A perfect sampling algorithm of random walks with forbidden arcs

Stéphane Durand

ENS of Lyon, 15 parvis René Descartes, 69007 Lyon, France
Inria 655 Avenue de l'Europe, 38330
Montbonnot-Saint-Martin, France
stephane.durand@ens-lyon.fr

1. Introduction

I will present a result obtained with Bruno Gaujal and Florence Perronin on simulation of grid Markov chain. Random walk are frequently used to model behavior of systems or complex randomized algorithm. one often needs to be able to compute or sample the stationary distribution.
The problem we consider here is sampling the stationary distribution of a random walk on a multidimensional grid of great dimension, when its size prevents from directly computing it.
A known solution is Monte-Carlo method, but it only give a asymptotically perfect approximation. Instead, I will present a method giving a perfect sample in finite time.

2. Model

We work on a finite grid $\mathcal{S} := \{1, \dots, N\}^d$, where both the span N and the dimension d are large, to which we add forbidden moves, couple of (point,direction) retrieved from the possibles moves on the grid. We make the hypothesis that the grid stay strongly connected. Hence there exists an unique stationary distribution.

3. Perfect sampling

This section is devoted to the construction of a perfect sampling algorithm of a random walk $X(n)$ over $\mathcal{S}$ where certain arcs are forbidden. This is done in several steps

3.1. Coupling and Rejection

The random walk over a grid with forbidden arcs is an irreducible, finite, discrete time Markov chain over a finite state space $\mathcal{S}$ denoted $X(n)_{n\in\mathbb{N}}$, with transition matrix P. By definition, for any position $\mathbf{a}$ and any direction $\mathbf{m} = \pm\mathbf{e}_i$, $P_{\mathbf{a},\mathbf{a}+\mathbf{m}} = \frac{1}{q_\mathbf{a}}$ where $q_\mathbf{a}$ is the number of possible moves from $\mathbf{a}$.

From $(X(n))_{n\in\mathbb{N}}$, one can construct a continuous time Markov chain $Y(t)_{t\in\mathbb{R}}$ over the same state space. The generator Q of Y is obtained by multiplying each line $\mathbf{a}$ in P by $q_\mathbf{a}$ and defining the diagonal element $Q_{\mathbf{a},\mathbf{a}}$ as $Q_{\mathbf{aa}} = -\sum_\mathbf{b} q_\mathbf{a} P_\mathbf{ab}$. Therefore, the rates from $\mathbf{a}$ to $\mathbf{a}+\mathbf{m}$ are all equal to one : $Q_{\mathbf{a},\mathbf{a}+\mathbf{m}} = 1$.
From $Y(t)$, it is possible to extract a new discrete time Markov chain, $Y(n)_{n\in\mathbb{N}}$ by uniformization. Its transition matrix is $\mathrm{Id} + \Lambda^{-1}Q$, where Λ (uniformization constant) is any positive real number larger than all $q_\mathbf{a}$'s. Since the total rate out of any state in Y is bounded by $2d$, it can be uniformized by $\Lambda = 2d$.

While it can be difficult to construct a grand coupling for chain X, such a construction is easy and natural for the chain Y since the rates are all equal. To couple the walks starting from all states, just pick one direction uniformly among the $2d$ possibilities and make every walk move in that direction. Those for which the move is not possible stay at the same position.
The stationary distribution of the first chain can be obtained from the new one through rejection. We just have to simulate one more step and reject if the new step is the same

Theorem 1. *If we can sample Y under its stationary distribution, then using the rejection we obtain samples distributed according to the stationary distribution of $X(n)$.*

3.2. Perfect Sampling, coupling from the past

In order to obtain a perfect sample, we need to have a sufficiently long random chain such that the walks starting from every state using this chain end at the same point, which will be the answer of the algorithm.
For that, we need to provide an efficient criterion choosing the move function of the current state and a random word. This was difficult with the first chain, but immediate with the second.
If we go forward, adding moves at the end of the simulation until all trajectories converge, we add a bias caused by the dependence of coupling time on the step.
Instead we need to go backward, adding the new moves at the beginning of the simulation. For every move added, the set of new trajectories is a subset of the previous one, meaning that when the last set is a point, it will stay unchanged by any addition of moves.

Lemma 1. *The coupling from the past converges to a point sampled according to the stationary distribution*

3.3. intervals

The size of the grid forbid us to compute every trajectories, but we can compute any supersets of the possi-

ble positions. This can increase the coupling time, and could make it infinite in the case of a bad choice of superset (for example, the classical interval). In case of convergence, we know that the point obtained is the same as the one we would obtain with computing trajectories separately.
We use a double superset, made of one interval, behaving as if there were no forbidden move on the grid, and a set of point, created by the forbidden moves. We can show that the expectation of the coupling time is polynomial ($O(N^2 d \log d)$) and thus the number of point stays computable

4. Complexity

The goal of this section is to bound the expected time and space cost for random grid, ie a grid with a known number k of random forbidden moves. We suppose the dimension d to be smaller than the span N of the grid.

4.1. Rejection

The reject probability is the probability that the next move cannot be taken by the walker who is in a stationary state (of Y). In general this probability is bounded by $\frac{1}{2d}$ (for example if the walker has all moves forbidden but one). So that the expected number of rejections is always bounded by 2d. However this bound is very loose. In the case without any forbidden move, the expected number of tries needed is $\frac{N}{N-1}$. The numerical experiments on random graph with a number of forbidden moves smaller or comparable to N show a probability of this order.

4.2. Interval Coupling time

We have some results on random or without forbidden moves grid, as approximations.

Lemma 2 (Coalescence in dimension 1)**.** *For any* $T > 0$, $\mathbb{P}(C > T) \leq \cos^T\left(\frac{\pi}{N+1}\right)\left(1 + O(\frac{1}{N^2})\right)$.

Lemma 3 (Coalescence in dimension d)**.** *Let us consider a random walk in a grid with no forbidden arcs. Let* C_d *be its coalescence time.*
(i) The number of simulated step before obtaining the coupled point is at most four time the coalescence time.
(ii) The expected coalescence time satisfies $\mathbb{E}[C_d] = O\left(N^2 d \log d\right)$.

Lemma 4. *If forbidden arcs are chosen randomly, uniformly among all arcs in the grid and if* k *is the expected number of forbidden arcs, then the maximal size* $|E|$ *of the set* E*, is bounded in expectation :* $\mathbb{E}[\,|E|\,] \leq \frac{kN}{\pi^2} + O(kd + \frac{kN}{d})$.

With this, we can conclude that most of the time is taken by the convergence of the interval, taking an average of $O(d \log dN^2)$ steps, and the average memory cost of the separated point is of the order of $\frac{kN}{\pi^2}$. A step can be computed in time O(k), so we obtain an algorithm for perfect sampling of grids whose time complexity is $O(kd \log dN^2)$.