Combinatorics, Probability and Computing (2016) 25, 577–591. © Cambridge University Press 2016
doi:10.1017/S0963548315000371

Nearly Optimal Bernoulli Factories for Linear Functions

MARK HUBER

Claremont McKenna College, 850 Columbia Avenue, Claremont, CA 91711, USA
(e-mail: mhuber@cmc.edu)

Received 14 June 2013; revised 25 November 2015; first published online 1 February 2016

Suppose that $X_1, X_2, \ldots$ are independent identically distributed Bernoulli random variables with mean p. A Bernoulli factory for a function f takes as input $X_1, X_2, \ldots$ and outputs a random variable that is Bernoulli with mean $f(p)$. A fast algorithm is a function that only depends on the values of $X_1, \ldots, X_T$, where T is a stopping time with small mean. When $f(p)$ is a real analytic function the problem reduces to being able to draw from linear functions Cp for a constant $C > 1$. Also it is necessary that $Cp \leqslant 1 - \epsilon$ for known $\epsilon > 0$. Previous methods for this problem required extensive modification of the algorithm for every value of C and ϵ. These methods did not have explicit bounds on $\mathbb{E}[T]$ as a function of C and ϵ. This paper presents the first Bernoulli factory for $f(p) = Cp$ with bounds on $\mathbb{E}[T]$ as a function of the input parameters. In fact, $\sup_{p \in [0,(1-\epsilon)/C]} \mathbb{E}[T] \leqslant 9.5\epsilon^{-1}C$. In addition, this method is very simple to implement. Furthermore, a lower bound on the average running time of any Cp Bernoulli factory is shown. For $\epsilon \leqslant 1/2$, $\sup_{p \in [0,(1-\epsilon)/C]} \mathbb{E}[T] \geqslant 0.004C\epsilon^{-1}$, so the new method is optimal up to a constant in the running time.

2010 *Mathematics subject classification*: Primary 65C50
Secondary 68Q17

1. Introduction

Let $\mathsf{Bern}(p)$ denote the Bernoulli distribution with mean p, so $X \sim \mathsf{Bern}(p)$ means $\mathbb{P}(X = 1) = p$ and $\mathbb{P}(X = 0) = 1 - p$. A Bernoulli random variable will also be referred to as a coin flip, with heads equating to the value 1 and tails to the value 0. Let $X_1, X_2, \ldots$ be an infinite sequence of independent, identically distributed (i.i.d.) $\mathsf{Bern}(p)$ random variables.

Following [5], a Bernoulli factory is an algorithm that takes independent realizations of $X \sim \mathsf{Bern}(p)$ and auxiliary variables with known distributions, and simulates a Bernoulli variable with success probability $f(p)$. Of course, the algorithm is not allowed to know the value of p. Only draws from $\mathsf{Bern}(p)$ and an independent uniform on $[0, 1]$ are given to the algorithm.

Recall that $U \sim \mathsf{Unif}([0, 1])$ can be viewed as an i.i.d. sequence of $\mathsf{Bern}(1/2)$ random variables simply by reading off the bits in the number U. These bits can then be used to

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

build an i.i.d. sequence of uniform random numbers in $[0,1]$. Hence a single U represents for notational purposes all the extra randomness utilized by any randomized algorithm.

Definition. Given $p^* \in (0,1]$ and a function $f : [0,p^*] \to [0,1]$, a *Bernoulli factory* is a computable function $\mathcal{A}$ that takes as input a number $u \in [0,1]$ together with a sequence of values in $\{0,1\}$, and returns an output in $\{0,1\}$ where the following holds. For any $p \in [0,p^*]$, $X_1, X_2, \ldots$ i.i.d. $\mathsf{Bern}(p)$, and $U \sim \mathsf{Unif}([0,1])$, let T be the infimum of times t such that the value of $\mathcal{A}(U, X_1, X_2, \ldots)$ only depends on the values of $X_1, \ldots, X_t$. Then

(1) T is a stopping time with respect to the natural filtration and $\mathbb{P}(T < \infty) = 1$,
(2) $\mathcal{A}(U, X_1, X_2, \ldots) \sim \mathsf{Bern}(f(p))$.

Call T the *running time* of the Bernoulli factory.

Informally, $\mathcal{A}$ is a Bernoulli factory if it takes T flips of a coin with $p \in [0,p^*)$ chance of heads together with a source of random bits, and returns a single coin flip that has probability $f(p)$ chance of heads.

The applications of Bernoulli factories go back at least as far as von Neumann [9], who showed how to build such a factory for $p^* = 1$ and $f(p) = 1/2$ without using the external randomness U. His method works as follows:

$$T = \inf\{t \in \{2,4,\ldots\} : (X_{t-1}, X_t) \in \{(0,1),(1,0)\}\},$$
$$Y = \mathbf{1}((X_{T-1}, X_T) = (0,1)),$$

where $\mathbf{1}$(expression) denotes the indicator function that evaluates to 1 if the expression is true and 0 otherwise.

In 1992 Asmussen raised the question of whether it was possible to construct a Bernoulli factory for $f(p) = Cp$, the application being perfect sampling for certain positive recurrent regenerative processes [1].

Keane and O'Brien [5] showed that a Bernoulli factory where T is almost surely finite exists if and only if $f(p)$ is continuous over its domain and either $f(p)$ is identically 0 or 1, or both $f(p)$ and $1 - f(p)$ are polynomially bounded away from 0 over the domain. They gave an explicit algorithm for building a Bernoulli factory in such a case using the fact that continuous $f(p)$ can be well approximated by Bernstein polynomials (linear combinations of $p^k(1-p)^{n-k}$ where n and $k \leqslant n$ are integers). However, they did not prove any bounds on $\mathbb{E}[T]$ (upper or lower).

Nacu and Peres further refined the approach of Keane and O'Brien, showing that for $\epsilon > 0$, and $p^* = (1-\epsilon)/2$, the function $f(p) = 2p$ has a Bernoulli factory where T has exponentially declining tails (Theorem 1 of [7]). Moreover, they showed that in a sense $f(p) = 2p$ is the most important case, as a Bernoulli factory for $f(p) = 2p$ can be used to build a Bernoulli factory for any function that is real analytic in $(0,1)$ and bounded away from 1.

Łatuszyński, Kosmidis, Papaspiliopoulos and Roberts [6] gave the first practical implementation of the Nacu–Peres approach. Instead of dealing with sets of outcomes of coins, they created a continuous approach using a sequence of functions L_n that lower-bound (in expectation) $f(p)$, and U_n that upper-bound (in expectation) $f(p)$. L_n is

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

a reverse time supermartingale, while U_n is a reverse time submartingale, both of which monotonically converge to $f(p)$ as the number of coins grows to infinity. This allowed the algorithm to proceed without having to keep track of an (exponentially growing) number of possible sequences of coin flips, and allowed the Nacu–Peres algorithm to be implemented.

Flegal and Herbei [4] developed a faster algorithm by changing $f(p)$. In the Nacu–Peres algorithm, instead of restricting themselves to $Cp \leqslant 1-\epsilon$, they used $f(p) = \min\{Cp, 1-\epsilon\}$ so that the function was defined over the entirety of $[0,1]$. Of course this makes the derivative of $f(p)$ discontinuous at $p = (1-\epsilon)/C$. The idea of Flegal and Herbei was to handle the $p \in [(1-\epsilon)/C, 1]$ situation differently, in such a way that $f(p)$ became twice differentiable at $p = (1-\epsilon)/C$. They then employed the reverse time martingale method of [6] to perform the simulations.

Experimental results in [4] indicated that this greatly improved upon the running time, although the running time still appeared to be superlinear in C.

With the positive recurrent regenerative process application, it makes sense to measure the running time of the algorithm in terms of the number of coin flips, since for applications flipping a single coin involves running a Markov chain for a large number of steps. In other words, the coin flips dominate the time needed to compute $\mathcal{A}$ by a large factor.

Thomas and Blanchet [8] gave a different improvement to the Nacu–Peres algorithm, but did not analyse the running time. Their numerical experiments indicate that their algorithm also does not scale linearly with C. As with the other approaches mentioned, this was a variation of the original Keane and O'Brien method, which relies on approximating the $f(p)$ function using Bernstein polynomials.

The novel approach presented here is very different. Throughout the rest of this work, assume that $f(p) = Cp$, where $C > 1$, $p < (1-\epsilon)/C$, and $\epsilon > 0$. The new method operates by flipping coins or utilizing extra randomness to change the target function $f(p)$ from step to step. Moreover, it has a form that allows C and ϵ to change without requiring extensive calculation of new parameter values. This allows for the first time a direct computation of an upper bound for $\mathbb{E}[T]$.

Theorem 1.1. *The algorithm of Section 2 is a Bernoulli factory with running time*

$$\sup_{p\in[0,(1-\epsilon)/C]} \mathbb{E}[T] \leqslant 9.5\epsilon^{-1}C.$$

(Theorem 3.6 in Section 3 gives a more precise upper bound on $\mathbb{E}[T]$.) This new approach has the correct order of running time, as shown by the following lower bound on the average number of flips.

Theorem 1.2. *Any Bernoulli factory has*

$$\sup_{p\in[0,(1-\epsilon)/C]} \mathbb{E}[T] \geqslant (3/16)e^{-2}\ln(7)C\epsilon^{-1}(1-\epsilon^{-1/2})^2(1-(1-\epsilon)/C).$$

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

The remainder of the paper is organized as follows. Section 2 presents the new method and proves correctness. Section 3 proves the precise upper bound on $\mathbb{E}[T]$ and compares this upper bound to simulations of the algorithm. Then Section 4 proves the lower bound on $\mathbb{E}[T]$.

2. Algorithm

This section is organized as follows. First the intuition of the new method is presented. Next the pseudocode is given based on the intuition, and finally the rigorous details are given.

2.1. Intuition

Suppose that instead of flipping one coin, first flip $B \sim \mathsf{Bern}(p)$, and then another Bernoulli Y whose mean depends on B. In particular, $[Y|B=1] \sim \mathsf{Bern}(p_1)$, and $[Y|B=0] = \mathsf{Bern}(p_0)$. Then

$$\mathbb{P}(Y=1) = pp_1 + (1-p)p_0.$$

Our goal is to make $\mathbb{P}(Y=1) = Cp$ where $C > 1$. Let $p_1 = 1$, then to make this work $p_0 = (C-1)p/(1-p)$ so that

$$p(1) + (1-p)(C-1)p/(1-p) = Cp.$$

At this point the problem of flipping a Cp-coin has been reduced to flipping a $(C-1)p/(1-p)$-coin. Now employ a trick in Nacu and Peres [7] used there to reduce a Bernoulli factory problem for real analytic $f(p)$ to a Bernoulli factory for $2p$. The trick is to write $(C-1)p/(1-p)$ using the power series expansion.

$$(C-1)\frac{p}{1-p} = (C-1)p + (C-1)p^2 + (C-1)p^3 + \cdots.$$

Let $G \sim \mathsf{Geo}(a)$ denote the geometric distribution, so for positive integers g,

$$\mathbb{P}(G=g) = (1-a)^{g-1}a.$$

Let $G \sim \mathsf{Geo}((C-1)/C)$. Then let $[W|G] \sim \mathsf{Bern}((Cp)^G)$. Then

$$\mathbb{P}(W=1) = \sum_{i=1}^{\infty} \mathbb{P}(G=i)(Cp)^i = \sum_{i=1}^{\infty} \left(\frac{1}{C}\right)^{i-1} \left(\frac{C-1}{C}\right)(Cp)^i = \sum_{i=1}^{\infty} (C-1)p^i.$$

So the problem has been reduced from generating a $(C-1)p/(1-p)$ coin to generating some positive number of Cp-coins. That is, it is necessary to flip k independent (Cp)-coins, or equivalently, a single $(Cp)^k$-coin for some k.

If k is sufficiently large, say $k > (1-Cp)^{-1} > \epsilon^{-1}$, then $(Cp)^k \leqslant e^{-1}$. So there is at least a one-half chance that the coin flip will result in a 0. Suppose that $R \sim \mathsf{Bern}(1/2)$, and that $[Z|R=1] \sim \mathsf{Bern}(0)$, and $[Z|R=0] \sim \mathsf{Bern}(r_0)$. Then

$$\mathbb{P}(Z=1) = (1/2)(0) + (1/2)r_0,$$

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

LINEAR_BERNOULLI_FACTORY *Input:* C, $\mathsf{Bern}(p)$, ϵ

$\gamma \leftarrow 1/2,\ k \leftarrow 2.3(\gamma\epsilon)^{-1},\ i \leftarrow 1,\ \epsilon \leftarrow \min\{\epsilon, 0.644\}$
Repeat
  Repeat
    Draw $B \leftarrow \mathsf{Bern}(p)$, $G \leftarrow \mathsf{Geo}((C-1)/C)$
    $i \leftarrow i - 1 + (1-B)G$
  Until $i = 0$ or $i \geqslant k$
  If $i \geqslant k$
    Draw $R \leftarrow \mathsf{Bern}((1+\gamma\epsilon)^{-i})$
    $C \leftarrow C(1+\gamma\epsilon)$, $\epsilon \leftarrow (1-\gamma)\epsilon$, $k \leftarrow k/(1-\gamma)$.
Until $i = 0$ or $R = 0$.
Return $\mathbf{1}(i = 0)$

so to make $\mathbb{P}(Z = 1) = (Cp)^{100}$, $r_0 = 2(Cp)^{100} = (2^{1/100}Cp)^{100}$. In other words, in this case it is necessary to simulate 100 coins from the Bernoulli factory $(2^{1/100}C)p$. Note that $2^{1/100}$ is very close to 1, so $2^{1/100}C$ is not too far away from the original C.

2.2. Pseudocode

There are three phases that the algorithm goes through. A single p-coin flip gets us started, either moving straight to heads or to a $(C-1)p/(1-p)$-coin. A draw of a geometric random variable then allows us to reduce the $(C-1)p/(1-p)$-coin to a $(Cp)^i$-coin for some i. That of course is the same as flipping a Cp-coin followed by a $(Cp)^{i-1}$-coin. Flip the Cp-coin in the same way, to either get a head or get a $(Cp)^{j+1}$-coin, which combines with the remaining coins to give a $(Cp)^{j+i}$-coin.

Always after the p-coin flip and the Bernoulli, the remaining task is to flip a $(Cp)^i$-coin for some integer k. Hopefully $i = 0$ at some point, in which case the answer is heads since $(Cp)^0 = 1$. On the other hand, if i gets too large ($4.6\epsilon^{-1}$ to be precise), flip a $(1+\epsilon/2)^{-1}$-coin. If that coin comes up 0, return the answer 0, otherwise the new goal is to flip a $(C(1+\epsilon/2)p)^i$-coin. Note that $C(1+\epsilon/2)p \leqslant 1-\epsilon/2$, so our bound away from 1 has shrunk by a factor of 2.

This procedure is encapsulated in algorithm LINEAR_BERNOULLI_FACTORY.

2.3. Details

To prove the algorithm works, begin by building a stochastic process $M_0, M_1, M_2, \ldots$ where M_t represents the mean of the Bernoulli to be drawn. Begin with $M_0 = Cp$.

Lemma 2.1. *Let $M_0, M_1, \ldots$ be a martingale on $[0,1]$ with $M_0 = \theta$. Let $T(\theta) = \inf\{t : M_t \in \{0,1\}\}$. If $\mathbb{P}(T(\theta) < \infty) = 1$, then*

$$M_{T(\theta)} \sim \mathsf{Bern}(\theta).$$

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

Proof. Since M_t is a bounded (and hence uniformly integrable) martingale, the Optional Stopping Theorem applies (see for instance [3, p. 269]) and $\mathbb{E}[M_{T(\theta)}] = \mathbb{E}[M_0] = \theta$. Since

$$\mathbb{P}(T(\theta) < \infty) = 1,$$

$M_{T(\theta)}$ is a $\{0,1\}$ random variable with mean θ, so $M_{T(\theta)} \sim \mathsf{Bern}(\theta)$. □

The next lemma presents the moves the martingale will take.

Lemma 2.2. *Consider a stochastic process $\{M_k : k \geqslant 0\}$ satisfying the following. For each t, given $M_0, \ldots, M_{t-1}$, the value of M_t is chosen according to either the transition kernel $\mathbf{K}_{1,C}(\cdot)$ or the transition kernel $K_{2,\alpha}$. The kernel $K_{1,C}$ is defined as*

$$\mathbf{K}_{1,C}((Cp)^i, (Cp)^{i-1}) = p,$$
$$\mathbf{K}_{1,C}((Cp)^i, (Cp)^{i+j}) = (1-p)(C-1)C^{-j-1}, \quad \text{where } j \in \{0,1,2,\ldots\}.$$

(This kernel can only be applied when M_{t-1} is of the form $(Cp)^i$ for some $i \in \{1,2,\ldots\}$.)

The kernel $K_{2,\alpha}$ is defined (for $\alpha > 1$ and $0 < x \leqslant 1/\alpha$) as

$$\mathbf{K}_{2,\alpha}(x, \alpha x) = 1/\alpha,$$
$$\mathbf{K}_{2,\alpha}(x, 0) = 1 - 1/\alpha.$$

Then $\{M_t\}$ is a martingale with respect to the natural filtration.

Proof. Let $\mathcal{F}_i$ be the σ-algebra generated by $M_0, \ldots, M_i$. To show the result, check that $\mathbb{E}[M_t | \mathcal{F}_{t-1}] = M_{t-1}$. If kernel $\mathbf{K}_{1,C}$ is used,

$$\begin{aligned}
\mathbb{E}[M_t | M_{t-1} = (Cp)^i, \mathcal{F}_{t-2}] &= p(Cp)^{i-1} + (1-p)\sum_{j=0}^{\infty}(Cp)^{i+j}(C-1)(1/C)^{j+1} \\
&= (Cp)^i\left[C^{-1} + (1-p)(C-1)C^{-1}\sum_{j=0}^{\infty}p^j\right] \\
&= (Cp)^i[C^{-1} + (C-1)C^{-1}] \\
&= (Cp)^i = M_{t-1}.
\end{aligned}$$

When $\mathcal{F}_{t-1}$ leads to the use of a $\mathbf{K}_{2,\alpha}$ kernel,

$$\mathbb{E}[M_t | M_{t-1} = x, \mathcal{F}_{t-2}] = (1/\alpha)\alpha x + 0(1 - 1/\alpha) = x = M_{t-1}.$$ □

Taking a step in a $\mathbf{K}_{1,C}$ chain requires the generation of one $\mathsf{Bern}(p)$, and generation of one geometric random variable with mean $C/(C-1)$ if the Bernoulli is 0. Taking a step in any of the α chains requires no $\mathsf{Bern}(p)$ flip, only a single $\mathsf{Bern}(1/\alpha)$.

Now everything is in place to solve the original problem of drawing a $Cp = (Cp)^1$-coin. Given C and α, fix $\gamma \in (0,1)$ and $k > 0$ as

$$\gamma = 1/2, \; k = 2.3(\gamma\epsilon)^{-1}.$$

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

(Any choice of $\gamma \in (0,1)$ gives a correct algorithm; using $\gamma = 1/2$ will give the running time bound in Theorem 1.1.) Start with $M_0 = Cp$ and take steps using the $\mathbf{K}_{1,C}$ kernel until the state is either $(Cp)^0$ or $(Cp)^i$ where $i \geqslant k$.

At this point, let $\alpha = (1+\gamma\epsilon)^i$, and take one step in $\mathbf{K}_{2,\alpha}$. The result is that either the state moves to 0 (in which case return 0 and quit) or the state moves to $(C(1+\gamma\epsilon)p)^i$. Now start taking moves in $\mathbf{K}_{1,C(1+\gamma\epsilon)}$ until either $(C(1+\gamma)p)^0$ or $(C(1+\gamma\epsilon)p)^i$ with $i \geqslant k/(1-\gamma)$ is reached.

Keep repeating this process, taking away a fraction γ of ϵ each time and multiplying k by $(1-\gamma)^{-1}$ to compensate at each step until either a 0 coin state is reached, or the $\mathbf{K}_\alpha$ chain returns 0, at which point terminate.

At each step in the algorithm the parameter of the coin being flipped changes according to one of the Markov chains from Lemma 2.2, and so forms a martingale. In light of Theorem 2.1, to prove the algorithm is correct requires only that $\mathbb{P}(T < \infty) = 1$, where T is the number of calls to $\mathsf{Bern}(p)$. In fact, in the next section a much stronger result, a bound on $\mathbb{E}[T]$, is shown.

Note that the $\mathbf{K}_{1,C}$ and $\mathbf{K}_{2,\alpha}$ kernels given here are not the only transitions possible. There are an uncountable number of ways to build such a martingale. As a simple example, for $C > 2$ you could flip two p-coins initially instead of just one and then work from there. The choices made here are for the simplicity of the algorithm and the ability to bound the expected running time, as shown in the next section.

3. Upper bound on running time

Line 4 is the only line in LINEAR_BERNOULLI_FACTORY where a call to $\mathsf{Bern}(p)$ is made. This line is inside a repeat loop from lines 3–6. To bound the average number of times this repeat loop is run, let i_t denote the value of i at line 5 after this line has been executed t times (let $i_0 = 1$). From Lemma 2.2, $(Cp)^{i_t}$ is a martingale. Let $\tau = \inf\{t : i_t = 0 \text{ or } i_t \geqslant k\}$. Since $i_{t\wedge\tau}$ is a Markov chain with transient states $\{1, 2, \ldots, \lceil k \rceil - 1\}$, $\mathbb{P}(\tau < \infty) = 1$. So i_τ is either 0 or at least k.

Lemma 3.1. *Let* $p_k = \mathbb{P}(i_\tau \geqslant k)$. *Then*

$$p_k \leqslant \frac{1-Cp}{1-(Cp)^k}.$$

Proof. Since $(Cp)^{i_t}$ is a martingale, so is $(Cp)^{i_{t\wedge\tau}}$. So $\mathbb{E}[(Cp)^{i_{t\wedge\tau}}] = Cp$, and

$$\begin{aligned} Cp &= 1 \cdot \mathbb{P}(i_{t\wedge\tau} = 0) + \mathbb{P}(i_{t\wedge\tau} \geqslant k)\mathbb{E}[(Cp)^i_{t\wedge\tau} | i_{t\wedge\tau} \geqslant k] + \mathbb{P}(t < \tau)\mathbb{E}[(Cp)^{i_t} | t < \tau] \\ &\leqslant \mathbb{P}(i_{t\wedge\tau} = 0) + (Cp)^k \mathbb{P}(i_{t\wedge\tau} \geqslant k) + \mathbb{P}(t < \tau)(Cp). \end{aligned}$$

Since this inequality holds for all t and involves probabilities (which are just expectations of bounded random variables), taking the limit as t goes to infinity gives

$$Cp \leqslant [1 - \mathbb{P}(i_\tau \geqslant k)] + (Cp)^k \mathbb{P}(i_\tau \geqslant k),$$

and solving for $p_k = \mathbb{P}(i_\tau \geqslant k)$ gives the result. □

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

Lemma 3.2. *For $\{i_t\}$ and τ as before,*

$$\mathbb{E}[\tau] \leqslant \frac{k(C-1)+C}{1-(Cp)^k} - \frac{C-1}{1-Cp}.$$

Proof. When $0 < i_t < k$,

$$\mathbb{E}[i_t | i_{t-1}, \ldots, i_0] = i_{t-1} - 1 + (1-p)\frac{C}{C-1} = i_{t-1} + \frac{1-Cp}{C-1}.$$

Hence

$$i'_t = i_{t\wedge\tau} - [(1-Cp)/(C-1)](t \wedge \tau)$$

is a martingale. Since $i'_0 = i_{0\wedge\tau} = 1 = \mathbb{E}[i'_t]$, this gives

$$1 = \mathbb{E}[i'_t] = \mathbb{E}[i'_\tau \cdot \mathbf{1}(\tau \leqslant t)] + \mathbb{E}[i'_t \cdot \mathbf{1}(\tau > t)].$$

Each of these terms can be bounded:

$$\begin{aligned}\mathbb{E}[i'_\tau \cdot \mathbf{1}(\tau \leqslant t)] &= \mathbb{E}[(i_\tau - [(1-Cp)/(C-1)]\tau) \cdot \mathbf{1}(\tau \leqslant t)],\\ \mathbb{E}[i'_t \mathbf{1}(\tau > t)] &\leqslant \mathbb{E}[i_t \mathbf{1}(\tau > t)].\end{aligned}$$

Using $\mathbb{E}[i_\tau \mathbf{1}(\tau \leqslant t)] \leqslant \mathbb{E}[i_\tau]$ gives

$$1 \leqslant \mathbb{E}[i_\tau] - [(1-Cp)/(C-1)]\mathbb{E}[\tau \mathbf{1}(\tau \leqslant t)] + \mathbb{E}[i_t \mathbf{1}(\tau > t)]. \tag{3.1}$$

Since $i_{t\wedge\tau}$ is a Markov chain where the states between 0 and k are transient, $\mathbb{E}[\tau]$ is finite. Hence $\mathbb{E}[\tau \mathbf{1}(\tau \leqslant t)] \leqslant \mathbb{E}[\tau] < \infty$ for all t.

On the last step before $t = \tau$, i_t either moves to 0 or increases by a geometric random variable with mean $C/(C-1)$, bringing the bound to $(k + C/(C-1))\mathbb{P}(i_\tau \geqslant k)$. Hence

$$\begin{aligned}\mathbb{E}[i_\tau] &= \mathbb{E}[i_\tau | i_\tau \geqslant k]\mathbb{P}(i_\tau \geqslant k) + 0 \cdot \mathbb{P}(i_\tau = 0)\\ &\leqslant [k + C/(C-1)](1-Cp)/(1-(Cp)^k),\end{aligned}$$

using $\mathbb{P}(i_\tau \geqslant k) \leqslant (1-Cp)/(1-(Cp)^k)$ from the previous lemma.

Now consider $\mathbb{E}[i_t \mathbf{1}(t < \tau)]$. This is bounded as well because $i_t \mathbf{1}(t < \tau)$ must lie in $(0, k)$. Therefore the Lebesgue-dominated convergence theorem allows us to take the limit as $t \to \infty$ inside the expectations, giving

$$1 \leqslant [k + C/(C-1)](1-Cp)/(1-(Cp)^k) - [(1-Cp)/(C-1)]\mathbb{E}[\tau].$$

So

$$\mathbb{E}[\tau] \leqslant [k(C-1)+C]/(1-(Cp)^k) - (C-1)/(1-Cp). \qquad \square$$

The previous lemma bounds the expected number of flips on the first pass through lines 3–6, but this is embedded in a larger repeat loop in lines 2–10. Let T_i denote the number of flips on pass i through the larger repeat loop (so $T_1 = \tau$) if this pass actually occurs. Consider the second pass where $i_{2,t}$ is the value of i after t times through lines 3–6 in the second pass through 2–10. (Set $i_{2,0} = i_\tau$ to be the ending value of first pass.) For the

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

second pass to run, $i_{2,0} \geqslant k$, but now the parameter value is $C(1+\gamma\epsilon)p$, and the $i_{2,t}$ process stops when it is either 0 or at least $k/(1-\gamma)$. Let $\tau_2 = \inf\{t : i_{2,t} = 0 \text{ or } i_{2,t} \geqslant k/(1-\gamma)\}$.

Lemma 3.3. *Let $k_2 = k/(1-\gamma)$ and $C_2 = C(1+\gamma\epsilon)$. Then*

$$\mathbb{E}[T_2] \leqslant [\gamma k_2(C_2-1) + C_2](1-C_2p)^{-1}.$$

Proof. Since $i_{2,t\wedge\tau_2}$ is a Markov chain where all states between 0 and $k_2 = k/(1-\gamma)$ are transient, τ_2 is finite and has bounded expectation. As in Lemma 3.2,

$$i'_t = i_{2,t\wedge\tau_2} - [(1-C_2p)/(C_2-1)](t\wedge\tau)$$

is a martingale, so

$$\mathbb{E}[i_{2,t\wedge\tau_2}] - [(1-C_2p)/(C_2-1)]\mathbb{E}(t\wedge\tau) = i_{2,0} \geqslant k.$$

Any step that puts $i_{2,t}$ past $k/(1-\gamma)$ has expected value $C_2/(C_2-1)$, so

$$\mathbb{E}[i_{2,t\wedge\tau_2}] \leqslant k(1-\gamma)^{-1} + C_2/(C_2-1).$$

Hence

$$\mathbb{E}(t\wedge\tau) \leqslant [k[(1-\gamma)^{-1}-1] + C_2/(C_2-1)](C_2-1)/(1-C_2p).$$

This holds for all t, and since $t\wedge\tau$ is an increasing random variable that converges to τ with probability 1, the monotone convergence theorem gives that it converges to τ as well. Using

$$k[(1-\gamma)^{-1}-1] = \gamma k(1-\gamma)^{-1} = \gamma k_2$$

finishes the proof. □

Now generalize. Let T_j be the number of flips in the jth stage, where $i_{j,t}$ reaches 0 or $k/(1-\gamma)^{j-1}$ in τ_j time.

Lemma 3.4. *Let $k_j = k/(1-\gamma)^{j-1}$. Set $C_1 = C$ and for $j \geqslant 2$ make*

$$C_j = C_{j-1}(1+\gamma(1-\gamma)^{j-2}\epsilon)^{-1}.$$

Then

$$\mathbb{E}[T_j] = [\gamma k_j(C_j-1) + C_j](1-C_jp)^{-1}.$$

Proof. Use induction. The previous lemma is the base case $j = 2$, and the induction step is the same as in the proof of the previous lemma. □

Fortunately, the algorithm is unlikely to reach stage j for large j.

Lemma 3.5. *The chance that stage j is reached is at most*

$$\exp(-(j-1)\gamma\epsilon k)(1-Cp)(1-(Cp)^k)^{-1}.$$

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

Proof. From Lemma 3.1 there is at most a $(1-Cp)/(1-(Cp)^k)$ chance that $i \geqslant k$ at the end of the first stage. Then in order for the second stage to run, R must equal 1, which happens with probability

$$(1+\gamma\epsilon)^{-i} \leqslant \exp(-i\gamma\epsilon) \leqslant \exp(-\gamma\epsilon k).$$

At the next stage ϵ has been multiplied by a factor of $1-\gamma$, but k has been multiplied by a factor of $(1-\gamma)^{-1}$, so $k_j\epsilon_j$ is constant at all stages. Hence at stage $\ell \geqslant 3$, the chance of making it to stage $\ell+1$ is at most $\exp(-\gamma\epsilon k)$. There are $j-1$ stages to pass through to stage j, so the chance of making it to stage j is at most

$$\exp(-(j-1)\gamma\epsilon k)(1-Cp)(1-(Cp)^k)^{-1}. \qquad \square$$

Theorem 3.6. *For $k = 2.3(\gamma\epsilon)^{-1}$, the total expected number of coin flips in the algorithm is bounded above by*

$$\mathbb{E}[T] \leqslant \frac{k(C-1)+C}{1-(Cp)^k} - \frac{C-1}{1-Cp} + \frac{r[\gamma k(C(1-\epsilon)^{-1}-1)+(1-\gamma)^2C(1-\epsilon)^{-1})](1-r)^{-1}}{1-(Cp)^k},$$

where $r = \exp(-k\epsilon\gamma)(1-\gamma)^{-2}$.

Before proving this theorem, it will be helpful to know how $1-C_jp$ behaves. In essence, each time the algorithm advances to the next stage, $1-C_jp$ is at least $(1-C_{j-1}p)(1-\gamma)$.

Lemma 3.7.

$$1-C_jp \geqslant (1-Cp)(1-\gamma)^{j-1}.$$

Proof. Proceed by induction. In the base case $j=1$ both sides are equal since $C_1 = C$. Suppose the lemma holds for C_j and consider C_{j+1}.

For $x \in (0,1)$ and $\alpha > 1$,

$$1-\alpha x \geqslant (1-x)(1-\gamma) \Leftrightarrow \alpha \leqslant \frac{1-(1-x)(1-\gamma)}{x}.$$

Set $x = C_jp$ and $\alpha = C_{j+1}p/(C_jp) = (1+\gamma(1-\gamma)^{j-1}\epsilon)$. The induction hypothesis upper-bounds the numerator and lower-bounds the denominator:

$$\frac{1-(1-x)(1-\gamma)}{x} = \frac{1-(1-C_jp)(1-\gamma)}{x} \leqslant \frac{1-(1-Cp)(1-\gamma)^j}{1-(1-Cp)(1-\gamma)^{j-1}}.$$

Let $y = (1-Cp)(1-\gamma)^{j-1}$. Since

$$\frac{1-(1-\gamma)y}{1-y} = 1 + \frac{\gamma y}{1-y},$$

the question reduces to showing $\alpha = 1+\gamma(1-\gamma)^{j-1}\epsilon \leqslant 1+\gamma y/(1-y)$. Since $y/(1-y)$ is an increasing function, and $y \geqslant \epsilon(1-\gamma)^{j-1}$, this will be true if and only if

$$(1-\gamma)^{j-1}\epsilon \leqslant \frac{(1-\gamma)^{j-1}\epsilon}{1-(1-\gamma)^{j-1}\epsilon},$$

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

which is true for $\gamma \in (0,1)$. □

Now a stronger statement can be shown.

Lemma 3.8.

$$1 - C_{j+1}p \geqslant (1 - C_j p)(1 - \gamma).$$

Proof. Using

$$1 - \alpha x \geqslant (1-x)(1-\gamma) \Leftrightarrow \alpha \leqslant (1 - (1-x)(1-\gamma))x^{-1}$$

for $x \in (0,1)$ and $\alpha > 1$ as in the last lemma, let $x = C_j p$ and

$$\alpha = C_{j+1}p/(C_j p) = (1 + \gamma(1-\gamma)^{j-1}\epsilon)$$

as before. For $y = 1 - C_j p$, the goal is to show

$$\alpha \leqslant \frac{1-(1-x)(1-\gamma)}{x} \leqslant \frac{1-(1-C_j p)(1-\gamma)}{1-(1-C_j p)} \leqslant 1 + \frac{\gamma y}{1-y},$$

which is true if and only if

$$1 + \gamma(1-\gamma)^{j-1}\epsilon \leqslant 1 + \gamma y/(1-y).$$

As $y/(1-y)$ is increasing, this is true if and only if it is true for the smallest value of y. By the previous lemma,

$$y = (1 - C_j p) \geqslant (1 - Cp)(1-\gamma)^{j-1} \geqslant \epsilon(1-\gamma)^{j-1}.$$

This gives the result. □

Proof of Theorem 3.6. By the monotone convergence theorem, the total expected number of flips is just the expected number of flips executed at each stage. By Lemma 3.5 the chance of reaching stage j is

$$\exp(-(j-1)\gamma\epsilon k)(1 - Cp)(1 - (Cp)^k)^{-1}.$$

Hence

$$\mathbb{E}[T] \leqslant \mathbb{E}[\tau] + \frac{1-Cp}{1-(Cp)^k} \sum_{j=2}^{\infty} \exp(-(j-1)\gamma\epsilon k)\mathbb{E}[T_j].$$

Since $1 - C_j p \leqslant 1$, $C_j \leqslant C_\epsilon \stackrel{\text{def}}{=} C/(1-\epsilon)$ for all j. So for all j,

$$\mathbb{E}[T_j] \leqslant w_j \stackrel{\text{def}}{=} [\gamma(1-\gamma)^{-j+1}k(C_\epsilon - 1) + C_\epsilon](1 - C_j p)^{-1}.$$

By the previous lemma,

$$\frac{w_{j+1}}{w_j} = \frac{\gamma(1-\gamma)^{-j}k(C_\epsilon - 1) + C_\epsilon}{\gamma(1-\gamma)^{-j+1}k(C_\epsilon - 1) + C_\epsilon} \cdot \frac{1 - C_j p}{1 - C_{j+1}p} \leqslant (1-\gamma)^{-2}.$$

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

Therefore

$$\mathbb{E}[T] \leqslant \mathbb{E}[\tau] + \frac{1-Cp}{1-(Cp)^k} \cdot \frac{w_2 \exp(-\gamma\epsilon k)}{1-\exp(-\gamma\epsilon k)(1-\gamma)^{-2}},$$

as long as $r \stackrel{\text{def}}{=} \exp(-\gamma\epsilon k)(1-\gamma)^{-2} < 1$.

Using

$$(1-Cp)(1-C(1+\gamma\epsilon)p)^{-1} \leqslant (1-\gamma)^{-1}$$

gives

$$(1-Cp)w_2 \leqslant (1-\gamma)^{-1}[\gamma(1-\gamma)^{-1}k(C_\epsilon - 1) + C_\epsilon] = (1-\gamma)^{-2}[\gamma k(C_\epsilon - 1) + (1-\gamma)^2 C_\epsilon].$$

Hence

$$\mathbb{E}[T] \leqslant \mathbb{E}[\tau] + (1-(Cp)^k)^{-1} \frac{r}{1-r}[\gamma k(C_\epsilon - 1) + (1-\gamma)^2 C_\epsilon].$$

Lemma 3.2 then gives the result. □

Proof of Theorem 1.1. Theorem 3.6 holds for any γ and ϵ in $(0,1)$, as long as k makes $r < 1$. This, however, is very difficult to optimize for general C and ϵ. Since $C = 2$ is so important to the general analytic case, consider optimal values for this case. Since $\exp(-\gamma k\epsilon)$ must be small, k will always contain an ϵ^{-1} factor. Writing k as $k = (\gamma\epsilon)^{-1}m$ gives $\exp(-\gamma k\epsilon) = \exp(-m)$.

Optimizing the bound over m and γ gives (when $\epsilon = 0.2$) $m \approx 2.3$ and $\gamma \approx 1/2$. So

$$k = 4.6\epsilon^{-1}, \quad r = \exp(-2.3)(1-1/2)^{-2} = 0.4010354\cdots.$$

Note that $(Cp)^k \leqslant (1-\epsilon)^{4.6/\epsilon} \leqslant \exp(-4.6)$, so $1 - Cp \leqslant 1 - (Cp)^k \geqslant 0.99$.

Putting this in the bound from Theorem 3.6 gives

$$\mathbb{E}[T] \leqslant \frac{4.6\epsilon^{-1}(C-1)+1}{0.99} + \frac{0.4011/0.99}{1-0.4011}\left[2.3\epsilon^{-1}\left(\frac{C}{1-\epsilon} - 1\right) + \frac{C/4}{1-\epsilon}\right].$$

When $\epsilon \geqslant 0.644$, the algorithm can be run with $\epsilon = 0.644$, which makes

$$(1 - \min\{0.644, \epsilon\})^{-1} \leqslant 2.809$$

and the overall bound at most $9.5C\epsilon^{-1}$. □

Of course, given Theorem 3.6, a better way to run line 1 in the algorithm is given C and ϵ, choose m and γ (and then $k = m(\gamma\epsilon)^{-1}$) to minimize the expected running time bound.

In [8], computer experiments were run to determine the effectiveness of several Bernoulli factories. Their method, the Thomas–Blanchet cascade approach, proved to be the fastest of the computer experiments.

Table 1 includes the data from [8] and adds three columns: the optimal (m, γ) values given C and ϵ, a theoretical bound for the expected number of flips using Theorem 3.6, and an experimental bound found through computer experiments simulating the coin 10 000 times. The value of ϵ was 0.2 throughout. The results are reported as (mean, standard deviation). (Note that m^* and γ^* are the near-optimal m and γ values for a particular

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

Table 1. Comparison of number of flips used by Bernoulli factories. The 'Theory bound' column comes from Theorem 3.6. The 'Exper.' column gives the experimental expected value found by running the algorithm 10 000 times, and the Thomas–Blanchet results come from [8]. The simulation results are given as (sample mean, sample standard deviation).

C	(m^*, γ^*)	Theory bound (new method)	Experiment (new method)	Thomas–Blanchet
2	(2.31, 0.463)	35.56	(28, 43)	(66, 512)
5	(2.01, 0.425)	133.7	(107, 62)	(246, 1215)
10	(1.91, 0.410)	296.9	(239, 140)	(614, 1851)
20	(1.81, 0.394)	623.2	(516, 426)	(1410, 3047)

value of C.) The results show not only a lower mean for the algorithm of Section 2 but a much reduced standard deviation as well.

In addition, the results show that the optimal k and γ values are relatively insensitive to the value of C, and that the precise bound of Theorem 3.6 is relatively close to the true behaviour of the algorithm.

4. Lower bound on running time

In order to establish the lower bound, it is helpful to consider the more general problem of developing a randomized algorithm for estimating the value of p.

Definition. Given $p^* \in (0, 1]$ an *estimation algorithm* $\mathcal{B}$ is a computable function that takes input $U \sim \mathsf{Unif}([0, 1])$ together with $X_1, X_2, \ldots \sim \mathsf{Bern}(p)$ i.i.d. and returns $\hat{p} \in [0, 1]$. Let $T_{\mathcal{B}}$ be the infimum of times t such that the value of $\hat{p}$ only depends on $X_1, \ldots, X_t$. Call $T_{\mathcal{B}}$ the *running time* of the algorithm.

The goal is to make $\hat{p}$ an accurate estimate of p. The definition below comes from [2].

Definition. Call $\mathcal{B}$ with output $\hat{p}$ an (α, β)-*approximation algorithm* if

$$\mathbb{P}(p(1-\alpha) \leqslant \hat{p} \leqslant p(1+\alpha)) \geqslant 1 - \beta.$$

Dagum, Karp, Luby and Ross [2] lower-bounded the expected running time of any such algorithm $\mathcal{B}$ for fixed ϵ. Their result is very general, and applies for any distribution in $[0, 1]$. That means it applies to the $\{0, 1\}$-coin flips considered here. The following form of their lemma takes their general result and applies it to the special case of $\{0, 1\}$ random variables.

Lemma 4.1 (Lemma 7.5 of [2] applied to $X_i \sim \mathsf{Bern}(p)$). *Any (α, β) approximation algorithm that uses T_B flips must satisfy*

$$\mathbb{E}[T_{\mathcal{B}}] \geqslant (1-\beta)(1-\alpha)^2 \ln\left(\frac{2-\beta}{\beta}\right) \frac{1-p}{4e^2\alpha^2 p}.$$

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

BERNOULLI_FACTORY_APPROXIMATION *Input:* C, $\mathsf{Bern}(p)$.

1 $A \leftarrow 0$, $S \leftarrow 0$
2 Repeat
3 Draw $W \leftarrow \mathsf{Bern}(Cp)$
4 $A \leftarrow A + 1$, $S \leftarrow S + W$.
5 Until $S = 4$.
6 $\hat{p} \leftarrow 4/(CA)$

To apply this lemma, fix $C \geqslant 1$ and $\epsilon > 0$, and consider the approximation for p given in algorithm BERNOULLI_FACTORY_APPROXIMATION.

Lemma 4.2. *The algorithm is an* $(\epsilon^{1/2}, 1/4)$*-approximation algorithm for* $p \in [0, 1/2]$*, and the expected number of* $\mathsf{Bern}(p)$ *draws is* $4\mathbb{E}[T]/(Cp)$.

Proof. Here A has a negative binomial distribution with parameters 4 and Cp and support $\{4, 5, \ldots\}$. The mean and standard deviation of $CA/4$ are

$$\mathbb{E}[CA/4] = 1/p, \quad \mathrm{SD}(CA/4) = (C/4)((1 - Cp))^{1/2}/(Cp) \leqslant \epsilon^{1/2}/(4p).$$

Hence, by Chebyshev's inequality,

$$\begin{aligned}
\mathbb{P}(1/p - 2\epsilon^{1/2}/(4p) \leqslant CA/4 \leqslant 1/p + 2\epsilon^{1/2}/(4p)) &\geqslant 1 - 1/2^2, \\
\mathbb{P}(1 - \epsilon^{1/2}/2 \leqslant pCA/4 \leqslant 1 + \epsilon^{1/2}/2) &\geqslant 1 - 1/4, \\
\mathbb{P}((1 - \epsilon^{1/2}/2)^{-1} \geqslant 4/(pCA) \geqslant (1 + \epsilon^{1/2}/2)^{-1} &\geqslant 1 - 1/4, \\
\mathbb{P}(p(1 + \epsilon^{1/2}) \geqslant 4/(CA) \geqslant p(1 - \epsilon^{1/2})) &\geqslant 1 - 1/4,
\end{aligned}$$

where the last line used the fact that $1/(1 - x) \leqslant 1 + 2x$ and $1/(1 + x) \geqslant 1 - 2x$ for all $x \in [0, 1/2]$.

Finally, the expected number of $\mathsf{Bern}(p)$ draws is just $4/(Cp)$ times the expected number of draws of $\mathsf{Bern}(p)$ to generate one $\mathsf{Bern}(Cp)$-coin. □

Proof of Theorem 1.2. From the previous two lemmas,

$$\frac{4\mathbb{E}[T]}{Cp} \geqslant (1 - 1/4)(1 - \epsilon^{1/2})^2 \ln\left(\frac{2 - 1/4}{1/4}\right) \frac{1 - p}{4e^2\epsilon p}.$$

If $p \in [0, (1 - \epsilon)/C]$ then $1 - p \geqslant (1 - (1 - \epsilon)C^{-1})$, and solving for $\mathbb{E}[T]$ gives the lower bound. □

5. Conclusions

This algorithm is the first Bernoulli factory for $f(p) = Cp$ and $Cp \leqslant 1 - \epsilon$ that has a provable upper bound for the expected number of flips. Moreover, this algorithm is both simple to implement and faster in practice than existing methods. Furthermore, when

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press

considering the running time for all such factories that work for all $p \in [0, (1-\epsilon)/C]$, this method comes with a constant factor of the best possible expected running time.

References

[1] Asmussen, S., Glynn, P. W. and Thorisson, H. (1992) Stationarity detection in the initial transient problem. *ACM Trans. Model. Comput. Simul.* **2** 130–157.

[2] Dagum, P., Karp, R., Luby, M. and Ross, S. (2000) An optimal algorithm for Monte Carlo estimation. *SIAM J. Comput.* **29** 1484–1496.

[3] Durrett, R. (2005) *Probability: Theory and Examples*, third edition, Brooks/Cole.

[4] Flegal, J. and Herbei, R. (2012) Exact sampling for intractable probability distributions via a Bernoulli factory. *Electron. J. Statist.* **6** 10–37.

[5] Keane, M. S. and O'Brien, G. L. (1994) A Bernoulli factory. *ACM Trans. Model. Comput. Simul.* **4** 213–219.

[6] Łatuszyński, K., Kosmidis, I., Papaspiliopoulos, O. and Roberts, G. O. (2011) Simulation events of unknown probability via reverse time martingales. *Random Struct. Alg.* **38** 441–452.

[7] Nacu, S. and Peres, Y. (2005) Fast simulation of new coins from old. *Ann. Appl. Probab.* **15** 93–115.

[8] Thomas, A. C. and Blanchet, J. (2011) A practical implementation of the Bernoulli factory. `arXiv:1105.2508`

[9] von Neumann, J. (1951) Various techniques used in connection with random digits. In *Monte Carlo Methods*, Vol. 12 of *Applied Mathematics Series*, National Bureau of Standards.

https://doi.org/10.1017/S0963548315000371 Published online by Cambridge University Press