A Festschrift for Herman Rubin
Institute of Mathematical Statistics
Lecture Notes – Monograph Series
Vol. 45 (2004) 342–347

Generalized Accept–Reject sampling schemes

George Casella[1,*] **Christian P. Robert**[2,†]**, and Martin T. Wells**[3,*]

University of Florida, Université Paris 9 – Dauphine and Cornell University

Abstract: This paper extends the Accept–Reject algorithm to allow the proposal distribution to change at each iteration. We first establish a necessary and sufficient condition for this generalized Accept–Reject algorithm to be valid, and then show how the resulting estimator can be improved by Rao-Blackwellization. An application of these results is to the perfect sampling technique of Fill (1998), which is a generalized Accept–Reject algorithm.

1. Preface by GC

This paper is especially appropriate for a volume dedicated to Herman Rubin, as he was the first person who ever mentioned the Accept–Reject algorithm to me although, at the time, I didn't understand a word that he was talking about. I was a graduate student at Purdue in the mid-70s, and Herman was always working on some problem, and if he saw you in the hall he would tell you about it. One day he told me he was working on an algorithm that generated "test exponentials" to get normal random variables. I had no idea why anyone would want to do such a thing (remember the 70s ? – we were proving theorems!). Herman eventually wrote a technical report, but I don't think I ever read it and don't know if it ever was published. And then Herman got interested in other things. But when I think of this story I often wonder how much further along Monte Carlo methods would be today if Herman kept his interest in those "test exponentials"!

2. Introduction

Accept–Reject algorithms are based on the use of a proposal distribution g which serves to simulate from a given target density f, when the ratio f/g is bounded by $1/\epsilon$, say. The standard Accept–Reject Algorithm is

Algorithm $\mathbf{A}_1$—Accept–Reject.
`At iteration` i $(i \geq 1)$

1. `Generate` $X_i \sim g$ `and` $U_i \sim \mathcal{U}([0,1])$, *independently.*

2. `If` $U_i \leq \epsilon f(X_i)/g(X_i)$, `accept` $X_i \sim f$;

3. `otherwise, move to iteration` $i+1$.

†Research supported by EU TMR network ERB-FMRX-CT96-0095.
*Both Prof. Casella and Prof. Wells are supported by National Science Foundation Grant DMS-9971586.
[1]Department of Statistics, University of Florida, Gainesville, FL 32611-8545, USA. e-mail: casella@stat.ufl.edu
[2]Laboratoire de Statistique, CREST, Insee, Paris and Université Paris 9 – Dauphine, France. e-mail: xian@ceremade.u-dauphine.fr
[3]Cornell University, Ithaca, NY 14851. e-mail: mtw1@cornell.edu
Keywords and phrases: Monte Carlo methods, Accept–Reject, stopping rule, recycling, uniform variable.
AMS 2000 subject classifications: 65C60.

Since the inequality is not always satisfied, the algorithm generates pairs (X_i, U_i) that are rejected. These pairs can be recycled in many ways, including the Rao-Blackwellizing approach by Casella and Robert (1996) which replaces the standard estimator δ based on the accepted pairs with the conditional expectation $\mathbb{E}[\delta|x_1, \ldots, x_n, n]$, which integrates out the uniform variables.

We give in this note a necessary and sufficient condition for a generalized Accept–Reject algorithm to be valid and show that Rao–Blackwellization also applies here, allowing the use of the rejected samples to produce an improved estimator.

This work was partially motivated by that of Fill (1998), who developed an interruptible perfect sampling algorithm as an alternative to Propp and Wilson's (1996) coupling from the past technique. Perfect sampling results in iid outputs from the stationary distribution of the MCMC Markov chain (see Dimakos (2001), Robert and Casella (1999) or Casella, Lavine and Robert (2000) for introductions to perfect sampling). At the core of Fill's algorithm, described in Section 5, is an Accept–Reject algorithm with the feature that the proposal distribution can be modified at each step.

The possibility of changing the proposal distribution at each failure/rejection implies that his method does not fall in the category of a standard Accept–Reject algorithm. It is this more general Accept–Reject algorithm that we are interested in.

3. A generalized Accept–Reject algorithms

We consider the following extension to the standard Accept–Reject algorithm:

Algorithm A_2—Generalized Accept–Reject.
`At iteration` i $(i \geq 1)$

1. `Generate` $X_i \sim g_i$ `and` $U_i \sim \mathcal{U}([0,1])$, *independently.*

2. `If` $U_i \leq \epsilon_i f(X_i)/g_i(X_i)$, `accept` $X_i \sim f$;

3. `otherwise, move to iteration` $i+1$. -

Thus, at each iteration i $(0 < i < \infty)$, the algorithm uses a different pair (g_i, ϵ_i) such that $\epsilon_i f(x)/g_i(x) \leq 1$, uniformly in x. Each of these pairs is thus acceptable for the original Accept–Reject scheme. However, the proposal distribution keeps changing at each reject iteration and may be more adaptive than the single Accept–Reject proposal distribution, or even the adaptive rejection algorithm of Gilks and Wild (1992), which uses an envelope on the target density.

If the proposal distribution is parameterized by a parameter θ, we can select a pre-determined sequence of values of θ to monitor the performance in simulating the distribution of interest f. The value of θ at the time of acceptance can then be exploited in further simulations without jeopardizing the independence properties of the algorithm.

The extension of the Accept–Reject Algorithm does not hold in full generality, in the sense that the distribution of the accepted random variable may not necessarily be the correct one. A minimum requirement must be imposed on the sequence of the ϵ_i's (and hence on the g_i's).

If we denote by Z the (possibly defective) random variable that is output by Algorithm A_2, Z has the cdf (for simplicity, in the univariate continuous case):

$$\begin{aligned}
&P(Z \le z)\\
&\quad = \sum_{i=1}^{\infty} P(Z \le z, Z = X_i)\\
&\quad = \sum_{i=1}^{\infty} P\big(X_i \le Z, U_i \le f(X_i)\epsilon_i / g_i(X_i)\big) \prod_{j=1}^{i-1} P\big(U_j \ge f(X_j)\epsilon_j / g_j(X_j)\big)\\
&\quad = \sum_{i=1}^{\infty} \int_{-\infty}^{z} \frac{f(x)\epsilon_i}{g_i(x)} g_i(x)\,dx \prod_{j=1}^{i-1}(1-\epsilon_j) = \int_{-\infty}^{z} f(x)\,dx \sum_{i=1}^{\infty} \epsilon_i \prod_{j=1}^{i-1}(1-\epsilon_j).
\end{aligned}$$

Therefore, the output is distributed from f if $\sum_{i=1}^{\infty} \epsilon_i \prod_{j=1}^{i-1}(1-\epsilon_j) = 1$. The following theorem ties this condition to the divergence of an associated series.

Theorem 3.1. *The Generalized Accept–Reject Algorithm is valid if, and only if, the series* $\sum_{i=1}^{\infty} \log(1-\epsilon_i)$ *diverges, since*

$$\sum_{i=1}^{\infty} \epsilon_i \prod_{j=1}^{i-1}(1-\epsilon_j) = 1 \quad \textit{if and only if} \quad \sum_{i=1}^{\infty} \log(1-\epsilon_i) \ \textit{diverges.} \tag{1}$$

Proof. Note first that $\sum \epsilon_i \prod_{j=1}^{i-1}(1-\epsilon_j)$ necessarily converges to a limit less than, or equal to, 1 since

(a) for every $n \ge 1$,

$$\begin{aligned}
\xi_n &= \sum_{i=1}^{n} \epsilon_i \prod_{j=1}^{i-1}(1-\epsilon_j)\\
&= \epsilon_1 + (1-\epsilon_1)\big\{\epsilon_2 + (1-\epsilon_2)[\ldots(1-\epsilon_{n-1})\epsilon_n)\ldots]\big\}\\
&\le \epsilon_1 + (1-\epsilon_1)\big\{\epsilon_2 + (1-\epsilon_2)[\ldots\epsilon_{n-1} + (1-\epsilon_{n-1}))\ldots]\big\}\\
&= 1.
\end{aligned}$$

(b) the sequence $\{\xi_n\}$ is increasing with n.

Now, $\{\xi_n\}$ converges to 1 if, and only if, for every $0 < \eta < 1$, there exists n_0 such that

$$\xi_n > 1 - \eta \quad \text{for} \quad n > n_0. \tag{2}$$

The condition (2) is equivalent to, for $n > n_0$,

$$\begin{aligned}
&\epsilon_1 + (1-\epsilon_1)\big\{\epsilon_2 + (1-\epsilon_2)[\ldots(1-\epsilon_{n-1})\epsilon_n)\ldots]\big\} > 1-\eta\\
&\quad \Leftrightarrow \epsilon_2 + (1-\epsilon_2)\big\{\epsilon_3 + \ldots(1-\epsilon_{n-1})\epsilon_n]\ldots\big\} > \frac{1-\epsilon_1-\eta}{1-\epsilon_1} = 1 - \frac{\eta}{1-\epsilon_1}\\
&\quad \Leftrightarrow \ldots\\
&\quad \Leftrightarrow \epsilon_n > 1 - \frac{\eta}{\prod_{i=1}^{n-1}(1-\epsilon_i)}.
\end{aligned} \tag{3}$$

The sequence $\omega_n = \prod_{i=1}^{n-1}(1-\epsilon_i)$ with $\omega_1 = 1$ is decreasing and nonnegative. Thus, it either converges to 0 or to $\alpha > 0$. If it converges to 0, that is, if $\sum \log(1-\epsilon_i)$

diverges, the ratio η/ω_n goes to $+\infty$ with n and the right hand side in (3) is negative for n large enough, which ensures that (2) holds.

If $\{\omega_n\}$ converges to $\alpha > 0$, the series $\sum \log(1-\epsilon_i)$ converges and $\log(1-\epsilon_n)$ goes to 0 as n goes to infinity by Cauchy's criterion. Thus, $\{\epsilon_n\}$ converges to 0. Therefore, for δ small enough, there exists n_1 such that $\epsilon_n < \delta$ for $n > n_1$. If one chooses η such that $1 - \frac{\eta}{\alpha} = \delta$ and if (2) holds, one gets $\epsilon_n < \delta < \epsilon_n$ for $n > \max(n_0, n_1)$, which is impossible. □

This result has several implications. First, it shows that continued modifications of the proposal distribution in the Accept–Reject algorithm are legitimate as long as the acceptance rate ϵ_n does not converge to zero too fast. Second, the acceptance rate ϵ_n does not have to go to 1 with n, so some ϵ_n's (even an infinity of them) may be equal to 0, and the algorithm remains valid. Note however, that if one ϵ_n is equal to 1, the sequence terminates.

Theorem 3.1 applies to and validates the generalized Accept–Reject algorithm not only when ϵ_n is constant, but also when the ϵ_n's are periodic in n, and when the sequence $\{\epsilon_n\}$ is uniformly bounded away from 0.

4. Rao–Blackwellization

The output from the generalized Accept–Reject algorithm is as follows: A sequence $Y_1, Y_2, \ldots$ of independent random variables is generated from the g_i's along with a corresponding sequence $U_1, U_2, \ldots$ of uniform random variables. We show how to extend the results of Casella and Robert (1996) to this more general algorithm.

Given a function h, the Accept–Reject estimator of $\mathbb{E}^f\{h(X)\}$, based upon a sample $X_1, \ldots, X_t$, with t fixed, is made of the t accepted values among the Y_j's and is given by

$$\hat{\tau}_1 = \frac{1}{t}\sum_{i=1}^{t} h(X_i) = \frac{1}{t}\sum_{i=1}^{N} \mathbf{I}(U_i \le W_i) h(Y_i), \tag{4}$$

where N, the number of Y_j's generated, is a random integer satisfying

$$\sum_{i=1}^{N} I(U_i \le W_i) = t \qquad \text{and} \qquad \sum_{i=1}^{N-1} I(U_i \le W_i) = t-1,$$

with $W_i = f(Y_i)\epsilon_i/g_i(Y_i)$. By the Rao-Blackwell Theorem, the conditional expectation

$$\hat{\tau}_2 = \frac{1}{t}\mathbb{E}\left\{ \sum_{i=1}^{N} \mathbf{I}(U_i \le W_i) h(Y_i) \Big| N, Y_1, \ldots, Y_N \right\} \tag{5}$$

improves upon (4).

The joint distribution of $(N, Y_1, \ldots, Y_N, U_1, \ldots, U_N)$ is given by

$$\begin{aligned} &P(N = n, Y_1 \le y_1, \ldots, Y_n \le y_n, U_1 \le u_1, \ldots, U_n \le u_n) \\ &\quad = \int_{-\infty}^{y_n} g_n(v_n)(u_n \wedge w_n) dv_n \int_{-\infty}^{y_1} \cdots \int_{-\infty}^{y_{n-1}} g_1(v_1)\ldots g_{n-1}(v_{n-1}) \\ &\qquad \times \sum_{(i_1,\ldots,i_{t-1})} \prod_{j=1}^{t-1} (w_{i_j} \wedge u_{i_j}) \prod_{j=t}^{n-1} (u_{i_j} - w_{i_j})^+ dv_1 \ldots dv_{n-1}, \end{aligned}$$

where $w = \varepsilon f(v)/g(v)$ (with appropriate subscripts) and the last sum is over all subsets of $\{1, \ldots, n-1\}$ of size $t-1$. Therefore, the conditional density of the U_i's

is given by

$$f(u_1,\ldots,u_n|N=n,y_1,\ldots,y_n) = \left\{\sum_{(i_1,\ldots,i_{t-1})}\prod_{j=1}^{t-1} w_{i_j}\prod_{j=t}^{n-1}(1-w_{i_j})\right\}^{-1} \times\left\{\sum_{(i_1,\ldots,i_{t-1})}\prod_{j=1}^{t-1}\mathbf{I}(u_{i_j}\le w_{i_j})\prod_{j=t}^{n-1}\mathbf{I}(u_{i_j}>w_{i_j})\right\}\frac{\mathbf{I}(u_n\le w_n)}{w_n},$$

where, analogously, $w=\varepsilon f(y)/g(y)$. Using this distribution we can calculate, conditional on $(N,y_1,\ldots,y_N)$, the probability ρ_i of the events $\{U_i\le w_i\}$ and thus derive the weights of $h(Y_i)$ in the estimator $\hat{\tau}_2$. The calculations involve averaging over permutations of the realized sample and yield, for $i<n$,

$$\rho_i = w_i\sum_{(i_1,\ldots,i_{t-2})}\prod_{j=1}^{t-2} w_{i_j}\prod_{j=t-1}^{n-2}(1-w_{i_j})\Big/\sum_{(i_1,\ldots,i_{t-1})}\prod_{j=1}^{t-1} w_{i_j}\prod_{j=t}^{n-1}(1-w_{i_j}), \qquad (6)$$

while $\rho_n=1$. The numerator sum is over all subsets of $\{1,\ldots,i-1,i+1,\ldots,n-1\}$ of size $t-2$, and the denominator sum is over all subsets of size $t-1$. The following result therefore holds.

Theorem 4.1. *For $N=n$, the Rao–Blackwellized version of* (4) *is given by*

$$\hat{\tau}_2=\frac{1}{t}\sum_{i=1}^{n}\rho_i h(Y_i)$$

where ρ_i is given by equation (6).

5. Perfect sampling

A *perfect sampling algorithm* for a Markov chain is an algorithm that produces a random variable that is exactly distributed according to the stationary distribution of the Markov chain using variables that are (typically) generated from the conditional distributions of the chain. Perfect sampling in Markov chains originated with the ingenious "coupling from the past" algorithm of Propp and Wilson (1996). In practice, however, this algorithm has some drawbacks, such as – for example – not being interruptible and thus creating biases in the output in cases of interruption for insufficient memory and such.

An alternative, interruptible, perfect sampling algorithm was proposed by Fill (1998). Since it is interruptible, Fill's perfect sampling algorithm seems to be somewhat more practical than coupling from the past, although it requires delicate reversibility and coupling arrangements as shown below.

Fill's algorithm (see also Fill *et al.* 1999) can be described as follows:

(a) Starting at an arbitrary state 0, run a finite state Markov chain (X_i) for t (fixed) steps, and record $X_t=x$.

(b) Starting Markov chains in at all possible states at time t, run them in reversed time, coupled with the original chain.

(c) If all these chains have coalesced, that is, if they all are in state 0 at time 0, then accept $X_t=x$ as an observation from the stationary distribution. If not, reject X_t and start again, possibly with different values of 0 and of t.

We now relate the result of the previous sections to Fill's algorithm.

- The surprising feature of this method is that it is a rejection algorithm with the clever twist that the probability of acceptance is exactly the probability of coalescence. This circumvents the problem of calculating this acceptance probability, which is typically not feasible.
- Fill's (1998) algorithm depends on a parameter t, which is the number of forward steps in the Markov chain and which can be modified at each iteration, by, for instance, doubling the value of t in a typical CFTP manner. Thus, the proposal distribution is changing at every iteration, and the algorithm is covered by Theorem 3.1 (but is not covered by the standard Accept–Reject algorithm).
- For Theorem 3.1 to validate Fill's (1998) algorithm, the series $\sum \log(1-\epsilon_i)$ of acceptance probabilities ϵ_i must diverge. The difficulty then lies in establishing this without the ϵ_i's being available, which is the essence of Fill's technique. However, if the selection is periodic, Fill's algorithm is indeed valid, provided some ϵ_i's are different from 0. In fact, in most practical cases Fill's algorithm will have an increasing acceptance rate, so will be covered by Theorem 3.1.
- The application of Theorem 4.1 to Fill's algorithm requires some further work since, in that case, the weights $w_i = f(x_i)/K^t(0, x_i)$ are not directly available. Note however that in some setups $K^t(0, x)$ may be known, while, in others, it can be estimated, since it is also equal to the probability of acceptance, that is, the probability of coalescence in state 0. Thus, we can implement the Rao–Blackwellized improvement with estimated weights.

References

[1] Casella, G., Lavine, M., and Robert, C.P. (2000). An introduction to perfect sampling. *Amer. Statist.* **55** 299–305. MR1939363

[2] Casella, G., and Robert, C. P. (1996). Rao-Blackwellization of sampling schemes. *Biometrika* **83**(1), 81–94. MR1399157

[3] Dimakos, X. K. (2001). "A Guide to Exact Simulation". *International Statistical Review* **69** 27–48.

[4] Fill, J. A. (1998). An interruptible algorithm for perfect sampling via Markov chains. *Annals of Applied Probability*, **8**(1) 131–162. MR1620346

[5] Fill, J. A., Machida, M., Murdoch, D. J. and Rosenthal, J. S. (1999). Extension of Fill's perfect rejection sampling algorithm to general chains. *Random Structures and Algorithms* **17** 219–316. MR1801136

[6] Gilks, W. R. and Wild, P. (1992). Adaptive rejection sampling for Gibbs sampling. *Appl. Statist.* **41**, 337–348.

[7] Propp, J.G. and Wilson, D. B. (1996). Exact sampling with coupled Markov chains and applications to statistical mechanics. *Random Structures and Algorithms*, **9**(1& 2), 223–252. MR1611693

[8] Robert, C. P. and Casella, G. (1999). *Monte Carlo Statistical Methods.* Springer-Verlag, New York. MR1707311