

# Minimum Cross-Entropy Methods for Rare Event Simulation

Ad Ridder\*

Vrije Universiteit  
Amsterdam

Reuven Rubinstein

Faculty of Industrial Engineering and Management  
Technion, Haifa

November 21, 2007

## Abstract

In this paper we apply the minimum cross-entropy method (MinxEnt) for estimating rare-event probabilities for the sum of i.i.d. random variables. MinxEnt is an analogy of the Maximum Entropy Principle in the sense that the objective is to minimize a relative (or cross) entropy of a target density  $h$  from an unknown density  $f$  under suitable constraints. The main idea is to use the solution to this optimization program as the simulation density in importance sampling. We shall see that some existing importance sampling methods can be cast in a MinxEnt program, such as the large deviations approach for light tails and the hazard rate twisting for heavy tails. As an extension we shall consider a correlated version of this hazard rate twisted solution which give better simulation results. The sample generation is based on a Gibbs sampler algorithm.

---

\*Corresponding author. Address: Department of Econometrics, de Boelelaan 1105, 1081 HV Amsterdam, Netherlands; Email: aridder@feweb.vu.nl

**keywords:** minimum cross-entropy, rare-events, importance sampling, asymptotically optimal, Gibbs sampler.

## 1 Introduction

In this paper we study an importance sampling algorithm for simulating efficiently the rare event problem

$$\ell \stackrel{\text{def}}{=} P_h(\mathbf{X} \in A),$$

where  $\mathbf{X}$  denotes the random process under study,  $h$  represents the statistical law or probability densities of the process, and  $A$  is the rare event, meaning that the probability  $\ell$  is very small, say  $10^{-9}$  or less. Whereas the naive simulation would use the original densities  $h$  to generate samples  $\omega$  of the process—resulting in hardly any observation of the rare event—, in importance sampling one generates samples from other densities  $f$  in such a way that one sees more successful observations. Unbiasedness is obtained by compensating the fraction of successes by the likelihood ratio  $h(\mathbf{x})/f(\mathbf{x})$ . We refer to [7] for a recent survey on importance sampling simulations and rare events, including many applications.

More specifically, we shall consider the case where the rare event is parameterized by a rarity parameter, say  $\gamma$ , such that

$$\ell(\gamma) \stackrel{\text{def}}{=} P_h(\mathbf{X} \in A(\gamma)) \rightarrow 0 \quad \text{as } \gamma \rightarrow \infty.$$

We denote the importance sampling estimator of this probability by  $\bar{Y}(\gamma)$  while using the importance sampling density  $f$  for generating samples. The assessment of the statistical quality of  $\bar{Y}(\gamma)$  is done by analysing its second moment (or variance) with respect to its first moment. Ideally, we like to obtain strong efficiency:

$$\limsup_{\gamma \rightarrow \infty} \frac{\text{Var}_f[\bar{Y}(\gamma)]}{(E_f[\bar{Y}(\gamma)])^2} \leq M < \infty,$$

which says that the relative error of the estimator remains bounded. When the estimator satisfies this property, a constant sample size suffices to obtain the same relative width of confidence intervals irrespective of how large  $\gamma$  (or

how small  $\ell(\gamma)$  may be [11]. Only in rare occasions it is possible to construct a strongly efficient importance sampling algorithm. A weaker condition is [3, 7, 11]

$$\lim_{\gamma \rightarrow \infty} \frac{\log E_f[\bar{Y}^2(\gamma)]}{\log E_f[\bar{Y}(\gamma)]} = 2.$$

When the estimator satisfies this property, one calls the algorithm (and the estimator) logarithmically efficient, or asymptotically optimal. Then the sample sizes grow polynomially (or at some other subexponential rate) to remain the same relative width of confidence intervals, as  $\gamma \rightarrow \infty$ .

There are numerous techniques that construct importance sampling algorithms, in all kind of statistical settings. Concerning efficient importance sampling methods in applications to stochastic operations research problems (queues, reliability, telecommunications, etc), we refer to the overviews in [3, 7, 11, 17]. In our paper we propose an approach that is based on an optimization program involving the Kullback-Leibler divergence. Here we like to mention the well-known cross-entropy method for finding importance sampling densities [26]. This method seeks the new density that minimizes the Kullback-Leibler divergence of the zero-variance density from a parameterized family of probability densities:

$$\inf_{\Theta} \int f(\mathbf{x}|\theta) \log \frac{f(\mathbf{x}|\theta)}{f^*(\mathbf{x})} d\mathbf{x},$$

where  $f^*$  is the ‘optimal’ density

$$f^*(\mathbf{x}) = \frac{h(\mathbf{x})1\{\mathbf{x} \in A\}}{\ell(\gamma)},$$

which is the original density  $h$  conditioned on the rare event. The first-order conditions of the minimization program cannot be solved analytically or numerically (in general), but they can be reformulated to a kind of fixed-point equation which is solved approximately by simulating iteratively its stochastic counterpart. In most cases one does not obtain a closed-form solution, and thus to prove efficiency might be cumbersome.

Our program seeks the importance sampling density that minimizes the Kullback-Leibler divergence of the original probability density from a family of densities that is subjected to constraints:

$$\inf_{f \geq 0} \left\{ \int f(\mathbf{x}) \log \frac{f(\mathbf{x})}{h(\mathbf{x})} d\mathbf{x} : \int f(\mathbf{x}) w_i(\mathbf{x}) d\mathbf{x} = \theta_i, i = 1, \dots, m \right\}.$$

The major benefit of this optimization program is that its solution is directly available in a closed form expression, as we shall discuss in Section 2. The major difficulty lies in the choice of constraints for which we do not have general clues other than that these constraints force the rare event to occur more frequently than under the original density. Clearly we need nonnegativity ( $f \geq 0$ ), and unit mass, which means that  $\int f(\mathbf{x}) d\mathbf{x} = 1$  is one of the constraints.

Our optimization program is an analogy of the Maximum Entropy Principle [13]—where the target density  $h$  is the uniform density—, and is also known as the Principle of Minimum Discrimination Information [19], or, as we shall do, the minimum cross-entropy method (MinxEnt) [18, 25]. Finding densities by an entropy principle is a classical approach in areas such as information theory, decision analysis, Bayesian information theory, thermodynamics, statistical mechanics, statistical data analysis, etc, but seems to be new in the theory of rare-event simulation. The objective of our paper is to report some of our experiences of the MinxEnt approach to rare-event simulation.

The paper is organized as follows. Section 2 introduces the minimum cross-entropy (MinxEnt) program and gives its solution in general form. In Section 3 we consider level-crossing problems with light-tailed random variables. We shall see that a MinxEnt program coincides with the large deviations approach to importance sampling. Also in this section we present an adapted MinxEnt program that resolves a counter example to this approach. In Section 4 we consider a level-crossing problem with heavy-tailed random variables for which a hazard rate twisted solution has been developed that gives asymptotic optimality of the importance sampling algorithm [16]. Again we see that a MinxEnt program coincides with this approach, but more importantly, we present an adapted MinxEnt program which gives asymptotic optimality in all cases (assuming some distributional properties). Because our solution is a correlated multi-variate density, we analyse in Section 5 several algorithms to generate samples from this density, specifically we analyse convergence diagnostics of a Gibbs sampler. Finally we present in Section 6 simulation results of the heavy-tailed problem of Section 4, and compare them with two existing algorithms, one based on hazard rate twisting [16], and the other on conditional Monte Carlo [2]. We consider Weibull, Pareto and Lognormal distributed increments in the level-crossing problem. Our solution improves in all these cases the ha-

zard rate twisted solution in terms of performance of the associated estimator, but compared to the conditional Monte Carlo solution, only in the case of not so heavy-tailed Weibull increments we obtain better performance.

Nonetheless, we think that our approach is interesting from the point of view that it unifies several existing methods in one framework. Furthermore, in the heavy-tailed case we give an elegant proof of asymptotic optimality that covers all distributions (subject to regularity properties, see Section 4); we do not know whether it has been proven for the Lognormal in the conditional Monte Carlo method.

## 2 The minimum cross-entropy program

The minimum cross-entropy program (MinxEnt) for finding an optimal density  $f$  subject to constraints reads as follows [18, 25]:

$$\begin{aligned} & \inf_{f \geq 0} \mathcal{D}_{\text{KL}}(f|h) \\ & \text{s.t.} \quad \int f(\mathbf{x}) d\mathbf{x} = 1, \\ & \quad E_f[w_i(\mathbf{X})] = \theta_i, \quad i = 1, \dots, m. \end{aligned} \tag{1}$$

The  $\mathbf{X} = (X_1, \dots, X_n)$  in this program is a random vector distributed according to density  $h(\mathbf{x})$ . The constraints involve some known functions  $w_i : \mathbb{R}^n \rightarrow \mathbb{R}$  ( $i = 1, \dots, m$ ). The objective is minimizing the Kullback-Leibler divergence:

$$\mathcal{D}_{\text{KL}}(f|h) \stackrel{\text{def}}{=} \int f(\mathbf{x}) \log \frac{f(\mathbf{x})}{h(\mathbf{x})} d\mathbf{x}.$$

The concept of assigning probabilities to the outcomes of uncertain events has a long history going back to Laplace [20]. Laplace suggests to assign equal probabilities for the case when no information is available. This has been formalized by Shannon [29] who introduced the idea of entropy as a measure of uncertainty. The entropy is maximal for the uniform distribution. Later, Jaynes [13] considers the problem of maximizing Shannon's entropy under additional constraints, also known as the Maximum Entropy Principle for probability inference. The corresponding optimization program is similar to our (1) by replacing the objective by maximizing Shannon's entropy

$$H(f) \stackrel{\text{def}}{=} - \int f(\mathbf{x}) \log f(\mathbf{x}) d\mathbf{x}.$$

Our program (1) is a natural extension of this maximum entropy program and has been formulated already in [14, 19] under the name of the Principle of Minimum Discrimination Information. Finally, it has been noted [18] that a further generalization implements a more general probability divergence or cross-entropy as the objective function in the MinxEnt program.

Initially, the development of these (cross-)entropy programs was triggered to answer questions concerning which distributions satisfy the principle under moment constraints. The Maximum Entropy Principle received attention in a wide area of academic research fields, for instance information theory [15], natural language processing [4], utility theory [1], computer vision [6], spatial physics [30], and many others.

After leaving out the nonnegativity constraint, the MinxEnt program (1) is solved easily by applying the method of Lagrange multipliers:

$$f(\mathbf{x}) = h(\mathbf{x}) \exp(\lambda_0 + \sum_{i=1}^m \lambda_i w_i(\mathbf{x})),$$

where the  $\lambda_i$ 's are Lagrange multipliers satisfying

$$\begin{aligned} E_h \left[ \exp \left( \lambda_0 + \sum_{i=1}^m \lambda_i w_i(\mathbf{X}) \right) \right] &= 1 \\ E_h \left[ w_i(\mathbf{X}) \exp \left( \lambda_0 + \sum_{i=1}^m \lambda_i w_i(\mathbf{X}) \right) \right] &= \theta_i, \quad i = 1, \dots, m. \end{aligned}$$

Working out (see also [25]) we see that multiplier  $\lambda_0$  cancels out and we obtain the more illustrative expression

$$f(\mathbf{x}) = \frac{h(\mathbf{x}) \exp \left( \sum_{i=1}^m \lambda_i w_i(\mathbf{x}) \right)}{c(\boldsymbol{\lambda})},$$

where  $c(\cdot)$  is the normalizing constant:

$$c(\boldsymbol{\lambda}) = c(\lambda_1, \dots, \lambda_m) = E_h \left[ \exp \left( \sum_{i=1}^m \lambda_i w_i(\mathbf{X}) \right) \right],$$

and the  $\lambda_i$ 's ( $i \geq 1$ ) solve the nonlinear equations  $\nabla \log c(\boldsymbol{\lambda}) = \boldsymbol{\theta}$ , i.e.,

$$\frac{E_h \left[ w_i(\mathbf{X}) \exp \left( \sum_{i=1}^m \lambda_i w_i(\mathbf{X}) \right) \right]}{E_h \left[ \exp \left( \sum_{i=1}^m \lambda_i w_i(\mathbf{X}) \right) \right]} = \theta_i, \quad i = 1, \dots, m.$$

Notice that nonnegativity comes 'for free'.

### 3 MinxEnt and rare events with light tails

A classic problem involving rare events is

$$\ell(\gamma) \stackrel{\text{def}}{=} P_h(X_1 + \dots + X_\gamma > \gamma a), \quad (2)$$

where the increments  $X_1, X_2, \dots$  are independent and distributed as some random variable  $X$  with finite mean  $E[X] < a$ . The rarity parameter  $\gamma$  takes integer values, and therefore we follow the traditional notation by replacing  $\gamma$  by  $n$ . Notice that both the number  $n$  of increments and the overflow level  $na$  tends to infinity by letting  $n \rightarrow \infty$ . We denote  $S(\mathbf{X}) = X_1 + \dots + X_n$ , and let  $h_1(x)$  be the probability density of a single increment  $X$ . Thus,  $h(\mathbf{x}) = \prod_{j=1}^n h_1(x_j)$  is the joint probability density of the vector  $\mathbf{X} = (X_1, \dots, X_n)$ .

In this section we assume that  $X$  has a light-tailed distribution. Recall that a random variable  $X$  is light-tailed if its moment generating function  $E[\exp(\alpha X)] < \infty$  in an open neighbourhood of zero, that is,  $\alpha \in (-\epsilon, \epsilon)$  for some  $\epsilon > 0$ ; otherwise  $X$  is said to be heavy-tailed.

Let us explore the following heuristic for finding an importance sampling density  $f$  of the vector  $\mathbf{X}$ .

- To control the likelihood ratio we wish to have the new density ‘close’ (w.r.t. Kullback-Leibler divergence) to  $h$ .
- To obtain sufficiently many observations of the rare event during the simulations we wish to have  $S(\mathbf{X}) > na$  very likely.

This heuristic leads to the following MinxEnt program with a single (non-trivial) constraint:

$$\inf_f \left\{ \mathcal{D}_{\text{KL}}(f|h) : \int f(\mathbf{x}) d\mathbf{x} = 1, E_f[S(\mathbf{X})] = na \right\}. \quad (3)$$

Since  $S(\cdot)$  is an additive function the solution (see Section 2)

$$f(\mathbf{x}) = \frac{h(\mathbf{x})e^{\lambda S(\mathbf{x})}}{c(\lambda)} = \frac{\prod_{j=1}^n h_1(x_j) \exp\left(\lambda \sum_{j=1}^n x_j\right)}{(c_1(\lambda))^n},$$

factorises in a product density  $f(\mathbf{x}) = \prod_{j=1}^n f_1(x_j)$ , with marginals

$$f_1(x) = \frac{h_1(x)e^{\lambda x}}{c_1(\lambda)}, \quad c_1(\lambda) = E_{h_1}[e^{\lambda X}]. \quad (4)$$

Notice that this marginal density is an exponentially twisted version of the original  $h_1$  with twisting parameter  $\lambda$ , and that the normalizing constant equals the moment generating function of the variable  $X$ . The Lagrange multiplier  $\lambda$  solves  $(\log c(\lambda))' = na$ , which is—due to the factorization  $c(\lambda) = (c_1(\lambda))^n$ —equivalent to

$$(\log c_1(\lambda))' = a.$$

We see that the resulting exponentially twisted density is the same as the density that one would get from large deviations of the level-crossing probabilities (2) by letting  $\gamma$  ( $\gamma = n$ ) tend to infinity. There is an abundance of literature on the large deviations approach to rare event simulation, see for instance [21, 27, 28] and the recent monograph [7]. Under mild conditions these ‘large deviations solutions’ give efficient importance sampling algorithms for the one-sided level-crossing problem (2). The next section deals with a counter example.

### 3.1 A counter example to the large deviations approach

Consider the example in [10, Section 3], or example 5.2.13 of [7, page 114]. The jumps  $\mathbf{X} = (X_1, \dots, X_n)$  are i.i.d. standard Gaussian random variables, with partial sums  $S_n = \sum_{i=1}^n X_i$ . Let the target probability be

$$\ell(n) = P(S_n \leq -na(1+\epsilon) \text{ or } S_n \geq na) = P\left(\frac{1}{n}S_n \leq -a(1+\epsilon) \text{ or } \frac{1}{n}S_n \geq a\right),$$

where  $a > 0$  and  $\epsilon > 0$ . The large deviations solution would apply an exponential shift making the minimum rate point of the associated large deviations rate function  $J(\cdot)$  the most likely point of the simulation [7, 10]. That would mean here that the jumps are again i.i.d. Gaussian with unit variance, but with mean  $a$ , since  $J(a) < J(-a(1+\epsilon))$ . However, [7, 10] have shown that the associated importance sampling estimator is not efficient, its variance blows up. The reason being that the likelihood ratios of successful observations in  $(-\infty, -na(1+\epsilon)]$  explode.

We propose a MinxEnt program for finding correlated jumps. The rare event happens certainly when  $S_n^2 \geq n^2a^2(1+\epsilon)^2$ . Therefore we consider the constraint  $E_f[S_n^2] = \rho n^2$ , for some positive  $\rho$  to be determined later. Thus,

$$\inf_f \left\{ \mathcal{D}_{\text{KL}}(f|h) : \int f(\mathbf{x}) d\mathbf{x} = 1, E_f[S_n^2] = \rho n^2 \right\},$$



where  $h(\cdot)$  is the original multivariate normal density function of the  $n$  independent jumps. In the appendix A we shall show that the importance sampling estimator  $\bar{Y}(n)$  obtained by sampling with the solution density  $f$  satisfies

$$\liminf_{n \rightarrow \infty} \frac{\log E_f[\bar{Y}^2(n)]}{\log E_f[\bar{Y}(n)]} \geq \frac{2}{(1 + \epsilon)^2}.$$

As an illustration we give in Figure 1 two plots of this estimated logratio for the cases  $a = 1.5, \epsilon = 0.05$ , and  $a = 1.5, \epsilon = 1.0$ , respectively. The (importance sampling) simulation uses sample size  $k = 50000$  for all  $n$  in the range 5 until 200. We observed about the same numbers of the logratio for several choices of the righthand-side parameter  $\rho$ , either constant such as  $\rho = 1$ , or proportional to  $n$  such as  $\rho = 0.25n$ . Apparently, the theoretical lower bound  $2/(1 + \epsilon)^2$  is not very tight, as there seems to be not much difference in the ratios (at  $n = 200$  both are just above 1.96). Figure 2 shows the bad behaviour of the large deviations solution:  $n = 5, a = 1.0, \epsilon = 0.1$  (exact probability  $\ell = 1.63 \cdot 10^{-2}$ ) and sample sizes ranging from 50K until 10M. A ‘shock’ occurs whenever a negative successful sample is observed. The MinxEnt estimator remains extremely accurate.

*Figure 1. Estimated logratios in the range  $n = 5, \dots, 200$ .*

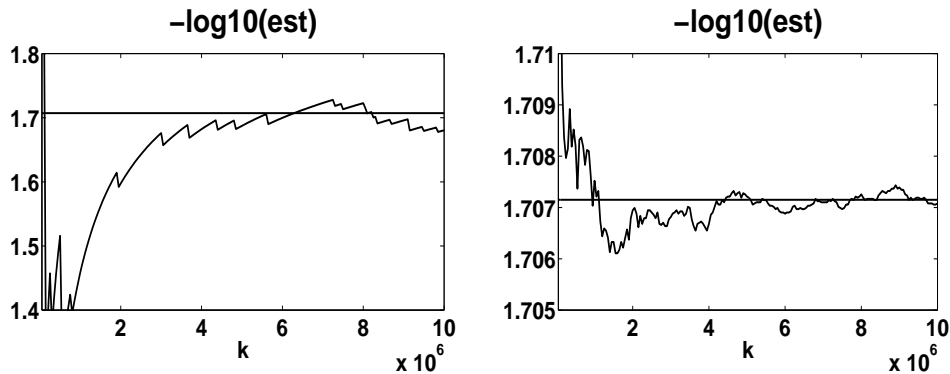


Figure 2. Logarithm of the estimations for sample sizes  $k = 5 \cdot 10^4, \dots, 10^7$ . The constant line marks the exact probability. Left: Large deviations solution. Right: MinxEnt solution.

## 4 MinxEnt and rare events with heavy tails

For the heavy-tailed case we consider the level-crossing probability with a constant number of increments:

$$\ell(\gamma) \stackrel{\text{def}}{=} P_h(X_1 + \dots + X_n > \gamma),$$

where  $\gamma \rightarrow \infty$ . The increments  $X_1, X_2, \dots \stackrel{\text{dist}}{\sim} X$  are i.i.d. and are assumed to have the following distributional properties.

### Assumption 1.

- A.  $X$  is a positive random variable ( $P(X \leq 0) = 0$ ).
- B.  $X$  is subexponentially distributed.
- C.  $X$  has a concave cumulative hazard function.

The increments are subexponentially distributed when they satisfy [9, Section 1.4]:

$$P(\sum_{i=1}^n X_i > x) \sim nP(X > x) \quad (x \rightarrow \infty),$$

where  $f(x) \sim g(x), x \rightarrow \infty$  means that  $\frac{f(x)}{g(x)} \rightarrow 1$  as  $x \rightarrow \infty$ . We denote the hazard rate of the density  $h_1$  of a single increment  $X$  by  $q(x) \stackrel{\text{def}}{=} h_1(x)/P(X > x)$ . The cumulative hazard function is defined as

$$Q(x) \stackrel{\text{def}}{=} \int_0^x q(y) dy.$$

It is easy to see that for  $x > 0$  (see also [9, 16])

$$h_1(x) = q(x)e^{-Q(x)}, \quad P(X > x) = e^{-Q(x)}.$$

The representation  $Q(x) = -\log P(X > x)$  gives immediately the following properties

$$(i) \ Q(x) \text{ is nondecreasing; } (ii) \ Q(0) = 0, \quad (iii) \ \lim_{x \rightarrow \infty} Q(x) = \infty.$$

Typical distributions that satisfy Assumption 1 are Weibull with shape parameter  $0 < \beta < 1$ , Pareto, and Lognormal. The cumulative hazard functions of the Weibull and Pareto are concave on the whole positive axis, whereas of the Lognormal distribution for  $x > E[\log X]$ .

A MinxEnt program similar to (3) of Section 3 with a constraint  $E_f[S(\mathbf{X})] = \gamma$  does not work in the heavy-tailed environment because the moment generating function  $c_1(\lambda)$  of the marginal increment (4) has infinite values for  $\lambda > 0$ . Then the idea is to apply a logarithmic transformation to the moment constraint to ‘make’ it finite. The cumulative hazard function is such a logarithmic transformation, and we shall present two MinxEnt programs involving it. But first, let us introduce the hazard rate twisting approach of [16].

#### 4.1 Hazard rate twisting

The idea of hazard rate twisting, introduced in [16], is to change the cumulative hazard function to  $\tilde{Q}(x) = (1 - \theta)Q(x)$  for some hazard rate parameter  $0 < \theta < 1$ . This results in marginal densities with heavier tails, and in an i.i.d. joint density. In [16, Theorem 3.2] it is proved that, under regularity conditions mentioned above, the importance sampling estimator is asymptotically optimal if one chooses for the parameter  $\theta = 1 - n/Q(\gamma)$ .

#### 4.2 MinxEnt and hazard rate twisting

Consider the MinxEnt program

$$\inf_f \left\{ \mathcal{D}_{\text{KL}}(f|h) : \int f(\mathbf{x}) d\mathbf{x} = 1, \ E_f \left[ \sum_{j=1}^n Q(X_j) \right] = \zeta \right\}, \quad (5)$$

where  $\zeta$  a parameter that we shall specify. The solution to this program factorises:

$$\begin{aligned} f(\mathbf{x}) &= \frac{h(\mathbf{x}) \exp\left(\lambda \sum_{j=1}^n Q(x_j)\right)}{c(\lambda)} \\ &= \prod_{j=1}^n \frac{h_1(x_j) \exp(\lambda Q(x_j))}{c_1(\lambda)} = \prod_{j=1}^n f_1(x_j), \end{aligned}$$

with  $c_1(\lambda) = E_{h_1}[\exp(\lambda Q(X))]$  being the normalizing constant of the marginal density. Because the normalization constant factorizes as well, we get

$$\frac{c'(\lambda)}{c(\lambda)} = \zeta \quad \Leftrightarrow \quad \frac{c'_1(\lambda)}{c_1(\lambda)} = \frac{\zeta}{n}.$$

However, for  $0 \leq \lambda < 1$ :

$$c_1(\lambda) = E_{h_1}[e^{\lambda Q(X)}] = \int_0^\infty q(x) e^{-(1-\lambda)Q(x)} dx = \frac{1}{1-\lambda}.$$

Hence, we get

$$\frac{c'_1(\lambda)}{c_1(\lambda)} = \frac{1}{1-\lambda} = \frac{\zeta}{n},$$

and so the marginal becomes

$$\begin{aligned} f_1(x) &= \frac{h_1(x) e^{\lambda Q(x)}}{c_1(\lambda)} = (1-\lambda) q(x) e^{-(1-\lambda)Q(x)} \\ &= (n/\zeta) q(x) e^{-(n/\zeta)Q(x)}. \end{aligned}$$

This is exactly the hazard rate twisted density discussed in the previous section with

$$\frac{n}{\zeta} = 1 - \theta.$$

The optimal hazard rate parameter  $\theta = 1 - n/Q(\gamma)$  (according to [16]) is obtained by  $\zeta = Q(\gamma)$ .

### 4.3 Correlated hazard rate twisting

Next, we consider a minor adaptation in the constraint:

$$\inf_f \left\{ \mathcal{D}_{\text{KL}}(f|h) : \int f(\mathbf{x}) d\mathbf{x} = 1, E_f \left[ Q\left(\sum_{j=1}^n X_j\right) \right] = \zeta \right\}, \quad (6)$$

Note that

- The solution to this MinxEnt program does not factorise. We have

$$f(\mathbf{x}) = \frac{h(\mathbf{x}) \exp(\lambda Q(S(\mathbf{X})))}{c(\lambda)},$$

and thus the components of  $X_i$ 's are not independent.

- Because the cumulative hazard function  $Q(\cdot)$  is increasing and concave, we have for any density of the jumps

$$E[Q(S(\mathbf{X}))] = E \left[ Q \left( \sum_{j=1}^n X_j \right) \right] \leq E \left[ \sum_{j=1}^n Q(X_j) \right].$$

Consequently, the solution to the second MinxEnt program (6) must have a heavier tail than the hazard rate twisted solution to the first MinxEnt program (5).

- Applying the Markov inequality (for bounding below), and using the concavity of  $Q$  (for bounding above) we readily obtain the following inequalities

$$Q(\gamma)P_f(S(\mathbf{X}) > \gamma) \leq E_f[Q(S(\mathbf{X}))] \leq Q(E_f[S(\mathbf{X}))].$$

Since we wish to have  $E_f[S] \approx \gamma$  in the importance sampling simulations we set the right hand-side parameter  $\zeta$  in (6) as

$$\zeta = \rho Q(\gamma)$$

for some  $0 < \rho < 1$ , unspecified at this stage.

Our goal is to prove asymptotic optimality when using this correlated hazard rate twisted solution in importance sampling. Recall the normalizing constant

$$c(\lambda) = E_h[\exp(\lambda Q(S(\mathbf{X})))],$$

and Lagrange multiplier  $\lambda$  satisfying

$$(\log c(\lambda))' = \rho Q(\gamma). \tag{7}$$

We have the following properties.

**Lemma 1.**

- (i)  $c(\lambda) < \infty$  iff  $\lambda < 1$ .
- (ii)  $c(\lambda) \rightarrow \infty$  as  $\lambda \uparrow 1$ .
- (iii)  $(\log c(\lambda))' \rightarrow \infty$  as  $\lambda \uparrow 1$ .

Denote the solution to (7) by  $\lambda(\gamma)$ . Then

- (iv)  $\lambda(\gamma) \rightarrow 1$  as  $\gamma \rightarrow \infty$ .
- (v)  $\log c(\lambda(\gamma)) = o(Q(\gamma))$  as  $\gamma \rightarrow \infty$ .

*Proof.* (i) Necessity: suppose that  $\lambda \geq 1$ . Because all increments  $X_j$  are nonnegative and the cumulative hazard function  $Q(x)$  is nondecreasing, we have

$$\lambda Q(S(\mathbf{X})) = \lambda Q\left(\sum_{j=1}^n X_j\right) \geq Q(X_1).$$

And so, for  $\lambda \geq 1$ ,

$$\begin{aligned} c(\lambda) &= E_h[\exp(\lambda Q(S(\mathbf{X})))] \\ &\geq E_h[\exp(Q(X_1))] = E_{h_1}[\exp(Q(X_1))] \\ &= \int_0^\infty h_1(x)e^{Q(x)} dx = \int_0^\infty q(x) dx = \lim_{x \rightarrow \infty} Q(x) = \infty. \end{aligned}$$

Therefore,  $c(\lambda) < \infty$  implies  $\lambda < 1$ , and then we get (for  $0 \leq \lambda < 1$ ):

$$\begin{aligned} c(\lambda) &\geq E_{h_1}[\exp(\lambda Q(X_1))] \\ &= \int_0^\infty q(x)e^{-(1-\lambda)Q(x)} dx = \frac{1}{1-\lambda}. \end{aligned} \tag{8}$$

Sufficiency: since  $Q(x)$  is nondecreasing and concave the inequality

$$Q\left(\sum_{j=1}^n x_j\right) \leq \sum_{j=1}^n Q(x_j)$$

holds for all nonnegative  $x_j$ 's. Hence, applying that the  $X_j$ 's are i.i.d.,

$$\begin{aligned} c(\lambda) &= E_h[\exp(\lambda Q(S(\mathbf{X})))] = E_h\left[\exp\left(\lambda Q\left(\sum_{j=1}^n X_j\right)\right)\right] \\ &\leq E_h\left[\exp\left(\sum_{j=1}^n \lambda Q(X_j)\right)\right] = E_h\left[\prod_{j=1}^n \exp(\lambda Q(X_j))\right] \\ &= (E_h[\exp(\lambda Q(X_1))])^n = \left(\frac{1}{1-\lambda}\right)^n \end{aligned} \tag{9}$$

which is positive and finite for all  $\lambda < 1$ .

(ii) Follows immediately from (8).

(iii) In Appendix B we will show that for all  $0 \leq \lambda < 1$

$$\frac{1}{1-\lambda} \leq (\log c(\lambda))' \leq n \frac{1}{1-\lambda}. \quad (10)$$

(iv) The bounds in (10) show that a solution  $\lambda(\gamma)$  to  $(\log c(\lambda))' = \rho Q(\gamma)$  must increase to 1 as  $\gamma \rightarrow \infty$ .

(v) The lower bound in (10) says that a solution  $\lambda(\gamma)$  to  $(\log c(\lambda))' = \rho Q(\gamma)$  satisfies

$$\frac{1}{1-\lambda(\gamma)} \leq \rho Q(\gamma).$$

Apply the upper bound (9):

$$c(\lambda(\gamma)) \leq \left( \frac{1}{1-\lambda(\gamma)} \right)^n \leq (\rho Q(\gamma))^n.$$

Taking logarithms:

$$\log c(\lambda(\gamma)) \leq n \log \rho + n \log Q(\gamma).$$

And clearly  $(\log Q(\gamma))/(Q(\gamma)) \rightarrow 0$  when  $\gamma \rightarrow \infty$  because  $Q(\gamma) \rightarrow \infty$ .  $\square$

The main result follows.

**Theorem 1.** *The importance sampling estimator  $\bar{Y}(\gamma)$  of  $P(S(\mathbf{X}) > \gamma)$  using the MinxEnt solution  $f$  is asymptotically optimal.*

*Proof.* The importance sampling estimator based on a sample size  $k$  is

$$\bar{Y}(\gamma) = \frac{1}{k} \sum_{i=1}^k Y_i(\gamma),$$

with the  $Y_i(\gamma)$  ( $i = 1, \dots, k$ ) i.i.d. as

$$\begin{aligned} Y(\gamma) &= \frac{h(\mathbf{X})}{f(\mathbf{X})} 1\{S(\mathbf{X}) > \gamma\} \\ &= c(\lambda) \exp(-\lambda Q(S(\mathbf{X}))) 1\{S(\mathbf{X}) > \gamma\}. \end{aligned}$$

It is easy to see that

$$\liminf_{\gamma \rightarrow \infty} \frac{\log E_f[\bar{Y}^2(\gamma)]}{\log E_f[\bar{Y}(\gamma)]} \geq \liminf_{\gamma \rightarrow \infty} \frac{\log E_f[Y^2(\gamma)]}{\log E_f[Y(\gamma)]}.$$

Then

$$\begin{aligned}
E_f [Y(\gamma)^2] &= E_f \left[ \left( \frac{h(\mathbf{X})}{f(\mathbf{X})} \right)^2 1\{S(\mathbf{X}) > \gamma\} \right] \\
&= E_h \left[ \frac{h(\mathbf{X})}{f(\mathbf{X})} 1\{S(\mathbf{X}) > \gamma\} \right] \\
&= c(\lambda) E_h [\exp(-\lambda Q(S(\mathbf{X}))) 1\{S(\mathbf{X}) > \gamma\}] \\
&\leq c(\lambda) e^{-\lambda Q(\gamma)} P_h(S(\mathbf{X}) > \gamma) \\
&\sim nc(\lambda) e^{-\lambda Q(\gamma)} P_h(X_1 > \gamma) \\
&= nc(\lambda) e^{-\lambda Q(\gamma)} e^{-Q(\gamma)} \\
&= nc(\lambda) e^{-(\lambda+1)Q(\gamma)}.
\end{aligned}$$

And because

$$E_f[Y(\gamma)] = P_h(S(\mathbf{X}) > \gamma) \sim nP_h(X_1 > \gamma) = ne^{-Q(\gamma)},$$

we get by the nonnegativity of  $\log E_f[Y(\gamma)]$ :

$$\begin{aligned}
\frac{\log E_f [Y(\gamma)^2]}{\log E_f [Y(\gamma)]} &\gtrsim \frac{\log nc(\lambda) - (\lambda + 1)Q(\gamma)}{\log n - Q(\gamma)} \\
&= \frac{\log nc(\lambda)}{\log n - Q(\gamma)} - \frac{\lambda + 1}{\frac{\log n}{Q(\gamma)} - 1} \\
&= \frac{\log n}{\log n - Q(\gamma)} + \frac{\log c(\lambda)}{\log n - Q(\gamma)} + \frac{\lambda + 1}{1 - \frac{\log n}{Q(\gamma)}}.
\end{aligned}$$

Now let  $\gamma \rightarrow \infty$  and recall that  $n$  remains constant, and that  $\lambda = \lambda(\gamma)$ . The first term goes to zero because  $Q(\gamma) = -\log P(X > \gamma) \rightarrow \infty$ , the second term goes to zero because  $\log c(\lambda) = o(Q(\gamma))$  according to Lemma 1(v), and the denominator in the last factor goes to one because  $Q(\gamma) \rightarrow \infty$ . Hence, by Lemma 1(iv) we obtain

$$\liminf_{\gamma \rightarrow \infty} \frac{\log E_f [Y(\gamma)^2]}{\log \ell(\gamma)} \geq \liminf_{\gamma \rightarrow \infty} \frac{\lambda + 1}{1 - \frac{\log n}{Q(\gamma)}} = 2.$$

□

## 5 Generating from the correlated MinxEnt density

Generating samples  $\mathbf{X}$  from the correlated hazard rate twisted density  $f$  in the importance sampling simulations is not trivial because of the dependency



of the increments. We give three algorithms for which we need the following preliminaries. Because of the concavity of the cumulative hazard function  $Q(x)$ , it holds for any  $x_1, \dots, x_n > 0$ :

$$Q\left(\sum_{j=1}^n x_j\right) \leq \sum_{j=1}^n Q(x_j).$$

Furthermore, we denote the independent hazard rate twisted density of Sections 4.1 and 4.2 by

$$\begin{aligned} g(\mathbf{x}) &\stackrel{\text{def}}{=} (1 - \lambda)^n h(\mathbf{x}) \exp\left(\lambda \sum_{j=1}^n Q(x_j)\right) \\ &= \prod_{j=1}^n (1 - \lambda) h_1(x_j) \exp(\lambda Q(x_j)) = \prod_{j=1}^n g_1(x_j). \end{aligned}$$

Generating from this density is easy. For instance, when  $h_1 \sim \text{Weibull}(\kappa, \beta)$ , then  $g_1 \sim \text{Weibull}((1 - \lambda)\kappa, \beta)$ .

## 5.1 Algorithm 1: acceptance-rejection

We bound the density:

$$\begin{aligned} f(\mathbf{x}) &= \frac{h(\mathbf{x}) \exp\left(\lambda Q\left(\sum_{j=1}^n x_j\right)\right)}{c(\lambda)} \\ &\leq \frac{h(\mathbf{x}) \exp\left(\sum_{j=1}^n \lambda Q(x_j)\right)}{c(\lambda)} = K_1 g(\mathbf{x}), \end{aligned}$$

with the bounding constant  $K_1$ :

$$K_1 = \frac{1}{c(\lambda)} \frac{1}{(1 - \lambda)^n}.$$

When the vector  $\mathbf{x}$  is generated according to the (joint) density  $g$ , it is accepted with probability

$$\begin{aligned} p_1 &= \frac{f(\mathbf{x})}{K_1 g(\mathbf{x})} = f(\mathbf{x}) \frac{1}{K_1} \frac{1}{g(\mathbf{x})} \\ &= \frac{h(\mathbf{x}) \exp\left(\lambda Q\left(\sum_{j=1}^n x_j\right)\right)}{c(\lambda)} \left(c(\lambda) (1 - \lambda)^n\right) \frac{1}{(1 - \lambda)^n h(\mathbf{x}) \exp\left(\lambda \sum_{j=1}^n Q(x_j)\right)} \\ &= \exp\left(\lambda \left(Q\left(\sum_{j=1}^n x_j\right) - \sum_{j=1}^n Q(x_j)\right)\right). \end{aligned}$$

## 5.2 Algorithm 2: Metropolis-Hastings

We construct a Markov chain  $\{\mathbf{X}(t) : t = 0, 1, \dots\}$  on  $\mathbb{R}_{\geq 0}^n$  according to an independent Metropolis-Hastings algorithm. Suppose  $\mathbf{X}(t) = \mathbf{x}$ . Then we generate a random variate  $\mathbf{Y}$  on  $\mathbb{R}_{\geq 0}^n$  from the independent hazard rate twisted density  $g(\cdot)$  and set

$$\mathbf{X}(t+1) = \begin{cases} \mathbf{Y} & \text{with probability } \min \left\{ \frac{f(\mathbf{Y})g(\mathbf{x})}{f(\mathbf{x})g(\mathbf{Y})}, 1 \right\}, \\ \mathbf{x} & \text{otherwise} \end{cases}$$

The chain is iterated until convergence. Notice that the acceptance probability is

$$p_2 = \frac{f(\mathbf{y})g(\mathbf{x})}{f(\mathbf{x})g(\mathbf{y})} = \frac{\exp \left( \lambda \left( Q \left( \sum_{j=1}^n y_j \right) - \sum_{j=1}^n Q(y_j) \right) \right)}{\exp \left( \lambda \left( Q \left( \sum_{j=1}^n x_j \right) - \sum_{j=1}^n Q(x_j) \right) \right)}. \quad (11)$$

## 5.3 Algorithm 3: Gibbs sampler

We construct a Markov chain  $\{\mathbf{X}(t) : t = 0, 1, \dots\}$  on  $\mathbb{R}_{\geq 0}^n$  according to the Gibbs sampler method, which means that we generate consecutively from conditional densities. Suppose  $\mathbf{X}(t) = \mathbf{x}^{(t)} = (x_1^{(t)}, \dots, x_n^{(t)})$ . Then for  $j = 1, 2, \dots, n$  the  $j$ -th component  $X_j(t+1)$  is generated from

$$f_j(x_j | X_1^{(t+1)}, \dots, X_{j-1}^{(t+1)}, x_{j+1}^{(t)}, \dots, x_n^{(t)}).$$

The chain is iterated until convergence. To work out the conditional density, we set  $s_j = \sum_{i \neq j} x_i$ , use the concavity of the cumulative hazard function  $Q(\cdot)$ , and use the factorization of the original density  $h(\mathbf{x}) = \prod_{i=1}^n h_1(x_i)$ :

$$\begin{aligned} f_j(x_j | x_i, i \neq j) &= \frac{f(\mathbf{x})}{\int_0^\infty f(\mathbf{x}) dx_j} \\ &= \frac{h(\mathbf{x}) \exp \left( \lambda Q(x_j + s_j) \right)}{\int_0^\infty h(\mathbf{x}) \exp \left( \lambda Q(x_j + s_j) \right) dx_j} \\ &= \frac{h_1(x_j) \exp \left( \lambda Q(x_j + s_j) \right)}{\int_0^\infty h_1(x_j) \exp \left( \lambda Q(x_j + s_j) \right) dx_j} \\ &\leq h_1(x_j) \exp(\lambda Q(x_j)) \frac{\exp(\lambda Q(s_j))}{\int_0^\infty h_1(x_j) \exp \left( \lambda Q(x_j + s_j) \right) dx_j} \\ &= K_3 g_1(x_j). \end{aligned}$$

The bounding constant  $K_3$  is

$$K_3 = \frac{\exp(\lambda Q(s_j))}{(1 - \lambda) \int_0^\infty h_1(x_j) \exp(\lambda Q(x_j + s_j)) dx_j} \leq \frac{1}{1 - \lambda}, \quad (12)$$

where the inequality follows from  $Q(x_j + s_j) \geq Q(s_j)$  and from  $h_1$  being a probability density. The acceptance-rejection algorithm is applied to generate from the conditional density, namely by generating  $x_j$  from  $g_1(\cdot)$  and accepting it with probability

$$\begin{aligned} p_3 &= \frac{f_j(x_j | x_i, i \neq j)}{K_3 g_1(x_j)} \\ &= \frac{h_1(x_j) \exp(\lambda Q(x_j + s_j))}{\int_0^\infty h_1(x_j) \exp(\lambda Q(x_j + s_j)) dx_j} \frac{1}{K_3} \frac{1}{(1 - \lambda) h_1(x_j) \exp(\lambda Q(x_j))} \\ &= \exp\left(\lambda(Q(x_j + s_j) - Q(s_j) - Q(x_j))\right). \end{aligned} \quad (13)$$

## 5.4 Convergence issues

The advantage of algorithm 1 is that it gives independent observations, but a closer look at its details shows that the acceptance probability becomes very small when  $Q(\sum_{j=1}^n x_j)$  is much less than  $\sum_{j=1}^n Q(x_j)$ . And this will occur in most cases. For instance an experiment with  $n = 5$  Weibull( $\kappa = 1, \beta = 0.5$ ) random variables gave on average about 1500 iterations before an accepted observation.

For the other two algorithms we need to assess the rate of convergence of the Markov chain to its stationary regime.

- The independent Metropolis-Hastings algorithm.

In Section 5.1 we showed  $f(\mathbf{x}) \leq K_1 g(\mathbf{x})$  for all  $\mathbf{x} \in \mathbb{R}_{\geq 0}^n$ , with, according to (8),

$$K_1 = \frac{1}{c(\lambda)(1 - \lambda)^n} \leq \frac{1}{(1 - \lambda)^{n-1}}.$$

Thus, we can bound the total variation of the difference between the  $t$ -th power of the transition probability kernel and the stationary density of the Markov chain  $\{X(t), t \geq 0\}$  [22], [24, section 7.4]:

$$\|P^t(\mathbf{x}, \cdot) - f(\cdot)\|_{\text{TV}} \leq 2 \left(1 - \frac{1}{K_1}\right)^t \leq 2(1 - (1 - \lambda)^{n-1})^t.$$

- The Gibbs sampler algorithm.

In this algorithm we deal with both acceptance probabilities and the burn-in time. From equation (12) we see that the expected number of iterations until acceptance is less than  $1/(1 - \lambda)$  per component. Furthermore, let  $\mathbf{X}(t) = \mathbf{x}$ , then the probability density that  $\mathbf{X}(t + 1) = \mathbf{y}$  is

$$\begin{aligned}
P(\mathbf{x}, \mathbf{y}) &= \prod_{j=1}^n f_j(y_j | y_1, \dots, y_{j-1}, x_{j+1}, \dots, x_n) \\
&= \prod_{j=1}^n \frac{h_1(y_j) \exp(\lambda Q(y_j + s_j))}{\int_0^\infty h_1(y_j) \exp(\lambda Q(y_j + s_j)) dy_j} \\
&\geq \prod_{j=1}^n \frac{h_1(y_j) \exp(\lambda Q(s_j))}{\int_0^\infty h_1(y_j) \exp(\lambda Q(y_j)) \exp(\lambda Q(s_j)) dy_j} \\
&= \frac{\prod_{j=1}^n h_1(y_j)}{\prod_{j=1}^n \int_0^\infty h_1(y_j) \exp(\lambda Q(y_j)) dy_j} \\
&= (1 - \lambda)^n h(\mathbf{y}).
\end{aligned}$$

Thus, we can bound the total variation of the difference between the  $t$ -th power of the transition probability kernel and the stationary density of the Markov chain  $\{X(t), t \geq 0\}$  [22], [23, chapter 16]:

$$\|P^t(\mathbf{x}, \cdot) - f(\cdot)\|_{\text{TV}} \leq (1 - (1 - \lambda)^n)^t.$$

Notice that we have established geometric convergence of the algorithms 2 and 3, though the bounding rate is close to 1 when  $\lambda$  is close to 1:  $1 - (1 - \lambda)^n$ . However, usually the convergence of the underlying Markov chains goes much more rapidly than based on these (loose) bounds [8]. From experiments we experienced small acceptance probabilities (11) in the independent Metropolis-Hastings algorithm. This would mean that to obtain independent samples  $\mathbf{X}(t_i)$  the subsampling rate must be small, that is, large subsampling periods  $t_{i+1} - t_i$ . This is reflected in the autocorrelation of the time series  $\{S(t) : t = 0, 1, \dots, T\}$ , where  $S(t) = \sum_{j=1}^n X_j(t)$ . For instance, the following autocorrelation plot was obtained by a simulation of the model with

$n = 5$  Weibull( $\kappa = 1, \beta = 0.5$ ) random variables, and horizon  $T = 10000$  after a burn-in period of 1000 iterations. The Lagrange multiplier  $\lambda$  was set to 0.9. The horizontal axis contains the lags 1–200.

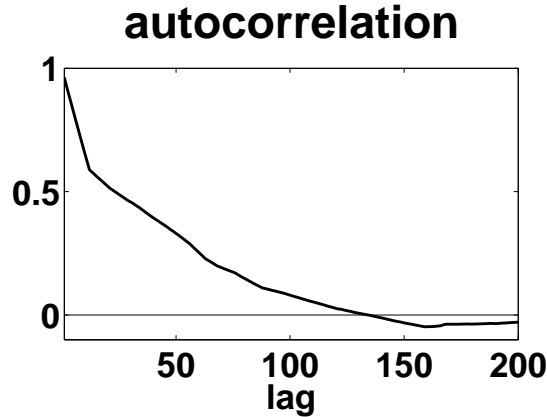


Figure 3. Autocorrelation of the partial sums in the Metropolis-Hastings algorithm with lags up to 200.

## 5.5 Convergence diagnostics of the Gibbs sampler

For analysing the Gibbs sampler (algorithm 3), we applied the following empirical diagnostics to the time series  $\{S(t), t = 0, 1, \dots, T\}$ , where  $S(t) = \sum_{j=1}^n X_j(t)$  is the sum of the increments, and  $T$  the simulation horizon of the Gibbs sampler.

1. The Heidelberger-Welch procedure for determining the burn-in period [12].
2. An autocorrelation plot for determining the dependency structure.
3. The Kolmogorov-Smirnov test for testing stationarity.

These tests gave satisfactory clues that the chain is mixing rapidly, and that the samples obtained with a high subsampling rate may be considered to be stationary and (almost) independent. All tests are executed with Weibull( $\kappa = 1, \beta = 0.5$ ) increments and Lagrange multiplier  $\lambda$  set to 0.9.

### The burn-in period

We iterated the Gibbs sampler  $T = 2000$  times, discarded the first 10%, 20%, etc. of the iterations and calculated the resulting Cramer-von

Mises statistic (CVM). When it passes the stationarity test, the length of the discarded portion may be considered as the transient or burn-in period. We also estimated the associated relative half-width of the 95%-confidence intervals (RHW) in percentages.

burn-in	$n = 5$		$n = 10$		$n = 20$	
	CVM	RHW	CVM	RHW	CVM	RHW
0	0.525	7.44	0.326	8.12	0.217	5.20
200	0.212	7.98	0.110	6.14	0.135	5.05
400	0.072	7.89	0.038	6.71	0.096	5.67
600	0.034	7.74	0.049	6.91	0.148	7.03
800	0.079	8.67	0.038	9.68	0.126	9.04

*Table 1. The Cramer-von Mises statistic (CVM) and relative half-width of the 95%-confidence intervals (RHW).*

The 95% critical value of the CVM is approximately 0.45, thus to be on a save side, we decided upon a burn in of 400 iterations of the Gibbs sampler.

### **Autocorrelation function**

Similarly to the Metropolis-Hastings, we constructed autocorrelation plots for  $n = 5, 10, 20$  Weibull( $\kappa = 1, \beta = 0.5$ ) random variables, horizon  $T = 5000$  after a burn-in period of 400 iterations. These plots in Figures 4-6 show clearly that consecutive samples are weakly dependent.

### **Kolmogorov-Smirnov test**

We applied the Kolmogorov-Smirnov test for testing stationarity of the sampled time series. After a burn-in period of 400 iterations, we continued with  $T = 2000$  more iterations. After each 20 iterations we applied the two-sample Kolmogorov-Smirnov test for comparing the first and second halves. In Figures 4-6 we give plots of the computed p-values in case of  $n = 5, 10, 20$  Weibull( $\kappa = 1, \beta = 0.5$ ) random variables. The p-values are well above the 0.05 uncertainty.

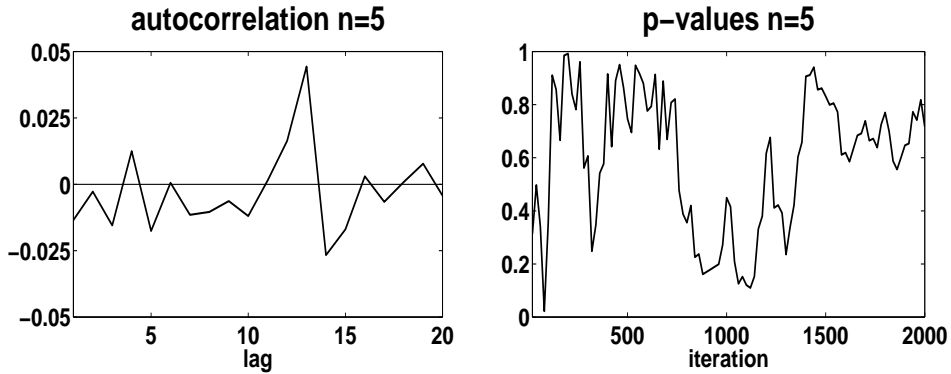


Figure 4. Dimension  $n = 5$ . Left: autocorrelation of the partial sums in the Gibbs sampler with lags up to 20. Right: p-values.

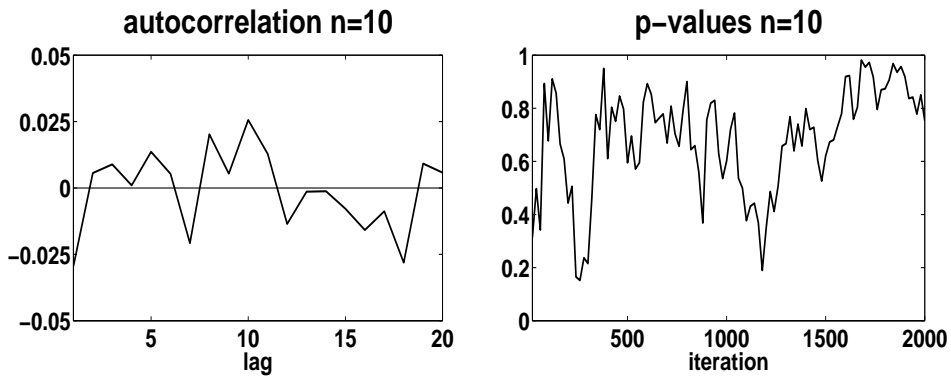


Figure 5. Dimension  $n = 10$ . Left: autocorrelation of the partial sums in the Gibbs sampler with lags up to 20. Right: p-values.

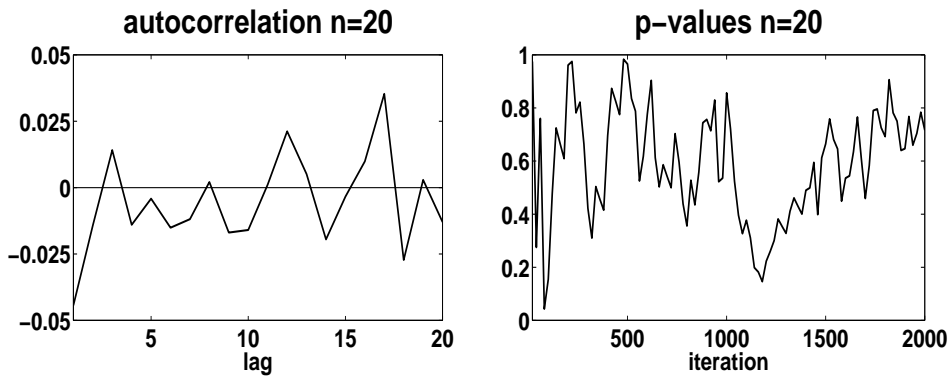


Figure 6. Dimension  $n = 20$ . Left: autocorrelation of the partial sums in the Gibbs sampler with lags up to 20. Right: p-values.

## 6 Simulation experiments

We executed simulation runs for increments  $X$  distributed according to

- Weibull( $\kappa = 1, \beta \in (0, 1)$ ), i.e.,

$$h_1(x) = \beta x^{\beta-1} e^{-x^\beta}, \quad \bar{H}_1(x) = P_{h_1}(X > x) = e^{-x^\beta}.$$

- Pareto( $\kappa = 1, \alpha > 0$ ), i.e.,

$$h_1(x) = \alpha(1+x)^{-\alpha-1}, \quad \bar{H}_1(x) = P_{h_1}(X > x) = (1+x)^{-\alpha}.$$

- Lognormal( $\mu = 1, \sigma^2$ ), i.e.,  $X = e^{\mu + \sigma Z}$  with  $Z$  standard Gaussian.

We considered a range of sample sizes  $n$ , target levels  $\gamma$ , and shape parameters  $\beta, \alpha$  and  $\sigma$ , respectively. We applied the Gibbs sampler of Section 5.3 to generate the samples of the correlated MinxEnt densities with the appropriate burn-in periods determined as we explained in Section 5.5.

We compared our results (RR) with those obtained by two other algorithms:

1. JS: the original i.i.d. hazard rate twisted importance sampling method of [16] (see also Section 4.2).
2. AK: the conditional Monte-Carlo importance sampling algorithm of [2]. This method is based on

$$\begin{aligned} P(S_n > \gamma) &= nP(S_n > \gamma; M_n = X_n) \\ &= nE \left[ P(S_n > \gamma; M_n = X_n \mid X_1, \dots, X_{n-1}) \right] \\ &= nE \left[ \bar{H}_1 \left( \max(M_{n-1}, \gamma - S_{n-1}) \right) \right], \end{aligned}$$

where  $M_n \stackrel{\text{def}}{=} \max\{X_1, \dots, X_n\}$ .

The target level  $\gamma$  in the experiments is set such that the asymptotic approximation

$$P(X_1 + \dots + X_n > \gamma) \sim nP(X > \gamma) = n\bar{H}_1(\gamma) = 10^{-r},$$

i.e.,  $\gamma = \bar{H}_1^{-1}(10^{-r}/n)$  for varying  $r$ .

Our algorithm needs the Lagrange multiplier  $\lambda$  as a function of the righthand side constraint  $\rho Q(\gamma)$  with  $0 < \rho < 1$ , see Section 4.3. However, from Lemma 1 we know that  $\lambda \uparrow 1$  as  $\gamma \rightarrow \infty$ , for any  $0 < \rho < 1$ . And from the proof of



Theorem 1 we see that the ratio  $\log E_f[Y(\gamma)^2]/\log E_f[Y(\gamma)]$  is asymptotically at least  $1 + \lambda$ . Thus it suffices to implement the algorithm with  $\lambda$  chosen to be close to 1. On the other hand, the Gibbs sampler algorithm uses an acceptance-rejection procedure with an acceptance probability getting smaller as  $\lambda$  gets larger, see (13). The convergence diagnostics of Section 5.5 were executed with  $\lambda = 0.9$ , but we decided to execute the actual importance sampling simulations with  $\lambda = 0.8$ . This choice increases acceptance and convergence, but might give slightly worse efficiency.

After each simulation experiment we collect three (estimated) performance measures of the estimators:

- RHW: the relative half width of the 95% confidence interval

$$1.96\sqrt{\text{Var}[Y]}/Y \quad (\text{in percentages}).$$

- RAT: the logarithmic efficiency ratio

$$\log E_f[Y^2]/\log E_f[Y].$$

- EFF: the ( $-\log$ arithm of the) effort

$$-\log_{10}(\text{Var}[Y] \times \text{CPU}[Y]).$$

Better performance is obtained by smaller RHW, higher RAT, and larger EFF. Notice that to compare the RAT and EFF measures we do not need to execute the three methods with the same sample size  $k$ , assuming that  $k$  is large enough (meaning RHW small enough) so that the resulting estimates are reliable. We do need the same sample size for comparing RHW. Empirically we found that the JS importance sampling method requires much longer sample sizes to obtain the same RHW when the tails become heavier. On the contrary, the AK method requires longer sample sizes when the tails become lighter. In stead of setting equal sample sizes, we decided to simulate with different sample sizes until we got sufficiently small RHW, and compensated the RHW by a correction factor, since for the theoretical RHW:

$$\sqrt{k} \times \text{RHW of } k \text{ samples} = \sqrt{k'} \times \text{RHW of } k' \text{ samples}.$$

## Weibull

We varied the number of increments  $n = 5, 10, 15, 20$ , and  $r = 6, 9, 12, 15$  of the target probability  $10^{-r}$ . Table 2 presents the performance measures of the associated estimators, RHW and EFF are averaged over these 16 scenarios, while RAT is averaged over the 4 scenarios at the highest level  $r = 15$ . This has been done for each of the three choices of Weibull shape parameter  $\beta = 0.25, 0.5, 0.75$ . The larger  $\beta$ , the lighter the tail becomes.

$\beta$	RHW			RAT			EFF		
	RR	JS	AK	RR	JS	AK	RR	JS	AK
0.25	7.98	697.62	0.01	1.77	1.56	2.00	22.33	20.21	29.75
0.50	11.30	350.93	0.32	1.74	1.63	1.99	21.60	20.74	25.99
0.75	6.07	14.80	17.50	1.74	1.69	1.72	18.02	17.99	18.18

*Table 2. Average performances of the three importance sampling algorithms for Weibull increments.*

## Pareto

We varied the number of increments  $n = 5, 10, 15$ , and  $r = 6, 9, 12, 15$  of the target probability  $10^{-r}$  (for  $n = 20$  the problem in the JS method remains rare and demands a huge sample size). The Pareto shape parameter was chosen to be  $\alpha = 0.5$  (infinite mean and variance),  $\alpha = 1.5$  (finite mean and infinite variance),  $\alpha = 5.0$  (finite mean and variance). The larger  $\alpha$ , the lighter the tail becomes. Before we executed the importance sampling algorithms we analysed the convergence diagnostics of the Gibbs sampler as we explained in Section 5.5. We found a faster mixing of the sampler and even less correlation between samples than with the Weibull increments.

$\beta$	RHW			RAT			EFF		
	RR	JS	AK	RR	JS	AK	RR	JS	AK
0.5	5.23	237.39	0.01	1.77	1.61	2.00	22.70	21.03	37.49
1.5	5.26	227.82	0.01	1.77	1.59	2.00	22.45	20.95	36.81
5.0	13.12	132.28	0.02	1.71	1.62	2.00	21.74	21.08	29.10

*Table 3. Average performances of the three importance sampling algorithms for Pareto increments.*

## Lognormal

We varied the number of increments  $n$ , and the target probability  $10^{-r}$  as in the Pareto case. The Lognormal shape parameter was chosen to be  $\sigma^2 = 1, 4, 9$ . The larger  $\sigma$ , the heavier the tail becomes. The convergence diagnostics of Section 5.5 gave a slightly slower mixing of the sampler, but again a weak correlation between samples after the burn-in period.

$\sigma^2$	RHW			RAT			EFF		
	RR	JS	AK	RR	JS	AK	RR	JS	AK
1.0	10.80	169.60	0.02	1.72	1.59	2.00	21.51	20.64	28.21
4.0	5.56	162.62	0.01	1.76	1.59	2.00	22.21	20.69	32.04
9.0	5.02	209.09	0.01	1.77	1.59	2.00	22.22	20.81	35.18

*Table 4. Average performances of the three importance sampling algorithms for Lognormal increments.*

In all cases we observe that our MinxEnt solution improves considerably the independent hazard rate twisted solution. This comes as no surprise considering that the tail of the correlated density is heavier than of the independent density as we explained in Section 4.3. On the other hand, in almost all cases our algorithm is outperformed by the conditional Monte Carlo method. But notice that Table 2 with the Weibull results shows empirically that the conditional Monte Carlo method degrades when the tails become lighter (the method is asymptotically optimal for  $\beta$  below some critical level  $\beta^*$  [2]), whereas our algorithm remains stable.

## 7 Conclusion and further research

In this paper we investigated the idea of applying the minimum cross-entropy method (MinxEnt) for estimating rare event probabilities for the sum of i.i.d. random variables. We saw that some existing importance sampling methods for determining the new simulation densities can be cast in a Kullback-Leibler MinxEnt program, such as the large deviations approach for light tails and the hazard rate twisting for heavy tails. However, the MinxEnt approach allows for generalizations or extensions in several directions, viz. other divergence

measures as objective, other constraints, more constraints. Our investigations of these other directions yielded until now limited successes. Our investigations were restricted in the sense that we considered MinxEnt programs with a single (non-trivial) constraint, and thus we suggest as for further research to consider programs with possibly other divergence measures, in combination with more and other constraints. This idea has been studied recently in [5] in the context of density estimation. Currently we are investigating the generalization to rare events involving a random number of increments and to long waiting times in queues. The main challenge is to find the appropriate divergence measure and constraints for which the MinxEnt program can be solved and for which the solution admits a fast algorithm to generate samples.

## References

- [1] Abbas, A.E., Maximum entropy utility, *Operations Research*, 54, 277 – 290, 2006.
- [2] Asmussen S. and D.P. Kroese, Improved algorithms for rare-event simulation with heavy tails, *Advances of Applied Probability*, 38, 545 – 558, 2006.
- [3] Asmussen S. and R.Y. Rubinstein, Steady-state rare-events simulation in queueing models and its complexity properties, *Advances in Queueing: Models, Methods and Problems* (ed. J. Dshalalow), 429 – 466, CRC Press, 1995.
- [4] Berger A.L., S.D. Pietra and V.J.D. Pietra, A maximum entropy approach to natural language processing, *Computational Linguistics*, 22, 39 – 71, 1996.
- [5] Botev Z.I., D.P. Kroese and T. Taimre, Generalized cross-entropy methods for rare events and optimization, In *Proceedings of the 6-th International Workshop on Rare Event Simulation* (ed. W. Sandmann), 1 – 30, Bamberg, 2006.

- [6] Bouzouba K. and L. Radouane, Image identification and estimation using the maximum entropy principle, *Pattern Recognition Letters*, 21, 691 – 700, 2000.
- [7] Bucklew J.A., *Introduction to rare-event simulation*, Springer, 2004.
- [8] Cowles M.K. and B.P. Carlin, Markov Chain Monte Carlo convergence diagnostics: a comparative review, *Journal of the American Statistical Association*, 91, 883-904, 1996.
- [9] Embrechts P, C. Klüppelberg and T. Mikosch, *Modeling extremal events*, Springer Verlag, 1997.
- [10] Glassermann P. and Y. Wang, Counter examples in importance sampling for large deviations probabilities, *Annals of Applied Probability*, 7, 731 – 746, 1997.
- [11] Heidelberger P., Fast simulation of rare-events in queueing and reliability models, *ACM Transactions on Modeling and Computer Simulation*, 5, 43 – 85, 1995.
- [12] Heidelberger P. and P.D. Welch, Simulation run length control in the presence of an initial transient, *Operations Research*, 31, 1109-1144, 1983.
- [13] Jaynes E.T., Information theory and statistical mechanics, *Physical Review*, 106, 620 – 630, 1957.
- [14] Jaynes E.T., Information theory and statistical mechanics, in *Statistical Physics* (ed. K. Ford), 181 – 218, Benjamin Inc., 1963.
- [15] Johansson M. and M. Sternad, Resource allocation under uncertainty using the maximum entropy principle, *IEEE Transaction on Information Theory*, 51, 4103 – 4117, 2005.
- [16] Juneja S. and P. Shahabuddin, Simulating heavy tailed processes using delayed hazard rate, *ACM Transactions on Modeling and Computer Simulation*, 12, 94 – 118, 2002.

- [17] Juneja S. and P. Shahabuddin, Rare-event simulation techniques: an introduction and recent advances. To appear in *Handbook of Simulation* (eds. S. Henderson and B. Nelson), Elsevier.
- [18] Kapur J.N. and H.K. Kesavan, *Entropy Optimization with Applications*, Academic Press, Inc., 1992.
- [19] Kullback S. and M.A. Khairat, A note on minimum discrimination information, *Annals of Mathematical Statistics*, 37, 279 – 280, 1966.
- [20] Laplace P.S., *Theorie Analytique des Probabilities*, 1812.
- [21] Lehtonen T. and H. Nyrrhinen, Simulating level-crossing probabilities by importance sampling, *Advances in Applied Probability*, 24, 858 – 874, 1992.
- [22] Mengersen K.L. and R.L. Tweedie, Rates of convergence of the hastings and Metropolis algorithms, *Annals of Statistics*, 24, 101-121, 1996.
- [23] Meyn S.P. and R.L. Tweedie, *Markov Chains and Stochastic Stability*, Springer, 1993.
- [24] Robert C.P. and G. Casella, *Monte Carlo Statistical Methods*, Second edition, Springer, 2004.
- [25] Rubinstein R.Y, A stochastic minimum cross-entropy method for combinatorial optimization and rare event estimation, *Methodology and Computing in Applied Probability*, 1, 1 – 46, 2005.
- [26] Rubinstein R.Y. and D.P. Kroese, *The cross-entropy method: a unified approach to combinatorial optimization, Monte-Carlo simulation and machine learning*, Springer, 2004.
- [27] Sadowsky J.S., On the optimality and stability of exponential twisting in Monte Carlo estimation, *IEEE Transactions on Information Theory*, 39, 119 – 128, 1993.
- [28] Sadowsky J.S., On Monte Carlo estimation of large deviations probabilities, *Annals of Applied Probability*, 6, 399 – 422, 1996.
- [29] Shannon, C.E., A mathematical theory of communication, *Bell System Technical Journal*, 27, 379 – 423, 623 – 656, 1948.

- [30] Trovato M. and L. Reggiani, Maximum entropy principle for hydrodynamic transport in semiconductor devices, *Journal of Applied Physics*, 85, 4050 – 4065, 1999.

## Appendix A

In this appendix we solve the MinxEnt program

$$\inf_f \left\{ \mathcal{D}_{\text{KL}}(f|h) : \int f(\mathbf{x}) d\mathbf{x} = 1, E_f[S_n^2] = \rho n^2 \right\},$$

when the increments  $X_j$  are i.i.d. standard normals, and  $S_n = \sum_{j=1}^n X_j$ . Denote  $S(\mathbf{x}) = \sum_{j=1}^n x_j$ , denote by  $I$  the  $n \times n$  identity matrix, and by  $U$  the  $n \times n$  matrix of all ones. Notice that

$$S^2(\mathbf{x}) = \left( \sum_{j=1}^n x_j \right)^2 = \mathbf{x}^T U \mathbf{x}.$$

Thus, the solution to the MinxEnt program is

$$\begin{aligned} f(\mathbf{x}) &= \frac{1}{c(\lambda)} h(\mathbf{x}) e^{\lambda S^2(\mathbf{x})} \\ &= \frac{1}{c(\lambda)} \frac{1}{(\sqrt{2\pi})^n} e^{-\frac{1}{2} \mathbf{x}^T I \mathbf{x}} e^{\lambda \mathbf{x}^T U \mathbf{x}} \\ &= \frac{1}{(\sqrt{2\pi})^n} \frac{1}{c(\lambda)} e^{-\frac{1}{2} \mathbf{x}^T (I - 2\lambda U) \mathbf{x}}. \end{aligned}$$

We recognize a multivariate normal density if we set  $\Sigma^{-1} \stackrel{\text{def}}{=} I - 2\lambda U$  as the inverse of the variance-covariance matrix  $\Sigma$  and the constant  $c(\lambda)$  as the square root of its determinant:  $c(\lambda) = \sqrt{|\Sigma|}$ . Let the scalar  $\tau = \tau(\lambda)$  be

$$\tau = \frac{2\lambda}{1 - 2\lambda n}.$$

Then, since  $(I - 2\lambda U)(I + \tau U) = I$  we obtain the variance-covariance matrix  $\Sigma = I + \tau U$ . Assuming

$$0 < \lambda < \frac{1}{2n},$$

the scalar  $\tau$  is positive, and the matrix  $\Sigma = I + \tau U$  is symmetric and positive definite. The next issue is to determine the determinant  $|\Sigma| = \det(\Sigma)$ . We

found  $\det(I + \tau U) = 1 + n\tau$ . We do not know whether this is a known result, we found it by solving a recursion relation for these determinants: we denote  $d_n = \det(I + \tau U)$  when the matrices  $I$  and  $U$  have dimension  $n$ . Then from the classical way of calculating a determinant:

$$d_n = (n - 1)\tau d_{n-2} - ((n - 2)\tau - 1)d_{n-1}.$$

It is easy to check that  $d_n = 1 + n\tau$  satisfies this relation ( $n \geq 1$ ). Thus

$$|\Sigma| = 1 + n\tau = 1 + n \frac{2\lambda}{1 - 2\lambda n} = \frac{1}{1 - 2\lambda n}. \quad (14)$$

Finally we can determine the Lagrange multiplier  $\lambda$  by solving  $(\log c(\lambda))' = \rho n^2$ . With  $c(\lambda) = \sqrt{|\Sigma|}$ , and  $|\Sigma|$  given in (14) we readily solve the equation and obtain

$$2\lambda n = 1 - \frac{1}{\rho n}. \quad (15)$$

Substitution in (14) gives  $|\Sigma| = \rho n$ . In order the Lagrange multiplier  $\lambda$  being positive we require

$$\rho > \frac{1}{n}.$$

### Complexity of the importance sampling estimator

The density  $f(\mathbf{x})$  is used as the importance sampling density to estimate

$$\ell(n) \stackrel{\text{def}}{=} P(S_n \leq -na(1 + \epsilon) \text{ or } S_n \geq na),$$

where  $a > 0$  and  $\epsilon > 0$ . Let us calculate the likelihood ratio:

$$\begin{aligned} L(\mathbf{X}) &\stackrel{\text{def}}{=} \frac{h(\mathbf{X})}{f(\mathbf{X})} = c(\lambda) e^{-\lambda S_n^2} \\ &= \sqrt{\rho n} e^{-\lambda S_n^2}. \end{aligned}$$

Denote the two intervals that define the rare event as  $B \stackrel{\text{def}}{=} }(-\infty, -na(1 + \epsilon)] \cup [na, \infty)$ . The importance sampling estimator is an average of i.i.d. copies of  $Y(n) \stackrel{\text{def}}{=} L(\mathbf{X})1\{S_n \in B\}$ . We shall determine its first two moment  $E_f[Y(n)]$  and  $E_f[Y^2(n)]$ . Firstly we notice that  $S_n$  being a linear transformation of the



multivariate Gaussian variables  $X_1, \dots, X_n$ , is itself Gaussian with zero mean and variance  $\text{Var}_f[S_n] = E_f[S_n^2] = \rho n^2$ . Denote this density by  $g(s)$ . Then

$$\begin{aligned} E_f[Y(n)] &= \int_B \sqrt{\rho n} e^{-\lambda s^2} g(s) ds \\ &= \int_B \frac{1}{\sqrt{2\pi}} \frac{1}{\sqrt{n}} e^{-\lambda s^2} e^{-\frac{1}{2\rho n^2} s^2} ds \\ &= \int_B \frac{1}{\sqrt{2\pi}} \frac{1}{\sqrt{n}} e^{-\frac{1}{2n} s^2} ds \end{aligned}$$

where we have used (15) to get

$$\lambda + \frac{1}{2\rho n^2} = \frac{1}{2n}.$$

Hence,

$$\begin{aligned} E_f[Y(n)] &= P(N(0, n) \leq -na(1 + \epsilon)) + P(N(0, n) \geq na) \\ &\geq 2P(N(0, n) \leq -na(1 + \epsilon)) = 2P(N(0, 1) \leq -a(1 + \epsilon)\sqrt{n}) \\ &= 2\Phi(-a(1 + \epsilon)\sqrt{n}) = 2(1 - \Phi(a(1 + \epsilon)\sqrt{n})). \end{aligned}$$

Almost in the same way, omitting the details:

$$E_f[Y^2(n)] = KP(N(0, n\tilde{\sigma}^2) \in B),$$

with

$$K = \frac{\rho n}{\sqrt{2\rho n - 1}}, \quad \tilde{\sigma}^2 = \frac{\rho n}{2\rho n - 1}.$$

Thus,

$$\begin{aligned} P(N(0, n\tilde{\sigma}^2) \in B) &\leq 2P(N(0, n\tilde{\sigma}^2) \geq na) = 2P\left(N(0, 1) \geq \frac{a}{\tilde{\sigma}}\sqrt{n}\right) \\ &= 2\left(1 - \Phi\left(\frac{a}{\sqrt{\rho}}\sqrt{2\rho n - 1}\right)\right). \end{aligned}$$

In order to bound the logarithmic ratio we apply a well known inequality for the tail of the standard normal distribution:

$$\frac{x^2 - 1}{x^2} \frac{1}{x} \phi(x) \leq 1 - \Phi(x) \leq \frac{1}{x} \phi(x), \quad (x > 0).$$

We use the upper bound with  $x = a\sqrt{2\rho n - 1}/\sqrt{\rho}$  in  $E_f[Y^2(n)]$ :

$$\begin{aligned} \log E_f[Y^2(n)] &\leq \log \frac{2K}{x} + \log \phi(x) \\ &= \log 2na\sqrt{\rho} - \frac{1}{2} \log 2\pi - \frac{1}{2} \frac{a^2(2\rho n - 1)}{\rho}. \end{aligned}$$

We use the lower bound with  $x = a(1 + \epsilon)\sqrt{n}$  in  $E_f[Y(n)]$ , thus we may set  $(x^2 - 1)/x^2 \geq \frac{1}{2}$ , assuming that  $n$  is sufficiently large:

$$\log E_f[Y(n)] \geq -\log a(1 + \epsilon)\sqrt{n} - \frac{1}{2} \log 2\pi - \frac{1}{2}a^2(1 + \epsilon)^2n.$$

Consider their ratio, take  $\frac{1}{n}$  of numerator and denominator, let  $n \rightarrow \infty$ , and take into account that the denominator is negative:

$$\begin{aligned} \liminf_{n \rightarrow \infty} \frac{\frac{1}{n} \log E_f[Y^2(n)]}{\frac{1}{n} \log E_f[Y(n)]} &\geq \frac{\limsup_{n \rightarrow \infty} \frac{1}{n} \log E_f[Y^2(n)]}{\liminf_{n \rightarrow \infty} \frac{1}{n} \log E_f[Y(n)]} \\ &\geq \frac{-\frac{1}{2} 2a^2}{-\frac{1}{2} a^2(1 + \epsilon)^2} = \frac{2}{(1 + \epsilon)^2}. \end{aligned}$$

## Appendix B

**Lemma 2.** For all  $0 \leq \lambda < 1$ :

$$\frac{1}{1 - \lambda} \leq \left( \log c(\lambda) \right)' \leq \frac{n}{1 - \lambda}. \quad (16)$$

*Proof.* In part (i) of the proof of Lemma 1 we showed that

$$\frac{1}{1 - \lambda} \leq c(\lambda) \leq \left( \frac{1}{1 - \lambda} \right)^n. \quad (17)$$

It is well known that  $c(\lambda)$ , being a moment generating function of a nonnegative random variable, is convex nondecreasing. Clearly, the lower and upper bounds in (17) are convex and increasing functions. We show now that the ‘gap’ between  $c(\lambda)$  and its lower bound is nondecreasing on  $[0, 1)$ . That is

$$\left( c(\lambda) - \frac{1}{1 - \lambda} \right)' \geq 0.$$

As follows

$$\begin{aligned} \left( c(\lambda) - \frac{1}{1 - \lambda} \right)' &= \left( E \left[ e^{\lambda Q(S(\mathbf{X}))} \right] - E \left[ e^{\lambda Q(X_1)} \right] \right)' \\ &= E \left[ Q(S(\mathbf{X})) e^{\lambda Q(S(\mathbf{X}))} \right] - E \left[ Q(X_1) e^{\lambda Q(X_1)} \right] \\ &\geq E \left[ Q(X_1) e^{\lambda Q(X_1)} \right] - E \left[ Q(X_1) e^{\lambda Q(X_1)} \right] = 0, \end{aligned}$$

where we used that  $Q \left( \sum_{j=1}^n X_j \right) \geq Q(X_1)$  because the  $X_j$ ’s are nonnegative and  $Q$  is nondecreasing.

Similarly, the ‘gap’ between  $c(\lambda)$  and its upper bound is nondecreasing on  $[0, 1)$ . That is

$$\left( \left( \frac{1}{1-\lambda} \right)^n - c(\lambda) \right)' \geq 0.$$

As follows

$$\begin{aligned} \left( \left( \frac{1}{1-\lambda} \right)^n - c(\lambda) \right)' &= \left( E \left[ e^{\lambda \sum_{j=1}^n Q(X_j)} \right] - E \left[ e^{\lambda Q(S(\mathbf{X}))} \right] \right)' \\ &= E \left[ \sum_{j=1}^n Q(X_j) e^{\lambda \sum_{j=1}^n Q(X_j)} \right] - E \left[ Q(S(\mathbf{X})) e^{\lambda Q(S(\mathbf{X}))} \right] \\ &\geq E \left[ Q \left( \sum_{i=1}^n X_i \right) e^{\lambda Q \left( \sum_{j=1}^n X_j \right)} \right] - E \left[ Q(S(\mathbf{X})) e^{\lambda Q(S(\mathbf{X}))} \right] = 0, \end{aligned}$$

where we used that  $\sum_{j=1}^n Q(x_j) \geq Q \left( \sum_{j=1}^n x_j \right)$  because  $Q$  is nondecreasing and concave.

Taking logarithms in (17):

$$-\log(1-\lambda) \leq \log c(\lambda) \leq -n \log(1-\lambda).$$

All these three functions are nondecreasing and convex on  $[0, 1)$  and again reasoning as above we obtain that the gaps between  $\log c(\lambda)$  with its lower bound and its upper bound, respectively, are nondecreasing. That is, we obtain the required bounds (16).  $\square$