A new look at perfect simulation for chains with infinite memory

Emilio De Santis* and Kádmo Laxa† and Eva Löcherbach‡

October 30, 2025

Abstract

In this article we introduce two new perfect simulation algorithms for chains with infinite memory. Both algorithms belong to the coupling of past procedures. The novelty of our approach is that it allows to include unknown states to the possible past symbols such that we can also deal with sparsely distributed past dependencies. In our first algorithm, spontaneous occurrence of symbols is possible. This means that there is a positive probability that the chain chooses the next symbol independently of the past. Our second algorithm deals with the case in which spontaneous occurrence of symbols is not possible. Chains with infinite memory are discrete-time stochastic processes in which the distribution of the next symbol depends on all past symbols. These transition probabilities are described by a probability kernel. Our results give conditions on the way the dependency of the transition kernel on long past strings decays, guaranteeing that our algorithms stop after a finite number of steps almost surely. Strengthening these conditions, we show that the mean number of steps of our algorithms is finite. We discuss the consequence of having a coupling from the past algorithm with such properties and we present examples in which our results can be applied while others result in the literature cannot be applied.

Keywords: Perfect simulation, Chains of infinite order, Coupling from the past.
AMS MSC: 60G10, 68W20.

1 Introduction

In this article, we introduce two new perfect simulation algorithms for chains with infinite memory. Chains with infinite memory, also known in the literature as *chains of infinite order*, are discrete-time stochastic processes in which the distribution of the next symbol depends on all past symbols. They were first studied by Onisescu and Mihoc (1935) under the name *chains with complete connections*. This class of stochastic processes generalizes the finite memory dependence of Markov chains, and the chains with memory of variable length introduced by Rissanen (1983) are a subclass of them.

By *perfect simulation algorithm* we mean a procedure to obtain a sample of a stochastic process in its stationary regime. To do so, we construct a convenient function that maps independent uniform random variables into the stationary sample of the process. Such a function is called *coupling function*. The key idea is to explore the past to determine a *relevant subset* of all possible past sequences that suffices to sample from the stationary process within a prescribed finite time window.

Commonly used procedures are the *coupling from the past* algorithms which have been introduced in the seminal work of Propp and Wilson (1996) for Markov chains. Comets et al. (2002) extends the ideas of this algorithm to introduce a perfect simulation algorithm for chains with infinite memory. This subject was further explored by other articles, which present different perfect simulation algorithms for chains with infinite memory in the coupling from the past framework (see, for example, Gallo (2011), De Santis and Piccioni (2012), Gallo and Garcia (2013),Gallo and Takahashi (2014) and Garivier (2015)).

*Sapienza University of Rome, Department of Mathematics Piazzale Aldo Moro, 5, 00185, Rome, Italy E-mail: desantis@mat.uniroma1.it

†Faculdade de Filosofia, Ciências e Letras de Ribeirão Preto, Universidade de São Paulo, Av. Bandeirantes, 3900, Ribeirão Preto-SP, 14040-901, Brazil. E-mail: kadmo.laxa@usp.br

‡CMAP, Ecole Polytechnique, Institut Polytechnique de Paris, 91120 Palaiseau, France. E-mail: eva.loecherbach@polytechnique.edu

We propose two perfect simulation algorithms for chains with infinite memory that belong to the coupling of past procedures. The novelty of our approach is that it allows to include unknown states to the possible past symbols such that we can also deal with sparsely distributed past dependencies. In our first algorithm, Algorithm 1, spontaneous occurrence of symbols is possible. This means that there is a positive probability that the chain chooses the next symbol independently of the past. The procedure proposed by Algorithm 1 goes backward in time searching for spontaneously generated symbols. Once a spontaneous symbol is obtained, we go forward in time, updating the symbols that are not spontaneous. This procedure is repeated until the symbols of interest are obtained. Each spontaneous symbol is an information about the past history that works as a source for updating the following symbols.

Algorithm 2 deals with the case in which spontaneous occurrence of symbols is not possible. In this case, the above regenerative construction is not possible any more. Instead, we follow the original idea proposed by Propp and Wilson (1996). We simultaneously construct the process starting from all possible past strings of a certain length. When all these simultaneous processes hit a single symbol, we can assign this symbol as the original process value at this precise time. The symbols generated by this simultaneous construction of the process with different pasts are the source we need to replicate the procedure introduced by Algorithm 1.

Let us now informally present our results. Theorems 1 and 2 show that Algorithms 1 and 2 belong to the family of *coupling from the past* algorithms. We then give conditions on the way the dependency of the transition kernel on long pasts decays guaranteeing that Algorithms 1 and 2 stop after a finite number of steps almost surely. Strengthening these conditions, we show that the mean number of steps of Algorithm 1 and 2 is finite. This is the content of Theorems 3, 4, 5 and 6.

The existence of perfect simulation algorithms that stop after a finite number of steps almost surely, which is guaranteed by Theorems 3, 4, 5 and 6, has mathematical consequences which are well known. The process generated by each of our perfect simulation algorithms is in fact a sample of the unique stationary measure of the chain. This measure satisfies a loss of memory property, and the chain admits the concentration of measure phenomenon. Moreover, there exists a regeneration scheme for the chain. These results are summarized in Propositions 3, 4 and 5.

Our construction is original and, as far as we know, the results we prove are more general than those presented in the literature, since we allow unknown states within the possible past strings such that we may also deal with sparsely distributed past dependencies. In particular, we present examples of chains in which our results guarantee that our perfect simulation algorithm stops in a finite time almost surely, while other results in the literature cannot guarantee this (see Sections 3 and 6).

Our article is organized as follows. In Section 2 we present our algorithms, present the basic notation, and state the main results. In Section 3 we present some examples illustrating the conditions in which our algorithms can be used. In Section 4 we prove Theorems 1 and 2. In Section 5 we prove Theorems 3, 4, 5 and 6. In Section 6 we compare our results with other results in the literature and make some final observations.

2 The algorithms, basic notation and main results

Let $\mathbb{Z} = \{\ldots, -2, -1, 0, +1, 2, \ldots\}$ be the set of integer numbers and $\mathbb{N} = \{0, 1, 2, \ldots\}$ be the set of natural numbers. Consider a set E and let $(a_n)_{n\in\mathbb{Z}} \in E^{\mathbb{Z}}$ be a sequence indexed by $\mathbb{Z}$ with values in E. For any $z \in \mathbb{Z}$ and $m \in \mathbb{N}$, let

$$a_z^{z+m} := (a_{z+m}, a_{z+m-1}, \ldots, a_z) \in E^{m+1}$$

and

$$a_{-\infty}^{z} := (a_z, a_{z-1}, \ldots) \in E^{\mathbb{N}},$$

reading a sequence from the present to the past. By convention, a_z^{z-m} is the empty string if $m > 0$. For any $z_1, z_2 \in \mathbb{Z}$ and $m_1, m_2 \in \mathbb{N}$, we denote the concatenation of the strings $a_{z_1}^{z_1+m_1} \in E^{m_1+1}$ and $b_{z_2}^{z_2+m_2} \in E^{m_2+1}$ as follows

$$a_{z_1}^{z_1+m_1} b_{z_2}^{z_2+m_2} := (a_{z_1+m_1}, a_{z_1+m_1-1}, \ldots, a_{z_1}, b_{z_2+m_2}, b_{z_2+m_2-1}, \ldots, b_{z_2}) \in E^{(m_1+1)+(m_2+1)}.$$

Definition 1. *Let $\mathcal{A}$ be a countable and ordered set. A probability kernel is a function $p : \mathcal{A}\times\mathcal{A}^{\mathbb{N}} \to [0,1]$ such that for any $a_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}}$, $p(\cdot|a_{-\infty}^{-1}) : \mathcal{A} \to [0,1]$ is a probability distribution.*

*A process $(X_n)_{n\in\mathbb{Z}}$ is **compatible** with the probability kernel $p = (p(\cdot|a_{-\infty}^{-1}) : a_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}})$ if for all $g \in \mathcal{A}$, for all n, almost surely,*

$$\mathbb{P}(X_n = g|X_{-\infty}^{n-1}) = p(g|X_{-\infty}^{n-1}).$$

A probability kernel is extended to conditional distributions of finite sequences, given the past, by putting for any $n \geq 2$ and for any $a_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}}$,

$$p(a_{-n}^{-1}|a_{-\infty}^{-(n+1)}) := p(a_{-(n-1)}^{-1}|a_{-\infty}^{-n})p(a_{-n}|a_{-\infty}^{-(n+1)}).$$

In what follows we will often restrict attention to infinite past sequences which are compatible with the transition kernel p. To do so, we define the set of admissible histories as follows

$$\mathcal{H} := \left\{ a_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}} : \sup_{b_{-\infty}^{-(n+1)} \in \mathcal{A}^{\mathbb{N}}} p(a_{-n}^{-1}|b_{-\infty}^{-(n+1)}) > 0, \forall n \geq 1 \right\}. \tag{1}$$

The set $\mathcal{H}$ contains all infinite strings $a_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}}$ in which any finite substring $a_{-n}^{-1} \in \mathcal{A}^n$ is possible, for $n \geq 1$. By possible we mean that there exists $b_{-\infty}^{-(n+1)} \in \mathcal{A}^{\mathbb{N}}$ such that $p(a_{-n}^{-1}|b_{-\infty}^{-(n+1)}) > 0$. $\mathcal{H}$ satisfies the *remove invariance* property ($a_{-\infty}^{-1} \in \mathcal{H}$ implies that $a_{-\infty}^{-2} \in \mathcal{H}$) and the *add invariance* property ($a_{-\infty}^{-1} \in \mathcal{H}$ and $p(g|a_{-\infty}^{-1}) > 0$ implies that $ga_{-\infty}^{-1} \in \mathcal{H}$).

Proposition 1. *If $(X_n)_{n\in\mathbb{Z}}$ is compatible with p, then $\mathbb{P}(X_{-\infty}^n \in \mathcal{H},$ for all $n \in \mathbb{Z}) = 1$.*

Proof. Suppose that $(X_n)_{n\in\mathbb{Z}}$ is compatible with p. For any $m, n \in \mathbb{Z}$, with $m \leq n$, put $E_m^n := \{a_{-\infty}^{-1}X_m^n \notin \mathcal{H},$ for all $a_{-\infty}^{-1} \in \mathcal{H}\}$. Assume that there exists $m, n \in \mathbb{Z}$ such that $\mathbb{P}(E_m^n) > 0$. On E_m^n, there exists $m \leq k = k(\omega) \leq n$ such that $p(X_k|X_{-\infty}^{k-1}) = 0$. This implies that

$$p(X_m^n|X_{-\infty}^{m-1}) = 0 \text{ on } E_m^n. \tag{2}$$

However, conditioning on $\sigma\{X_k, k \leq n-1\}$, we have that

$$\begin{aligned}
\mathbb{P}(p(X_m^n|X_{-\infty}^{m-1}) > 0) &= \sum_{x_n\in\mathcal{A}} \mathbb{E}[1_{\{p(x_n|X_{-\infty}^{n-1})>0\}} 1_{\{X_n=x_n\}} \prod_{j=m}^{n-1} 1_{\{p(X_j|X_{-\infty}^{j-1})>0\}}] \\
&= \sum_{x_n\in\mathcal{A}} \mathbb{E}[1_{\{p(x_n|X_{-\infty}^{n-1})>0\}} p(x_n|X_{-\infty}^{n-1}) \prod_{j=m}^{n-1} 1_{\{p(X_j|X_{-\infty}^{j-1})>0\}}] \\
&= \mathbb{E}[\sum_{x_n\in\mathcal{A}} p(x_n|X_{-\infty}^{n-1}) \prod_{j=m}^{n-1} 1_{\{p(X_j|X_{-\infty}^{j-1})>0\}}] = \mathbb{E}[\prod_{j=m}^{n-1} 1_{\{p(X_j|X_{-\infty}^{j-1})>0\}}].
\end{aligned}$$

Iterating this argument, by conditioning next on $\sigma\{X_k, k \leq n-2\}$, then on $\sigma\{X_k, k \leq n-3\}$, and so on, and finally on $\sigma\{X_k, k \leq m-1\}$, we see that

$$\mathbb{P}(p(X_m^n|X_{-\infty}^{m-1}) > 0) = 1,$$

that is, almost surely, $p(X_m^n|X_{-\infty}^{m-1}) > 0$, which is in contradiction with (2). □

Remark 1. *If we know that there exists a set $\tilde{\mathcal{H}} \subset \mathcal{H}$ such that if $(X_n)_{n\in\mathbb{Z}}$ is compatible with p, then $\mathbb{P}(X_{-\infty}^n \in \tilde{\mathcal{H}},$ for all $n \in \mathbb{Z}) = 1$, then we can replace $\mathcal{H}$ by $\tilde{\mathcal{H}}$ in the sequel.*

The aim of this article is to construct perfect simulation algorithms to simulate X_{-m}^0 for a fixed $m \geq 0$, where $(X_n)_{n\in\mathbb{Z}}$ is compatible with the probability kernel p. We propose two algorithms to achieve this goal. The first algorithm, Algorithm 1, applies for kernels satisfying

$$\beta := \sum_{g\in\mathcal{A}} \inf_{x_{-\infty}^{-1}\in\mathcal{H}} p(g|x_{-\infty}^{-1}) > 0. \tag{3}$$

The second algorithm, Algorithm 2, deals with kernels without this condition. Before presenting the algorithms and summarizing their properties, we have to introduce the following notation.

For any $g \in \mathcal{A}$, let

$$\alpha(g) := \inf_{a_{-\infty}^{-1}\in\mathcal{H}} p(g|a_{-\infty}^{-1}).$$

Note that $\beta = \sum_{g\in\mathcal{A}} \alpha(g)$. Moreover, for any $n \geq 1$, $g \in \mathcal{A}$ and $b_{-n}^{-1} \in \mathcal{A}^n$, let

$$\alpha(g|b_{-n}^{-1}) := \inf\{p(g|a_{-\infty}^{-1}) : a_{-\infty}^{-1} \in \mathcal{H}, a_{-j} = b_{-j}, j = 1, \ldots, n\}, \tag{4}$$

where we put $\inf \emptyset := 0$, and

$$\beta(b_{-n}^{-1}) := \sum_{g\in\mathcal{A}} \alpha(g|b_{-n}^{-1}). \tag{5}$$

In (4) we take the infimum over the set of all histories that have the first n symbols in the past fixed. We extend this notation to the case where the fixed part of the history is not necessarily given by the first symbols. To do so, we adjoin to our state space an extra symbol "$*$" that denotes "unknown states" and put $\mathcal{A}_* = \mathcal{A} \cup \{*\}$.

For any $n \geq 1$, $g \in \mathcal{A}$ and $b_{-n}^{-1} \in \mathcal{A}_*^n$, let

$$\alpha(g|b_{-n}^{-1}) := \inf\{p(g|a_{-\infty}^{-1}) : a_{-\infty}^{-1} \in \mathcal{H}, a_{-j} = b_{-j}\ \forall j \in \{1, \ldots, n\} \text{ with } b_{-j} \in \mathcal{A}\} \tag{6}$$

and

$$\beta(b_{-n}^{-1}) := \sum_{g\in\mathcal{A}} \alpha(g|b_{-n}^{-1}). \tag{7}$$

Finally, let $\alpha(*) := 1 - \beta$ and $\alpha(*|b_{-n}^{-1}) := 1 - \beta(b_{-n}^{-1})$. If $m > 0$, then b_{z+m}^{z} is the empty string, and in this case we use the conventions $\alpha(g|b_{z+m}^{z}) = \alpha(g)$ and $\beta(b_{z+m}^{z}) = \beta$.

Note that (4) is a particular case of (6) and is defined to improve the readability of the text.

Remark 2. *For any $b_{-\infty}^{-1} \in \mathcal{A}_*^{\mathbb{N}}$, it follows that for any $m > n \geq 0$ and for any $g \in \mathcal{A}$,*

$$\alpha(g|b_{-m}^{-1}) \geq \alpha(g|b_{-n}^{-1}).$$

Moreover, if $\tilde{b}_{-\infty}^{-1} \in \mathcal{A}_*^{\mathbb{N}}$ satisfies $\tilde{b}_{-k} \in \{*, b_{-k}\}$, for any $k \geq 1$, then it follows that for any $m \geq 1$ and for any $g \in \mathcal{A}$,

$$\alpha(g|b_{-m}^{-1}) \geq \alpha(g|\tilde{b}_{-m}^{-1}).$$

Note also that $\alpha(g|b_{-n}^{-1}*) = \alpha(g|b_{-n}^{-1})$, for any $g \in \mathcal{A}$ and $b_{-n}^{-1} \in \mathcal{A}_*^n$. In particular, $\alpha(g|*) = \alpha(g)$.

2.1 An introductory example

Before presenting Algorithm 1, let us consider an example that will help us understand how the algorithm works. Let $(U_n)_{n\in\mathbb{Z}}$ be a sequence of i.i.d. uniform random variables in $[0, 1)$.

Consider the finite alphabet $\mathcal{A} = \{1, \ldots, m\}$. We wish to sample X_0 where $(X_n)_{n\in\mathbb{Z}}$ is compatible with p. To do so, we first consider a partition $(I_0(k), 1 \leq k \leq m)$ of the interval $[0, \beta)$ containing m right-open intervals of size $\alpha(1), \ldots, \alpha(m)$, respectively (see Figure 1).

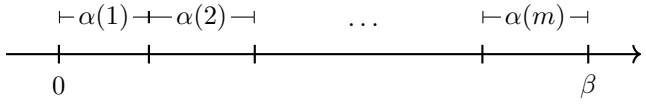


Figure 1

We then sample the uniform random variable U_0. If $U_0 < \beta$, we set $X_0^{(0)} = k$ if and only if $U_0 \in I_0(k)$, that is, U_0 belongs to the k-th sub-interval of $[0, \beta)$ which has size $\alpha(k)$, $1 \leq k \leq m$. In the contrary case, we temporarily set $X_0^{(0)} = *$, meaning that we do not yet know the value of X_0.

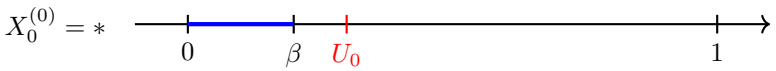


Figure 2

If $X_0^{(0)} = k$, for some $k \in \mathcal{A}$, we stop the algorithm and put $X_0 = X_0^{(0)}$. In the contrary case, we have to continue the algorithm. In our example, $U_0 > \beta$ (see Figure 2), such that we have to continue the algorithm.

So we look back trying to obtain the value of X_{-1}. We sample the random variable U_{-1}. If $U_{-1} < \beta$, we set $X_{-1}^{(-1)} = k$ if and only if $U_{-1} \in I_0(k)$, $1 \leq k \leq m$. In the contrary case, we temporarily set $X_{-1}^{(-1)} = *$ and keep the previous value of the process at time 0, that is, we set $X_0^{(-1)} = *$. In this example $U_{-1} > \beta$ (see Figure 3) such that we have to go even further back into the past.

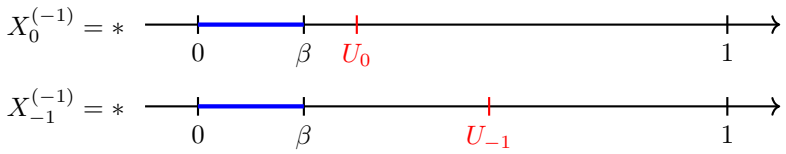


Figure 3

Since $X_{-1}^{(-1)}$ is unknown, we look back trying to obtain the value of X_{-2}. In this example $U_{-2} > \beta$ and then, $X_{-2}^{(-2)} = *$. Finally, in this example $U_{-3} < \beta$ and we denote a_{-3} the index of the interval to which U_{-3} belongs, i.e., $U_{-3} \in I_0(a_{-3})$ which has size $\alpha(a_{-3})$ (see Figure 4). Note that in Figure 4 the values at times $-2, -1$ and 0 are not yet updated.

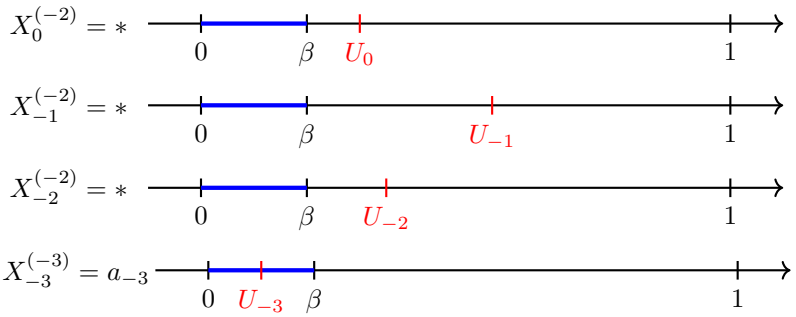
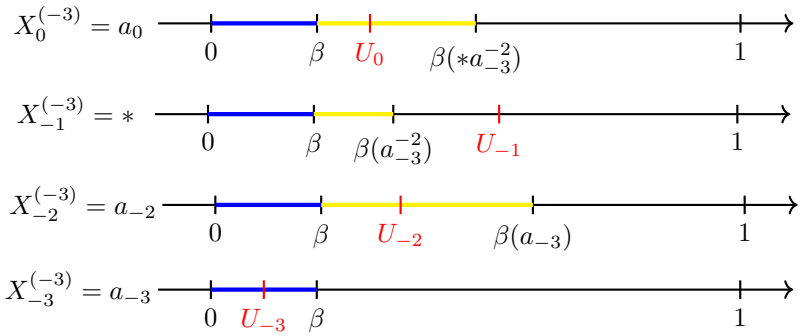


Figure 4

As we have obtained $X_{-3}^{(-3)} \in \mathcal{A}$, we now sequentially check if this is enough to obtain the values of $X_{-2}^{(-3)}, X_{-1}^{(-3)}$ and $X_0^{(-3)}$. For $X_{-2}^{(-3)}$, we consider a partition $(I_1(k, U_{-3}), 1 \leq k \leq m)$ of the interval $[\beta, \beta(a_{-3}))$ containing m right-open intervals of size $\alpha(k|a_{-3}) - \alpha(k)$, respectively. By Remark 2, note that we have that $\alpha(g|b) \geq \alpha(g)$ for any $g \in \mathcal{A}$ and $b \in \mathcal{A}$. If $U_{-2} < \beta(a_{-3})$, we set $X_{-2}^{(-3)} = k$ if and only if $U_{-2} \in I_1(k, U_{-3})$, that is, U_{-2} belongs to the k-th sub-interval of $[\beta, \beta(a_{-3}))$, which has size $\alpha(k|a_{-3})$. In the contrary case, we keep the temporary value $X_{-2}^{(-3)} = *$. In this example, $U_{-2} < \beta(a_{-3})$ and we denote a_{-2} the index of the interval to which U_{-2} belongs (i.e, $U_{-2} \in I_1(a_{-2}, U_{-3})$).

As we have obtained $X_{-3}^{(-3)}$ and $X_{-2}^{(-3)}$, we now check if this is enough to obtain the value of $X_{-1}^{(-3)}$. We consider a partition $(I_2(k, U_{-3}^{-2}), 1 \leq k \leq m)$ of the interval $[\beta, \beta(a_{-3}^{-2}))$ containing m right-open intervals of size $\alpha(k|a_{-3}^{-2}) - \alpha(k)$, respectively. In this example, $U_{-1} > \beta(a_{-3}^{-2})$ and then we keep $X_{-1}^{(-3)} = *$.

Finally, let us check if we have enough information to obtain the value of $X_0^{(-3)}$. We consider a partition $(I_3(k, U_{-3}^{-1}), 1 \leq k \leq m)$ of the interval $[\beta, \beta(*a_{-3}^{-2}))$ containing m right-open intervals of size $\alpha(k| * a_{-3}^{-2}) - \alpha(k)$, respectively. In this example, $U_0 < \beta(*a_{-3}^{-2})$ and then we set $X_0^{(-3)} = a_0$, the index of the interval to which U_0 belongs (i.e, $U_0 \in I_3(a_0, U_{-3}^{-1})$).



Since we have finally obtained a non-star value for the process at time 0, we can stop the algorithm at this point.

2.2 A perfect simulation algorithm with spontaneous symbols

In the following $(U_n)_{n\in\mathbb{Z}}$ are i.i.d. uniform random variables in $[0,1)$. Consider a countable alphabet $\mathcal{A}$, a set of admissible histories $\mathcal{H} \subset \mathcal{A}^{\mathbb{N}}$ associated to a probability kernel $p : \mathcal{A}\times\mathcal{H} \to [0,1]$. Assume that the kernel p satisfies $\beta > 0$. The procedure we propose to obtain a sample $X_{n-k}, \dots, X_n$ for some fixed $k \geq 0$ of a chain compatible with p is given by Algorithm 1, and it is described below.

In what follows, for any n, we will introduce recursively a sequence of temporary values $X_n^{(n-j)}$, where the iteration is with respect to $j \geq 0$, the number of steps of the algorithm. More precisely, $X_n^{(n-j)}$ is the temporary value of X_n based only on U_{n-j}^n. The exponent $(n-j)$ indicates up to which past time we have to go back in the past when considering the past values of the sequence $U_{-\infty}^n$ before time n. Notice that this involves $j+1$ steps of the algorithm.

Definition 2. *For any $n \in \mathbb{Z}$, we define*

$$X_n^{(n)} := \sum_{g\in\mathcal{A}} g\mathbf{1}\left\{U_n \in \left[\sum_{b<g}\alpha(b), \sum_{b\leq g}\alpha(b)\right)\right\} + *\mathbf{1}\left\{U_n \geq \beta\right\}.$$

With the convention $\inf\emptyset = *$, we have that

$$X_n^{(n)} = \inf\left\{g \in \mathcal{A} : U_n < \sum_{b\leq g}\alpha(b)\right\}.$$

With this convention, for any $m \in \mathbb{Z}$ and $n < m$, we define $X_m^{(n)} = X_m^{(n+1)}$ if $X_m^{(n+1)} \in \mathcal{A}$, and

$$X_m^{(n)} := \inf\left\{g \in \mathcal{A} : U_m < \beta((X^{(n+1)})_{n+1}^{m-1}) + \sum_{b\leq g}\alpha(b|(X^{(n)})_n^{m-1}) - \alpha(b|(X^{(n+1)})_{n+1}^{m-1})\right\},$$

in the contrary case. Notice that in the above recursion, $(X^{(n)})_n^{m-1}$ has already been attributed a value. Finally, we recall that by convention, $\beta((X^{(n+1)})_{n+1}^{m-1}) = \beta$ in the case $n = m-1$.

Let us explain the above definitions. Suppose for simplicity that we only want to simulate a single value X_n for a fixed time $n \in \mathbb{Z}$. We first obtain $X_n^{(n)}$, which is the temporary value of X_n based only on U_n. If $X_n^{(n)} \in \mathcal{A}$, then we can stop and the algorithm returns $X_n = X_n^{(n)}$. If not, we obtain $X_{n-1}^{(n-1)}$, the temporary value of X_{n-1} based only on U_{n-1}, and $X_n^{(n-1)}$, the temporary value of X_n based only on U_{n-1} and U_n. If $X_n^{(n-1)} \in \mathcal{A}$, the algorithm returns $X_n = X_n^{(n-1)}$. If not, we obtain $X_{n-2}^{(n-2)}$, $X_{n-1}^{(n-2)}$ and $X_n^{(n-2)}$ and so on.

In Algorithm 1 we keep the convention $\inf\emptyset = *$.

Algorithm 1 PERFECT SIMULATION ALGORITHM TO SAMPLE $X_{-k},\dots,X_0$

Inputs: $k\in\mathbb{N}$. Outputs: X_{-j}, for all $j=-k,\dots,0$.
Initialization: $X_{-j}^{(-n)}\leftarrow *$, for all $j=0,\dots,k$ and $n=j-1,j-2,\dots,-1$.
$n\leftarrow -1$
while $X_{-j}^{(-n)}=*$, for any $j=0,\dots,k$ **do**
  $n\leftarrow n+1$
  Draw the random variable U_{-n}.
  $X_{-n}^{(-n)}\leftarrow \inf\left\{g\in\mathcal{A}: U_{-n}<\sum_{b\leq g}\alpha(b)\right\}$
  for $m=n-1,n-2,\dots,0$ **do**
    if $X_{-m}^{(-n+1)}=*$ and $X_{-n}^{(-n)}\neq *$ **then**
      $X_{-m}^{(-n)}\leftarrow$
      $\inf\left\{g: U_{-m}<\beta((X^{(-n+1)})_{-n+1}^{-m-1})+\sum_{b\leq g}\alpha(b|(X^{(-n)})_{-n}^{-m-1})-\alpha(b|(X^{(-n+1)})_{-n+1}^{-m-1})\right\}$
    else
      $X_{-m}^{(-n)}=X_{-m}^{(-n+1)}$.
return $X_{-k}^{(-n)},\dots,X_0^{(-n)}$.

Definition 3. *For any $n\in\mathbb{Z}$, let*

$$T[n]:=\sup\{j\leq n: X_n^{(j)}\neq *\}. \tag{8}$$

For any $m,n\in\mathbb{Z}$ such that $m<n$, let

$$T[m,n]:=\inf\{T[k],k=m,\dots,n\},$$

and for any $m\in\mathbb{Z}$, let

$$T[m,\infty):=\inf\{T[k],k\geq m\}.$$

Note that $-T[n]$ (recall (8)) is a stopping time with respect to the filtration $(\mathcal{F}_k^{(n)})_{k\geq 0}$, where for any $k\geq 0$, $\mathcal{F}_k^{(n)}=\sigma\{U_{n-k},\dots,U_{n-1},U_n\}$. For any $n\in\mathbb{Z}$, on the event $\{|T[n]|<\infty\}$, the value

$$X_n=X_n^{(T[n])} \tag{9}$$

is exactly the value that Algorithm 1 returns. Note that X_n does not depend on past values of the uniform random variables before time $T[n]$. Finally, note that

$$X_m^{(m-k)}=\begin{cases}*, & \text{if } m-k>T[m],\\ X_m, & \text{if } m-k\leq T[m].\end{cases}$$

In Subsection 2.4 we will give conditions to guarantee that $|T[n]|<\infty$, a.s.

2.3 A perfect simulation algorithm without spontaneous symbols

The assumption $\beta>0$ in Algorithm 1 makes the spontaneous occurrence of symbols possible. Each obtained spontaneous symbol works as a source for updating the following symbols. Without assuming $\beta>0$, we need a different source in order to have a perfect simulation algorithm. In the following, to obtain a new source, we will follow the original idea proposed by Propp and Wilson (1996). This means that we will simultaneously construct the process starting from all possible past strings of a certain length. When all these simultaneous processes hit a single symbol, this symbol will be the value of the original process that we want to simulate at that time. The symbols generated when the processes simultaneously hit a symbol are the source we need to replicate the procedure introduced by Algorithm 1. In the following, we propose an algorithm without the assumption $\beta>0$ and with the mechanism described above. Let us consider some initial definitions.

Definition 4. *Given a probability kernel p, for any $n \geq 1$ we define*

$$\beta_n := \inf\{\beta(a_{-n}^{-1}) : a_{-\infty}^{-1} \in \mathcal{H}\}.$$

For any $n \geq 1$ satisfying $\beta_n > 0$ we define the transition matrix

$$p_{Markov}^{[n]} : \mathcal{A} \times \{a_{-n}^{-1} \in \mathcal{A}^n : a_{-\infty}^{-1} \in \mathcal{H}\} \to [0,1)$$

by

$$p_{Markov}^{[n]}(g|b_{-n}^{-1}) = \frac{\alpha(g|b_{-n}^{-1})}{\beta(b_{-n}^{-1})}.$$

The procedure we propose to obtain a sample $X_{n-k}, \ldots, X_n$ for some fixed $k \geq 0$ of a chain compatible with p without the condition $\beta > 0$ is given by Algorithm 2. For this algorithm, we consider the following assumption. Let us introduce the notation

$$\hat{n} := \inf\{n \in \mathbb{N} : \beta_n > 0 \text{ and only one of the communicating classes associated to } p_{Markov}^{[n]} \text{ is closed, and this class is aperiodic}\},$$

where we put $\inf \emptyset = \infty$.

Assumption 1. *The alphabet $\mathcal{A}$ is finite. The probability kernel p satisfies $\beta = 0$ and $\hat{n} < \infty$.*

Let us denote $\mathcal{U}$ the unique closed class associated to $p_{Markov}^{[\hat{n}]}$. Recall that we intend to simultaneously construct the process starting from all possible past strings of a certain length. This length will be $\hat{n}$. Let

$$\mathcal{C} := \{a_{-\hat{n}}^{-1} \in \mathcal{A}^{\hat{n}} : a_{-\infty}^{-1} \in \mathcal{H}\}$$

and $(U_n^{a_{-\hat{n}}^{-1}} : n \geq 0, a_{-\hat{n}}^{-1} \in \mathcal{C})$ be i.i.d. uniform random variables in $[0,1)$. For any $n \geq 0$ and $a_{-\hat{n}}^{-1} \in \mathcal{C}$, define the process $(Y_n^{a_{-\hat{n}}^{-1}})_{n \geq 0}$ by

$$Y_n^{a_{-\hat{n}}^{-1}} = \inf\left\{ g \in \mathcal{A} : U_n^{a_{-\hat{n}}^{-1}} < \sum_{b \leq g} \alpha(b|(Y^{a_{-\hat{n}}^{-1}})_0^{n-1} a_{-\hat{n}}^{-1}) \right\}, \text{ with the convention } \inf \emptyset = *. \quad (10)$$

This process is a forward in time version of the update procedure proposed by Algorithm 1 where we consider the fixed past $(Y^{a_{-\hat{n}}^{-1}})_{-\hat{n}}^{-1} = a_{-\hat{n}}^{-1}$. The collection of processes $(Y_n)_{n \geq 0} = ((Y_n^{a_{-\hat{n}}^{-1}} : a_{-\hat{n}}^{-1} \in \mathcal{C}))_{n \geq 0}$ provides a coupled construction of the $|\mathcal{C}|$ processes starting from all possible initial pasts in $\mathcal{C}$. This coupled construction lets the different trajectories move independently.

For the next proposition, consider the following definition

$$n_0 := \inf\left\{ m \geq \hat{n} : \mathbb{P}\left((Y^{a_{-\hat{n}}^{-1}})_{m-\hat{n}+1}^{m} = b_{-\hat{n}}^{-1}, \text{ for all } a_{-\hat{n}}^{-1} \in \mathcal{C} \right) > 0, \text{ for all } b_{-\hat{n}}^{-1} \in \mathcal{U} \right\}.$$

Proposition 2. *Suppose that Assumption 1 holds. Then, $n_0 < \infty$.*

Proof. By Assumption 1, we have that $\beta_{\hat{n}} > 0$. Therefore, the probability that $(Y_n^{a_{-\hat{n}}^{-1}})_{n \geq 0}$ follows the $\hat{n}$-Markovian regime for all $a_{-\hat{n}}^{-1} \in \mathcal{C}$ and for all $n = 0, 1, \ldots, m$ is positive, for each fixed $m \geq 0$.

By assumption, only one of the communication classes associated to the transition kernel $p_{Markov}^{[\hat{n}]}$ is closed, and this class is also aperiodic. This implies the following: If we consider independent copies of the Markov chain with transition matrix $p_{Markov}^{[\hat{n}]}$, starting from all possible values in $\mathcal{C}$, then there is a positive probability that after some time m, all these Markov chains visit the same state $b_{-\hat{n}}^{-1} \in \mathcal{U}$ - a string of length $\hat{n}$ - at the same time (Chapter 11.4 of Brémaud (2020)). This property also holds for the original process following the kernel p, since it has a positive probability of following the $\hat{n}$-Markovian regime. This finishes the proof. □

We now describe our algorithm. As before, let $(U_n^{a_{-\hat{n}}^{-1}} : n \geq 0, a_{-\hat{n}}^{-1} \in \mathcal{C})$ be i.i.d. uniform random variables in $[0,1)$. In the procedure proposed by Algorithm 2 we will use time windows $I_j = \{(j-1)n_0 + 1, \ldots, jn_0\}, j \in \mathbb{Z}$.

1. The algorithm considers sequentially the time windows of length n_0, starting from the window I_0. We consider the coupled construction defined in (10) with all possible initial states in $\mathcal{C}$. When these coupled chains hit simultaneously a single symbol which does not equal the star symbol, then the value of the process is the value that was hit simultaneously.

2. After considering the procedure above during the time window I_{-k}, for a given $k \geq 1$, we have to update the value of the process for the times belonging to I_{-j}, $j = k-1, \ldots, 1$. For $n \in I_{-j}$, the update for the value X_n occurs only if the whole string $X_{-jn_0-\hat{n}}^{-jn_0}$ is known. These successive updates follow the ideas of the construction proposed in Algorithm 1.

Let us now describe the algorithm to simulate a single value of the process at a fixed time, say X_0. In this description, we will use the convention $\inf \emptyset = *$. We first obtain $X_m^{(0)}$ for all $m \in I_0$, which are the temporary values of X_m, for $m \in I_0$. These values are constructed using the random variables $(U_m^{a_{-\hat{n}}^{-1}} : m \in I_0, a_{-\hat{n}}^{-1} \in \mathcal{C})$. To do this, we consider for $a_{-\hat{n}}^{-1} \in \mathcal{C}$ and $m \in I_0$,

$$X_m^{a_{-\hat{n}}^{-1}} = \inf \left\{ g \in \mathcal{A} : U_m^{a_{-\hat{n}}^{-1}} < \sum_{b \leq g} \alpha(b|(X^{a_{-\hat{n}}^{-1}})_{-n_0+1}^{m-1} a_{-\hat{n}}^{-1}) \right\}.$$

Notice that this prescription corresponds to a backwards in time version of (10), in the spirit of the Propp and Wilson algorithm.

For any $a_{-\hat{n}}^{-1} \in \mathcal{C}$, to decide about $X_m^{a_{-\hat{n}}^{-1}}$, for any $m \in I_0$, we consider the fixed past $X_{-n_0-\hat{n}+1}^{-n_0} = a_{-\hat{n}}^{-1}$ to update the following values using the procedure proposed by Algorithm 1. The values of $X_m^{a_{-\hat{n}}^{-1}}$ and $X_m^{b_{-\hat{n}}^{-1}}$, for $m \in I_0$ and $a_{-\hat{n}}^{-1} \neq b_{-\hat{n}}^{-1}$ are independent. For $m \in I_0$, if $X_m^{a_{-\hat{n}}^{-1}} = g$ for all $a_{-\hat{n}}^{-1} \in \mathcal{C}$ and for some $g \in \mathcal{A}$, we set $X_m^{(0)} = g$. In other words, we obtain the value of the chain when the trajectories starting from all possible past strings visit the same point. If this is not the case, we set $X_m^{(0)} = *$. Finally, if $X_0^{(0)} \in \mathcal{A}$, the algorithm returns $X_0 = X_0^{(0)}$. This is the first round of the algorithm.

If during the first round of the algorithm we were not able to find a value for X_0, we go further back to the past and inspect the next time window I_{-1} and obtain $X_m^{(-1)}, m \in I_{-1}$, the temporary values of $X_m, m \in I_{-1}$. These values depend on $U_m^{a_{-\hat{n}}^{-1}}, m \in I_{-1}, a_{-\hat{n}}^{-1} \in \mathcal{C}$. To do this, we consider for $a_{-\hat{n}}^{-1} \in \mathcal{C}$ and $m \in I_{-1}$,

$$X_m^{a_{-\hat{n}}^{-1}} = \inf \left\{ g \in \mathcal{A} : U_m^{a_{-\hat{n}}^{-1}} < \sum_{b \leq g} \alpha(b|(X^{a_{-\hat{n}}^{-1}})_{-2n_0+1}^{m-1} a_{-\hat{n}}^{-1}) \right\}.$$

Here, for any $a_{-\hat{n}}^{-1} \in \mathcal{C}$, to decide about $X_m^{a_{-\hat{n}}^{-1}}$, for any $m \in I_{-1}$, we consider the fixed past $X_{-2n_0-\hat{n}+1}^{-2n_0} = a_{-\hat{n}}^{-1}$ to update the following values using the procedure proposed by Algorithm 1. For $m \in I_{-1}$, if $X_m^{a_{-\hat{n}}^{-1}} = g$ for all $a_{-\hat{n}}^{-1} \in \mathcal{C}$ and for some $g \in \mathcal{A}$, we set $X_m^{(-1)} = g$. If not, we set $X_m^{(-1)} = *$. The superscript of $X_m^{(-1)}$ indicates that we had to look back into the past up to the time window I_{-1}.

Now, it is time to obtain $X_m^{(-1)}, m \in I_0$. We first check if $(X^{(-1)})_{-n_0-\hat{n}+1}^{-n_0} = b_{-\hat{n}}^{-1}$, for some $b_{-\hat{n}}^{-1} \in \mathcal{C}$ (in particular, there is no $*$ symbol in this sequence). If not, we set $X_m^{(-1)} = X_m^{(0)}$ for all $m \in I_0$. In the positive case, for $m \in I_0$, if $X_m^{(0)} \neq *$ we set $X_m^{(-1)} = X_m^{(0)}$ and if $X_m^{(0)} = *$, we set $X_m^{(-1)}$ equal to

$$\inf \left\{ g : U_m^{b_{-\hat{n}}^{-1}} < \beta((X^{b_{-\hat{n}}^{-1}})_{-n_0+1}^{m-1} b_{-\hat{n}}^{-1}) + \sum_{b \leq g} \alpha(b|(X^{(-1)})_{-2n_0+1}^{m-1}) - \alpha(b|(X^{b_{-\hat{n}}^{-1}})_{-n_0+1}^{m-1} b_{-\hat{n}}^{-1}) \right\}.$$

This means that we update the values of the process associated to the past string of length $\hat{n}$ that we have obtained, which is $(X^{(-1)})_{-n_0-\hat{n}+1}^{-n_0} = b_{-\hat{n}}^{-1}$. Note that in this update we use the uniform random variable $U_m^{b_{-\hat{n}}^{-1}}$, which is associated to the past we obtained. Finally, if $X_0^{(-1)} \in \mathcal{A}$, the algorithm returns $X_0 = X_0^{(-1)}$, and so on.

To have a complete description of the algorithm, we need to explain how we proceed in case we need to go one step further in the past. The construction of $X_m^{(-2)}, m \in I_{-2} \cup I_{-1}$ is analogous to the construction described above, the difference appears in the construction of $X_m^{(-2)}, m \in I_0$, as we explain in the following. We first check if $(X^{(-2)})_{-n_0-\hat{n}+1}^{-n_0} = b_{-\hat{n}}^{-1}$, for some $b_{-\hat{n}}^{-1} \in \mathcal{C}$. If not, we set $X_m^{(-2)} = X_m^{(-1)}$ for all $m \in I_0$. In the positive case, for $m \in I_0$, if $X_m^{(-1)} \neq *$ we set $X_m^{(-2)} = X_m^{(-1)}$. If $X_m^{(-1)} = *$, we now have two possibilities. If $(X^{(-1)})_{-n_0-\hat{n}+1}^{-n_0} \neq b_{-\hat{n}}^{-1}$ (i.e., we are in the iteration in which the symbols of the process for all times $-n_0 - \hat{n} + 1, \ldots, n_0$ are obtained for the first time), we set $X_m^{(-2)}$ equal to

$$\inf \left\{ g : U_m^{b_{-\hat{n}}^{-1}} < \beta((X^{b_{-\hat{n}}^{-1}})_{-n_0+1}^{m-1} b_{-\hat{n}}^{-1}) + \sum_{b \leq g} \alpha(b|(X^{(-2)})_{-3n_0+1}^{m-1}) - \alpha(b|(X^{b_{-\hat{n}}^{-1}})_{-n_0+1}^{m-1} b_{-\hat{n}}^{-1}) \right\}.$$

In the contrary case, we have that $X_{-m}^{(-2)}$ is equal to

$$\inf \left\{ g : U_{-m}^{b_{-\hat{n}}^{-1}} < \beta((X^{(-1)})_{-2n_0+1}^{-m-1}) + \sum_{b \leq g} \alpha(b|(X^{(-2)})_{-3n_0+1}^{-m-1}) - \alpha(b|(X^{(-1)})_{-2n_0+1}^{-m-1}) \right\}.$$

Let $l_j = n_0(j-1)+1$, and $r_j = n_0 j$ denote the smallest and the greatest value of the set I_j, respectively, for $j \in \mathbb{Z}$. The above description can be generalized to describe the construction of $X_{r_n}^{(m)}, X_{r_n-1}^{(m)} \ldots, X_{l_n}^{(m)}$, for any $n \in \mathbb{Z}$ and for any $m \leq n$. This means that the construction we are considering always starts from times of the form r_n, for $n \in \mathbb{Z}$. In Algorithm 2 we keep the convention $\inf \emptyset = *$

Remark 3. *We considered a coupling construction in which all the trajectories evolve independently. Different constructions would be possible, the choice we made is the one which simplifies the presentation and readability of the algorithm.*

Definition 5. *For any $z \in \mathbb{Z}$ and for any $n \in I_z$, let*

$$\tilde{T}[n] := \sup\{j \leq z : X_n^{(j)} \neq *\}. \tag{11}$$

We define for any $n, m \in \mathbb{Z}$, with $m < n$,

$$\tilde{T}[m, n] := \inf\{\tilde{T}[k], k = m, \ldots, n\}$$

and for any $m \in \mathbb{Z}$,

$$\tilde{T}[m, \infty) := \inf\{\tilde{T}[k], k \geq m\}.$$

Note that $\tilde{T}[n] = k$ means that X_n is obtained from $U_{l_k}, U_{l_k+1}, \ldots, U_n$. For any $n \in \{0, 1, \ldots\}$, on the event $\{|\tilde{T}[n]| < \infty\}$, the value

$$X_n = X_n^{(\tilde{T}[n])} \tag{12}$$

is exactly the value that Algorithm 2 returns. Note that X_n does not depend on past values of the uniform random variables before time $l_{T[n]}$. Finally, note that

$$X_m^{(m-k)} = \begin{cases} *, & \text{if } m - k > \tilde{T}[m], \\ X_m, & \text{if } m - k \leq \tilde{T}[m]. \end{cases}$$

In Subsection 2.4 we will give conditions to guarantee that $|\tilde{T}[n]| < \infty$, a.s.

2.4 Main results

We first give two Theorems stating that Algorithms 1 and 2 belong to the family of algorithms called *coupling from the past* in the literature.

Theorem 1. *Suppose that $|T[n]| < \infty$ a.s. for all $n \in \mathbb{Z}$. Let $(X_n)_{n \in \mathbb{Z}}$ be the output of Algorithm 1. Then there exists a function $F : [0,1)^{\mathbb{N}} \to \mathcal{A}$, such that for any $n \in \mathbb{Z}$,*

$$X_n = F(U_n, U_{n-1}, \ldots, U_{T[n]}, u_{-1}, u_{-2}, \ldots)$$

a.s., for any $u_{-1}, u_{-2}, \ldots \in [0,1)$.

Algorithm 2 PERFECT SIMULATION ALGORITHM TO SAMPLE $X_{-k},\dots,X_0$

Inputs: $k\in\mathbb{N}$. Outputs: X_{-j}, for all $j=-k,\dots,0$.
Initialization: $X_{-j}^{(-n)}\leftarrow *$, for all $j=0,\dots,k$ and $n=j-1,j-2,\dots,-1$.
$n\leftarrow -1$
while $X_{-j}^{(-n)}=*$, for any $j=0,\dots,k$ **do**
  $n\leftarrow n+1$
  for $m=l_{-n},l_{-n}+1,\dots,r_{-n}$: **do**
    for $a_{-\hat{n}}^{-1}\in\mathcal{C}$ **do**
      Draw the random variable $U_m^{a_{-\hat{n}}^{-1}}$.
      $$X_m^{a_{-\hat{n}}^{-1}}\leftarrow \inf\left\{g\in\mathcal{A}: U_m^{a_{-\hat{n}}^{-1}}<\sum_{b\le g}\alpha(b|(X^{a_{-\hat{n}}^{-1}})_{l_{-n}}^{m-1}a_{-\hat{n}}^{-1})\right\}$$
    if $X_m^{a_{-\hat{n}}^{-1}}=g$, for all $a_{-\hat{n}}^{-1}\in\mathcal{C}$ and for some $g\in\mathcal{A}$ **then**
      $X_m^{(-n)}=g$.
    else
      $X_m^{(-n)}=*$.
  for $k=n-1,n-2,\dots,0$: **do**
    if $(X^{(-n)})_{l_{-k}-\hat{n}}^{l_{-k}-1}=b_{-\hat{n}}^{-1}$, for some $b_{-\hat{n}}^{-1}\in\mathcal{C}$ **then**
      for $m=l_{-k},\dots,r_{-k}$: **do**
        if $X_m^{(-n+1)}=*$ **then**
          if $(X^{(-n+1)})_{l_{-k}-\hat{n}}^{l_{-k}-1}=b_{-\hat{n}}^{-1}$ **then**
            $X_m^{(-n)}\leftarrow$
$$\inf\left\{g: U_m^{b_{-\hat{n}}^{-1}}<\beta((X^{(-n+1)})_{l_{-(n-1)}}^{-m-1})+\sum_{b\le g}\alpha(b|(X^{(-n)})_{l_{-n}}^{m-1})-\alpha(b|(X^{(-n+1)})_{l_{-(n-1)}}^{m-1})\right\}$$
          else
            $X_m^{(-n)}\leftarrow$
$$\inf\left\{g: U_m^{b_{-\hat{n}}^{-1}}<\beta((X^{b_{-\hat{n}}^{-1}})_{l_{-k}}^{m-1}b_{-\hat{n}}^{-1})+\sum_{b\le g}\alpha(b|(X^{(-n)})_{l_{-n}}^{m-1})-\alpha(b|(X^{b_{-\hat{n}}^{-1}})_{l_{-k}}^{m-1}b_{-\hat{n}}^{-1})\right\}$$
        else
          $X_m^{(-n)}\leftarrow X_m^{(-n+1)}$.
    else
      $X_m^{(-n)}\leftarrow X_m^{(-n+1)}$.
return $X_{-k}^{(-n)},\dots,X_0^{(-n)}$.

The proof of the above result is given in Section 4 where we construct a function F, called *coupling function* in the sequel, with the desired properties.

Theorem 2. *Suppose that Assumption 1 holds and suppose that $|\tilde{T}[n]|<\infty$ a.s. for all $n\in\mathbb{Z}$. Let $(X_n)_{n\in\mathbb{Z}}$ be the output of Algorithm 2. Then there exist functions*

$$\tilde{F}^{(j)}:[0,1)^{\mathbb{N}}\to\mathcal{A},\ \textit{for } j=0,\dots,n_0-1,$$

such that for any $n\in\mathbb{Z}$ and for any $j=1,\dots,n_0$

$$X_{nn_0-j}=\tilde{F}^{(j)}(U_{nn_0-j},U_{nn_0-j-1},\dots,U_{l_{\tilde{T}[nn_0-j]}},u_{-1},u_{-2},\dots)$$

a.s., for any $u_{-1},u_{-2},\dots\in[0,1)$.

We now state two results, Theorems 3 and 4, which provide sufficient conditions guaranteeing that the assumptions of Theorem 1 hold for Algorithm 1.

For any $n\ge 1$, let

$$\rho_n:=\sum_{a_{-n}^{-1}\in\mathcal{A}^n}\alpha(a_{-n})\alpha(a_{-(n-1)}|a_{-n})\dots\alpha(a_{-1}|a_{-n}^{-2}). \tag{13}$$

Since

$$\rho_n = \sum_{a_{-n}^{-2} \in \mathcal{A}^{n-1}} \alpha(a_{-n})\alpha(a_{-(n-1)}|a_{-n}) \dots \alpha(a_{-2}|a_{-n}^{-3}) \sum_{a_{-1} \in \mathcal{A}} \alpha(a_{-1}|a_{-n}^{-2}) \leq$$

$$\sum_{a_{-n}^{-2} \in \mathcal{A}^{n-1}} \alpha(a_{-n})\alpha(a_{-(n-1)}|a_{-n}) \dots \alpha(a_{-2}|a_{-n}^{-3}) = \rho_{n-1},$$

$(\rho_n)_{n \geq 1}$ is a non-increasing sequence of positive numbers, which therefore admits a limit.

Theorem 3. *Suppose that there exists $c > 0$ such that*

$$\lim_{n \to \infty} \rho_n \geq c.$$

Then, the following holds.

1. $\mathbb{E}(|T[0]|) \leq \frac{1-c}{c}$.
2. *For any $n \geq 1$,* $\mathbb{E}(|T[0,n]|) \leq \frac{(1-c)(n+1)}{c}$.
3. $\mathbb{P}(|T[0,+\infty)| < +\infty) = 1$.

Theorem 4. *Suppose that*

$$\sum_{n=1}^{\infty} \rho_n = \infty.$$

Then, for any $n \geq 1$, $\mathbb{P}(|T[0,n]| < +\infty) = 1$.

We now state two results, Theorems 5 and 6, which provide sufficient conditions guaranteeing that the assumptions of Theorem 2 hold for Algorithm 2.

Recall that $\mathcal{U}$ denotes the unique closed class associated to $p^{[\hat{n}]}_{Markov}$. For any $n \geq 1$, let

$$\tilde{\rho}_n := \sum_{x_{-\hat{n}}^{-1} \in \mathcal{U}} \sum_{a_{-n}^{-1} \in \mathcal{A}^n} \alpha(a_{-n}|x_{-\hat{n}}^{-1})\alpha(a_{-(n-1)}|a_{-n}x_{-\hat{n}}^{-1}) \dots \alpha(a_{-1}|a_{-n}^{-2}x_{-\hat{n}}^{-1}).$$

Note that $(\tilde{\rho}_n)_{n \geq 1}$ is a non-negative and non-increasing sequence, and therefore admits a limit.

Theorem 5. *Suppose that there exists $c > 0$ such that*

$$\lim_{n \to \infty} \tilde{\rho}_n \geq c,$$

Then, the following holds.

1. $\mathbb{E}(|\tilde{T}[0]|) < \infty$.
2. *For any $n \geq 1$,* $\mathbb{E}(|\tilde{T}[0,n]|) < \infty$.
3. $\mathbb{P}(|\tilde{T}[0,+\infty)| < +\infty) = 1$.

Theorem 6. *Suppose that*

$$\sum_{n=1}^{\infty} \tilde{\rho}_n = \infty$$

Then, for any $n \geq 1$, $\mathbb{P}(|\tilde{T}[0,n]| < +\infty) = 1$.

The coupling functions considered in Algorithms 1 and 2 are original, since they allow for unknown positions – those taking the symbol $*$ – in the past. In the coupling from the past framework, other articles consider perfect simulation algorithms based on coupling functions for chains with infinite memory. Comets et al. (2002) and De Santis and Piccioni (2012) consider a perfect simulation algorithm based on the construction of a function $G : [0,1] \times \mathcal{H} \to \mathcal{A}$ for which a similar result as in Theorem 1 or Theorem 2 holds. Once such a construction is achieved, under suitable assumptions, the sequence $(\tilde{X}_n)_{n \in \mathbb{Z}}$ is easily shown to be the unique (in law) process which is compatible with the kernel p. Similar ideas have been presented in Gallo (2011), where a perfect simulation algorithm for chains with memory of variable length is introduced, in Gallo and Garcia (2013), where chains of infinite order are considered having

a transition kernel that is only locally continuous, and in Gallo and Takahashi (2014), where regular probability kernels p on a finite alphabet $\mathcal{A}$ are studied.

This being said, once the construction of a perfect simulation algorithm is achieved, the same consequences as those presented in the above cited works hold also in our case. In particular we have existence and uniqueness of a stationary measure (Corollaries 4.1 and Proposition 6.1 (iii) of Comets et al. (2002) and Corollary 4.1 of Gallo and Garcia (2013)), the loss of memory property (Corollary 4.1 of Comets et al. (2002)), the existence of a regeneration scheme (Corollary 6.1 of Comets et al. (2002) and Corollary 2 of Gallo (2011)) and the concentration of measure phenomenon (Corollary 1 of Gallo and Takahashi (2014)). These results are summarized in the next propositions. Their proofs, which are omitted, use standard arguments and can be found in the articles cited above.

The following propositions are stated considering $(X_n)_{n\in\mathbb{Z}}$ as the output of Algorithm 1 and the associated times as in Definition 3. These propositions also hold when we consider $(X_n)_{n\in\mathbb{Z}}$ as the output of Algorithm 2 and the associated times as in Definition 5.

Proposition 3. *Suppose that $\mathbb{P}(|T[m,n]| < \infty) = 1$, for any $m, n \in \mathbb{Z}$, $m < n$. Then the following holds.*

1. *$(X_n)_{n\in\mathbb{Z}}$ is the unique stationary chain compatible with p.*
2. *(Loss of memory) For any $k \leq 1$, $a_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}}$ and $b_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}}$,*

$$|\mathbb{P}(X_0 \in \cdot | X_{-\infty}^{-k} = a_{-\infty}^{-k}) - \mathbb{P}(X_0 \in \cdot | X_{-\infty}^{-k} = b_{-\infty}^{-k})| \leq \mathbb{P}(|T[0]| \geq k).$$

Proposition 4. *Suppose that $\mathbb{P}(|T[0,+\infty)| < \infty) = 1$. Then the sequence $(\mathbf{1}\{T[n,+\infty) = n\})_{n\in\mathbb{Z}}$ is a stationary renewal process. This implies that $(X_n)_{n\in\mathbb{Z}}$ has a regeneration scheme.*

The following concentration of measure result is due to Corollary 1 of Gallo and Takahashi (2014). Let $f : \mathcal{A}^n \to \mathbb{R}$ be measurable. Define $\delta f = (\delta_1 f, \ldots, \delta_n f)$, where for any $j = 1, \ldots, n$,

$$\delta_j f = \sup\{|f(a_1^n) - f(b_1^n)| : a_i = b_i \text{ for } i \neq j\}.$$

Proposition 5. *Suppose that the alphabet $\mathcal{A}$ is finite and $\mathbb{E}(|T[0]|) < \infty$. Then the following holds. For all $\epsilon > 0$ and all functions $f : \mathcal{A}^n \to \mathbb{R}$,*

$$\mathbb{P}(|f(X_1^n) - \mathbb{E}[f(X_1^n)]| > \epsilon) \leq 4 \exp\left\{ \frac{-2\epsilon^2}{9(1 + \mathbb{E}(T[0]))^2 ||\delta f||_{\ell_2(\mathbb{N})}^2} . \right\}$$

Section 8 of Gallo and Garcia (2013) describes more consequences of the existence of a perfect simulation algorithm such as error bounds for the coupling of chains with infinite memory and Markov chains (see Gallo et al. (2013)), the control of the decay of correlations and functional central limit theorems.

3 Examples

Before giving the proofs of our main results, we present some examples. Examples 7, 8 and 9 and Remark 4 deal with probability kernels for which the assumptions of Algorithm 1 hold. Example 10 and Remark 5 deal with probability kernels for which the assumptions of Algorithm 2 hold. In Section 6 we will come back to these examples to compare our results with previously obtained results in the literature. In the following, for any set A, $\mathbf{1}_A(\cdot)$ denotes the indicator function of A.

Example 7. *Consider a collection of non-negative numbers $\{\theta_j : j \geq 0\}$ such that $\sum_{j=0}^{\infty} \theta_j = 1$. Let us consider the linear autoregressive binary model (see De Santis and Piccioni (2013)) taking values in $\mathcal{A} = \{0, +1\}$ such that for any $x_{-\infty}^{-1} \in \mathcal{H} = \mathcal{A}^{\mathbb{N}}$, the probability kernel is given by*

$$p(1|w_{-\infty}^{-1}) = \theta_0(1 - \delta) + \sum_{j=1}^{\infty} \theta_j w_{-j}.$$

Here, $\delta \in (0, 1)$. In this example, the state of the process at a time is either chosen independently of the past with probability θ_0 or it is the copy of the symbol k steps in the past with probability θ_k.

Assume $\theta_0 > 0$. For this kernel, we have $\beta = \theta_0$ and for any $w_{-n}^{-1} \in \mathcal{A}^n, n \geq 1$, we have $\beta(w_{-n}^{-1}) = \sum_{j=0}^{n} \theta_j$. This implies that for $n \geq 1$, by denoting $s_n = \theta_n + \theta_{n+1} + \ldots$ we have that

$$\rho_n = \prod_{j=1}^{n} (1 - s_j).$$

Note that this is the probability of first choosing a spontaneous symbol, then either choosing a spontaneous symbol or copying the symbol one step in the past, then either choosing a spontaneous symbol or copying a symbol at most two steps back in the past, and so on. If we suppose that

$$\lim_{n \to \infty} \prod_{j=1}^{n} (1 - s_j) > 0, \tag{14}$$

then Theorem 3 holds. The condition above is equivalent to $\sum_{m=1}^{\infty} \ln(1 - s_m) > -\infty$. Since $s_n \to 0$ as $n \to \infty$, this is equivalent to $\sum_{m=1}^{\infty} s_m < \infty$. Therefore, we have to assume that the mean memory length is finite. Here, by memory length we mean the number of steps we look back in the past to copy the symbol at that past time.

A condition that is sufficient to ensure that Theorem 4 holds is the following

$$\sum_{n=1}^{\infty} \prod_{j=1}^{n} (1 - s_j) = +\infty.$$

By noting that

$$\frac{\prod_{j=1}^{n+1} (1 - s_j)}{\prod_{j=1}^{n} (1 - s_j)} = (1 - s_{n+1}),$$

by Raabe's test for infinite series (see Bonar and Khoury (2006)), we have that Theorem 4 holds if there exists $n_0 \in \{1, 2, \ldots\}$ and $\epsilon > 0$ such that

$$s_n \leq \frac{(1 - \epsilon)}{n},$$

for all $n \geq n_0$. Therefore, Theorem 4 holds even in some cases in which the mean memory length is not finite.

The following two examples are imitation models (see Comets et al. (2002), De Santis and Piccioni (2015)) with memory of variable length.

Example 8. *Consider $\mathcal{A} = \{1, 2, \ldots\}$ and let $\{c_g, g \in \mathcal{A}\}$ be a collection of nonnegative numbers such that $c = \sum_{g \in \mathcal{A}} c_g < 1$. For any $x_{-\infty}^{-1} \in \mathcal{H} = \mathcal{A}^{\mathbb{N}}$ and for any $g \in \mathcal{A}$, let*

$$p(g | x_{-\infty}^{-1}) = c_g + (1 - c) \frac{\sum_{k=1}^{x_{-1}} \mathbf{1}\{x_{-k} = g\}}{x_{-1}}. \tag{15}$$

In this example, the state of the process at a time is either chosen independently of the past with probability c or it depends on the past symbols with probability $1 - c$. In the first case, symbol $g \in \mathcal{A}$ is chosen with probability c_g/c. In the second case, we choose uniformly and copy one of the last x_{-1} symbols of the past, where $x_{-1} \in \mathcal{A}$ is the first symbol in the past of the process.

Clearly,

$$\sum_{a_{-n}^{-1} \in \mathcal{A}^n} \alpha(a_{-n}) \alpha(a_{-(n-1)} | a_{-n}) \ldots \alpha(a_{-1} | a_{-n}^{-2}) \geq$$

$$\sum_{a_{-n}^{-1} : a_{-n+k-1} \leq k, k = 1, \ldots, n} \alpha(a_{-n}) \alpha(a_{-(n-1)} | a_{-n}) \ldots \alpha(a_{-1} | a_{-n}^{-2}) \geq$$

$$c_1 (c_1 + c_2 + (1 - c)) \ldots (c_1 + \ldots + c_n + (1 - c)).$$

Indeed, the right-hand side of the above inequality gives the probability of first choosing a spontaneous symbol which equals 1. Thus, if in the next step we either sample a spontaneous symbol that equals either 1 or 2 or if we decide, with probability $1 - c$, to look into the past, then the

memory length, which is given by the first symbol, is given by 1. *This argument can then be iterated. Note that the above event is a sort of ladder event* $X_1 = 1$, *followed by* $X_2 \leq 2$, $X_3 \leq 3$ *(*$X_3 = 1$ *is a possible choice) and so on.*

If we suppose that

$$\lim_{n\to\infty} c_1(c_1 + c_2 + (1-c)) \dots (c_1 + \dots + c_n + (1-c)) > 0, \tag{16}$$

that is, infinite ladder events have a positive probability, then Theorem 3 holds.

By denoting $s_m = c_m + c_{m+1} + \dots$, *and proceeding as in Example 7, the condition above is equivalent to*

$$\sum_{m=1}^{\infty} s_m < \infty.$$

Therefore, we have to assume that the mean value of the spontaneously generated symbol (which also determines the memory length in this example) is finite.

As in Example 7, a condition that is sufficient to ensure that Theorem 4 holds is that there exists $n_0 \in \{1, 2, \dots\}$ *and* $\epsilon > 0$ *such that*

$$s_n \leq \frac{(1-\epsilon)}{n},$$

for all $n \geq n_0$. *This condition does not exclude the possibility of having*

$$\sum_{n\geq 1} s_n = +\infty,$$

and therefore Theorem 4 holds even in some cases in which the mean value of the spontaneously generated symbol (and thus of the associated memory length) is not finite.

Note that in this example, we do not have a regeneration time which is a stopping time. By going forward in time, it is always possible to choose a memory length that goes back as many steps in the past as we want.

Remark 4. *In Example 8, the process uses only the last step to define the size of the memory. Moreover, it decides to copy uniformly one of the past symbols in the range of the memory length. This kernel can be easily generalized so that the presented results hold in a similar way.*

Consider as a generalization the kernel

$$p(g|x_{-\infty}^{-1}) = c_g + (1-c)\sum_{k=1}^{\infty} f_{x_{-1}}(k)\mathbf{1}\{x_{-k} = g\}, \tag{17}$$

for each $x_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}}$. *Here,* $\sum_{k\geq 1} f_{x_{-1}}(k) = 1$. *For* $m \geq 1$, *we have* $\sum_{k=1}^{m} f_m(k) \geq d_m \in (0,1)$, *with* $d_m \to 1$ *as* $m \to \infty$.

This kernel generalizes the one in Example 8 in the following way. The position of the symbol which will be copied is chosen with a distribution which is not uniform. This distribution has support in $\{1, 2, 3, \dots\}$ *but it is concentrated in the most* x_{-1} *recent symbols with high probability, where* x_{-1} *is the first symbol in the past of the process.*

Now, we have

$$\sum_{a_{-n}^{-1} \in \mathcal{A}^n} \alpha(a_{-n})\alpha(a_{-(n-1)}|a_{-n}) \dots \alpha(a_{-1}|a_{-n}^{-2}) \geq c_1(c_1 + c_2 + (1-c)d_1) \dots (c_1 + \dots c_n + (1-c)d_n).$$

As we did in Example 8, we can consider $c_n \to 0$ *and* $d_n \to 1$ *fast enough as* $n \to \infty$ *such that Theorems 3 and 4 hold.*

Example 9. *Consider* $\mathcal{A} = \{1, 2, \dots\}$ *and let* $\{c_g, g \in \mathcal{A}\}$ *be a collection of non negative numbers such that* $c = \sum_{g\in\mathcal{A}} c_g < 1$. *For each* $k \geq 1$, *let* $(q_k(g) : g \in \mathcal{A})$ *be a probability distribution. For any* $x_{-\infty}^{-1} \in \mathcal{H} = \mathcal{A}^{\mathbb{N}}$ *and for any* $g \in \mathcal{A}$, *let*

$$p(g|x_{-\infty}^{-1}) = c_g + (1-c)q_{Y(x_{-\infty}^{-1})}(g), \tag{18}$$

where

$$Y(x_{-\infty}^{-1}) = \inf\left\{ m \geq 1 : \sum_{n=1}^{m} x_{-n} \leq \frac{m(m+1)}{2} \right\}.$$

Recall that $\sum_{n=1}^{m} n = \frac{m(m+1)}{2}$.

In this example, the state of the process at a time is either chosen independently of the past with probability c *or it depends on the past symbols with probability* $1-c$. *In the first case, symbol* $g \in \mathcal{A}$ *is chosen with probability* c_g/c. *In the second case, we choose the new state using a probability distribution which depends only on* $Y(x_{-\infty}^{-1})$. *Note that for any* $k \geq 1$ *and for any* $x_{-\infty}^{-1}$, *to check if* $Y(x_{-\infty}^{-1}) = k$ *we need to check only* x_{-k}^{-1}, *i.e., the first* k *elements of* $x_{-\infty}^{-1}$.

Let us suppose that $c_1 > 0$ *and that for each* $k \geq 1$,

$$\min\left\{\sum_{n=1}^{k+1} q_m(n) : m \leq k\right\} \geq d_k,$$

with $d_k \to 1$ *as* $k \to \infty$. *This implies that for any* $k \geq 1$ *and* $x_{-\infty}^{-1}$ *satisfying* $\sum_{n=1}^{k} x_{-n} \leq \frac{k(k+1)}{2}$, *we have that*

$$\sum_{g \leq k+1} p(g|x_{-\infty}^{-1}) \geq c_1 + \ldots + c_{k+1} + (1-c)\min\left\{\sum_{n=1}^{k+1} q_m(n) : m \leq k\right\} \geq c_1 + \ldots + c_{k+1} + (1-c)d_k.$$

As a consequence,

$$\sum_{a_{-n}^{-1} \in \mathcal{A}^n} \alpha(a_{-n})\alpha(a_{-(n-1)}|a_{-n}) \ldots \alpha(a_{-1}|a_{-n}^{-2}) \geq$$

$$\sum_{a_{-n}^{-1} : a_{-n+k-1} \leq k, k=1,\ldots,n} \alpha(a_{-n})\alpha(a_{-(n-1)}|a_{-n}) \ldots \alpha(a_{-1}|a_{-n}^{-2}) \geq$$

$$c_1(c_1 + c_2 + (1-c)d_1) \ldots (c_1 + \ldots + c_n + (1-c)d_{n-1}).$$

Once more, we can find $c_n \to 0$ *and* $d_n \to 1$ *fast enough as* $n \to \infty$ *such that the conditions of Theorems 3 and 4 hold.*

As in the preceding example, the first symbol in the past is necessary to obtain the memory length. If the actual symbol is not spontaneous, nor is the past symbol, there is no possibility to decide about the actual symbol by only looking to the spontaneous symbols of the past.

For the same reasons as in Example 8, we do not have a regeneration time which is a stopping time.

Example 10. *Consider a collection of non-negative numbers* $\{\theta_j : j \geq 0\}$ *such that* $\sum_{j=0}^{\infty} \theta_j = 1$. *Suppose that* $\theta_0 > 0$. *Consider a model taking values in* $\mathcal{A} = \{0, 1, 2, 3\}$ *with the convention* $3 + 1 = 0$ *and* $0 - 1 = 3$. *Let*

$$\mathcal{H} = \{a_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}} : a_{-n+1} \in \{a_{-n} - 1, a_{-n}, a_{-n} + 1\}, \text{ for all } n \geq 2\}.$$

For any $w_{-\infty}^{-1} \in \mathcal{H}$, *the probability kernel is given as follows. For any* $w_{-\infty}^{-1} \in \mathcal{H}$ *and for* $g = w_{-1}$,

$$p(g|w_{-\infty}^{-1}) = \frac{1}{3}\theta_0 + \sum_{j=1}^{\infty} \theta_j \mathbf{1}_{\{w_{-j}, w_{-j}+2\}}(g).$$

For $g \in \{w_{-1} - 1, w_{-1} + 1\}$

$$p(g|w_{-\infty}^{-1}) = \frac{1}{3}\theta_0 + \sum_{j=1}^{\infty} \theta_j \mathbf{1}_{\{w_{-j}\}}(g).$$

Finally, we define $p(g|w_{-\infty}^{-1}) = 0$ *for* $g = w_{-1} + 2$. *By construction, the process can only stay on the same symbol or jump to one of the neighbors symbols.*

Observe that for $g, b \in \mathcal{A}$ *such that* $b \neq g + 2$,

$$\alpha(b|g) = \frac{\theta_0}{3}\mathbf{1}_{\{g-1,g,g+1\}}(b) + \theta_1 \mathbf{1}_{\{g\}}(b).$$

Recalling Definition 4, we have that

$$\beta_1 = \theta_0 + \theta_1$$

and for any $g, b \in \mathcal{A}$ such that $b \neq g+2$,

$$p^{[1]}_{Markov}(b|g) = \frac{1}{3}\frac{\theta_0}{\theta_0+\theta_1}\mathbf{1}_{\{g-1,g,g+1\}}(b) + \frac{\theta_1}{\theta_0+\theta_1}\mathbf{1}_{\{g\}}(b).$$

This implies that Assumption 1 holds with $\hat{n}=1$. Finally, note that $n_0 = 2$.

For this kernel, we have $\beta = 0$, and for any $w_{-n}^{-1} \in \mathcal{A}^n, n \geq 1$, $\beta(w_{-n}^{-1}) = \sum_{j=0}^{n}\theta_j$. This implies that for $n \geq 1$, by denoting $s_n = \theta_n + \theta_{n+1} + \ldots$, we have that

$$\tilde{\rho_n} = \prod_{j=1}^{n+1}(1-s_j).$$

By recalling Example 7, we have that Theorem 5 holds if we suppose that

$$\sum_{m=1}^{\infty} s_m < \infty,$$

and Theorem 6 holds if we suppose that there exists $m \in \{1,2,\ldots\}$ and $\epsilon > 0$ such that

$$s_n \leq \frac{(1-\epsilon)}{n}$$

for all $n \geq m$.

Remark 5. *We consider the following generalization of the above example. Consider a connected and non-directed graph (V,E), where V is a finite set of vertices and E is the set of edges. For any $v \in V$, let $E(v)$ be the set of vertices connected to v by an edge. We consider the convention $v \in E(v)$.*

Consider moreover a collection of non-negative numbers $\{\theta_j : j \geq 0\}$ such that $\sum_{j=0}^{\infty}\theta_j = 1$. Suppose that $\theta_0 > 0$. Consider a model taking values in $\mathcal{A} = V$ and let

$$\mathcal{H} = \{a_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}} : a_{-n+1} \in E(a_{-n}),\ \textit{for all } n \geq 2\}.$$

For any $x_{-\infty}^{-1} \in \mathcal{H}$, the probability kernel is given as follows. For any $w_{-\infty}^{-1} \in \mathcal{H}$ and for $g = w_{-1}$,

$$p(g|w_{-\infty}^{-1}) = \frac{1}{|E(w_{-1})|}\theta_0 + \sum_{j=1}^{\infty}\theta_j \mathbf{1}_{E(w_{-1})^c \cup \{w_{-1}\}}(w_{-j}).$$

For $g \in E(v) \setminus \{w_{-1}\}$,

$$p(g|w_{-\infty}^{-1}) = \frac{1}{|E(w_{-1})|}\theta_0 + \sum_{j=1}^{\infty}\theta_j \mathbf{1}_{\{w_{-j}\}}(g).$$

Finally, we define $p(g|w_{-\infty}^{-1}) = 0$ for $g \notin E(w_{-1})$.

This kernel represents a process that can only jump to its neighbors on the graph (V,E). However, the probability of jumping to each neighbor depends on the entire history of the process in a similar way to Example 10. With assumptions similar to those made in Example 10, we can use Algorithm 2 for this kernel with $\hat{n}=1$.

4 Proof of Theorems 1 and 2

In the following $(U_n)_{n\in\mathbb{Z}}$ are i.i.d. uniform random variables in $[0,1)$. We will construct a function F such that $X_n = F(U_n, U_{n-1}, \ldots)$ in such a way that $U_{n-1}, U_{n-2}, \ldots$ defines a partition of the interval $[0,1)$, associating each part of the interval to an element of $\mathcal{A}$, and X_n will be the defined taking into account to which of these intervals U_n belongs.

To define the coupling function $F(u_0, u_{-1}, \ldots)$ for $u_0, u_{-1}, \ldots \in [0,1)$ we need to define the partition of the interval $[0,1)$ cited above. The intervals belonging to this partition are denoted

$$I_0(g),\ \text{for } g \in \mathcal{A}$$

and

$$I_m(g, u_{-1}, u_{-2}, \ldots, u_{-m}) = I_m(g, u_{-m}^{-1}),\ \text{for } m \geq 1 \text{ and } g \in \mathcal{A}.$$

These intervals are used to define the coupling function in the following way

$$F(u_{-\infty}^0) = \sum_{g\in\mathcal{A}} g\mathbf{1}\left\{u_0 \in \bigcup_{m=0}^{\infty} I_m(g, u_{-m}^{-1})\right\} + *\mathbf{1}\left\{u_0 \in [0,1) \setminus \bigcup_{m=0}^{\infty} I_m(g, u_{-m}^{-1})\right\}. \tag{19}$$

Here and in the following we use the convention $I_0(b, u_0, u_{-1}) = I_0(b)$.

Algorithm 1 uses the notion of "temporary" values for X_m, $m \in \mathbb{Z}$. In the perfect simulation algorithm we check if X_m can be obtained only by using the values of $U_m, \ldots, U_{m-n}$, for $n \geq 0$. This is formalized by the following notation. For any $n \geq 0$,

$$F^{(n)}(u_{-n}^0) := \sum_{g\in\mathcal{A}} g\mathbf{1}\left\{u_0 \in \bigcup_{m=0}^{n} I_m(g, u_{-m}^{-1})\right\} + *\mathbf{1}\left\{u_0 \in [0,1) \setminus \bigcup_{m=0}^{n} I_m(g, u_{-m}^{-1})\right\}. \tag{20}$$

Note that for any $u_{-\infty}^0 \in [0,1)^{\mathbb{N}}$, $F(u_{-\infty}^0) = \lim_{n\to\infty} F^{(n)}(u_{-n}^0)$.

The intervals $(I_n(g, u_{-n}^{-1}), n \geq 0, g \in \mathcal{A})$ will be defined by means of a collection of positive numbers

$$J_n(g, u_{-n}^{-1}) : \mathcal{A} \times [0,1]^n \to [0,1]$$

taking values in $[0,1]$ in the following way: for any $n \geq 0$ and $g \in \mathcal{A}$,

$$I_n(g, u_{-n}^{-1}) = \left[\sum_{m=0}^{n-1}\sum_{b\in\mathcal{A}} J_m(b, u_{-m}^{-1}) + \sum_{b<g} J_n(b, u_{-n}^{-1}), \sum_{m=0}^{n-1}\sum_{b\in\mathcal{A}} J_m(b, u_{-m}^{-1}) + \sum_{b\leq g} J_n(b, u_{-n}^{-1})\right). \tag{21}$$

Note that $|I_n(g, u_{-n}^{-1})| = J_n(g, u_{-n}^{-1})$. Note also that for the case $n = 0$, the definition above reads

$$I_0(g) = \left[\sum_{b<g} J_0(b), \sum_{b\leq g} J_0(b)\right).$$

Before giving the definition of $(J_n(g, u_{-n}^{-1}), n \geq 0, g \in \mathcal{A})$, let us stress that since all $J_n(g, u_{-n}^{-1})$ are non negative (as will be proved in Proposition 6), in (19), for any given sequence $u_0, u_{-1}, \ldots$, there exist at most one index $m \geq 0$ and one value g such that the condition $u_0 \in I_m(g, u_{-m}^{-1})$ is fulfilled. Therefore, $I_n(g, u_{-n}^{-1})$ is a right-open sub-interval of $[0,1)$ (including the empty set) for any $g \in \mathcal{A}$.

In what follows, for any $j \in \mathbb{Z}$, we shall also consider shifted versions $F^{(n)}(u_{j-n}^j)$ of $F^{(n)}(u_{-n}^0)$, $I_n(g, u_{j-n}^{j-1})$ of $I_n(g, u_{-n}^{-1})$ and $J_n(g, u_{j-n}^{j-1})$ of $J_n(g, u_{-n}^{-1})$, which are obtained by applying the respective definitions of $F^{(n)}, I_n$ and J_n to u_{j-n}^j rather than to u_{-n}^0.

Let us now give the definition of the successive values of $J_n(g, U_{-n}^{-1})$. We take $J_0(g) := \alpha(g)$ and define recursively for any $n \geq 1$,

$$J_n(g, u_{-n}^{-1}) := \alpha(g|F^{(n-1)}(u_{-n}^{-1})F^{(n-2)}(u_{-n}^{-2})\ldots F^{(0)}(u_{-n})) - \alpha(g|F^{(n-2)}(u_{-(n-1)}^{-1})F^{(n-3)}(u_{-(n-1)}^{-2})\ldots F^{(0)}(u_{-(n-1)})). \tag{22}$$

Note that by construction,

$$J_n(g, u_{-n}^{-1}) = \alpha(g|F^{(n-1)}(u_{-n}^{-1})F^{(n-2)}(u_{-n}^{-2})\ldots F^{(0)}(u_{-n})) - \sum_{m=0}^{n-1} J_m(g, u_{-m}^{-1}).$$

Remark 6. *Recalling Algorithm 1, the term defined in* (22) *satisfies for any $g \in \mathcal{A}$ and for any $n \geq 1$,*

$$J_n(g, U_{-n}^{-1}) = \alpha(g|X_{-1}^{(-n)}\ldots X_{-n}^{(-n)}) - \alpha(g|X_{-1}^{-(n-1)}\ldots X_{-(n-1)}^{-(n-1)}).$$

The positive random variable $J_n(g, U_{-n}^{-1})$ quantifies how much the minimal probability of having the symbol g at time 0 increases when we consider the temporary values $X_{-1}^{(-n)}, \ldots, X_{-n}^{(-n)}$, which depend on the random variables U_{-n}^{-1}, instead of considering $X_{-1}^{-(n-1)}, \ldots, X_{-(n-1)}^{-(n-1)}$, which depend on the random variables $U_{-(n-1)}^{-1}$. In other words, it quantifies how much the minimal probability of observing the symbol g at time 0 increases when we additionally consider the uniform random variable U_{-n}.

Remark 7. *The sequential construction of the functions $F^{(n)}(u^0_{-n})$ given in (20) above is well defined. Indeed, to obtain $F^{(n)}(u^0_{-n})$, one only needs to know all intervals $I_m(g, u^{-1}_{-m}), m \leq n$. And in turn, to define these intervals through their respective lengths in (21) and (22), we see that we only rely on previously chosen functions $F^{(n-1)}, \dots, F^{(0)}$, which are applied to substrings of u^{-1}_{-n}.*

The following proposition is a direct consequence of the definitions above.

Proposition 6. *Let $u^0_{-\infty} \in [0,1)^{+\infty}$. Suppose there exists $n \geq 0$ such that $F^{(n)}(u^0_{-n}) = b$ for some $b \in \mathcal{A}$. Then $F^{(n+k)}(u^0_{-(n+k)}) = b$ for all $k \geq 1$. In particular, $J_n(g) \geq 0$ for all $n \geq 0$ and $g \in \mathcal{A}$.*

Proof. For the first assertion, just note that for any $k \geq 1$,

$$\{F^{(n)}(u^0_{-n}) = b\} = \left\{ u_0 \in \bigcup_{j=0}^{n} I_j(b, u^{-1}_{-j}) \right\} \subset \left\{ u_0 \in \bigcup_{j=0}^{n+k} I_j(b, u^{-1}_{-j}) \right\} = \{F^{(n+k)}(u^0_{-(n+k)}) = b\}.$$

The second assertion follows from (22) together with the monotonicity properties stated in Remark 2 above. □

Now we can prove Theorem 1.

Proof. The proof follows directly by the construction above by noticing that for any $m \in \mathbb{Z}$ and for any $k \geq 0$,

$$X_m^{(m-k)} = F^{(k)}(U^m_{m-k})$$

and that

$$X_m = \lim_{n\to\infty} F^{(n)}(U^m_{m-n}) = F^{(T[m])}(U^m_{T[m]}).$$

From this the assertion follows. □

The proof of Theorem 2 is analogous to the proof of Theorem 1 and therefore omitted. The construction of the functions $\tilde{F}^{(j)} : [0,1)^{\mathbb{N}} \to \mathcal{A}$, for $j = 0, \dots, n_0 - 1$, are analogous to the construction of the function F above. Having constructed these functions, the proof of Theorem 2 follows as the the proof of Theorem 1 above.

5 Proof of Theorems 3, 4, 5 and 6

Recall that according to Definition 2, for $k \geq 0$, $X_0^{(-k)}, \dots, X_{-k}^{(-k)} \in \mathcal{A}_*$ are the values of $X_0, \dots, X_{-k}$ after the first $k+1$ iterations of Algorithm 1, that is, based on the simulation of U^0_{-k}. For any $n \geq 0$,

$$\begin{aligned}\mathbb{P}(|T[0]| > n) = \\ \sum_{a^{-1}_{-n} \in \mathcal{A}^n_*} \mathbb{P}(X^{(-n)}_{-n} = a_{-n}) \dots \mathbb{P}(X^{(-n)}_{-1} = a_{-1} | (X^{(-n)})^{-2}_{-n} = a^{-2}_{-n}) \mathbb{P}(X^{(-n)}_0 = * | (X^{(-n)})^{-1}_{-n} = a^{-1}_{-n}) \\ = \sum_{a^{-1}_{-n} \in \mathcal{A}^n_*} \alpha(a_{-n}) \alpha(a_{-(n-1)} | a_{-n}) \dots \alpha(a_{-1} | a^{-2}_{-n}) \alpha(* | a^{-1}_{-n}). \quad (23)\end{aligned}$$

Note that in the case $n = 0$ this reads

$$\mathbb{P}(|T[0]| > 0) = \alpha(*).$$

In particular,

$$\mathbb{E}(|T[0]|) = \sum_{n=0}^{+\infty} \sum_{a^{-1}_{-n} \in \mathcal{A}^n_*} \alpha(a_{-n}) \alpha(a_{-(n-1)} | a_{-n}) \dots \alpha(a_{-1} | a^{-2}_{-n}) \alpha(* | a^{-1}_{-n}).$$

We interpret the above formula in terms of an auxiliary chain $(Y_n)_{n\geq 0}$, assuming values in $\mathcal{A}_*$ and having increasing memory, which we introduce now. We start from $\mathbb{P}(Y_0 = g) = \alpha(g)$ and then sequentially define

$$\mathbb{P}(Y_n = g | Y_0^{n-1} = a_0^{n-1}) = \alpha(g|a_0^{n-1}),$$

for any $g \in \mathcal{A}_*$ and $a_0^{n-1} \in \mathcal{A}_*^n$.

Remark 8. *The purpose of introducing this auxiliary chain is to compare the probability of events related to $(Y_n)_{n\geq 0}$ with the probability of events related to $(T[n] : n \in \mathbb{Z})$.*

Note that for any $k \in \mathbb{Z}$ and $n \in \mathbb{N}$,

$$Y_n \sim X_k^{(k-n)}.$$

In particular,

$$Y_n \sim X_n^{(0)}, \text{ for any } n \in \mathbb{N}.$$

Therefore, $(Y_n)_{n\geq 0}$ has the law of a 'forward in time' version of Algorithm 1, which instead of starting from a certain time and going backward in time, updating the values of the chain for each new information obtained, goes forward in time checking which values of the chain can be obtained from a certain time on.

We have that

$$p_n := \mathbb{P}(|T[0]| > n) = \mathbb{P}(Y_n = *) \tag{24}$$

and

$$\mathbb{E}(|T[0]|) = \sum_{n\geq 0} \mathbb{P}(Y_n = *) = \mathbb{E}\left(\sum_{n\geq 0} \mathbf{1}\{Y_n = *\}\right). \tag{25}$$

Moreover,

$$\mathbb{P}(|T[0,n]| > m) \leq \sum_{j=0}^{n} \mathbb{P}(|T[j]| > m) = \sum_{j=0}^{n} \mathbb{P}(|T[0]| > m + j).$$

Therefore,

$$\mathbb{P}(|T[0,n]| > m) \leq \sum_{k=m}^{m+n} \mathbb{P}(Y_k = *) \tag{26}$$

and

$$\mathbb{E}(|T[0,n]|) \leq \sum_{m\geq 0} \sum_{k=m}^{m+n} \mathbb{P}(Y_k = *) \leq (n+1)\mathbb{E}\left(\sum_{n\geq 0} \mathbf{1}\{Y_n = *\}\right). \tag{27}$$

Now we can prove Theorem 3.

Proof. Let

$$\eta_1 = \inf\{m \geq 0 : Y_m = *\}$$

with the convention $\inf \emptyset = \infty$, and for $n \geq 1$, let

$$\eta_{n+1} = \inf\{m \geq \eta_n + 1 : Y_m = *\},$$

with the convention $\eta_{n+1} = \eta_n$ in the case $\eta_n = \infty$. Notice that

$$\mathbb{P}(\eta_1 \geq n) = \mathbb{P}\left(\bigcap_{k=0}^{n-1} \bigcup_{a_k \in \mathcal{A}} \{Y_k = a_k\}\right) = \sum_{a_0^{n-1} \in \mathcal{A}^n} \alpha(a_0)\alpha(a_1|a_0)\dots\alpha(a_{n-1}|a_0^{n-2}) = \rho_n. \tag{28}$$

By assumption,

$$\mathbb{P}(\eta_1 < \infty) = 1 - \lim_{n\to\infty} \mathbb{P}(\eta_1 \geq n) = 1 - \lim_{n\to\infty} \rho_n \leq 1 - c.$$

Now we introduce

$$\Omega_n := \bigcup_{j=n-1}^{\infty} \left\{ w_1^j \in A_*^j : w_j = *, \sum_{k=1}^{j-1} \mathbf{1}\{w_k = *\} = n - 2 \right\},$$

the set of all possible past sequences corresponding to the event $\bigcap_{m=1}^{n-1}\{\eta_m<\infty\}$. Then,

$$\mathbb{P}\left(\eta_n<\infty\Big|\bigcap_{m=1}^{n-1}\{\eta_m<\infty\}\right)=$$
$$\sum_{w\in\Omega_n}\mathbb{P}(w)\left[1-\lim_{n\to\infty}\sum_{a_0^{n-1}\in\mathcal{A}^n}\alpha(a_0|w)\alpha(a_1|a_0w)\dots\alpha(a_{n-1}|a_0^{n-2}w)\right].$$

By monotonicity (recall Remark 2), we have that the term above is upper bounded by

$$\sum_{w}\mathbb{P}(w)\left[1-\lim_{n\to\infty}\sum_{a_0^{n-1}\in\mathcal{A}^n}\alpha(a_0)\alpha(a_1|a_0)\dots\alpha(a_{n-1}|a_0^{n-2})\right]\le 1-c.$$

This implies that

$$\mathbb{P}(\eta_n<\infty)=\mathbb{P}(\eta_j<\infty, j=1,\dots,n)\le(1-c)^n.$$

To conclude part 1 of Theorem 3 note that, since by (25), $\mathbb{E}(|T[0]|)$ is the expected number of visits of the auxiliary chain to $*$, we have that

$$\mathbb{E}(|T[0]|)=\mathbb{E}\left(\sum_{n\ge0}\mathbf{1}\{Y_n=*\}\right)=\sum_{m\ge1}\mathbb{E}\left(\mathbf{1}\{\eta_m<\infty\}\right)\le\sum_{m\ge1}(1-c)^m=\frac{1-c}{c}.$$

The proof of part 2 of Theorem 3 follows directly from (27). To prove part 3 of Theorem 3, note that for any $n\ge0$,

$$\mathbb{P}(|T[0,+\infty)|>n)=$$
$$\sum_{a_{-n}^{-1}\in\mathcal{A}_*^n}\mathbb{P}(X_{-n}^{(-n)}=a_{-n})\cdots\mathbb{P}(X_{-1}^{(-n)}=a_{-1}|(X^{(-n)})_{-n}^{-2}=a_{-n}^{-2})\times$$
$$\times\,\mathbb{P}\left(\bigcup_{k=0}^{\infty}\{X_k^{(-n)}=*\}|(X^{(-n)})_{-n}^{-1}=a_{-n}^{-1}\right).$$

By noticing that

$$\mathbb{P}\left(\bigcup_{k=0}^{\infty}\{X_k^{(-n)}=*\}|(X^{(-n)})_{-n}^{-1}=a_{-n}^{-1}\right)=$$
$$\sum_{k=0}^{\infty}\mathbb{P}\left(\{X_k^{(-n)}=*, X_j^{(-n)}\neq *, j=0,\dots,k-1\}|(X^{(-n)})_{-n}^{-1}=a_{-n}^{-1}\right),$$

we conclude that

$$\mathbb{P}(|T[0,+\infty)|>n)=$$
$$\sum_{a_{-n}^{-1}\in\mathcal{A}_*^n}\sum_{k=0}^{\infty}\sum_{b_0^{k-1}\in\mathcal{A}^k}\alpha(a_{-n})\alpha(a_{-(n-1)}|a_{-n})\cdots\alpha(a_{-1}|a_{-n}^{-2})\alpha(b_0|a_{-n}^{-1})$$
$$\cdots\alpha(b_{k-1}|b_0^{k-2}a_{-n}^{-1})\alpha(*|b_0^{k-1}a_{-n}^{-1}).\quad(29)$$

Let

$$\bar\eta=\sup\{m\ge0: Y_m=*\},$$

with the convention $\sup\emptyset=-\infty$. Interpreting (29) in terms of the auxiliary chain, we see that

$$\mathbb{P}(|T[0,+\infty)|>n)=\mathbb{P}(\bar\eta\ge n).$$

Now, note that

$$\{\bar\eta>n,\ \text{for all } n=1,2,\dots\}=\{\eta_j<\infty,\ \text{for all } j=1,2,\dots\},$$

such that

$$\lim_{n\to\infty}\mathbb{P}(|T[0,+\infty]|>n)=\lim_{n\to\infty}\mathbb{P}(\bar\eta>n)=\lim_{n\to\infty}\mathbb{P}(\eta_n<\infty)=0.$$

□

Now we prove Theorem 4.

Proof. Let us consider a representation of the process $(Y_n)_{n=0,1,\dots}$ using a sequence of i.i.d. uniform random variables $(V_n)_{n=1,2,\dots}$ assuming values in $[0,1)$. Let

$$Y_n = \begin{cases} g & \text{, if } V_n \in \left[\sum_{b<g} \alpha(b|Y_0^{n-1}), \sum_{b\leq g} \alpha(b|Y_0^{n-1})\right), \text{ for } g \in \mathcal{A}, \\ * & \text{, in the contrary case.} \end{cases}$$

Now, we define

$$\tilde{Y}_n = \begin{cases} g & \text{, if } V_n \in \left[\sum_{b<g} \alpha(b|Y_0^{n-1}), \sum_{b<g} \alpha(b|Y_0^{n-1}) + \alpha(g|Y_{L_{n-1}+1}^{n-1})\right), \text{ for } g \in \mathcal{A}, \\ * & \text{, in the contrary case,} \end{cases}$$

with $L_{n-1} = \sup\{m = 0, \dots, n-1 : \tilde{Y}_m = *\}$. Here, we use the convention $\sup \emptyset = 0$. Note that, by Remark 2, we have that $\alpha(g|Y_{L_{n-1}+1}^{n-1}) \leq \alpha(g|Y_0^{n-1})$, for any $Y_0^{n-1} \in \mathcal{A}_*^n$, and therefore the process $(\tilde{Y}_n)_{n=0,1,\dots}$ is well defined. Moreover, for all n, if $\tilde{Y}_n = g$, then also $Y_n = g$. So we have that $\tilde{Y}_n \in \{Y_n, *\}$, for any $n \geq 0$, and the two processes coincide during excursions of $\tilde{Y}$ out of $*$. Therefore, for any $m \geq 0$,

$$\sum_{n=0}^{m} \mathbf{1}\{Y_n = *\} \leq \sum_{n=0}^{m} \mathbf{1}\{\tilde{Y}_n = *\}.$$

Moreover,

$$p_m = \mathbb{P}(Y_m = *) \leq \mathbb{P}(\tilde{Y}_m = *). \tag{30}$$

Now, considering $\tau_0 := 0$ and $\tau_n := \inf\{m > \tau_{n-1} : \tilde{Y}_m = *\}$, we have that $(\tau_m)_{m\geq 1}$ are the marks of a renewal process, and the excursions $(\tilde{Y}_{\tau_n+k})_{0\leq k<\tau_{n+1}}, n \geq 1$, are i.i.d. In particular, standard renewal arguments imply that we have almost sure convergence

$$\frac{1}{n}\sum_{k=1}^{n} \mathbf{1}\{\tilde{Y}_k = *\} \to \frac{1}{\mathbb{E}(\tau_1)},$$

and thus, by dominated convergence, also

$$\frac{1}{n}\sum_{k=1}^{n} \mathbb{P}(\tilde{Y}_k = *) \to \frac{1}{\mathbb{E}(\tau_1)}.$$

But, by assumption,

$$\mathbb{E}(\tau_1) = \sum_{m=1}^{+\infty} \mathbb{P}(\tau_1 \geq m) = \sum_{m=1}^{+\infty} \rho_m = +\infty,$$

such that

$$\frac{1}{n}\sum_{k=1}^{n} \mathbb{P}(\tilde{Y}_k = *) \to 0.$$

We now show that this implies necessarily that

$$\lim_{n\to\infty} \mathbb{P}(Y_n = *) = \lim_n p_n = 0.$$

Indeed, by (24), the sequence $(p_n)_n$ is a non-increasing sequence of positive numbers which thus converges. Suppose that $\lim_n p_n > 0$. Then $\liminf_n \mathbb{P}(\tilde{Y}_n = *) > 0$ as well and then there exist $c_1, n_1 > 0$ such that for all $n \geq n_1$, $\mathbb{P}(\tilde{Y}_n = *) \geq c_1 > 0$. This is in contradiction with $\frac{1}{n}\sum_{k=1}^{n} \mathbb{P}(\tilde{Y}_k = *) \to 0$.

Together with (30) and (26), this concludes the proof. □

The proofs of Theorems 5 and 6 are analogous to the proofs of Theorems 3 and 4, and in the following we just sketch the main ideas.

Proof. Let $(V_n^{a_{-\hat{n}}^{-1}} : n \geq 0, a_{-\hat{n}}^{-1} \in \mathcal{C})$ be i.i.d. uniform random variables in $[0,1)$. For any $n \geq 0$ and $a_{-\hat{n}}^{-1} \in \mathcal{C}$, the process $(Z_n^{a_{-\hat{n}}^{-1}})_{n\geq 0}$ is defined as follows. In the following, let us consider the convention $\inf \emptyset = *$. For $k \geq 0$ and $n = kn_0, \ldots, kn_0 - 1$, we have

$$Z_n^{a_{-\hat{n}}^{-1}} = \inf \left\{ g \in \mathcal{A} : V_n^{a_{-\hat{n}}^{-1}} < \sum_{b \leq g} \alpha(g | (Z^{a_{-\hat{n}}^{-1}})_{kn_0}^{n-1} a_{-\hat{n}}^{-1}) \right\}.$$

Now, we define the process $(Z_n)_{n\geq 0}$ as follows. For $n = kn_0, \ldots, kn_0 - 1$, we have

$$Z_n = \begin{cases} b, & \text{if } Z_n^{a_{-\hat{n}}^{-1}} = b, \text{ for all } a_{-\hat{n}}^{-1} \in \mathcal{C} \text{ and some } b \in \mathcal{A}, \\ *, & \text{in the contrary case.} \end{cases}$$

For $k \geq 1$ and $n = kn_0, \ldots, kn_0 - 1$, we have two cases. If $Z_{kn_0-\hat{n}}^{kn_0-1} = b_{-\hat{n}}^{-1}$, for some $b_{-\hat{n}}^{-1} \in \mathcal{C}$, we have

$$Z_n = \inf \left\{ g \in \mathcal{A} : V_n^{b_{-\hat{n}}^{-1}} < \sum_{b \leq g} \alpha(g | Z_0^{n-1}) \right\}.$$

In the contrary case, we have

$$Z_n = \begin{cases} b, & \text{if } Z_n^{a_{-\hat{n}}^{-1}} = b, \text{ for all } a_{-\hat{n}}^{-1} \in \mathcal{C} \text{ and some } b \in \mathcal{A}, \\ *, & \text{in the contrary case.} \end{cases}$$

We have that

$$\mathbb{P}(|\tilde{T}[0]| > n) = \mathbb{P}(Y_{nn_0} = *)$$

and

$$\mathbb{E}(|T[0]|) = \sum_{n\geq 0} \mathbb{P}(Y_{nn_0} = *) = \mathbb{E}\left(\sum_{n \geq 0} \mathbf{1}\{Y_{nn_0} = *\} \right).$$

By proceeding in a similar way as in the proofs Theorems 3 and 4, the assertions of Theorems 5 and 6 follow. □

6 Discussion

In this section we will compare our results to algorithms and results presented in other articles. In particular we will show that the results presented in these articles cannot be applied to the examples presented in Section 3. Finally we will also discuss some limitations of our approach.

6.1 Comparing to other results

Our framework is similar to the one presented in De Santis and Piccioni (2012), and the present article is very much inspired by this work. For a probability kernel p, De Santis and Piccioni (2012) consider a coupling function $F : [0,1] \times \mathcal{H} \to \mathcal{A}$ such that

$$F(u_0, x_{-1}, x_{-2}, \ldots) = \sum_{g \in \mathcal{A}} g \mathbf{1} \left\{ u_0 \in \bigcup_{m=0}^{-\infty} I_m(g, x_{-1}, \ldots, x_m) \right\},$$

where $\mathcal{H} \subset \mathcal{A}^{\mathbb{N}}$ is the set of admissible histories. This means that each possible past defines a partition of the interval $[0,1]$. The authors study the backward coalescence time $\tau_0(U_{-\infty}^0)$: $-\tau_0$ is a stopping time, and X_0 can be defined from $U_0, U_{-1}, \ldots, U_{\tau_0}$. The existence of such a backward coalescence time and its finiteness directly induce a perfect simulation algorithm.

Theorem 1 of De Santis and Piccioni (2012) deals with perfect simulation algorithms having backward coalescence times that are based on *information depths*. A necessary condition for this theorem to work is that almost surely

$$\lim_{h \to \infty} A_h(U_{-h}^{-1}) = 1,$$

where the information depth $A_h(U_{-h}^{-1})$ is given by

$$A_h(U_{-h}^{-1}) = \inf\{\beta(w_{-h}^{-1}) : w_{-\infty}^{-1} \in \mathcal{H} : w_{-k} = g \text{ if } U_{-k} \in I_0(g), k \leq h\}.$$

Let us compare this approach with our Example 8. Notice that in this example,

$$\beta(*a_{-k}^{-1}) = c,$$

for any $a_{-k}^{-1} \in \mathcal{A}^k$. Therefore, $A_h(U_{-h}^{-1}) = c$, whenever $U_{-1} > c$. So $\lim_{h\to\infty} \mathbb{P}(A_h(U_{-h}^{-1}) < 1) \geq 1 - c$, such that Theorem 1 of De Santis and Piccioni (2012) cannot be applied in the situation of this example.

In the framework of Remark 4, assuming additionally that

$$\inf_m f_m(k) = 0,$$

for each $k \geq 1$, then

$$\beta(*a_{-k}^{-1}) = c,$$

for any $a_{-k}^{-1} \in \mathcal{A}^k$. Again Theorem 1 of De Santis and Piccioni (2012) cannot be applied here.

Finally, in Example 9, by supposing that for any $k \geq 1$,

$$\lim_{j\to\infty} \inf\{q_m(k) : m > j\} = 0,$$

we have again that

$$\beta(*a_{-k}^{-1}) \leq c + (1-c)\sum_{k=1}^{\infty} \inf_{m \geq \sum_{j=1}^{k} a_{-j}} q_m(k) = c,$$

for any $a_{-k}^{-1} \in \mathcal{A}^k$.

Gallo (2011) presents a perfect simulation algorithm for chains with memory of variable length. Considering a countable alphabet A and a tree $\tau \subset \{a_{-n}^0 : a_{-n}^0 \in A^n, n \in \{0, 1, \ldots\} \cup \{\infty\}\}$, they propose a perfect simulation algorithm to obtain a sample of the stationary chain compatible with the probabilistic context tree (τ, p), where $p = (p(g|w) : g \in \mathcal{A}, w \in \tau)$. Note that probabilistic context trees are special cases of chains with memory of infinite length.

Gallo (2011) constructs a coupling function $F : [0, 1] \times \tau \to [0, 1]$. In a framework similar to the one presented in Comets et al. (2002) and De Santis and Piccioni (2012), each context defines a partition in the interval $[0, 1]$. A perfect simulation algorithm and a visible regeneration scheme are obtained imposing conditions on $(l(n) : n \in \mathbb{N})$. Examples 8 and 9 considered in this article are in fact chains with memory of variable length however, the results presented by Gallo (2011) can not be applied to these examples.

The perfect simulation algorithm introduced by Garivier (2015) uses the *canonical coupling function* to propose a construction that follows the idea of the original coupling from the past algorithm in Propp and Wilson (1996). At each step of the algorithm, after sampling the uniform random variable that is used to obtain the value of the chain at a certain instant, the algorithm checks all possible values of the chain at that instant for all the possible past infinite strings. Garivier (2015) present kernels in which this algorithm can be applied even in the case $\beta = 0$. This article focuses on algorithmic aspects and does not provide a deep study on the conditions guaranteeing that the algorithm stops almost surely in a finite time.

The perfect simulation algorithm introduced by Gallo and Garcia (2013) uses the *canonical coupling function* to propose a construction that deals with chains with memory of infinite length which are locally continuous with respect to a context tree. In particular, this perfect simulation algorithm assumes $\beta > 0$. Our construction is obviously different from this one since it needs a different coupling function than the canonical one. The comparison between our results in the cases $\beta > 0$ and the results presented by Gallo and Garcia (2013) is difficult.

Gallo and Takahashi (2014) consider regular probability kernels p on a finite alphabet A. This means that p is strongly non-null ($\inf_g \alpha(g) > 0$) and continuous. Moreover, the considered kernel is attractive, which means that p satisfies some monotonicity properties. Note that our results and examples do not require any of the assumption above.

Gallo and Takahashi (2014) also present a concentration of measure result at exponential rate that holds for any process assuming values in a finite alphabet admitting a coupling from the past algorithm. We can use this result in our framework (see Proposition 5). With the additional assumption that the kernel is regular, Theorem 4 of Gallo and Takahashi (2014) proves that the concentration of measure at exponential rate implies the uniqueness.

6.2 Limitations of our method

We continue this discussion with an example showing that there exist processes that cannot have a perfect simulation using our algorithms even though they are ergodic and possess a unique translation-invariant stationary measure.

Example 11. *Consider $\mathcal{A} = \{0,1\}$ and let $r_n, n \geq 2$ be an increasing sequence of positive numbers such that $r_n \to 1$ as $n \to \infty$. Let $\mathbf{0}, \mathbf{1}$ be the histories formed by all zero and all one, respectively. Let $\mathcal{H} = \{0,1\}^{\mathbb{N}} \setminus \{\mathbf{0}, \mathbf{1}\}$. For any $x_{-\infty}^{-1} \in S$ and for any $g \in \mathcal{A}$, let*

$$p(g|x_{-\infty}^{-1}) = r_{\tau(x_{-\infty}^{-1})}\mathbf{1}_{\{x_{-1}\}}(g) + (1 - r_{\tau(x_{-\infty}^{-1})})\mathbf{1}_{\{x_{-1}\}^c}(g), \tag{31}$$

where

$$\tau(x_{-\infty}^{-1}) = \sup\{k \in \mathbb{N} : x_{-1} = \cdots = x_{-k}\}.$$

Let us assume that

$$\sum_{n=2}^{\infty}\prod_{j=2}^{n} r_j < \infty.$$

First, note that $\beta = 0$. Moreover, for any $n \geq 1$ and for any $b \in \mathcal{A}$, we have that if $a_{-j} = b$, for any $j = 1, \ldots, n$,

$$\alpha(g|a_{-n}^{-1}) = 0,$$

for $g \neq b$. Therefore, for any $n \geq 1$ the Markov kernel of order n defined in Definition 4 has the states $\mathbf{0}_{-n}^{-1}$ and $\mathbf{1}_{-n}^{-1}$ as absorbing states and therefore, once more, the condition of possessing a unique closed class is violated. However, we know that the process is ergodic and that there exists a translation-invariant stationary measure of the process. In fact, just note that the time in which the chain remains in a single symbol before jumping to the other symbol has finite expectation.

6.3 Comparing our coupling function to the "canonical" one

The main difference between the framework presented in Comets et al. (2002) and De Santis and Piccioni (2012) and the one introduced in Section 4 for Algorithm 1 is the fact that our coupling function is defined on the set $[0,1]^{+\infty}$ instead of $[0,1]\times\mathcal{H}$. In other words, the coupling function depends on the uniform random variables instead of depending on a string of symbols of $\mathcal{A}$. However, the process $(X_n)_{n\in\mathbb{Z}}$ that we defined using the coupling function F in Section 4 can be constructed using a coupling function "of the same type" as those considered in Comets et al. (2002) or in De Santis and Piccioni (2012). This implies that our construction here is more general. In what follows, we explain exactly what we mean by "of the same type".

The coupling function proposed by Comets et al. (2002) (which is called *canonical coupling function* by De Santis and Piccioni (2012)) has the form

$$G(u, x_{-\infty}^{-1}) := \sum_{g\in\mathcal{A}} g \sum_{n=0}^{+\infty} \mathbf{1}\{u \in \tilde{I}_n(g|x_{-n}^{-1})\}, \tag{32}$$

where $(\tilde{I}_n(g|x_{-n}^{-1}) : g \in \mathcal{A}, n = 0, 1, \ldots)$ are right-open intervals. By assuming that

$$\alpha(g|x_{-n}^{-1}) \to p(g|x_{-\infty}^{-1}), \text{ as } n \to \infty, \text{ for any } g \in \mathcal{A} \text{ and } x_{-\infty}^{-1} \in \mathcal{H}, \tag{33}$$

we have that $(\tilde{I}_n(g|x_{-n}^{-1}) : g \in \mathcal{A}, n = 0, 1, \ldots)$ is a partition of the interval $[0,1)$, where the first intervals in the left are $\tilde{I}_0(g)$, sequentially for $g \in \mathcal{A}$, satisfying $|\tilde{I}_0(g)| = \alpha(g)$, and the next intervals are $I_1(g|x_{-1})$, sequentially for $g \in \mathcal{A}$, satisfying $|\tilde{I}_1(g|x_{-1})| = \alpha(g|x_{-1}) - \alpha(g)$, and so on. If the condition in (33) is not satisfied, we additionally define $G(u, x_{-\infty}^{-1}) = *$ for

$$u \in [0,1) \setminus \bigcup_{n=0}^{\infty}\bigcup_{g\in\mathcal{A}} \tilde{I}_n(g|x_{-n}^{-1}).$$

This coupling function follows the natural order of considering the n-th first symbols of the past fixed successively for $n = 0, 1, 2, \ldots$. This idea can be generalized by considering that we

successively fix the symbols of the past in a different way. For a sequence of sets $\tilde{S}_0 \subseteq \tilde{S}_1 \subseteq \ldots$, where $\tilde{S}_n \subseteq \{1, \ldots, n\}$ for any $n = 0, 1, \ldots$, and for $x_{-\infty}^{-1} \in \mathcal{H}$, let

$$x_{-k}^{(\tilde{S}_n)} = \begin{cases} x_{-k}, & \text{if } k \in \tilde{S}_n, \\ *, & \text{if } k \notin \tilde{S}_n. \end{cases}$$

In this framework we may construct a coupling function $G^{(\tilde{S}_n)_{n\geq 0}} : [0,1] \times \mathcal{H} \to \mathcal{A}$, similar to (32), in which the first intervals in the left are $\tilde{I}_0^{(\tilde{S}_0)}(g)$, sequentially for $g \in \mathcal{A}$, satisfying $|\tilde{I}_0^{(\tilde{S}_0)}(g)| = \alpha(g)$, and the next intervals are $I_1^{(\tilde{S}_1)}(g|x_{-1}^{(1)})$, sequentially for $g \in \mathcal{A}$, satisfying $|\tilde{I}_1^{(\tilde{S}_1)}(g|x_{-1}^{(1)})| = \alpha(g|x_{-1}^{(\tilde{S}_1)}) - \alpha(g)$, and so on. This means that we sequentially choose to fix symbols in the past based on the increasing sequence of sets $\tilde{S}_1, \tilde{S}_2, \ldots$.

Coming back to our Algorithm 1, for any $m \in \mathbb{Z}$ and $n \geq 1$, let $S_n(U_{m-n}^{m-1}) = \{j \in \{1, \ldots, n\} : X_{m-j}^{(m-n)} \neq *\}$. Moreover, let $S_0 = \emptyset$. It follows that $(X_n)_{n\in\mathbb{Z}}$ satisfies

$$X_n = G^{(S_j(U_{-n-j}^{-n-1}))_{j\geq 0}}(U_n, X_{-\infty}^{n-1}),$$

where the increasing sequence that we consider is random. Therefore, the above construction shows that the coupling function introduced in Section 4 for Algorithm 1 can be interpreted as a modification of the *canonical coupling function* which instead of successively considering the n-th first symbols of the past fixed successively for $n = 0, 1, 2, \ldots$, consider that the part of the past to be fixed is random, depending only on $(U_n)_{n\in\mathbb{Z}}$. The part of the past to be fixed is exactly the spontaneous symbols and those obtained by the update procedure introduced in Algorithm 1.

6.4 The set of admissible histories

The set $\mathcal{H}$ (see Definition 1) contains all the histories that are present in the stationary regime as proved in Proposition 1. However, not all elements of $\mathcal{H}$ are present in the stationary regime of the process, as illustrated by the following example.

Example 12. *Consider the Markov chain assuming values in $\mathcal{A} = \{1, 2, 3\}$ with transition probabilities given by the matrix $p : \mathcal{A} \times \mathcal{A} \to [0, 1]$ defined as follows*

$$p(1|2) = p(2|1) = p(1|1) = p(2|2) = \frac{1}{2}, \quad p(3|3) = 0.9, \text{ and } p(1|3) = 0.1.$$

In this example, the symbol 3 is transient, and therefore, this symbol has no mass in the stationary measure of the process. However, we have that $3^{\mathbb{N}} \in \mathcal{H}$.

The following example illustrates the importance of the definition of the set of admissible histories. We present a kernel in which the assumptions of our results do not hold when we consider a set greater than $\mathcal{H}$ instead of $\mathcal{H}$.

Example 13. *Consider $\mathcal{A} = \{0, 1, 2\}$ and let $r_n, n \geq 2$ be an increasing sequence of positive numbers such that $r_n \to 1$ as $n \to \infty$. Denote $\mathbf{0}, \mathbf{1}, \mathbf{2}$ as the histories formed by all zero, all one and all two, respectively. Let $S = \{0, 1, 2\}^{\mathbb{N}} \setminus \{\mathbf{0}, \mathbf{1}, \mathbf{2}\}$. For any $x_{-\infty}^{-1} \in S$ and for any $g \in \mathcal{A}$, let*

$$p(g|x_{-\infty}^{-1}) = \begin{cases} r_n & \text{if } g = x_{-1} \text{ and } x_{-1} = \cdots = x_{-n}, x_{-n} \neq x_{-n-1}, \text{with } n \geq 2, \\ \frac{1}{2}(1 - r_n) & \text{if } g \neq x_{-1} \text{ and } x_{-1} = \cdots = x_{-n}, x_{-n} \neq x_{-n-1}, \text{with } n \geq 2, \\ \frac{1}{2}\mathbf{1}\{g \neq x_{-1}\}, & \text{iff } x_{-1} \neq x_{-2}. \end{cases} \tag{34}$$

Let us assume that

$$\sum_{n=2}^{\infty} \prod_{j=2}^{n} r_j < \infty.$$

In this example, clearly $\mathcal{H} \subset S$. We will now explain why, when taking $\mathcal{H} = S$, the assumptions required for Algorithm 1 and Algorithm 2 are not satisfied. In fact, first we have that

$$\sum_{g\in\mathcal{A}} \inf\{p(g|x_{-\infty}^{-1}) : x_{-\infty}^{-1} \in S\} = 0.$$

Moreover, for any $n \geq 1$, for any $b, g \in \mathcal{A}$ with $g \neq b$,

$$\inf\{p(g|x_{-\infty}^{-1}) : x_{-\infty}^{-1} \in S, x_{-j} = b, \text{ for all } j = 1, \ldots, n\} = 0.$$

Therefore, for any $n \geq 1$ the Markov kernel of order n defined in Definition 4 always has the states $\mathbf{0}_{-n}^{-1}$, $\mathbf{1}_{-n}^{-1}$, $\mathbf{2}_{-n}^{-1}$ as absorbing states, which violates the condition of possessing a unique closed class.

Finally, recalling (1), we have that $\mathcal{H} = \{a_{-\infty}^{-1} \in \mathcal{A}^{\mathbb{N}} : a_{-j} \neq a_{-(j-1)}, \text{ for all } j = 2, 3, \ldots\}$. Therefore,

$$\alpha(g|b) = \frac{1}{2},$$

for any $b \in \mathcal{A}$ and $g \neq b$. This implies that the Markov kernel of order 1 defined in Definition 4 is aperiodic. Therefore, Assumption 1 holds for this kernel.

Note that by considering $\mathcal{H}$ instead of S does not cause problems when our aim is to sample the chain adapted to the kernel in stationary regime. By the first lemma of Borel-Cantelli, all histories with two equal adjacent symbols are not present in the stationary regime.

Acknowledgments

This work was produced as part of the activities of FAPESP Research, Innovation and Dissemination Center for Neuromathematics (grant # 2013/ 07699-0 , S.Paulo Research Foundation (FAPESP)). KL was successively supported by FAPESP fellowship (grant 2022/07386-0 and 2023/12335-9). We thank Mauro Piccioni, Sandro Gallo and Vicenzo Bonasorte for interesting discussions.

References

Bonar, D. D. and Khoury, M. J. (2006). *Real Infinite Series*. Classroom resource materials. Mathematical Association of America.

Brémaud, P. (2020). *Markov chains. Gibbs Fields, Monte Carlo Simulation and Queues*. Springer.

Comets, F., Fernández, R., and Ferrari, P. A. (2002). Processes with long memory: Regenerative construction and perfect simulation. *The Annals of Applied Probability*, 12(3):921 – 943.

De Santis, E. and Piccioni, M. (2012). Backward coalescence times for perfect simulation of chains with infinite memory. *Journal of Applied Probability*, 49(2):319 – 337.

De Santis, E. and Piccioni, M. (2013). Perfect simulation of autoregressive models with infinite memory. *J Stat Phys*, 150:1017–1029.

De Santis, E. and Piccioni, M. (2015). One-Dimensional Infinite Memory Imitation Models with Noise. *Journal of Statistical Physics*, 161:346 – 364.

Gallo, S. (2011). Chains with unbounded variable length memory: perfect simulation and a visible regeneration scheme. *Advances in Applied Probability*, 43(3):735 – 759.

Gallo, S. and Garcia, N. L. (2013). Perfect simulation for locally continuous chains of infinite order. *Stochastic Processes and their Applications*, 123(11):3877–3902.

Gallo, S., Lerasle, M., and Takahashi, D. Y. (2013). Markov approximation of chains of infinite order in the d-metric. *Markov Processes And Related Fields*, 19(1):51–82.

Gallo, S. and Takahashi, D. Y. (2014). Attractive regular stochastic chains: perfect simulation and phase transition. *Ergodic Theory and Dynamical Systems*, 34(5):1567–1586.

Garivier, A. (2015). Perfect simulation of processes with long memory: A "coupling into and from the past" algorithm. *Random Structures & Algorithms*, 46(2):300–319.

Onisescu, O. and Mihoc, G. (1935). Sur les chaînes statistiques. *C. R. Acad. Sci. Paris*, 200:511–512.

Propp, J. G. and Wilson, D. B. (1996). Exact sampling with coupled markov chains and applications to statistical mechanics. *Random Structures & Algorithms*, 9(1-2):223–252.

Rissanen, J. (1983). A universal data compression system. *IEEE Transactions on Information Theory*, 29(5):656–664.