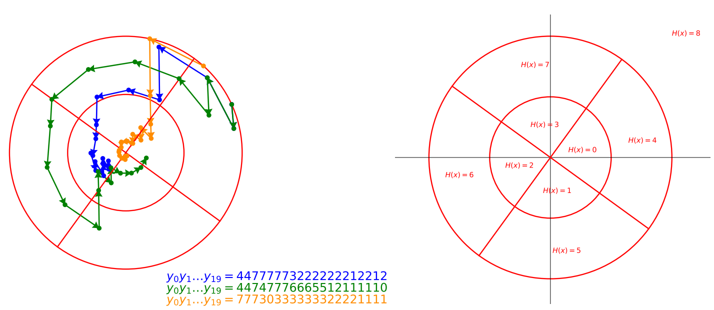

Figure 3: Illustration of the stochastic switched system in Example 2. (Right) The output map H for this system, with circles of radius 1 and 2. (Left) Three different trajectories sampled from the stochastic kernel $T(\cdot|x_k)$ are illustrated, and their output reported in like colour.

In these examples, we assume that one knows the closed-form description of the systems, but would like to find an abstraction of them. Results of multiple executions of the algorithm described above for Example 1 and Example 2 can respectively be found in Figure 4 and Figure 5. One can observe in these figures the red curve $d_h(\Sigma_\ell, \Sigma_{\ell+1})$, which we can compute in practice, and the blue

4. δ_A is the Dirac function, it is $\delta_A(x) = 1$ if $x \in A$, and $\delta_A(x) = 0$ otherwise.

curve $d_h(\Sigma_\ell, \Sigma_h)$, which shows that the successive models indeed converge to the concrete model in terms of their behaviours for the (large) horizon h. This suggests a heuristic argument that, using the method described in Algorithm 1, one can infer the probabilistic precision of the memory-ℓ abstract Markov model with any horizon h. Moreover, in Figure 6 and Figure 7, we display how in practice we can automatically build non-trivial abstractions of the concrete models. Observe that the method works well even for Example 2, which does not satisfy all assumptions of Theorem 8.

The generated abstract models can be further used to perform analysis or verification on the initial system, leveraging information from the probabilistic behaviour of transitions between abstract cells. This goal requires proper handling of the results in Theorem 8, and is left to future work.

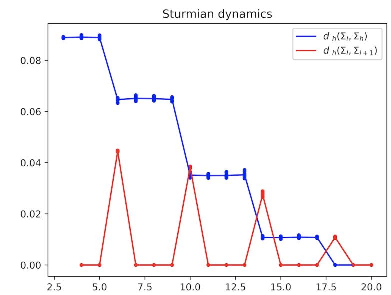

Figure 4: Abstraction results for Example 1. Averages obtained from executing the algorithm 10 times.

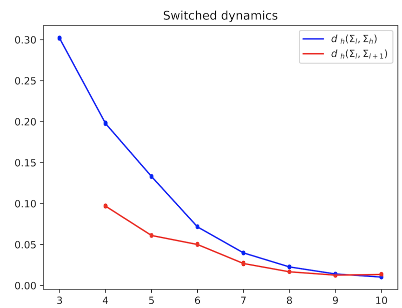

Figure 5: Abstraction results for Example 2. Averages obtained from executing the algorithm 5 times.

Figure 6: State-space partitioning generated by the algorithm for the abstraction built for Example 1, for $\ell = 10$.

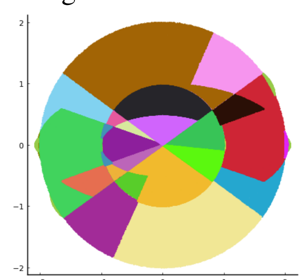

Figure 7: State-space partitioning generated by the algorithm for the abstraction built for Example 2, for $\ell = 2$.

5. Conclusions

In this work, we have proposed a new approach to build data-driven abstraction of rather general dynamical systems. We approximate the concrete system with a Markov model, thus aggregating the (aleatoric and) epistemic nondeterminism of the given model in the exclusively aleatoric uncertainty of the abstract stochastic model.

This technique can be expanded in many directions, both theoretical and practical: by making our computations more efficient, by leveraging the obtained abstraction as an actionable symbolic model, by adding control inputs, or by relaxing or removing some of the raised assumptions. We finally note that, as done recently in a non-probabilistic setting (Yang et al., 2020), one could push this methodology further and refine only certain memory-states, rather than increasing the memory level uniformly from ℓ to $\ell + 1$. We leave this for further work.

References

A. Abate, M. Prandini, J. Lygeros, and S. Sastry. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, 44(11):2724–2734, 2008.

Thom Badings, Licio Romao, Alessandro Abate, D. Parker, H. Poonawala, M. Stoelinga, and N. Jensen. Robust control for dynamical systems with non-gaussian via formal abstractions. *Journal of Artificial Intelligence Research. To appear*, 2022.

Christel Baier and Joost Pieter Katoen. *Principles of Model Checking*. MIT Press Books, 2008.

Calin Belta, Boyan Yordanov, and Ebru Aydin Gol. *Formal Methods for Discrete-time Dynamical Systems*. Springer, 2017.

Rudi Coppola, Andrea Peruffo, and Manuel Mazo. Data-driven abstractions for verification of deterministic systems, 2022. URL https://arxiv.org/abs/2211.01793.

Carl P. Dettmann, Raphael M. Jungers, and Paolo Mason. Lower bounds and dense discontinuity phenomena for the stabilizability radius of linear switched systems. *Systems Control Letters*, 142:104737, 2020.

Alex Devonport, Adnane Saoud, and Murat Arcak. Symbolic abstractions from data: A pac learning approach. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 599–604, 2021.

N Pytheas Fogg, Valéré Berthé, Sébastien Ferenczi, Christian Mauduit, and Anne Siegel. *Substitutions in dynamics, arithmetics and combinatorics*. Springer, 2002.

Luciano Frezzatto, Maurício C. de Oliveira, Ricardo C.L.F. Oliveira, and Pedro L.D. Peres. Robust h filter design with past output measurements for uncertain discrete-time systems. *Automatica*, 71:151–158, 2016.

Milad Kazemi, Rupak Majumdar, Mahmoud Salamati, Sadegh Soudjani, and Ben Wooding. Data-driven abstraction-based control synthesis. 6 2022. URL http://arxiv.org/abs/2206.08069.

Morteza Lahijanian, Sean B. Andersson, and Calin Belta. Formal verification and synthesis for discrete-time stochastic systems. *IEEE Transactions on Automatic Control*, 60:2031–2045, 8 2015.

Luca Laurenti, Morteza Lahijanian, Alessandro Abate, Luca Cardelli, and Marta Kwiatkowska. Formal and efficient synthesis for continuous-time linear stochastic hybrid processes. *IEEE Transactions on Automatic Control*, 66:17–32, 1 2021.

Edward Lee and Sanjit Seshia. *Introduction to Embedded Systems: A Cyber-Physical Approach*. MIT Press, 2011.

Rupak Majumdar, Kaushik Mallik, and Sadegh Soudjani. Symbolic controller synthesis for büchi specifications on stochastic systems. Association for Computing Machinery, Inc, 4 2020a.

Rupak Majumdar, Necmiye Ozay, and Anne Kathrin Schmuck. On abstraction-based controller design with output feedback. Association for Computing Machinery, Inc, 4 2020b.

Andrew Kachites McCallum. *Reinforcement learning with selective perception and hidden state*. University of Rochester, 1996.

Muhammad Syifa'ul Mufid, Dieky Adzkiya, and Alessandro Abate. Bounded model checking of max-plus linear systems via predicate abstractions. volume 11750 LNCS, pages 142–159. Springer, 2019.

Jan Willem Polderman and Jan C. Willems. *Introduction to Mathematical Systems Theory: A Behavioral Approach*. Springer-Verlag, Berlin, Heidelberg, 1997.

Gunther Reissig, Alexander Weber, and Matthias Rungger. Feedback refinement relations for the synthesis of symbolic controllers. *IEEE Transactions on Automatic Control*, 62:1781–1796, 4 2017.

Matteo Della Rossa, Zheming Wang, Lucas N. Egidio, and Raphaël M. Jungers. Data-driven stability analysis of switched affine systems. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 3204–3209, 2021.

Walter Rudin. *Real and Complex Analysis*. McGraw-Hill, 1970.

Ali Salamati, Abolfazl Lavaei, Sadegh Soudjani, and Majid Zamani. Data-driven safety verification of stochastic systems via barrier certificates. *IFAC-PapersOnLine*, 54(5):7–12, 2021. 7th IFAC Conference on Analysis and Design of Hybrid Systems ADHS 2021.

Dietmar Salamon. *Measure and Integration*. European Mathematical Society, 2016, 2016.

Anne Kathrin Schmuck and Jörg Raisch. Asynchronous l-complete approximations. *Systems and Control Letters*, 73:67–75, 2014.

David P. Stanford and José Miguel Urbano. Some convergence properties of matrix sets. *SIAM Journal on Matrix Analysis and Applications*, 15(4):1132–1140, 1994.

Pablo Tabuada. *Verification and Control of Hybrid Systems*. Springer, 2009.

Terence Tao. *An Introduction to Measure Theory*. American Mathematical Society, 2011.

A. van der Schaft. Equivalence of dynamical systems by bisimulation. *IEEE Transactions on Automatic Control*, 49:2160–2172, 2004.

Marcelo Viana and Krerley Oliveira. *Foundations of ergodic theory*. Number 151. Cambridge University Press, 2016.

Zheming Wang and Raphael M. Jungers. A data-driven method for computing polyhedral invariant sets of black-box switched linear systems. *IEEE Control Systems Letters*, 5:1843–1848, 11 2021.

Jung-Min Yang, Thomas Moor, and Jörg Raisch. Refinements of behavioural abstractions for the supervisory control of hybrid systems. *Discrete Event Dynamic Systems*, 30(3):533–560, 2020.

Majid Zamani, Manuel Mazo, Mahmoud Khaled, and Alessandro Abate. Symbolic abstractions of networked control systems. *IEEE Transactions on Control of Network Systems*, 5:1622–1634, 12 2018.