OPERATIONS RESEARCH
Vol. 60, No. 1, January–February 2012, pp. 78–91
ISSN 0030-364X (print) | ISSN 1526-5463 (online)

http://dx.doi.org/10.1287/opre.1110.1008
© 2012 INFORMS

Sequential Importance Sampling and Resampling for Dynamic Portfolio Credit Risk

Shaojie Deng
Microsoft, Redmond, Washington 98052, alexdeng@microsoft.com

Kay Giesecke
Department of Management Science and Engineering, Stanford University, Stanford, California 94305, giesecke@stanford.edu

Tze Leung Lai
Department of Statistics, Stanford University, Stanford, California 94305, lait@stanford.edu

We provide a sequential Monte Carlo method for estimating rare-event probabilities in dynamic, intensity-based point process models of portfolio credit risk. The method is based on a change of measure and involves a resampling mechanism. We propose resampling weights that lead, under technical conditions, to a logarithmically efficient simulation estimator of the probability of large portfolio losses. A numerical analysis illustrates the features of the method and contrasts it with other rare-event schemes recently developed for portfolio credit risk, including an interacting particle scheme and an importance sampling scheme.

Subject classifications: event timing models; portfolio credit risk; rare-event simulation.
Area of review: Simulation.
History: Received September 2010; revision received March 2011; accepted June 2011.

1. Introduction

Portfolio credit risk is the distribution of financial loss due to defaults in a portfolio of credit-sensitive assets such as loans and corporate bonds. Monte Carlo simulation is widely used to estimate the portfolio loss distribution. It applies to virtually any model of correlated default timing and is often relatively easy to implement. On the other hand, the computational effort required to accurately estimate the probability of large losses could be excessive. This probability is at the center of risk management applications; for example, the estimation of risk measures, such as value at risk.

This paper describes and analyzes a sequential Monte Carlo method for the efficient and unbiased estimation of the probability of large portfolio losses and of other rare events. The method applies to dynamic point process models of correlated default timing. In these widely used models, a default of a portfolio constituent is governed by a stochastic intensity process. The intensity processes are correlated across constituents to reflect the default dependence structure in the portfolio. Analogous to conventional importance sampling (IS) schemes, the method involves a change of probability measures. Default events are sampled sequentially, according to a probability measure that might differ from the reference measure. In addition, the sequentially generated sample paths are resampled using a set of state-dependent weights. The resampling mechanism distinguishes our sequential importance sampling and resampling (SISR) scheme from existing sequential IS schemes. We analyze the asymptotically optimal selection of the resampling weights and provide conditions guaranteeing the logarithmic efficiency of the estimator of the probability of large portfolio losses generated by the SISR method.

The SISR method is related to the interacting particle system (IPS) scheme analyzed by Del Moral and Garnier (2005), which was adapted to estimating the probability of large portfolio losses by Carmona and Crépey (2010), Carmona et al. (2009), and Giesecke et al. (2010). The IPS scheme proceeds sequentially, under the reference measure, and involves a resampling ("selection") mechanism. The resampling mechanism corresponds to a change of measure on path space. Its effectiveness, and hence the performance of the estimators generated by the IPS scheme, depends heavily on a judicious choice of a parameter specifying the resampling weights. The asymptotically optimal choice of this parameter has not yet been addressed in the literature. Therefore, ad-hoc approaches have been used to determine this parameter. Our SISR scheme eliminates the need to select a parameter and generates rare-event estimators that are provably efficient. Moreover, the SISR scheme allows one to use a sampling measure that differs from the reference measure. This can make sampling more convenient and might also lead to additional variance reduction.

Numerical experiments illustrate the performance of the SISR algorithm for a self-exciting model of portfolio credit

risk, under which the constituent default intensities follow correlated jump-diffusion processes. For a given computational budget, the SISR algorithm provides more accurate estimates of the probability of large losses than an IPS scheme. Moreover, the SISR scheme yields meaningful estimates of very small probabilities. If the portfolio is relatively heterogeneous, the SISR scheme can also outperform the logarithmically efficient IS scheme of Giesecke and Shkolnik (2010).

While SISR algorithms have long been used to sample from complex, high-dimensional distributions occurring in nonlinear filtering and other areas, their usefulness for rare-event simulation has only recently been recognized by Chan and Lai (2011). They provide a CLT for a generic SISR estimator and a consistent estimator of the asymptotic variance. Specializing to a classical large deviations setting, Chan and Lai (2011) then show how to select the resampling weights in order to obtain logarithmically efficient estimators of certain rare-event probabilities for random walks with finite moment generating functions. In this paper, we propose resampling weights in a setting of multivariate, intensity-based point processes that model the arrival of correlated events. Building on the arguments of Chan and Lai (2011), we then develop conditions guaranteeing that these weights generate a logarithmically efficient estimator of large loss probabilities.

The rest of this paper is organized as follows. Section 2 formulates the portfolio credit risk problem. Section 3 describes a basic SISR algorithm, and §4 analyzes the asymptotically optimal selection of the resampling weights. Section 5 describes an extended SISR algorithm with occasional resampling. Section 6 provides numerical results. Section 7 concludes. There is a technical appendix.

2. Dynamic Portfolio Credit Risk

Consider a portfolio of n firms that are subject to default risk. The random default times of these firms are modeled by almost surely distinct stopping times $\tau^i > 0$, which are defined on a complete probability space $(\Omega, \mathcal{F}, \mathbb{P})$ with right-continuous and complete information filtration. In risk management applications, $\mathbb{P}$ is the statistical probability; while in derivatives pricing applications, $\mathbb{P}$ is a risk-neutral pricing measure. Associated to the τ^i are indicator processes N^i given by $N_t^i = I(\tau^i \leqslant t)$, where $I(A)$ is the indicator function of an event $A \in \mathcal{F}$. For each i, there is a strictly positive, integrable, and progressively measurable process λ^i such that the random variables

$$N_t^i - \int_0^t \lambda_s^i (1 - N_s^i)\, ds \tag{1}$$

form a martingale. The process $\lambda^i(1 - N^i)$ represents the conditional default rate, or intensity of firm i. The λ^i are correlated stochastic processes that we take as given. The correlations among the λ^i reflect the default dependence structure of the portfolio constituents. A number of different specifications of $\lambda = (\lambda^1, \ldots, \lambda^n)$ have been developed in the literature. See, for example, di Graziano and Rogers (2009), Duffie and Garleanu (2001), Duffie et al. (2006), Duffie and Singleton (1999), Eckner (2009), Herbertsson and Rootzen (2008), Jarrow et al. (2005), Jarrow and Yu (2001), Mortensen (2006), Papageorgiou and Sircar (2007), and many others.

The credit risk associated with the portfolio is described by the distribution of the portfolio loss $L = l \cdot N$, where $N = (N^1, \ldots, N^n)$ is the vector of default indicators and $l = (l^1, \ldots, l^n)$ is the vector of position losses. Our primary objective is to compute the tail of the loss distribution, which represents the probability of atypically large default losses. These *rare-event probabilities* are at the center of portfolio risk management and other applications. For example, they are used to estimate portfolio risk measures such as value at risk at high confidence levels.

The problem of computing the distributions of N_T and L_T at a fixed horizon $T > 0$ can be cast as a Markov chain problem. Proposition 3.1 of Giesecke et al. (2010) states that there exists a continuous-time Markov chain $M = (M^1, \ldots, M^n) \in \mathbb{S} = \{0, 1\}^n$ such that $\mathbb{P}(N_t = B) = \mathbb{P}(M_t = B)$ for fixed t and all $B \in \mathbb{S}$. The mimicking chain M has no joint transitions in any of its components, and a component M^i starts at 0 and has transition intensity $\pi^i(\cdot, M)$, where

$$\pi^i(t, B) = \mathbb{E}(\lambda_t^i (1 - N_t^i) \mid N_t = B), \quad B \in \mathbb{S}. \tag{2}$$

The expectation (2) can be computed for many standard models of λ; see Giesecke et al. (2010, §5). The existence of M reduces the problem of computing the distribution of N_T to that of computing the distribution of M_T. Similarly, under the assumption that each l^i is drawn, independently of N, from a fixed distribution, it reduces the problem of computing the distribution of L_T to that of computing the distribution of J_T, where $J = l \cdot M$. This reduction is significant; it allows us to analyze a Markov chain model M rather than a potentially complex point process model N.

The jump process J is not itself a Markov chain, because its jump times have an intensity of the form $\sum_{i=1}^n \pi^i(t, M_t)$. Nevertheless, the distribution of J_T can be obtained from the distribution of M_T, which can be computed by solving the forward Kolmogorov equation. However, this approach is often not practical. This is due to the dimensionality of the state space $\mathbb{S} = \{0, 1\}^n$ of M, which tends to be large in practical applications. In pricing problems, the number of portfolio constituent firms n is between 100 and 125. In risk management settings, n can be substantially larger.

We propose to estimate the distribution of J_T using a Monte Carlo simulation of M. The transitions of M can be sampled exactly using the thinning scheme of Lewis and Shedler (1979), leading to unbiased simulation estimators of the distribution. However, a large number of replications might be required to obtain accurate estimates

of the tail of the distribution, which is at the center of our interest. We develop a sequential simulation method designed to dramatically reduce the number of simulation trials required to obtain accurate estimates of tail probabilities. We focus on the case that $l = (1, \ldots, 1)$; §7 comments on the treatment of the general case.

3. Sequential Importance Sampling and Resampling

This section describes a sequential importance sampling and resampling (SISR) scheme for the efficient estimation of tail probabilities.

3.1. Overview

Let $Y_k = (T_k, U_k)$, where T_k is the kth arrival time of the jump process J, and $U_k = M_{T_k}$. Moreover, let $Y_0 = (0, (0, \ldots, 0))$ and $K \leqslant n$. The sequence $Y = (Y_k)_{0 \leqslant k \leqslant K}$ is a discrete-time Markov chain on $\mathbb{R}_+ \times \mathbb{S}$. We denote its $\mathbb{P}$-transition probabilities by $p_k(x, y)$. Suppose the rare event of interest takes the form $\{\mathbf{Y}_K \in \Gamma\}$ for some suitable set Γ. Here and in the sequel, we let $\mathbf{Y}_k = (Y_0, \ldots, Y_k)$ for $0 \leqslant k \leqslant K$. Our algorithm for estimating $\mathbb{P}(\mathbf{Y}_K \in \Gamma)$ combines sequential importance sampling with resampling. The estimator takes the product form

$$\frac{d\mathbb{P}}{d\tilde{\mathbb{P}}} \times \frac{d\tilde{\mathbb{P}}}{d\mathbb{Q}} \times I(\mathbf{Y}_K \in \Gamma), \tag{3}$$

where $\mathbb{Q}$ is an importance measure and $\tilde{\mathbb{P}}$ is a sampling measure, both defined on the sigma-field $\sigma(Y_k\colon k \leqslant K)$. Suppose that under $\tilde{\mathbb{P}}$, Y is a Markov chain with transition density $\tilde{p}_k(x, y)$. Then the first likelihood ratio in (3) is given by

$$\frac{d\mathbb{P}}{d\tilde{\mathbb{P}}} = \prod_{k=1}^{K} \frac{p_k(Y_{k-1}, Y_k)}{\tilde{p}_k(Y_{k-1}, Y_k)}. \tag{4}$$

The transitions of Y are sampled sequentially according to $\tilde{p}_k(x, y)$. The resulting sample paths are resampled using a set of state-dependent weights; this resampling mechanism attempts to have the effect of the likelihood ratio $d\tilde{\mathbb{P}}/d\mathbb{Q}$ in the estimator (3). While the choice $\tilde{\mathbb{P}} = \mathbb{P}$ is not ruled out, the sampling under a measure different from $\mathbb{P}$ might be convenient and might also lead to additional variance reduction. We illustrate this in the context of our numerical case study in §6.

3.2. Basic Algorithm

We describe the basic SISR algorithm with general resampling weight functions $w_k \geqslant 0$, not necessarily those that attempt to mimic $d\tilde{\mathbb{P}}/d\mathbb{Q}$ in (3). The steps are summarized in Algorithm 1. Let $Y_0^{(r)} = Y_0$ for $r = 1, \ldots, m$, where m is an integer. At each stage $k = 1, \ldots, K$, we sample m independent variables $\tilde{Y}_k^{(r)}$ from the density $\tilde{p}_k(Y_{k-1}^{(r)}, \cdot)$ and form the m sample paths, or particles, $\tilde{\mathbf{Y}}_k^{(r)} = (\mathbf{Y}_{k-1}^{(r)}, \tilde{Y}_k^{(r)})$. From these, we draw m particles $\mathbf{Y}_k^{(r)} = (Y_0^{(r)}, \ldots, Y_k^{(r)})$ with replacement using the probabilities defined by the normalized weights

$$\frac{w_k(\tilde{\mathbf{Y}}_k^{(r)})}{\sum_{j=1}^{m} w_k(\tilde{\mathbf{Y}}_k^{(j)})} \tag{5}$$

for $r = 1, \ldots, m$. We stop at stage K without resampling. The basic SISR estimator $\hat{\alpha}$ of the probability $\alpha = \mathbb{P}(\mathbf{Y}_K \in \Gamma)$ is given by

$$\hat{\alpha} = m^{-1} \sum_{r=1}^{m} Z_K(\tilde{\mathbf{Y}}_K^{(r)}) V_{K-1}(\mathbf{Y}_{K-1}^{(r)}) I(\tilde{\mathbf{Y}}_K^{(r)} \in \Gamma), \tag{6}$$

where $V_0 \equiv 1$ and

$$Z_k(\tilde{\mathbf{Y}}_k) = \frac{d\mathbb{P}}{d\tilde{\mathbb{P}}}(\tilde{\mathbf{Y}}_k) = \prod_{i=1}^{k} \frac{p_i(Y_{i-1}, \tilde{Y}_i)}{\tilde{p}_i(Y_{i-1}, \tilde{Y}_i)}, \tag{7}$$

$$V_k(\mathbf{Y}_k) = \prod_{i=1}^{k} \frac{\bar{w}_i}{w_i(\mathbf{Y}_i)}, \qquad \bar{w}_i = m^{-1} \sum_{r=1}^{m} w_i(\tilde{\mathbf{Y}}_i^{(r)}). \tag{8}$$

Algorithm 1 (SISR with bootstrap resampling)

Let m be an integer. Initialize $Y_0^{(1)}, \ldots, Y_0^{(m)}$ to Y_0.
for $k = 1, \ldots, K$ **do**
  for all $r = 1, \ldots, m$ **do**
    Generate $\tilde{Y}_k^{(r)}$ from $\tilde{p}_k(Y_{k-1}^{(r)}, \cdot)$. Set $\tilde{\mathbf{Y}}_k^{(r)} = (\mathbf{Y}_{k-1}^{(r)}, \tilde{Y}_k^{(r)})$. Compute $w_k^{(r)} = w_k(\tilde{\mathbf{Y}}_k^{(r)})$.
  end for
  if $k < K$ **then**
    Resample using weight $w_k(\tilde{\mathbf{Y}}_k^{(r)}) / \sum_{j=1}^{m} w_k(\tilde{\mathbf{Y}}_k^{(j)})$ to draw $\mathbf{Y}_k^{(r)}$, $r = 1, \ldots, m$.
  end if
end for
Return estimator $\hat{\alpha} = m^{-1} \sum_{r=1}^{m} Z_K(\tilde{\mathbf{Y}}_K^{(r)}) V_{K-1}(\mathbf{Y}_{K-1}^{(r)}) \cdot I(\tilde{\mathbf{Y}}_K^{(r)} \in \Gamma)$.

In the estimator (6), the term $Z_K(\tilde{\mathbf{Y}}_K^{(r)})$ is the likelihood ratio between the reference measure $\mathbb{P}$ and the sampling measure $\tilde{\mathbb{P}}$ of a particle $\tilde{\mathbf{Y}}_K^{(r)}$. It takes account of the fact that we sample from $\tilde{\mathbb{P}}$ rather than $\mathbb{P}$. The term $V_{K-1}(\mathbf{Y}_{K-1}^{(r)})$ takes account of the resampling at stages $k = 1, \ldots, K-1$. The resampling weight functions w_i can be chosen so that $V_k(\mathbf{Y}_k)$ approximates the likelihood ratio of $\tilde{\mathbb{P}}_k$ to some target importance measure $\mathbb{Q}_k$. In this case, the denominator of $V_k(\mathbf{Y}_k)$ approximates the likelihood ratio of the target importance measure $\mathbb{Q}_k$ to $\tilde{\mathbb{P}}_k$, and the numerator of $V_k(\mathbf{Y}_k)$ approximates the normalizing constant by the product of the sample means of these weights. Specifically, define the weights $w_k(\mathbf{Y}_k)$ recursively by

$$\prod_{i=1}^{k} w_i(\mathbf{Y}_i) \propto \frac{d\mathbb{Q}_k}{d\tilde{\mathbb{P}}_k}(\mathbf{Y}_k); \tag{9}$$

see (3). When Y_k is a Markov chain under $\mathbb{Q}$ and $\tilde{\mathbb{P}}$, the right side of (9) reduces to $q_k(Y_k \mid Y_{k-1})/\tilde{p}_k(Y_k \mid Y_{k-1})$,

where $q_k(x, y)$ is the transition density of Y under $\mathbb{Q}$ and $\tilde{p}_k(x, y)$ is that under $\tilde{\mathbb{P}}$. The resampling mechanism eliminates the need to sample directly from $\mathbb{Q}_k$. Using $V_k(\mathbf{Y}_k)$ to approximate the likelihood ratio $d\tilde{\mathbb{P}}_k/d\mathbb{Q}_k$, we have

$$Z_k(\mathbf{Y}_k^{(r)})V_k(\mathbf{Y}_k^{(r)}) \approx \frac{d\mathbb{P}_k}{d\tilde{\mathbb{P}}_k} \times \frac{d\tilde{\mathbb{P}}_k}{d\mathbb{Q}_k} = \frac{d\mathbb{P}_k}{d\mathbb{Q}_k}, \tag{10}$$

$$Z_K(\tilde{\mathbf{Y}}_K^{(r)})V_{K-1}(\mathbf{Y}_{K-1}^{(r)}) \approx \frac{d\mathbb{P}_K}{d\tilde{\mathbb{P}}_K} \times \frac{d\tilde{\mathbb{P}}_{K-1}}{d\mathbb{Q}_{K-1}}. \tag{11}$$

Thus, SISR can be used to approximate an importance measure $\mathbb{Q}$ by using sequential IS via the $\tilde{\mathbb{P}}_k$, which corresponds to $d\mathbb{P}_k/d\tilde{\mathbb{P}}_k$ in (10), and resampling, which corresponds to $d\tilde{\mathbb{P}}_k/d\mathbb{Q}_k$. Note that there is no resampling at stage $k = K$.

3.3. Central Limit Theorem and Variance Estimator

It is important to note that due to the resampling at each stage, the SISR particles are no longer independent. Therefore, the CLT for independent trials is no longer valid. Chan and Lai (2011) provide a CLT for the estimators generated by the SISR scheme as well as a consistent estimator of the asymptotic variance. These results apply to any choice of weight functions w_k and facilitate the analysis of the SISR estimator (6).

The "ancestral origin" $a_k^{(r)}$ of a particle $\mathbf{Y}_k^{(r)}$ plays an important role. The first generation of the m particles, before resampling, is $\tilde{Y}_1^{(1)}, \ldots, \tilde{Y}_1^{(m)}$. Set $a_k^{(r)} = j$ if the first coordinate of $\mathbf{Y}_k^{(r)}$ is $Y_1^{(j)}$, i.e., if $Y_1^{(j)}$ is the ancestor of $\mathbf{Y}_k^{(r)}$. Denote by $\#_k^{(r)}$ the number of copies of $\tilde{\mathbf{Y}}_k^{(r)}$ generated from $\tilde{\mathbf{Y}}_k^{(1)}, \ldots, \tilde{\mathbf{Y}}_k^{(m)}$ to form the m particles of the kth generation. Furthermore, denote by $w_k^{(r)}$ the normalized incremental weight (5) for the rth particle of the kth generation. Chan and Lai (2011) show that the SISR estimator $\hat{\alpha}$ of α satisfies

$$\sqrt{m}(\hat{\alpha} - \alpha) \;\Rightarrow\; \mathcal{N}(0, \sigma^2) \quad \text{as } m \to \infty, \tag{12}$$

and that the asymptotic variance σ^2 can be consistently estimated by

$$\hat{\sigma}^2 = m^{-1} \sum_{j=1}^{m} \Bigg\{ \sum_{r:\, a_{K-1}^{(r)} = j,\, \tilde{\mathbf{Y}}_K^{(r)} \in \Gamma} Z_K(\tilde{\mathbf{Y}}_K^{(r)}) V_{K-1}(\mathbf{Y}_{K-1}^{(r)}) - \hat{\alpha} \Bigg[1 + \sum_{k=1}^{K-1} \sum_{r:\, a_{K-1}^{(r)} = j} (\#_k^{(r)} - m w_k^{(r)}) \Bigg] \Bigg\}^2. \tag{13}$$

The derivation of (12) and (13) is based on a martingale representation of $m(\hat{\alpha} - \alpha)$ that is outlined in the appendix for $K = n$. This representation is relevant for the analysis of the asymptotic efficiency of $\hat{\alpha}$ in §4. It also implies that $\hat{\alpha}$ is unbiased. Finally, it indicates that each resampling contributes a term to the variance of $\hat{\alpha}$. Therefore, one should avoid unnecessary resampling, i.e., one should resample only if it is effective. The intuition here is that resampling is ineffective when the histogram of the relative weights (5) is relatively flat, i.e., when the weights are all similar. In that case, resampling generates additional variance without significantly improving the estimator. We will return to this issue in §5 below.

3.4. SISR and IPS

The SISR scheme is related to the interacting particle systems (IPS) scheme developed by Del Moral and Garnier (2005). While designed from a different perspective, the IPS scheme can be cast as an SISR scheme for which the sampling measure $\tilde{\mathbb{P}} = \mathbb{P}$ and the resampling weight w_k takes a particular form. For the purpose of estimating $\mathbb{P}(f(Y_K) \geqslant x)$, one often takes $w_k(\mathbf{Y}_k) = \exp\{\delta(f(Y_k) - f(Y_{k-1}))\}$ for some parameter δ. Although the general IPS scheme of Del Moral and Garnier (2005) can, in principle, allow for other choices, this formulation is widely adopted in the literature on portfolio credit risk; see Carmona and Crépey (2010), Carmona et al. (2009), and Giesecke et al. (2010).

The IPS tilting parameter δ remains fixed through all stages $k = 1, \ldots, K$. To ensure a reasonable performance of the algorithm, δ must take the form $\delta = \delta(x)$. The asymptotically optimal choice of $\delta(x)$ has not yet been addressed in the literature. Therefore, ad-hoc approaches have been used to determine $\delta(x)$. One approach is to choose $\delta(x)$ so as to minimize the relative error of the estimator for a given x. The corresponding optimization is based on a number of auxiliary simulation experiments, in which a set of proposal values of $\delta(x)$ are tested. These additional experiments reduce the computational budget available for the actual simulation. Moreover, the values of $\delta(x)$ so obtained tend to be unstable because they are influenced by the variance of the estimator. For this reason it is difficult to obtain reasonable estimates of extreme tail probabilities.

In the next section, we propose resampling weight functions w_k for the SISR Algorithm 1 that lead to an estimator (6) of $\alpha = \mathbb{P}(\mathbf{Y}_K \in \Gamma)$ that is provably efficient. These weight functions do not require the selection of a tuning parameter.

4. Efficiency Analysis

The SISR algorithm requires the selection of a sampling measure $\tilde{\mathbb{P}}$ and the weight functions w_k. Consider an event $\{\mathbf{Y}_K \in \Gamma\}$ of the form $\{J_T = x\} = \{T_x \leqslant T < T_{x+1}\}$, setting $K = \min(x+1, n)$, which is the event that the portfolio loss at T is equal to $x \in \{0, 1, \ldots, n\}$. We propose resampling weight functions $w_k = w_k(\mathbf{Y}_k)$ of the form

$$w_k = \begin{cases} \dfrac{Z_k}{\pi_{k-1} Z_{k-1}} \left(\dfrac{\pi_k}{\pi_{k-1}} \right)^{x-k} \exp\left\{ \left(\pi_k - \dfrac{x}{T} \right)(T_k - T_{k-1}) + (\pi_{k-1} - \pi_k)T \right\} \\ \quad \text{if } T_k < T, \\ 0 \quad \text{otherwise}, \end{cases} \tag{14}$$

where $\pi_k = \sum_{i=1}^n \pi^i(T_k, U_k)$ is the intensity of J at the kth arrival time T_k for $1 \leqslant k \leqslant x$, and Z_k is the likelihood ratio (7). Below, we develop conditions guaranteeing that, with the choice of the weight function (14) and the sampling measure $\tilde{\mathbb{P}} = \mathbb{P}$, the SISR estimator of $\alpha = \mathbb{P}(J_T = x)$ is logarithmically efficient. Recall that if $\alpha = \alpha_n$ is a rare-event sequence, i.e., if $\alpha_n \to 0$ as $n \to \infty$, then a Monte Carlo estimator $\hat{\alpha}_n$ of α_n is said to be *logarithmically efficient*, if

$$\mathrm{Var}(\hat{\alpha}_n) \leqslant m^{-1}\alpha_n^{2+o(1)} \quad \text{as } n \to \infty; \tag{15}$$

an estimator is said to be strongly efficient if $\mathrm{Var}(\hat{\alpha}_n) = O(m^{-1}\alpha_n^2)$. The SISR estimator (6) of $\mathbb{P}(J_T = x)$ is not expected to be strongly efficient because resampling introduces additional variance and the errors due to using resampling weights to approximate optimal importance sampling, such as in (9), aggregate. On the other hand, the growth of the additional variance can be made subexponential by choosing the resampling weights appropriately, leading to logarithmic efficiency.

4.1. Rare-Event Regime and Result

The intuitive notion that corporate defaults are rare events can be formalized by prescribing each π^i to approach 0. On the other hand, the number of firms n in a credit portfolio is typically large, and letting $n \to \infty$ may still produce a nonnegligible number of defaults within a time horizon T. Because of default correlations among this large number of firms and the variations of π^i over times $t \leqslant T$ and states B, it is difficult to carry out an asymptotic analysis without assuming uniformity in convergence of $\pi^i(t, B)$ to 0. For the purpose of our asymptotic analysis, we therefore make two assumptions.

Assumption 1. *Suppose* $\pi^i(t,B) = (\pi^* + \delta_i(t,B))(1 - N_t^i)$ *for all* $1 \leqslant i \leqslant n$, *where* $\pi^* = \pi^*(n)$ *and for some positive constants* $c_1 < c_2$ *and* $r_n = o(n)$, $\leqslant r_n$ *and* $c_1/n \leqslant \pi^*(n) \leqslant c_2/n$ *for all* $0 \leqslant t \leqslant T$ *and* $B \in \mathbb{S}$.

Assumption 2. *Letting* $x = x_n$ *and* $T = T(n)$, *assume that* $x \to \infty$, $x_n = o(n)$, *and* $T \sim bx$ *for some* $0 < b < 1/c_2$.

The function $\delta_i(t, B)$ describes the impact of a default before t on the intensity of firm i, at the level of the mimicking Markov chain M. Assumption 1 states that, in the limit that $n \to \infty$, the impact of a default is negligible. Moreover, it states that the variation of $\pi^i(t, B)$ over i, t, and B vanishes in the limit, so that asymptotically, all firms default at intensity $\pi^* = \pi^*(n)$. Under Assumption 1, the number J_T of defaults up to time T is $n\pi^*(T + o_{\mathbb{P}}(1))$, because the interarrival times of J are asymptotically i.i.d. exponential with mean $1/\pi^*$. In view of this and Assumption 2, the probability of the event $\{J_T = x\} = \{T_x \leqslant T < T_{x+1}\}$ has a large deviation approximation involving sums of i.i.d. exponential random variables. This leads to the following result.

Theorem 1. *Let* $\hat{\alpha}_n$ *be the estimator of* $\alpha_n = \mathbb{P}(T_{x_n} \leqslant T < T_{x_n+1})$ *generated by the SISR Algorithm* 1 *using* $\tilde{\mathbb{P}} = \mathbb{P}$ *and the resampling weights* (14) *with* $Z_k = 1$ *for all* k. *Under Assumptions* 1 *and* 2, $\hat{\alpha}_n$ *is logarithmically efficient.*

4.2. Proof of Logarithmic Efficiency

The proof of Theorem 3 builds on an argument of Chan and Lai (2011), who establish the logarithmic efficiency of an SISR estimator of the large deviation probability $\mathbb{P}(g(S_n/n) \geqslant b)$. Here, $b > g(\mathbb{E}\xi)$ and $S_n = \sum_{i=1}^n \xi_i$, where $\xi, \xi_1, \xi_2, \ldots$ are i.i.d. d-dimensional random vectors such that $\psi(\theta) = \log \mathbb{E}(e^{\theta'\xi}) < \infty$ for $\|\theta\| < \theta_0$. Let $\Theta = \{\theta\colon \psi(\theta) < \infty\}$ and let Λ be the closure of $\nabla\psi(\Theta)$. For μ belonging to the interior of Λ, let $\theta_\mu = (\nabla\psi)^{-1}(\mu)$ and define the rate function $\phi(\mu) = \theta_\mu'\mu - \psi(\theta_\mu)$. Here the Markov chain Y_n is S_n, and in the special case that $d = 1$ and $g(x) = x$, standard large deviation approximations suggest the following choice of the resampling weights when $\tilde{\mathbb{P}} = \mathbb{P}$:

$$w_k = e^{\theta_b \xi_k - \psi(\theta_b)}. \tag{16}$$

Large deviation approximations are also available in the general case under certain smoothness assumptions on g, yielding the resampling weights

$$w_k(\mathbf{Y}_k) = \exp\big\{\hat{\theta}_k' S_k - k\psi(\hat{\theta}_k) - [\hat{\theta}_{k-1}' S_{k-1} - (k-1)]\psi(\hat{\theta}_{k-1})\big\}, \tag{17}$$

in which $\hat{\theta}_k = \arg\max_{\theta \in \vartheta}\{\theta' S_k/k - \psi(\theta)\}$, where $\vartheta = \{\theta\colon \phi(\mu_\theta) \leqslant \inf_{g(\mu) \geqslant b} \phi(\mu)\}$ contains all "dominating points" in large deviation theory. As explained in the appendix, there are two important elements of the efficiency analysis. The first is the bound

$$\mathbb{P}(g(S_n/n) \geqslant b \mid S_k) \leqslant e^{-nI + o(n)} \prod_{i=1}^k w_i, \tag{18}$$

where $I = \inf\{\phi(\mu)\colon g(\mu) \geqslant b\}$ and w_i is defined in (17). The second are the following bounds on the first two conditional moments of w_i:

$$\mathbb{E}(w_i \mid \mathcal{F}_{i-1}) \leqslant e^{\rho_i},$$
$$\mathbb{E}(w_i^2 \mid \mathcal{F}_{i-1}) \leqslant C \quad \text{for all } 1 \leqslant i \leqslant n, \tag{19}$$

where ρ_i and C are nonrandom constants such that $\lim_{i\to\infty} \rho_i = 0$, and where the σ-field $\mathcal{F}_{i-1}$ is defined in the appendix.

We modify these ideas to prove Theorem 1. In particular, first note that $\mathbb{P}(g(S_n/n) \leqslant b)$ with $b < g(\mu)$ can be handled similarly. Consider the special case in which ξ is exponential with intensity λ. Here $\psi(\theta) = \log(\lambda/(\lambda - \theta))$ and $(d/d\theta)\psi(\theta) = (\lambda - \theta)^{-1}$ for $\theta < \lambda$, so $\theta_\mu = \lambda - \mu^{-1}$.

Therefore, the resampling weights (16) that yield a logarithmically efficient SISR algorithm (with $\tilde{\mathbb{P}} = \mathbb{P}$) to evaluate $\mathbb{P}(S_n \leqslant bn)$, with $b < \mu$, reduce to

$$w_k = \exp\{\theta_b \xi_k - \psi(\theta_b)\} = (\lambda/b^{-1}) e^{(\lambda - b^{-1})\xi_k}. \tag{20}$$

To prove Theorem 3, let $\lambda_n = n\pi^*(n)$. For notational simplicity, we shall drop the subscript n in x_n and denote $T(n)$ by T. Note that because $x = o(n)$, we have uniformly in $k \leqslant x$

$$\pi_{k-1}/(n\pi^*(n)) \to 1, \quad (\pi_k/\pi_{k-1})^{x-k} \sim (x-k)/n,$$
$$(\pi_{k-1} - \pi_k)T = O(x/n), \tag{21}$$

in view of Assumptions 1 and 2. Let $\xi_k = T_k - T_{k-1}$. As noted in §4.1, Assumptions 1 and 2 imply that the ξ_k are approximately i.i.d. exponential with intensity λ_n for $k \leqslant x = o(n)$, and $c_1 \leqslant \lambda_n \leqslant c_2$. Therefore, we have the following analog of (18):

$$\begin{aligned}\mathbb{P}(T_x \leqslant T \mid Y_k) &= \mathbb{P}(x^{-1}T_x \leqslant (1+o(1))b \mid T_k) \\ &= e^{-(I_b + o(1))x} e^{\theta_b T_k - k\psi(\theta_b)} \\ &\propto e^{\theta_b T_k - k\psi(\theta_b)},\end{aligned} \tag{22}$$

where $\theta_b = \lambda_n - b^{-1}$ and $I_b = (b^{-1} - \lambda_n)b - \log(\lambda_n/b^{-1})$. Moreover, $x/T \sim b^{-1}$ by Assumption 2. In view of (21) and that $c_1 \leqslant \lambda_n \leqslant c_2$ and $\lambda_n = n\pi^*(n) = \pi_k(1+o(1))$ for $1 \leqslant k \leqslant x = o(n)$, this argument has proved the logarithmic efficiency of the resampling weights (20), with λ replaced by λ_n, for the SISR estimator of $\mathbb{P}(T_x \leqslant T)$. Although the event of interest is $\{T_x \leqslant T < T_{x+1}\}$ rather than $\{T_x \leqslant T\} = \{J_T \geqslant x\}$, (22) still holds with $\{T_x \leqslant T < T_{x+1}\}$ replacing $\{T_x \leqslant T\}$, which can be shown by making use of saddlepoint approximations for the density functions; see Jensen (1995, §2.2) for the i.i.d. case and Chan and Lai (2003) for the present Markovian setting. In view of (21), using the resampling weights w_i defined by (14) instead of $e^{\theta_b \xi_k - \psi(\theta_b)}$ associated with (22) still yields

$$\mathbb{P}(T_x \leqslant T < T_{x+1} \mid Y_k) \leqslant e^{-(I_b + o(1))x} \prod_{i=1}^{k} w_i \tag{23}$$

and the bounds (19). Thus, we again have the two key elements of the analysis of efficiency as described above, with (23) replacing (18) in the present setting. We are in a position to apply the same argument as that summarized in the appendix to complete the proof of Theorem 3.

4.3. Discussion

The preceding argument basically uses the resampling weights $w_k = \exp((\lambda_n - b^{-1})\xi_k)$ when the ξ_k are i.i.d. exponential and extends the argument to the present setting by exploiting Assumptions 1 and 2. For the case $\tilde{\mathbb{P}} = \mathbb{P}$, it is natural to ask why we do not stick to these resampling weights but use (14) instead, while using (21) to show that those more complicated weights are asymptotically equivalent to $w_k = \exp((\lambda_n - b^{-1})\xi_k)$ under Assumptions 1 and 2. The reason is that we want the weights (14) to be generally applicable, far beyond what Assumptions 1 and 2 allow. We start by replacing λ_n in $\exp((\lambda_n - b^{-1})\xi_k)$ by some representative intensity over the interval (T_{k-1}, T_{k+1}) within which T_k assumes its value. Two obvious choices of this representative intensity are π_{k-1} and π_k, which are asymptotically equivalent under Assumptions 1 and 2 but can differ substantially in general. The additional factor $\pi_{k-1}^{-1}(\pi_k/\pi_{k-1})^{x-k}\exp\{(\pi_{k-1} - \pi_k)T\}$ in (14) is used to adjust for this difference. We have arrived at the adjustment by the following heuristics.

The key idea underlying the preceding argument to establish logarithmic efficiency under Assumptions 1 and 2 is that the resampling weights w_k can be chosen such that they have good variance properties, as specified by (19), and also satisfy

$$\mathbb{P}(J_T = x \mid Y_k) \leqslant \alpha_n^{1+o(1)} \prod_{i=1}^{k} w_i, \tag{24}$$

for which (22) is the analog in the case of $\mathbb{P}(T_x \leqslant T \mid Y_k)$. The preceding treatment of the ξ_k as approximately i.i.d. exponential suggests that a simple approximation to the seemingly intractable conditional distribution of J_T given Y_k, which we shall denote by $J_T \mid Y_k$, is Poisson with mean $\lambda(T - T_k)$. In particular, setting $\lambda = \pi_k$ for $J_T \mid Y_k$ and $\lambda = \pi_{k-1}$ for $J_T \mid Y_{k-1}$ suggests that a suitable approximation of the ratio

$$\mathbb{P}(\text{Poisson}(\pi_k(T - T_k)) = x)/\mathbb{P}(\text{Poisson}(\pi_{k-1}(T - T_{k-1})) = x)$$

might be a good choice for w_k, in view of (24). The reason why we use an approximation instead of the ratio itself is to ensure that the weights w_k thus constructed have good variance properties. Starting with the case $\pi_k = \pi_{k-1} = \lambda_n$ considered in the proof of Theorem 3 suggests the form of the approximation, which we modify as (14).

We close this section by emphasizing that the scope of the resampling weights (14) we have proposed and analyzed in this section is much broader than that of Assumptions 1 and 2. The numerical experiments in §6 illustrate that the weights (14) generate an SISR estimator that can perform well also in more complex intensity model formulations that violate Assumptions 1 and 2.

5. Occasional Resampling

Section 4 has shown how the resampling weights $w_k = w_k(\tilde{\mathbf{Y}}_k)$ can be chosen to obtain an efficient SISR estimator. For a given sequence of resampling weights w_k, the performance of the SISR scheme can be improved by resampling only at certain prespecified or data-dependent times.

Algorithm 2 (SISR with occasional bootstrap resampling)

Let m be an integer. Initialize $Y_0^{(1)}, \ldots, Y_0^{(m)}$ to Y_0.
  Set the CV threshold κ. Set $T_R = 0$ and $V^{(r)} = 1$ for $r = 1, \ldots, m$.
for $k = 1, \ldots, K$ **do**
  for all $r = 1, \ldots, m$ **do**
    Generate $\tilde{Y}_k^{(r)}$ from $\tilde{p}_k(Y_{k-1}^{(r)}, \cdot)$. Set $\tilde{\mathbf{Y}}_k^{(r)} = (\mathbf{Y}_{k-1}^{(r)}, \tilde{Y}_k^{(r)})$. Compute $w_k^{(r)} = w_k(\tilde{\mathbf{Y}}_k^{(r)})$.
    The resampling weights are defined as $W_k^{(r)} = \prod_{s=T_R+1}^{k} w_s^{(r)}$.
  end for
  Calculate the CV of the resampling weights $W_k^{(r)}$.
  if $k < K$ and CV$> \kappa$ **then**
    Update: $T_R \leftarrow k$. $V^{(r)} \leftarrow V^{(r)} \times \bar{W}_k / W_k^{(r)}$ where $\bar{W}_k = \sum_{r=1}^{m} W_k^{(r)}/m$.
    Resample using weights $W_k^{(r)} / \sum_{j=1}^{m} W_k^{(j)}$ to draw $(\mathbf{Y}_k^{(r)}, V^{(r)})$, $r = 1, \ldots, m$.
  else
    Let $\mathbf{Y}_k^{(r)} = \tilde{\mathbf{Y}}_k^{(r)}$.
  end if
end for
Return estimator $\hat{\alpha} = m^{-1} \sum_{r=1}^{m} Z_K(\tilde{\mathbf{Y}}_K^{(r)}) V^{(r)} I(\tilde{\mathbf{Y}}_K^{(r)} \in \Gamma)$, where Z_K is defined in (7).

As pointed out in the discussion of (23) and (24), the product $\prod_{i=1}^{k} w_i$ plays a basic role in the performance of the SISR estimator of a rare-event probability when resampling is carried out at every stage $k = 1, 2, \ldots, K$. This suggests that if one performs resampling only at times $R_1 < R_2 < \cdots$, one should keep track of the product

$$W_k = \prod_{s=R_j+1}^{k} w_s \quad \text{for } R_j < k \leqslant R_{j+1}$$

so that an SISR scheme with occasional resampling has properties similar to those of the basic Algorithm 1. Kong et al. (1994) have proposed a method of choosing the resampling times, which has been widely used in the IPS literature and the closely related literature on particle filters. The method is based on the coefficient of variation (CV), which is the standard deviation divided by the mean, of the weights $W_k^{(1)}, W_k^{(2)}, \ldots, W_k^{(m)}$ of the m particles. When the CV is large, the effective sample size,[1] given by $m/(1 + CV^2)$, is small. This leads to the recommendation of Kong et al. (1994) to resample whenever the CV of the weights exceeds a specified threshold. Algorithm 2 summarizes the SISR scheme with occasional resampling with a CV threshold κ. In the special case that the CV threshold $\kappa = 0$, we obtain Algorithm 1, which prescribes resampling at each stage. When $\kappa \to \infty$, we never resample and obtain a plain sequential IS scheme. The arguments used to prove (12) and (13) can be modified to show that the estimator $\hat{\alpha}$ of α generated by Algorithm 2 satisfies $\sqrt{m}(\hat{\alpha} - \alpha) \Rightarrow \mathcal{N}(0, \sigma^2)$ as $m \to \infty$, and that the asymptotic variance σ^2 can be consistently estimated by

$$\hat{\sigma}^2 = m^{-1} \sum_{j=1}^{m} \Bigg\{ \sum_{r:\, a^{(r)}_{R_{s(K)}-1} = j,\, \tilde{\mathbf{Y}}_K^{(r)} \in \Gamma} Z_K(\tilde{\mathbf{Y}}_K^{(r)}) V_{R_{s(K)}}(\mathbf{Y}^{(r)}_{R_{s(K)}}) - \hat{\alpha} \Bigg[1 + \sum_{s=1}^{s(K)} \sum_{r:\, a^{(r)}_{R_s-1} = j} (\#^{(r)}_{R_s} - m W^{(r)}_{R_s}) \Bigg] \Bigg\}^2, \quad (25)$$

where $R_1, R_2, \ldots, R_{s(K)}$ are the resampling times and all other notation is defined in §3.3.

6. Numerical Experiments

This section evaluates the performance of the SISR Algorithms 1 and 2. The SISR estimators are contrasted with those generated by direct Monte Carlo simulation, an IPS scheme, and an IS scheme. The simulations are performed on a desktop computer with an Intel Core i5 processor and 4 GB of RAM. The methods are implemented in *R*, which uses only a single CPU core.

6.1. Stochastic Intensity Model

We specialize to the setting of §2 and consider a self-exciting stochastic intensity model $\lambda = (\lambda^1, \ldots, \lambda^n)$ proposed and analyzed using an IPS scheme by Giesecke et al. (2010). Variants and special cases of this model have been studied by Jarrow and Yu (2001), Kusuoka (1999), Yu (2007), and others. Suppose firm i has intensity

$$\lambda^i = X^i + \beta^i \cdot N, \quad (26)$$

where $N = (N^1, \ldots, N^n)$ is the vector of default indicators, $\beta^i = (\beta^{i1}, \ldots, \beta^{in})$ is a vector of non-negative reals with $\beta^{ii} = 0$, and X^i is a risk factor following a Feller diffusion

$$dX_t^i = \kappa_i(\theta_i - X_t^i)dt + \sigma_i \sqrt{X_t^i} dW_t^i, \quad X_0^i > 0. \quad (27)$$

Here κ_i is a parameter controlling the speed of mean-reversion of X_i, θ_i is the level of mean reversion, and σ_i controls the diffusive volatility of X_i. The process $(W^1, \ldots, W^n)$ is a standard Brownian motion. The parameter β^{ij} determines the impact on firm i of firm j's default. The corresponding jump terms $\beta^i \cdot N$ generate correlations between the firms' intensities. Thus, the vectors β^i govern the default dependence structure.

Our goal is to estimate the distribution of portfolio loss $L_T = l \cdot N_T$ under the model (26)–(27), where $T > 0$ is a fixed horizon and l is the vector of position losses. As explained in §2, it suffices to estimate the distribution of $J_T = l \cdot M_T$, which agrees with the distribution of L_T. Here, M is the Markov chain mimicking N. We estimate the probabilities $\mathbb{P}(J_T = x) = \mathbb{P}(L_T = x)$ for integers $x \in [0, n]$, taking $l = (1, \ldots, 1)$. Proposition 3.5 of Giesecke et al. (2010) implies that the transition rate (2) of the Markov chain M takes the form

$$\pi^i(t, B) = (1 - B^i) \left(\beta^i \cdot N_t - \frac{\partial_z \phi^i(t, z)|_{z=0}}{\phi^i(t, 0)} \right), \quad B \in \mathbb{S}, \quad (28)$$

where $\phi^i(t,z)=\mathbb{E}(\exp(-\int_0^t X_s^i\,ds - zX_t^i))$ is exponentially affine in X_0^i under the model (27), with coefficient functions given in Cox et al. (1985). Thus, for $\gamma_i=\sqrt{\kappa_i^2+2\sigma_i^2}$ and $B^i=0$, we have the explicit formula

$$\pi^i(t,B)=\frac{4X_0^i\gamma_i^2e^{\gamma_i t}}{(\gamma_i-\kappa_i+(\gamma_i+\kappa_i)e^{\gamma_i t})^2} - \frac{\theta_i\kappa_i}{\sigma_i^2}\frac{(\kappa_i^2-\gamma_i^2)(e^{\gamma_i t}-1)}{\gamma_i-\kappa_i+(\gamma_i+\kappa_i)e^{\gamma_i t}}+\beta^i\cdot B. \tag{29}$$

Unless stated otherwise, the model parameters are selected randomly. We draw κ_i from $U[0.5,1.5]$ and θ_i from $U[0.001,0.051]$. We take $\sigma_i=\min(\sqrt{2\kappa_i\theta_i},\bar{\sigma}_i)$ where $\bar{\sigma}_i$ is drawn from $U[0,0.2]$. We draw β^{ij} from $U[0,0.01]$, for each $i,j=1,\ldots,n$, $i\neq j$. This selection procedure results in a relatively homogenous portfolio of relatively high credit quality. In practical pricing applications, the parameters would be calibrated from market rates of credit derivatives referenced on the constituent issuers and on the portfolio, as in Eckner (2009) and Mortensen (2006). In risk management applications, the parameters would be estimated from historical default experience, as in Duffie et al. (2006).

6.2. SISR vs. Direct Monte Carlo

We begin by comparing the estimators of the loss probabilities $\mathbb{P}(J_T=x)$ generated by Algorithms 1 and 2 with those generated by a direct Monte Carlo (MC) scheme. In the SISR algorithms, we use the weight function (14). We also take the sampling measure $\tilde{\mathbb{P}}=\mathbb{P}$; another choice is considered in §6.4. For Algorithm 2, we use the CV threshold $\kappa=\sqrt{2}$. Thus, we resample whenever the effective sample size reaches one third of the number of weighted samples m. Our results indicate that the performance of Algorithm 2 is relatively insensitive to the particular value of κ chosen.

The direct MC scheme and the SISR schemes with $\tilde{\mathbb{P}}=\mathbb{P}$ require the simulation of transitions of M under $\mathbb{P}$. We generate a jump time T_k of J using the exact thinning scheme of Lewis and Shedler (1979). This scheme relies on the fact that the intensity $\sum_{i=1}^n\pi^i(t,M_t)$ of J is deterministic between the T_k. At a jump time T_k, we draw the component of M in which the transition took place from the discrete distribution given by $\pi^i(T_k,U_{k-1})/\sum_{j=1}^n\pi^j(T_k,U_{k-1})$ for $i=1,\ldots,n$.

We estimate the loss probabilities $\mathbb{P}(J_1=x)$ for a portfolio of $n=100$ firms. Table 1 reports the estimates for $x=15,\ldots,30$. For a given SISR algorithm, we perform separate simulation experiments for each of the 16 values of x. An experiment uses 1,000 particles. For direct MC, a single simulation experiment yields estimates of $\mathbb{P}(J_1=x)$ for all values of x, unlike the SISR schemes that depend on x. The number of direct MC trials, 80,000, is chosen to roughly match the total computation time required by the 16 SISR experiments. This choice facilitates the comparison of the estimators.

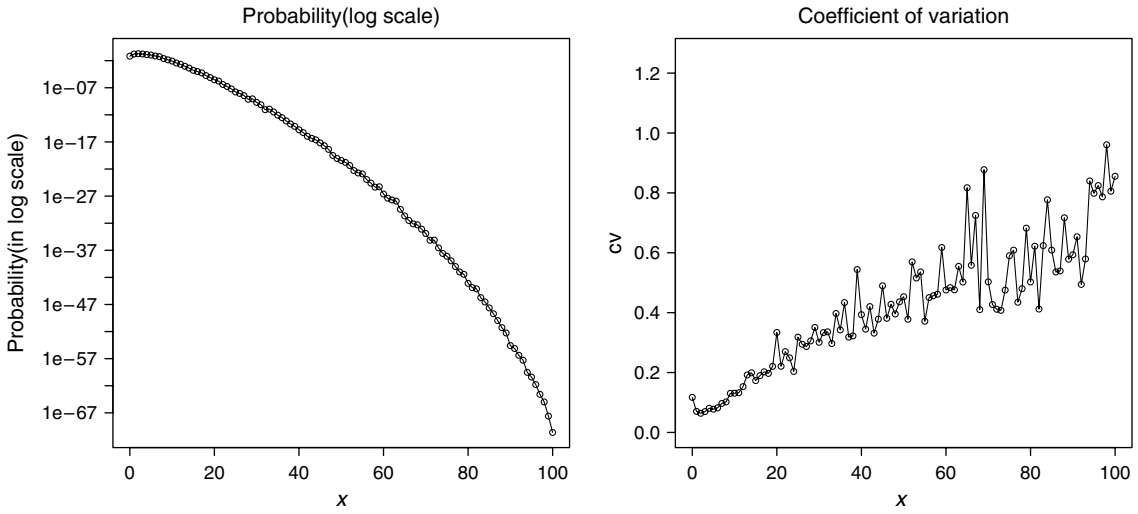
For $16\leqslant x\leqslant 27$, Algorithm 1, which resamples at every stage, is more accurate than Algorithm 2 but requires slightly more CPU time. While the SISR schemes provide

Table 1. Estimating $\mathbb{P}(J_1=x)$ for various values of x for a portfolio of $n=100$ firms using SISR Algorithms 1 and 2 with sampling measure $\tilde{\mathbb{P}}=\mathbb{P}$, resampling weight functions w_k given by (14), CV threshold $\kappa=\sqrt{2}$ (if applicable) and 1,000 particles, and direct Monte Carlo using 80,000 trials.

	SISR Algorithm 1			SISR Algorithm 2				Direct MC	
x	Estimate	CV	CPU	Estimate	CV	#	CPU	Estimate	CV
15	1.805e−04	0.1750	15.90	2.149e−04	0.1726	6	13.96	1.375e−4	0.30
16	8.972e−05	0.1687	16.78	1.029e−04	0.1957	7	14.68	8.750e−5	0.38
17	6.127e−05	0.1468	17.55	4.120e−05	0.1899	7	15.32	1.250e−5	1.00
18	1.470e−05	0.1891	18.72	1.378e−05	0.2000	8	16.52	0	N/A
19	6.797e−06	0.2790	19.55	4.852e−06	0.2555	9	17.12	0	N/A
20	3.276e−06	0.2021	20.47	2.630e−06	0.1896	10	18.26	0	N/A
21	1.633e−06	0.1941	20.98	1.316e−06	0.2411	10	18.77	0	N/A
22	5.596e−07	0.2191	22.34	4.948e−07	0.2101	11	20.09	0	N/A
23	1.932e−07	0.2157	23.26	2.268e−07	0.2311	12	20.75	0	N/A
24	6.194e−08	0.2362	23.65	5.173e−08	0.2752	13	21.56	0	N/A
25	2.371e−08	0.2016	24.60	2.168e−08	0.2523	14	22.79	0	N/A
26	1.140e−08	0.3111	25.60	1.076e−08	0.3370	14	23.53	0	N/A
27	3.415e−09	0.2565	26.80	2.914e−09	0.3344	16	24.30	0	N/A
28	9.140e−10	0.3057	27.35	1.313e−09	0.2651	16	25.12	0	N/A
29	7.588e−10	0.3051	28.17	6.167e−10	0.2873	18	26.38	0	N/A
30	1.603e−10	0.2640	28.58	1.534e−10	0.2567	18	26.92	0	N/A

Notes. The "Estimate" column reports the estimate. The "CV" column reports the coefficient of variation of an SISR estimator, given by the ratio of standard deviation to the mean of the estimator. The standard deviation of an SISR estimator is given by the square root of the asymptotic variance (13) and (25), respectively. The standard deviation of a direct MC estimator is given by the sample standard deviation of the simulation output. The "#" column reports the number of resampling steps. The "CPU" column reports the CPU time in seconds required to perform a simulation.

Figure 1. SISR estimates of $\mathbb{P}(J_1 = x)$ for all values of $x = 0, 1, \ldots, 100$ for a portfolio of $n = 100$ firms, along with the CV for each of the estimates.



Note. The SISR scheme is based on $\tilde{\mathbb{P}} = \mathbb{P}$ and weight function (14) with 1,000 particles.

accurate estimates for all 16 values of x with relative error below 35%, the direct MC simulation fails to provide estimates for $x \geqslant 18$. Figure 1 plots the estimates for all $x = 0, 1, \ldots, 100$ along with their CVs generated by Algorithm 2. We observe that SISR yields meaningful estimates for the extreme tail of the loss distribution, where the probabilities are of order 10^{-67}.

We compare the computation time requirements of the SISR Algorithm 2 with those of the direct MC scheme. It takes 4.1 seconds to generate 1,000 direct MC replications. The average number of defaults per replication is 3.22. Viewing a default as an event in the thinning scheme of Lewis and Shedler (1979), the cost of generating one default per 1,000 replications is 1.27. In comparison with direct MC simulation, the SISR schemes require the storage of particles and an additional resampling step. A regression analysis indicates that the computation time needed by the SISR scheme grows roughly linearly, with an intercept representing fixed costs (memory allocation, etc.) of 0.50. After these fixed costs, the cost of generating 1 default per 1,000 particles is roughly 0.88. Adjusting the costs of the direct MC scheme for these fixed costs, we see that both schemes require roughly the same amount of computation time to generate 1 default per 1,000 replications (particles), which is roughly the cost required by the thinning scheme of Lewis and Shedler (1979) to generate an event.

6.3. SISR vs. IPS

Next we compare the estimators of $\mathbb{P}(J_T = x)$ generated by the SISR Algorithm 2, as configured in §6.2, with those generated by an IPS scheme for the Markov chain $Y_k = (T_k, U_k)$. To facilitate a meaningful comparison, in the IPS scheme we perform a selection (i.e., resample) whenever the effective sample size is less than one third of the number of IPS particles. As explained in §3.4, the IPS scheme requires the selection of a parameter $\delta = \delta(x)$ specifying the weight function, for each value of x. We use a grid search over $[0.2, 3]$ with a step size 0.2 to determine the value of $\delta(x)$ for which the IPS estimator obtained using 1,000 particles has the smallest CV. The variance of the estimator is obtained from (13).

We estimate the loss probabilities $\mathbb{P}(J_1 = x)$ for a portfolio of $n = 100$ firms. Table 2 reports the estimates for $x = 15, \ldots, 30$. As in the case of SISR, we perform 16 separate IPS experiments to obtain these estimates. We choose the number of IPS particles in a given experiment to roughly match the computation time required by the corresponding SISR experiment (Algorithm 2). Here, when calculating the IPS computation time, we ignore the time required to determine the value of $\delta(x)$.

The CV column of Table 2 indicates that the IPS estimates of $\mathbb{P}(J_1 = x)$ have a relative error exceeding 50% for all values $x \geqslant 18$. Figure 2, which plots the estimates and their CVs, indicates that the IPS estimates become unreliable for large x, although they are based on a larger number of particles than the SISR scheme. The values of $\delta(x)$ shown in Table 2 are not monotone in x, suggesting that the CV criterion used to determine the value of $\delta(x)$ in the IPS scheme might lead to unstable choices. It seems difficult to avoid these issues, because working with a fixed δ for any x degrades the performance of the IPS scheme. Figure 3 shows the IPS estimates and their CVs when a fixed value of δ is used, for each of several values of δ. When δ is not large enough, the rare-event set is not hit. On the other hand, the IPS estimator becomes unreliable when δ is too large because of its large variance.

Table 2. Estimating $\mathbb{P}(J_1 = x)$ for various values of x for a portfolio of $n = 100$ firms using an IPS scheme. The "$\delta(x)$" column reports the value of the parameter $\delta(x)$ that minimizes the CV of an estimator.

		IPS scheme				
x	$\delta(x)$	Particles	Estimate	CV	#	CPU
15	2.2	1,500	1.529e–04	0.3367	5	12.87
16	2.0	1,500	4.923e–05	0.3694	6	13.06
17	2.0	1,500	2.329e–05	0.3914	6	13.64
18	2.0	1,500	1.080e–05	0.5571*	7	14.21
19	2.4	1,500	1.811e–06	0.5140*	7	14.72
20	2.2	1,500	9.530e–07	0.6117*	8	15.26
21	2.0	1,500	1.080e–06	0.7471*	8	15.65
22	2.2	1,500	1.031e–06	0.5021*	8	16.67
23	1.8	1,500	1.469e–07	0.6931*	9	15.81
24	3.0	1,800	9.702e–09	0.8374*	10	21.46
25	3.0	1,800	9.683e–10	0.9694*	11	21.80
26	2.2	1,800	1.706e–09	0.8120*	11	21.14
27	2.0	1,800	3.102e–10	0.7712*	11	20.55
28	3.0	1,800	3.291e–10	0.7440*	13	24.86
29	2.2	2,000	5.047e–10	0.9232*	12	26.35
30	2.6	2,000	3.245e–11	1.0052*	13	27.33

Notes. The "Particles" column reports the number of IPS particles used; this number is chosen to roughly match the computation time required by the corresponding SISR Algorithm 2 experiment in Table 1. The "CV" column reports the coefficient of variation of an IPS estimator, given by the ratio of standard deviation to the mean of the estimator. The standard deviation of an IPS estimator is given by the square root of the asymptotic variance (25). The "#" column reports the number of selections performed (resampling steps). The "CPU" column reports the CPU time in seconds required to perform the simulation, excluding the time it takes to determine $\delta(x)$. An "$*$" indicates cases with CV exceeding 50%.

6.4. SISR vs. IS and Choice of the Sampling Measure

The choice of a sampling measure $\tilde{\mathbb{P}}$ different from $\mathbb{P}$ can be beneficial in some situations. Suppose the portfolio contains a high-quality firm whose default has a strong impact on the other firms in the portfolio. When this firm defaults, it causes, with high probability, a cascade of other failures. A failure cascade might be associated with contagion, by which the default by one firm has a direct impact on the health of other firms, channeled through the complex relationships in the economy. Let constituent 1 be this firm, and let $\theta_1 = 0.001$ and $\beta^{i1} = 0.05$ for each $i = 2, \ldots, n$. Let all other $\beta^{ij} = 0$; all other parameters are as in the preceding §§6.2 and 6.3. This configuration of β^{ij} implies that a default of firm 1 increases the intensities of all surviving firms by 0.05.

By choosing a sampling measure $\tilde{\mathbb{P}}$ that increases the default probability of firm 1 relative to the reference measure $\mathbb{P}$, we can improve the efficiency of the SISR scheme. We consider a sampling measure $\tilde{\mathbb{P}}$ such that the reversion parameter $\tilde{\theta}_1 = 1$; all other model parameters under $\tilde{\mathbb{P}}$ are the same as those under $\mathbb{P}$. We denote the SISR scheme by $\text{SISR}_{\tilde{\mathbb{P}}}$ to emphasize this choice of $\tilde{\mathbb{P}}$. Although we need fewer particles to achieve a given accuracy with this choice of $\tilde{\mathbb{P}}$, the computational effort required to evaluate the estimator (6) is larger. This is because we need to evaluate the likelihood ratio (7) of $\mathbb{P}$ to $\tilde{\mathbb{P}}$, which at stage k is given by

$$Z_k = \prod_{i=1}^{k} \exp\left(\int_0^{T_k \wedge T} \log\left(\frac{\pi^i(s-, M_{s-})}{\tilde{\pi}^i(s-, M_{s-})} \right) dM_s^i - \int_0^{T_k \wedge T} (\pi^i(s, M_s) - \tilde{\pi}^i(s, M_s))\, ds \right).$$

Here, π^i is the transition rate of the component M^i of M under $\mathbb{P}$, and $\tilde{\pi}^i$ is the transition rate under $\tilde{\mathbb{P}}$. With our choice of $\tilde{\mathbb{P}}$, Z_k simplifies to

$$Z_k = \left(\frac{\pi^1(\tau^1-, M_{\tau^1-})}{\tilde{\pi}^1(\tau^1-, M_{\tau^1-})} \right)^{M^1_{T_k \wedge T}} \cdot \exp\left(- \int_0^{T_k \wedge T} (\pi^1(s, M_s) - \tilde{\pi}^1(s, M_s)) ds \right), \tag{30}$$

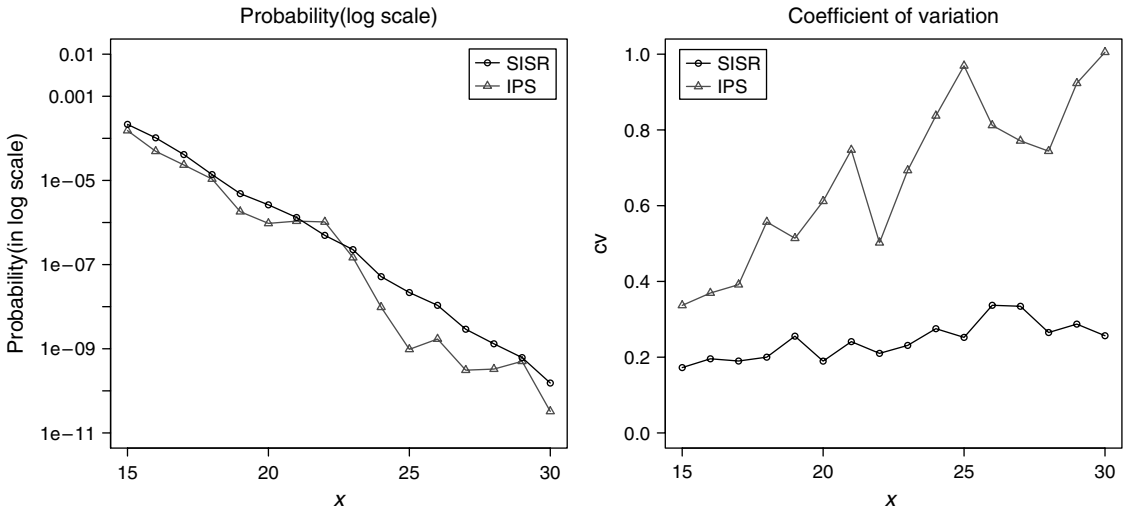
where $\pi^1(t, B)$ is given by (29), and $\tilde{\pi}^1(t, B)$ is given by (29) with $\tilde{\theta}_1$ replacing θ_1. Note that $\pi^1(t, B) = \tilde{\pi}^1(t, B) = 0$ for all $B \in \mathbb{S}$ with $B^1 = 1$. Despite the additional computational cost associated with evaluating (30), the efficiency gains due to using $\tilde{\mathbb{P}}$ rather than $\mathbb{P}$ for sampling can be substantial, as we will illustrate.

We use the weights (14) in conjunction with $\tilde{\mathbb{P}}$ for $\text{SISR}_{\tilde{\mathbb{P}}}$ Algorithm 2. For comparison, we also consider the IS scheme of Giesecke and Shkolnik (2010), which rescales the constituent intensities proportionally so that J is a stopped Poisson process with rate x. While $\text{SISR}_{\tilde{\mathbb{P}}}$ uses 1,000 particles, IS uses 10,000 simulation trials. Table 3 shows that although IS requires more CPU time than $\text{SISR}_{\tilde{\mathbb{P}}}$, its estimates have considerably larger relative errors than those of $\text{SISR}_{\tilde{\mathbb{P}}}$. In fact, all except two of the CV values of IS in Table 3 are larger than 60%, while those of $\text{SISR}_{\tilde{\mathbb{P}}}$ are all below 37%.

We have also considered an alternative ad-hoc IS scheme $\text{IS}_{\tilde{\mathbb{P}}}$ that uses $\tilde{\mathbb{P}}$ as the importance measure rather than the measure identified by Giesecke and Shkolnik (2010). To roughly match the CPU time of $\text{SISR}_{\tilde{\mathbb{P}}}$, $\text{IS}_{\tilde{\mathbb{P}}}$ is based on 4,000 simulation runs. $\text{IS}_{\tilde{\mathbb{P}}}$ performs better than IS only for a few values of x. It fails for $x \geqslant 16$, yielding no simulation run that hits the rare event. The reason SISR can work well with the importance measure $\tilde{\mathbb{P}}$, whereas $\text{IS}_{\tilde{\mathbb{P}}}$ fails, is that resampling makes up for the inadequacy of $\tilde{\mathbb{P}}$ to appropriately transform the rare event under $\mathbb{P}$. This is similar to the situation in §6.2, in which both SISR and direct Monte Carlo sample from $\mathbb{P}$.

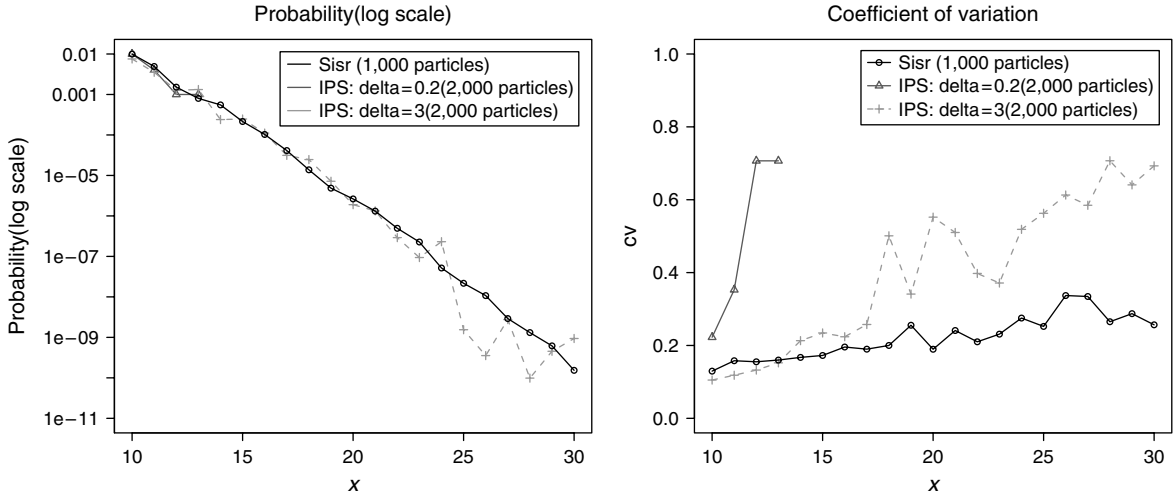
We have also studied the choice $\mathbb{P}$ for the sampling measure in SISR, which still uses (14) for w_k and which we denote by $\text{SISR}_{\mathbb{P}}$. Note that $\text{SISR}_{\tilde{\mathbb{P}}}$ takes more time to run than $\text{SISR}_{\mathbb{P}}$ because of the need to evaluate the likelihood ratio (7). To roughly match the total computation times, we use 2,000 particles for $\text{SISR}_{\mathbb{P}}$. The CV values are

Figure 2. Estimates of $\mathbb{P}(J_1 = x)$ generated by the IPS scheme and SISR Algorithm 2 in a portfolio of $n = 100$ firms, along with the CV of each of the estimates.



Note. The value of the IPS parameter $\delta(x)$ is chosen according to the CV criterion.

Figure 3. Estimates of $\mathbb{P}(J_1 = x)$ generated by the IPS scheme in a portfolio of $n = 100$ firms, along with the CV of each of the estimates.



Note. The value of the parameter δ is fixed for all x in advance rather than chosen according to the CV criterion.

even worse than those of the IS scheme of Giesecke and Shkolnik (2010), reported in Table 3. Note that $\tilde{\mathbb{P}}$ raises the intensity of firm 1 by increasing $\theta_1 = 0.001$ to $\tilde{\theta}_1 = 1$, while keeping all other parameters under $\mathbb{P}$ unchanged. The performance of $\text{SISR}_{\tilde{\mathbb{P}}}$ is robust with respect to the choice of $\tilde{\theta}_1$, yielding similar results when we vary $\tilde{\theta}_1$ from 0.3 to 2.

The IS scheme of Giesecke and Shkolnik (2010) performs much better for relatively homogeneous portfolios, as in the parameter configuration treated in §§6.2 and 6.3. Using an appropriate number of simulation runs to roughly match the CPU time of the SISR Algorithm 2 reported in Table 1, Table 4 indicates the performance of the IS estimate of $\mathbb{P}(J_1 = x)$ under the parameter configuration treated in §§6.2 and 6.3. A comparison of Tables 4 and 1 indicates that because IS is faster than SISR when estimating $\mathbb{P}(J_1 = x)$, which in turn makes it possible to use more trials in each experiment, the IS estimator has a smaller CV than the estimator generated by SISR Algorithm 2.

7. Conclusion

The management of credit risk in portfolios of defaultable assets such as loans and corporate bonds requires accurate estimates of the probability of large losses due to defaults. We provide a sequential Monte Carlo method for the efficient and unbiased estimation of such probabilities in the widely used dynamic point process models of portfolio credit risk. At the center of the method is a resampling

Table 3. Estimating $\mathbb{P}(J_1 = x)$ for various values of x for a portfolio of $n = 100$ firms under a contagion model in which the default by firm 1 has a large impact on the other firms.

	$\text{SISR}_{\tilde{\mathbb{P}}}$				IS			
x	Particles	Estimate	CV	CPU	Runs	Estimate	CV	CPU
15	1,000	1.065e−06	0.1694	21.82	10,000	2.386e−08	0.3394	47.04
16	1,000	5.203e−07	0.2909	22.83	10,000	7.084e−08	0.6227	44.88
17	1,000	9.100e−08	0.1826	23.71	10,000	2.778e−10	0.3139	47.72
18	1,000	5.435e−08	0.1940	24.66	10,000	1.017e−07	0.9885	49.39
19	1,000	1.844e−08	0.2407	26.09	10,000	3.594e−08	0.7408	53.01
20	1,000	5.466e−09	0.2399	28.23	10,000	3.415e−09	0.8280	53.43
21	1,000	8.984e−10	0.2466	30.72	10,000	7.783e−12	0.7008	51.88
22	1,000	2.656e−10	0.3337	31.01	10,000	2.983e−11	0.9990	52.56
23	1,000	6.641e−11	0.2884	33.06	10,000	7.302e−12	0.9989	56.13
24	1,000	1.345e−11	0.2868	33.52	10,000	4.859e−11	0.8631	55.52
25	1,000	4.460e−12	0.3146	34.16	10,000	8.424e−12	0.9363	55.10
26	1,000	6.672e−13	0.3629	37.70	10,000	1.104e−13	0.9944	57.39
27	1,000	1.115e−13	0.3334	38.25	10,000	1.209e−13	0.8446	61.62
28	1,000	3.342e−14	0.3299	40.03	10,000	1.096e−15	0.6062	57.72
29	1,000	3.808e−15	0.3082	42.54	10,000	1.997e−15	0.9972	61.11
30	1,000	1.518e−15	0.3105	43.64	10,000	5.039e−15	0.8476	60.17

Notes. The estimates are obtained using the SISR Algorithm 2 with $\tilde{\mathbb{P}} \neq \mathbb{P}$, w_k given by (14) and $\kappa = \sqrt{2}$, and the IS scheme of Giesecke and Shkolnik (2010). The "Particles" column gives the number of SISR particles used. The "Runs" column reports the number of simulation runs for IS. The "Estimate" columns report the estimates. The "CV" columns report the coefficients of variation of the various simulation estimators. The "CPU" columns report the CPU time in seconds required to perform the simulations.

mechanism that uses state-dependent weights. We analyze the selection of the resampling weights and provide conditions guaranteeing the logarithmic efficiency of the estimator of the probability of large losses. Numerical experiments illustrate the performance of the method and contrast it with alternative schemes. The method has potential applications in other areas, including reliability and insurance.

Table 4. Estimating $\mathbb{P}(J_1 = x)$ for various values of x for a portfolio of $n = 100$ firms under the model treated in §§6.2 and 6.3 using the IS scheme of Giesecke and Shkolnik (2010).

	IS			
x	Runs	Estimate	CV	CPU
15	2,968	2.162e−04	0.05855	14.26
16	3,271	9.733e−05	0.05762	15.88
17	3,210	4.181e−05	0.05924	15.74
18	3,345	1.796e−05	0.06214	16.11
19	3,230	7.999e−06	0.06348	16.12
20	3,418	2.971e−06	0.06169	18.02
21	3,618	1.351e−06	0.06238	19.64
22	3,822	5.342e−07	0.06040	21.04
23	3,697	2.226e−07	0.06208	21.27
24	3,883	8.291e−08	0.06600	21.60
25	4,136	2.859e−08	0.06728	21.59
26	4,100	1.083e−08	0.06327	22.72
27	3,944	4.415e−09	0.06396	24.91
28	4,352	1.532e−09	0.06268	26.63
29	4,317	5.352e−10	0.06623	25.72
30	4,474	1.923e−10	0.06361	26.93

Notes. The "Runs" column reports the number of IS runs; this number is chosen to roughly match the computation time required by the corresponding SISR Algorithm 2 experiment reported in Table 1. The "Estimate" column reports the estimate. The "CV" column reports the coefficient of variation of the IS estimator.

The efficiency analysis in §4 can be extended to the case of stochastic position losses l by exploiting large deviations arguments and saddlepoint approximations for Markov random walks; see Chan and Lai (2003). The reference Markov chain in this more general setting is the Markov random walk $Y_k = (T_k, J_{T_k}, U_k)$. However, the dimensionality of U_k might be an issue, and one needs to use the special structure of the model at hand to design a good SISR scheme, as we have illustrated in §6.4. We leave that extension to future work.

Appendix

This appendix reviews Chan and Lai's (2011) martingale representation of $m(\hat{\alpha} - \alpha)$, where $\alpha = \mathbb{P}(A)$ is the probability of an event A that is estimated by $\hat{\alpha}$ using SISR Algorithm 1. We also summarize the main steps of their proof of the logarithmic efficiency of the SISR estimator when $A = \{g(S_n/n) \geqslant b\}$. Besides providing a key analytic tool for establishing logarithmic efficiency, the martingale representation also yields the central limit theorem (12) and the consistent variance estimate (13) as simple corollaries.

Let $f_0 = \alpha$, $f_k(\mathbf{Y}_k) = Z_k(\mathbf{Y}_k)\mathbb{P}(A \mid \mathbf{Y}_k)$ for $k = 1, \ldots, n$, and $\#_k^{(i)}$ denote the number of copies of $\tilde{\mathbf{Y}}_k^{(i)}$ generated from $\{\tilde{\mathbf{Y}}_k^{(1)}, \ldots, \tilde{\mathbf{Y}}_k^{(m)}\}$ to form the m particles of the kth generation. Moreover, let $a_k^{(i)}$ be the ancestral origin of a particle and $w_k^{(i)}$ the

normalized incremental weight; see §3.3. For $r = 1, \ldots, m$, define

$$\begin{aligned}\epsilon_{2k-1}^{(r)} &= \textstyle\sum_{i:\, a_{k-1}^{(i)}=r}[f_k(\tilde{\mathbf{Y}}_k^{(i)}) - f_{k-1}(\mathbf{Y}_{k-1}^{(i)})]V_{k-1}(\mathbf{Y}_{k-1}^{(i)}) \\ &\qquad\qquad \text{for } 1 \leqslant k \leqslant n, \\ \epsilon_{2k}^{(r)} &= \textstyle\sum_{i:\, a_{k-1}^{(i)}=r}(\#_k^{(i)} - mw_k^{(i)})[f_k(\tilde{\mathbf{Y}}_k^{(i)})V_k(\tilde{\mathbf{Y}}_k^{(i)}) - \alpha] \\ &\qquad\qquad \text{for } 1 \leqslant k \leqslant n-1.\end{aligned} \tag{31}$$

For each fixed r, the sequence $\{\epsilon_k^{(r)}, 1 \leqslant k \leqslant 2n-1\}$ is a martingale difference sequence with respect to the measure $\tilde{\mathbb{P}}$ from which $\tilde{\mathbf{Y}}_k^{(r)}$ and $\mathbf{Y}_k^{(r)}$ are drawn,[2] and the filtration $\{\mathscr{F}_k, 1 \leqslant k \leqslant 2n-1\}$. Here, $\mathscr{F}_{2k-1}$ is the σ-field generated by $\{\tilde{Y}_1^{(r)}, 1 \leqslant r \leqslant m\}$ and $\{(\mathbf{Y}_s^{(r)}, \tilde{\mathbf{Y}}_{s+1}^{(r)}, a_s^{(r)}), 1 \leqslant s < k, 1 \leqslant r \leqslant m\}$, and $\mathscr{F}_{2k}$ is the σ-field generated by $\mathscr{F}_{2k-1}$ and $\{(\mathbf{Y}_k^r, a_k^{(r)}), 1 \leqslant r \leqslant m\}$. Moreover,

$$m(\hat{\alpha} - \alpha) = \sum_{k=1}^{2n-1}\sum_{r=1}^{m}\epsilon_k^{(r)}. \tag{32}$$

An immediate consequence of this martingale representation is the unbiasedness of $\hat{\alpha}$, i.e., $\mathbb{E}_{\tilde{\mathbb{P}}}(\hat{\alpha}) = \alpha$. Another corollary of (32), together with the central limit theorem for martingales, is (12). Moreover, this martingale representation also yields

$$\mathrm{Var}_{\tilde{\mathbb{P}}}(m(\hat{\alpha} - \alpha)) = \sum_{k=1}^{2n-1}\mathrm{Var}_{\tilde{\mathbb{P}}}\left(\sum_{r=1}^{m}\epsilon_k^{(r)}\right). \tag{33}$$

One can regard the $\epsilon_k^{(r)}$ in (31) as encapsulating the random fluctuation due to resampling when k is even, and encapsulating the randomness in sampling from $\tilde{\mathbb{P}}$ when k is odd.

Chan and Lai (2011) exploit (33) to prove the logarithmic efficiency of the SISR estimator of the large deviation probability $\mathbb{P}(g(S_n/n) \geqslant b)$, using $\tilde{\mathbb{P}} = \mathbb{P}$ and resampling weights (17), as noted in the first paragraph of §4.2. They show that the summands of $\sum_{r=1}^{m}\epsilon_k^{(r)}$ are either independent or negatively correlated when conditioned on $\mathscr{F}_{k-1}$, and therefore

$$\mathrm{Var}(m(\hat{\alpha} - \alpha)) \leqslant \sum_{k=1}^{2n-1}\sum_{r=1}^{m}\mathbb{E}(\epsilon_k^{(r)})^2. \tag{34}$$

The rest of the task is to use the form of the resampling weights to bound $\sum_{r=1}^{m}\mathbb{E}(\epsilon_k^{(r)})^2 = m\mathbb{E}(\epsilon_k^{(1)})^2$, because the expectations are the same for all r. Lemma 1 of Chan and Lai (2011) gives the bound (18) and Lemma 2 gives the bound (19). Making use of (31) and (18), they first show that

$$\mathbb{E}((\epsilon_k^{(1)})^2) \leqslant \begin{cases} e^{-2nI+o(n)}\mathbb{E}(\bar{w}_1^2\cdots\bar{w}_{k-1}^2) & \text{if } k \text{ is odd,} \\ e^{-2nI+o(n)}\mathbb{E}(\bar{w}_1^2\cdots\bar{w}_k^2) & \text{if } k \text{ is even.}\end{cases} \tag{35}$$

They then use (19) and an induction argument to show that

$$\mathbb{E}(\bar{w}_1^2\ldots\bar{w}_k^2) \leqslant (1 + Cm^{-1})^k e^{2\rho_1+\cdots+2\rho_k}. \tag{36}$$

Because $\lim_{i\to\infty}\rho_i = 0$, the right-hand side of (36) is bounded, uniformly in $k \leqslant n$, by $\exp\{Cn/m + o(n)\}$. As the number m of particles becomes infinite, applying (35) and (36) to (34) yields $\mathrm{Var}(\hat{\alpha}) \leqslant m^{-1}e^{-2n(I+o(1))}$, proving logarithmic efficiency of the SISR estimator of $\mathbb{P}(g(S_n/n) \geqslant b)$, which they have shown to be equal to $e^{-nI+o(n)}$.

Endnotes

1. Kong et al. (1994) show that the ratio of the variance of an estimator based on weighted samples to the variance of a direct Monte Carlo estimator based on i.i.d. samples is approximately $1 + CV^2$.

2. We assume an augmented probability space in which all random variables involved in resampling are measurable and described by the specified probability measure. Here $\tilde{\mathbb{P}}$ corresponds to that probability measure, and we use $\mathbb{E}_{\tilde{\mathbb{P}}}$ and $\mathrm{Var}_{\tilde{\mathbb{P}}}$ to denote the mean and the variance with respect to $\tilde{\mathbb{P}}$. When $\mathbb{P} = \tilde{\mathbb{P}}$, we simply denote $\mathbb{E}_{\tilde{\mathbb{P}}}$ and $\mathrm{Var}_{\tilde{\mathbb{P}}}$ by $\mathbb{E}$ and Var. This avoids the more complicated notation $\mathbb{P}^*$ used by Chan and Lai (2011) to include the randomization variables in the probability space.

Acknowledgments

The authors are grateful to participants of the INFORMS 2010 Annual Meeting in Austin, Texas, and the Stanford Probability Seminar for comments. They also thank three anonymous referees and Shane Henderson for thoughtful reviews of this paper. Lai's research is supported by National Science Foundation grant DMS-0805879.

References

Carmona, R., S. Crépey. 2010. Particle methods for the estimation of Markovian credit portfolio loss distributions. *Internat. J. Theoret. Appl. Finance* **13**(4) 577–602.

Carmona, R., J.-P. Fouque, D. Vestal. 2009. Interacting particle systems for the computation of rare credit portfolio losses. *Finance and Stochastics* **13**(4) 613–633.

Chan, H. P., T. L. Lai. 2003. Saddlepoint approximations and nonlinear boundary crossing probabilities of Markov random walks. *Ann. Appl. Probability* **31**(2) 395–429.

Chan, H. P., T. L. Lai. 2011. A sequential Monte Carlo approach to computing tail probabilities in stochastic models. *Ann. Appl. Probab.* **21**(6) 2315–2342.

Cox, J., J. Ingersoll, S. Ross. 1985. A theory of the term structure of interest rates. *Econometrica* **53**(2) 385–408.

Del Moral, P., J. Garnier. 2005. Genealogical particle analysis of rare events. *Ann. Appl. Probab.* **15**(4) 2496–2534.

di Graziano, G., C. Rogers. 2009. A dynamic approach to the modelling of correlation credit derivatives using Markov chains. *Internat. J. Theoret. Appl. Finance* **12**(1) 45–62.

Duffie, D., N. Garleanu. 2001. Risk and valuation of collateralized debt obligations. *Financial Anal. J.* **57**(1) 41–59.

Duffie, D., K. Singleton. 1999. Simulating correlated defaults. Working paper, Stanford University, Stanford, CA.

Duffie, D., L. Saita, K. Wang. 2006. Multi-period corporate default prediction with stochastic covariates. *J. Financial Econom.* **83**(3) 635–665.

Eckner, A. 2009. Computational techniques for basic affine models of portfolio credit risk. *J. Comput. Finance* **15** 63–97.

Giesecke, K., A. Shkolnik. 2010. Importance sampling for indicator Markov chains. B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan E. Yücesan, eds. *Proc. 2010 Winter Simulation Conf.*, IEEE Press, Piscataway, NJ, 2742–2750.

Giesecke, K., H. Kakavand, M. Mousavi, H. Takada. 2010. Exact and efficient simulation of correlated defaults. *SIAM J. Financial Math.* **1** 868–896.

Herbertsson, A., H. Rootzen. 2008. Pricing k-th-to-default-swaps under default contagion. *J. Comput. Finance* **12** 49–78.

Jarrow, R. A., F. Yu. 2001. Counterparty risk and the pricing of defaultable securities. *J. Finance* **56**(5) 555–576.

Jarrow, R. A., D. Lando, F. Yu. 2005. Default risk and diversification: Theory and applications. *Math. Finance* **15**(1) 1–26.

Jensen, J. 1995. *Saddlepoint Approximations.* Oxford University Press.

Kong, A., J. S. Liu, W. H. Wong. 1994. Sequential imputations and Bayesian missing data problems. *J. Amer. Statist. Assoc.* **89**(425) 278–288.

Kusuoka, S. 1999. A remark on default risk models. *Adv. Math. Econom.* **1** 69–82.

Lewis, P., G. Shedler. 1979. Simulation of nonhomogeneous Poisson processes by thinning. *Naval Logist. Quart.* **26**(3) 403–413.

Mortensen, A. 2006. Semi-analytical valuation of basket credit derivatives in intensity-based models. *J. Derivatives* **13**(4) 8–26.

Papageorgiou, E., R. Sircar. 2007. Multiscale intensity models and name grouping for valuation of multi-name credit derivatives. *Appl. Math. Finance* **15**(1) 73–105.

Yu, F. 2007. Correlated defaults in intensity based models. *Math. Finance* **17**(2) 155–173.