The Barnoulli Factory, its extensions and applications

Krzysztof Łatuszyński[1]

University of Warwick, Department of Statistics,
CV4 7AL, Coventry, UK,
(e-mail: latuch@gmail.com)

Abstract. The Bernoulli factory problem is to sample a single output of an $f(p)-$coin using a black box generating $p-$coins, where the value of $p \in (0,1)$ is unknown and the function f is known. We sketch the recent reverse time martingale approach to the problem developed in [9] and indicate its applications in computational statistics.

Keywords. Bernoulli factory, perfect sampling, retrospective sampling, Exact Algorithms for diffusions.

1 Introduction

Let $p \in \mathcal{P} \subseteq [0,1]$ be unknown and let $f : \mathcal{P} \to [0,1]$. Then the problem known as the Bernoulli Factory is to generate Y, a single coin toss of an $s = f(p)-$coin, given a sequence $X_1, X_2, ...$ of independent tosses of a $p-$coin. For the historical context of this question and a range of theoretical results see [8,12] and references therein. In particular [8] provides necessary and sufficient conditions for f, under which an algorithm generating an $f(p)-$coin exists. Nacu and Peres in [12] suggest an algorithm for simulating na $f(p)-$coin based on polynomial envelopes of f (see Proposition 1). Ongoing research in Markov chain Monte Carlo and rejection sampling indicates that the Bernoulli factory problem is not only of theoretical interest, c.f. Section 4. To run the Nacu-Peres algorithm one has to deal with sets of exponential size. However, the recent martingale formulation of the Bernoulli Factory problem in [9] allows to use the polynomial envelopes of f in a different way. It performs a random walk on the polynomial coefficients resulting in lower and upper stochastic approximations of $f(p)$ and in practical algorithms.

In Section 2 we sketch the martingale approach of [9] and link it to the Bernoulli Factory in Section 3. In Section 4 we indicate how the Bernoulli Factory problem naturally arises as a crucial step in some algorithms in computational statistics and report on extensions of the Bernoulli Factory that are currently investigated.

2 The Martingale Approach

We shall discuss settings where the amount of information about $s \in [0,1]$ is limited but sampling events of probability s is still possible. All of the subsequent algorithms have emerged in MCMC related simulation problems in our recent work. The idea behind them is the equivalence of unbiased estimation and perfect simulation exploited in [9]. We refer to [9] for proofs of validity and more elaborated description with applications.

Clearly if s is known explicitly, one uses

Algorithm 1
1. simulate $G_0 \sim U(0,1)$;
2. obtain $\hat{S}$;
3. if $G_0 \leq \hat{S}$ *set* $C_s := 1,$ *otherwise set* $C_s := 0$;
4. output C_s.

Next assume that real valued monotone sequences $l_n \nearrow s$ and $u_n \searrow s$ are available.

Algorithm 2
1. simulate $G_0 \sim U(0,1)$; *set* $n = 1$;
2. compute l_n *and* u_n;
3. if $G_0 \leq l_n$ *set* $C_s := 1$;
4. if $G_0 > u_n$ *set* $C_s := 0$;
5. if $l_n < G_0 \leq u_n$ *set* $n := n + 1$ *and GOTO 2;*
6. output C_s.

The next step is to combine the above ideas and work with randomized bounds, i.e. in a setting where we have estimators L_n and U_n of the upper and lower bounds l_n and u_n. The estimators shall live on the same probability space and have the following properties.

$$\mathbb{P}(L_n \leq U_n) = 1 \qquad \text{for every} \quad n = 1, 2, ... \tag{1}$$
$$\mathbb{P}(L_n \in [0,1]) = 1 \text{ and } \mathbb{P}(U_n \in [0,1]) = 1 \quad \text{for every} \quad n = 1, 2, ... \tag{2}$$
$$\mathbb{E}\, L_n = l_n \nearrow s \text{ and } \mathbb{E}\, U_n = u_n \searrow s \tag{3}$$
$$\mathbb{P}(L_{n-1} \leq L_n) = 1 \text{ and } \mathbb{P}(U_{n-1} \geq U_n) = 1 \tag{4}$$

Let $\qquad \mathcal{F}_0 = \{\emptyset, \Omega\}, \qquad \mathcal{F}_n = \sigma\{L_n, U_n\}, \qquad \mathcal{F}_{k,n} = \sigma\{\mathcal{F}_k, \mathcal{F}_{k+1}, ...\mathcal{F}_n\} \quad \text{for } k \leq n.$

Under these assumptions one can use the following algorithm.

Algorithm 3
1. simulate $G_0 \sim U(0,1)$; *set* $n = 1$;
2. obtain L_n *and* U_n *given* $\mathcal{F}_{0,n-1}$,
3. if $G_0 \leq L_n$ *set* $C_s := 1$;
4. if $G_0 > U_n$ *set* $C_s := 0$;
5. if $L_n < G_0 \leq U_n$ *set* $n := n + 1$ *and GOTO 2;*
6. output C_s.

The final step is to weaken condition (4) and let L_n be a reverse time supermartingale and U_n a reverse time submartingale with respect to $\mathcal{F}_{n,\infty}$. Precisely, assume that we have

$$\mathbb{E}\,(L_{n-1} \mid \mathcal{F}_{n,\infty}) \;=\; \mathbb{E}\,(L_{n-1} \mid \mathcal{F}_n) \;\leq\; L_n \quad \text{a.s.} \qquad \text{and} \tag{5}$$
$$\mathbb{E}\,(U_{n-1} \mid \mathcal{F}_{n,\infty}) \;=\; \mathbb{E}\,(U_{n-1} \mid \mathcal{F}_n) \;\geq\; U_n \quad \text{a.s.} \qquad \text{for every } n = 1, 2, ... \tag{6}$$

The following algorithm uses auxiliary random sequences $\tilde{L}_n$ and $\tilde{U}_n$ constructed online.

Algorithm 4
1. simulate $G_0 \sim U(0,1)$; *set* $n = 1$; *set* $L_0 \equiv \tilde{L}_0 \equiv 0$ *and* $U_0 \equiv \tilde{U}_0 \equiv 1$
2. obtain L_n *and* U_n *given* $\mathcal{F}_{0,n-1}$,
3. compute $L_n^* = \mathbb{E}\,(L_{n-1} \mid \mathcal{F}_n)$ *and* $U_n^* = \mathbb{E}\,(U_{n-1} \mid \mathcal{F}_n)$.
4. compute

$$\tilde{L}_n := \tilde{L}_{n-1} + \frac{L_n - L_n^*}{U_n^* - L_n^*}\Big(\tilde{U}_{n-1} - \tilde{L}_{n-1}\Big) \qquad \tilde{U}_n := \tilde{U}_{n-1} - \frac{U_n^* - U_n}{U_n^* - L_n^*}\Big(\tilde{U}_{n-1} - \tilde{L}_{n-1}\Big)$$

5. if $G_0 \leq \tilde{L}_n$ *set* $C_s := 1$;
6. if $G_0 > \tilde{U}_n$ *set* $C_s := 0$;
7. if $\tilde{L}_n < G_0 \leq \tilde{U}_n$ *set* $n := n + 1$ *and GOTO 2;*
8. output C_s.

3 Application to the Bernoulli Factory Problem

Here the following minor modification of Proposition 3 in [12] is reproved in a way that links polynomial envelopes of f with the framework of Section 2 by identifying terms. It results in an immediate application of Algorithm 4 and in practical algorithms. See [9] for more details.

Proposition 1. *An algorithm that simulates a function f on $\mathcal{P} \subseteq (0,1)$ exists if and only if for all $n \geq 1$ there exist polynomials $g_n(p)$ and $h_n(p)$ of the form*

$$g_n(p) = \sum_{k=0}^{n} \binom{n}{k} a(n,k) p^k (1-p)^{n-k} \quad \textit{and} \quad h_n(p) = \sum_{k=0}^{n} \binom{n}{k} b(n,k) p^k (1-p)^{n-k},$$

s.t. (i) $0 \leq a(n,k) \leq b(n,k) \leq 1$, (ii) $\lim_{n\to\infty} g_n(p) = f(p) = \lim_{n\to\infty} h_n(p)$, (iii) For all $m < n$, their coefficients satisfy

$$a(n,k) \geq \sum_{i=0}^{k} \frac{\binom{n-m}{k-i}\binom{m}{i}}{\binom{n}{k}} a(m,i), \qquad b(n,k) \leq \sum_{i=0}^{k} \frac{\binom{n-m}{k-i}\binom{m}{i}}{\binom{n}{k}} b(m,i). \tag{7}$$

Proof. We focus on proving *polynomials* $\Rightarrow$ *algorithm* using framework of Section 2. Let $X_1, X_2, \ldots$ be a sequence of independent tosses of a $p-$coin. Define random sequences $\{L_n, U_n\}_{n\geq 1}$ as follows: if $\sum_{i=1}^{n} X_i = k$, then let $L_n = a(n,k)$ and $U_n = b(n,k)$. In the rest of the proof we check that (1), (2), (3), (5) and (6) hold for $\{L_n, U_n\}_{n\geq 1}$ with $s = f(p)$. Thus executing Algorithm 4 with $\{L_n, U_n\}_{n\geq 1}$ yields a valid $f(p)-$coin.

Clearly (1) and (2) hold due to (i). For (3) note that $\mathbb{E}\ L_n = g_n(p) \nearrow f(p)$ and $\mathbb{E}\ U_n = h_n(p) \searrow f(p)$. To obtain (5) and (6) define the sequence of random variables H_n to be the number of heads in $\{X_1, \ldots, X_n\}$, i.e. $H_n = \sum_{i=1}^{n} X_i$ and let $\mathcal{G}_n = \sigma(H_n)$. Thus $L_n = a(n, H_n)$ and $U_n = b(n, H_n)$, hence $\mathcal{F}_n \subseteq \mathcal{G}_n$ and it is enough to check that $\mathbb{E}(L_m|\mathcal{G}_n) \leq L_n$ and $\mathbb{E}(U_m|\mathcal{G}_n) \geq U_n$ for $m < n$. The distribution of H_m given H_n is hypergeometric and

$$\mathbb{E}(L_m|\mathcal{G}_n) = \mathbb{E}(a(m, H_m)|H_n) = \sum_{i=0}^{H_n} \frac{\binom{n-m}{H_n-i}\binom{m}{i}}{\binom{n}{H_n}} a(m,i) \leq a(n, H_n) = L_n.$$

Clearly the distribution of H_m given H_n is the same as the distribution of H_m given $\{H_n, H_{n+1}, \ldots\}$. The argument for U_n is identical.

4 Closing Remarks

The setting of Section 2, where s is uniquely determined but its exact computation is not possible, has been encountered in various computational contexts. In particular Algorithm 2 has emerged as an element of more complex procedures in Exact Algorithms for diffusions in [4,5,13], and the version of Exact Algorithm introduced in [3] can be obtained as an application of Algorithm 3, c.f. [9]. The Bernoulli Factory has possible applications to regenerative MCMC simulation (c.f. [1], [7], Chapter 16 of [2]). Moreover it appears unavoidable in certain settings of exact MCMC algorithms for parameter inference in diffusions (c.f. [6,10]).

On the technical side, [9] indicates that for some functions with alternating series expansion, there is no need of Algorithm 4 and Algorithm 3 can be applied resulting in a critical efficiency improvement. This has been further investigated in [11] relating applicability of Algorithm 3 to completely monotone functions. Moreover [11] investigates the multivariate version of the Bernoulli Factory problem by extending the martingale approach.

References

[1] Assmussen, S., Glynn P. W. and Thorisson H. (1992): Stationarity detection in the initial transient problem. *ACM Transactions on Modelling and Computer Simulation. 2(2)*, 130-157.

[2] Asmussen S. and Glynn P. W. (2007): *Stochastic simulation: algorithms and analysis.* Springer, New York.

[3] Beskos, A. and Roberts, G.O. (2005): Exact Simulation of Diffusions. *Ann. Appl. Probab. 15(4)*, 2422–2444.

[4] Beskos, A., Papaspiliopoulos, O. and Roberts, G.O. (2008): A factorisation of diffusion measure and finite sample path constructions. *Methodol. Comput. Appl. Probab. 10(1)*, 85–104.

[5] Beskos, A., Papaspiliopoulos, O., Roberts, G.O. and Fearnhead, P. (2006): Exact and computationally effcient likelihood-based estimation for discretely observed diffusion processes (with discussion). *Journal of the Royal Statistical Society: Series B (Statistical Methodology), 68(3)*, 333-382.

[6] Gonçalves, F.B., Łatuszyński, K., Roberts G.O. (2010): Exact MCMC inference for log Gaussian Cox process. *In preparation.*

[7] Henderson S.G. and P. W. Glynn P. W.(2003): Non-existence of a class of variate generation schemes. *Operations Research Letters, 31*, 83–89.

[8] Keane, M.S. and OBrien, G.L. (1994): A Bernoulli factory. *ACM Transactions on Modeling and Computer Simulation (TOMACS), 4(2)*, 213-219.

[9] Łatuszyński, K. , Kosmidos, I., Papaspiliopoulos, O. and Roberts, G. O. (2009): Simulating Events of Unoknown Probabilities via Reverse Time Martingales. *Random Structures and Algorithms, to appear.*

[10] Łatuszyński, K., Palczewski, J. and Roberts G.O. (2010): Exact MCMC inference for a Markov switching diffusion models with discreetly observed data. *In preparation.*

[11] Łatuszyński, K., Papaspiliopoulos, O., Roberts, G. O. and Spano, D. (2010): Beyond the Bernoulli Factory. *In preparation.*

[12] Nacu, S. and Peres, Y. (2005): Fast simulation of new coins from old. *Annals of Applied Probability, 15(1)*, 93–115.

[13] Papaspiliopoulos, O. and Roberts, G.O. (2008): Retrospective Markov chain Monte Carlo for Dirichlet process hierarchical models. *Biometrika, 95*, 169–186.