Simulação Perfeita
Aula MI 625 - 03/11/2009

1 General definition

Let us start with an abstract definition embodying the three exact simulation algorithms, Coupling from the Past (CFTP), Fill's Interruptible (IA) and Backward-Forward Algorithm (BFA) described in the Introduction.

Definição 1.1 *A perfect simulation (or exact sampling) scheme for a probability space* $(\mathbf{x}, \Omega, \mathcal{F}, \mu)$ *consists in:*

(i) *A process* $\underline{V} = (V_t)_{t\geq 0}$,

(ii) *A* $\{\mathcal{G}_t\}$*-stopping-time,* $\tau = \tau(\underline{V})$*, where* $\mathcal{G}_t = \sigma(V_s, 0 \leq s \leq t)$*, such that*

$$\mathbb{P}\big(\tau(\underline{V}) < \infty\big) \;=\; 1 \;, \tag{1.2}$$

(iii) *A random function* $\Phi_{\underline{V}} : \mathbb{R}_+ \to \mathbf{x}$ *such that*

$$\mathbb{P}\big(\Phi_{\underline{V}}(\tau) \in A\big) \;=\; \mu(A) \;. \tag{1.3}$$

The definition is completely general: $\underline{V}$ is some underlying process and $\mathbb{P}$ is a probability measure defined in a sufficiently large space encompassing $\mathbf{x}$ and the state-space of the process $\underline{V}$. In fact, in CFTP and our algorithm, the set up is such that

$$\mathbb{P}\big(\Phi_{\underline{V}}(t) \in A\big) \;=\; \mu(A) \quad , \quad \forall t \geq \tau \;. \tag{1.4}$$

This is not so in the IA algorithm. Property (1.4) stems from the fact that in CFTP and in our case, the algorithm "looks into the past", and the process $\underline{V}$ is related to *past history* or *ancestry* of what happens at a *fixed* time, say time zero. The IA algorithm, instead, is constructed on the basis of the forward evolution but incorporates a time-reversed trajectory for the acceptance-rejection procedure.

2 Coupling from the past

The *coupling from the past algorithm* (CFTP) was the first feasible algorithm for a perfect simulation scheme. It was introduced by Prop and Wilson (1996) and after this paper was made available, there was a sequence of articles applying the method to several different situations. We are going to describe the application to some situations ranging from the simplest (discrete-time finite state Markov chain, vertical CFTP) to more complicated problems (area-interaction point process and Strauss processes, dominated CFTP).

2.1 Discrete-time

Consider the problem of generating a random sample from a distribution μ on a finite set $\mathcal{S}$ which is the unique invariant measure of a discrete-time aperiodic, irreducible, positive recurrent Markov chain $\{X_t, t \in \mathbb{Z}\}$ with state space $\mathcal{S}$. Let P denote its transition matrix. Their approach can be described as follows: simulate the Markov chain, coupling all the paths beginning from all possible initial states, a predetermined amount of time (from $-T$ to 0), if all paths coalesce at time 0, the coalescent state X_0 has the desired distribution π. If the paths did not coalesce, restart the chain at $-T' < -T$, from all possible initial states, preappending new moves to the old ones. They show that if enough moves are preapended, eventually all the paths will coalesce and the resulting coalescent state X_0 is an unbiased sample from π.

In this case, the ingredients of the algorithm are:

- A discrete-time backwards process defined by a sequence $(U_i)_{i\leq 0}$ of independent random variables uniformly distributed in $[0,1]$. The forward process $\underline{V}$ is simply defined as its time-inversion: $V_i = U_{-i}$.
- An updating function $F : \mathbf{x} \times [0,1] \to \mathbf{x}$ such that the Markov chain constructed by setting $X_n = F(X_{n-1}, V_n)$ has μ as unique invariant measure.

The definition of τ and $\Phi_{\underline{V}}$ is based on iterations of F:

$$F_{[k,k']}(x,\underline{V}) \;=\; F(F_{[k,k'-1]}(x,\underline{V}), V_{k'}) \tag{2.1}$$

for $k' \geq k$, where $F_{[k,k]}(x,\underline{V}) \;=\; F(x, V_k)$. Notice that $F_{[k,k']}(x,\underline{V})$ depends only on $(V_k, \ldots, V_{k'})$. Now,

$$\tau = \min\Big\{n : F_{[-n,0]}(x,\underline{V}) \text{ does not depend on } x\Big\} \tag{2.2}$$

and

$$\Phi_{\underline{V}}(t) \;=\; F_{[-t,0]}(x,\underline{V}) \;=\; F_{[-\tau,0]}(x,\underline{V}) \tag{2.3}$$

for any $x \in \mathbf{x}$ and $t \geq \tau$. For $t < \tau$ the value of $\Phi_{\underline{V}}$ is arbitrary.

In words, the process is simulated from time $-t$ to time 0, using the *same* realization $u_{-t}, \ldots, u_{-1}$ of the random variables $(U_i)_{-t\leq i\leq -1}$ for *all* possible initial states $X_{-t} = x$. If all the resulting trajectories coalesce at or before time 0, the value of X_0 is taken to be a sample of μ. If not, the simulation is started some other time $t' > t$, using, for the period $[-t, 0]$, the previous realization $u_{-t}, \ldots, u_{-1}$ of the independent random variables. This (backwards) iteration is continued until all trajectories are seen to coalesce before time 0. The key points of this prescription are: (i) the use of the same random numbers to generate trajectories for different initial states (coupling), (ii) the keeping of a given realization of random numbers for a given period in all iterations, and (iii) the use of a *fixed* time —called time 0— to register the sample.

The efficiency of the algorithm depends on the choice of the function F. A badly designed coupling can lead to extremely large values of τ. As an example of this, consider a process with $\mathbf{x} = \{0, 1\}$ and with probability 1/2 of jumping from any state to any other. Here $\mu(0) = \mu(1) = 1/2$. If one chooses $F(0, v) = 1 - F(1, v) = \mathbf{1}\{v < 1/2\}$, the resulting coupling time τ is infinite with probability one. The construction of "good" couplings requires the maximization of $\min_{x,y} \mathbb{P}(F(x, V_n) = F(y, V_n))$. This condition is strongly model-dependent. Every homogeneous (uniform) ergodic Markov process admits an F yielding a finite coalescence time, e.g. the Vaserstein coupling used by Dobrushin (1965) or the construction provided by Nummelin's splitting technique to get a constant $\epsilon > 0$ and a probability measure Q such that $\mathbb{P}(F_{[-1,0]}(x, \underline{V}) \in \cdot) \geq \epsilon Q(\cdot)$. See Foss and Tweedie (1992) for more details.

For completeness, let us see why the algorithm performs as stated. By definition of F,

$$\begin{aligned} \mu(x) &= \lim_{-T \to -\infty} \mathbb{P}\Big(F_{[-T,0]}(a, \underline{V}) = x\Big) \\ &= \lim_{-T \to -\infty} \Big[\mathbb{P}\Big(F_{[-T,0]}(a, \underline{V}) = x\, ,\, \tau \leq T\Big) \;+\; \mathbb{P}\Big(F_{[-T,0]}(a, \underline{V}) = x\, ,\, \tau > T\Big)\Big] \end{aligned} \tag{2.4}$$

and by definition of τ,

$$\mathbb{P}\Big(F_{[-T,0]}(a, \underline{V}) = x\, ,\, \tau \leq T\Big) \;=\; \mathbb{P}\Big(\Phi_{\underline{V}}(t) = x\Big) \qquad \forall t \geq \tau\, . \tag{2.5}$$

On the other hand, if F is well chosen, τ is finite with probability one, hence

$$\mathbb{P}\Big(F_{[-T,0]}(a, \underline{V}) = x\, ,\, \tau > T\Big) \;\leq\; \mathbb{P}(\tau > T \mid X_{-T} = a) \;=\; \mathbb{P}(\tau > T) \underset{T\to\infty}{\longrightarrow} 0\, . \tag{2.6}$$

This shows property (1.3).

While it is true that trajectories also coalesce when looked *forward* in time, an algorithm based on this fact does not lead to a perfect simulation scheme. Indeed, if τ^* is the forward coalescing time, the analogous of (2.4) holds,

$$\begin{aligned} \mu(x) &= \lim_{T \to \infty} \mathbb{P}(F_{[0,T]}(a, \underline{V}) = x) \\ &= \lim_{T \to \infty} \Big[\mathbb{P}(F_{[0,T]}(a, \underline{V}) = x\, ,\, \tau^* \leq T) + \mathbb{P}(F_{[0,T]}(a, \underline{V}) = x\, ,\, \tau^* > T)\Big]\, , \end{aligned} \tag{2.7}$$

but in general

$$\lim_{T \to \infty} \mathbb{P}\Big(F_{[0,T]}(a, \underline{V}) = x\, ,\, \tau^* \leq T\Big) \;\neq\; \mathbb{P}\Big(\Phi_{\underline{V}}(\tau^*) = x\Big) \tag{2.8}$$

(in fact, the limit may not even exist).

Example 1: Let's consider a very simple example, take the random walk on $\{0, 1, 2, 3, 4\}$ with transition probabilities

$$P(i, i+1) = P(i+1, i) = 1/2, \text{ for } i = 1, 2, 3;$$

$$P(0, 0) = P(0, 1) = P(4, 4) = P(4, 3) = 1/2.$$

$$\pi = (1/5, 1/5, 1/5, 1/5, 1/5, 1/5)$$

In this case we have a stochastic flow defined as: if $X_0 = i$ then $X_n = F_{0,n}(i)$

The usual MCMC approach is to run the chain $X_n = F_{0,n}(i)$ for n large. Define

$$Y_n(i) := F_{-n,0}(i) \overset{\mathcal{D}}{=} X_n \text{ given } X_0 = i.$$

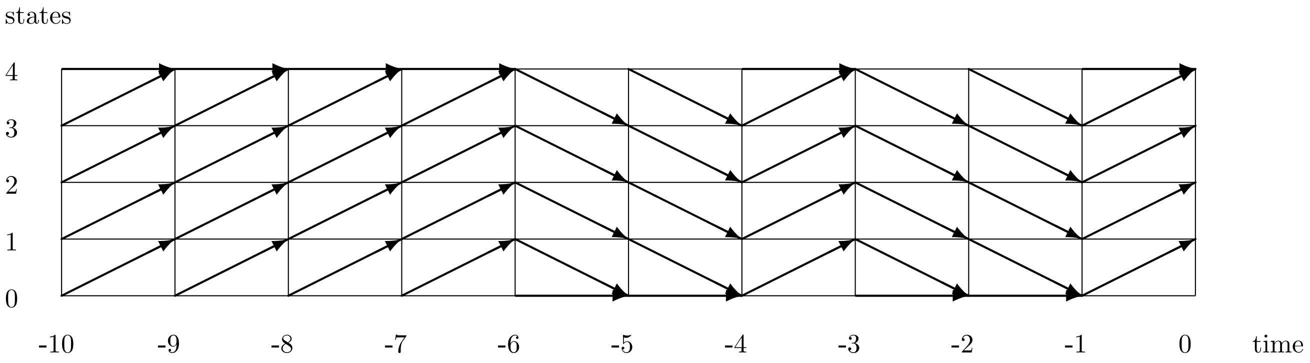

Figura 1: CFTP for the symmetric random walk in Example **??**

Notice that $\{Y_n, n \geq 0\}$ is not a Markov chain. Define

$$T_c := \inf\{n; Y_n(i) \text{ do not depend on } i\}.$$

Then, it is easy to see that

$$Y_\infty = \lim_{n\to\infty} Y_n(i) = Y_{T_c}(i) \sim \pi.$$

We have that T_c is *coalescence time* and it is a stopping time in the reverse filtration:

$$T_c = \inf\{n; F_{-n,0} \text{is constant}\}$$

and $F_{-n,0}(i) = Y_\infty$ has the desired distribution π.

In the example, showed in Figure 1 $T_c = 10$ and $F_{-10,0}(i) = 1$ is an unbiased sample from π.

Notice that if we run the chain forward in time until coupling of all trajectories we have that the only possible coupling states are 0 and 4. In this case,

$$\mathbb{P}[X_{\tau^*} = 0] = \mathbb{P}[X_{\tau^*} = 4] = 1/2.$$

Let $\dots,, U_{-3}, U_{-2}, U_{-1}, U_0$ be independent identically distributed random variables and $\phi(\cdot,\cdot)$ a deterministic function such that

$$\mathbb{P}[\phi(i, U_0) = j] = P_{i,j}$$

for all $i, j \in \mathcal{S}$. Define

$$F_{m,n} = \phi(\phi(\phi(\dots, U_{-m}), U_{-m+1}), \dots, U_{-n}).$$

If, with probability one, there exists a T such that $F_{T,0}$ is constant, which value we denote by $\phi(\dots, U_{-2}, U_{-1}, U_0)$. Then,

$$\phi(\dots, U_{-2}, U_{-1}, U_0) \sim \pi.$$

2.1.1 Monotonicity

The CFTP algorithm has been designed for use with finite state space $\mathbf{x}$. The problem is to verify coalescence when the state space is very large or even infinite. Assume that we can update the chain using a monotone rule ϕ:

$$x \leq y \Rightarrow \phi(x, U_0) \leq \phi(y, U_0)$$

almost surely with respect to U_0.

Moreover, assume that $\mathcal{S}$ has maximal $\hat{1}$ and minimal $\hat{0}$ such that $\hat{0} \leq x \leq \hat{1}$ for all $x \in \mathcal{S}$ in some partial ordering. In this case only the extremal states have to be followed. In actual simulations τ is not really computed. In general $F_{[-n,0]}(x, \underline{V})$ is computed for all x and for different —but not all— values of n (for instance powers of 2) up to the first time $F_{[-n,0]}$ is constant in x. Notice that the number of simulation involved is equal to $2(1+2+4+\ldots) < 2^{k+2}$, (2 comes from following 2 trajectories). However, this is close to the optimum since T_c has to exceed 2^{k-1}, therefore to verify the coalescence we have to run $2.2^{k-1} = 2^k$ simulations. Consequently, the procedure of doubling until "overshooting" is within a factor of 4 from the best algorithm that prevent "overshooting".

Pseudo-code:

```
$T \leftarrow 1$
    $U_{-1} \sim U(0,1)$
repeat
    $upper \leftarrow \hat{1}$
    $lower \leftarrow \hat{0}$
    for $n = -T$ to $-1$
        $upper \leftarrow \phi(upper, U_n)$
        $lower \leftarrow \phi(lower, U_n)$
    $T \leftarrow 2T$
    for $n = -T$ to $-T/2$
        $U_n \sim U(0,1)$
until $upper = lower$
return $upper$
```

Remark: It is never to much stressing the fact that the same uniform random variables are used in each loop, that is, U_{-1} is going to be generated only once to go from step -1 to 0.

2.2 Jump processes in continuous-time

The previous algorithm can be trivially adapted for invariant measures of Markov jump processes with an embedded ergodic Markov chain. Let (Y_t) be a process of this type, with finite state space $\mathbf{x}$, rates $Q(x, y)$, $x, y \in \mathbf{x}$, and (unique) invariant measure μ. It is convenient to consider another process, with rescaled transition times, having the same invariant measure. The new process has transition times given by a Poisson process $N(t)$ of rate $\lambda = \max_x \sum_y Q(x,y)$ and transitions determined by an skeleton Markov chain $\underline{X}$ with transition probabilities

$$P(x,y) = \begin{cases} \lambda^{-1} Q(x,y) & \text{if } y \neq x \\ 1 - \lambda^{-1} \sum_{z \neq x} Q(x,z) & \text{if } y = x \end{cases} . \tag{2.9}$$

We have that

$$Y_t \overset{\mathcal{D}}{=} X_{N(t)} . \tag{2.10}$$

Therefore, the invariant measure for $\underline{X}$ coincides with that of the original process (Y_t). It is therefore enough to proceed as in the discrete-time case.

2.3 Dominated CFTP

In the case of unbounded infinite (or very large) state space, it is not possible to use the above described method. Kendall (1998) introduced a modification in Propp and Wilson's algorithm in order to apply it to point processes. The idea, however, is not limited to this case and has been used to generate from continuous unbounded state space Green and Murdoch (1999). It is also called horizontal CFTP Kendall and Møller (1999) and coupling into and from the past Wilson (2000).

The idea is to find another Markov chain $\{C_t, t \in \mathbb{Z}\}$ — chosen in such way that we know how to generate exactly from its invariant measure — that dominates the chain under study. Assume, without loss of generality, that the state space has a minimal state $\hat{0}$, but not a maximal state. In this case, the ingredients of the algorithm are:

- A coupling that guarantees that if for some t we have $C_t \geq X_t$ that the same is true for all subsequent times. Usually this is reached using the same variables $\underline{V}$ to update both chains at the same time. That is, there exist ϕ_1 and ϕ_2 such that $(X_{t+1}, C_{t+1}) = (\phi_1(X_t, U_t), \phi_2(C_t, U_t))$ and if $x \leq c$ we have $\phi_1(x,u) \leq \phi_2(x,u)$ for all $u \in [0,1]$.
- For any value $x \in \mathcal{S}$ and time $t < 0$, there exists a.s. an $s < t$ such that $X_t \leq C_t$ if $X_u = x$ for $u < s$.
- We can simulate directly from the invariant distribution of C_t.
- Given C_t the conditional distribution of (C_{t-1}, U_{t-1}) is known and we can sample from this distribution. That is, we can simulate C_t into the past. This can be obtained easily if C_t is reversible.

In this case, we can use the CFTP algorithm based on generating an upper process in the same way that vertical CFTP.

2.4 Point processes

Kendall (1997, 1998) introduced dominated CFTP algorithm to obtain simulations of processes that can be obtained as weighted Boolean models using quermass integrals. These include the *area-interaction* processes considered by Baddeley and van Lieshout (1999) described in Section **??**, where the point process is produced by the germs of a Boolean model Θ under the weighting

$$\gamma^{-\text{area}(\Theta)}.$$

Its Radon-Nikodym derivative (restricted to a bounded window Λ) is given by (*cf.* (**??**))

$$\mu_\Lambda(dN) \;=\; \frac{\kappa^{N(\Lambda)}\,\phi^{-m_d(N\oplus G)}}{Z_\Lambda(\kappa,\phi)}\,\mu^0_\Lambda(dN)\;, \tag{2.11}$$

where κ and ϕ are positive parameters, $Z_\Lambda(\kappa,\phi)$ is a normalizing constant and $N\oplus G$ is the *coverage process* given by

$$N\oplus G \;:=\; \bigcup_{x\in N}\{x+G\}\;. \tag{2.12}$$

The attractive processes can be simulated using CFTP methods in the presence of monotonicity, when models can be sandwiched between a "maximal" and a "minimal" weighted Boolean models. In fact, through a minor modification the algorithm is also applicable to repulsive point processes Kendall (1997). We describe his scheme.

Consider the space-time Boolean model of cylinders constructed in Section **??**.

Finite-volume construction

Now, fix $-T<0$; for $t\in[-T,0]$ we are going to follow the evolution of three processes: $\eta^{\max}_{-T}(t)$, $\eta^{\min}_{-T}(t)$ and $\eta_{-T}(t)$ on Λ. Each process will have initial configuration $\eta^{\min}_{-T}(-T)\subset\eta_{-T}(-T)\subset\eta^{\max}_{-T}(-T)$ and they will use the finite set of marked cylinders

$$\mathcal{C}^\Lambda_{-T}=\{C\in\mathcal{C};\text{basis}(C)\in\Lambda,\text{Life}(C)\cap[-T,0]\neq\emptyset\}. \tag{2.13}$$

The initial "maximal" and "minimal" configurations are defined by

$$\eta^{\max}_{-T}(-T) \;=\; \{\text{basis}(C);C\in\mathcal{C}^\Lambda_{-T},\text{Life}(C)\ni -T\} \tag{2.14}$$

$$\eta^{\min}_{-T}(-T) \;=\; \{\text{basis}(C);C\in\mathcal{C}^\Lambda_{-T},\text{Life}(C)\ni -T,\text{Flag}(C)\le\phi^{-m_d(G)}\} \tag{2.15}$$

while $\eta_{-T}(-T)$ can be any arbitrary subset of $\eta^{\max}_{-T}(-T)$ and superset of $\eta^{\min}_{-T}(-T)$.

Using the graphical construction defined in Section **??**, it is possible to couple monotonically the trajectories $\eta^{\max}_{-T}(t)$ and $\eta^{\min}_{-T}(t)$ for all $-T\le t\le 0$ until the coalescence time

$$T_C:=\min\{T;\eta^{\max}_{-T}(0)=\eta^{\min}_{-T}(0)\}. \tag{2.16}$$

In general, the algorithm is run for fixed times $T_1 < T_2 < T_3 < \ldots$, with $T_N = e^N$.

The modification needed for the repulsive case, is that in **F.V.3** (Section **??**),

- a point ξ is added to $\eta^{\max}_{-T}(t)$ if the mark

$$z > \phi^{m_d(G) - m_d((\xi+G)\setminus(\eta^{\min}_{-T}(t-)\oplus G))}.$$

- a point ξ is added $\eta^{\min}_{-T}(t)$ if the mark

$$z > \phi^{m_d(G) - m_d((\xi+G)\setminus(\eta^{\max}_{-T}(t-)\oplus G))}.$$

The above references analyze processes in a finite window with *fixed* boundary conditions. On the other hand, it seems interesting to consider finite windows of an infinite-volume distribution. The only mention to this is by Kendall (1997), who points out a scheme that requires that the underlying Boolean model do not exhibit percolation in space-time. In this case, the CFTP method can be extended by looking at $[-T, 0] \times [-K, K]^d$ for ever increasing T and K. The lack of percolation ensures that eventually the area-interaction process will not be affected by whatever boundary conditions are imposed at time $-T$ and outside $[-K, K]^2$. This lack of percolation argument is the same used in Section **??** for continuous unbounded loss networks, as before an oriented-percolation argument can lead to a scheme that can be applied to a broad regimen.

2.5 User Impatience Bias.

The coupling from the past algorithm possesses the impatient-user bias. That is, it has a running time which is not independent of the state sampled, thus if the user aborts a long run of the algorithm a bias is introduced. The following simple example is presented in Thonnes (1999). Consider the Markov chain X with state space $\{0, 1, 2\}$ and transition matrix

$$P = \begin{pmatrix} 1/2 & 1/2 & 0 \\ 0 & 0 & 1 \\ 1/2 & 0 & 1/2 \end{pmatrix}.$$

The stationary distribution is given by $\pi = (2/5, 1/5, 2/5)$. We can simulate X using the following rule:

$$\phi(x, U) = \begin{cases} 0, & \text{for } x = 0, 2 \text{ and } U \leq 1/2 \\ \min\{x+1, 2\}, & \text{for } x = 0, 2 \text{ and } U > 1/2 \\ 2, & \text{for } x = 1 \end{cases}$$

where U is a $U(0, 1)$ random variable. Notice that this rule is not monotone and to run CFTP we need to follow the simulation for all 3 states.

Denote by

$$\phi(x, U_0, U_1, \ldots, U_T) = \phi(\phi(\ldots\phi(\phi(x, U_0), U_1), \ldots), U_T) \tag{2.17}$$

the state of the chain at time 0 starting the chain at time $-T$ in state x.

Suppose now that the user always terminates a run of CFTP after I iterations without obtaining coalescence. In this case, if coalescence is attained at time T we are sampling from the distribution defined by

$$\mathbb{P}\big(\phi(x, U_1, \ldots, U_T) \in A \,|\, T < I\big) \tag{2.18}$$

which is different from π since

$$\mathbb{P}\big(\phi(x, U_1, \ldots, U_T) = r \,|\, T < I\big) = \frac{\mathbb{P}\big(T < I \text{ and } \phi(x, U_1, \ldots, U_T) = r\big)}{\mathbb{P}(T < I)}. \tag{2.19}$$

Let $N = 2^{I-1}$, then by combinatorial arguments we can show that

$$\mathbb{P}\big(\phi(x, U_1, \ldots, U_T) = r \,|\, T < I\big) = \frac{\sum_{k=2}^{N} \big(1/2\big)^{k-1} \big[1/2 P_{0r}^{(N-k)} + 1/2 P_{2r}^{(N-k)}\big]}{1 - \big(1/2\big)^{N-1}} \tag{2.20}$$

where $P^{(k)}$ is the k-step transition matrix. Therefore, for $I \geq 3$ we have

$$\begin{aligned}
\mathbb{P}\big(\phi(x, U_1, \ldots, U_T) = 0 \,|\, T < I\big) &= \frac{2}{5}\Big[\frac{1 - 2^{-N}}{1 - 2^{-N+1}}\Big] \\
\mathbb{P}\big(\phi(x, U_1, \ldots, U_T) = 1 \,|\, T < I\big) &= \frac{1}{5}\Big[\frac{1 - 2^{-N}}{1 - 2^{-N+1}}\Big] \\
\mathbb{P}\big(\phi(x, U_1, \ldots, U_T) = 2 \,|\, T < I\big) &= 1 - \frac{3}{5}\Big[\frac{1 - 2^{-N}}{1 - 2^{-N+1}}\Big].
\end{aligned}$$

Although the bias decreases with I as expected, the sample will always be biased.

3 Fill's interruptible algorithm

Fill (1998) describes a perfect sampling scheme based on a rejection sampling method, which protects against the user impatience bias. Consider a Markov chain on a partially ordered finite state space $(\mathcal{S}, \leq)$ with stationary distribution μ. Suppose that the state space has a maximum element $\hat{1}$ and a minimal element $\hat{0}$. The transition matrix P is such that its *time reversal* $\tilde{P}$ defined by

$$\tilde{P}(x, y) = \frac{\mu(y) P(y, x)}{\mu(x)} \tag{3.1}$$

for x such that $\mu(x) > 0$ is monotone with respect to $\leq$. As $\tilde{P}$ is monotone, it is well-known, *e.g.* Theorem 1 in Kam?? (1977), that there exists an upward kernel $K_{(x,y)}(\cdot, \cdot)$ such that

$$\tilde{P}(y, y') = \sum_{x' \in \mathcal{S}} \tilde{P}(x, x') K_{(x,y)}(x', y') \tag{3.2}$$

for all $y \in \mathcal{S}$.

It is assumed that it is possible to sample from the measure $K_{(x,y)}(x', \cdot)$ whenever $x \leq y$ and $\tilde{P}(x, x') > 0$.

Definição 3.3 *A monotone transition rule for a transition matrix $\tilde{P}$ on a partially ordered space $(\mathcal{S}, \leq)$ is a measurable function $\tilde{f} : \mathcal{S} \times \mathcal{U} \to \mathcal{S}$ together with a random variable U taking values in an arbitrary probability space $\mathcal{U}$ such that:*

(i) $\tilde{f}(x,u) \le \tilde{f}(y,u)$ *for all* $u \in \mathcal{U}$ *whenever* $x \le y$*;*

(ii) $\mathbb{P}\big(\tilde{f}(x,U) \in \cdot\big) \;=\; \tilde{P}(x,\cdot)$ *for all* $x \in \mathcal{S}$*.*

If $\tilde{P}$ has a monotone transition rule $\tilde{f}$ then we can take as upward kernel

$$K_{(x,y)}(x',y') \;:=\; \mathbb{P}\big(\tilde{f}(y,U) = y' \mid \tilde{f}(x,U) = x'\big), \tag{3.4}$$

for all $y' \in \mathcal{S}$ when $x \le y$ and $\tilde{P}(x,x') > 0$.

Algorithm: Fill's algorithm consists in three steps:

1. Start X in $\hat{0}$ and runs it for t steps. Record the obtained trajectory $(X_0 = \hat{0}, X_1, \ldots, X_t = z)$.
2. Reverse the obtained trajectory in time leading to the time-reversed trajectory (which is regarded as a $\tilde{P}$ trajectory conditioned to start at z and to end at $\hat{0}$)
$$(\tilde{X}_0 = z, \tilde{X}_1, \ldots, \tilde{X}_t = \hat{0}) = (X_t = z, X_{t-1}, \ldots, X_0 = \hat{0}).$$
3. A second Markov chain $\tilde{Y}$ is simulated for t steps using the upward kernels $K_{(x,y)}(\cdot,\cdot)$ together with the time-reversed trajectory. The initial state of $\tilde{Y}$ is set to be $\hat{1}$. Then $\tilde{Y}_k$ for $k = 1, \ldots, t$ is simulated according to the kernel
$$K_{(\tilde{X}_{k-1},\tilde{Y}_{k-1})}(\tilde{X}_k,\cdot).$$
If $\tilde{Y}_t = \hat{0}$ the proposed sample z is accepted.
4. If the sample is not accepted, reinitiate the process with $t+1$ independently.

Notice that this algorithm is based on the well-known acceptance-rejection algorithm for sampling, see Ripley (1987).

(a) Generate an observation from $P^t(\hat{0},\cdot)$;

(b) Find a constant c such that
$$c > \frac{\pi(z)}{P^t(\hat{0},z)}, \qquad \text{for all } z \in \mathcal{S} \text{ such that } P^t(\hat{0},z) > 0.$$

(c) Accept z as an observation from π with probability $c^{-1}\pi(z)/P^t(\hat{0},z)$.

Notice that for this algorithm to work there is a couple of questions to be answered.

- How to choose c? By definition of $\tilde{P}$ we know that
$$\frac{\pi(z)}{P^t(\hat{0},z)} \;=\; \frac{\pi(\hat{0})}{\tilde{P}^t(z,\hat{0})} \tag{3.5}$$

and by the monotonicity of $\tilde{P}$, we can choose

$$c = \frac{\pi(\hat{0})}{\tilde{P}^t(\hat{1},\hat{0})}. \tag{3.6}$$

Thus, step (c) says to accept z as an observation from π with probability

$$\frac{\tilde{P}^t(\hat{1},\hat{0})}{\pi(\hat{0})} \times \frac{\pi(z)}{P^t(\hat{0},z)} = \frac{\tilde{P}^t(\hat{1},\hat{0})}{\pi(\hat{0})} \times \frac{\pi(\hat{0})}{\tilde{P}^t(z,\hat{0})} = \frac{\tilde{P}^t(\hat{1},\hat{0})}{\tilde{P}^t(z,\hat{0})} \tag{3.7}$$

- How to design a coin-flip with probability of heads equals to $\tilde{P}^t(\hat{1},\hat{0})/\tilde{P}^t(z,\hat{0})$? Running the coupled-reversed chain starting at $\hat{1}$ for t steps, if the $\tilde{Y}_t = \hat{0}$, the coin flips head. In fact,

$$\mathbb{P}\big(\tilde{Y}_t = \hat{0} \mid \tilde{X}_0 = z, \tilde{X}_t = \hat{0}, \tilde{Y}_0 = \hat{1}\big) = \frac{\tilde{P}^t(\hat{1},\hat{0})}{\tilde{P}^t(z,\hat{0})}. \tag{3.8}$$

In this case we can construct the process $\underline{V} = (\mathbb{V}_t, t \geq 1)$ where $V_t = ((X_0,\ldots,X_t),(\tilde{Y}_0,\ldots,\tilde{Y}_t))$ are independently generated by the rejection algorithm described above.

$$\tau = \min\{t; \tilde{Y}_t = \hat{0}\} \tag{3.9}$$

and

$$\Phi_{\underline{V}}(t) = X_t. \tag{3.10}$$

Fill suggests the the algorithm should be run using t as powers of 2 similarly to CFTP.

3.1 Application to attractive spin systems

Consider an attractive spin system with attractive equilibrium measure π. Consider the Gibbs sampler (heat-bath algorithm) with uniform random update for attractive spins systems as described in Section **??**. In this case, the chain is reversible with $\tilde{P} = P$ which is monotone since the system is attractive, then (**??**) gives us a monotone transition rule $\phi = \tilde{f}$. IA algorithm becomes:

1. Start the chain σ at $\sigma^{\min} \equiv -1$ and run it for t steps using ϕ as an updating rule. We obtain

$$(\sigma_0 \equiv -1, \sigma_1, \ldots, \sigma_{t-1}, \sigma_t = \zeta).$$

2. Construct the time reversed trajectory

$$(\tilde{\sigma}_0 = \zeta, \tilde{\sigma}_1 = \sigma_{t-1}, \ldots, \tilde{\sigma}_t \equiv -1).$$

3. Simulate a second Markov chain $\tilde{\eta}$ for t steps starting at $\tilde{\eta}_0 \equiv +1$ using the following rule:

   (a) When $\tilde{\sigma}_{n-1} \neq \tilde{\sigma}_n$ (they disagree at a unique site i),

      - If $\tilde{\sigma}_{n-1}(i) = -1$ and $\tilde{\sigma}_n(i) = +1$ then set $\tilde{\eta}_n(i) = +1$;

- If $\tilde{\sigma}_{n-1}(i) = +1$ and $\tilde{\sigma}_n(i) = -1$ then set

$$\begin{cases} \tilde{\eta}_n(i) = -1, & \text{with probability } \frac{\pi(\tilde{\eta}^-_{n-1})}{\pi(\tilde{\eta}^-_{n-1})+\pi(\tilde{\eta}^+_{n-1})} \frac{\pi(\tilde{\sigma}_{n-1})+\pi(\tilde{\sigma}_n)}{\pi(\tilde{\sigma}_n)} \\ \tilde{\eta}_n(i) = +1, & \text{with probability } 1 - \frac{\pi(\tilde{\eta}^-_{n-1})}{\pi(\tilde{\eta}^-_{n-1})+\pi(\tilde{\eta}^+_{n-1})} \frac{\pi(\tilde{\sigma}_{n-1})+\pi(\tilde{\sigma}_n)}{\pi(\tilde{\sigma}_n)} \end{cases}$$

(b) $\tilde{\sigma}_{n-1} = \tilde{\sigma}_n = \tilde{\sigma}^*$, then the computation of the conditional probability is a little messy, but we can overcome this problem by noticing that $\tilde{\sigma}_n = \sigma_{t-n}$ and $\tilde{\sigma}_{n-1} = \sigma_{t-n+1}$ and this transition was produced in step 1 by generating $U \sim U(0,1)$ and $V \sim U(\Lambda)$ independent and updating

$$\tilde{\sigma}^* = \phi(\tilde{\sigma}^*, U, V).$$

So we can store (U, V) and use it to set

$$\tilde{\eta}_n = \phi(\tilde{\eta}_{n-1}, U, V).$$

4. If $\tilde{\eta}_t \equiv -1$ then the proposed sample ζ is accepted, if not reinitiate the process at Step 1.

3.2 Point processes

Thonnes (1999) uses IA algorithm to simulate from penetrable sphere model without impatient user bias. Her argument follows Fill's interruptible algorithm and uses a forward construction and a backward checking.

The objective is to simulate from the bi-dimensional point process $(N, M) \in \mathcal{N} \times \mathcal{N}$ described in Section **??**. The crucial observation is that under the model (**??**), the the conditional distribution of N given M is a homogeneous Poisson process with intensity β_1 on $\Lambda \setminus (M \oplus G)$. Where G is a sphere of radius R. Similarly, the conditional distribution of M given N is a homogeneous Poisson process with intensity β_2 on $\Lambda \setminus (N \oplus G)$.

In this case, a convenient Markov chain to be used is the Gibbs sampler. Given the configuration of the process at time t to be (n_t, m_t), then in the next step

$$\begin{aligned} m_{t+1} &\sim \beta_2 \text{ Poisson on } \Lambda \setminus (n_t \oplus G) && (3.11) \\ n_{t+1} &\sim \beta_1 \text{ Poisson on } \Lambda \setminus (m_{t+1} \oplus G). && (3.12) \end{aligned}$$

Notice that, this Markov chain has an uncountable state space we can partially order this space by considering

$$(n, m) \leq (n', m') \qquad \text{if} \qquad n \subset n' \quad \text{and } m \supset m'.$$

It is easy to check that the Gibbs sampler defined by (3.11) and (3.12) defines a monotone transition rule f given by

$$f(n, m, V_1, V_2) = (n', m') \tag{3.13}$$

where V_1 and V_2 are independent Poisson point processes on Λ with rates β_1 and β_2 respectively and $n' = V_1 \setminus (m \oplus G)$ and $m' = V_2 \setminus (n' \oplus G)$. Notice that

$$f(n, m, V_1, V_2) \leq f(n', m', V_1, V_2) \quad \text{whenever} \quad (n, m) \leq (n', m'). \tag{3.14}$$

In fact, if $(n_1, m_1) = f(n, m, V_1, V_2)$, and $(n'_1, m'_1) = f(n', m', V_1, V_2)$, then

$$m_1 = V_1 \setminus (n \oplus G) \supset m'_1 = V_1 \setminus (n' \oplus G), \text{ since } n \oplus G \subset n' \oplus G$$

and

$$n_1 = V_2 \setminus (m_1 \oplus G) \subset n'_1 = V_2 \setminus (m'_1 \oplus G), \text{ since } m_1 \oplus G \supset m'_1 \oplus G.$$

However, the state space $\mathcal{N} \times \mathcal{N}$ does not have a maximal or minimal element. Instead, Haggstrom, van Lieshout and Møller (1999) call an element (n, m) $q$uasi-maximal if

$$\Lambda \subset n \oplus G \qquad \text{and} \qquad m = \emptyset. \tag{3.15}$$

Similarly, the element (n, m) is quasi-minimal if

$$n = \emptyset \qquad \text{and} \qquad \Lambda \subset m \oplus G. \tag{3.16}$$

It is easy to check that if (n^0, m^0) is a minimal state and (n^1, m^1) is a maximal state, then for an arbitrary configuration (n, m), if we call (N^0, M^0), (N, M) and (N^1, M^1) the Markov chains obtained with initial states $(N^0_0, M^0_0) = (n^0, m^0)$, $(N_0, M_0) = (n, m)$ and $(N^1_0, M^1_0) = (n^1, m^1)$ respectively, we have

$$(N^0_n, M^0_n) \leq (N_n, M_n) \leq (N^1_n, M^1_n), \text{ for all } n \geq 1. \tag{3.17}$$

In fact, if $(N^0_1, M^0_1) = f(n^0, m^0, V_1, V_2)$, $(N_1, M_1) = f(n, m, V_1, V_2)$ and $(N^1_1, M^1_1) = f(n^1, m^1, V_1, V_2)$, then

$$M^0_1 = V_1 \text{ and } N^0_1 = V_2 \setminus (V_1 \oplus G)$$

$$M_1 = V_1 \setminus (n \oplus G) \text{ and } N_1 = V_2 \setminus (M_1 \oplus G)$$

and

$$M^1_1 = V_1 \setminus \Lambda = \emptyset \text{ and } N^1_1 = V_2$$

Then,

$$M^0_1 \supset M_1 \supset M^1_1 \quad \text{and} \quad N^0_1 \subset N_1 \subset N^1_1.$$

This proves (3.17) for $n = 1$, for $n > 1$ it is a consequence of the monotonicity of f.

Notice that, the Markov chain defined by the monotone rule f defined by (3.13), is reversible. However, the two-step rule (3.11) and (3.12) is not reversible. The two step rule for the reversed chain is:

$$\tilde{n}_{t+1} \sim \beta_1 \text{ Poisson on } \Lambda \setminus (\tilde{m}_t \oplus G) \tag{3.18}$$

$$\tilde{m}_{t+1} \sim \beta_2 \text{ Poisson on } \Lambda \setminus (\tilde{n}_{t+1} \oplus G). \tag{3.19}$$

Fill's algorithm

1. Start the chain (n, m) at a quasi-minimal point (n^0, m^0) and run it for t steps using (3.11) and 3.12 as a two step update rule. We obtain

$$\big((n_0, m_0) = (n^0, m^0), (n_1, m_1), \ldots, (n_{t-1}, m_{t-1}), (n_t, m_t) = (n^*, m^*)\big).$$

2. Construct the time reversed trajectory

$$\big((\tilde{n}_0, \tilde{m}_0) = (n^*, m^*), (\tilde{n}_1, \tilde{m}_1) = (n_{t-1}, m_{t-1}), \ldots, (\tilde{n}_t, \tilde{m}_t) = (n^0, m^0)\big).$$

3. Simulate a second Markov chain $(\tilde{v}, \tilde{w})$ for t steps starting at a quasi-maximal state using the following rule:

   - If at time n we have the following transitions on the time-reversed chain $(\tilde{n}_{n-1}, \tilde{m}_{n-1}) \rightarrow (\tilde{n}_{n-1}, \tilde{m}_n) \rightarrow (\tilde{n}_n, \tilde{m}_n)$, then let

$$v_n \sim \beta_1 \text{ Poisson on } \tilde{m}_{n-1} \oplus G \tag{3.20}$$

$$\tilde{v}_n \leftarrow [\tilde{n}_n \cup v_n] \setminus [\tilde{w}_{n-1} \oplus G] \tag{3.21}$$

$$\tilde{w}_n \leftarrow \tilde{m}_n \setminus [\tilde{v}_n \oplus G]. \tag{3.22}$$

4. If $(\tilde{v}_t, \tilde{w}_t)$ is quasi-minimal then the proposed sample ζ is accepted, if not reinitiate the process at Step 1.