Numerical experiments of the optimal state-dependent exponential tilting algorithm for rare event simulation

Brennan Hall

University of California Santa Barbara

March 21, 2019

Summary

We discuss the algorithm presented in (Blanchet, Leder & Glynn 2009) used to ensure bounded relateive error in events with vanishing probability. The so-called optimial state-dependent exponential tilting method applies an exponential tilt as the optimal importance measure at each step, making use of the conditional information of the random walk's current value. Numerical simulations of the optimial state-dependent importance sampling method are performed to verify the theoretical results presented by Blanchet, Leder, and Glynn for estimating $\mathbf{P}\left(S_n/n > \beta\right)$. In addition to verfication, various comparisons are made to the standard optimal exponential tilting method.[1]

[1] These experiments were conducted using the R software programming language. Select code has been included in Appendix A.1.

1. Introduction

Efficient estimation of the probabilities of rare events is a classical problem in many different fields – essentially any field that measures events that happen infrequently compared to the time horizon. The problem is typically posed as estimating the probability of the sample mean of n independent and identically distributed (iid) random variables being in a closed and convex set. Large deviations theory has provided many results in the vein of rare event simulation, and we will make use of some of those results here. Cramer's Theorem is perhaps the most important to us as it provides us with the behavior of the asymptotic distributions for the sample mean of iid random variables. In particular, it tells us that the distributions follow a large deviations principle with a rate function defined by the Legendre transform of the log-moment generating function, $\psi(\theta) = \log \mathbf{E}[\exp\{\theta X\}]$. The Legendre transform is known to be a nonegative, proper, and strictly convex function so it is a nice functin to work with in optimization problems.

In the context of rare event simulation, an optimization problem often considered is one of determining an optimum alternative sampling distribution for the sample mean. Since the behavior of the sample mean is asymptotically described by the rate function, we can formulate what is done in importance sampling as a convex optimization problem so long as we limit the set of distributions we can consider to an appropriate form. In the context of this study, that form will be what is known as an exponentially tilted distribution, of which the family of these distributions forms a convex set.

2. Rare event simulation and importance sampling

To introduce the problem of simulating rare events, we will establish a general setting in this section and then narrow our focus to a particular case in the following sections. In general, we will be looking at a random walk generated by a sequence of iid d-dimensional random variables $\{X_i\}_{i=1}^{n}$ under a measure $\mathbf{P}$.

$$S_n = X_1 + X_2 + \ldots + X_n. \tag{1}$$

The need to simulate events is to estimate some expectation. We will look at estimating a value for $\alpha_n = \mathbf{E}[\mathbb{1}_{\{S_n/n \in A\}}] = \mathbf{P}(S_n/n \in A)$ for some closed and convex set A. The difficult in doing this arises as the event $\{S_n/n \in A\}$ becomes "rare" in the sense that, in a Monte Carlo simulation setting, the event will appear very few times in proportion to the total number of trials. The basic technique used to estimate α_n would be to generate a sequence of iid random variables, $Y_1, Y_2, \ldots$ (in our case $Y_k = \mathbb{1}_{\{S_n/n \in A\}}$ for some fixed n), and take the sample mean $Q_K = (Y_1 + \ldots + Y_K)/K$ to be the estimator. In cases where Y_k are rare, the variance of Q will typically be large (at least compared to $\mathbf{E}[Y]$), and so the convergence rate of Q_K will be particularly slow, requiring a large number of samples before the variance is acceptably small. Multiple methods have been studied to improve on this method. For now, we will consider a variation of an importance sampling method to reduce the variance.

Methods of importance sampling are predicated around identifying an alternative sampling distribution such that, when measured under the new measure, the event of interest is at least less rare than it is under the original measure. In general, the importance sampling "step" to estimation is performed via the Radon-Nikodym derivative, defined as

$$\mathbf{f}(y) = \frac{\mathrm{d}\mathbf{P}}{\mathrm{d}\mathbf{P}_\theta}(y) \tag{2}$$

where it is required that $\mathbf{P}$ be absolutely continuous with respect to $\mathbf{P}_\theta$. Using the new measure $\mathbf{f}$, we consider the alternative estimator,

$$\bar{Q}_K = \frac{1}{K} \sum_{k=1}^{K} \bar{Y}_k \mathbf{f}(\bar{Y}_k) \tag{3}$$

where each $\bar{Y}_k$ is sampled from the distribution $\mathbf{P}_\theta$. The use of the notation $\mathbf{P}_\theta$ for the alternative sampling distribution is indicative of our need to constrain the set of possible distributions to a parametrized family of sampling distributions of the original distribution. Without this restriction, we would be optimizing over a set of distribtions that includes elements such as $\tau(\mathrm{d}y) = m^{-1} y \mathbf{P}(\mathrm{d}y)$ where if we let $m = \mathbf{E}[Y]$ then $\mathbf{P} \ll \tau$ and $\mathbf{f}(y) = m/y$. However, this would be an unreasonable distribution to consider since it depends on the value we are trying to estimate.

One family of distributoions to consider are those said to be *exponentially tilted* by the parameter θ, defined as distributions of the form

$$\mathrm{d}F_\theta = \exp\Big\{\theta x - \psi(\theta)\Big\}\mathrm{d}F \tag{4}$$

where $\psi(\theta)$ is the log moment generating function for the distribution $\mathbf{P}$.

The efficiency of the exponential tilting method is highlighed by the previous work in Asmussen & Glynn (2007)

Proposition 2.1 (Asmussen & Glynn (2007))**.** *Optimal exponential tiliting is the only iid importance sampling algorithm that provides an estimator Q with at least logarithmic efficiency, such that*

$$\liminf_{n\uparrow\infty} \frac{\log \mathbf{E}\left[Q^2\right]}{\log \mathbf{E}\left[Q\right]^2} = 1$$

However, there are some important shortfalls of the OET method that arise when this study is applied under the large deviations framework. Under this framework, we will assume the following:

1. Without loss of generality, $\mathbf{E}\left[Y\right] = 0$ and $\mathbf{Var}\left(Y\right) = \sigma^2 < \infty$
2. The log-MGF, $\psi\left(\theta\right)$ is said to be *steep* in the sense that for each w, there exists $\theta_w > 0$ such that $\psi'\left(\theta_w\right) = w$.
3. The random variable Y is nonlattice, meaning the modulus of the characteristic function is strictly less than one except at the origin.

Furthermore, we will introduce the large deviations rate function as

$$J\left(w\right) = \sup_{\theta\geq 0}\left\{\theta w - \psi\left(\theta\right)\right\}. \tag{5}$$

This will be of particular use in the algorithm proposed later as this provides us with the relation

$$J'\left(w\right) = \psi^{'-1}\left(w\right) = \theta_w \tag{6}$$

for $w \geq 0$. which

To implement the OET method, one would take $\theta = \theta_\beta$ in (4) to identify the alternative sampling distribution. However, the following statements will detail why this method will perform insufficiently.

Theorem 2.2 (Bahadur, Rao et al. (1960))**.** *Under the above assumptions, for fixed $\beta > 0$*

$$\lim_{n\uparrow\infty} \mathbf{P}\left(S_n > n\beta\right) = \frac{\exp\left\{-nJ\left(\beta\right)\right\}}{\theta_\beta\sqrt{2\pi n\psi''\left(\theta_\beta\right)}}\left(1 + o\left(1\right)\right) \tag{7}$$

Proposition 2.3 (Proposition 3, Blanchet et al. (2009)).

$$\lim_{n\to\infty} \mathbf{P}\left(S_n - n\beta > x \,|\, S_n > n\beta\right) = \exp\{-\theta_\beta x\} \tag{8}$$

Using these results, we see that $n^{-1/2}(S_n - n\beta) \overset{\mathcal{D}}{\approx} \mathcal{N}(0, \psi''(\theta_\beta))$ by the CLT, meaning the "overshoot" $S_n - n\beta$ is of order $O(n^{1/2})$ in distribution. But Proposition 2.3 indicates that the conditional distribution is of order $O(1)$ so the distribution $\mathbf{P}_{\theta_\beta}$ may not describe the random walk accurately for scales finer than $O(1)$. To conclude, the estimator induced by $\mathbf{P}_{\theta_\beta}$, is not *strongly efficient*, meaning the squared coefficient of variation[2] is unbounded as $n \uparrow \infty$.

3. Optimal State-Dependent Exponential Tiliting

To address this shortcoming of the OET method, a new method, named Optimal State-Dependent Exponential Tiliting (OSDET), which makes use of the current state of the random walk and has been proven to be strongly efficient is proposed. The idea behind the algorithm is to recompute the standard OET change of measure at each increment of the random walk in the hope of controlling the overshoot. For the purposes of description now and demonstration in the numerical experiments performed in the next section, we will shift to a more narrow setting. So from now on, we will consider the $d = 1$-dimensional case to estimate the probability $\mathbf{P}(S_n > n\beta)$.

Here we describe algorithm in detail.
Set $w = \beta > n^{-1/2}$, $L = 1$, $s = 0$, $\bar{s} = 0$, $k = 0$, $\lambda > 2\beta$.
Repeat Step 1. until $n = k$ or $w \le (n-k)^{-1/2}$ or $w > \lambda$.

1. Sample X from F_{θ_w} as defined by (4) and (6)

$$\begin{aligned} &L \leftarrow \exp\{-\theta_w X + \psi(\theta_w)\} L \\ &s \leftarrow s + X \\ &k \leftarrow k + 1 \\ &w \leftarrow (n\beta - s)/(n-k) \end{aligned}$$

2. If $k < n$, sample $X_{k+1}, \ldots, X_n$ from F_{θ_w}

$$\begin{aligned} &\bar{s} \leftarrow X_{k+1} + \ldots + X_n \\ &L \leftarrow \exp\left\{-\theta_w \bar{s} + (n-k)\psi(\theta_w)\right\} L \end{aligned}$$

[2] Coefficient of variation for an estimator Q defined as $cv_n(Q) = \frac{\mathbf{Var}(Q)}{(\mathbf{E}[Q])^2}$

3. Output $Y_n = L \times \mathbb{1}_{\{s+\bar{s}>n\beta\}}$

The variable w represents a measure of relative distance between the value of the random walk and the boundary $n\beta$. Using this in our determnining of the parameter θ_w at each step, allows us to establish two conditions on when to stop the dynamic updating of OET and revert to the standard OET. When the condition $w \leq (n-k)^{-1/2}$ is met, it indicates that importance sampling is no longer needed as the event is not considered rare under the current measure $\mathbf{P}_{\theta_w}$. On the other hand, if the condition $w > \lambda$ is met, it indicates that the random walk has entered a region where the distance to the boundary is no longer at least linear related to the time remaining. In both cases, the algorithm applies the exponential tilt one last time for the remaining steps of the random walk. The algorithm yields an estimator defined in the following manner. First, define for $1 \leq j \leq n$, the random variable

$$W_j = (n\beta - S_j)/(n-j) \tag{9}$$

and the stopping times, $\tau_0^{(n)} = \inf\{1 \leq k \leq n : n\beta - S_k \leq (n-k)^{-1/2}\}$, $\tau_1^{(n)} = \inf\{1 \leq k \leq n : W_k > \lambda\}$, and $\tau^{(n)} = \tau_0^{(n)} \wedge \tau_1^{(n)} \wedge n$. Then the estimator obtained from the algorithm is

$$\begin{aligned} Y_n = &\exp\left\{-\sum_{j=1}^{\tau^{(n)}-1} (\theta_{W_j}\bar{X}_j - \psi(\theta_{W_j}))\right\} \times \\ &\exp\left\{-\theta_{\tau^{(n)}}(\bar{S}_n - \bar{S}_{\tau^{(n)}}) + (n-\tau^{(n)})\psi(\theta_{\tau^{(n)}})\right\}\mathbb{1}_{\{S_n>n\beta\}}. \end{aligned} \tag{10}$$

Of most importance is the theorem proved rigorously in Blanchet et al. (2009),

Theorem 3.1 (Theorem 2, Blanchet et al. (2009))**.** *For each $p > 1$,*

$$\sup_{n\geq 1} \frac{\widetilde{\mathbf{E}}[Y_n^p]}{\mathbf{P}(S_n > n\beta)^p} < \infty$$

$\widetilde{\mathbf{E}}$ denotes the expectation operator induced by the OSDET algorithm. Our main concern is only for $p = 2$ which demonstrates strong efficiency. In the next section we will apply the OSDET algorithm to a random walk to numerically verify this main result.

4. Numerical Results

As there are no numerical experiments presented in Blanchet et al. (2009) that demonstrate their results in application, we provide some analysis to the OSDET algorithm and compare its efficiency to that of the standard OET algorithm.

Staying in the same setting as Section 3, we further fix $\beta = 1$ so that we consider the problem of finding an estimate for $\mathbf{P}(S_n > n)$ for increasing values of n. Additionally, we take the increments of the random walk to be Gaussian random variables. Using the algorithm for OSDET and OET, we then perform $K = 1e6$ Monte Carlo simulations to obtain the estimator (3).

The first consideration is to check that the estimates are converging to the expected value to verify the estimator is unbiased. We should expect the values of $\hat{\alpha}_n \downarrow 0$. Figures 1 and 2 verify this is the case. Note in Figure 2 that for $n > 11$, the estimate of $\hat{\alpha}_n$ is recorded as zero in machine precision so the log-estimate is set to -Inf, hence the vanishing plot.

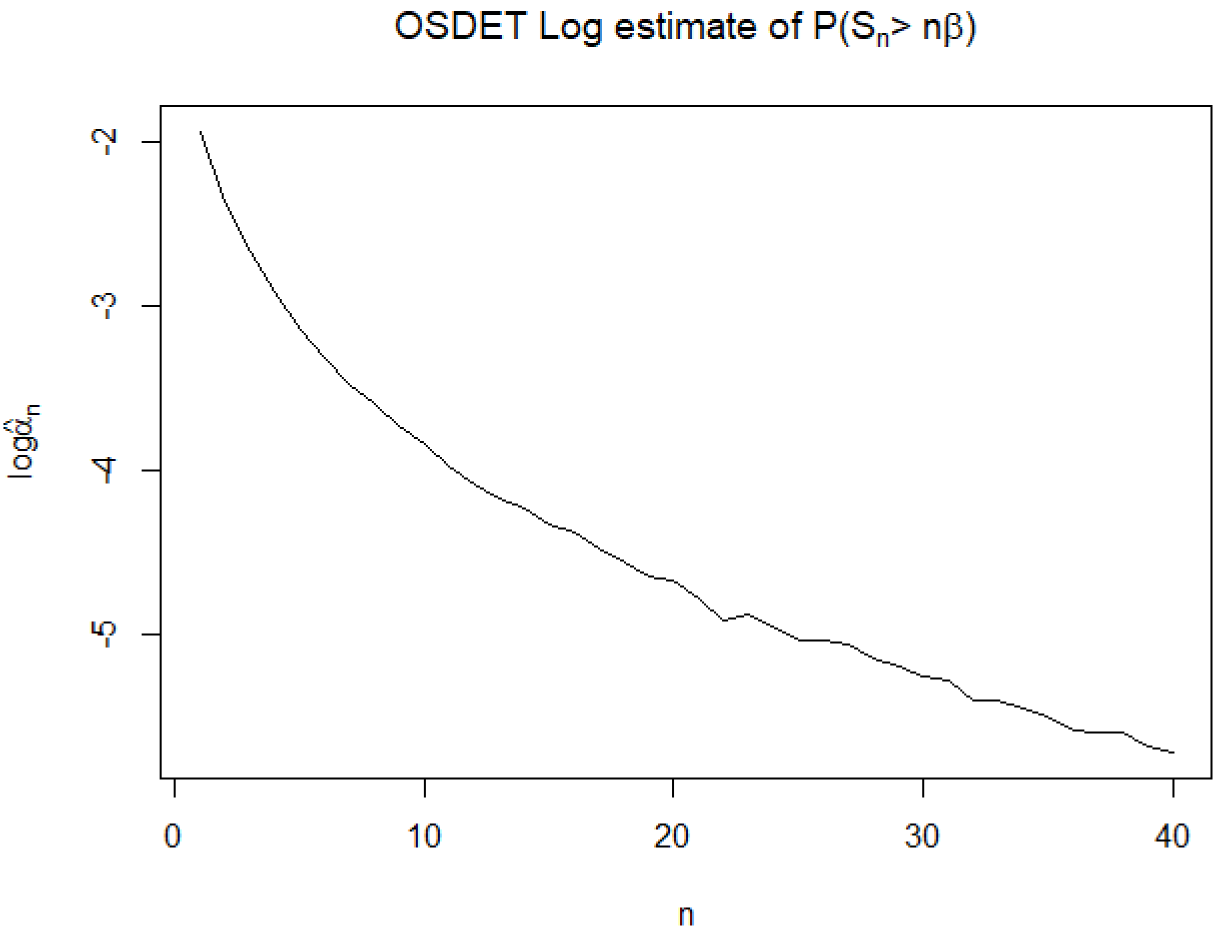

Figure 1. *Log probability estimates for OSDET method.*

Next we verify that the OSDET method provides bounded relative error while

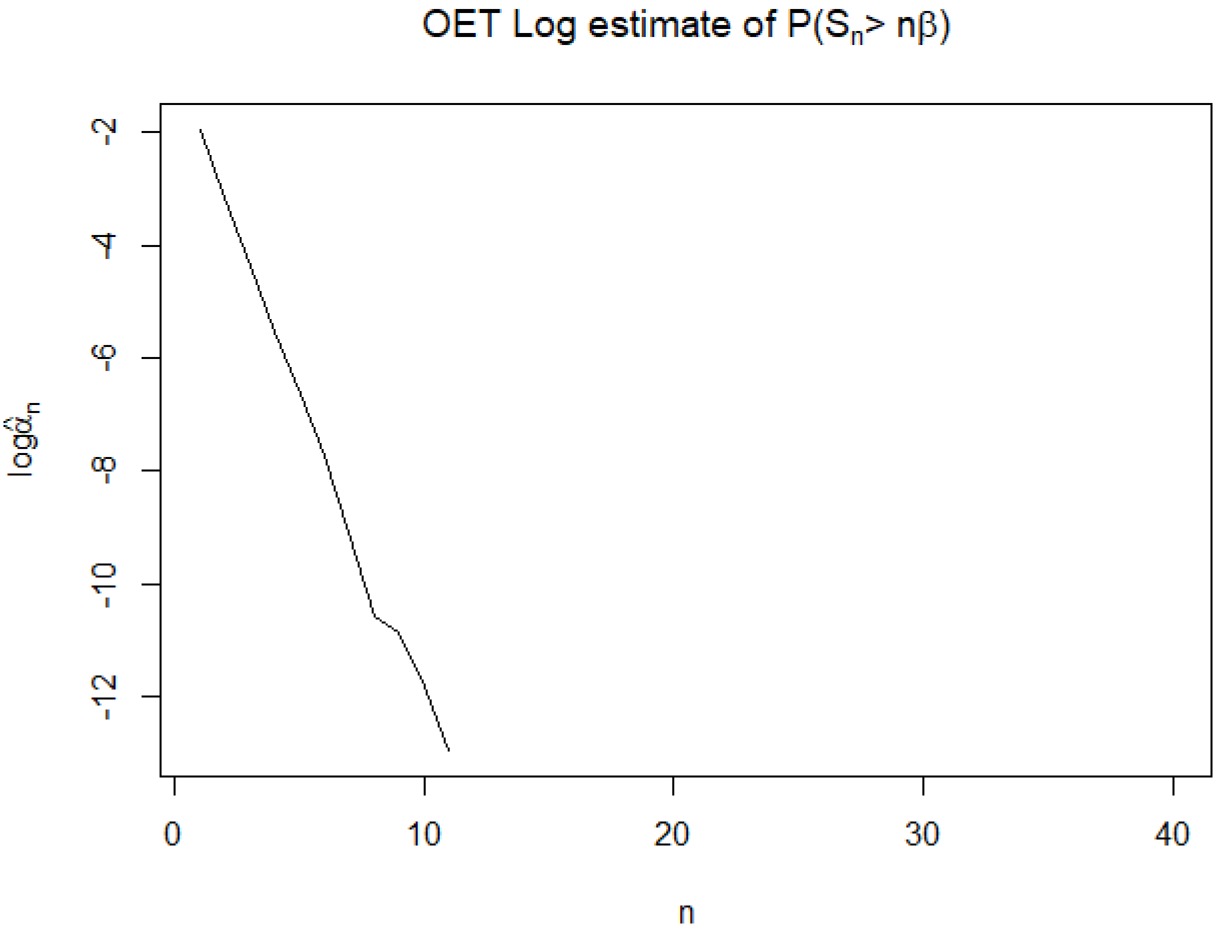

Figure 2. *Log probability estimates for OET method.*

comparing to the unbounded nature of the OET method. It should be quite clear that the squared coefficient of variation for the standard OET method is unbounded in Figure 3. The verification for bounded relative error seems to be slightly more nuanced, however. Under the current implementation of the algorithm, the bounded relative error was not able to be verified.

As a final note, we address the system compute time for these algorithms. The simulations for each algorithm were run in parallel on a local machine utilizing a 6 core/12 thread CPU, each clocked to 4.7GHz. Table 4 reports the system time to complete the 1e6 Monte Carlo simulations for each algorithm. It is quite clear that although, we get strong efficiency from the OSDET, the drastically longer compute time should be considered, especially when simulating more complex variables.

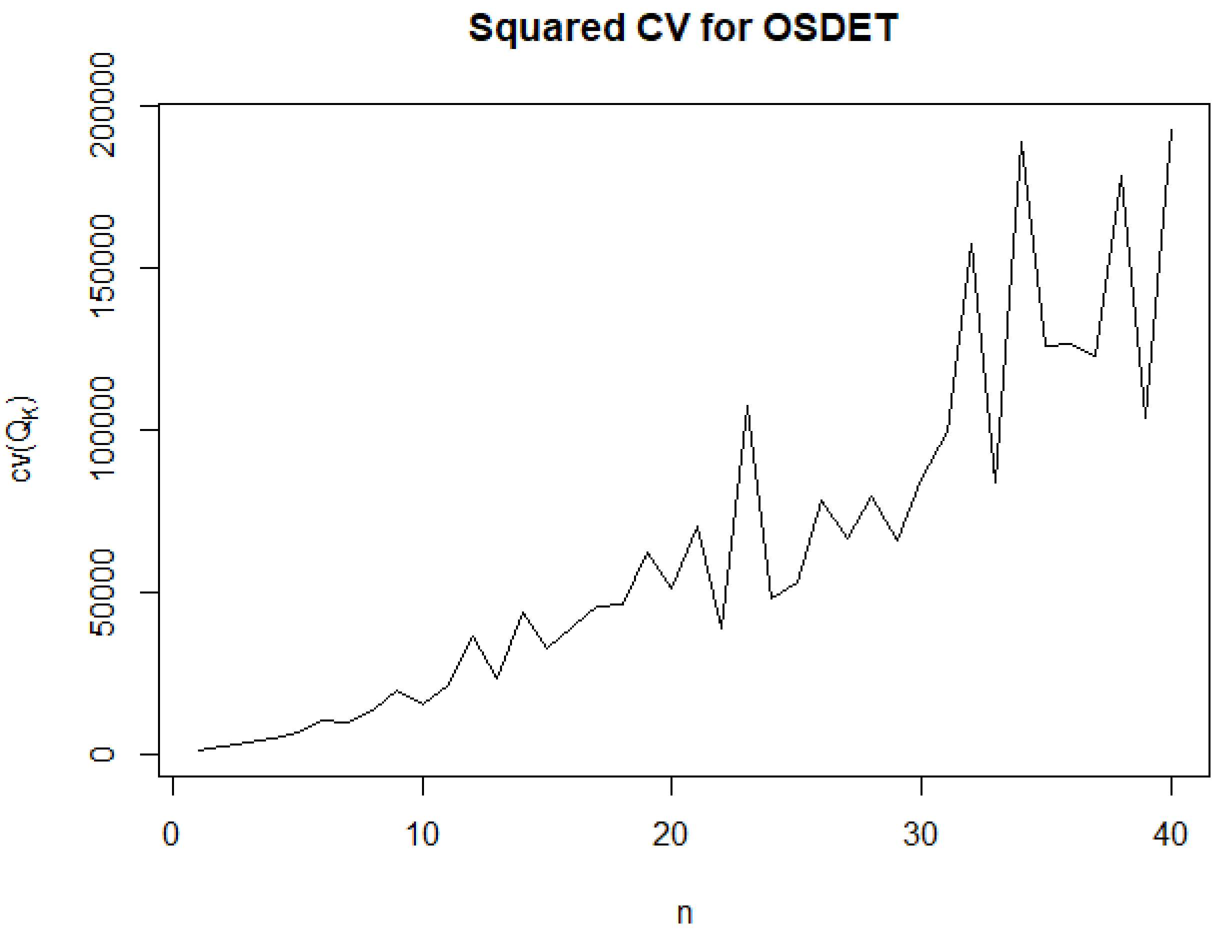

Figure 3. *Squared coefficient of variation for OSDET method.*

	OSDET	OET
System time	722.80	53.78

Table 4: *System compute time (sec) for each algorithm*

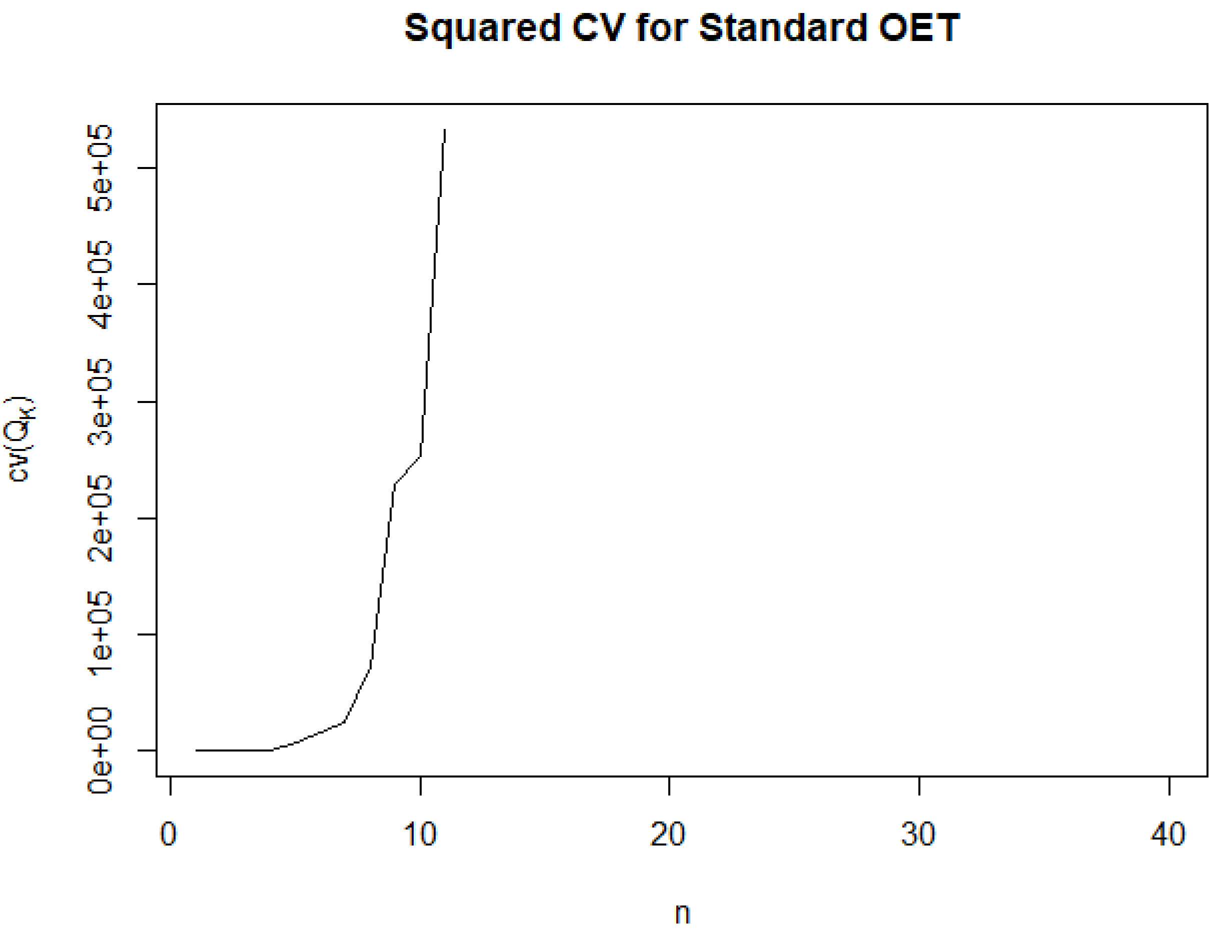

Figure 4. *Squared coefficient of variation for OET method.*

References

Asmussen, S. & Glynn, P. W. (2007), *Stochastic simulation: algorithms and analysis*, Vol. 57, Springer Science & Business Media.

Bahadur, R. R., Rao, R. R. et al. (1960), ‘On deviations of the sample mean’, *Ann. Math. Statist* **31**(4), 1015–1027.

Blanchet, J. H., Leder, K. & Glynn, P. W. (2009), Efficient simulation of light-tailed sums: an old-folk song sung to a faster new tune..., *in* ‘Monte Carlo and Quasi-Monte Carlo Methods 2008’, Springer, pp. 227–248.

A. Appendix

A.1. Selected R code.

```
#### packages for parallelization ####
library(foreach)
library(doParallel)

##### phi(theta) #####
gauss_phi <- function(theta, mu, sigma2){
        phi = mu*theta + 0.5*sigma2*theta^2
        return(phi)
}

##### OSDET Algorithm #####
gauss_osdet <- function(n, b, l, s = 0, mu, sigma2){
## to compute estimates a_n = P(S_n > n*b)
### with S_n = X1 +...+ Xn

        k = 0
        L = 1
        w = b
        mu1 = mu
        sigma21 = sigma2

        ## perform state-dependent OET while event of interest is rare
        repeat{

                # update parameters by exp tilt
                theta <- max((w - mu1) / sigma21, 0)
                mu1 = mu1 + theta*sigma21

                # sample from distn
                X = rnorm(1)
                # update IS measure
                L = exp(theta*X -
                        gauss_phi(theta, mu = mu1, sigma2 = sigma21))*L

                # update RW
                s <- s + X*L
                k = k + 1
                w <- max((n*b - s)/(n - k), 0)

                if (n == k | w <= 1/sqrt(n-k) | w > 1) {
                        break
                }
        }

        ### 'while' loop ended implies event of interest is no longer rare
        #### so finish algorithm with standard OET
        barS = 0
        sbar = 0
        if (k < n){
                size = length((k+1):n)
                theta = max((w - mu1) / sigma21,0)
                mu1 = mu1 + theta*sigma21

                barS = sum(rnorm(size))

                L = exp(theta*barS -
                        (n-k)*gauss_phi(theta, mu = mu1, sigma2 = sigma21))*L
                sbar = barS*L
        }

        Ind <- 0+(s + sbar > n*b)
        Y = L*Ind
        return(Y)
}
```

```
##### OET Algorithm #####
oet <- function(n, b, s = 0, mu, sigma2){

theta = (b-mu)/sigma2
mu1 = mu + theta*sigma2

s = sum(rnorm(n))

L = exp(theta*s - n*gauss_phi(theta, mu = mu1, sigma2 = sigma2))
sbar = s*L

Ind <- 0+(sbar > n*b)
Y = L*Ind
return(Y)
}


######### Simulations ############
cores=12

#### OSDET Simulation #####
set.seed(232)
M = 1e6
N = seq(from=100, to = 1000, by = 100)

cl <- makeCluster(cores)
registerDoParallel(cl)

osdet <- foreach(k = N) %dopar% {
        replicate(M, gauss_osdet(n = k, b = 2, l = 50, mu = 0, sigma2 = 1))
}
stopCluster(cl)

### Analysis ###
avg_osdet <- array(as.numeric(unlist(osdet)), dim=c(M, length(N)))
avg_osdet <- apply(avg_osdet, 2, sum)/M
```

```
logAvg <- log(avg_osdet)
m2_osdet <- array(as.double(unlist(osdet)), dim=c(M, length(N)))
m2_osdet <- apply(m2_osdet, 2, function(x) sum(x^2))/M
var_osdet <- m2_osdet - avg_osdet^2

re_osdet = var_osdet/(avg_osdet^2)


#### OET Simulation #####
set.seed(232)
Mo = 1e6
No = seq(from=100, to = 1000, by = 100)

cl <- makeCluster(cores)
registerDoParallel(cl)

out <- foreach(k = No) %dopar% {
replicate(Mo, oet(n = k, b = 2, mu = 0, sigma2 = 1))
}

stopCluster(cl)

### Analysis ###
avg_oet <- array(as.numeric(unlist(out)), dim=c(Mo, length(No)))
avg_oet <- apply(avg_oet, 2, sum)/Mo
logAvg_oet <- log(avg_oet)
m2_oet <- array(as.double(unlist(out)), dim=c(Mo, length(No)))
m2_oet <- apply(m2_oet, 2, function(x) sum(x^2))/Mo
var_oet <- m2_oet - avg_oet^2

re_oet = var_oet/(avg_oet^2)
```