Rare Events in Stochastic Systems:
Modeling, Simulation Design and Algorithm Analysis

Yixi Shi

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
in the Graduate School of Arts and Sciences

Columbia University

2013

Abstract

Rare Events in Stochastic Systems: Modeling, Simulation Design and Algorithm Analysis

Yixi Shi

This dissertation explores a few topics in the study of *rare events in stochastic systems*, with a particular emphasis on the simulation aspect. This line of research has been receiving a substantial amount of interest in recent years, mainly motivated by scientific and industrial applications in which system performance is frequently measured in terms of events with very small probabilities.

The topics mainly break down into the following themes:

- Algorithm Analysis: Chapters 2, 3, 4 and 5.

- Simulation Design: Chapters 3, 4 and 5.

- Modeling: Chapter 5.

Contents

List of Tables

List of Figures

Acknowledgments

As the ancient Romans put it: Every new beginning comes from other beginning's end, while this dissertation unlatches my journeys ahead, it also marks, sadly, the end of my PhD life here at the IEOR department of Columbia University. I would like to dearly thank everyone that brought strength and joy to me during this otherwise arduous experience.

I am indebted to Professor Jose Blanchet, my advisor, teacher, mentor and friend. Jose took me as his first PhD student in Columbia when he came to IEOR from Harvard Statistics, and during the course of the past four and half years, he has been ungrudgingly sharing with me his sophisticated understanding of the field of rare event simulation. I enjoyed every bit of our discussion, whether it was in Mudd 340, in the pizza place on Amsterdam Ave., or over Skype (and believe it or not, we almost pulled off an academic meeting in Metropolitan Museum of Art). I am in awe of his abysmal knowledge, his astute intuition, his passionate and meticulous attitude towards research, and his humble and affable personality. Without his patience and support, I can hardly imagine getting this far.

I am wholeheartedly thankful to my dissertation committee, Professors Ward Whitt, Karl Sigman, Jingchen Liu and Henry Lam, for taking time to be the readers of my dissertation, and providing useful feedbacks on my work. I am also grateful to Professor Martin Haugh, who had been a reader of Chapter 5 of the dissertation and returned candid comments and constructive suggestions on the insurance network model therein; and Professor Kevin Leder (from Industrial and Systems Engineering Department of University of Minnesota) whom I collaborated with along with Jose in the work of Chapter 2. I also benefited lifelong from the remarkable teachings of Professors Ward Whitt, Donald Goldfarb, Daniel Bienstock, David Yao, Steve Kou, Jose Blanchet, Rama Cont, Mariana Olvera-Cravioto, Cliff Stein, Jingchen Liu (Statistics) Julien Dubedat (Maths), Duong Hong Phong (Maths), Assaf Zeevi (CBS), Michael Johannes (CBS) and Mark Broadie (CBS). I would like to also extend my gratitude to all the staff from IEOR department, who has being doing an awesome job creating a pleasant and homey atmosphere in the department, including the high-frequency supply of free food of course.

Research life in a windowless cubicle (Mudd 313) could have been depressing and monotonous. But thanks to my unique office mates, those days in the office have been my most cherished memories in the past few years. After all, how many PhD offices have their own t-shirts? I would certainly miss all of you, Cecilia Zenteno, Rodrigo Carrasco, Tulia Humphries, Jinbeom Kim, Xingbo Xu, Haowen Zhong, Tony Qin, Arseniy Kukanov, Andrew Ang, and those who have already graduated: Serhat Aybat, Rishi Talreja, Nur Ayvaz, Ohad Perry, Zongjian Liu and Rouba Ibrahim. Indeed, all of my friends and colleagues in Columbia added colors to my PhD life.

I would like to reserve my last gratitudes to my family. In particular, I would like

to dedicate this dissertation to my parents, Bingcheng Shi and Huanya Jiang, who have given me unconditional support and love in every dimension imaginable. And special thanks to my beloved wife Jingjing Song, for keeping me smiling, giving me confidence and putting up with my random schedules along the way.

To my parents Huanya and Bingcheng.

Organize, don't agonize.

Nancy Pelosi

1 Introduction

1.1 Overview

This dissertation explores a few topics in the study of *rare events in stochastic systems*, with a particular emphasis on the simulation aspect. This line of research has been receiving a substantial amount of interest in recent years, mainly motivated by scientific and industrial applications in which system performance is frequently measured in terms of events with very small probabilities.

The topics mainly break down into the following themes:

- Algorithm Analysis: Chapters 2, 3, 4 and 5.

- Simulation Design: Chapters 3, 4 and 5.

- Modeling: Chapter 5.

After this overview we shall briefly review some standard definitions and results that are used throughout the development in this dissertation. In order to have a better overview of the topics covered in the ensuing chapters, I lay out the organizations of the main chapters as follows.

1) Chapter 2 is devoted to the study of splitting methodology in rare event simulation. The study is inspired by the recent work of [31], in which a splitting estimator is proposed and shown to possess asymptotic optimality (see the definition in Subsection 1.2.4) for estimating small probabilities in a light-tailed setting that can be properly approximated using large deviations techniques. Our curiosity is fueled by the fact that in many circumstances the large deviation scaling seems not sufficient to make a precise statement on the performance advantage of splitting over system-specific benchmark algorithms. In addition, it is also helpful to better understand the connection and therefore make guidance implications between splitting and importance sampling strategies. We therefore attempt a sharper analysis on the splitting estimator developed in [31] (a variant of the class of splitting based strategies proposed by [58]), for the particular problem of estimating overflow probabilities in an open Jackson network. Recognizing that crude Monte Carlo is not the correct benchmark to use in this problem setup, we directly compare the complexity of splitting algorithm to that of solving a system of linear equations. While we find out that splitting does outperform the benchmark solution algorithm, it does hold its bells and whistles against competing importance sampling strategies. The analysis serves

as a natural supplement to the series of papers by Paul Dupuis, Hui Wang and their students (e.g., [37], [35], [36], [39] and [31]) on the use of rigorous control theory to construct provably efficient rare event simulation algorithms.

2) The endeavor in Chapter 2 raises a natural question to *the applicability of splitting-based strategies that goes beyond the light-tailed setting.* The construction of importance sampling and splitting algorithms are shown to share a similar root, (see e.g., [37] and [31] and the discussion in Subsection 3.1). In fact, splitting based estimators are in some sense more convenient to come up with. *Do we have a similar story in heavy-tailed systems?* These are the questions we attempt to address in Chapter 3. We try to open the door this line of research by exploring two related splitting-based algorithms designed for a suitable class of heavy-tailed stochastic systems. Both algorithms circumvent the original state space of the underlying stochastic process, and take advantage of some desirable properties of the hazard functions of the increment distributions. More precisely, we embed a splitting procedure in the hazard function space, for which we refer to as the *hazard function splitting (HFS)* strategy. The algorithms are shown to enjoy a uniform setup across the class of input structure of the system. However, on the flip side, although these algorithms are both proved to satisfy the designated asymptotic optimality property, they are not as efficient as some importance sampling based strategies that exploit the distinct large deviation characterizations of heavy-tailed systems.

3) In Chapter 4, we switch gear to study a parametric class of state-dependent importance sampling (SDIS) estimators that is more consistent with how rare events tend to occur in heavy-tailed systems. Quite different from their light-tailed counterparts, in which large deviations occur in a more "cooperative" fashion among the system inputs, the occurrence of rare events for heavy-tailed systems complies with the so

called "principle of large jumps" (see the brief introduction in Subsections 1.2.2 and 1.2.6). In earlier works, for example [22], this mixture based SDIS is shown to be closely tracking the most likely paths of heavy-tailed systems. As a result, with very mild conditions on the parameters, this class of estimators is guaranteed to possess strong efficiency. This desirable "closedness" property enables us to leverage the tool of *cross entropy* to achieve a better performance within the class of strongly efficient mixture-based importance sampling estimators. Closed form recursive formulas to update the mixing probability parameters are provided in this chapter, and a few interesting observations are discussed following the numerical examples illustrated at the end of the chapter.

4) The last chapter, Chapter 5, takes on a holistic approach to study rare events in a specific heavy-tailed financial network system, which is carried out in three major steps, namely a) system modeling, b) asymptotic analysis and c) simulation design and analysis. After carefully specifying the model in step a), the analysis in step b) provides a qualitative but enlightening description on how the system tends to go wrong (in terms of the failure of a specific set of companies). And the goal is to develop efficient Monte Carlo strategies in Step c) to obtain a more quantitative and precise gauge of the degree of systemic risk embedded in this highly inter-correlated risk network system. The measure of the systemic risk comes in the form of the conditional default impact given the failure of a subset of the entire network. The high degree of inter-correlation in the network system is a result of both contractual links and network connectedness. While we are aware of the proliferate amount of research in the area of financial network modeling, the task of finding a unified approach to blend modeling, analysis and risk evaluation remains a very challenging one. Our contribution is the proposition of such an integrated modeling framework

in light of an insurance application.

We carry out our plan in the following way:

Step a) A factor-based discrete time dynamic risk model is built from top down to accommodate typical features in the insurance-reinsurance market, among which the stop-loss contracts written by the insurers, the proportional reinsurance contracts between insurers and reinsurers, and retrocessions among the reinsurance companies, to name a few. Moreover, payment and default settlements at the end of each period are distributed according to the system equilibrium associated with the unique optimal solution to a linear optimization program, properly set up for each period.

Step b) The linear program sheds light on how rare event tends to occur in the system. The large deviations characterization of the system is subsequently shown to be equivalent to solving an integer programming problem, which is identified as a multidimensional Knapsack type of problem.

Step c) Last but not least, aided by the asymptotic description of the system thanks to Step b), we deploy a state-dependent importance sampling strategy, similar in spirit to the one investigated in Chapter 4, to make a more precise quantitative statement on the degree of systemic risk in the network. The associated estimator is shown to be strongly efficient.

1.2 Rare Event Simulation: Preliminaries

1.2.1 Asymptotic Notations

We first list a few notation conventions which will be heavily used in the asymptotic analysis throughout this dissertation.

Definition 1.1 (Big O, Θ, Ω, little o, and aymptotically equivalent $\sim$)**.** *Given two non-negative functions $f(\cdot)$ and $g(\cdot)$, we say*

1) $f(n) = O\big[g(n)\big]$ if there exists c, n_0 such that $f(n) \le cg(n)$ for all $n \ge n_0$;

2) $f(n) = \Omega\big[g(n)\big]$ if there exists c, n_0 such that $f(n) \ge cg(n)$ for all $n \ge n_0$;

3) $f(n) = \Theta\big[g(n)\big]$ if $f(n) = \Omega\big[g(n)\big]$ and $f(n) = O\big[g(n)\big]$;

4) $f(n) = o\big[g(n)\big]$ if for any $\epsilon > 0$, there exists n_1, such that $f(n) \le \epsilon g(n)$ for all $n \ge n_1$;

5) $f \sim g$ if $f(n) = \big(1 + o(1)\big)g(n)$, or equivalently, $f(n)/g(n) \to 1$, as $n \nearrow \infty$.

In Chapter 5 we also use the following probabilistic analogues to the big O, Ω and Θ notations.

Definition 1.2 (Big O, Ω and Θ in Probability)**.** *Let X_n and a_n be a set of random variables and a set of constants, respectively. We denote by*

1. $X_n = O_p(a_n)$ if there exists $M_1(\omega)$, non-negative and finite almost surely, such that

$$\mathbb{P}\left(\lim_{n\to\infty} |X_n/a_n| \le M_1(\omega)\right) = 1.$$

2. $X_n = \Omega_p(a_n)$ if there exists $M_2(\omega)$, non-negative and finite almost surely, such that

$$\mathbb{P}\left(\lim_{n\to\infty} |X_n/a_n| \ge M_2(\omega)\right) = 1.$$

3. $X_n = \Theta_p(a_n)$ *if* $X_n = O_p(a_n)$ *and* $X_n = \Omega_p(a_n)$.

1.2.2 Heavy-tailed Distributions

In this subsection we review some standard definitions and properties of heavy-tailed distributions that are subsequently used in the dissertation.

Conventionally, *heavy-tailed* distributions refer to all those distributions that fail to have moment generating functions. Formally, let $\{X_j\}_{j\geq 1}$ be a series of independent random variables on $(0,\infty)$, with common distribution function $F(x) = \mathbb{P}(X > x)$. And let $\overline{F}(x) = 1 - F(x)$ be its tail distribution function. We have the following definition.

Definition 1.3 (Heavy-tailedness)**.** *A distribution function* F *is said to be heavy-tailed if for all* $\epsilon > 0$,

$$\mathbb{E}\left(e^{\epsilon X}\right) = \int_0^\infty e^{\epsilon X} F(dx) = \infty,$$

or equivalently, $e^{\epsilon X} F(x) \to \infty$, *as* $x \nearrow \infty$.

One useful subclass of heavy-tailed distributions that has been extensively used in the area of queueing theory and insurance risk modeling is the class of *subexponential distributions*. We use the following weakened characterization of subexponentiality given in [41].

Definition 1.4 (Subexponentiality)**.** *A distribution function* F *is subexponential,* $F \in \mathcal{S}$, *if*

$$\frac{\overline{F^{*n}}(x)}{\overline{F}(x)} = \frac{\mathbb{P}(X_1 + \cdots + X_n > x)}{\mathbb{P}(X > x)} \longrightarrow n, \tag{1.1}$$

as $x \nearrow \infty$.

An equally useful characterization of the class $\mathcal{S}$ is given as follows.

Definition 1.5. *$F \in \mathcal{S}$ if for some $n \geq 2$,*

$$\mathbb{P}\left(X_1 + \cdots + X_n > x\right) \sim \mathbb{P}\left(\max_{1 \leq j \leq n} X_j > x\right).$$

The preceding two characterizations of $\mathcal{S}$ sheds light on how large deviations tend to occur in systems with subexponential inputs. In particular, large exceedance of sums is most likely caused by the occurrence of one extremal component. This so-called *catastrophe principle* is very much different in nature from the large deviations principle in a light-tailed systems (see, e.g., [67] and [32]), in which rare events tend to occur because of a more "concerted" effort among all the components. As mentioned in the Introduction, this well-known discrepancy in large deviations characterization has been the key that drives dichotomous developments in the design of rare event simulation algorithms for light-tailed and heavy-tailed systems.

In addition to the previous characterizations of subexponentiality, the following result from [61] allows one to identify subexponentiality from the hazard rate function of the distribution function F, defined as

$$\lambda(x) = d\Lambda(x)/dx = -d \log \overline{F}(x)/dx,$$

where $\Lambda(\cdot)$ is called the hazard function of F.

Lemma 1.1 (Pitman's Condition)**.** *Let $\lambda(x)$ be the hazard rate function of F. Suppose $\lambda(x)$ is eventually decreasing to 0. Then $F \in \mathcal{S}$ if and only if*

$$\int_0^t \lambda(x) e^{x\lambda(t) - \Lambda(x)} dx \longrightarrow 1,$$

as $t \nearrow \infty$.

In Chapters 3 and 4 we shall both work on large classes of the subexponential family, which are specified based on conditions on the hazard rate function $\lambda(x)$ and the hazard function $\Lambda(x)$.

A very important example of the subexponential family is the *regularly varying* distribution.

Definition 1.6 (Slowly Varying Function)**.** *A function f is said to be slowly varying if for all $t > 0$,*

$$\frac{f(tx)}{f(x)} \longrightarrow 1,$$

as $x \nearrow \infty$.

Definition 1.7 (Regularly Varying Distribution)**.** *A non-negative random variable X is called regularly varying of index $-\alpha$, $X \in \mathcal{RV}_{-\alpha}$ if*

$$\overline{F}(x) = L(x)x^{-\alpha},$$

for $\alpha \geq 0$, where $L(\cdot)$ is some slowly varying function.

The following properties of regularly varying distributions are particularly useful in the analysis in Chapter 5.

Lemma 1.2. *Let $X \in \mathcal{RV}_{-\alpha}$.*

1) (Breiman's Theorem [27]). If Y is a non-negative random variable, independent of X that satisfies $\mathbb{E}\left[Y^{\alpha+\epsilon}\right] < \infty$ for some $\epsilon > 0$, then $XY \in \mathcal{RV}_{-\alpha}$. Moreover,

$$\mathbb{P}\left(XY > x\right) \sim \mathbb{E}\left(Y^{\alpha}\right)\mathbb{P}\left(X > x\right).$$

2) (Pareto Conditional Overshoot). We have

$$\mathbb{P}\left(X - bx > by | X > bx\right) \longrightarrow \frac{1}{(1 + y/x)^{\alpha}},$$

as $b \nearrow \infty$.

1.2.3 Importance Sampling and Multilevel Splitting

One powerful tool to achieve variance reduction in estimating rare event probabilities is importance sampling, which involves obtaining samples of the system from an alternative probability measure under which the target event is no longer rare. Specifically, let $\widetilde{\mathbb{P}}(\cdot)$ be this alternative, or "importance sampling" measure. If the *likelihood ratio* or Radon-Nikodym derivative between the original probability measure $\mathbb{P}(\cdot)$ and $\widetilde{\mathbb{P}}(\cdot)$ is well defined on the event of interest, E_n, then the importance sampling estimator for $p_n = \mathbb{P}(E_n)$ is simply set as

$$\widetilde{p}_n \stackrel{\Delta}{=} \frac{d\mathbb{P}}{d\widetilde{\mathbb{P}}}(\omega) I\left(w \in E_n\right),$$

where ω denotes the random outcome or sample path of the underlying system simulated under the probability measure $\widetilde{\mathbb{P}}$. Unbiasedness of the estimator $\widetilde{p}_n$ is guaranteed because

$$\widetilde{\mathbb{E}}\widetilde{p}_n = \int_{E_n} \frac{d\mathbb{P}}{d\widetilde{\mathbb{P}}}(\omega) d\widetilde{\mathbb{P}}\left(\omega\right) = \mathbb{P}\left(E_n\right) = p_n.$$

It turns out that a judiciously picked importance sampling measure oftentimes leads to estimators that enjoy desirable efficiency characteristics, for example strong efficiency as described by Definition 1.9 later. An interesting case from a theoretical standpoint is obtained by setting

$$\widetilde{\mathbb{P}}\left(\cdot\right) = \mathbb{P}_n^*\left(\cdot\right) \stackrel{\Delta}{=} \mathbb{P}\left(\cdot | E_n\right),$$

which yields the corresponding estimator

$$\widetilde{p}_n^* = \frac{d\mathbb{P}}{d\mathbb{P}_n^*}(\omega) I\left(\omega \in E_n\right) = \mathbb{P}\left(E_n\right), \tag{1.2}$$

which is non-random and is therefore called the *zero variance change of measure* (ZVCM) (see for example, [8]). The ZVCM cannot be implemented since the quantity of interest is unfortunately involved. The characterization of the ZVCM as the conditional distribution of the system given the rare event of interest left us with a handy guidance behind the construction of many efficient importance sampling estimators. Obtaining descriptions of $\mathbb{P}_n^*(\cdot)$ as $n \nearrow \infty$ using asymptotic theories acts as a very useful first step in the design of efficient importance sampling estimators. Many existing provably efficient algorithms benefit from tailoring their importance sampling distributions to tracking the conditional behavior of the system according to the descriptions of $\mathbb{P}_n^*(\cdot)$, see for example [23], [20], [36] and [2].

Multilevel splitting (in what follows we shall simply refer to it as *splitting*) is a popular alternative machinery to importance sampling in rare event simulation, particularly in light tailed setting as we have mentioned in the previous introductory section. The prototype of a splitting based algorithm proceeds as follows. The target rare event is decomposed into a series of nested "milestone" events or levels, with the last event coinciding with the target event. Particles representing the underlying stochastic processes are then propagated and split (or replaced by "offspring" particles) whenever such milestone events are hit along the propagation. A weight is endowed to each particle during this process, with the initial particle given a unit weight. Whenever a particle splits, its offspring carries a weight equal to the weight of its parent, divided by the number of offspring particles generated at that split. The final estimate is given by the weighted average of

the particles that make it to the last milestone level. The root of the splitting idea can trace back as early as [53]. Some early developments on splitting are documented in [45]. In the early nineties the conference papers of [58] and [70] introduced the algorithm of *RESTART (REstart Simulation Trials After Reaching Thresholds)*, which blends the idea of splitting into the research of rare event simulation. Since then a few implementation variations of RESTART have been studied (see the conference paper of [43]).

The rationale behind splitting in achieving variance reduction is that, particles or paths that survive longer or manage to enter "later" milestone levels are emphasized and given more importance in terms of the degree of "presence". The design of the splitting algorithm benefits a great deal from the analyses such as in [45], [44] and [43]. We mention in particular [45], which is among the first works that guide the design of splitting based algorithms by a formal notion of efficiency, (work-normalized) asymptotic optimality (see the definition in given in the next subsection) in particular, which turns out to be the common efficiency characteristics for splitting based estimators in general (see Chapters 2 and 3).

The development in Chapter 2 is inspired by the *Splitting Algorithm (SA)* proposed by the recent work of [31]. The techniques used therein in analyzing the splitting algorithm (for example, decomposing the final particles by their *last common ancestors*) are also valuable for a similar class of estimators, and are key techniques used in the analysis in Chapters 2 and 3.

1.2.4 Notions of Efficiency

We shall review concepts of efficiency and complexity in rare event simulation. Let us consider, in a general setting, a sequence of events indexed by a rarity parameter n, $\{E_n, n = 1, 2, ...\}$ such that $p_n = \mathbb{P}(E_n) \to 0$ as $n \nearrow \infty$. The design of efficient rare

event simulation algorithms involves the construction of an unbiased estimator $\hat{p}_n$ such that $\mathbb{E}\hat{p}_n = p_n$. A probability estimate is then formed by averaging a number, say m of i.i.d. replications $\{\hat{p}_n^{(1)}, ..., \hat{p}_n^{(m)}\}$, i.e.,

$$\hat{p}_n(m) = \frac{1}{m}\sum_{j=1}^{m} \hat{p}_n^{(j)}.$$

The goal of algorithm design for rare event probabilities is, generally speaking, to achieve variance reduction over some benchmark algorithms, often naturally taken to be crude Monte Carlo. More precisely, define the *coefficient of variation* of $\hat{p}_n$ as

$$\overline{CV}\left(\hat{p}_n\right) \triangleq \left[\frac{Var\left(\hat{p}_n\right)}{p_n^2}\right]^{1/2}.$$

Given $\epsilon > 0$, we have, by virtue of Chebychev's inequality,

$$\mathbb{P}\left(\frac{|\hat{p}_n - p_n|}{p_n} > \epsilon\right) \leq \frac{\overline{CV}\left(\hat{p}_n\right)^2}{m\epsilon^2}.$$

This implies that the number of replications needed to control the relative error (in a probabilistic sense) is proportional to the squared coefficient of variation:

$$m^* = \lceil \epsilon^{-2}\delta^{-1}\overline{CV}\left(\hat{p}_n\right)^2 \rceil.$$

That is, if $m \geq m^*$, the probability that the relative error $|\hat{p}_n - p_n|/p_n$ exceeds ϵ is at most $1 - \delta$. With this guidance in mind, the notorious inefficiency (in an asymptotic sense) for crude Monte Carlo stems from the fact that the coefficient of variation grows as fast as $1/p_b^{1/2}$. As a result, the number of replications necessary to control the relative error grows exponentially, i.e., $\Omega\left(1/p_n^{1/2}\right)$. In order to control the relative error significantly over that of crude Monte Carlo, the estimator must be constructed with a coefficient of

variation growing subexponentially, or even remaining bounded in the rarity parameter n, which lead to the following two notions of efficiency, respectively.

Definition 1.8 (Asymptotic Optimality)**.** *An estimator $\widehat{p}_n$ is said to be weakly efficient, or asymptotically optimal, logarithmically efficient if $\log \overline{CV}\left(\widehat{p}_n\right) = o\left(1/\log p_n\right)$, as $n \nearrow \infty$. Or equivalently, if for any $\epsilon > 0$ we have*

$$\mathbb{E}\widehat{p}_n^2 = O\left(p_n^{2-\epsilon}\right),$$

as $n \nearrow \infty$.

Definition 1.9 (Strong Efficiency)**.** *An estimator $\widehat{p}_n$ is said to have bounded relative error, or strong efficiency, if $\overline{CV}\left(\widehat{p}_n\right) = O(1)$, as $n \nearrow \infty$. Or equivalently,*

$$\mathbb{E}\widehat{p}_n^2 = O\left(p_n^2\right),$$

as $n \nearrow \infty$.

The discussion up to now leaves aside the issue of the cost of generating a single replication. It is important to recognize that for any splitting based algorithm the computational effort varies drastically with the degree of splitting performed. Splitting, simply put, involves progressively multiplying sample paths of the underlying system. In general, holding the number of replications constant, the faster the propagation rate, the smaller relative error one is able to achieve. However, the increase of the corresponding computation time effectively increases the number of replications. In other words, if the cost of replication grows exponentially, an estimator which is logarithmically efficient is no better than its benchmark crude Monte Carlo counterpart. We shall therefore consider efficiency in a *work-normalized* sense. Let $\mathcal{W}_n$ be the cost per replication of $\widehat{p}_n$. (We measure such cost in terms of the number of elementary function evaluations which we take

to be simple addition, multiplication, comparison and the generation of a single uniform random variable. Depending on the particular setup of the splitting algorithm, we may also need to include operations such as taking logarithms and computing exponentials.) We shall base our analysis on the following definition of the work-normalized version of logarithmic efficiency.

Definition 1.10 (Work-normalized Asymptotic Efficiency)**.** *A splitting estimator $\widehat{p}_n$ is said to be logarithmically efficient if, for each $\epsilon > 0$ we have that*

$$\mathbb{E}\left(\widehat{p}_n^2\right) \mathcal{W}_n = O\left(p_n^{2-\epsilon}\right), \tag{1.3}$$

as $n \nearrow \infty$.

The criterion (1.3) is equivalent to requiring the total number of function evaluations necessary to obtain one single estimate has to grow at least at the same rate as the work-normalized squared coefficient of variation $\overline{CV}\left(\widehat{p}_n\right)^2 \mathcal{W}_n$. One has to keep in mind that, when considering splitting based estimator, this notion of efficiency is by far the most common, although not the strongest.

1.2.5 Constructing Efficient Simulation Estimators in Light-tailed Systems: The Subsolution Approach

A meaningful takeaway from the work of [31] is that in the light-tailed setting (which we shall make precise shortly in the next paragraph), the design of provably efficient splitting-based rare-event simulation algorithms can be put in the same design framework of their importance sampling counterparts. Moreover, splitting estimators constructed in this way are in some sense easier to construct. The aforementioned design framework, systemically developed in a series of papers following [37], uses a control theoretical approach

and the use of *subsolutions of the associated PDE system* underlying the large deviations rate function for the target probability to construct asymptotically optimal (see previous Subsection for definition) importance sampling and splitting-based estimators. In order to better appreciate the design framework just mentioned, in this subsection we shall briefly review this methodology in the setting of multi-dimensional state-dependent random walks.

Formally, let the family of systems $Y^{(\Delta)} = \{Y_t\}_{t\in\{0,\Delta,2\Delta,\dots\}}$, indexed by the scaling parameter $\Delta > 0$, taking values in a subset $\mathcal{D}$ of $\mathbb{R}^d$, and having dynamics defined via

$$Y_{t+\Delta} = Y_t + \Delta V_{t+\Delta}\left(Y_t\right).$$

Here the increment $V_t(y)$'s are assumed to be i.i.d. random variables, dependent upon the current position y. Define the log-moment generating function

$$\psi(\theta, y) = \log \mathbb{E}\left[\exp\left(\theta^T V_t(y)\right)\right]. \tag{1.4}$$

We only consider the light-tailed setting, in the sense that $\psi(\theta, y) < \infty$ for each $y \in \mathcal{D}$, for all $\theta \in \mathbb{R}^d$. It is well-known (see e.g., [67]) that the large deviation behavior of the system as $\Delta \searrow 0$ is governed by the *rate function* of the system, determined by

$$J(w) = \int_0^\tau I\left(w(s), \dot{w}(s)\right) ds,$$

where

$$I\left(w(s), \dot{w}(s)\right) = \max_\theta \Big(\theta\dot{w}(s) - \psi\left(\theta, w(s)\right)\Big),$$

and $0 < \tau < \infty$ is some deterministic time.

Consider the problem of computing the following probability

$$\begin{aligned}\alpha_\Delta(y) &= \mathbb{P}_y^{(\Delta)}\left(T_A < T_B, T_{A\cup B} < \infty\right) \\ &= \mathbb{P}\left(T_A(\Delta) < T_B(\Delta), T_{A\cup B}(\Delta) < \infty | Y_0 = y\right),\end{aligned}$$

where, for any set C, $T_C = \inf\{t \geq 0 : Y_t \in C\}$; moreover, A and B are assumed to be disjoint sets. Furthermore, we assume the following large deviations requirement holds, (see e.g., [32]),

$$-\Delta \log \alpha_\Delta(y) \longrightarrow I_{A,B}(y),$$

where

$$I_{A,B}(y) = \inf_{w(\cdot)\in C} J(w),$$

where the infimum is taken over the set C of absolutely continuous functions satisfying $w(0) = y, w(t) \in A$ for some $t < \infty$ and $w(s) \notin A \cup B$ for any $s < t$. Consider the following natural choice of parametric family of *exponential changes of measure* (see e.g., [9]),

$$\mathbb{P}_{\theta(y)}\left(V_{t+\Delta}(y) \in v + dv\right) = \exp\left(\theta(y)^T v - \psi\left(\theta(y), y\right)\right) \mathbb{P}\left(V_{t+\Delta}(y) \in v + dv\right), \tag{1.5}$$

where $\psi(\cdot,\cdot)$ is the log-moment generating function defined in (1.4). The following result, which is modified from Theorem 8.1 of [38], summarizes the subsolution approach in the particular setting we are considering.

Lemma 1.3 (Subsolution Approach to Construct IS Estimators)**.** *Let $G(\cdot)$ be a smooth*

differentiable function satisfying

$$\begin{aligned} & \psi\left(\theta^*(y), y\right) + \psi\left(-\nabla G_\Delta(y) - \theta^*(y), y\right) \\ = \quad & \min_\theta \Big[\psi\left(\theta(y), y\right) + \psi\left(-\nabla G_\Delta(y) - \theta(y), y\right)\Big] \leq 0,\ y \notin A \cup B. \end{aligned} \tag{1.6}$$

And $G_\Delta(y) \leq 0,\ y \in A$. *Suppose further that* $G_\Delta(y) \geq 2I_{A,B}(y)$. *Then the estimator,* $Z_\Delta(\theta^*)$ *corresponding to sampling the k-th increment of the system using the change of measure given by*

$$\begin{aligned} & \mathbb{P}_{\theta^*\left(Y_{(k-1)\Delta}\right)}\left(V_{k\Delta}\left(Y_{(k-1)\Delta}\right) \in v + dv\right) \\ = \quad & \exp\Big(\theta^*\left(Y_{(k-1)\Delta}\right)^T v - \psi\left(\theta^*\left(Y_{(k-1)\Delta}\right), V_{t+\Delta}\left(Y_{(k-1)\Delta}\right)\right)\Big) \\ & \cdot\ \mathbb{P}\left(V_{t+\Delta}\left(Y_{(k-1)\Delta}\right) \in v + dv\right) \end{aligned} \tag{1.7}$$

has second moment satisfying

$$\liminf_{\Delta \to 0}\left(-\Delta \log \mathbb{E}\left[Z_\Delta(\theta^*)^2\right]\right) \geq G(0).$$

Note that (1.6) can be easily expressed, by virtue of first order optimality conditions, as

$$\psi\left(-\nabla G_\Delta(y)/2, y\right) \leq 0,\ y \notin A \cup B.$$

In other words, $G_\Delta(y)/2$ is a *subsolution* to the associated system $\psi\left(-\nabla U(y), y\right) = 0$, with $U(y) = \infty$ for $y \in B$ and $U(y) = 0$ for $y \in A$, which is called the *Isaacs equation* (see [38]). The essence of this approach is that, finding a subsolution to the Isaacs equation is in some sense equivalent to obtaining a tight upper bound, say $W_\Delta(y)$, for the second moment of parametric family of estimators, $Z_\Delta\left(\theta\right)$. The latter is in turn sufficiently

achieved by requiring that (see Lemma 1, [17])

$$W_\Delta(y) \geq \min_\theta \mathbb{E}\Big[\exp\big(-\theta(y)^T V(y) + \psi\left(\theta(y), y\right)\big) W_\Delta\left(y + V(y)\right)\Big],$$

for $y \notin A \cup B$, and the boundary condition that $W_\Delta(y) \geq 1$ for $y \in B$. We shall illustrate this idea using a heuristic argument, following closely the developments given in Subsections 4.1 and 4.2 of [17].

Large deviations scaling suggests writing $W_\Delta(y) = \exp\left(-\Delta^{-1} G_\Delta(y)\right)$. We shall postulate that $G_\Delta(y) \to G(y)$ as $\Delta \searrow 0$ for some function $G(y)$. Proceeding using this expected limit, we obtain

$$\begin{aligned}
& -\Delta^{-1} G(y) \\
\overset{\geq}{\approx} \quad & \min_\theta \log \mathbb{E}\Big[\exp\left(-\theta(y)^T V(y) + \psi\left(\theta(y), y\right) - \Delta^{-1} G\left(y + \Delta V(y)\right)\right)\Big] \\
= \quad & \min_\theta \log \mathbb{E}\Big[\exp\left(-\theta(y)^T V(y) + \psi\left(\theta(y), y\right) - \Delta^{-1} G\left(y\right) + \nabla G(y)^T V(y)\right) + o(1)\Big] \\
\approx \quad & \min_\theta \left[\log \exp\left(\psi\left(\theta(y), y\right) - \Delta^{-1} G\left(y\right) + \psi\left(-\nabla G(y) - \theta(y)\right), y\right)\right],
\end{aligned}$$

where we have used first order Taylor approximation to reach the second equation. This yields, approximately,

$$\min_\theta \Big[\psi\left(\theta(y), y\right) + \psi\left(-\nabla G(y) - \theta(y), y\right)\Big] \leq 0,$$

precisely (1.6).

We need to emphasize that the *smoothness* condition of the subsolution used in the construction of IS estimator is *sufficient*, which makes the application of this approach more subtle to random walks with constrained behavior on the boundaries, such as *Jackson network* (see Chapter 2). Construction of efficient importance sampling estimators using

subsolutions in such cases can be performed using a mollification technique (see [39] and also [17]) to slightly modify the candidate subsolution function on the boundaries.

Interestingly, efficient splitting based estimators can be constructed based on a very similar subsolution approach. The authors in [31] suggest that if *level placement* is designed according to some *viscosity subsolution* to the associated Isaacs equation, then the resulting splitting estimator is guaranteed to be asymptotically optimal. The difference, also viewed as an advantage of splitting-based strategies over their importance sampling alternatives, lies in the fact that these subsolutions need *not* be smooth. A similar heuristic development as we did following Lemma 1.3 above is carried out in Chapter 2.

1.2.6 State-dependent Importance Sampling for Heavy-tailed Systems

In Subsection 1.2.2 we mentioned that large deviations in heavy-tailed systems occur out of the so-called *principle of large jumps*. The event $\{S_m > b\}$, in particular, belongs to the "single jump domain". The following result from [17] is based on this large deviation characterization in the context of tail probabilities of sums, and has useful implication on the construction of efficient simulation algorithms for heavy-tailed systems. In Chapter 5, in particular, we shall leverage knowledge of this result to develop a similar result on a specific heavy-tailed system with more complex structures.

Lemma 1.4. *Let X_j, $j \leq m$ be i.i.d. random variables having common distribution $F \in \mathcal{S}$, then*

$$\mathbb{P}\left(\max_{1\leq j\leq m} X_j > n | X_1 + \cdots + X_m > n\right) \longrightarrow 1,$$

as $n \nearrow \infty$. *Moreover, for each Borel set* $A \subset \mathbb{R}^m$, *define* $\widehat{\mathbb{P}}_n(\cdot)$ *via*

$$\widehat{\mathbb{P}}_n\left((X_1,\ldots,X_m)\in A\right)=\sum_{j=1}^{m}\mathbb{P}\left((X_1,\ldots,X_m)\in A|X_j>n\right)/m.$$

Then,

$$\sup_A\left|\mathbb{P}\left((X_1,\ldots,X_m)\in A|X_1+\cdots+X_m>n\right)-\widehat{\mathbb{P}}_n\left((X_1,\ldots,X_m)\in A\right)\right|\longrightarrow 0,$$

as $n \nearrow \infty$.

In this dissertation (in Chapters 4 and 5 in particular), we shall consider a parametric class of *state-dependent* importance samplers (SDIS) that are compatible with the way in which rare event occurs in heavy-tailed systems. In simple words, SDIS is designed to sample the increments of the system from a distribution that is dependent on the current status of the system being simulated. The family of estimators we consider is in the form of a mixture. Let us denote by $\underline{p}_j = (p_{j,0}, ..., p_{j,K})$ the vector of mixture probabilities applied to the j-th increment, $j = 1, 2, \ldots$, where $K + 2$ is the number of mixture determined by the heaviness of the tail (the lighter the tail is, the larger K is). Assume that the increments X_j's have densities, which is denoted by $f(\cdot)$. We consider the following general form of the mixture-based sampling density for the k-th increment of the system,

$$\begin{aligned} & h_k\left(x;\underline{p}_k|S_{k-1}=s\right)\\ = \quad & \left(\sum_{j=0}^{K}p_{k,j}I\left(A_j\left(s\right)\right)w_j\left(s,x\right)+\left(1-\sum_{j=0}^{K}p_{k,j}\right)I\left(A_\dagger(s)\right)w_\dagger\left(s,x\right)\right)f(x), \quad (1.8)\end{aligned}$$

where $A_\dagger(s)=\overline{\bigcup_{j=0}^{K}A_j(s)}$, and $w_j\left(s,x\right), w_\dagger\left(s,x\right)>0$ satisfy $\mathbb{E}\left(w_j\left(s,X\right)\right)=\mathbb{E}\left(w_\dagger\left(s,X\right)\right)=1$. Here the event $A_\dagger(s)$ specifies the region in which the increment is *determined to be a*

large shock. One can think of the mixture as a mechanism to control the magnitude of the increments based on evaluations of the current status of the system, and therefore it's a natural choice in order to induce the "principle of big jump" in the sampled paths.

1.2.7 Variance Control via Lyapunov Functions

A useful tool developed for systemically controlling the relative errors of SDIS estimators for heavy-tailed systems is the construction of *Lyaponov inequalities*. This approach has been successfully applied to the design and analysis of the mixture family introduced in the previous subsection for the heavy-tailed setting, see for example [15], [16], [23], and has been shown to be in close relation to the subsolution approach introduced in subsection 1.2.5, see [18]. It turns out that judiciously constructed Lyapunov function, $v(\cdot)$, as we shall introduce momentarily, almost effortless guarantees controlled second moment of the associated SDIS estimators.

Let us again put ourselves in the setting of estimating the probability of the sum of the tails, $\{S_m > b\}$. Denote by $\zeta\left(\widetilde{S}_{k-1}, \widetilde{X}_k\right)$ the local likelihood for the k-th sampling step, $k = 1, \ldots, m$, between the original measure and the measure induced by the state-dependent change of measure, where the notation $\widetilde{S} = \left(\widetilde{S}_k : k \geq 0\right)$ is used to emphasize that the process follows the law induced by the change of measure. For the mixture sampler in (1.8), in particular,

$$\left[\zeta\left(\widetilde{S}_{k-1}, \widetilde{X}_k\right)\right]^{-1} = \sum_{j=0}^{K} p_{k,j} I\left(A_j\left(s\right)\right) w_j\left(s, x\right) + \left(1 - \sum_{j=0}^{K} p_{k,j}\right) I\left(A_\dagger(s)\right) w_\dagger\left(s, x\right).$$

Define $\tau_b = \inf\{k \geq 1 : S_k > b\}$, and $\tau = \tau_b \wedge m$. The associated estimator therefore takes the form

$$R_m(b) = \prod_{k=0}^{\tau-1} \zeta\left(\widetilde{S}_k, \widetilde{X}_{k+1}\right) I\left(\widetilde{S}_\tau > b\right).$$

Note that the applicability of this approach extends beyond this problem setting, the version we illustrate here is simply tailored for the class of problems studied in the ensuing chapters of this dissertation (in particular, Chapter 4).

Lemma 1.5 (Lyapunov Inequality)**.** *Suppose that there exists a non-negative function* $v(\cdot)$*, a constant* $\rho > 0$ *and* $\delta \geq 0$ *such that*

$$v(s)\exp(\delta) \geq \mathbb{E}_s\big[\zeta\left(s, X\right) v(s+X)\big],$$

for $s \leq b$*, and* $v\left(\widetilde{S}_\tau\right) \geq \rho I\left(\widetilde{S}_\tau > b\right)$*. Then we have,*

$$\frac{v(0)}{\rho} \geq \widetilde{E}\left[\exp(-\delta\tau)\prod_{k=0}^{\tau-1}\zeta\left(\widetilde{S}_k, \widetilde{X}_{k+1}\right)^2 I\left(\widetilde{S}_\tau > b\right)\right]. \tag{1.9}$$

Proof. We follow directly the proof given in [15]. Note first that

$$M_k = v\left(S_{\tau\wedge k}\right)\prod_{j=0}^{\tau\wedge k-1}\Big(\exp(-\delta)\zeta\left(S_j, X_{j+1}\right)\Big),$$

defines a non-negative *super-martingale*, adapted to the filtration $\mathcal{F}_k = \sigma\left(S_j, j \leq k\right)$. In particular,

$$\begin{aligned}
& \mathbb{E}\left(M_{k+1}|\mathcal{F}_k\right) I\left(\tau > k\right) \\
= \quad & \prod_{j=0}^{k-1}\Big(\exp(-\delta)\zeta\left(S_j, X_{j+1}\right)\Big)\mathbb{E}\Big[v\left(S_{k+1}\right)\exp(-\delta)\zeta\left(S_k, X_{k+1}\right)|\mathcal{F}_k\Big] I\left(\tau > k\right) \\
\leq \quad & v\left(S_k\right)\prod_{j=0}^{k-1}\Big(\exp(-\delta)\zeta\left(S_j, X_{j+1}\right)\Big) I\left(\tau > k\right) = M_k I\left(\tau > k\right).
\end{aligned}$$

As a result,

$$\begin{aligned}\mathbb{E}\left(M_{k+1}|\mathcal{F}_k\right) &= \mathbb{E}\left(M_{k+1}|\mathcal{F}_k\right) I\left(\tau \leq k\right) + \mathbb{E}\left(M_{k+1}|\mathcal{F}_k\right) I\left(\tau > k\right) \\ &\leq M_k I\left(\tau \leq k\right) + M_k I\left(\tau > k\right) = M_k.\end{aligned}$$

Therefore

$$\begin{aligned}v(0) &= M_0 \geq \mathbb{E}\left(M_m\right) \geq \mathbb{E}\left[v(S_\tau)\prod_{j=0}^{\tau-1}\Big(\exp(-\delta)\zeta\left(S_j, X_{j+1}\right)\Big)\right] \\ &\geq \rho\mathbb{E}\left[I\left(S_\tau > b\right)\exp\left(-\delta\tau\right)\prod_{j=0}^{\tau-1}\zeta\left(S_j, X_{j+1}\right)\right] \\ &\geq \rho\widetilde{E}\left[\exp(-\delta\tau)\prod_{j=0}^{\tau-1}\zeta\left(\widetilde{S}_j, \widetilde{X}_{j+1}\right)^2 I\left(\widetilde{S}_\tau > b\right)\right].\end{aligned}$$

□

Immediately from the previous result we can obtain the following upper bound for the second moment of the estimator $R_m(b)$. In particular,

$$\widetilde{\mathbb{E}}\left[R_m(b)^2\right] = \widetilde{E}\left[\prod_{j=0}^{\tau-1}\zeta\left(\widetilde{S}_j, \widetilde{X}_{j+1}\right)^2 I\left(\widetilde{S}_\tau > b\right)\right] \leq \rho^{-1}\exp(\delta m)v(0). \tag{1.10}$$

The previous equation suggests a strategy for selecting the Lyapunov function in order to enforce strong efficiency (see the definition in Subsection 1.2.4) of the estimator: if the Lyapunov function at step k is chosen to be $O\left[\mathbb{P}\left(S_m > b|S_{k-1} = s\right)^2\right]$, (1.10) provides a "certificate" for the strong efficiency of the SDIS estimator. We shall explore this choice in detail in Chapter 4. In general, the choice of Lyapunov functions is usually guided by large deviations approximations and heuristics available for the square of the target probabilities. For example, [23] successfully utilizes the so-called *fluid heuristics* to

construct Lyapunov functions in the setting of estimating large deviations probabilities for heavy-tailed random walks, $\{S_n\}_{n=1,2,\ldots}$, such as $u(b) = \mathbb{P}(S_n > b)$ as $b \nearrow \infty$, where $b = an^{1/2+\epsilon}$. See also [15], [22], [16] and the survey paper [17] for more discussions.

The journey is the reward.

Chinese Proverb

2

Analysis of a Splitting Estimator for Rare Event Probabilities in Jackson Networks

We consider a standard splitting algorithm for the rare-event simulation of overflow probabilities in any subset of stations in a Jackson network at level n, starting at a fixed initial position. It was shown in [31] that a subsolution to the Isaacs equation guarantees that a subexponential number of function evaluations (in n) suffices to estimate such overflow probabilities within a given relative accuracy (see Definition 1.8). Our analysis here shows that in fact $O\left(n^{2\beta_V+1}\right)$ function evaluations suffice to achieve a given

relative precision, where β_V is the number of bottleneck stations in the subset of stations under consideration in the network. This is the first rigorous analysis that favorably compares splitting against directly computing the overflow probability of interest, which can be evaluated by solving a linear system of equations with $O(n^d)$ variables.

2.1 Introduction

The development of rare-event simulation algorithms for overflow probabilities in stable open Jackson networks has been the subject of a substantial amount of papers in the literature during the last decades (see Section 2 for the specification of an open Jackson network). A couple of early references on the subject are [60] and [4]. Subsequent work which has also been very influential in the development of efficient algorithms for overflows of Jackson networks include [70, 45, 46, 55, 50, 35, 59, 39] and [31]. The survey papers of [52] and [24] provide additional references on this topic.

The two most popular approaches that are applied to the construction of efficient rare-event simulation algorithms are importance sampling and splitting (see [8]). Importance sampling involves simulating the system under consideration (in our case the Jackson network) according to a different set of probabilities in order to induce the occurrence of the rare event. Then, one attaches a weight to each simulation corresponding to the likelihood ratio of the observed outcome relative to the nominal/original distribution. In splitting, on the other hand, there is no attempt to bias the behavior of the system. Instead, the rare event of interest (in our case overflow in a Jackson network) is decomposed into a sequence of nested "milestone" events whose subsequent occurrence is not rare. The rare event occurs when the last of the milestone events occurs. The idea is to keep splitting the particles as they reach subsequent milestones. Of course, each particle is associated with a weight corresponding to the total number of times it has split, so that the overall

estimation (which is the sum of the weights corresponding to the particles that make it to the last milestone) provides an unbiased estimator of the probability of interest.

The most popular performance measure for efficiency analysis of rare-event simulation algorithms for Jackson networks corresponds to that of "asymptotic optimality" or "weak efficiency"(see the definitions in Subsection 1.2.4). In order to both explain the computational complexity implied by this notion and to put in perspective our contributions let us discuss the class of problems we are interested in: Starting from any fixed state, we consider the problem of computing the probability that the total number of customers in any fixed set of stations in the network reaches level n prior to reaching the origin. *In other words, we consider the probability that the sum of the queue lengths in any given subset of stations reaches level n within a busy period.* The number of stations in the whole network is assumed to be d and the number of bottleneck stations (i.e. stations with the maximum traffic intensity in equilibrium) is β.

Weak efficiency guarantees that a subexponential number of replications (as a function of the overflow level, say n) suffices for computing the underlying overflow probability of interest within a given relative accuracy. In contrast, as we shall explain in Section 2.2, overflow probabilities in the setting of Jackson networks can be computed by solving a linear system of equations with $O(n^d)$ unknowns. It is well known that Gaussian elimination then requires $O(n^{3d})$ operations (additions and multiplications) to find the exact solution. Moreover, since in our case the associated linear system has some sparsity properties the linear equations can be solved in at most $O(n^{3d-2})$ operations (see the discussion in Section 2.2). Our analysis for the solution of the associated linear system of equations is not intended to be exhaustive. Our objective is simply to make the point that naive Monte Carlo (which indeed takes an exponential number of replications in n to achieve a given relative accuracy) is not the natural benchmark that one should be using in order to test

the performance of an efficient simulation estimator for overflows in Jackson networks. Rather, a more natural benchmark is the application of a straightforward method for solving the associated system of linear equations. It would be interesting to provide a detailed study of various methods for solving linear systems of equations (such as multigrid procedures) that are suitable for our environment and can even be combined with the ideas behind efficient simulation procedures. This, however, would be the subject of an entire paper and therefore is left as a topic for future research.

Our goal here is to analyze a class of splitting algorithms similar to those introduced in [70] for the evaluation of overflow probabilities at level n. Further analysis was given in [31], where the authors provide necessary and sufficient conditions for the design of the "milestone events" in order to achieve subexponential complexity in n.

Our contribution is to show that if the milestone events are properly placed as suggested by [31], the splitting algorithm requires $O(n^{2\beta+1})$ function evaluations (basically simple operations, see page 5 for a definition and discussion) to achieve a fixed relative error. Since clearly the number of bottleneck stations β is at most d, the complexity of splitting is $O(n^{2d+1})$, which is substantially smaller than that of the direct solution of the associated linear system. Our analysis therefore provides theoretical justification for the superior performance observed when applying splitting algorithms compared to directly solving the associated linear system. The precise statement of our main results is given in Theorem 2.1, at the end of Section 2.5.

We believe that our results shed light into the type of performance that can be expected when applying particle algorithms beyond the setting of Jackson networks. This feature should be emphasized, specially given the fact that a linear time algorithm for computing overflows in Jackson networks has been developed very recently (see [13]). Contrary to particle methods, which are versatile and that can in principle be applied in great

generality, the algorithm in [13] takes advantage of certain properties of Jackson networks which are not shared by all classes of systems.

In addition, our results also provide interesting connections to recent performance analyses studied in the context of state-dependent importance sampling algorithms for a class of Jackson networks. These connections might eventually help guide the users of rare event simulation algorithms to decide when to apply importance sampling or splitting. For instance, consider the overflow at level n of the total population of a tandem network with d stations. The work of [35] proposes an importance sampling estimator based on the subsolution of an associated Isaacs equation. In particular, [35] shows that if exponential tiltings are applied using the gradient of the associated subsolution as the tilting parameter (depending on the current state), the corresponding algorithm is weakly efficient. It turns out that many subsolutions can be constructed by varying certain so-called "mollification parameters". A recent analysis based on Lyapunov inequalities given in [18] shows that a natural selection of mollification parameters guarantees $O(n^{2(d-\beta)+1})$ function evaluations to achieve a given relative error. Our analysis here therefore guarantees that one can achieve a running time of order $O(n^{d+1})$ if one chooses importance sampling when there are more than $d/2$ bottleneck stations in the network and splitting if there are less than $d/2$ bottleneck stations. Although our analysis is still not sharp we believe that our results provide a significant step forward in understanding the connections between splitting and importance sampling.

The rest of the chapter is organized as follows. A brief discussion on complexity and efficiency considerations is given in Section 2.2. Then we discuss the necessary large deviations asymptotics for Jackson networks required for our analysis in Section 2.3. The introduction of the splitting algorithm as well as connections to the theory developed in [31] is given in Section 2.4. Our complexity analysis is finally given in Section 2.5.

2.2 Benchmark to the Splitting Algorithm

In the setting of Jackson networks, it is important to recognize that overflow probabilities can be obtained by solving a system of linear equations. Therefore, a reasonable benchmark procedure for testing "efficiency" in any simulation based algorithm is to compare costs with those associated with directly solving the linear system. Jackson networks are basically multidimensional simple random walks with constrained behavior on the boundaries. In particular, they are Markov chains living on a countable state-space. The overflow probabilities can be conveniently expressed as first passage time probabilities, which in turn can be characterized as the solution to certain linear system of equations thanks to its countable state-space Markov chain structure. We shall quickly review how to obtain such linear system for a generic Markov chain $Q = \{Q_k : k \geq 0\}$ living on a countable state-space $\mathcal{S}$ with transition matrix $\{K(x,y) : x, y \in \mathcal{S}\}$. Let A, B be two disjoint subsets of $\mathcal{S}$, define $\sigma_A \triangleq \inf\{k \geq 0 : X \in A\}$, $\sigma_B \triangleq \inf\{k \geq 0 : X \in B\}$ and put $p(x) = \mathbb{P}_x(\sigma_A \leq \sigma_B)$. A simple conditioning argument on the first transition leads to

$$p(x) = \sum_{y \in \mathcal{S}} K(x,y)\, p(y) \tag{2.1}$$

subject to the boundary conditions

$$p(x) = 1 \text{ for } x \in A, \quad p(x) = 0 \text{ for } x \in B.$$

In fact, $p(\cdot)$ is the minimum non-negative solution to the above system (see [15]).

Now, if Q describes the state of the embedded discrete time Markov chain corresponding to a Jackson network with d stations then $\mathcal{S} = \mathcal{Z}_+^d$. The transition dynamics of a Jackson network are specified as follows (see [64] p. 92). Inter-arrival times and service times are all independent and exponentially distributed random variables. The

arrival rates are given by the vector $\lambda = (\lambda_1, \ldots, \lambda_d)^T$ and service rates are given by $\mu = (\mu_1, \ldots, \mu_d)^T$. (By convention all of the vectors in this dissertation are taken to be column vectors and ^{T} denotes transposition.) A job that leaves station i joins station j with probability $P_{i,j}$ and it leaves the system with probability

$$P_{i,0} \triangleq 1 - \sum_{j=1}^{d} P_{i,j}.$$

The matrix $P = \{P_{i,j} : 1 \leq i, j \leq d\}$ is called the routing matrix. We shall consider open Jackson networks, which satisfy the following conditions:

i) $\forall i$, either $\lambda_i > 0$ or $\lambda_{j_1} P_{j_1 j_2} \ldots P_{j_k i} > 0$ for some $j_1, \ldots, j_k$.

ii) $\forall i$, either $P_{i0} > 0$ or $P_{i j_1} P_{j_1 j_2} \ldots P_{j_k 0} > 0$ for some $j_1, \ldots, j_k$.

iii) The network is stable (i.e. a stationary distribution exists).

These conditions simply require that each station will receive jobs either directly from the outside or routed from other stations, and each job will leave the system eventually. Our main interest lies in the evaluation of $p_n(x)$ assuming that $B = \{0\}$ and $A_n = \{y : v^T y = n\}$ where v is a binary vector which encodes a particular subset of the network (i.e., the i-th position of the vector v is 1 if station i falls in the subset of interest, and 0 otherwise). We shall denote by $V(x) = x^T v$ the mapping recording the total population in the stations corresponding to the vector v. The case in which $v = \mathbf{1} = (1, 1, \ldots, 1)^T$ corresponds to the total population of the system. So, $p_n(x)$, or more precisely $p_n^V(x)$, corresponds to the overflow probability in the subset encoded by v within a busy period starting from x. In this setting, it follows (as we shall review in the next section) that $p_n^V(x) \longrightarrow 0$ exponentially fast in n as $n \nearrow \infty$ and the system of equations (2.1) has $O(n^d)$ unknowns. Gaussian elimination requires $O(n^{3d})$ function evaluations to find the

solution of such system. But since each state of the Markov chain in this case has possible interactions with only a small fraction of the entire state-space, it is therefore possible to permute the states (say in lexicographic order) so that the system is banded (i.e. the associated matrix is sparse in the sense that its non-zero entries fall to a diagonal band.) One can show that the bandwidth is $O(n^{d-1})$, and therefore solving such a banded linear system requires $O(n^d \cdot (n^{d-1})^2) = O(n^{3d-2})$ operations (see, e.g., [5]).

Estimators that possess weak efficiency (in a work-normalized sense) are guaranteed to run at subexponential complexity, see Subsection 1.2.4. When comparing to the above *polynomial* algorithms of solving systems of linear equations, the efficiency analysis of such estimators appears to be insufficient. We will show in later analysis that the multilevel splitting algorithm suggested by Dean and Dupuis [31], applied to estimate the overflow probabilities in Jackson networks, requires fewer function evaluations than directly solving the associated system of linear equations.

2.3 Jackson Networks: Notation and Properties

As we mentioned in the previous section, a Jackson network is encoded by two vectors of arrival and service rates, $\lambda = (\lambda_1, \ldots, \lambda_d)^T$ and $\mu = (\mu_1, \ldots, \mu_d)^T$, together with a routing matrix $P = \{P_{i,j} : 1 \leq i, j \leq d\}$. Without loss of generality, we assume that $\sum_{i=1}^{d} (\lambda_i + \mu_i) = 1$. The network is assumed to be open and stable so conditions i), ii), and iii) described in the previous section are in place.

Given the stability assumption, the system of equations given by

$$\phi_i = \lambda_i + \sum_{j=1}^{d} \phi_j P_{ji}, \quad \forall i = 1, 2, \ldots, d \tag{2.2}$$

admits a unique solution $\phi^T = \lambda^T (I - P)^{-1}$ (see [8]). The traffic intensity at station i in

the system in equilibrium is given by ρ_i which is defined by

$$\rho_i = \frac{\phi_i}{\mu_i} = \frac{[\lambda^T (I - P)^{-1}]_i}{\mu_i}, \tag{2.3}$$

and satisfies $\rho_i \in (0, 1)$ for all $i = 1, 2, \ldots, d$. Define $\rho_* = \max_{1 \le i \le d} \rho_i$ and let β be the cardinality of the set $\{i : \rho_i = \rho_*\}$.

We shall study the queueing network by means of the embedded discrete time Markov chain $Q = \{Q(k) : k \ge 0\}$, where $Q(k) = (Q_1(k), \ldots, Q_d(k))$. For each k, $Q_i(k)$ represents the number of customers in station i immediately after the k-th transition epoch of the system. As mentioned before, the process Q lives in the space $\mathcal{S} = \mathcal{Z}_+^d$.

Let $V(x) = x^T v$ be the total population in the stations corresponding to the binary vector v. We are interested in the overflow probability in any given subset of the Jackson network. More precisely, we wish to estimate

$$\begin{aligned} p_n^V = \mathbb{P}\{&\text{total population in stations encoded by } v \text{ reaches} \\ &n \text{ before returning to 0, starting from 0}\}. \end{aligned} \tag{2.4}$$

In turn, p_n^V can be expressed in terms of the following stopping times,

$$\begin{aligned} T_{\{x\}} &\triangleq \inf\{k \ge 1 : Q(k) = x\}, \\ T_n^V &\triangleq \inf\{k \ge 1 : V(Q(k)) \ge n\}. \end{aligned}$$

Indeed, if we use the notation $\mathbb{P}_x(\cdot) \triangleq \mathbb{P}(\cdot | Q(0) = x)$ then we can rewrite p_n^V as

$$p_n^V = \mathbb{P}_0(T_n^V \le T_{\{0\}}). \tag{2.5}$$

Similarly,

$$p_n^V(x) = \mathbb{P}_x(T_n^V \leq T_{\{0\}}). \tag{2.6}$$

The asymptotic analysis of $p_n^V(x)$ can be studied by means of large deviations theory. We shall indicate how this theory can be applied to specify an efficient splitting algorithm in the next section. In the mean time, let us provide a representation for the dynamics of the queue length process that will be convenient in order to motivate the elements of the efficient splitting algorithm that we shall analyze.

As mentioned earlier, Jackson networks are basically constrained random walks. The constraints arise because the number of customers in each station must be non-negative. Thinking about Jackson networks as constrained random walks facilitates the introduction and motivation of the necessary large deviations elements behind the description of the splitting algorithm. In order to specify the dynamics of the embedded discrete time Markov chain in terms of a random walk type representation we need to introduce notations which will be useful to specify the transitions at the boundaries induced by the non-negativity constraints.

The state-space $\mathcal{Z}_+^d$ can be partitioned into 2^d different regions which are indexed by all the subsets $E \subseteq \{1, \ldots, d\}$. The region encoded by a given subset E is defined as

$$\partial_E = \{z \in \mathbb{Z}_+^d : z_i = 0, i \in E, z_i > 0, i \notin E\}.$$

The interior of the domain is given by $\partial_\emptyset$ and the origin is represented by $\partial_{\{1,2,\ldots,d\}}$. Subsets other than the empty set represent the "boundaries" of the state-space and correspond to system configurations in which at least one station is empty. The collection of all possible values that the increments of the process Q can take depends on the current region at

which Q is positioned. However, in any case, such collection is a subset of

$$\mathbb{V} \triangleq \{e_i, -e_i + e_j, -e_j : i, j = 1, 2, \ldots, d\},$$

where e_i is the vector whose i-th component is one and the rest are zero. An element of the form e_i represents an arrival at station i, an element of the form $-e_i + e_j$ represents a departure from station i that flows to station j and an element of the form $-e_j$ represents a departure from station j out of the system. The set of all possible departures from station i is a subset of

$$\mathbb{V}_i^- \triangleq \{w : w = -e_i \text{ or } w = -e_i + e_j \text{ for some } j = 1, \ldots, d\}.$$

Because of the non-negativity constraints on the boundaries of the system we have to be careful when specifying the transition dynamics. First we define a sequence of i.i.d. random variables $\{Y(k) : k \geq 1\}$ so that for each $w \in \mathbb{V}$

$$\mathbb{P}(Y(k) = w) = \begin{cases} \lambda_i & \text{if } w = e_i, \\ \mu_i P_{ij} & \text{if } w = -e_i + e_j, \\ \mu_i P_{i0} & \text{if } w = -e_i. \end{cases}$$

The dynamics of the queue-length process admit the random walk type representation given by

$$Q(k+1) = Q(k) + \zeta\left(Q(k), Y(k+1)\right), \tag{2.7}$$

where $\zeta(\cdot)$ is the constrained mapping and it is defined for $x \in \partial_E$ via

$$\zeta(x,w) \triangleq \begin{cases} 0 & \text{if } w \in \cup_{i\in E}\mathbb{V}_i^-, \\ w & \text{otherwise.} \end{cases}$$

The large deviations theory associated with Jackson networks is somewhat similar (at least in form) to that of random walks, technical results can be found in [33, 49] and [57]. One has to recognize, of course, that the non-smoothness of the constrained mapping as a function of the state of the system creates substantial technical complications, but we will leave aside this issue in our discussion because our objective is simply to describe the form of the necessary large deviations results for our purposes. An extremely important role behind the development of large deviations theory for light-tailed random walks is played by the log-moment generating function of the increment distribution. So, given the similarities suggested by the dynamics of (2.7) and those of a simple random walk it is not surprising that the log-moment generating function of the increments, namely,

$$\psi(x,\theta) \triangleq \log \mathbb{E}\left[\exp\left(\theta^T \zeta(x, Y(k))\right)\right] \tag{2.8}$$

also plays a crucial role in the large deviations behavior of $p_n^V(x)$ as $n \nearrow \infty$.

In order to understand the large deviations behavior of p_n^V it is useful to scale space by $1/n$, thereby introducing a scaled queue length process $\{Q_n(k) : k \geq 0\}$ which evolves according to

$$Q_n(k+1) = Q_n(k) + \frac{1}{n}\zeta\left(Q_n(k), Y(k+1)\right).$$

Suppose that $Q_n(0) = y = x/n$ and note that $T_{\{0\}}$ and T_n^V can also be written as

$$T_{\{0\}} = \inf\{k \geq 1 : Q_n(k) = 0\}, T_n^V = \inf\{k \geq 1 : V(Q_n(k)) \geq 1\}.$$

Note that using the scaled queue length process one can write

$$p_n^V(y) = \mathbb{E}\left[p_n^V(y + \frac{1}{n}\zeta(y, Y(1)))\right]. \tag{2.9}$$

Here with a slight abuse of notation we use $p_n^V(y)$ to mean

$$\mathbb{P}\left(T_n^V \leq T_{\{0\}} | Q_n(0) = y\right).$$

Large deviations theory dictates that

$$p_n^V(y) = \exp\left(-nW_V(y) + o(n)\right) \tag{2.10}$$

as $n \nearrow \infty$ for some non-negative function $W_V(\cdot)$. In order to characterize $W_V(\cdot)$ we can combine the previous expression together with (2.9) and a formal Taylor expansion to obtain

$$\begin{aligned}
1 &= \frac{1}{p_n^V(y)}\mathbb{E}\left[p_n^V(y + \frac{1}{n}\zeta(y, Y(1)))\right] \\
&\approx \mathbb{E}\exp\{-nW_V[y + \frac{1}{n}\zeta(y, Y(1))] + nW_V(y)\} \\
&= \mathbb{E}\exp\{-\partial W_V(y)^T\zeta(y, Y(1)) + o(1)\} \\
&= \exp\left(\psi(y, -\partial W_V(y)) + o(1)\right).
\end{aligned}$$

Sending $n \nearrow \infty$ we formally arrive at the equation

$$\psi(y, -\partial W_V(y)) = 0 \tag{2.11}$$

together with the boundary condition $W_V(y) = 0$ if $V(y) \geq 1$. The previous equation is

the so-called Isaacs equation which characterizes the large deviations behavior of $p_n^V(\cdot)$ and it was introduced together with a game theoretic interpretation by Dupuis and Wang in [37]. The solution to (2.11) is understood in a weak sense (as viscosity solution) because the function $W_V(\cdot)$ is typically not differentiable everywhere. Nevertheless, it coincides with a certain calculus of variations representation which can be obtained out of the local large deviations rate function for Jackson networks (see [57]).

An asymptotic lower bound for $W_V(y)$ can be obtained by finding an appropriate subsolution to the Isaacs equation, in which the equality signs in (2.11) are appropriately replaced by inequalities thereby obtaining a so-called subsolution to the Isaacs equation. In particular, $\overline{W}_V(\cdot)$ is said to be a subsolution to the Isaacs equation if

$$\psi(y, -\partial\overline{W}_V(y)) \leq 0 \tag{2.12}$$

subject to $\overline{W}_V(y) \leq 0$ if $V(y) \geq 1$. The subsolution property guarantees $\overline{W}_V(y) \leq W_V(y)$, which translates to an asymptotic logarithmic upper bound of $p_n^V(y)$. The subsolution is said to be maximal at zero if $\overline{W}_V(0) = W_V(0)$. Not surprisingly, subsolutions are easier to construct than solutions and, as we shall discuss in the next section, beyond their use in the development of asymptotic upper bounds they can be applied to the design of efficient simulation procedures. The use of subsolutions to the Isaacs equation for the design of efficient simulation algorithms was introduced in [37]. A derivation of the subsolution equation (2.12) following the same spirit leading to (2.11) using Lyapunov inequalities is given in [18].

As we mentioned in Section 2.2, the efficiency analysis of a rare-event simulation estimator depends on the growth rate of its coefficient of variation. We are interested in an asymptotic analysis that goes beyond the error term $\exp(o(n))$ given by the large deviations approximation (2.10). So, we must enhance the large deviations approximations in

order to provide a more precise estimate for p_n^V. Developing such an estimate is the aim of the following proposition which follows as a consequence of Proposition 2.3 in Section 2.5 of this chapter (see also Proposition 1 and the analysis in Section 5 in [13]).

Proposition 2.1. *There exists $K > 0$ (independent of x and n) such that*

$$p_n^V(x) \leq KP\{V(Q(\infty)) = n\}/P\{Q(\infty) = x\},$$

where Q_∞ is the steady state queue length. Moreover, if $\|x\| \leq c$ for some $c \in (0, \infty)$ then

$$p_n^V(x) = \Omega[P\{V(Q(\infty)) = n\}/P\{Q(\infty) = x\}] \tag{2.13}$$

as $n \nearrow \infty$.

Remark 2.1. *It is important to keep in mind that we shall mostly work with the process $Q(\cdot)$ directly, as opposed to the scaled version $Q_n(\cdot)$ which is used in the analysis of [31].*

The previous proposition provides the necessary means to estimate p_n^V up to a constant; we just need to recall that the distribution of $Q(\infty)$ is computable in closed form (see [64] p. 95). In particular, we have that

$$\begin{aligned}\pi(m_1, \ldots, m_d) &= \prod_{j=1}^{d} \mathbb{P}(Q_j(\infty) = m_j) \\ &= \prod_{j=1}^{d} (1 - \rho_j)\rho_j^{m_j}, \quad j = 1, \ldots, d, \text{ and } m_j \geq 0.\end{aligned}$$

We shall use $\pi(\cdot)$ to denote the stationary measure of Q. In simple words, the previous equation says that the steady state queue length process has independent components which are geometrically distributed. In particular, $P(Q_j(\infty) = m) = \rho_j^m(1 - \rho_j)$ for $m \geq 0$. The next proposition follows directly from standard properties of the geometric

distribution (see Proposition 3 in [13]). Before we proceed, it's useful to look at $V(Q(\infty))$ in the following way. Without loss of generality, we assume

$$V(Q(\infty)) = v^T Q(\infty) = Q_{j_1}(\infty) + \cdots + Q_{j_s}(\infty),$$

i.e., $\{j_1, j_2, \ldots, j_s\}$ are the stations encoded by the vector v. Further suppose that we can group these s stations into k groups by their traffic intensities. In other words, stations in $\{i_1^{\{1\}}, \ldots, i_{m_1}^{\{1\}}\}$ have traffic intensity equal to ρ_{t_1}, ..., stations in $\{i_1^{\{k\}}, \ldots, i_{m_k}^{\{k\}}\}$ have traffic intensity equal to ρ_{t_k}; and we have $m_1 + \cdots + m_k = s$. Now if we define

$$M_i = Q_{j_1^{\{i\}}}(\infty) + \cdots + Q_{j_{m_i}^{\{i\}}}(\infty),$$

then it's clear that the M_i's are negative binomially distributed with parameters m_i and $p_i = 1 - \rho_{t_i}$. Therefore,

$$V(Q(\infty)) = M_1 + \cdots + M_k,$$

is the sum of negative binomial random variables.

Proposition 2.2. *$P[V(Q(\infty)) = n] = \Theta(e^{-n\gamma_V} n^{\beta_V - 1})$, where $\gamma_V = -\log \rho_*^V$, in which $\rho_*^V = \max\{\rho_i : v_i = 1\}$; and $\beta_V = \sum_i I\{\rho_i = \rho_*^V, v_i = 1\}$ is the number of bottleneck stations in the target subset corresponding to v.*

Proof. We have just showed that $V(Q(\infty))$ is the sum of negative binomial random variables, so it suffices to show that if $M_1, \ldots, M_k$ are independent random variables so that M_i is negative binomial with parameters (m_i, p_i) and $p_1 < \cdots < p_k$, then

$$\mathbb{P}(M_1 + \cdots + M_k = n) = \Theta(\mathbb{P}(M_1 = n)) \tag{2.14}$$

as $n \nearrow \infty$; that is, the tail of the probability mass function of the sum of independent

negative binomials has the same behavior as the tail of the heaviest terms in the sum (in this case M_1 has the heaviest tail among the M_j's). In turn, it is easy to verify that $\mathbb{P}(M_1 = n) = \Theta((1-p_1)^n n^{m_1-1})$, so to show the proposition we just need to verify (2.14). We proceed by induction in k. First, let us treat the case $k = 2$. Assume that $p_1 < p_2$ and note that

$$\begin{aligned}
& \mathbb{P}(M_1 + M_2 = n) \\
= & \sum_{j=0}^{n} \mathbb{P}(M_1 = n-j)\,\mathbb{P}(M_2 = j) \\
= & \sum_{j=0}^{n} (1-p_1)^{n-j} p_1^{m_1} \binom{m_1+n-j-1}{m_1-1} (1-p_2)^j p_2^{m_2} \binom{m_2+j-1}{m_2-1} \\
= & \sum_{j=0}^{n} (1-p_1)^{n-j}(1-p_2)^j \Theta((n-j)^{m_1-1} j^{m_2-1}) \\
= & (1-p_1)^n n^{m_1-1} \sum_{j=0}^{n} \left(\frac{1-p_2}{1-p_1}\right)^j \Theta(j^{m_2-1}).
\end{aligned}$$

Since $(1-p_2)/(1-p_1) \in (0,1)$ it follows that the previous sum converges as $n \nearrow \infty$ and therefore we conclude that (2.14) for $k = 2$. Now we assume that the claim is valid for some value $k > 2$, we need to verify the claim for $k+1$. Assume without loss of generality that $p_1 < \cdots < p_k < p_{k+1}$ (otherwise re-label the random variables so that the order of the probabilities is as stated). Note that, by induction hypothesis,

$$\begin{aligned}
\mathbb{P}(M_1 + \cdots + M_{k+1} = n) &= \sum_{j=0}^{n} \mathbb{P}(M_1 + \cdots + M_k = n-j)\,\mathbb{P}(M_{k+1} = j) \\
&= \Theta\left(\sum_{j=0}^{n} \mathbb{P}(M_1 = n-j)\right) \mathbb{P}(M_{k+1} = j).
\end{aligned}$$

The rest of the analysis then proceeds just as in the case of $k = 2$ analyzed earlier, therefore we conclude the proof of the proposition. □

2.4 The Splitting Algorithm

The previous section discussed some large deviations properties required to guide the construction of an efficient splitting scheme using the theory developed in the work of Dean and Dupuis [31]. In order to explain the construction suggested by Dean and Dupuis let us first discuss the general idea behind the splitting algorithm that we shall analyze; a variation of which was first applied to Jackson networks by Villen-Altamirano and Villen-Altamirano [58].

The strategy is to divide the state-space into a collection of regions $\{C_j^n : 0 \leq j \leq l_n(x)\}$ which are nested and that help define "milestone" events that interpolate between the initial position of the process and the target set, which corresponds to the region C_0^n. That is, in our setting we put $C_0^n \triangleq \{x \in \mathcal{S} : V(x) \geq n\}$ and the remaining C_j^n's are placed so that $C_0^n \subseteq C_1^n \subseteq \cdots \subseteq C_{M_n}^n$. How to construct the level sets C_j^n in order to induce efficiency will be discussed below. An observation that is intuitive at this point, however, is that one should have $M_n = \Theta(n)$ so that the next milestone event becomes accessible given the current level. For the moment, let us assume that the C_j^n's have been placed. The splitting algorithm proceeds as follows.

Algorithm SA

1.– *Initiate the simulation procedure with a single particle starting from position $x \in C_k^n$ for a given $k \geq 1$. Let $w_1 = 1$ be the initial weight associated with such particle.*

2.– *Evolve the initial particle until either it hits $\{0\}$ or it hits level C_{k-1}^n. If the particle hits $\{0\}$, then the particle is said to die. If the particle reaches level C_{k-1}^n then it is replaced by r identical particles (for a given integer $r > 1$). The replacing particles are called the immediate descendants or children of the initial particle, which in turn is said to be their parent. The children are positioned precisely at the place where the*

parent particle reached level C_{k-1}^n. The weight w_j associated with the j-th children (enumerate the children arbitrarily) has a value equal to the weight of the parent particle multiplied by $1/r$.

3.– The procedure starting from step 1 is replicated for each of the offspring particles in place; carrying over the value of each of the weights at each level for the surviving particles (the weights of the particles that die can be disregarded).

4.– Steps 1 to 3 are repeated until all the particles either die or reach level C_0^n.

Dean and Dupuis in [31] show how to apply large deviations theory to select the C_j^n's in order to obtain a weakly efficient splitting algorithm. One needs to balance the number of the C_j^n's so that it is not unlikely for a given particle to reach the next level while keeping the total number of particles controlled. We now provide a formal motivation for the use of large deviations for constructing the C_j^n's in a balanced way.

It is convenient, as we did in our formal large deviations discussion in the previous section, to consider the scaled process $Q_n(\cdot)$. Let us assume that the splitting mechanism indicated in Algorithm SA is in place and that our initial position is set at level $Q(0) = x$, so that $Q_n(0) = y = x/n$. The C_j^n's are typically constructed in terms of the level sets of a so-called importance function which we shall denote by $U(\cdot)$. In particular, put $D_n \triangleq \{y \in n^{-1}\mathcal{S} : V(y) < 1\}$ and set $C_j^n = nL_{z_n(j)}$, where

$$L_z \triangleq \{y \in D_n : U(y) \leq z\}, \tag{2.15}$$

and the $z_n(j)$'s are appropriately chosen momentarily. Then, define

$$l_n(x) = \min\{j \geq 0 : x \in C_j^n\} = \min\{j \geq 0 : y \in L_{z_n(j)}\}. \tag{2.16}$$

The total weight corresponding to a particle that reaches level C_0^n given that it started at level $l_n(x)$ is $r^{-l_n(x)}$. In order to have at least a weakly efficient algorithm we wish to achieve two constraints. The first one imposes the aggregate weight of a particle reaching level C_0^n to be $p_n^V(x)\exp(-o(n))$; this would guarantee that the second moment of the resulting estimator achieves asymptotic optimality. The second constraint dictates that the expected number of particles that make it to C_0^n, which is roughly $r^{l_n(x)}p_n^V(x)$ exhibits subexponential growth (i.e. $\exp(o(n))$); this would guarantee a cost per replication that is subexponential. Note that both constraints lead to the requirement of $r^{l_n(x)}p_n^V(x) = \exp(o(n))$. So, given a subsolution $\overline{W}_V(\cdot)$ to the corresponding Isaacs equation, which implies that

$$p_n^V(x) \leq \exp\left(-n\overline{W}_V(x/n) + o(n)\right),$$

it suffices to ensure that

$$l_n(x)\log(r) - n\overline{W}_V(x/n) = o(n). \tag{2.17}$$

The behavior of $l_n(x)$ as $n \nearrow \infty$ only relates to the properties of the function $U(\cdot)$ and it is really independent of the large deviations behavior of the system. In particular, picking $z_n(j) = \Delta j/n, \Delta > 0$ yields $l_n(x) = \lceil nU(x/n)/\Delta \rceil$ and therefore, equation (2.17) suggests that one should select $U(y) = \Delta\overline{W}_V(y)/\log(r)$ with $\overline{W}_V(0) = W_V(0)$ in order to obtain a weakly efficient estimator for p_n^V. This is precisely the conclusion obtained in the work of [31] who present a rigorous analysis that justifies the previous heuristic discussion. Our development in the next section will sharpen the efficiency properties of the sampler proposed in [31] when applied to Jackson networks. So, we content ourselves with the previous heuristic motivation for the splitting method that we will analyze in

the next section and which in turn is based on the viscosity subsolution given by

$$\overline{W}_V(y) = \varrho^T y - \log \rho_*^V, \tag{2.18}$$

where $\varrho_i = \log \rho_i$ for $i = 1, \ldots, d$, see e.g., [39] and [31].

We close this section with a precise definition of the estimator that we will analyze. First, given a constant $\Delta > 0$ (the level size) define $\overline{W}_V(\cdot)$ as indicated in (2.18) for each $y = x/n$ with $x \in \mathcal{S}$. Then, select an integer $r > 1$ and define $U(y) = \Delta \overline{W}_V(y) / \log(r)$. Given the initial position x define the sets $\{C_j^n : 1 \leq j \leq l_n(x)\}$ as indicated above (see equation (2.16)). Run Algorithm SA and let N_n be the number of particles that survive up to C_0^n; their corresponding final weight is $1/r^{l_n(x)}$. Our estimator for $p_n^V(x)$ is simply

$$R_n(x) = N_n(x) / r^{l_n(x)}. \tag{2.19}$$

Now, for the sake of analytical convenience, when analyzing the second moment of $R_n(x)$ we will adopt the so-called *fully branching* representation of the previous estimator (see [31]). Such fully branching representation is obtained by splitting death particles at level zero. In particular, we modify *Algorithm SA* to obtain the following algorithm:

Algorithm SFB

1.– *Initiate the simulation procedure with a single particle starting from position $x \in C_k^n$ for a given $k \geq 1$. Let $w_1 = 1$ be the initial weight associated with such particle.*

2.– *Evolve the initial particle until it either hits $\{0\}$ (and die) or hits level C_{k-1}^n (remain active or alive),* in either case *the particle becomes the parent and is replaced by r descendants, positioned where the parent is located (either $\{0\}$ or the location where it enters level C_{k-1}^n). The weight of the j-th particle is set to equal the weight of its parent multiplied by $1/r$.*

3.– For each living *offspring particle, the procedure starting from step 1 is replicated. For each* dead *offspring particle, replace it by r descendants, set the weight of each child to be that of the parent multiplied by $1/r$.*

4.– Steps 1 to 3 are repeated until all the particles either die or reach level C_0^n.

In other words, after $l_n(x)$ iterations we have $r^{l_n(x)}$ total particles labeled $1, 2, \ldots, r^{l_n(x)}$, each with weight $1/r^{l_n(x)}$. We define I_j as the indicator function of the event that the j-th particle is in C_0^n so that $N_n(x) = \sum_{j=1}^{r^{l_n(x)}} I_j$. The fully branching representation of $R_n(x)$ is simply

$$R_n(x) = r^{-l_n(x)} \sum_{j=1}^{r^{l_n(x)}} I_j. \tag{2.20}$$

2.5 Analysis of Splitting Estimators

We are now in a good position to perform a refined efficiency analysis for the estimator $R_n(x)$. We shall break our analysis into two parts. The first part corresponds to the expected number of particles generated per run and the second part deals with the second moment of $R_n(x)$. We establish upper bounds on both quantities that enable us to reach the conclusion that this multilevel splitting algorithm substantially outperforms the direct polynomial time algorithm for solving the associated system of linear equations.

Our analysis takes advantage of the time reversed process associated with the underlying Jackson network which we shall now define. Given the transition matrix $\{K(x,y) : x, y \in \mathcal{S}\}$ of the process Q, we define the reversed Markov chain $\widetilde{Q} = \{\widetilde{Q}(k) : k \geq 0\}$ via the transition matrix $\widetilde{K}(\cdot)$:

$$\widetilde{K}(y,x) = K(x,y)\,\pi(x)/\pi(y),$$

for $x, y \in \mathcal{S}$. It turns out that $\widetilde{Q}$ also describes the queue length process of an open stable Jackson network with stationary distribution equal to $\pi(\cdot)$, (see [64] p. 95). We will use $\widetilde{P}_x(\cdot)$ to denote the probability measure in path space associated with $\widetilde{Q}$ given that $\widetilde{Q}(0) = x$.

The following result is similar to that of Proposition 1 in [13]. However, our representation in (2.21) is slightly more useful for our purposes.

Proposition 2.3.

$$p_n^V(x) = \frac{\widetilde{\mathbb{P}}_\pi\big(\widetilde{Q}(0) \in C_0^n, \widetilde{T}_{\{x\}} \leq \widetilde{T}_{\{0\}}, \widetilde{T}_{\{x\}} < \widetilde{T}_n^V\big)}{\pi(x) P_x(T_{\{x\}} \geq T_n^V \wedge T_{\{0\}})} \tag{2.21}$$

$$= \frac{\widetilde{\mathbb{P}}_\pi\big(\widetilde{Q}(0) \in C_0^n, \widetilde{\sigma}_{\{x\}} < \widetilde{T}_{\{0\}} < \widetilde{T}_n^V\big)}{\pi(0) P_0\left(\sigma_{\{x\}} < T_n^V \wedge T_{\{0\}}\right)} \tag{2.22}$$

where $\widetilde{T}_n^V = \inf\{k \geq 1 : V(\widetilde{Q}(k)) \geq n\} = \inf\{k \geq 1 : \widetilde{Q}(k) \in C_0^n\}$, $\widetilde{T}_{\{x\}} = \inf\{k \geq 1 : \widetilde{Q}(k) = x\}$, $\sigma_{\{x\}} \triangleq \inf\{k \geq 0 : Q(k) = x\}$ *and* $\widetilde{\sigma}_{\{x\}} \triangleq \inf\{k \geq 0 : \widetilde{Q}(k) = x\}$. *Moreover, there exists* $\delta > 0$ *(independent of* $x \neq 0$ *and* n*) such that*

$$P_x(T_{\{x\}} \geq T_n^V \wedge T_{\{0\}}) \geq \delta. \tag{2.23}$$

Proof. We assume that $x \neq 0$. The case $x = 0$ is included in the analysis of (2.22). First, we observe that

$$\begin{aligned} p_n^V(x) &= \mathbb{P}_x\big(T_n^V < T_{\{0\}}, T_{\{x\}} < T_n^V \wedge T_{\{0\}}\big) + \mathbb{P}_x\left(T_n^V < T_{\{0\}}, T_{\{x\}} \geq T_n^V \wedge T_{\{0\}}\right) \\ &= p_n^V(x)\, \mathbb{P}_x\left(T_{\{x\}} < T_n^V \wedge T_{\{0\}}\right) + \mathbb{P}_x\left(T_n^V < T_{\{0\}}, T_{\{x\}} \geq T_n^V \wedge T_{\{0\}}\right). \end{aligned}$$

Therefore,

$$p_n^V(x) = \frac{\mathbb{P}_x\left(T_n^V < T_{\{0\}}, T_{\{x\}} \geq T_n^V \wedge T_{\{0\}}\right)}{\mathbb{P}_x\left(T_{\{x\}} \geq T_n^V \wedge T_{\{0\}}\right)}.$$

Following the same technique as in Proposition 1 in [13] we have that

$$\begin{aligned}
&\pi(x)\,\mathbb{P}_x\left(T_n^V < T_{\{0\}}, T_{\{x\}} \geq T_n^V \wedge T_{\{0\}}\right) \qquad (2.24)\\
&= \sum_{k=0}^{\infty} \pi(x)\,\mathbb{P}_x\left(T_n^V < T_{\{0\}}, T_{\{x\}} \geq T_n^V \wedge T_{\{0\}}, T_n^V = k\right)\\
&= \sum_{k=1}^{\infty} \pi(x) \sum_{y_0=x, y_1, .., y_{k-1} \in \mathcal{S}\setminus(\{0,x\}\cup C_0^n), y_k \in C_0^n} K(y_0, y_1) \times \cdots \times K(y_{k-1}, y_k)\\
&= \sum_{k=1}^{\infty} \sum_{y_0=x, y_1, .., y_{k-1} \in \mathcal{S}\setminus(\{0,x\}\cup C_0^n), y_k \in C_0^n} \widetilde{K}(y_1, y_0) \times \cdots \times \widetilde{K}(y_k, y_{k-1})\,\pi(y_k).
\end{aligned}$$

Letting $\widetilde{y}_i = y_{k-i}$ for $i = 1, \ldots k$ we see that the summation in each of the terms above ranges over paths $\widetilde{y}_0, \ldots, \widetilde{y}_k$ satisfying that $\widetilde{y}_0 \in C_0^n$, $\widetilde{T}_{\{x\}} = k$ (so in particular $\widetilde{y}_k = x$) and also that $\widetilde{T}_{\{0\}} \geq k, \widetilde{T}_n^V > k$. So, we can interpret the previous sum as

$$\widetilde{\mathbb{P}}_\pi\left(\widetilde{Q}(0) \in C_0^n, \widetilde{T}_{\{x\}} \leq \widetilde{T}_{\{0\}}, \widetilde{T}_{\{x\}} < \widetilde{T}_n^V\right).$$

This yields part (2.21). Part (2.22) corresponds to Proposition 1 of [13]; it follows using the same trick as in the analysis of display (2.24), after multiplying and dividing by $\pi(0)$ when computing the probability of going from zero to the target set via the point x. The most interesting part is the bound (2.23), which is essentially the argument in Proposition 7 of [13], but we discuss it here to make our exposition self contained. We need to show that there exists $\delta > 0$, such that $\mathbb{P}_x(T_{\{x\}} \geq T_n^V \wedge T_{\{0\}}) \geq \delta$ uniformly over $x \neq 0$. The strategy follows the following steps: 1) Argue first that the probability is positive if $x \neq 0$ and, therefore, bounded away from zero over compact sets in x, 2) Now consider the case in which x is outside a suitably defined compact set, then argue that by intersecting with an event involving finitely many service times and routing events inside the network, we can reach a system configuration with m_1 *fewer* customers in the system than the total

number initially present in configuration x, 3) Finally, once we have m_1 fewer customers, argue, using the stability of the Jackson network, that with high probability, the system will eventually empty before coming back to *any* configuration with as many customers as the initial configuration x. Thus, effectively our plan is to show that

$$\inf_{x:x\neq 0} \mathbb{P}_x(T_{\{x\}} \geq T_n^V \wedge T_{\{0\}}) \geq \delta.$$

We now proceed to carry over the previous program. First, if $x \neq 0$, we must clearly have that $\mathbb{P}_x(T_{\{x\}} \geq T_{\{0\}}) > 0$ (i.e. for each $x \neq 0$, the event $T_{\{x\}} > T_{\{0\}}$ is a possible event). To see this, we argue as follows. Note that we have an open Jackson network, so each customer in the system must eventually leave the system if no arrivals are allowed to enter the network. So, if we intersect with the event that the next inter-arrival time into the system is sufficiently large (which clearly is an event with positive probability), we can work *only* with the current customers inside the network, which are distributed in each of the stations according to the state of the system x. Let us use $||x||$ to denote the L_1 norm of x (since the components of x are non-negative, $||x||$ is just the sum of the components of x). If $||x|| \leq m_0$ for some constant m_0, we can always construct an event with the property that, given the initial configuration of the system x, everybody leaves the network prior to an arrival *and* before we find the network once again in the initial configuration x. Observe that if we are forced to cycle back to the initial configuration x with probability one assuming that no arrivals are allowed into the system, then it would *not* be true that each customer must eventually leave the system and this violates the condition that the network is open. Therefore, since the set of configurations x such that $||x|| \leq m_0$ is finite we can find $\delta_0 > 0$ (possibly depending on m_0) such that

$$\inf_{x:x\neq 0, ||x||\leq m_0} \mathbb{P}_x(T_{\{x\}} \geq T_{\{0\}}) \geq \delta_0. \tag{2.25}$$

Now, we proceed with part 2) of the program. Let us assume that $||x|| > m_0$ for $m_0 > 0$ chosen momentarily. Following the same type of reasoning described earlier we have that if $m_1 < m_0$, then we can find $\delta_1 > 0$ (possibly depending on m_1) such that

$$\inf_{||x||\geq m_0} \mathbb{P}_x(T_{\{x\}} \geq T_{||x||-m_1}) > \delta_1,$$

where $T_{||x||-m_1} = \inf\{k \geq 1 : ||Q(k)|| = ||x|| - m_1\}$. In simple words, we can make sure that m_1 customers leave the system prior to an arrival and prior to cycling back to configuration x, regardless of the initial configuration x; this is done by intersecting with an event that depends on the order in which finitely many services are completed and jobs are routed through the network. Therefore, we have that

$$\begin{aligned}\mathbb{P}_x(T_{\{x\}} \geq T_{\{0\}}) &\geq \mathbb{P}_x(T_{\{x\}} \geq T_{\{0\}}, T_{\{x\}} \geq T_{||x||-m_1}) \\ &\geq \delta_1 \inf_{\xi:||\xi||=||x||-m_1} \mathbb{P}_\xi(T_{||x||} \geq T_{\{0\}}).\end{aligned}$$

Finally, we proceed with step 3) of the program, namely, arguing that if m_1 is chosen sufficiently large, then one can actually find $\varepsilon > 0$ such that

$$\sup_{\xi:||\xi||=||x||-m_1} \mathbb{P}_\xi(T_{||x||} < T_{\{0\}}) < 1 - \varepsilon. \tag{2.26}$$

Let $\widetilde{N} = ||x||$ and assume that ξ is such that $||\xi|| = \widetilde{N} - m_1$. We observe that if $\delta_2 > 0$ is chosen small enough, then

$$\mathbb{P}_\xi(T_{||x||} < T_{\{0\}}) = \mathbb{P}_\xi(T_{||x||} < T_{\{0\}}, T_{||x||} \leq \widetilde{N}\delta_2) + \mathbb{P}_\xi(T_{||x||} < T_{\{0\}}, T_{||x||} > \widetilde{N}\delta_2). \tag{2.27}$$

Now, note that

$$\mathbb{P}_\xi(T_{||x||} < T_{\{0\}}, T_{||x||} > \widetilde{N}\delta_2) = \mathbb{E}_\xi[I(T_{||x||} > \widetilde{N}\delta_2)\mathbb{P}_{Q(\widetilde{N}\delta_2)}(T_{||x||} < T_{\{0\}})]. \tag{2.28}$$

Given the initial configuration ξ, large deviation results for Jackson networks (see [49]) guarantee that for any $\epsilon_0 > 0$,

$$\mathbb{P}_\xi\left(||Q(\widetilde{N}\delta_2) - \widetilde{N}q\left(\delta_2\right)|| > \widetilde{N}\epsilon_0\right) = \exp\left(-\widetilde{N}I\left(\epsilon_0\right) + o(\widetilde{N})\right),$$

as $\widetilde{N} \nearrow \infty$ for some $I\left(\epsilon_0\right) > 0$ and some $q\left(\delta_2\right)$ (which corresponds to the fluid limit evaluated at δ_2). In the language of large deviations, the fluid limit corresponds to the zero-cost trajectory. And trajectories outside of the band that centers on the fluid limit have probabilities that decay exponentially fast. Moreover, since the network is stable and open, we have that $||q\left(\delta_2\right)|| < 1-\delta_3$ for some $\delta_3 > 0$. Therefore, once again appealing to the large deviations results of [49], we obtain that if $\epsilon_0 < \delta_3$, then

$$\begin{aligned}\sup_{\{q:||q-q(\delta_2)||<\epsilon_0\}} \mathbb{P}_{\widetilde{N}q}\left(T_{||x||} < T_{\{0\}}\right) &\leq \sup_{\{q:||q||\leq 1-\delta_3+\epsilon_0<1\}} \mathbb{P}_{\widetilde{N}q}\left(T_{\widetilde{N}} < T_{\{0\}}\right) \\ &= O\left(e^{-\delta\widetilde{N}}\right),\end{aligned}$$

for some $\delta > 0$. Consequently,

$$\begin{aligned}& \mathbb{E}_\xi\left(I\big(T_{||x||} > \widetilde{N}\delta_2\big)\mathbb{P}_{Q\left(\widetilde{N}\delta_2\right)}\left(T_{||x||} < T_{\{0\}}\right)\right) \\ \leq\ & \mathbb{P}\left(||Q\big(\widetilde{N}\delta_2\big) - \widetilde{N}q\left(\delta_2\right)|| > \epsilon_0\widetilde{N}\right) + \sup_{\{q:||q||\leq 1-\delta_3+\epsilon_0<1\}} \mathbb{P}_{\widetilde{N}q}\left(T_{||x||} < T_{\{0\}}\right) \\ =\ & O\left(e^{-\delta\widetilde{N}}\right),\end{aligned}$$

for some $\delta > 0$. Therefore the right hand side of (2.28) decreases exponentially fast in $\widetilde{N}$.

It suffices then to study the first term in (2.27). Note that

$$\begin{aligned}\mathbb{P}_\xi(T_{||x||} < T_{\{0\}}, T_{||x||} \leq \widetilde{N}\delta_2) &\leq \mathbb{P}_\xi(\cup_{k\leq \widetilde{N}\delta_2}\{||Q\,(k)\,|| \geq \widetilde{N}\}) \\ &\leq \sum_{k\leq \widetilde{N}\delta_2} \mathbb{P}_\xi(||Q\,(k)\,|| \geq \widetilde{N}).\end{aligned} \tag{2.29}$$

We will apply a Chernoff-bound argument to bound the right hand side of the previous display. Fix an integer $m_3 > 0$ and write $k = m_3 s + l$ for some integer $s \geq 0$ and $l \in \{0, 1, \ldots, m_3 - 1\}$. Let $Q\,(0) = \xi$ and note that

$$\begin{aligned}||Q\,(k)\,|| - ||\xi|| &= ||Q\,(m_3 s + l)\,|| - ||Q\,(m_3 s)\,|| \\ &\quad + \sum_{j=0}^{s-1}[||Q\,(m_3(j+1))\,|| - ||Q\,(m_3 j)\,||].\end{aligned}$$

Because the network is stable it follows that one can choose $m_3 > 0$ (depending only on the characteristics of the network) so that if $||z|| \geq \widetilde{N}(1 - 2\delta_2) > m_3$, then

$$\mathbb{E}_z[||Q\,(m_3)\,|| - ||z||] \leq -\varepsilon_1.$$

In simple words, if the initial population is very large, on average we shall expect more customers to leave than those who arrive. Clearly, one also has that $||Q\,(m_3)\,|| - ||z|| \leq m_3$ (at most m_3 people leave or arrive in m_3 transitions of the network), so we have that one can compute a constant $m_4 > 0$, uniform in z as long as $||z|| \geq \widetilde{N}(1 - 2\delta_2) > m_3$ such that

$$\log \mathbb{E}_z \exp(\theta[||Q\,(m_3)\,|| - ||z||]) \leq -\varepsilon_1\theta + m_4\theta^2.$$

So, selecting $\theta^* > 0$ sufficiently small we obtain that

$$\log \mathbb{E}_z \exp(\theta^*[||Q\,(m_3)\,|| - ||z||]) \leq -\varepsilon_1\theta^*/2. \tag{2.30}$$

Now we are in good shape to apply the Chernoff-bound argument. Note that

$$\begin{aligned}\mathbb{P}_\xi(||Q\,(k)\,|| \geq \widetilde{N}) &\leq \mathbb{P}_\xi(||Q\,(k)\,|| - ||\xi|| \geq m_1) \\ &\leq \exp\left(-\theta^* m_1\right) \exp(\theta^* m_3) \\ &\quad \cdot \mathbb{E}_\xi\left(\theta^* \exp\left(\sum_{j=0}^{s-1}[||Q\,(m_3(j+1))\,|| - ||Q\,(m_3 j)\,||]\right)\right).\end{aligned}$$

Note that we can apply (2.30) repeatedly to estimate the exponential of the the expectation in the previous display given that $||\xi|| = \widetilde{N} - m_1$ and that $k \leq \widetilde{N}\delta_2$, which in particular (because Jackson networks increase or decrease by at most one unit in each transition, and recall that $\widetilde{N}$ is large, so that $m_1 < \widetilde{N}\delta_2$), implies that $||Q\,(k)|| \geq \widetilde{N}(1 - 2\delta_2)$ if $k \leq \widetilde{N}\delta_2$. Therefore, we obtain that

$$\begin{aligned}\mathbb{P}_\xi(||Q\,(k)\,|| \geq \widetilde{N}) &\leq \exp\left(-\theta^* m_1\right) \exp(\theta^* m_3) \exp(-s\varepsilon_1\theta^*/2) \\ &= \exp\left(-\theta^*(m_1 - m_3)\right) \exp(-[k/m_3]\varepsilon_1\theta^*/2).\end{aligned}$$

Adding over k and choosing m_1 sufficiently large we conclude that the right hand side of (2.29) can be made arbitrarily small. (Note that having selected m_1, we then choose $m_0 > m_1$ in the discussion following (2.25)). This combined with our analysis for (2.28) allows us to conclude (2.26) and therefore we conclude our result. □

Proposition 2.1 and 2.2 from Section 2.3 follow as a consequence of this result, the rest of the details are given in Section 5 of [13]. Nevertheless, in the interest of making this chapter as self-contained as possible, without compromising its length, we mention that

the most difficult part remaining in Proposition 2.1 involves the lower bound in equation (2.13). For this part, one can use identity (2.22) combined with a similar analysis behind (2.23) to show that there exists $\delta > 0$ such that for all n large enough

$$\widetilde{\mathbb{P}}_{\pi}\left(\widetilde{\sigma}_{\{x\}} < \widetilde{T}_{\{0\}} < \widetilde{T}_{n}^{V} | \widetilde{Q}\left(0\right) \in C_{0}^{n}\right) \geq \delta.$$

The rest of the argument behind Proposition 2.1 and 2.2 from Section 2.3 then follows from elementary properties of the steady-state distribution $\pi\left(\cdot\right)$.

Given the subsolution we proposed in Section 2.4, the importance function can be written as

$$\begin{aligned} U\left(x/n\right) = \overline{W}_{V}\left(x/n\right)\frac{\Delta}{\log r} &= \left(\frac{1}{n}\varrho^{T}x - \log\rho_{*}^{V}\right)\frac{\Delta}{\log r} \\ &= C\left(\Delta - \frac{1}{n}\alpha^{T}x\Delta\right), \end{aligned} \tag{2.31}$$

where $C = -\log\rho_{*}^{V}/\log r$, and $\alpha = \varrho/\log\rho_{*}^{V}$. The level index function also simplifies to

$$l_{n}\left(x\right) = \left\lceil\frac{nU\left(x/n\right)}{\Delta}\right\rceil = \left\lceil nC\left(1 - \frac{1}{n}\alpha^{T}x\right)\right\rceil = \lceil C(n - \alpha^{T}x)\rceil. \tag{2.32}$$

We shall first look at the expected number of surviving particles of the splitting algorithm which characterizes the stability of the algorithm. One shall keep in mind that when the complexity of the splitting algorithm is concerned, what actually matters is the total function evaluation involved in each run. An upper bound is obtained for this quantity, as measured by the sum of all particles generated at interim levels weighted by the maximum remaining function evaluations associated with each of them. We first have the following result.

Proposition 2.4. *The expected terminal number of particles for the splitting algorithm specified by* (Δ, U) *above satisfies*

$$\mathbb{E}\left[N_n\left(x\right)\right] = \Theta\left(n^{\beta_V - 1}\right) \tag{2.33}$$

where β_V*, introduced in Proposition 2.2, denotes the number of bottleneck stations corresponding to the vector* v*.*

Proof. It can be seen from the *fully-branching* algorithm that

$$\mathbb{E}\left[N_n\left(x\right)\right] = r^{l_n(x)}\, p_n^V\left(x\right).$$

From Proposition 2.2 we know that $p_n^V\left(x\right) = \Theta(\pi^{-1}(x)e^{-\gamma_V n}n^{\beta_V - 1})$. Since $e^{-\gamma_V} = e^{\log \rho_*^V} = e^{-C\log r} = r^{-C}$, we can write $p_n^V\left(x\right) = \Theta(\pi^{-1}(x)r^{-nC}n^{\beta_V - 1})$. Hence, plug in $l_n(x) = \lceil C(n - \alpha^T x)\rceil$, and note that $\pi^{-1}(x) = \widetilde{c}r^{C\alpha^T x}$ for some positive constant $\widetilde{c}$, we have

$$\mathbb{E}\left[N_n\left(x\right)\right] = \Theta\left(r^{C\alpha^T x} r^{-nC} n^{\beta_V - 1} r^{\lceil C\left(n - \alpha^T x\right)\rceil}\right) = \Theta\left(n^{\beta_V - 1}\right).$$

□

As pointed out earlier, the number of terminal surviving particles, although a reasonable proxy to measure the stability of the algorithm, is not suitable for quantifying the complexity. We also need to take into account the number of function evaluations required to generate $R_n\left(x\right)$. The next result addresses precisely this issue.

Proposition 2.5. *The expected computational effort per run required to generate a single replication of* $R_n\left(x\right)$ *is* $O(n^{\beta_V + 1})$*.*

To prove this, we need the following result, which upper bounds the probability that a particle makes it to the level $C^n_{l_n(x)-m}$. We first state the result and postpone the proof

until after the proof of Proposition 2.5.

Proposition 2.6. *For a given generation m, denote by $Q_{m,j}$ the position of the j-th particle, then*

$$\mathbb{P}_x\left(Q_{m,1} \in C^n_{l_n(x)-m}\right) = O\left(\left(\frac{m-1}{C}\right)^{\beta_V - 1}\left(\rho^V_*\right)^{\frac{m-1}{C}}\right). \tag{2.34}$$

Given this result, we now proceed to prove Proposition 2.5.

Proof of Proposition 2.5. Let N^n_m, $m = 0, \ldots, l_n(x)$, be the number of particles that survive to level $C^n_{l_n(x)-m}$. Again fully-branching algorithm allows us to write

$$\mathbb{E}[N^n_m] = r^m \mathbb{P}_x\left(Q_{m,1} \in C^n_{l_n(x)-m}\right).$$

Thanks to Proposition 2.6, along with $\left(\rho^V_*\right)^{-1/C} = r$, we have

$$\mathbb{E}[N^n_m] = O\left(r^m\left(\frac{m-1}{C}\right)^{\beta_V-1}\left(\rho^V_*\right)^{\frac{m-1}{C}}\right) = O\left(r\left(\frac{m-1}{C}\right)^{\beta_V-1}\right). \tag{2.35}$$

Also let $\eta_{m,j}$ be the remaining computational effort of the j-th particle at the start of the m-th level until it either reaches the next level or it dies out. Put $\overline{\eta}_{m,j}(x_j)$ to be the expectation of $\eta_{m,j}$ given that the position of the j-th particle at the start of level m is x_j. Note that the norm of the position of x_j is less than $c \cdot m$ for a given constant c that depends on the traffic intensities of the system but not on the position of the particle per-se. Therefore, it is easy to see that

$$\sup_{1 \le j \le N^n_m} \overline{\eta}_{m,j}(x_j) \le c \cdot m, \tag{2.36}$$

for some $c \in (0, \infty)$. Intuitively, each particle at level m either advances to the next level,

or it dies out by hitting the zero level before moving to the next one, since it takes $\Theta(1)$ work to cross one single layer, $\eta_{m,j}$ is dominated by the work required to die out, and hence its mean is bounded from above by $c \times m$ for some constant c. Using (2.35) and (2.36), we can bound the expected total work per run as follows

$$\begin{aligned}
\mathbb{E}\left[\sum_{m=0}^{l_n(x)-1}\sum_{j=1}^{N_m^n}\eta_{m,j}\right] &= \sum_{m=0}^{l_n(x)-1}\mathbb{E}\left[\sum_{j=1}^{N_m^n}\overline{\eta}_{m,j}\left(x_j\right)\right] \\
&\leq \sum_{m=0}^{l_n(x)-1}\mathbb{E}\left[N_m^n\right]\cdot c\cdot m \\
&\leq c'\cdot\sum_{m=0}^{l_n(x)-1}\left(\frac{m-1}{C}\right)^{\beta_V-1} m \\
&= O\big(n^{\beta_V+1}\big),
\end{aligned}$$

for some positive constant c and c' where in the last step we use the definition of $l_n(x)$ given in (2.32). □

It remains to prove Proposition 2.6.

Proof of Proposition 2.6. We begin the proof with an important property implied by the splitting algorithm:

$$\begin{aligned}
V\left(Q_{m,1}\right) > 0 &\Leftrightarrow Q_{m,1} \in C^n_{l_n(x)-m} = nL_{(l_n(x)-m)\Delta/n} \\
&\Leftrightarrow Q_{m,1} \in \{z \in nD_n : U\left(z/n\right) \leq \left(l_n\left(x\right)-m\right)\Delta/n\} \\
&\Leftrightarrow Q_{m,1} \in \left\{z \in nD_n : C\left(1-\frac{1}{n}\alpha^T z\right) \leq \frac{1}{n}\left(C\left(n-\alpha^T x\right)-m+1\right)\right\} \\
&\Leftrightarrow Q_{m,1} \in \{z \in nD_n : \alpha^T z \geq \alpha^T x + \frac{m-1}{C}\} \\
&\Leftrightarrow Q_{m,1} \in \{z \in nD_n : \varrho^T z \leq \varrho^T x - (m-1)\log r\} \qquad (2.37)
\end{aligned}$$

where we used the representations of $U(\cdot)$ and $l_n(x)$ in (2.31) and (2.32) and the definition

of L_z in (2.15). In other words, if a particle survives m generations then its current position is beyond the m-th level, which implies that the weighted sum of system population, with weight given by the vector ϱ, is bounded from above by that of the initial position adjusted by a linear function in m. If we define the stopping time $\hat{T}_{\frac{m}{C}} \triangleq \inf\{k \geq 1 : \alpha^T Q(k) \geq \alpha^T x + \frac{m-1}{C}\} = \inf\{k \geq 1 : \varrho^T Q(k) \leq \varrho^T x - (m-1)\log r\}$, the above property also implies that $Q_{m,1} \in C^n_{l_n(x)-m} \Leftrightarrow \hat{T}_{\frac{m}{C}} < T_0$. Following an argument similar to the proof of (2.21) in Proposition 2.3 (in fact easier because here we are interested in an upper bound only), it follows that there exists constant $\hat{c} > 0$, independent of x and m, such that

$$\begin{aligned}
\mathbb{P}_x\left(Q_{m,1} \in C^n_{l_n(x)-m}\right) &= \mathbb{P}_x\left(\hat{T}_{\frac{m}{C}} < T_0\right) \\
&\leq \frac{\hat{c}}{\pi(x)}\mathbb{P}\left[\varrho^T Q(\infty) \leq \varrho^T x - (m-1)\log r\right] \\
&= \frac{\hat{c}}{\pi(x)}\mathbb{P}\left[\alpha^T Q(\infty) \geq \alpha^T x + \frac{(m-1)}{C}\right].
\end{aligned}$$

To finish the proof we need the following Lemma.

Lemma 2.1.

$$\begin{aligned}
\mathbb{P}\left[\alpha^T Q(\infty) \geq \alpha^T x + \frac{(m-1)}{C}\right] &= \Theta\left[\mathbb{P}\left(Z\left(\beta_V, 1-\rho_*^V\right) \geq \alpha^T x + \frac{m-1}{C}\right)\right] \\
&= \Theta\left[\left(\frac{m-1}{C}\right)^{\beta_V - 1}\left(\rho_*^V\right)^{\frac{m-1}{C}}\right]
\end{aligned}$$

where $Z(n,p)$ denotes a $NBin(n,p)$ (negative binomial) random variable.

Proof of Lemma. Note that

$$\begin{aligned}\alpha^T Q(\infty) &= Q(\infty)^T \frac{\varrho}{\log \rho_*^V} \\ &= \sum_{i=1}^{d} Q_i(\infty) I\left(\rho_i = \rho_*^V\right) + \sum_{i=1}^{d} Q_i(\infty) I\left(\rho_i \neq \rho_*^V\right) \frac{\log \rho_i}{\log \rho_*^V} \\ &= Z\left(\beta_V, 1-\rho_*^V\right) + W.\end{aligned}$$

One direction is elementary, since $\alpha^T Q(\infty) \geq Z\left(\beta_V, 1-\rho_*^V\right)$, we clearly have

$$\mathbb{P}\left[\alpha^T Q(\infty) \geq \alpha^T x + \frac{(m-1)}{C}\right] \geq \mathbb{P}\left[Z\left(\beta_V, 1-\rho_*^V\right) \geq \alpha^T x + \frac{(m-1)}{C}\right]. \tag{2.38}$$

For the other direction, note that there exists constants $c_4 > 0$, and $\widetilde{\rho} < \rho_*^V$ such that

$$\begin{aligned}W &= \sum_{i=1}^{d} Q_i(\infty) I\left(\rho_i \neq \rho_*^V\right) \frac{\log \rho_i}{\log \rho_*^V} \\ &\leq c_4 \sum_{i=1}^{d} Q_i(\infty) I\left(\rho_i \neq \rho_*^V\right) \\ &\leq_{st} c_4 Z\left(d-\beta_V, 1-\widetilde{\rho}\right),\end{aligned}$$

where " $\leq_{st}$" denotes that the left hand side is stochastically dominated by the right hand side. As a result,

$$\alpha^T Q(\infty) \leq_{st} Z\left(\beta_V, 1-\rho_*^V\right) + c_4 Z\left(d-\beta_V, 1-\widetilde{\rho}\right).$$

But since $1-\rho_*^V < 1-\widetilde{\rho}$, a similar argument as given in the proof of Proposition 2.2

allows us to obtain

$$\begin{aligned} & \mathbb{P}\left[\alpha^T Q(\infty) \geq \alpha^T x + \frac{(m-1)}{C}\right] \\ \leq \quad & c_0 \mathbb{P}\left[Z\left(\beta_V, 1-\rho_*^V\right) \geq \alpha^T x + \frac{(m-1)}{C}\right], \end{aligned} \tag{2.39}$$

for some finite constant c_0 that is independent of m. Combining (2.38) and (2.39), we have

$$\begin{aligned} & \mathbb{P}\left[\alpha^T Q(\infty) \geq \alpha^T x + \frac{(m-1)}{C}\right] \\ = \quad & \Theta\left[\mathbb{P}\left(Z\left(\beta_V, 1-\rho_*^V\right) \geq \alpha^T x + \frac{(m-1)}{C}\right)\right]. \end{aligned} \tag{2.40}$$

Using again Proposition 3 of [13], we reach the conclusion that

$$\mathbb{P}\left[\alpha^T Q(\infty) \geq \alpha^T x + \frac{(m-1)}{C}\right] = \Theta\left[\left(\frac{m-1}{C}\right)^{\beta_V - 1} \left(\rho_*^V\right)^{\frac{m-1}{C}}\right]$$

□

The result of Proposition 2.6 directly follows. □

To facilitate the analysis of the second moment of $R_n(x)$ we add the following notations. We follow the analysis in [31] to make our exposition here self-contained. For a given generation m, denote by $Q_{m,j}$ the position of the j-th particle; recall that the accumulated weight up to the m-th stage of such a particle is r^m. Let $\chi_{m,j}$ be the disjoint grouping of particles in the next generation (i.e., $m+1$) according to their "parents" in generation m. For $k \in \chi_{m,j}$, denote by d_k the offsprings of this particle at the final stage

$l_n(x)$. We then have the following expansion of the second moment of $R_n(x)$:

$$\begin{aligned} & \mathbb{E}_x\left[\left(\sum_{j=1}^{r^{l_n(x)}} I_j\, r^{-l_n(x)}\right)^2\right] \\ = & \sum_{m=0}^{l_n(x)-1} \mathbb{E}_x\left[\sum_{j=1}^{r^m} \sum_{k,l\in\chi_{m,j},k\neq l} \left(\sum_{m_k\in d_k} I_{m_k} r^{-l_n(x)}\right)\left(\sum_{m_l\in d_l} I_{m_l} r^{-l_n(x)}\right)\right] \\ & + \mathbb{E}_x\left[\sum_{j=1}^{r^{l_n(x)}} I_j\, r^{-2l_n(x)}\right], \end{aligned} \tag{2.41}$$

where we define I_{m_k} to be the indicator function of the event that particle m_k is in the set C_0^n. The second term above is essentially the diagonal terms of the second moment (2.41), and for the off-diagonal terms, for each generation, we categorize particles according to their common ancestors, a technique used by [31]. For the first term, we have

$$\begin{aligned} & \sum_{m=0}^{l_n(x)-1} \mathbb{E}_x\left[\sum_{j=1}^{r^m} \sum_{k,l\in\chi_{m,j},k\neq l} \left(\sum_{m_k\in d_k} I_{m_k} r^{-l_n(x)}\right)\left(\sum_{m_l\in d_l} I_{m_l} r^{-l_n(x)}\right)\right] \\ = & \sum_{m=0}^{l_n(x)-1} \mathbb{E}_x\left[\sum_{j=1}^{r^m} I\left(V\left(Q_{m,j}\right)>0\right)\left(r^{-m}\right)^2\right. \\ & \left.\cdot \sum_{k,l\in\chi_{m,j},k\neq l} \left(\frac{1}{r}\sum_{m_k\in d_k} I_{m_k} r^{-(l_n(x)-m-1)}\right)\left(\frac{1}{r}\sum_{m_l\in d_l} I_{m_l} r^{-(l_n(x)-m-1)}\right)\right]. \end{aligned}$$

Conditioning on the whole genealogy up to step m, we obtain

$$\begin{aligned} & \mathbb{E}_x\left[\sum_{j=1}^{r^m} I\left(V\left(Q_{m,j}\right)>0\right)\left(r^{-m}\right)^2\right. \\ & \left.\cdot \sum_{k,l\in\chi_{m,j},k\neq l} \left(\frac{1}{r}\sum_{m_k\in d_k} I_{m_k} r^{-(l_n(x)-m-1)}\right)\left(\frac{1}{r}\sum_{m_l\in d_l} I_{m_l} r^{-(l_n(x)-m-1)}\right)\right] \end{aligned}$$

$$
\begin{aligned}
= \quad & \mathbb{E}_x \left[\sum_{j=1}^{r^m} I\left(V\left(Q_{m,j}\right)>0\right) \left(r^{-m}\right)^2 \mathbb{E}_x \left(\sum_{k,l \in \chi_{m,j}, k \neq l} \right. \right. \\
& \left. \left. \left(\frac{1}{r} \sum_{m_k \in d_k} I_{m_k} r^{-(l_n(x)-m-1)} \right) \left(\frac{1}{r} \sum_{m_l \in d_l} I_{m_l} r^{-(l_n(x)-m-1)} \right) \Big| Q_{m,j} \right) \right] \\
= \quad & \mathbb{E}_x \left[\sum_{j=1}^{r^m} I\left(V\left(Q_{m,j}\right)>0\right) r^{-2m} \sum_{k,l \in \chi_{m,j}, k \neq l} \right. \\
& \left. \left(\frac{1}{r} \mathbb{E}_{Q_{m,j}} \left(\sum_{m_k \in d_k} I_{m_k} r^{-(l_n(x)-m-1)} \right) \frac{1}{r} \mathbb{E}_{Q_{m,j}} \left(\sum_{m_l \in d_l} I_{m_l} r^{-(l_n(x)-m-1)} \right) \right) \right].
\end{aligned}
$$

Note that

$$
\mathbb{E}_{Q_{m,j}}\left[\sum_{m_k \in d_k} I_{m_k} r^{-(l_n(x)-m-1)} \right] = p_n^V\left(Q_{m,j}\right),
$$

and $\mathcal{W} = \sum_{k,l \in \chi_{m,j}; k \neq l} r^{-2} = (r-1)/r$. Summing over m we obtain

$$
\begin{aligned}
& \mathbb{E}_x \left[\left(\sum_{j=1}^{r^{l_n(x)}} I_j \, r^{-l_n(x)} \right)^2 \right] - \mathbb{E}_x \left(\sum_{j=1}^{r^{l_n(x)}} I_j r^{-2l_n(x)} \right) \\
= \quad & \mathcal{W} \sum_{m=0}^{l_n(x)-1} \mathbb{E}_x \left[\sum_{j=1}^{r^m} I\left(V\left(Q_{m,j}\right)>0\right) r^{-2m} p_n^V\left(Q_{m,j}\right)^2 \right] \\
= \quad & \mathcal{W} \sum_{m=0}^{l_n(x)-1} r^{-m} \mathbb{E}_x \left[I\left(V\left(Q_{m,1}\right)>0\right) p_n^V\left(Q_{m,1}\right)^2 \right].
\end{aligned}
$$

Combining this with the diagonal term in (2.41), which can be readily expressed as $r^{-l_n(x)} p_n^V(x)$, we arrive at the following expansion for the second moment of $R_n(x)$:

$$
\begin{aligned}
\mathbb{E}_x\left[R_n(x)^2\right] \quad = \quad & \mathcal{W} \sum_{m=0}^{l_n(x)-1} r^{-m} \mathbb{E}_x \left[I\left(V\left(Q_{m,1}\right)>0\right) p_n^V\left(Q_{m,1}\right)^2 \right] \\
& + r^{-l_n(x)} p_n^V(x).
\end{aligned}
\tag{2.42}
$$

The next result takes advantage of expression (2.42) to obtain an upper bound for

$\mathbb{E}_x[R_n(x)^2]$.

Proposition 2.7. *The second moment of $R_n(x)$ satisfies*

$$\mathbb{E}\left[R_n(x)\right]^2 = p_n^V(x)^2\, O\left(n^{\beta_V}\right). \tag{2.43}$$

where β_V is the number of bottleneck stations in the subset corresponding to V.

In order to prove the previous result, we will show that the second moment of $R_n(x)$ is dominated by the first item on the right hand side of the equality in (2.42). In turn, the asymptotic behavior of such term hinges on the conditional distribution of the exact position of the particle in generation m, $Q_{m,1}$ in $C^n_{l_n(x)-m}$.

Proof. Using the equivalence observed in (2.37), the expectation term in the sum of (2.42) can be expressed as

$$\begin{aligned} &\mathbb{E}_x\left[I\left(V\left(Q_{m,1}\right)>0\right)p_n^V\left(Q_{m,1}\right)^2\right] \\ =\ &\mathbb{E}_x\left[I\left(\varrho^T Q_{m,1}\leq \varrho^T x-(m-1)\log r\right)p_n^V\left(Q_{m,1}\right)^2\right] \\ =\ &\mathbb{E}_x\left[p_n^V\left(Q_{m,1}\right)^2 \left|\varrho^T Q_{m,1}\leq \varrho^T x-(m-1)\log r\right.\right]\mathbb{P}_x\left(\hat{T}_{\frac{m}{C}}<T_0\right) \end{aligned} \tag{2.44}$$

where we used the property derived in (2.37). Before we proceed, let us define the inverse mapping $V^{-1}:\mathcal{Z}_+\to\mathcal{Z}_+^d$ by

$$V^{-1}(n)=\{x\in\mathcal{Z}_+^d : V(x)=n\},$$

i.e., the configuration of the network such that the total population in stations encoded by v is n. For the first item in (2.44), we have

$$\mathbb{E}_x\left[p_n^V\left(Q_{m,1}\right)^2 \left|\varrho^T Q_{m,1}\leq \varrho^T x-(m-1)\log r\right.\right]$$

$$\leq K\mathbb{E}\left[\frac{\pi^2\left(V^{-1}(n)\right)}{\pi^2\left(\{Q_{m,1}\}\right)}|\varrho^T Q_{m,1} \leq \varrho^T x - (m-1)\log r\right] \tag{2.45}$$
$$= K\pi^2\left(V^{-1}(n)\right)c_1\mathbb{E}_\pi\left[e^{-2\varrho^T Q_{m,1}}|\varrho^T Q_{m,1} \leq \varrho^T x - (m-1)\log r\right]$$

where c_1, K are some constants independent of n. Here for the inequality we used Proposition 1. To reach the equality we used the fact that $\pi^{-1}\left(\{Q_{m,1}\}\right) = c_1 e^{-\varrho^T Q_{m.1}}$ for some positive constant c_1. As for the expectation term in (2.45), since the process $Q(\cdot)$ has for each dimension an increment at most of unit size, we can write

$$\begin{aligned} & \mathbb{E}_\pi\left[e^{-2\varrho^T Q_{m,1}}|\varrho^T Q_{m,1} \leq \varrho^T x - (m-1)\log r\right] \qquad (2.46)\\ = \quad & \mathbb{E}_\pi\left[e^{-2\varrho^T Q_{m,1}}|\varrho^T x - (m-1)\log r - \delta \leq \varrho^T Q_{m,1} \leq \varrho^T x - (m-1)\log r\right] \\ \leq \quad & c_2\exp\left(-2\varrho^T x + 2(m-1)\log r\right) \\ = \quad & c_3\exp\left(-2\frac{m-1}{C}\log\rho_*^V\right) = c_3\left(\rho_*^V\right)^{-2\frac{m-1}{C}}, \end{aligned}$$

where c_2, c_3 and δ are some positive constants. Combining this with

$$\mathbb{P}_x\left(\hat{T}_{\frac{m}{C}} < T_0\right) = O\left(\left(\frac{m-1}{C}\right)^{\beta_V - 1}\left(\rho_*^V\right)^{\frac{m-1}{C}}\right)$$

according to Proposition 2.6, we obtain the following upper bound for the expectation term in the sum of expression (2.42):

$$\begin{aligned} & \mathbb{E}_x\left[I\left(V\left(Q_{m,1}\right) > 0\right)p_n^V\left(Q_{m,1}\right)^2\right] \\ = \quad & K\pi^2\left(V^{-1}(n)\right)\pi^{-2}(x)\left(\rho_*^V\right)^{-2\frac{m-1}{C}} O\left(\left(\frac{m-1}{C}\right)^{\beta_V-1}\left(\rho_*^V\right)^{\frac{m-1}{C}}\right) \\ = \quad & O\left(p_n^V(x)^2 r^{m-1}\left(\frac{m-1}{C}\right)^{\beta_V-1}\right) \qquad (2.47) \end{aligned}$$

where for the second equality we used again Proposition 2.1 and the fact that $\rho_*^V = r^{-C}$.

Putting the bound in (2.47) back to the sum in the first item of (2.42), we have

$$\begin{aligned} & \sum_{m=0}^{l_n(x)-1} r^{-m}\mathbb{E}_x\left[I\left(V\left(Q_{m,1}\right)>0\right)p_n^V\left(Q_{m,1}\right)^2\right] \\ = \quad & r^{-1}\sum_{m=0}^{l_n(x)-1} O\left(p_n^V(x)^2\left(\frac{m-1}{C}\right)^{\beta_V-1}\right) \\ = \quad & p_n^V(x)^2\, O\left(n^{\beta_V}\right). \end{aligned} \tag{2.48}$$

Finally, note that the second item of (2.42) is dominated by (2.48), and it follows immediately that

$$\mathbb{E}\left[R_n(x)\right]^2 = p_n^V(x)^2\, O\left(n^{\beta_V}\right).$$

□

Equipped with these results, we are ready to summarize our discussions in the statement of the following Theorem, which is the main result of this chapter.

Theorem 2.1. *To estimate the overflow probability $p_n^V(x)$ using $R_n(x)$, the number of function evaluations needed for a given level of relative error is $O(n^{2\beta_V+1})$.*

Proof. Recall from Section 2.2 that the number of function evaluations sufficient to achieve a pre-determined level of relative accuracy for the splitting estimator is proportional to the work-normalized squared coefficient of variation. This is therefore immediate by combining the upper bound analysis of the computational effort per run in Proposition 2.5 along with the upper bound of the second moment of $R_n(x)$ available in Proposition 2.7. □

A direct comparison to the $O(n^{3d-2})$ complexity of solving a system of linear equations (see Section 2.2) yields the immediate conclusion that the splitting algorithm is "efficient" in the sense that it is an improvement over the "benchmark" polynomial algorithm. Even

in the worst case scenario, when we look at the total population of the network and the network is totally symmetric, i.e., all stations are bottlenecks ($\beta_V = d > 3$), the number of function evaluations needed is a substantial reduction of n^{d-3}. In the case where $\beta_V = 1$, the algorithm only requires a number of function evaluations that at most grows cubically in the level of overflow n. Furthermore, if the number of bottlenecks is less than half of the total number of stations, i.e. $\beta_V < d/2$, the splitting algorithm enjoys a running time of order smaller than $O(n^d)$, which is not worse than storing the vector that encodes the solution to the associated linear system. If, on the other hand, more than half of the stations are bottlenecks, faster importance sampling based algorithms do exist at least for the case of tandem networks; see the analysis in [18], which implies that $O(n^{2(d-\beta)+1})$ function evaluations suffice to obtain an estimator with a given relative precision. Overall, the analysis thus provides some sort of guidance on the choice of simulation algorithms. It is meaningful to point out that the previous comparison is not based on the sharpest analysis. In fact we only resort to a rather crude upper bound in the analysis of the second moment of $R_n(x)$ in (2.45). A sharper result is possible by bounding the expectation term in (2.44) with more care. But as pointed out in the Introduction, even though there is still room for a more refined analysis, we believe our work provides substantial insights leading to a better understanding of the relations between these two classes of algorithms.

Remark 2.2. *Numerical experiments have been performed for this class of algorithms in [31]. We replicated some of their experiments and from the numerical evidence we could see that there is still room for a sharper bound. In particular, when studying overflow for the total population of the network, our experiments suggest a computational cost roughly similar to $O(n^{\beta_V})$ (as opposed to $O(n^{2\beta_V+1})$) for a fixed level of relative error. We have chosen not to present the numerical details in this chapter since we think a sharper analysis is needed for a better interpretation of the results. The rough $O(n^{\beta_V+1})$*

additional effort in our estimate, we believe, comes from the application of (2.34) in the proofs of both Proposition 2.5 and Proposition 2.7. Note that the bound becomes too loose when the position of the survival particle at level m satisfying $V(Q_{m,1}) > 0$ is no longer $O(1)$. Instead, conditional on a particle surviving at level $m = \Theta(n)$, the particle is with high probability in the most likely fluid trajectory to overflow. However, to account for its exact position, we would need a conditional local central limit theorem correction. This accounts for a factor of $n^{\beta_V/2}$ in both 1) expected computational effort per run for a single replication of the estimator and 2) the second moment of the estimator. Combining these two terms seems to explain most of the gap between our bound and what appears to be the actual empirical performance.

Do not fear going forward slowly; fear only to stand still.

Chinese Proverb

3

Splitting for Heavy-tailed Systems: An Exploration with Two Algorithms

3.1 Introduction

The design of simulation algorithms to estimate rare event probabilities in heavy-tailed systems has been dominated by importance sampling based strategies, for example [16], [34], [15], [23] and [20] , to name a few. In light-tailed systems where the inputs have exponentially decaying tails, in contrast, both importance sampling and

splitting are popular approaches applied in the construction of efficient rare event simulation algorithms (see [8]). In simple words, importance sampling involves simulating the system under consideration according to a different set of probabilities under which the occurrence of the rare event is less unlikely. A weight is then attached to each simulation corresponding to the likelihood ratio of the observed outcome relative to the original distribution. Whereas, in splitting, the effort of biasing the behavior of the system is replaced by laying out a sequence of "milestone" events (with the last milestone event corresponding to the target event) whose sequential occurrence is no longer rare. Particles are then evolved according to the system's dynamics and kept splitting whenever a new milestone is reached. Attached with each particle is a weight defined by the total number of times it has split so that the final estimator is unbiased. We refer readers to [45] for a review of earlier developments in the splitting method and the references therein.

In fact, recent research suggests that, in the light tailed setting, splitting and importance sampling based algorithms are very much related. When rare event probabilities can be approximated using conventional large deviations techniques, the exponential rate of decay is characterized by means of a variational problem (see [32]). The work of [35] and [36] shows that asymptotically optimal importance sampling strategies can be constructed out of *smooth* subsolutions of the HJB equations associated with the variational problem for the rate of decay of the target probability. Later [31] shows how to design splitting based algorithms for the same class of problems that enjoy a comparable asymptotic optimality properties. But the design, instead of requiring the construction of smooth subsolutions of the associated HJB equations, relies on subsolutions of a weaker sense, which are often easier to construct.

In contrast, we are not aware of any provably efficient splitting algorithms studied in the literature that are tailored for the heavy-tailed systems. Why is the landscape so much

different in the heavy-tailed realm? The difficulty stems from the fundamentally different large deviations descriptions of the heavy-tailed system from its light-tailed counterparts. In light-tailed systems, the story behind the applicability of efficient splitting technique lies in the "collaborative" effect among all the system inputs. Under the guidance of this principle, the "optimal" trajectory is predictable given the current position of the random walk. In contrast, it's not possible, in the heavy-tailed setting, to steer the system along the "most likely" path. This is because only one or very few jumps contribute to the occurrence of large deviations in systems with heavy-tailed inputs, which we refer to as the "single jump domain" and the "multiple jump domain", respectively. (For rigorous accounts we refer readers to [48], [42] and [71].) Such an "individual" effect among the increments, which differs considerably from the large deviations theory in the light-tailed setting, implies that *any* sample path can stand out to be an "optimal" one. Consider the classical problem of estimating $\mathbb{P}(X_1 + \cdots + X_n > b)$, where the X_i's are i.i.d. suitably heavy-tailed random variables. The observation that no large increments have occurred up to the $(n-k)$-th increment, $1 \leq k < n$, doesn't lead to the conclusion that the trajectory followed by the current path is not "important". Consequently, we expect that any *level placement* strategies would result in a splitting algorithm that performs no better than crude Monte Carlo.

In this chapter we take the step to explore rare event simulation via splitting based simulation algorithms for heavy-tailed stochastic systems. A very natural class of problems to start with is the tail probability of sums of random variables,

$$q(b) = \mathbb{P}(S_n > b), \tag{3.1}$$

where $S_n = X_1 + X_2 + ... + X_n$. Here the X_i's are i.i.d. random variables, with a suitable heavy-tailed structure. This class of problem has been a classical problem in the

operations research field, which is motivated by estimating the steady state large delay probabilities in a M/G/1 queue (see e.g, [6]) that has been served as a vehicle to initialize the studies of importance sampling algorithms for rare event simulations.

We have to point out, however, that there are indeed a few very efficient important sampling based algorithms, the development of which was enlightened by the distinct characteristics of the large deviations theory for heavy-tailed random walks. To name a few, the work of [34] develops a state-dependent two-point mixture importance sampling algorithm to estimate the probability $\mathbb{P}(S_N > b)$ where S_N is a random walk with regularly varying inputs and N can be either deterministic or random that satisfies $\mathbb{E}\left(z^N\right) < \infty$ for some $z > 1$. The authors of [22] propose using a multiple mixture as the importance sampling distribution for random walk that admits a large class of subexponential inputs (see the definition in Section 3.2 for the definition of subexponential distributions.). In [20], a state-dependent importance sampling estimator is constructed for estimating the tail distribution of compound sums of i.i.d. subexponential random variables. These three algorithms have been shown (albeit using different methods) to admit *strong efficiency*, which implies that the number of replications needed to achieve a pre-determined level of relative accuracy is bounded as the probability of interest decreases. Strong efficiency is a more powerful notion of efficiency than logarithmic efficiency (see again Section 3.2 for a brief review). (See also [17] for an in-depth survey on the recent advances of state-dependent importance sampling for rare-event simulation.) Therefore, the goal of this chapter is not trying to develop an algorithm that is superior in efficiency to some of the existing algorithms; but rather we contribute by giving a first attempt to explore the idea of splitting in rare event simulation for heavy-tailed systems, and we hope the work will lay the ground for future work in this direction. Our motivation is to see if, as in the light-tailed case, splitting algorithms might have a hope of being easier to set up while

still maintaining provable efficiency, in the form of logarithmic efficiency (also known as asymptotic optimality, see [17]). As we shall see, we conclude that, in some sense, there seems to be some evidence that this may well be the case.

The different nature of how large deviations occur in a heavy-tailed system forces us to abandon the idea of splitting in the original state space. Our idea is *hazard function splitting* for the system input X_j's. Instead of splitting in the original state space, we embed a splitting procedure in the hazard function space, and then transform back to the original space to obtain the sampled increments. We propose two related algorithms based on this idea. In the sense that we sample the increments via their hazard function, our algorithms are closest in spirit to the importance sampling based *hazard rate twisting* algorithm in [51]. We show that if properly set up, both splitting algorithms guarantee *logarithmic efficiency*. While it is in some sense not surprising that such a splitting based strategy is less efficient than importance sampling strategies, the design of these splitting algorithms is *uniform in the class of system inputs*. In contrast to importance sampling, which requires different types of distributions depending on tail properties (see [22]). In that regard, the splitting based algorithms benefit from an easier set-up, in a similar spirit to the light-tailed case.

The rest of the chapter is organized as follows. Section 3.2 formally defines the problem we work on, and lists the assumptions of the hazard function in which splitting occurs. A brief review on the notion of efficiency is also provided. We describe the first hazard function splitting idea in detail in Section 3.3. Based on this idea, we propose two related splitting-based algorithms. The first one, based on a resampling step on top of the splitting procedure, is introduced in Section 3.4, the analyses of which are carried out in Section 3.5. In Section 3.6, an improved algorithm is constructed and analyzed, in parallel to the development ins Section 3.4 and 3.5. We end the discussion with some numerical

examples in Section 3.7.

3.2 Problem Setting and Assumptions

Consider a probability space $(\Omega, \mathcal{F}, \mathbb{P})$. Let $\{X_j, j \leq n\}$ be a series of independent, continuous random variables with distribution function given by $F(\cdot)$, with support $(0, \infty)$. The spectrum of distributions we are considering is specified in the following assumption on the hazard function $\Lambda(x)$.

Assumption 3.1. *We assume the following conditions on the hazard functions,* $\Lambda(x) = -\log \overline{F}(x)$*, to hold:*

1) $\Lambda(X)$ *is strictly increasing in* x*.*

2) The hazard rate function, $\lambda(x) = \Lambda'(x)$*, is eventually everywhere differentiable.*

3) $\Lambda(x) \sim x^\beta L(x)$*, for some* $0 \leq \beta < 1$ *and* $L(\cdot)$ *is some slowly varying function, i.e.,* $\lim_{x\to\infty} L(tx)/L(x) = 1$ *for any* $t > 0$*.*

It's not hard to verify that the distributions covered by the previous assumption fall into the subexponential family (see Definition 1.4) by directly checking Pitman's condition (see Lemma 1.1). Note that the strictly increasing restriction implies that Λ is bijective and therefore allows a unique solution to $x = \Lambda^{-1}(y)$ for $y > 0$, which is critical to the applicability of our splitting algorithm.

These mild assumptions on the hazard function enable us to operate on a practical subset of the subexponential family:

i) $\beta = 0$. *Regularly varying* distributions (see Definition 1.7) belong to this realm. It's easy to see that $\Lambda(x) = -\log\left(\overline{F}(x)\right) = -\alpha \log x + o(\log x)$ which is slowly

varying. To a less obvious extent are *lognormal distributions*. Consider a lognormal distributed random variable X with parameters μ and σ, it's easy to verify that

$$\overline{F}(x) = \mathbb{P}(X > x) = \overline{\Phi}\left(\frac{\ln x - \mu}{\sigma}\right) \sim \frac{c}{\log x} \exp\left(-\frac{(\log x - \mu)^2}{2\sigma^2}\right)$$

for some positive constant c. It therefore implies that the hazard function satisfies $\Lambda(x) = -(\log x)^2 / (2\sigma^2) + o\left(\log^2 x\right)$, again slowly varying.

ii) $0 < \beta < 1$. *Weibull distributions* with decreasing failure rate (i.e., $\overline{F}(x) = \exp(-\lambda x^{-\eta})$, for $\eta \in (0, 1)$) fall into this category.

3.3 Hazard Rate Splitting

Our splitting algorithms builds upon the following well-known observation:

$$\mathbb{P}(\Lambda(X) > x) = \mathbb{P}\left(X > \Lambda^{-1}(x)\right) = \exp(-x), \tag{3.2}$$

where $\Lambda(\cdot)$ is the hazard function of X. It is convenient to take advantage of the memory-less property of the exponential distribution to implement a particle splitting procedure in terms of $\Lambda(X)$. In this section we introduce a splitting procedure with fixed step size in the space of the hazard function $\Lambda(X)$. In particular, particles that reach a high level are favored and split. Moreover, higher levels in the space of $\Lambda(X)$ correspond to subsequent larger jumps in the space of X.

3.3.1 Splitting Mechanism and "Tree" Construction

Sampling of a random variable X is conducted in two phases: in the first phase we use a splitting based procedure to sample the lifetime of $\Lambda(X)$, which is exponentially

distributed with unit rate according to (3.2), and in the second phase, we transform it back to the original space with the inverse function $\Lambda^{-1}(\cdot)$. Given the state independent nature of the idea, it suffices to focus our attention momentarily on the generation of a single component.

The splitting based procedure is perhaps best described in terms of a "tree" construction procedure. To fix ideas, let us denote by Π the tree to be constructed in the space of X's hazard function $\Lambda(\cdot)$. Let Δ be a pre-determined positive number. We first section the hazard function, $\Lambda(\cdot)$, into a series of milestone levels. Define $m(b)$, the total number of Δ-sized levels via

$$m = m(b) = \min\{k \leq 1 : k\Delta \geq \Lambda(b)\} = \lceil \Lambda(b)/\Delta \rceil.$$

Moreover, let us define the mapping $\tau(k)$, $k = 0, \ldots, m$ by $\tau(k) = [k\Delta, (k+1)\Delta)$, if $0 \leq k \leq m-1$, and $\tau(m) = [m\Delta, \infty)$. In other words, $\tau(k)$ is the k-th level in the hazard function space.

Now, we start with a single "active" particle, endowed with unit weight. A tree is constructed by propagating and splitting the particle in the space of the hazard function. During the tree construction procedure to be introduced shortly, the particles are grouped as active or inactive in a dynamic way. An active particle may keep splitting and propagating, until it becomes inactive, since then it remains at the position where it turns inactive. Each particle will evolve through at most m generations. Let us denote by $Z(k)$ and $D(k)$ the number of active and inactive particles at level k, or generation k, $0 \leq k \leq m$. The formal definitions will be provided later in (3.5) and (3.6). We shall refer to the set of all the inactive particles after m generations as the set of *leafs* in the

final tree, defined as

$$\mathcal{L}(\Pi) = \sum_{k=0}^{m} D(k). \tag{3.3}$$

The final tree, Π, is characterized by the heights of those leafs. For now let us denote by $V(s)$ the height of leaf s, $s \in \mathcal{L}(\Pi)$. The tree is constructed in the following "process-like" manner:

Tree Construction via Particle Propagation and Splitting

1) At the beginning of *generation* k, $1 \leq k \leq m$, each "active" particle $1 \leq s \leq Z(k-1)$ is given an exponential lifetime, $A_k(s)$. Set $Z(k) = D(k) = 0$. For $k \leq m-1$,

   - if $A_k(s) > \Delta$, the particle is "split" and replaced by $r \in \mathbb{N}$ "descendant" particles $\{s_1, \ldots, s_r\}$, each carrying a weight equal to $1/r$ times the weight of their "parent", and remains active at level $k+1$. Set $Z(k) = Z(k) + r$.
   - if, however, $A_k(s) < \Delta$, the particle is said to be "dead" or "inactive", and will stay in $\tau(k)$ until the end of the procedure. Set $D(k) = D(k) + 1$, and $V(s) = k\Delta + A_k(s)$.

2) For each $s \in Z(m)$, set $V(s) = m\Delta + A_m(s)$.

The final tree is therefore encoded by the vector $\{V(s)\}_{s \in \mathcal{L}(\Pi)}$. Note that if $V(s) \in \tau(k)$, it carries a weight equal to r^{-k}, $k = 1, 2, \ldots, m$. Furthermore, define the random variable $L = L(s)$ to be the level attained by leaf s. And define

$$W(L) = W(L(s)) = AI(L(s) = m) + AI(A \leq \Delta) I(L(s) < m). \tag{3.4}$$

Then we obtain

$$V = V(s) = L(s)\Delta + W(L(s)).$$

An illustration of a constructed tree is shown in Figure 3.1.

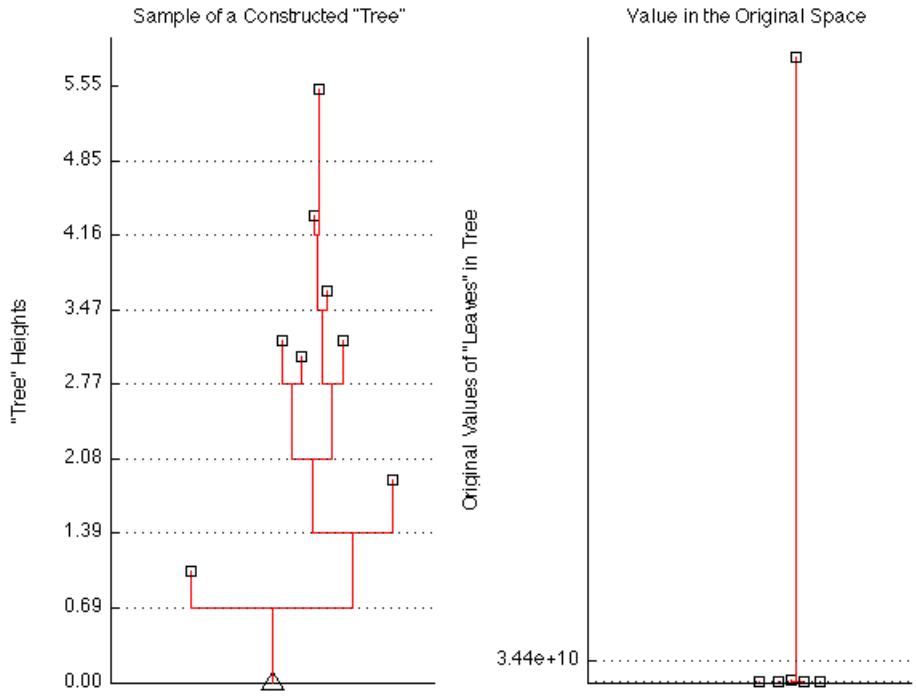


Figure 3.1: Example of a constructed tree. In this example, $b = 10^{12}, \alpha = 0.2$. The subgraph on the left illustrates a constructed tree in the hazard function of the increment X. The subgraph on the right shows the sampled values (in the original space) of those black-colored leafs in the tree on the left.

It's well-known that splitting procedures that take place in the original state space of the stochastic processes (see, e.g., [45] and [31]) require careful treatment of level placements in order to achieve logarithmic efficiency (see the analysis in, e.g., [44] and [19]). If one adopts a fixed number of descendants per split, one general guideline is (see Section VI.9 of [8]) to distribute the milestone levels such that the conditional probability of the process reaching the $(k+1)$-st level given it gets to the k-th level is roughly identical. However for many cases it's not easy to analytically find such an alignment of the levels. This becomes less of a concern in our tree construction procedure described above. In

particular, let q_k be the conditional probability of a particle reaching level k given it has reached level $k-1$, for $k = 1, \ldots, m$, then the memoryless property ensures that

$$p = p_k = \exp(-\Delta).$$

This particular feature brings up extra convenience in the performance analysis of the algorithm. The fixed level crossing probability p enables us to easily apply elementary properties of branching processes to analyze the performance of the splitting algorithm. In fact, it's not hard to realize that the active and inactive sets of the particles, $\{Z(k)\}_{1\le k\le m}$, $\{D(k)\}_{1\le k\le m}$ can be defined underlying a *standard Galton Watson branching process*. In particular,

$$Z(k+1) = \sum_{j=1}^{Z(k)} rI(j, k+1), \tag{3.5}$$

where $I(j, k+1)$ equals to one if the jth particle at level k makes it to the next level and zero otherwise. We have that $\mathbb{E}(I(j, k+1)) = q = \exp(-\Delta)$. Define

$$\begin{aligned} D(k) &= \sum_{j=1}^{Z(k)} \overline{I}(j, k+1) = Z(k) - \frac{Z(k+1)}{r},\ k = 0, \ldots, m-1, \\ D(m) &= Z(m), \end{aligned}$$

where $\overline{I} = 1 - I$.

3.3.2 Fully Branching Representation of Π

Before we proceed, we shall introduce a *fully branching* representation of the tree, Π, constructed using the procedure described in the previous subsection. A similar description can be found in [31]. The representation is particularly convenient in the second moment analysis (see Subsection 3.5.2) of the splitting estimator to be introduced in the next

section.

Let us denote the fully branching tree by Π'. In a nutshell, Π' can simply be constructed from Π by *replacing each* $s \in \mathcal{L}(\Pi)$ *with a "cluster" of* $r^{m-L(s)}$ *identical leafs*. Note that because

$$\sum_{k=0}^{m} D(k) r^{m-k} = r^m, \tag{3.6}$$

the fully branching tree, Π' has exactly r^m leafs at the top, each carrying weight equal to r^{-m}.

Recall that the tree Π is constructed via particle propagation and splitting through m generation in the hazard function space. We therefore have the following equivalent description in terms of the splitting procedure. In particular, Π' is obtained by *forcing each inactive particle to split until the end of the* m*-th generation*. More precisely, consider a single particle, instead of "killing" it at level k, we "pretend" that it keeps splitting for another $m-k$ times. When being inactive, each time it splits, it is replaced by r inactive descendent particles, inheriting the same position as their parent particle, and carrying a weight equal to $1/r$ times the weight of their parent. The particles and weights of Π therefore has a one-to-one correspondence with the leafs and weights of Π. In what follows we shall refer to a fully branching tree, Π' as a *full tree* (recall that we refer to Π simply as tree).

3.4 A Splitting-Resampling Algorithm

We are now in a good position to propose our first splitting based algorithm. Suppose that a tree Π has been constructed using the procedure introduced in the previous section. The idea of the algorithm is to judiciously resample a leaf s from $\mathcal{L}(\Pi)$. Once the leaf, say s_0, has been chosen, the corresponding sampled value for random variable X is realized

the following transformation

$$X = \Lambda^{-1}\left(L(s_0)\Delta + W\left(L(s_0)\right)\right).$$

The resampling distribution should, intuitively, place more probabilities to those leafs at higher levels, which correspond to larger values of X in the original space, due to the increasing function Λ^{-1}.

It's not hard to see that sampling from the leafs is equivalent to sampling from the associated level set $\{L(s)\}_{s\in\mathcal{L}(\Pi)}$. Conditioning on the realization of the tree, Π, define

$$\mathbb{P}_0\left(L = l\right) = \frac{D(l)r^{m-l}}{\sum_{k=0}^{m} D(k)r^{m-k}} = D(l)r^{-l},\ l = 0, \ldots, m, \tag{3.7}$$

where we have used (3.6). Simply put, under $\mathbb{P}_0$, the probability of the levels are proportional to the number of leafs at level l in Π'. From now on we shall refer to the probability measure given by $\mathbb{P}_0$ as the *full-tree measure*. Clearly, sampling the levels L from the full-tree measure is equivalent to uniformly sampling from the r^m leafs from the full tree, Π'. To this end we have left the choice of the integer r unspecified. With Δ fixed, the behavior of $D(k)$ is directly controlled by r; the larger the choice of r, the larger $D(k)$ turns out to be on average. We shall see momentarily that $D(k)$ grows approximately at a rate equal to $r\exp\left(-\Delta\right)$. It is meaningful at this point to reiterate the general principle of the splitting method: whether applied to the original state space, or in this case to the hazard function space, splitting aims to induce the occurrence of rare events by inflating the number of subpaths as they enter rarer intermediate levels. Translating this to the sampling of L means that we shall place $\Theta(1)$ probabilities to higher levels of the tree. Based on our discussions just now, sampling L from the full-tree measure amounts to,

approximately, sampling from

$$\mathbb{P}\left(L = l\right) = D(l)r^{-l} \approx e^{-\Delta l},$$

i.e., a geometric distribution with parameter $p = \exp\left(-\Delta\right)$, which is no different from the full-tree measure with $r = 1$. In other words, it seems almost futile from a variance reduction point of view to apply splitting to construct Π (Π'), and then sample the level L (and hence the leaf) using the full-tree measure. Indeed, the probabilities of the levels under $\mathbb{P}_0$ deflates too much the importance of those leafs at higher levels of the tree (due to the term r^{-l}). Therefore, we shall search for some alternative level sampling measure that balances out the following two criteria:

1. Places higher, $\Theta(1)$ probabilities to higher levels in the tree.
2. Produces a likelihood ratio (with respect to the tree measure) that does not grow too fast.

Sampling measures that satisfy these conditions will likely lead to an algorithm that enjoys logarithmical efficiency.

Consider the following parametric sampling distribution for L:

$$\widetilde{\mathbb{P}}_\theta\left(L = l\right) = \frac{\theta^{-l}D(l)}{\sum_{k=0}^{m}\theta^{-k}D(k)}, \tag{3.8}$$

where θ is some parameter satisfying $1 \leq \theta \leq r$ to be chosen in the sequel. Clearly $\widetilde{\mathbb{P}}_r$ is identical to $\mathbb{P}_0$. And $\theta = 1$ corresponds to sampling $L = l$ with probability proportional to the number of "clusters" present at level l in Π' (or equivalently, proportional to the number of leafs at level l in Π). We shall show in Section 3.5 that any $\widetilde{\mathbb{P}}_\theta$ with $\theta \leq 1$

won't produce a logarithmically efficient algorithm because it violates Criterion 2 above, i.e., the likelihood ratio grows too fast. In what follows we shall call the sampling measure associated with $\widetilde{\mathbb{P}}_\theta$ the θ- *sampling measure* for the level L.

Going back to the classical problem of estimating $q(b) = \mathbb{P}(S_n > b)$. Before we proceed to describe our first splitting estimator for $q(b)$, let's put up with a few additional notations. Denote by Π_j the tree constructed for X_j, $j \leq n$. Given $\Delta > 0$ and $1 \leq \theta \leq r$, define

$$Z_j(k) \stackrel{d}{=} Z(k),\ D_j(k) \stackrel{d}{=} D(k),\ m_j = \lceil \Lambda(b)/\Delta \rceil, N_j(\theta) = \sum_{k=0}^{m_j} \theta^{-k} D_j(k). \tag{3.9}$$

for $j = 1, \ldots, n$. Let $L_j(s_j)$ denote the sampled level for Π_j, where s_j is the associated leaf in $\mathcal{L}(\Pi_j)$. In what follows we shall simply write L_j to refer to $L_j(s_j)$ for notational convenience. Finally, let $W_j = W_j(L_j) \stackrel{d}{=} W(L)$, where $W(L)$ is defined in (3.4). The *Hazard Function Splitting-Resampling* (HFSR) algorithm for $q(b)$ is therefore described as follows.

The Hazard Function Splitting-Resampling (HFSR) Algorithm

For each $j = 1, \ldots, n$:

1) Construct Π_j.

2) Resample a leaf $s_j \in \mathcal{L}(\Pi_j)$ by resampling L_j from the θ-sampling measure $\widetilde{\mathbb{P}}_\theta(\cdot)$.

3) Given L_j, sample $W_j = W(L_j)$.

4) Estimate $q(b)$ with the following *HFSR estimator*

$$R_\theta(b) = \widetilde{\mathbb{E}}_\theta \left[I\left(\sum_{j=1}^{n} \Lambda^{-1}(L_j\Delta + W_j) > b \right) \prod_{j=1}^{n} \left(e^{-L_j\Delta} N_j(\theta) \right) \right], \tag{3.10}$$

where the expectation $\widetilde{\mathbb{E}}_\theta$ is taken under the θ-sampling measure $\widetilde{\mathbb{P}}_\theta$, and $\prod_{j=1}^{n}\left(e^{-L_j\Delta}N_j\left(\theta\right)\right)$ is the likelihood ratio between the nominal tree measure $\mathbb{P}_0$ and the θ-sampling measure $\widetilde{\mathbb{P}}_\theta$.

3.5 Analysis of the Splitting-Resampling Algorithm

To this point, the choices of the splitting parameters (Δ, r) along with the level sampling parameter θ have been left open. In this section, we fill these gaps while analyzing the performance of the HFSR estimator $R_\theta(b)$. We found out that, in order to guarantee logarithmic efficiency, one must properly

1. inflate the number of particles across the tree in the splitting phase;
2. resample the leaf according to a sampling measure which corresponds to resampling the leafs uniformly from a critical tree.

The first goal is achieved by tuning the parameter r such that the Galton-Watson process $Z(k)$ is slightly supercritical. To achieve the second goal, we must pick the sampling parameter θ in a savvy way. In fact, as we shall unveil soon, provided with a fixed pair of (Δ, r), only one choice of θ guarantees logarithmical efficiency.

3.5.1 Number of Particles

Recall from Subsection 1.2.4 that logarithmic efficiency requires the work normalized coefficient of variation $Var\left(R_\theta(b)\right)\mathcal{W}(b)/q(b)^2$ to grow at an $o\left[1/q(b)\right]$ rate. This implies that the work required for a single replication, given by $\mathcal{W}(b)$ can only grow at most at the following rate

$$\log \mathcal{W}(b) = o\big[-\log q\left(b\right)\big],$$

as $b \nearrow \infty$. Consider the tree constructed using the procedure introduced in Subsection 3.3.1, it's reasonable to proxy $\mathcal{W}(b)$ by the expected total number of leafs generated throughout the tree because the number of elementary function evaluations to generate and maintain each particle is $\Theta(1)$. In particular, we shall write in our case

$$\mathcal{W}(b) = O\left[\mathbb{E}\left(\sum_{j=1}^{n}\sum_{k=0}^{m_j} D_j(k)\right)\right].$$

Therefore, the splitting parameter r has to be chosen such that the total number of leafs generated in any of the n trees constructed satisfies $\log \mathbb{E}(N) = o\big[-\log q\,(b)\big]$, as $b \nearrow \infty$, where $N \stackrel{\Delta}{=} \sum_{k=0}^{m} D_1(k)$. We also need to keep in mind that, the level sampling distribution becomes meaningless if the resulting number of the leafs, $D(k)$'s, are insignificant. We therefore also need to appropriately choose r so that the tree is not too sparse. In addition, the expected number of leafs at the top level of the tree shall be expected to have the same order as the total number of leafs in the tree. It turns out that if we properly choose the splitting parameter r, the cost per replication $\mathcal{W}(b)$ satisfies the aforementioned requirements. Before proceeding to the result, we state the following lemma, which will be used in the second moment analysis as well.

Lemma 3.1. *Let* $\gamma = r\exp(-\Delta)$. *Recall that* $N(\gamma) = \sum_{k=0}^{m} \gamma^{-k} D(k)$, *where* $m = \lceil \Lambda(b)/\Delta \rceil = \lceil -\log q(b)/\Delta \rceil$. *We have*

$$\mathbb{E}\left[N(\gamma)^d\right] = \Theta\left[m^d\right] = \Theta\left[(-\log q(b))^d\right],\ d = 1, 2, \tag{3.11}$$

as $b \nearrow \infty$.

Proof. From the elementary theory of branching processes ([47]),

$$\mathbb{E}Z(k) = \left[\phi'(1)\right]^k = \left(re^{-\Delta}\right)^k = \gamma^k,$$

where $\phi(s) = s^r \exp(-\Delta) + 1 - \exp(-\Delta)$ is the probability generating function of the number of progeny of the Galton Watson process Z. And therefore,

$$\mathbb{E}D(k) = \mathbb{E}\left[Z(k) - Z(k+1)/r\right] = (1 - \exp(-\Delta))\,\gamma^k,$$

for $0 \leq k \leq m-1$, and $\mathbb{E}D(i) = \mathbb{E}Z(m) = \gamma^m$. As a result,

$$\mathbb{E}N(\gamma) = \sum_{k=0}^{m} \gamma^{-k}\mathbb{E}D(k) = (1 - \exp(-\Delta))\,m + 1 = \Theta\left[-\log q(b)\right].$$

On the other hand,

$$\mathbb{E}Z(k)^2 = \sigma^2 \frac{\gamma^{k-1}\left(\gamma^k - 1\right)}{\gamma - 1} + \gamma^{2k} = \Theta\left[\gamma^{2k}\right],$$

where $\sigma^2 = Var\left(Z(1)\right) = re^{-\Delta}\left(1 - e^{-\Delta}\right) = \gamma\left(1 - e^{-\Delta}\right)$. Moreover, observe that $D(k) \leq Z(k), \forall k \leq m$. Therefore, on assuming, without loss of generality, $k \leq l$ (the case $k \geq l$ is symmetric) we obtain the following by elementary algebra

$$\mathbb{E}\left[D(k)D(l)\right] = \Theta\left[\mathbb{E}\left(Z(k)Z(l)\right)\right] = \Theta\left[\mathbb{E}\left(Z(k)^2\gamma^{l-k}\right)\right] = \Theta\left[\gamma^{k+l}\right].$$

Finally,

$$\mathbb{E}N(\gamma)^2 = \sum_{k=0}^{m}\sum_{l=0}^{m} \gamma^{-(k+l)}\mathbb{E}\left[D(k)D(l)\right] = \Theta\left[m^2\right] = \Theta\left[\left(-\log q(b)\right)^2\right].$$

□

As a direct consequence of Lemma 3.1, we have the following bound on the cost per replication $\mathcal{W}(b)$.

Theorem 3.1. *There exists $\xi > 0$, independent of b, such that if $r = e^{\Delta(1+\xi)}$, then, given*

any $\epsilon > 0$,

$$\mathcal{W}(b) = \Theta\big[\mathbb{E}D(m)\big] = o\left[1/q(b)^{\epsilon}\right],$$

as $b \nearrow \infty$.

Proof. For the first equality of the result, note that

$$\begin{aligned}\mathbb{E}\left[\sum_{k=0}^{m} D(k)\right] &= \sum_{k=0}^{m-1}\Big(\mathbb{E}Z(k) - \mathbb{E}\left[Z(k+1)\right]/r\Big) + \mathbb{E}Z(m) \\ &= \left(1-e^{-\Delta}\right)\sum_{k=0}^{m-1}\exp\left(\xi\Delta k\right) + \gamma^{m} = \Theta\left[\mathbb{E}D(m)\right].\end{aligned}$$

For the second equality, just note from Lemma (3.1) that

$$\mathbb{E}\left[\sum_{k=0}^{m} D(k)\right] \geq \mathbb{E}\left[\sum_{k=0}^{m} \gamma^{-k} D(k)\right] = \mathbb{E}\left[N\left(\gamma\right)\right] = \Theta\left[m\right].$$

□

Remark 3.1. *We recognize that the sampling of each* X_j *does involve one array sorting and searching procedure. However, algorithms with modest complexity, for example, merge sort and binary search, require at most* $O\left[m\log m\right] = o\left[1/q(b)^{\epsilon}\right]$, *for any* $\epsilon > 0$ *as* $b \nearrow \infty$. *It therefore suffices to consider the expected number of particles generated throughout the trees.*

3.5.2 Logarithmic Efficiency and Optimal Choice of θ

The next and more challenging question to tackle is, what is a reasonable choice of θ to ensure a proper growth of $\overline{CV^2}\left(R_{\theta}^2(b)\right)$ in order to have logarithmic efficiency? The question ultimately boils down to the design of the level sampling distribution $\widetilde{\mathbb{P}}_{\theta}$. In the previous section we have briefly touched upon the general principle of choosing such

a distribution. In what follows let us assume that $\xi > 0$ has been chosen by the user and the trees have been constructed based on $r = \exp((1+\xi)\Delta)$. The first intuition amounts to a choice of θ such that under $\widetilde{\mathbb{P}}_\theta$, sampling levels that are close to level m shall have a significantly higher probability than that under the full-tree measure. We know that the tree is constructed such that both $Z(k)$ and $D(k)$ grows on average at the rate $\gamma = r\exp(-\Delta) = \exp(\xi\Delta)$. If $\theta = \exp(\xi\Delta)$,

$$\widetilde{\mathbb{P}}_\theta(L = l) \propto \exp(-\xi\Delta l)\, D(l) \approx 1.$$

Therefore, $\theta = \gamma = \exp(\xi\Delta)$ seems to be a good start. Note that this choice corresponds to sampling the leafs *from a critical tree*. The following theorem justifies this selection.

Theorem 3.2. *Given the notations in (3.9), if*

$$\theta = \gamma = \exp(\xi\Delta) = r\exp(-\Delta),$$

where $\xi > 0$ is some fixed small number, then the HFSR estimator

$$R_\gamma(b) = I\left(\sum_{j=1}^{n} \Lambda^{-1}(L_j\Delta + W_j) > b\right) \prod_{j=1}^{n} \left(e^{-L_j\Delta} N_j(\gamma)\right), \tag{3.12}$$

is a logarithmically efficient estimator for $q(b) = \mathbb{P}(S_n > b)$. Here the expectation $\widetilde{\mathbb{E}}_\gamma$ is taken under the γ-sampling measure defined as $\widetilde{\mathbb{P}}_\gamma = \widetilde{\mathbb{P}}_{\theta|\theta=\gamma}$, where $\widetilde{\mathbb{P}}_\theta$ is defined in (3.8).

In order to prove the result, we need the following result. It appeared as Lemma 3.1 in [51].

Lemma 3.2. *With the hazard functions $\Lambda(\cdot)$ satisfying Assumption 3.1, we have, for*

every $\epsilon > 0$*, there exists* $b(\epsilon) > 0$*, such that*

$$\sum_{j=1}^{n} \Lambda(x_j) \geq \Lambda\left(\sum_{j=1}^{n} x_j\right) - \epsilon,$$

for all $(x_1, \ldots, x_n) \geq 0$ *with* $\sum_{j=1}^{n} x_j > b(\epsilon)$.

Proof. See [51]. □

Proof of Theorem 3.2. For notational convenience let us suppress the subscript $_\gamma$ in $\widetilde{\mathbb{P}}_\gamma$ and $\widetilde{\mathbb{E}}_\gamma$ throughout the proof.

1) *Unbiasedness.* It suffices to show that

$$\widetilde{E}\left[R_\gamma(b)\right] = \mathbb{P}_0\left(\sum_{j=1}^{n} \Lambda^{-1}\left(L_j\Delta + W_j\right) > b\right) = \mathbb{P}\left(\sum_{j=1}^{n} X_j > b\right).$$

Let us again write $V_j = \Lambda(X_j) = L_j\Delta + W_j, j = 1, 2, \ldots, n$. Let $\tau(l)$ be defined as in the beginning of Subsection 3.3.1. We then have

$$\begin{aligned}
& \mathbb{P}_0\left(\sum_{j=1}^{n} \Lambda^{-1}\left(L_j\Delta + W_j\right) > b\right) \\
= \quad & \mathbb{E}_0\left[\mathbb{E}_0\left(I\left(\sum_{j=1}^{n} \Lambda^{-1}\left(V_j\right) > b\right) \middle| \{V_j\}_{j=1}^{n}\right)\right] \\
= \quad & \mathbb{E}_0\left[\sum_{j=1}^{n}\sum_{l_j=0}^{m_j} D_j(l_j) r^{-l_j} \mathbb{E}_0\left(I\left(\sum_{j=1}^{n} \Lambda^{-1}\left(V_j\right) > b\right) \middle| V_j \in \tau(l_j)\right)\right].
\end{aligned}$$

Note that by virtue of the definition of the full-tree measure in (3.7), $D_j(l_j) r^{-l_j} = \mathbb{P}_0\left(L_j = l_j\right) = \mathbb{P}_0\left(V_j \in \tau(l_j)\right)$. Therefore,

$$\mathbb{P}_0\left(\sum_{j=1}^{n} \Lambda^{-1}\left(L_j\Delta + W_j\right) > b\right)$$

$$
\begin{aligned}
&= \sum_{j=1}^{n}\sum_{l_j=0}^{m_j}\mathbb{E}_0\left[\mathbb{P}_0\big(V_j\in\tau\left(l_j\right)\big)\mathbb{E}_0\left(I\left(\sum_{j=1}^{n}\Lambda^{-1}\left(V_j\right)>b\right)\middle|V_j\in\tau(l_j)\right)\right]\\
&= \sum_{j=1}^{n}\sum_{l_j=0}^{m_j}\mathbb{P}\left(\sum_{j=1}^{n}\Lambda^{-1}\left(V_j\right)>b; V_j\in\tau\left(l_j\right), j=1,\ldots,n\right)\\
&= \mathbb{P}\left(\sum_{j=1}^{n}X_j>b\right).
\end{aligned}
$$

Unbiasedness follows.

2) *Efficiency.*

Note that, given $\epsilon>0$,

$$
\begin{aligned}
&\sum_{j=1}^{n}\Lambda^{-1}\left(V_j\right)>b\\
\Longrightarrow\quad &\sum_{j=1}^{n}\Lambda\left(\Lambda^{-1}\left(V_j\right)\right)=\sum_{j=1}^{n}V_j\geq\Lambda\left(\sum_{j=1}^{n}\Lambda^{-1}\left(V_j\right)\right)-\epsilon>\Lambda(b)-\epsilon,
\end{aligned}
$$

which is a direct consequence of Lemma 3.2. Therefore,

$$
\begin{aligned}
\widetilde{\mathbb{E}}\left[R_\gamma^2(b)\right] &= \widetilde{\mathbb{E}}\left[I\left(\sum_{j=1}^{n}\Lambda^{-1}\left(V_j\right)>b\right)\prod_{j=1}^{n}\left(e^{-L_j\Delta}N_j\left(\gamma\right)\right)^2\right]\\
&\leq \widetilde{\mathbb{E}}\left[I\left(\sum_{j=1}^{n}V_j>\Lambda(b)-\epsilon\right)\prod_{j=1}^{n}\left(e^{-L_j\Delta}N_j\left(\gamma\right)\right)^2\right]\\
&\leq \widetilde{\mathbb{E}}\left[I\left(\sum_{j=1}^{n}L_j\Delta>\Lambda(b)-n\Delta-\epsilon\right)\prod_{j=1}^{n}\left(e^{-L_j\Delta}N_j\left(\gamma\right)\right)^2\right]\\
&\leq \exp\big(-2\left(\Lambda(b)-n\Delta-\epsilon\right)\big)\mathbb{E}\left[N_1\left(\gamma\right)\right]^n\\
&= K\exp\big(-2\left(\Lambda(b)-\epsilon\right)\big)\mathbb{E}\left[N_1\left(\gamma\right)\right]^n,
\end{aligned}
$$

where we can change from $\widetilde{\mathbb{E}}$ to $\mathbb{E}$ in the last inequality because the quantity $N_j\left(\gamma\right)$ is independent of the sampling of the level L_j's. Combining this with Lemma 3.1, which

says that $\mathbb{E}\left[N_j(\gamma)^2\right] = o\left[\log^2 q(b)\right]$, we obtain, for any $\epsilon' > 0$,

$$\widetilde{\mathbb{E}}\left[R_\gamma^2(b)\right] = O\left[e^{-(2-\epsilon')\Lambda(b)}\right] = O\left[q(b)^{2-\epsilon'}\right],$$

as $b \nearrow \infty$. Logarithmic efficiency follows. □

Interestingly, $\theta = \gamma = \exp(\xi\Delta)$ turns out to be *the only* choice of parameter that leads to logarithmic efficiency in the parametric family of estimators $\{R_\theta(b)\}_{1\le\theta\le r}$. (Recall that $\xi > 0$ is pre-determined to enforce a super-critical tree constructed using the procedure introduced in Subsection 3.3.1.) The intuition is that, when $\theta < \gamma$, the likelihood ratio $\prod_{j=1}^n \exp(-L_j\Delta) N_j(\gamma)$ grows too fast. On the other hand, when $\theta > \gamma$, the θ-sampling measure $\widetilde{\mathbb{P}}_\theta$ doesn't give sufficiently large weight to higher levels in the tree to substantially improve over the full-tree sampling measure $\mathbb{P}_0$. We close this section with the following theorem on the optimal choice of θ, which makes the preceding intuitions precise.

Theorem 3.3. *The HFSR estimator $R_\theta(b)$ achieves logarithmic efficiency if and only if $\theta = \gamma = \exp(\xi\Delta)$.*

Proof. Again let $\widetilde{\mathbb{E}}_\theta$ be the expectation taken under $\widetilde{\mathbb{P}}_\theta$ defined in (3.7), and let $V_j = L_j\Delta + W_j$, for $j = 1, \ldots, n$. Note that the second moment of the estimator can be expressed in the following way:

$$\begin{aligned}
\widetilde{\mathbb{E}}_\theta\left[R_\theta^2(b)\right] &= \widetilde{\mathbb{E}}_\gamma\left[I\left(\sum_{j=1}^n \Lambda^{-1}(V_j) > b\right)\prod_{j=1}^n\left(\frac{r}{\gamma}\right)^{-2L_j} N_j(\gamma)^2 \prod_{j=1}^n\left(\frac{\gamma}{\theta}\right)^{-L_j}\frac{N_j(\theta)}{N_j(\gamma)}\right] \\
&= \widetilde{\mathbb{E}}_\gamma\left[I\left(\sum_{j=1}^n \Lambda^{-1}(V_j) > b\right)\prod_{j=1}^n\left(\frac{r}{\gamma}\right)^{-2L_j}\left(\frac{\gamma}{\theta}\right)^{-L_j} N_j(\theta)N_j(\gamma)\right].
\end{aligned}$$

Our strategy is to find $\eta > 0$ such that

$$\liminf_{b\to\infty} \widetilde{\mathbb{E}}_\theta\left(R_\theta^2(b)\right)/q(b)^{2-\eta} = \infty, \tag{3.13}$$

when $\theta \neq \exp(\xi\Delta)$. We separately treat the case $\theta < \gamma$ and $\theta > \gamma$.

1) $(1 \leq \theta < \gamma)$.

Note that

$$\{L_1 = m\} \subseteq \left\{ \sum_{j=1}^{n} \Lambda^{-1}(V_j) > b \right\}$$

Therefore, starting from (3.13), and taking advantage of the independence among the trees, we obtain

$$\begin{aligned} & \widetilde{\mathbb{E}}_\theta \left[R_\theta^2(b) \right] \qquad (3.14) \\ \geq \quad & \widetilde{\mathbb{E}}_\gamma \left[I(L_1 = m) \prod_{j=1}^{n} \left(\frac{r}{\gamma} \right)^{-2L_j} \left(\frac{\gamma}{\theta} \right)^{-L_j} N_j(\theta) N_j(\gamma) \right] \\ = \quad & \widetilde{\mathbb{E}}_\gamma \left[I(L_1 = m) \left(\frac{r}{\gamma} \right)^{-2m} \left(\frac{\gamma}{\theta} \right)^{-m} N_j(\theta) N_j(\gamma) \right] \\ & \cdot \widetilde{\mathbb{E}}_\gamma \left[\left(\frac{r}{\gamma} \right)^{-2L} \left(\frac{\gamma}{\theta} \right)^{-L} N_1(\theta) N_1(\gamma) \right]^{n-1}. \end{aligned}$$

The first expectation term in (3.14) can be further evaluated as

$$\begin{aligned} & \widetilde{\mathbb{E}}_\gamma \left[I(L_1 = m) \left(\frac{r}{\gamma} \right)^{-2m} \left(\frac{\gamma}{\theta} \right)^{-m} N_j(\theta) N_j(\gamma) \right] \\ = \quad & \mathbb{E} \left[\sum_{l=0}^{m} \left(\frac{r}{\gamma} \right)^{-2m} N_1(\theta) \gamma^{-m} D(m) \right] \\ = \quad & r^{-2m} \theta^m \sum_{l=0}^{m} \sum_{k=0}^{m} \theta^{-k} \mathbb{E}\big[D(k) D(m) \big] \\ = \quad & r^{-2m} \theta^m \sum_{l=0}^{m} \sum_{k=0}^{m} \theta^{-k} \Theta\big(\gamma^{k+l} \big). \end{aligned}$$

The last equality in the previous display follows because $\mathbb{E}\big[D(k)D(l)\big] = \Theta\Big(\mathbb{E}\big[Z(k)Z(l)\big]\Big) =$

$\Theta\big(\gamma^{k+l}\big)$, as shown in the proof of Lemma 3.1. We can therefore conclude that

$$\begin{aligned} & \widetilde{\mathbb{E}}_\gamma \left[I\left(L_1 = m\right) \left(\frac{r}{\gamma}\right)^{-2m} \left(\frac{\gamma}{\theta}\right)^{-m} N_j\left(\theta\right) N_j\left(\gamma\right) \right] \\ = \quad & \Omega\left[\left(\frac{r}{\gamma}\right)^{2m} \right] = \Omega\big[\exp\big(-2\Lambda(b)\big)\big]. \end{aligned} \tag{3.15}$$

On the other hand, a lower bound for the second expectation term in (3.14) can be obtained in a similar fashion:

$$\begin{aligned} & \widetilde{\mathbb{E}}_\gamma \left[\left(\frac{r}{\gamma}\right)^{-2L_1} \left(\frac{\gamma}{\theta}\right)^{-L_1} N_1\left(\theta\right) N_1\left(\gamma\right) \right] \\ = \quad & \mathbb{E}\left[\sum_{l=0}^{m} \gamma^{-l} D(l) \left(\frac{r}{\gamma}\right)^{-2l} \left(\frac{\gamma}{\theta}\right)^{-l} N_1\left(\theta\right) \right] \\ = \quad & \sum_{l=0}^{m} \sum_{k=0}^{m} r^{-2l} \theta^l \theta^{-k} \Theta\big(\gamma^{k+l}\big) \\ = \quad & \Theta\left[\sum_{l=0}^{m} \left(\frac{\gamma\theta}{r^2}\right)^l \sum_{k=0}^{m} \left(\frac{\gamma}{\theta}\right)^k \right] = \Omega\left[\left(\frac{\gamma}{\theta}\right)^m \right]. \end{aligned} \tag{3.16}$$

Combining (3.15) and (3.16), we have

$$\widetilde{\mathbb{E}}_\theta \left[R_\theta^2(b) \right] = \Omega\left[\exp\big(-2\Lambda(b)\big) \left(\gamma/\theta\right)^{m(n-1)} \right]. \tag{3.17}$$

Note that, by virtue of (1.1), we have $q(b) = \Theta\left(\exp\left(-\Lambda(b)\right)\right)$. Now, let us write $\theta = \exp\left((\xi - \varepsilon)\Delta\right)$, where ε is some constant satisfying $0 < \varepsilon \leq \xi$. Consequently, if we choose $0 < \eta < \varepsilon(n-1)$, equation (3.13) holds and therefore $R_\theta(b)$ fails to have logarithmic efficiency when $\theta < \gamma$.

2) $(\gamma < \theta \leq r)$.

Observe that $N_j\left(\theta\right) \geq 1$, the expectations in (3.14) therefore bear the following lower

bounds:

$$\begin{aligned} & \widetilde{\mathbb{E}}_\gamma \left[I\left(L_1 = m\right) \left(\frac{r}{\gamma}\right)^{-2m} \left(\frac{\gamma}{\theta}\right)^{-m} N_j\left(\theta\right) N_j\left(\gamma\right) \right] \\ \geq \quad & \left(\frac{r}{\gamma}\right)^{-2m} \left(\frac{\theta}{\gamma}\right)^{m} \widetilde{\mathbb{E}}\Big[I\left(L_1 = m\right) N_1(\gamma) \Big] \\ = \quad & \left(m+1\right) \exp\big(-2\Lambda(b)\big) \left(\frac{\theta}{\gamma}\right)^{m}. \end{aligned} \tag{3.18}$$

Here we used $\widetilde{\mathbb{E}}\left[I\left(L_1 = m\right) N_1(\gamma)\right] = \sum_{l=0}^{m} \gamma^{-m} \mathbb{E}\big[D(m)\big] = m+1$. Meanwhile, from the derivation in (3.16),

$$\widetilde{\mathbb{E}}_\gamma \left[\left(\frac{r}{\gamma}\right)^{-2L_1} \left(\frac{\gamma}{\theta}\right)^{-L_1} N_1\left(\theta\right) N_1\left(\gamma\right) \right] = \Omega\left[\sum_{l=0}^{m} \left(\frac{\gamma\theta}{r^2}\right)^{l} \sum_{k=0}^{m} \left(\frac{\gamma}{\theta}\right)^{k} \right] = \Omega(1). \tag{3.19}$$

We therefore conclude, as a result of (3.18) and (3.19),

$$\widetilde{\mathbb{E}}_\theta \left[H_\theta^2(b) \right] = \Omega\left[\left(m+1\right) \exp\big(-2\Lambda(b)\big) \left(\theta/\gamma\right)^m \right].$$

The same procedure as in the case $1 \leq \theta < \gamma$ can now be performed and we are done. □

3.6 An Improved Hazard Function Splitting Algorithm

Although the Splitting-Resampling (HFSR) algorithm studied so far is proved to be logarithmically efficient, there is potential room for improvement. Note from the description of the previous algorithm that, it takes some effort to construct a tree that is not too sparse (in the sense that the probability of having at least one particle/leaf at the top of the tree (see Figure 3.1) is bounded away from zero). However, for such trees, if the

leaf at the top is not sampled according to the "optimal" level sampling measure $\widetilde{\mathbb{P}}_\gamma(\cdot)$, much of the effort in the tree construction phase is wasted. In this section we propose an alternative splitting strategy in which we take the previous observation into account.

3.6.1 The "Mega" Splitting Algorithm

Recall that in the HFSR algorithm, we propagate and construct independent trees separately for each random variable X_j. The basic idea behind this alternative algorithm is to utilize *every* particle/leaf that has already been simulated. In order to do this, each time we have completed the construction of a tree, instead of re-sampling from the tree, we superimpose and grow a new tree at the position of each leaf of the preceding tree, thereby creating a "mega tree" for the random sum $S_n = \sum_{j=1}^n X_j$. Since every particle is fully utilized in the construction of the mega tree, we can in fact broaden the choices of r to include the case $r = \exp(\Delta)$, i.e., we allow the case when the resulting mega tree is critical. As usual, we need to endow each particle with a weight and keep diluting the weight when splitting occurs. In particular, starting from a weight equal to one, whenever a split occurs during the propagation phase, each offspring particle is endowed with a weight equal to the weight of its parent, multiplied by $1/r$.

To be more precise, our construction of the Mega-tree is sequential and it proceeds as follows. First we construct $\widetilde{\Pi}_1 = \Pi_1$, i.e., $\widetilde{\Pi}_1$ is identical to Π_1 in the HFSR algorithm described in previous sections. We call this *the first growth step*, and define $\mathcal{L}\left(\widetilde{\Pi}_1\right)$ the set of leafs on top of $\widetilde{\Pi}_1$. Then, for each leaf $s \in \mathcal{L}\left(\widetilde{\Pi}_1\right)$, we construct a subtree $\Pi(s) \stackrel{d}{=} \Pi_1$. In other words, the subtree $\Pi(s)$ is constructed in the same way as Π_1, but instead of rooted at zero, it is rooted at s. Let us call the constructions of the trees $\{\Pi(s)\}_{s\in\mathcal{L}\left(\widetilde{\Pi}_1\right)}$ *the second growth step*. Define the Mega-tree constructed at the end of the second growth

step to be $\widetilde{\Pi}_2$, and define the set of leafs on top of $\widetilde{\Pi}_2$ to be $\mathcal{L}\left(\widetilde{\Pi}_2\right)$. The j-th growth step, along with $\widetilde{\Pi}_j$ and $\mathcal{L}\left(\widetilde{\Pi}_j\right)$, for $j = 3, \ldots, n$, are similarly defined as in the second growth step. Therefore, at the end of the n-th growth step, the Mega-tree $\widetilde{\Pi}_n$ is in place.

At the time of each split, *each offspring particle generated inherits the same path along the Mega-tree of its "parent" particle, up to the point of splitting, and evolve independently thereafter.* Note that, for each $s \in \widetilde{\Pi}_j$, $1 \leq j \leq n$, we are able to extract the "stem information" carried by s, defined via

$$H_j(s) = \Big(w(s,1), w(s,2), \ldots, w(s,j)\Big)^T, \ s \in \mathcal{L}\left(\widetilde{\Pi}_j\right), \tag{3.20}$$

where $w(s,j) = s$, and $w(s,i)$, is the root of the $(i+1)$-st subtree, $1 \leq i \leq j-1$. In other words, $H_j(s)$ records all the roots of the $j-1$ subtrees that s belongs to, as well as s itself. Furthermore, let us define $0 \leq L\left(w(s,i)\right) \leq m$, the level attained by $w(s,i)$ in the i-th subtree, $\Pi\left(w(s,i)\right)$, $1 \leq i \leq j$. Define

$$\mathbb{L}\left(H_j(s)\right) = \left(L\left(w(s,1)\right), \ldots, L\left(w(s,j)\right)\right). \tag{3.21}$$

Note that each leaf $s \in \mathcal{L}\left(\widetilde{\Pi}_j\right)$ carries a cumulative weight equal to $r^{-\sum_{i=1}^{j} L(w(s,i))}$. Finally, define the sampled random sum associated with leaf s in the final Mega-tree, $\widetilde{\Pi}_n$, via

$$\widehat{S}_n(s) = \psi\left(H_n(s)\right) \stackrel{\Delta}{=} \sum_{i=1}^{n} \Lambda^{-1}\Big(L\left(w(s,i)\right)\Delta + W\big(L\left(w(s,i)\right)\big)\Big), \ s \in \mathcal{L}\left(\widetilde{\Pi}_n\right), \tag{3.22}$$

where $W(L)$ is defined in (3.9). The "Mega"-Splitting algorithm can therefore be performed in the following steps:

The "Mega" Hazard Function Splitting (MHFS) Algorithm

1) $j = 1$. Construct $\widetilde{\Pi}_1$.

2) For $1 \leq j \leq n-1$, obtain $\widetilde{\Pi}_{j+1}$ by constructing $\Pi(s)$, for each $s \in \mathcal{L}\left(\widetilde{\Pi}_j\right)$.

3) The final MHFS estimator for the tail probability $q(b) = \mathbb{P}\left(S_n > b\right)$ is therefore

$$Z\left(\widetilde{\Pi}_n\right) = \sum_{s \in \mathcal{L}\left(\widetilde{\Pi}_n\right)} I\left(\psi\left(H_n(s)\right) > b\right) r^{-\sum_{j=1}^{n} L(w(s,j))}. \tag{3.23}$$

Similar to the HFSR estimator, we shall measure the cost per replication of the previous MHFS estimator by the expected total number of leafs generated in a single Mega-tree, which says

$$\widehat{\mathcal{W}}(b) = O\left[\mathbb{E}\left(\left|\mathcal{L}\left(\widetilde{\Pi}_n\right)\right|\right)\right]. \tag{3.24}$$

A similar "fully branching" representation for the MHFS algorithm can be defined as follows. In the first growth step construct a tree identical to $\widetilde{\Pi}_1$. Then, each $s \in \mathcal{L}\left(\widetilde{\Pi}_1\right)$ is replaced by a cluster, $\mathcal{K}(s)$, of $r^{m-L(s)}$ of identical leafs, thereby obtaining a tree denoted by $\widetilde{\Pi}'_1$. Note that the clusters form a partition of $\mathcal{L}\left(\widetilde{\Pi}'_1\right)$. The set $\mathcal{L}\left(\widetilde{\Pi}'_1\right)$ of leafs at the top of $\widetilde{\Pi}'_1$ is of size r^m and each leaf is attached a weight equal to r^{-m}. This concludes the first growth step of the fully branching Mega-tree. The second growth step proceeds as follows. For each $s \in \mathcal{L}\left(\widetilde{\Pi}'_1\right)$ construct a subtree $\widetilde{\Pi}'_1(s)$ with distribution $\widetilde{\Pi}'_1$, rooted at s instead of at zero. The leafs of $\widetilde{\Pi}'(s)$ are partitioned into clusters as indicated earlier for $\widetilde{\Pi}'_1$. All of these subtrees are independent. We obtain a tree which we denote as $\widetilde{\Pi}'_2$, which has r^{2m} leafs at its top. And the clusters form a partition of $\mathcal{L}\left(\widetilde{N}'_2\right)$. Each leaf is attached with a weight equal to r^{-2m}. This concludes the second growth step of the fully branching tree.

In this way, at the j-th growth step, $j = 2, \ldots, n$, $\widetilde{\Pi}'_j$ is obtained recursively by con-

structing, independently, subtrees $\widetilde{\Pi}'_1(s)$ for each $s \in \mathcal{L}\left(\widetilde{\Pi}'_{j-1}\right)$, partitioning $\mathcal{L}\left(\widetilde{\Pi}'_1(s)\right)$ into clusters as indicated earlier. The Mega-tree $\widetilde{\Pi}'_j$ has r^{jm} leafs at its top, and each leaf is attached a weight equal to r^{-jm}. The particles and weights of our fully Mega-splitting procedure are in one-to-one correspondence with the leafs of the tree $\widetilde{\Pi}'_n$ and their corresponding weights. Consequently we arrive at the following MHFS estimator for the fully branching representation:

$$Z\left(\widetilde{\Pi}'_n\right) = \sum_{s=1}^{r^{n\times m}} I\left(\psi\left(H_n(s)\right) > b\right) r^{-n\times m} \stackrel{d}{=} Z\left(\widetilde{\Pi}_n\right), \tag{3.25}$$

where $\psi(\cdot)$ is defined in (3.22). Note that it is obviously inefficient from an implementation perspective to construct subtrees and hence the Mega-tree using the fully branching method, but the representation turns out to be particularly convenient in the analysis of the second moment of the estimator $Z\left(\widetilde{\Pi}_n\right)$. The benefit lies in the fact that, weight assignment and trajectory propagation can be treated as independent procedures in a fully branching tree. Since $Z\left(\widetilde{\Pi}'_n\right) \stackrel{d}{=} Z\left(\widetilde{\Pi}_n\right)$, we shall therefore consider $Z\left(\widetilde{\Pi}'_n\right)$ in our ensuing analysis of the algorithm.

3.6.2 Analysis of the Mega-Splitting Algorithm

Let us first simplify notation and define

$$\mathbb{1}_s(b) = I\left(\psi\left(H_n(s)\right) > b\right),$$

for $1 \le s \le \mathcal{L}\left(\widetilde{\Pi}'_n\right)$. In words, $\mathbb{1}_s(b)$ is equal to one if the s-th particle ends up with a position in the hazard function space that, when transformed back into the original space, leads to a sum that is larger than b; and it is equal to zero otherwise. It's not surprising that the MHFS algorithm is at least as efficient as the HFSR algorithm. The following

result summarizes the performance of the Mega-Splitting Algorithm.

Theorem 3.4. *Let $r = \exp\big((1+\xi)\,\Delta\big)$ be the number of offspring particles per splitting, where $\xi > 0$ is the criticality parameter, and Δ is the level size in the hazard function space, both pre-chosen by the user. Then the MHFS estimator,*

$$Z(\Pi) = \sum_{s=1}^{r^{n\times m}} \mathbb{1}_s(b) r^{-n\times m} \overset{D}{=} Z(\Pi') = \sum_{s\in\mathcal{L}\left(\widetilde{\Pi}_n\right)} \mathbb{1}_s(b) r^{-\sum_{j=1}^{n} L(w(s,j))},$$

is logarithmically efficient for estimating $q(b) = \mathbb{P}(S_n > b)$.

To prove the result, we shall take advantage of a technique used in [31] that *genealogically categorizes* different particles according to their *last common roots*, which is formally defined as follows.

Definition 3.1. *Let $\mathcal{D}_n(s) \subseteq \mathcal{L}\left(\widetilde{\Pi}'_n\right)$ denote the set of the offspring leafs of s at the top of $\widetilde{\Pi}'_n$. Let $d_v \in \mathcal{D}_n(v_{k+1}), d_w \in \mathcal{D}_n(w_{k+1})$, where $v_{k+1}, w_{k+1} \in \widetilde{\Pi}'(s_k)$, for some $1 \le k \le n-1$. Then s_k is called the* ***last common root*** *for d_v and d_w if*

$$\mathcal{K}(v_{k+1}) \neq \mathcal{K}(w_{k+1}),$$

where $\mathcal{K}(s)$ is the cluster that leaf s belongs to.

Proof of Theorem 3.4. First it's not hard to see that

$$\widehat{\mathcal{W}}(b) = O\left[\mathbb{E}\left(\left|\mathcal{L}\left(\widetilde{\Pi}_n\right)\right|\right)\right] = O\left[\mathbb{E}\left(\sum_{k=0}^{m} D(k)\right)^n\right],$$

where $D(k)$'s are defined in (3.6). Therefore, applying Lemma 3.1, we have

$$\widehat{\mathcal{W}}(b) = \Theta\left[(-\log q(b))^n\right] = o\left[1/q(b)^\epsilon\right], \tag{3.26}$$

for any $\epsilon > 0$.

Using the fully branching representation, the second moment of the estimator $Z\left(\widetilde{\Pi}_n\right)$ can be written as

$$\begin{aligned}
&\mathbb{E}\left[\left(\sum_{s=1}^{r^{nm}} \mathbb{1}_s r^{-nm}\right)^2\right] \\
&= \mathbb{E}\left[\sum_{s\in\mathcal{L}\left(\widetilde{\Pi}'_n\right)} \mathbb{1}_s r^{-2nm}\right] + \mathbb{E}\left[\sum_{v,w\in\mathcal{L}\left(\widetilde{\Pi}'_n\right), v\neq w} \mathbb{1}_v \mathbb{1}_w r^{-2nm}\right] \\
&= \mathbb{E}\left[\sum_{s=1}^{r^{nm}} \mathbb{1}_s r^{-2nm}\right] + \sum_{j=1}^{n} \mathbb{E}\left[\sum_{l^{(j)}\in\mathcal{H}_j} \sum_{s^{(j)}\in\mathcal{L}\left(\widetilde{\Pi}'_j\right)} r^{-2jm} I\left(\mathbb{L}\left(H_j(s^{(j)})\right) = l^{(j)}\right)\right. \qquad (3.27) \\
&\left. \cdot \sum_{\substack{v_{j+1},w_{j+1}\in\widetilde{\Pi}'\left(s^{(j)}\right) \\ \mathcal{K}(v_{j+1})\neq\mathcal{K}(w_{j+1})}} \left(\sum_{d_v\in\mathcal{D}_n(v_{j+1})} \frac{1}{r^m} r^{-(n-j-1)m} \mathbb{1}_{d_v}\right)\left(\sum_{d_w\in\mathcal{D}_n(w_{j+1})} \frac{1}{r^m} r^{-(n-j-1)m} \mathbb{1}_{d_w}\right)\right].
\end{aligned}$$

Here the second equality holds because we have decomposed pairs of *different* leafs in $\mathcal{L}\left(\widetilde{\Pi}'_n\right)$ into *disjoint* sets, according to their *last common ancestor root in the final Mega-tree*, see Definition 3.1. In particular, $s^{(j)}$ is the last common root for the pair of leafs, $(d_v, d_w) \in \mathcal{L}\left(\widetilde{\Pi}'_n\right)$.

Now let $\mathcal{F}_j = \sigma\left(\widetilde{\Pi}'_1, \ldots, \widetilde{\Pi}'_j\right)$ denote the sigma algebra generated by the random variables used to yield all the Mega-trees up to $\widetilde{\Pi}'_j$. For the expectation term in the summand in (3.28), we can condition on $\mathcal{F}_j$ and obtain

$$\mathbb{E}\left[\sum_{l^{(j)}\in\mathcal{H}_j} \sum_{s^{(j)}\in\mathcal{L}\left(\widetilde{\Pi}'_j\right)} r^{-2jm} I\left(\mathbb{L}\left(H_j(s^{(j)})\right) = l^{(j)}\right) \qquad (3.28)$$

$$\cdot \sum_{\substack{v_{j+1},w_{j+1}\in\widetilde{\Pi}'\left(s^{(j)}\right)\\ \mathcal{K}(v_{j+1})\neq\mathcal{K}(w_{j+1})}} \left(\sum_{d_v\in\mathcal{D}_n(v_{j+1})} \frac{1}{r^m} r^{-(n-j-1)m}\mathbb{1}_{d_v}\right)\left(\sum_{d_w\in\mathcal{D}_n(w_{j+1})} \frac{1}{r^m} r^{-(n-j-1)m}\mathbb{1}_{d_w}\right)\Bigg]$$

$$= \mathbb{E}\Bigg[\sum_{l^{(j)}\in\mathcal{H}_j}\sum_{s^{(j)}\in\mathcal{L}\left(\widetilde{\Pi}'_j\right)} r^{-2jm} I\left(\mathbb{L}\left(H_j(s^{(j)})\right)=l^{(j)}\right)$$

$$\cdot \sum_{\substack{v_{j+1},w_{j+1}\in\widetilde{\Pi}'\left(s^{(j)}\right)\\ \mathcal{K}(v_{j+1})\neq\mathcal{K}(w_{j+1})}} \left(r^{-m}\mathbb{E}\left[\sum_{t\in d_{k_{j+1}}} r^{-(n-j-1)m}\mathbb{1}_t\middle|\mathcal{F}_j\right]\right)^2\Bigg].$$

Define $\tau(l)$ as we did in Subsection 3.3.1. Using the property of the fully branching presentation, which says that the weight and trajectory can be viewed as independent objects, we have

$$\begin{aligned} q_{j,l^{(j)}}(b) &\triangleq \mathbb{E}\left[\sum_{t\in d_{k_{j+1}}} r^{-(n-j-1)m}\mathbb{1}_t\middle|\mathcal{F}_j\right] = \mathbb{P}\left(\sum_{j=1}^n X_j > b\middle|\mathcal{F}_j\right) \\ &= \mathbb{P}\left(\sum_{j=1}^n X_j > b\middle|\Lambda\left(X_h\right)\in\tau(l_h),\forall h\le j\right). \end{aligned}$$

Therefore, (3.28) can be expressed as

$$\mathcal{M}\mathbb{E}\left[\sum_{l^{(j)}\in\mathcal{H}_j}\sum_{s^{(j)}\in\mathcal{L}\left(\widetilde{\Pi}'_j\right)} r^{-2jm} I\left(\mathbb{L}\left(H_j(s^{(j)})\right)=l^{(j)}\right)\left[q_{j,l^{(j)}}(b)\right]^2\right], \tag{3.29}$$

where

$$\mathcal{M}\triangleq \sum_{v_{j+1},w_{j+1}\in\widetilde{\Pi}'\left(s^{(j)}\right),\mathcal{K}(v_{j+1})\neq\mathcal{K}(w_{j+1})} r^{-2m} = 1-r^{-m}.$$

Now, depending on the value of β, our strategy is to appropriately decompose the set $\{\mathbb{L}\left(H_j(s^{(j)})\right)=l^{(j)}\}$. We separate the development into two cases.

1) $\beta = 0$**.**

Note that $\Lambda(b) - \Lambda(b/n) \leq \Delta$ when b is sufficiently large. And recall that $m = \lceil \Lambda(b)/\Delta \rceil$. Therefore, we have, for b large enough, $\Lambda^{-1}\left((m-k)\Delta\right) < b/n$, for all $2 \leq k \leq m$. And hence $X_i \leq b/n$, for all $1 \leq i \leq j$. As a result, for $1 \leq j \leq n-1$, we have $q_{j,l^{(j)}}(b) \leq \mathbb{P}\left(\sum_{h=j+1}^{n} X_h > (1-j/n)b\right)$, and $q_{n,l^{(n)}}(b) = 0$. Moreover, from the property of regularly varying distributions, we know that

$$\mathbb{P}\left(\sum_{h=j+1}^{n} X_h > (1-j/n)b\right) = \Theta\left[q(b)\right].$$

We therefore conclude that

$$\begin{aligned}
&\mathcal{M}\mathbb{E}\Bigg[\sum_{l^{(j)}\in\mathcal{H}_j}\sum_{s^{(j)}\in\mathcal{L}\left(\tilde{\Pi}'_j\right)} r^{-2jm} I\left(\mathbb{L}\left(H_j(s^{(j)})\right) = l^{(j)}\right) \\
&\qquad \cdot I\left(L\left(w(s,i)\right) \leq m-2, \forall i \leq j\right)\left[q_{j,l^{(j)}}(b)\right]^2\Bigg] \\
&= \sum_{l^{(j)}\in\mathcal{H}_j}\sum_{s^{(j)}\in\mathcal{L}\left(\tilde{\Pi}'_j\right)} r^{-2jm}\mathbb{P}\left(\mathbb{L}\left(H_j(s^{(j)})\right) = l^{(j)}; L\left(w(s,i)\right) \leq m-2, \forall i \leq j\right)\left[q_{j,l^{(j)}}(b)\right]^2 \\
&\leq K_1 \prod_{i=1}^{j}\left(\sum_{l_i=0}^{m-2}\sum_{s_i=1}^{r^m} r^{-2m} e^{-l_i\Delta}\right) q(b)^2 \\
&\leq K_1\left[\frac{r^{-m}}{1-\exp(-\Delta)}\right]^j q(b)^2 = o\left[q(b)^2\right], \qquad (3.30)
\end{aligned}$$

where K_1 is a positive constant depending only n and Δ. Here we have used

$$\mathbb{P}\left(\mathbb{L}\left(H_j(s^{(j)})\right) = l^{(j)}\right) = \prod_{i=1}^{j}\mathbb{P}\left(L\left(w(s,i)\right) = l_i\right) \leq e^{-\sum_{i\leq j} l_i\Delta}.$$

On the other hand, for some positive constant K_2 that depends only on Δ, we have

$$\begin{aligned}
&\sum_{l^{(j)}\in\mathcal{H}_j}\sum_{s^{(j)}\in\mathcal{L}\left(\widetilde{\Pi}'_j\right)} r^{-2jm}\mathbb{P}\left(\mathbb{L}\left(H_j(s^{(j)})\right)=l^{(j)};\right.\\
&\qquad \left. L\left(w(s,i)\right)>m-2,\ \text{for some } i\le j\right)\left[q_{j,l^{(j)}}(b)\right]^2\\
&\le \sum_{i=1}^{j}\left(\sum_{l_i=m-1}^{m}\sum_{s_i=1}^{r^m} r^{-2m}r^{-l_i}\right)\le K_2 r^{-2m}=O\left[q(b)^2\right], \qquad (3.31)
\end{aligned}$$

where we have replaced $q_{j,l^{(j)}}(b)$ with one. The last equality holds because

$$r^{-m}=\exp\left(-(1+\xi)m\Delta\right)=q(b)^{1+\xi}\ge q(b).$$

Consequently, recognizing that $\mathbb{E}\left[\sum_{s=1}^{r^{mn}}\mathbb{1}_s r^{-2nm}\right]=q(b)^2$, we conclude by combining (3.30) and (3.31) with (3.28) that

$$\mathbb{E}\left[Z\left(\widetilde{\Pi}_n\right)^2\right]=\mathbb{E}\left[Z\left(\widetilde{\Pi}'_n\right)^2\right]=\mathbb{E}\left[\left(\sum_{s=1}^{r^{mn}}\mathbb{1}_s r^{-nm}\right)^2\right]=O\left[q(b)^2\right].$$

2) $0<\beta<1$**.**

Given $\delta>0$, let $\kappa_\delta(b)$ be defined via

$$\kappa_\delta(b)=\lfloor\left(\Lambda\left(b\right)-\delta\right)/\Delta\rfloor.$$

Note that

$$\begin{aligned}
\mathbb{E}\Bigg[&\sum_{l^{(j)}\in\mathcal{H}_j}\sum_{s^{(j)}\in\mathcal{L}\left(\widetilde{\Pi}'_j\right)} r^{-2jm}I\left(\mathbb{L}\left(H_j(s^{(j)})\right)=l^{(j)}\right)\\
&\cdot I\left(L\left(w(s,i)\right)\le\kappa_\delta(b),\forall i\le j\right)\left[q_{j,l^{(j)}}(b)\right]^2\Bigg]
\end{aligned}$$

$$
\begin{aligned}
&\leq \sum_{l^{(j)}\in\mathcal{H}_j}\sum_{s^{(j)}\in\mathcal{L}\left(\widetilde{\Pi}'_j\right)} r^{-2jm}\frac{\mathbb{P}\left(\sum_{i=1}^n X_i > b; \Lambda\left(X_h\right)\in\tau\left(l_h\right), \Lambda\left(X_h\right)\leq\kappa_\delta(b)\Delta, \forall h\leq j\right)^2}{\mathbb{P}\left(\Lambda\left(X_h\right)\in\tau\left(l_h\right), \Lambda\left(X_h\right)\leq\kappa_\delta(b)\Delta, \forall h\leq j\right)} \\
&\leq \sum_{l^{(j)}\in\mathcal{H}_j}\sum_{s^{(j)}\in\mathcal{L}\left(\widetilde{\Pi}'_j\right)} r^{-2jm}\frac{\mathbb{E}\left[\exp\left(-\rho\sum_{h\leq j}\Lambda(X_h)\right) I\left(\sum_{i=1}^n X_i > b\right)\right]^2 \exp\left(2j\rho\kappa_\delta(b)\Delta\right)}{\prod_{h\leq j}\left[\exp\left(-l_h\Delta\right)\cdot\min\left(1,\Delta\exp\left(-\Delta\right)\right)\right]} \\
&\leq \exp\left(2j\rho\kappa_\delta(b)\Delta\right)\prod_{h\leq j}\left[\sum_{l_h=0}^{\kappa_\delta(b)}\sum_{s_h=1}^{r^m} r^{-2m}e^{l_h\Delta}\right] q(b)^2 \\
&= K_3\left[\exp\left(\left(1+2\rho\right)\kappa_\delta(b)\Delta\right)r^{-m}\right]^j q(b)^2, \qquad (3.32)
\end{aligned}
$$

where $\rho > 0$ to be chosen momentarily, K_3 is some positive constant independent of b, and the second inequality holds by virtue of Chebychev's inequality and

$$\mathbb{P}\left(\Lambda(X_h)\in\tau(l_h)\right)\geq\min\left(\exp\left(-l_h\Delta\right), \Delta\exp\left(-(l_h+1)\Delta\right)\right).$$

Since $r = \exp\left((1+\xi)\Delta\right)$, it suffices to choose ρ so that

$$\log\left[\exp\left(\left(1+2\rho\right)\kappa_\delta(b)\Delta\right)r^{-m}\right] = \left(1+2\rho\right)\kappa_\delta(b)\Delta - (1+\xi)\Lambda(b)\leq 0.$$

Note that $\kappa_\delta(b)\Delta\leq\Lambda(b)-\delta$. We can therefore simply pick $0<\rho\leq\xi/2$, so that the expression in (3.32) is $O\left[q(b)^2\right]$.

On the other hand,

$$
\begin{aligned}
&\sum_{l^{(j)}\in\mathcal{H}_j}\sum_{s^{(j)}\in\mathcal{L}\left(\widetilde{\Pi}'_j\right)} r^{-2jm}\mathbb{P}\left(\mathbb{L}\left(H_j(s^{(j)})\right) = l^{(j)};\right. \\
&\qquad \left. L\left(w(s,i)\right) > \kappa_\delta(b), \text{ for some } i\leq j\right)\left[q_{j,l^{(j)}}(b)\right]^2 \\
\leq\quad &\sum_{i=1}^{j}\left(\sum_{l_i=\kappa_\delta(b)+1}^{m}\sum_{s_i=1}^{r^m} r^{-2m}r^{-l_i}\right)
\end{aligned}
$$

$$\leq \quad 2j \exp\left(-\left(m + \kappa_\delta(b)\right)\Delta\right) = O\left[q(b)\right]^2. \tag{3.33}$$

Combining (3.32) and (3.33), we have

$$\mathbb{E}\left[Z\left(\widetilde{\Pi}_n\right)^2\right] = \mathbb{E}\left[Z\left(\widetilde{\Pi}'_n\right)^2\right] = O\left[q(b)^2\right].$$

And the proof is complete. □

3.7 Numerical Examples

In this section, we implement and test the two proposed splitting based algorithms on the following examples, for various choices of b:

(i) $p_1 = \mathbb{P}\left(X_1 + \cdots + X_4 > b\right)$, where X_j's are Pareto with index $\alpha = 1.5$, i.e., $\mathbb{P}\left(X > x\right) = 1/\left(1+x\right)^\alpha$. Note that this corresponds to the case $\beta = 0$.

(ii) $p_2 = \mathbb{P}\left(Y_1 + \cdots + Y_4 > b\right)$, where Y_j's are Weibull, with parameter 1) $\gamma = 0.2$ and 2) $\gamma = 0.75$, i.e., $\mathbb{P}\left(Y > y\right) = \exp\left(-y^\gamma\right)$. This corresponds to the case $0 < \beta < 1$.

Both algorithms are benchmarked against crude Monte Carlo. The results are demonstrated in Tables 3.1 - 3.3 below. For each algorithm, we report the following quantities:

1) *Estimate.* Both the HFSR and MHFS algorithms are run $N = 10^6$ times. For crude Monte Carlo, we produce $N = 10^8$ replications for each example.

2) *Work-normalized relative error.* For each algorithm, this is calculated as the equivalent relative error of the estimate as if the algorithm is run for the same length of time as the benchmark crude Monte Carlo. In particular, let T, T_c be the running time for the splitting based algorithm and the crude Monte Carlo, respectively, then

the work-normalized relative error for this splitting algorithm is calculated as

$$\overline{RE}^{normalized} = \left(\frac{Var\left(\widehat{p}\right)}{N\widehat{p}} \frac{T_c}{T} \right)^{1/2},$$

where $\widehat{p}$ is the associated estimator under consideration.

3) *Variance reduction factor*, which is calculated as $\overline{RE}^{crudeMC} / \overline{RE}^{normalized}$, where $\overline{RE}^{crudeMC}$ is the relative error of the crude Monte Carlo estimator.

Table 3.1: Numerical results for p_1, i.e., sums of Pareto with $\alpha = 1.5$.

$b = 5 \times 10^4$	Crude MC	HFSR	MHFS
Estimate	3.80×10^{-7}	3.47×10^{-7}	3.51×10^{-7}
Work-normalized rel. err.	16.22%	3.07%	1.89%
Var. reduction factor	1.00	5.29	8.60
$b = 10^5$	Crude MC	HFSR	MHFS
Estimate	1.10×10^{-7}	1.27×10^{-7}	1.25×10^{-7}
Work-normalized rel. err.	30.15%	4.59%	1.25%
Var. reduction factor	1.00	7.02	24.15

Table 3.2: Numerical results for p_2, i.e., sums of Weibull with $\beta = 0.2$.

$b = 10^6$	Crude MC	HFSR	MHFS
Estimate	7.10×10^{-7}	6.23×10^{-7}	6.38×10^{-7}
Work-normalized rel. err.	11.87%	4.09%	1.65%
Var. reduction factor	1.00	2.90	7.18
$b = 2 \times 10^6$	Crude MC	HFSR	MHFS
Estimate	6.00×10^{-8}	5.93×10^{-8}	6.09×10^{-8}
Work-normalized rel. err.	40.82%	4.44%	2.56%
Var. reduction factor	1.00	9.20	15.98

The performance of HFSR and MHFS algorithms illustrated in the tables is consistent with the analysis provided in previous sections. Both algorithms, the MHFS algorithm in particular, display controlled growth of relative error as b increases in all of the three input structures. The less competitive performance of HFSR algorithm reflects our discussions

Table 3.3: Numerical results for p_2, i.e., sums of Weibull with $\beta = 0.75$.

$b = 50$	Crude MC	HFSR	MHFS
Estimate	4.00×10^{-7}	3.91×10^{-7}	3.66×10^{-7}
Work-normalized rel. err.	17.15%	4.40%	1.08%
Var. reduction factor	1.00	3.90	15.87
$b = 55$	Crude MC	HFSR	MHFS
Estimate	8.00×10^{-8}	7.90×10^{-8}	8.49×10^{-8}
Work-normalized rel. err.	35.36%	5.58%	1.38%
Var. reduction factor	1.00	6.34	25.56

at the beginning of Section 3.6. To emphasize again, note that the HFSR algorithm is in essence an importance sampling algorithm. However, the importance sampling phase is highly dependent on the effort taken in the splitting procedure. Moreover, most of the work in the first phase is not utilized when we proceed to sample a different increment. These observations, confirmed by the relative inferior performance of the HFSR estimator shown in the tables, motivated us to develop the MHFS algorithm.

A journey of a thousand miles begins with a single step.

Lao-tzu

4

State Dependent Importance Sampling with Cross Entropy for Heavy-tailed Systems

The cross entropy method is a popular technique that has been used in the context of rare event simulation in order to obtain a good selection (in the sense of variance performance tested empirically) of an importance sampling distribution. This iterative method requires the selection of a suitable parametric family to start with. The selection of the parametric family is very important for the successful application of the method. Two properties must be enforced in such a selection. First, subsequent updates of the

parameters in the iterations must be easily computable and, second, the parametric family should be powerful enough to approximate, in some sense, the zero-variance importance sampling distribution. In this chapter we obtain parametric families for which these two properties are satisfied for a large class of heavy-tailed systems including Pareto and Weibull tails. Our estimators are shown to be strongly efficient in these settings.

4.1 Introduction

Tail probabilities of sums of heavy-tailed increments are a fundamental problem in the applied probability field. A large number of applications boils down to these building blocks. In this chapter we focus our attention on the tail probabilities of a finite sum of heavy-tailed random variables, and we propose a method to improve variance reduction of an existing class of estimators with proved efficiency.

Let $S_m = X_1 + X_2 + ... + X_m$ be a sum of independently and identically distributed (i.i.d.) random variables, with $S_0 = 0$ and that the X_n's are suitably heavy-tailed. The primary interest is the design of efficient estimators for the tail probability of the sum

$$u(b) = \mathbb{P}(S_m > b). \tag{4.1}$$

The basic intuition behind the construction of efficient importance sampling estimators is that one should mimic the behavior of the zero variance change of measure, which coincides with the conditional distribution

$$\mathbb{P}(S \in \cdot | S_m > b) \tag{4.2}$$

(see for example, [8]). Therefore, the behavior of the heavy tailed random walk condi-

tional on the rare event becomes the target to be tracked by paths generated under the importance sampling distribution. It is well known from the theory of heavy-tailed large deviations that this "target" is characterized by the so-called "principle of big jump", which states that as $b \nearrow \infty$ the rare event occurs due to the contribution of a single large increment of size $\Omega(b)$ (see Definition 1.1). On the other hand, paths with more than one jumps of order $\Omega(b)$ shall not be neglected in the construction of importance sampler, because of an observation pointed out by [12] that the second moment of the estimator for heavy tailed large deviation probabilities is very much sensitive to the likelihood ratio of these paths (see also Example 4.1 in Section 4.2).

Guided by these observations, it is natural to suggest a mixture based sampler for the increments as the candidate importance sampler. Recently several state-dependent importance sampling estimators based on such mixtures ([34] and [22]) have been developed and shown to be strongly efficient (which means that the number of samples needed to achieve a fixed relative precision is bounded as $b \nearrow \infty$). In simple words, one samples the next increment from different regions of its support with different probabilities. We shall delay the specific form of the mixture to the next section.

Since the zero variance change of measure (4.2), optimal among all possible sampling distribution, involves the unknown quantity of interest $u(b)$ and is therefore infeasible, the search of global optimal sampling distribution is a futile attempt. But if one restricts optimization within a specific parametric family of sampler, there is hope that an improved change of measure within that family can be obtained. One powerful tool that exactly fits into this setting is *Cross Entropy (CE) minimization* (see for example, [66] and [56]). Instead of directly minimizing the variance of the estimator, the CE method minimizes the cross-entropy discrepancy between two densities. The main advantage of the CE method is that, if the parametric family is well chosen, the optimization problem often admits

closed-form solutions, as opposed to the variance minimization (VM) method (we refer readers to [28] for an in-depth comparison between these two methods).

The successful application of the CE method is closely tied to the quality of the selected parametric family of densities to start with. Two properties must be enforced in such a selection. First, the parametric family should be powerful enough to approximate, in some sense, the zero-variance importance sampling distribution and, second, subsequent updates of the parameters in the iterations must be easily computable. We shall focus on elaborating these properties on the mixture family of our choice in this chapter and demonstrate empirically the performance of this approach applied to the mixture family. We noticed that in existing works, the application of the CE method on estimating tail probabilities of sums of heavy-tailed random variables has been restricted to importance sampling densities that do not capture the "principle of big jump"; for example [28] and [14] considered importance sampling densities by tilting the scale parameters of the Weibull and log-normal increment distributions, respectively. As expected, the corresponding estimators are asymptotically efficient in a weak sense, as opposed to the strong efficiency criterion that our proposed family satisfies (see Theorem 4.1 below). The contribution of this chapter is to justify the applicability of the CE method to a parametric family of densities that capture the large deviations behavior of the heavy-tailed sum, and the resulting estimator is *strongly efficient*.

The rest of the chapter is organized as follows. In Section 4.2 we introduce the assumptions for the heavy-tailed increments, and put forward the parametric family of importance sampling densities to work on. Section 4.4 justifies the preservation of strong efficiency when switching among the same parametric mixture family. In Section 4.5 the CE method is reviewed and we discuss how it can be applied to the mixture family under consideration, after which the iterative equations are derived in closed-form. Finally in

Section 4.6 we test the performance of our approach on two examples and give further discussions.

4.2 Heavy-tailed Increment Distributions

We assume that the increments of the system satisfy the following two assumptions, which encompass virtually all models used in practice, including regularly varying (see Definition 1.7), Weibull and log-normal.

Assumption 4.1. *$X_i \in \mathcal{RV}_\alpha$, for some $\alpha > 1$. Recall from Definition 1.7 that $X_i \in \mathcal{RV}_{-\alpha}$ if $\overline{F}(x) = L(x)x^{-\alpha}$ where $L(\cdot)$ is a slowly varying function.*

Assumption 4.2. *There exists b_0 such that for all $x > b_0$ the following conditions hold.*

2a $\lim_{x\to\infty} x\lambda(x) = \infty$.

2b There exists $\beta_0 \in (0,1)$ such that $\partial \log\Lambda(x) = \lambda(x)/\Lambda(x) \leq \beta_0 x^{-1}$ for $x \geq b_0$.

2c $\Lambda(\cdot)$ is concave for all $x \geq b_0$; equivalently, $\lambda(\cdot)$ is assumed to be non-increasing for $x \geq b_0$.

We remark that under Assumption 4.2, the increment distribution F is essentially assumed to possess a tail at least as heavy as some Weibull distribution with shape parameter $\beta_0 < 1$. Note that under these Assumptions, adopted from [22], the increments X_i's are *subexponential*, i.e., $F \in \mathcal{S}$ (see Definition 1.4), which means that

$$\mathbb{P}\left(S_m > b\right) \sim m\mathbb{P}\left(X_i > b\right), \tag{4.3}$$

as $b \nearrow \infty$ (see Lemma 6 of [22]).

4.3 Parametric Family of IS Distributions

State-dependent importance sampler (SDIS) is designed to sample the increments of the system from a distribution that is dependent on the current status of the system being simulated. We consider a mixture based SDIS. Let us denote by $\underline{p}_j = (p_{j,0}, ..., p_{j,K})$ the vector of mixture probabilities applied to the j-th increment, $j = 1, 2, ..., m-1$, where $K+2$ is the number of mixture determined by the heaviness of the tail (the lighter the tail is, the larger K is). We consider the following family of mixture based densities parameterized by the mixing probabilities

$$\mathbf{p} = \{\underline{p}_1, \underline{p}_2, ..., \underline{p}_{m-1}\} = \{(p_{1,0}, p_{1,1}, ..., p_{1,K}), ..., (p_{m,0}, p_{m,1}..., p_{m,K})\}, \tag{4.4}$$

where $K \geq 0$, from which we sample the k-th increment of the heavy-tailed system:

$$h_k\left(x; \underline{p}_k | S_{k-1} = s\right) = p_{k,0} f_0\left(x|s\right) + \sum_{j=1}^{K} p_j f_j\left(x|s\right) + \left(1 - \sum_{j=0}^{K} p_j\right) f_\dagger\left(x|s\right), \tag{4.5}$$

where $f_\dagger$ and f_j for $j = 0, 1, ..., K$ are properly normalized density functions, which have disjoint supports and depend on the current position of the system $S_{k-1} = s$. The two prevalent specifications are from [34] and [22]. The former works for random walks with increments of regularly varying-type tails that satisfy Assumption 4.1, in which case a mixture of two is used, i.e., $K = 0$. In particular,

$$h_k\left(x|s\right) = \left(\frac{I\left(x > a(b-s)\right)}{\overline{F}\left(a(b-s)\right)} + \frac{I\left(x \leq a(b-s)\right)}{F\left(a(b-s)\right)}\right) f(x), \tag{4.6}$$

where $a \in (0, 1)$ is necessary for analytical reasons and is typically set to be close to 1.

For increments that have distributions covered by Assumption 4.2, for example Weibull, estimators based on two mixtures might fail to achieve bounded relative error. As dis-

cussed in the previous section, this is because the weight of the contribution of those "rogue" paths (i.e., paths with multiple jumps of order $\Omega(b)$) to the relative variance of the estimator is growing increasingly pronounced. Consider the following example.

Example 4.1. *Suppose we are interested in estimating* $\mathbb{P}(X_1 + X_2 > b)$*, where* X_1, X_2 *are i.i.d. Weibull with parameter* $\beta \in (0,1)$*, i.e.,* $\mathbb{P}(X_i > t) = \overline{F}(t) = \exp\left(-t^\beta\right)$*. Note that* $\mathbb{P}(X_1 + X_2 > b) \sim \mathbb{P}(X_1 > b) + \mathbb{P}(X_2 > b)$ *due to the properties of subexponential distributions. A two-mixture sampler leads to the following importance sampling strategy: sample the increments*

$$(Y_1, Y_2) = \begin{cases} \left(X_1, X_2 \middle| (X_1;\ X_2 > b - X_1)\right) & w.p. 1/2 \\ \left(X_1 \middle| (X_2;\ X_1 > b - X_2), X_2\right) & w.p. 1/2. \end{cases}$$

The corresponding IS estimator is therefore

$$\hat{\mu}_b = \frac{f_{X_1}(y_1) f_{X_2}(y_2)}{f_{X_1,X_2}(y_1, y_2)} = \frac{2\overline{F}(b - y_1)\overline{F}(b - y_2) I(y_1 + y_2 > b)}{\overline{F}(b - y_1) + \overline{F}(b - y_2)}.$$

It's not hard to see that for some choice of $\beta < 1$*, the relative error is unbounded as* $b \nearrow \infty$*. In particular, consider the path* $(y_1, y_2) = (b/2, b/2)$*, one has*

$$\begin{aligned} \frac{\mathbb{E}(\hat{\mu}_b^2)}{\mathbb{P}(X_1 + X_2 > b)^2} &= \frac{\mathbb{E}_{\mathbf{p}}(\hat{\mu}_b)}{\mathbb{P}(X_1 + X_2 > b)^2} \\ &\geq \frac{1}{\mathbb{P}(X_1 + X_2 > b)^2} \frac{f_{X_1}(b/2) f_{X_2}(b/2)}{f_{Y_1,Y_2}(b/2, b/2)} f_{X_1}(b/2) f_{X_2}(b/2) \\ &= \frac{\overline{F}(b/2)^2 f_{X_1}(b/2)^2}{\mathbb{P}(X_1 + X_2 > b)^2 \overline{F}(b/2)} \approx \frac{\exp\left(-3(b/2)^\beta + 2b^\beta\right)}{4}, \end{aligned}$$

which grows rapidly as $b \nearrow \infty$ *if e.g.,* $\beta = 2/3$*.*

As the previous example illustrates, more mixtures are needed for the increments

covered by Assumption 4.2 to absorb the impact of such "rogue" paths on the second moment of the estimator. Following this observation, [22] proposed a multi-point mixture family, which is general enough to cover all the increment types that satisfy Assumption 4.1 and Assumption 4.2. The support of the mixture based densities is defined in terms of the hazard function of the increments, and the number of mixtures used is dependent on the tail heaviness of the increments which is expressed in terms of the concavity of the hazard function of the increment distribution. More mixtures are needed when the tails are not as heavy as regularly varying, for example Weibull. More precisely, let $\Lambda(x) = -\log \overline{F}(x)$ be the integrated hazard function of the increments, given $a_*, a_{**} > 0$, let

$$f_0(x|s) = f(x)\frac{I\left(x \leq b - s - \Lambda^{-1}\left(\Lambda(b-s) - a_*\right)\right)}{\mathbb{P}\left(x \leq b - s - \Lambda^{-1}\left(\Lambda(b-s) - a_*\right)\right)}, \tag{4.7}$$

and

$$f_\dagger(x|s) = f(x)\frac{I\left(x > b - s - \Lambda^{-1}\left(\Lambda(b-s) - a_{**}\right)\right)}{\mathbb{P}\left(x > b - s - \Lambda^{-1}\left(\Lambda(b-s) - a_{**}\right)\right)}. \tag{4.8}$$

The densities f_j's are defined by a set of cut-off points $c_j = a_j(b-s)$ for $j = 1, 2, ..., K-1$ where $0 < a_1 < a_2 < ... < a_{K-1} < 1$ is a sequence satisfying, for given $\beta_0 \in (0,1)$ and a positive constant σ_1,

$$a_j^\beta + (1 - a_{j+1})^\beta \geq 1 + \sigma_2,$$

and

$$a_{j+1} - a_j \leq \sigma_1/2,$$

for each $1 \leq j \leq K-2$ for some $\sigma_2 > 0$, and $a_{K-1} \geq 1 - \sigma_1, a_1 \leq \sigma_1$. Set $c_0 = b - s - \Lambda^{-1}\left(\Lambda(b-s) - a_*\right)$, $c_K = b - s - \Lambda^{-1}\left(\Lambda(b-s) - a_{**}\right)$ and write $c_{-1} = -\infty$, we define

$$f_j(x) = \tag{4.9}$$

$$\begin{cases} f(x)I\left(x \in (c_{j-1}, c_j]\right)/\mathbb{P}\left(X \in (c_{j-1}, c_j]\right), & 0 \leq j \leq K-1 \\ f(b-s-x)I\left(x \in (c_{K-1}, c_K]\right)/\mathbb{P}\left(X \in (b-s-c_K, b-s-c_{K-1}]\right), & j = K \\ f(x)I\left(x \in (c_K, \infty)\right)/P\left(X \in (c_K, \infty)\right), & j = \dagger \end{cases}$$

for $j = 1, 2, ..., K$. Note that the two specifications of the mixtures (by [34] and [22]) have the same spirits when the increments are regularly varying (see equation (14) in [22]). [22] also showed that this mixture based distribution converges in total variation to the zero-variance distribution in a certain random walk problem, as $b \nearrow \infty$. In what follows, unless specified otherwise, we shall work on the general form of the mixture given in (1.8), i.e.,

$$\begin{aligned} & h_k\left(x; \underline{p}_k \middle| S_{k-1} = s\right) \\ = \quad & \left(\sum_{j=0}^{K} p_{k,j} I\left(A_j\left(s\right)\right) w_j\left(s, x\right) + \left(1 - \sum_{j=0}^{K} p_{k,j}\right) I\left(A_\dagger(s)\right) w_\dagger\left(s, x\right)\right) f(x), \end{aligned}$$

where $A_\dagger(s) = \overline{\bigcup_{j=0}^{K} A_j(s)}$, and $w_j(s,x), w_\dagger(s,x) > 0$ satisfy $\mathbb{E}(w_j(s,X)) = \mathbb{E}(w_\dagger(s,X)) = 1$. Note that the mixture family specified by [34] corresponds to setting

$$w_0\left(s, x\right) = \frac{I\left(x \leq a(b-s)\right)}{F\left(a(b-s)\right)}, \quad w_\dagger\left(s, x\right) = \frac{I\left(x > a(b-s)\right)}{\overline{F}\left(a(b-s)\right)}.$$

And the one proposed by [22] corresponds to setting

$$w_j\left(s, x\right) = \frac{I\left(A_j(s)\right)}{\mathbb{P}\left(A_j(s)\right)} = \frac{I\left(x \in (c_{j-1}, c_j]\right)}{\mathbb{P}\left(x \in (c_{j-1}, c_j]\right)},$$

for $j = 0, 1, ..., K-1$ and again write $c_{-1} = -\infty$. And

$$w_{\dagger}(s,x) = \frac{f(b-s-x)I\left(x \in (c_{K-1}, c_K]\right)}{f(x)\mathbb{P}\left(X \in (b-s-c_K, b-s-c_{K-1}]\right)}, \quad w_{\dagger}(s,x) = \frac{I\left(x \in (c_K, \infty]\right)}{\mathbb{P}\left(x \in (c_K, \infty]\right)}$$

If we write the joint density of the increments under the original measure as

$$\mathbf{f}(\mathbf{x}) = f(x_1) f(x_2) ... f(x_m),$$

where $\mathbf{x} = (x_1, ..., x_m)$, and we can express the joint importance sampling density for the mixture based SDIS as

$$\begin{aligned} & h(\mathbf{x}; \mathbf{p}) \\ = & \prod_{k=1}^{m-1} \left[\sum_{j=0}^{K} p_{k,j} I\left(A_j(s_{k-1})\right) w_j(s, x_k) + \left(1 - \sum_{j=0}^{K} p_{k,j}\right) I\left(A_{\dagger}(s_{k-1})\right) w_{\dagger}(s, x_k) \right] \\ & \cdot \left(I\left(S_{m-1} < b\right) \mathbb{P}\left(X_m > (b - S_{m-1})\right) + I\left(S_{m-1} \geq b\right) \right) \mathbf{f}(\mathbf{x}). \end{aligned} \tag{4.10}$$

And the associated SDIS estimator for $u(b)$ is therefore defined as

$$\begin{aligned} Z_m(b; \mathbf{p}) = & \prod_{k=1}^{m-1} \left[\sum_{j=0}^{K} \frac{I\left(A_j(S_{k-1})\right)}{p_{k,j} w_j(S_{k-1}, X_k)} + \frac{I\left(A_{\dagger}(S_{k-1})\right)}{w_{\dagger}(S_{k-1}, X_k)\left(1 - \sum_{j=0}^{K} p_{k,j}\right)} \right] \\ & \times \left(\frac{I(S_{m-1} > b)}{\mathbb{P}\left(X_m > b - S_{m-1}\right)} + I(S_{m-1} > b) \right), \end{aligned} \tag{4.11}$$

where $\mathbf{p}$ is the mixing probability vector defined in (4.4).

4.4 Strong Efficiency of the Family under Consideration

The following theorem states the efficiency property of the mixture family. In particular, the mixture family remains in the class of *strongly efficient* estimators, subject to mild conditions on the mixing parameters. The proof of which boils down to the construction of a valid *Lyapunov function*, as introduced in Subsection 1.2.7.

Theorem 4.1. *Let $\mathbb{P}_{\mathbf{p}}$ be the measure induced by the mixture family with mixing probability vector $\mathbf{p}$, and let $\mathbb{E}_{\mathbf{p}}$ be the associated expectation operator. If there exists a $\xi > 0$ such that $\mathbf{p} > \xi \cdot \mathbf{1}$, for all $b > 0$, where $\mathbf{1}$ is a vector of ones of dimension $(m-1) \times (K+2)$, then one can explicitly compute $\mathcal{K} \in (0, \infty)$, uniform in b, such that*

$$\frac{\mathbb{E}_{\mathbf{p}}\left[Z_m(b;\mathbf{p})^2\right]}{u(b)^2} < \mathcal{K},$$

as $b \nearrow \infty$, where the estimator $Z_m(b;\mathbf{p})$ is defined in (4.11). In particular, $Z_m(b;\mathbf{p})$ is strongly efficient for estimating $u(b)$.

Since the estimator introduced in [22] covers both Assumptions 4.1 and 4.2, and the mixture-based estimator proposed in [34] can be shown to be equivalent to the one given in [22] under Assumption 4.1, it suffices to work on the mixture given in [22]. The discussions at the end of Subsection 1.2.7 suggest that a natural candidate for the Lyapunov function, $v(s)$, at time k, is approximately $\mathbb{P}\left(S_m > b | S_{k-1} = s\right)^2$. In fact it suffices to work on the following straightforward choice,

$$v(s) = \overline{F}(b-s)^2. \tag{4.12}$$

The associated Lyapunov inequality (see Lemma 1.5) can therefore be written as

$$\mathbb{E}\left[\frac{v\left(s+X\right)}{v(s)}\zeta\left(s,X\right)\right]\leq c, \tag{4.13}$$

for some constant $c\in(0,\infty)$ independent of b, where $\zeta\left(S_{k-1},X_k\right)$ is the local likelihood function between the original measure and the one induced by the mixture sampling density at the k-th step. Let us write the left hand side of (4.13) according to the following decomposition

$$\mathbb{E}\left[\frac{v\left(s+X\right)}{v(s)}\zeta\left(s,X\right)\right]=\sum_{j=0}^{K}\frac{J_j}{p_{k,j}}+\frac{J_\dagger}{p_{k,\dagger}},$$

where $p_{k,\dagger}=1-\sum_{j=0}^{K}p_{k,j}$, and specifically,

$$J_\dagger = \mathbb{P}\left(X>\Lambda^{-1}\big(\Lambda(b-s)-a_{**}\big)\right)\mathbb{E}\left[\frac{v(s+X)}{v(s)};X>\Lambda^{-1}\big(\Lambda(b-s)-a_{**}\big)\right] \tag{4.14}$$

$$\begin{aligned}J_0 &= \mathbb{P}\left(X\leq b-s-\Lambda^{-1}\big(\Lambda(b-s)-a_{*}\big)\right)\\ &\quad\times\mathbb{E}\left[\frac{v(s+X)}{v(s)};X\leq b-s-\Lambda^{-1}\big(\Lambda(b-s)-a_{*}\big)\right]\end{aligned} \tag{4.15}$$

$$J_j = \mathbb{P}\left(X\in(c_{j-1},c_j]\right)\mathbb{E}\left[\frac{v(s+X)}{v(s)};X\in(c_{j-1},c_j]\right],\ \text{for } j=1,\ldots,K-1 \tag{4.16}$$

$$J_K = \mathbb{P}\left(b-s-X\in(c_{k-1},c_k]\right)\mathbb{E}\left[\frac{v(s+X)f(X)}{v(s)f(b-s-X)};X\in(c_{k-1},c_k]\right]. \tag{4.17}$$

Therefore the proof of the previous result boils down to carefully upper bounding each of the previous term so that

$$\sum_{j=0}^{K}\frac{J_j}{p_{k,j}}+\frac{J_\dagger}{p_{k,\dagger}}\leq c.$$

The following lemma is useful for deriving an upper bound for J_j, $1\leq j\leq K$, which corresponds to Lemma 4 in [22] and we therefore dispense ourselves with the proof.

Lemma 4.1. *Under Assumption 4.2, the following holds,*

$$\frac{\Lambda(x)}{\Lambda(x+y)} \geq \left(\frac{x}{x+y}\right)^{\beta_0},$$

for all $x \geq b_0$ *and* $y \geq 0$.

We now proceed to carry out our plan in details.

Proof. **1) The term $J_\dagger$.**

By definition simply note that $v(s) \leq 1$, therefore we have

$$\begin{aligned} J_\dagger &\leq \frac{\mathbb{P}\left(X > \Lambda^{-1}\big(\Lambda(b-s) - a_{**}\big)\right)^2}{v(s)} \\ &= \exp\left(2a_{**}\right) \frac{\overline{F}^2(b-s)}{v(s)} = \exp\left(2a_{**}\right). \end{aligned} \tag{4.18}$$

2) The term J_0.

We can bound J_0 from above as follows,

$$\begin{aligned} J_0 &\leq \mathbb{E}\left[\frac{v(s+X)}{v(s)}; X \leq b - s - \Lambda^{-1}\big(\Lambda(b-s) - a_*\big)\right] \\ &\leq \frac{\overline{F}\left(\Lambda^{-1}\big(\Lambda(b-s) - a_*\big)\right)^2}{\overline{F}(b-s)^2} = \exp\left(2a_*\right). \end{aligned} \tag{4.19}$$

3) The terms $J_j, j = 2, \ldots, K-1$.

By virtue of Lemma 4.1, we have

$$\Lambda(x) + \Lambda(y) - \Lambda(x+y+z) \geq \Lambda(x+y+z)\left(\left(\frac{x}{x+y+z}\right)^{\beta_0} + \left(\frac{y}{x+y+z}\right)^{\beta_0} - 1\right),$$

for sufficiently large x, y, z. Therefore, as $b - s \nearrow \infty$,

$$\begin{aligned} J_j &= \frac{\mathbb{P}\big(X \in (c_{j-1}, c_j]\big)}{\overline{F}\,(b-s)^2} \int_{c_{j-1}}^{c_j} \overline{F}\,(b-s-x)^2 f(x)dx \\ &\leq \frac{\overline{F}\,(c_{j-1})^2\,\overline{F}\,(b-s-c_j)^2}{\overline{F}\,(b-s)^2} \\ &\leq \exp\Big(2\Lambda(b-s) - 2\Lambda(c_{j-1}) - 2\Lambda(b-s-c_j)\Big) \\ &\leq \exp\Big(-2\Lambda(b-s)\Big(a_{j-1}^{\beta_0} + (1-a_j)^{\beta_0} - 1\Big)\Big) \leq 1. \end{aligned} \tag{4.20}$$

4) The term J_1.

Once again from Lemma 4.1, for $x \in \big[b - s - \Lambda^{-1}\left(\Lambda(b-s) - a_*\right), a_1(b-s)\big]$, we have

$$\Lambda(x) + \Lambda(b-s-x) - \Lambda(b-s) \geq \Lambda(b-s)\left(\left(\frac{x}{b-s}\right)^{\beta_0} + \left(\frac{b-s-x}{b-s}\right)^{\beta_0} - 1\right),$$

and

$$\Lambda(b-s) - \Lambda(b-s-x) \leq \Lambda(b-s)\left(1 - \left(1 - \frac{1}{(b-s)}\right)^{\beta_0}\right).$$

Combining the preceding two inequalities, we obtain

$$2\Lambda(b-s) - 2\Lambda(b-s-x) - \Lambda(x) \geq \Lambda(b-s)\left(2 - 2\left(1 - \frac{x}{b-s}\right)^{\beta_0} - \left(\frac{x}{b-s}\right)^{\beta_0}\right) \leq 0.$$

Hence, along with the fact that $\lim_{x\to\infty} \lambda(x) = 0$, we have, , as $b - s \nearrow \infty$,

$$\begin{aligned} J_1 &= \frac{\mathbb{P}\big(X > b - s - \Lambda^{-1}\left(\Lambda(b-s) - a_*\right)\big)}{\overline{F}\,(b-s)^2} \int_{b-s-\Lambda^{-1}(\Lambda(b-s)-a_*)}^{c_1} \overline{F}\,(b-s-x)^2 f(x)dx \\ &\leq \int_{b-s-\Lambda^{-1}(\Lambda(b-s)-a_*)}^{c_1} \exp\big(2\Lambda(b-s) - 2\Lambda(b-s-x) - \Lambda(x)\big)dx \leq \delta_1, \end{aligned} \tag{4.21}$$

for some $\delta_1 > 0$ independent of b.

5) The term J_K.

Note that by construction (see the paragraph before (4.9)),

$$c_{K-1} = a_{K-1}(b-s) \geq (1-\sigma_1)(b-s),$$

for sufficiently small but positive σ_1. Therefore, by resorting to Lemma 4.1 one last time, we have

$$2\Lambda(b-s) - 2\Lambda(x) - \Lambda(b-s-x) \leq 2 - 2\left(\frac{x}{b-s}\right)^{\beta_0} - \left(1-\frac{x}{b-s}\right)^{\beta_0} \leq 0, \qquad (4.22)$$

which leads to

$$\begin{aligned} J_K &= \mathbb{P}\left(b-s-c_K, b-s-c_{K-1}\right) \int_{c_{K-1}}^{c_K} \frac{\overline{F}(b-s-x)^2}{\overline{F}(b-s)^2} \frac{f^2(x)}{f(b-s-x)} dx \\ &\leq \int_{c_{K-1}}^{c_K} \frac{\lambda^2(x)}{\lambda(b-s-x)} \exp\big(2\Lambda(b-s) - 2\Lambda(x) - \Lambda(b-s-x)\big) dx \\ &\leq \delta_2, \end{aligned} \qquad (4.23)$$

for some $\delta_2 > 0$ independent of b, as $b-s \nearrow \infty$. Here the last inequality arises due to (4.22) and the fact that $\lambda^{-1}(x)$ grows at most linearly in x by Assumption 4.2-b).

In summary, by combining (4.18), (4.19), (4.20), (4.21) and (4.23), we arrive at

$$\sum_{j=0}^{K} \frac{J_j}{p_{k,j}} + \frac{J_\dagger}{p_{k,\dagger}} \leq \xi\overline{\delta} = c, \qquad (4.24)$$

where $\xi = \min_{1\leq k\leq m, j\in\{\dagger,0,\ldots,K\}} p_{k,j}$, and $\overline{\delta} = (K+2)\max\{\exp(2a_{**}), \exp(2a_*), 1, \delta_1, \delta_2\}$.

Now by definition it is clear that $v(0) = \mathbb{P}\left(S_m > b\right)^2$, and it suffices to pick $\rho = 1$ in

Lemma 1.5. The result in Lemma 1.5 allows us to conclude that

$$\mathbb{E}_{\mathbf{p}}\left[Z_m(b;\mathbf{p})^2\right] \leq c^m v(0) \leq c^m u^2(b),$$

where c is defined in (4.24). □

Remark 4.1. *The result enables us to comfortably switch to different choices of mixing probabilities within the same parametric family without violating the strong efficiency property of the final estimator, which lays the ground for the applicability of the CE method to be introduced shortly.*

4.5 Cross Entropy Method and the Iterative Equations for the Mixture Family

4.5.1 Review of Cross-Entropy Method

If we restrict our search of importance sampler to this particular parametric class, the optimal choice of the vector $\mathbf{p}$ can be obtained by minimizing the so-called *Kullback-Leibler divergence* or the *cross-entropy distance.*

Definition 4.1. *The Kullback-Leibler cross-entropy between two densities g and h is given by*

$$\begin{aligned}\mathcal{D}(g,h) &= \int g(\mathbf{x}) \log \frac{g(\mathbf{x})}{h(\mathbf{x})} d\mathbf{x} \\ &= \int g(\mathbf{x}) \log g(\mathbf{x}) d\mathbf{x} - \int g(\mathbf{x}) \log h(\mathbf{x}) d\mathbf{x}. \end{aligned} \tag{4.25}$$

If we fix g to be the optimal importance sampling density $g^*(\mathbf{x}) \propto \varphi(S(\mathbf{x};b))\, f(\mathbf{x})$, where $\varphi(S(\mathbf{x};b))$ is the performance measure of the system (for example, $S(\mathbf{X}) = \sum_{j=1}^m X_j$,

and $\varphi\left(S\left(\mathbf{x};b\right)\right)=I\left(S\left(\mathbf{x}\right)>b\right)$), then our search of the optimal mixture is the output of the following *parametric* optimization problem

$$\begin{aligned}\min_{\mathbf{p}}\mathcal{D}\left(g^{*},h(\cdot,\mathbf{p})\right)\Longleftrightarrow\max_{\mathbf{p}}D\left(\mathbf{p}\right)&=\max_{\mathbf{p}}\mathbb{E}_{\mathbf{p}^{\star}}\varphi\left(S\left(\mathbf{X};b\right)\right)\log h\left(\mathbf{X};\mathbf{p}\right)\\&=\max_{\mathbf{p}}\mathbb{E}_{\widetilde{\mathbf{p}}}\varphi\left(S\left(\mathbf{X};b\right)\right)\frac{h\left(\mathbf{X};\mathbf{p}^{\star}\right)}{h\left(\mathbf{X};\widetilde{\mathbf{p}}\right)}\log h\left(\mathbf{X};\mathbf{p}\right)\\&=\max_{\mathbf{p}}\mathbb{E}_{\widetilde{\mathbf{p}}}\varphi\left(S\left(\mathbf{X};b\right)\right)\frac{\mathbf{f}\left(\mathbf{X}\right)}{h\left(\mathbf{X};\widetilde{\mathbf{p}}\right)}\log h\left(\mathbf{X};\mathbf{p}\right),\end{aligned}\tag{4.26}$$

where $\mathbf{f}\left(\mathbf{X}\right)/h\left(\mathbf{X};\widetilde{\mathbf{p}}\right)$ is the likelihood ratio between the original measure and the measure induced by the mixture based density with some fixed parameter $\widetilde{\mathbf{p}}$ (Recall that $\mathbf{X}=(X_1,...,X_m)$). In particular,

$$\begin{aligned}\frac{\mathbf{f}\left(\mathbf{X}\right)}{h\left(\mathbf{X};\widetilde{\mathbf{p}}\right)}=\prod_{k=1}^{m-1}&\left(\sum_{j=0}^{K}\frac{I\left(x_k\in A_j\left(S_{k-1}\right)\right)}{\widetilde{p}_{k,j}w_j\left(S_{k-1},x_k\right)}+\frac{I\left(x_k\in A_{\dagger}(S_{k-1})\right)}{\left(1-\sum_{j=0}^{K}\widetilde{p}_{k,j}\right)w_{\dagger}\left(S_{k-1},x_k\right)}\right)\\&\cdot\left(I\left(S_{m-1}<b\right)\mathbb{P}\left(X_m>\left(b-S_{m-1}\right)\right)+I\left(S_{m-1}\geq b\right)\right).\end{aligned}\tag{4.27}$$

In most cases the expectation in (4.26) is analytically inaccessible. [66] suggested a recursive method based on the following stochastic counterpart of (4.26)

$$\max_{\mathbf{p}}\hat{D}\left(\mathbf{p}\right)=\max_{\mathbf{p}}\frac{1}{N}\sum_{i=1}^{N}\varphi\left(S\left(\mathbf{X}(i)\right);b\right)\frac{\mathbf{f}\left(\mathbf{X}(i)\right)}{h\left(\mathbf{X}(i);\widetilde{\mathbf{p}}\right)}\log h\left(\mathbf{X}(i),\mathbf{p}\right).\tag{4.28}$$

Cross Entropy (CE) Algorithm [66]

---

1. Choose an initial vector of mixing probabilities $\mathbf{p}^{(0)}$. Set $T=1$.
2. Generate a random sample $\mathbf{X}_1,...,\mathbf{X}_N$ from the joint density $h\left(\cdot;\mathbf{p}^{(T-1)}\right)$.

3. Solve the stochastic optimization program (4.28). Denote the solution by $\mathbf{p}^{(T)}$, i.e.,

$$\mathbf{p}^{(T)} = \arg\min_{\mathbf{p}} \frac{1}{N} \sum_{i=1}^{N} \varphi\left(S\left(\mathbf{X}(i)\right); b\right) \frac{\mathbf{f}\left(\mathbf{X}(i)\right)}{h\left(\mathbf{X}(i); \mathbf{p}^{(T-1)}\right)} \log h\left(\mathbf{X}(i), \mathbf{p}\right).$$

4. **Stop** if convergence is reached; otherwise, set $T = T + 1$, go to Step 2.

It's very convenient to embed the CE algorithm in the main SDIS algorithm to further reduce variance. Let M be the total simulation budget, and τ be the number of recursions in the CE algorithm until convergence of $\mathbf{p}$. If $\tau N < M$, then the SDIS with CE algorithm add-on corresponds to generating τ batches of independent samples from the mixture based importance sampling density parameterized by $\mathbf{p}^{(T)}$, for $T = 0, 1, ..., \tau - 1$, and one batch of size $M - \tau N$ of independent samples from the importance density with optimal CE probability vector $\mathbf{p}^*$. Depending on the size of $M - \tau N$, the final estimator can be obtained by averaging either the last batch of $M - \tau N$ samples, or the entire M samples from different batches. In either case we are able to achieve variance reduction while maintaining strong efficiency property. Even for the case where $\tau N \geq M$, the improved cross-entropy after each iteration typically will reduce the variance of the future samples over those from previous iterations, since each iteration gives us a parameterized density closer to the zero-variance importance density.

4.5.2 Iterative Equations for the Mixture IS Family

We now proceed to characterize the solution to (4.28). In the case where we are interested in the tail probability of the sum $\mathbb{P}\left(S_m > b\right)$, $\varphi\left(S\left(\mathbf{X}\right); b\right) = I\left(S_m > b\right)$. Note that $\hat{D}$ is concave and differentiable with respect to the components p_k, therefore the solution to

(4.28) is directly given by the first order optimality condition:

$$\sum_{i=1}^{N} I\left(S_m(i) > b\right) \frac{\mathbf{f}\left(\mathbf{X}(i)\right)}{h\left(\mathbf{X}(i); \widetilde{\mathbf{p}}\right)} \bigtriangledown_{\mathbf{p}} \log h\left(\mathbf{X}(i), \mathbf{p}\right) = 0. \tag{4.29}$$

The product structure of the likelihood function is particularly useful because the sensitivity of the likelihood function to the mixing probabilities can be localized. Indeed, a few lines of elementary algebra gives

$$\begin{aligned}
& \frac{d \log h\left(\mathbf{X}, \mathbf{p}\right)}{dp_{k,l}} \\
= \quad & \left(I\left(X_k \in A_l\left(S_{k-1}\right)\right) w_l\left(S_{k-1}, X_k\right) - I\left(X_k \in A_\dagger\left(S_{k-1}\right)\right) w_\dagger\left(S_{k-1}, X_k\right) \right) \Big/ \\
& \left[\sum_{j=0}^{K} p_{k,j} I\left(X_k \in A_j\left(S_{k-1}\right)\right) w_j\left(S_{k-1}, X_k\right) \right. \\
& \left. + \left(1 - \sum_{j=0}^{K} p_{k,j}\right) I\left(X_k \in A_\dagger(S_{k-1})\right) w_\dagger\left(S_{k-1}, X_k\right) \right] \\
= \quad & \frac{I\left(X_k \in A_l\left(S_{k-1}\right)\right)}{p_{k,l}} - \frac{I\left(X_k \in A_\dagger\left(S_{k-1}\right)\right)}{1 - \sum_{j=0}^{K} p_{k,j}}.
\end{aligned} \tag{4.30}$$

We denote

$$\begin{aligned}
& W\left(\mathbf{X}_{-l}(i); \mathbf{p}^\star, \widetilde{\mathbf{p}}\right) = \\
& \prod_{k=1, k\neq l}^{m-1} \frac{h\left(X_k(i); \underline{p}_k^\star\right)}{h\left(X_k(i); \widetilde{\underline{p}}_k\right)} \left(I\left(S_{m-1} < b\right) \mathbb{P}\left(X_m(i) > \left(b - S_{m-1}(i)\right)\right) + I\left(S_{m-1}(i) \geq b\right) \right),
\end{aligned}$$

where $\underline{p}_k^\star = \{p_{k,0}^\star, ...p_{k,K}^\star\}$, and $\widetilde{\underline{p}}_k = \{\widetilde{p}_{k,0}, ...\widetilde{p}_{k,K}\}$. And further let

$$\Theta_{l,j} = \frac{\sum_{i=1}^{N} W\left(\mathbf{X}_{-l}(i); \mathbf{p}^\star, \widetilde{\mathbf{p}}\right) \left(1 - \sum_{j=0}^{K} \widetilde{p}_{l,j}\right) w_\dagger\left(S_{l-1}, X_l(i)\right)}{\sum_{i=1}^{N} W\left(\mathbf{X}_{-l}(i); \mathbf{p}^\star, \widetilde{\mathbf{p}}\right) \widetilde{p}_{l,j} w_l\left(S_{l-1}, X_l(i)\right)}.$$

The first order optimality condition (4.29) therefore yields the following solution $\mathbf{p}^*$

to the stochastic optimization problem (4.28), we shall call this vector of optimal solution *optimal CE mixing probability vector*:

$$p_{l,j}^{*} = \frac{\Theta_{l,j}}{1 + \sum_{k=0}^{K} \Theta_{k,j}}, \tag{4.31}$$

for $j = 0, 1, ..., K$ and $l = 1, 2, ..., m$. It doesn't take long to realize that the previous expression has the following equivalent form

$$p_{l,j}^{\star} = \frac{\sum_{i=1}^{N} I\left(S_m(i) > b\right) W\left(\mathbf{X}(i); \mathbf{p}^{\star}, \widetilde{\mathbf{p}}\right) I\left(X_l \in A_j\left(S_{l-1}\right)\right)}{\sum_{i=1}^{N} I\left(S_m(i) > b\right) W\left(\mathbf{X}(i); \mathbf{p}^{\star}, \widetilde{\mathbf{p}}\right)}, \tag{4.32}$$

for $j = 0, 1, ..., K$ and $k = 1, 2, ..., m$, where $W\left(\cdot; \mathbf{p}^{\star}, \widetilde{\mathbf{p}}\right) = h\left(\cdot; \mathbf{p}^{\star}\right) / h\left(\cdot; \widetilde{\mathbf{p}}\right) = \mathbf{f}\left(\cdot\right) / h\left(\cdot; \widetilde{\mathbf{p}}\right)$ is given by (4.27). It's worth pointing out that (4.32) is computationally advantageous over (4.31), because it avoids dividing by zero in computing $\Theta_{l,j}$, especially when the number of "pilot" run is small. (Note that the sampling of the mth increment ensures $S_m(i) > b$.) Moreover, the expression (4.32) entails a nice interpretation: the optimal mixing probability is the proportion of the contribution to the likelihood function from the jth "band" of the kth increment.

For completeness we also include the explicit iteration equations for cases where the increments satisfy Assumption 4.1 and 4.2, respectively. We write, for ease of exposition,

$$W_m(i) = \left(I\left(S_{m-1}(i) < b\right) \mathbb{P}\left(X_m(i) > \left(b - S_{m-1}(i)\right)\right) + I\left(S_{m-1}(i) > b\right)\right).$$

For regularly varying increments, the solution for the Tth iteration of the recursive algorithm can be written as

$$p_k^{(T)} = \left[\sum_{i=1}^{N} I\left(S_m(i) > b; X_k > a(b - s_{k-1})\right) W_m(i)\right.$$

$$\cdot \prod_{k=1}^{m-1} \left(\frac{\mathbb{P}\left(X_k > a(b-s_{k-1})\right)}{p_k^{(T-1)} I\left(X_k > a(b-s_{k-1})\right)} + \frac{\mathbb{P}\left(X_k \le a(b-s_{k-1})\right)}{\left(1-p_k^{(T-1)}\right) I\left(X_k \le a(b-s_{k-1})\right)} \right) \Bigg] \Bigg/$$

$$\Bigg[\sum_{i=1}^{N} I\left(S_m(i) > b\right) W_m(i)$$

$$\cdot \prod_{k=1}^{m-1} \left(\frac{\mathbb{P}\left(X_k > a(b-s_{k-1})\right)}{p_k^{(T-1)} I\left(X_k > a(b-s_{k-1})\right)} + \frac{\mathbb{P}\left(X_k \le a(b-s_{k-1})\right)}{\left(1-p_k^{(T-1)}\right) I\left(X_k \le a(b-s_{k-1})\right)} \right) \Bigg] \quad (4.33)$$

For increment distributions that satisfy Assumption 4.2, $W(\cdot; \mathbf{p}^\star, \mathbf{p}^{(T-1)})$, the likelihood function, becomes

$$\begin{aligned} W\left(\mathbf{X}^{(T-1)}; \mathbf{p}^\star, \mathbf{p}^{(T-1)}\right) &= \frac{\mathbf{f}\left(\mathbf{x}^{(T-1)}\right)}{h\left(\mathbf{X}^{(T-1)}, \mathbf{p}^{(T-1)}\right)} \\ &= \prod_{k=1}^{m-1} W_m(i) \left(\frac{\mathbb{P}\left(X_k^{(T-1)} \le c_0\right)}{p_{k,0}^{(T-1)} I\left(x_k^{(T-1)} \le c_0\right)} \right. \\ &+ \frac{\mathbb{P}\left(X_k^{(T-1)} > c_K\right)}{\left(1-\sum_{j=0}^{K} p_{k,j}^{(T-1)}\right) I\left(X_k^{(T-1)} > c_K\right)} + \sum_{j=1}^{K-1} \frac{\mathbb{P}\left(X_k^{(T-1)} \in (c_{j-1}, c_j]\right)}{p_{k,j}^{(T-1)} I\left(x_k^{(T-1)} \in (c_{j-1}, c_j]\right)} \\ &\left. + \frac{f(b-s-x_k^{(T-1)}) \mathbb{P}\left(X_k^{(T-1)} \in (b-s-c_{K-1}, b-s-c_K]\right)}{p_{k,K}^{(T-1)} f(x_k^{(T-1)}) I\left(x_k^{(T-1)} \in (c_{K-1}, c_K]\right)} \right), \end{aligned}$$

where c_j's are the cutoff points of the "bands" and we have explicitly written out the iteration count. Note that at the beginning of iteration T, the only part that is dependent on the unknown parameters $\mathbf{p}$ in the stochastic program (4.28) is $\log h\left(\mathbf{X}(i), \mathbf{p}^{(T)}\right)$ and hence $\bigtriangledown_{\mathbf{p}} \ln h\left(\mathbf{X}(i), \mathbf{p}^{(T)}\right)$ in the optimality condition (4.29); $W\left(\cdot; \mathbf{p}^\star, \mathbf{p}^{(T-1)}\right)$ is a function of the probability vector passed from the $(T-1)$st iteration as well as the samples generated from IS density specified by that probability vector. In that regard at the beginning of the Tth iteration, all the ingredients in the expression above are available. The iteration

equation for the probability vector at iteration T is therefore given by

$$p_{k,j}^{(T)} = \frac{\sum_{i=1}^{N} I\left(S_m^{(T-1)}(i) > b\right) W\left(\mathbf{X}^{T-1}(i); \mathbf{p}^{\star}, \mathbf{p}^{(T-1)}\right) I\left(x_k^{(T-1)} \in (c_{j-1}, j_k]\right)}{\sum_{i=1}^{N} I\left(S_m^{(T-1)}(i) > b\right) W\left(\mathbf{X}(i)^{(T-1)}; \mathbf{p}^{\star}, \mathbf{p}^{(T-1)}\right)},$$

where $c_{-1} = -\infty$ with a slight abuse of notations.

Note that the iterative equations given so far reveal the ease of implementation of the CE subroutine: one only needs to keep $K+2$ buckets, indicating whether the kth increment falls into the jth band, $j = 1, 2, ..., K+2$, and aggregate the likelihood function for each bucket. The computational cost is of the same order as a vanilla SDIS iteration without the CE routine.

Remark 4.2. *One might consider further guiding the parametric family of samplers using large deviations ideas. For example, in the regularly varying case, one can force the probabilities to have the following structure,*

$$p_k = \frac{m-k+1}{m-k} p_{k-1},$$

for $k = 2, ..., M-1$, which is equivalent to $p_k = \frac{m-1}{m-k} p$, for $k = 1, 2, ..., m-1$. This choice reflects the intuition that the chance for the k-th increment to be a large one is roughly proportional to the inverse of the remaining steps to go. Note that this particular structure is very close to the optimal mixture found by [34] using a dynamic programming argument. However, due to the global dependence on the first probability parameter p. It is not difficult to see that the CE iteration equations will involve a root finding procedure, which could increase the computational cost significantly.

4.6 Numerical Examples

4.6.1 Example 1: Regularly Varying Increments

We illustrate the empirical performance of the SDIS with CE routine (SDIS-CE) by considering two examples. In the first example, the increments are regularly varying with index $\alpha = 1/2$, in particular, X_n's have tail distribution

$$\mathbb{P}(X_i > b) = (1+b)^{-1/2}.$$

Following [34], given the parameters of the model, a given number of increments m and a tail parameter b, we estimate $\mathbb{P}(S_m > b)$ and the standard deviation of the estimator as follows. We simulate 20000 replications of our estimator. The estimates are obtained based on averages of the replications. This is the output of a single run. Then we produce 500 independent runs. The results displayed are the averages of the outputs of these runs. We run the experiments with two different sets of input mixing probabilities. In the first case, which we shall later refer to as the "standard choice", we consider the heuristic choice $p_k = \theta/(m-k)$ where $\theta = 0.9$. And for the second set of input we use the optimal choice of the probabilities obtained by [34], i.e.,

$$p_k^* = \frac{a^{-\alpha/2}}{(m-k)a^{-\alpha/2}+1}, \tag{4.34}$$

which we call the "DLW" selection. In both cases we select $a = 0.9$. The results of the experiment are reported in the Table 4.1 and Table 4.2.

From the results of Table 4.1 we observe that even for a reasonable choice of mixing probabilities based on large deviations intuition, the CE algorithm produces a smaller relative error. On the other hand, it is outperformed by the optimal choice of the prob-

Table 4.1: Performance of the SDIS-CE estimator compared to the SDIS algorithm without CE procedure where the input mixing probabilities are set to be $p_k = 0.9/(m-k)$ for $k = 1, 2, ..., m-1$.

m	b	Standard	CE	Method
4	1e + 06	3.999E-03	4.000E-03	Average Estimate
		3.148E-05	1.395E-05	Average Std. Error
		0.787%	0.349%	Avg.SE/Avg.Est (%)
	1e + 12	3.999E-06	4.000E-06	
		3.151E-08	1.403E-08	
		0.788%	0.351%	
	1e + 18	4.000E-09	4.000E-09	
		3.153E-11	1.393E-11	
		0.788%	0.348%	
25	1e + 06	2.503E-02	2.498E-02	
		1.525E-03	3.404E-04	
		6.094%	1.363%	
	1e + 12	2.496E-05	2.499E-05	
		1.518E-06	3.458E-07	
		6.082%	1.384%	
	1e + 18	2.496E-08	2.502E-08	
		1.524E-09	3.409E-10	
		6.103%	1.363%	

Table 4.2: Performance of the SDIS-CE estimator compared to the SDIS without CE procedure where the input mixing probabilities are set to be the optimal choice obtained in Dupuis, Leder and Wang (2006).

m	b	DLW	CE	Method
4	1e + 06	4.000E-03	4.000E-03	Average Estimate
		5.660E-06	1.374E-05	Average Std. Error
		0.141%	0.344%	Avg.SE/Avg.Est (%)
	1e + 12	4.000E-06	4.000E-06	
		5.683E-09	1.382E-08	
		0.142%	0.346%	
	1e + 18	4.000E-09	4.001E-09	
		5.691E-12	1.373E-11	
		0.142%	0.343%	
25	1e + 06	2.499E-02	2.500E-02	
		3.925E-05	1.555E-04	
		0.157%	0.622%	
	1e + 12	2.500E-05	2.500E-05	
		4.032E-08	1.567E-07	
		0.161%	0.627%	
	1e + 18	2.500E-08	2.500E-08	
		4.027E-11	1.568E-10	
		0.161%	0.627%	

abilities obtained in [34], as can be seen in Table 4.2, one shall keep in mind, however, that in many applications, the structure of the problem doesn't allow for such analytical solutions easily. We also point out that the optimal solution from [34] hinges on the assumption that b is sufficiently large for large deviations asymptotics to be valid. For smaller exceedance level b, we might expect a better performance using the CE routine, which is underpinned by the results shown in Table 4.3.

Table 4.3: Comparison of performance between 1) SDIS using CE optimal mixing probabilities and 2) Analytical optimal mixing probabilities from Dupuis, Leder and Wang (2006), $m = 2$.

b	DLW	CE	Method
5	6.999E-01	6.999E-01	Average Estimate
	1.110E-03	5.742E-04	Average Std. Error
	0.159%	0.082%	Avg.SE/Avg.Est (%)
20	4.166E-01	4.166E-01	
	4.727E-04	4.410E-04	
	0.113%	0.106%	

We have mentioned in the previous section that since the recursive CE algorithm is carried out on the pilot sample, it neglects the fact that the increments are simulated in a sequential manner, but rather treats them in an independent way. We averaged the output CE optimal probability vector over the experiments, the near identical mixing probabilities in Table 4.4 is in line with the expected behavior of the method that each increment has probability at roughly 1/4 of causing the rare event.

Table 4.4: Average optimal CE .mixing probabilities, $m = 4,\ b = 10^6$.

k	1	2	3
p_k	0.248	0.253	0.251

4.6.2 Example 2: Weibull Increments

We now proceed to the second example where the increments are assumed to have the following Weibull-type of distribution,

$$\mathbb{P}\left(X > b\right) = e^{-2\sqrt{b+1}},$$

for $t \geq -1$. This corresponds to the case considered by [22], where the authors use a 5-point mixtures specified by the cut-off points $c_0 = 0.1\sqrt{b-s}, c_1 = 0.1(b-s), c_2 = 0.5(b-s), c_3 = 0.9(b-s)$ and $c_4 = b - s - 0.1\sqrt{b-s}$. Since the number of cut-off points increases from the previous mixture sampler, we increase the pilot sample number to 5000; all the other algorithmic parameters (number of runs and number of replications per run) remain the same. The results of the experiments are summarized in Table 4.5.

Table 4.5: Performance of the SDIS-CE estimator compared to SDIS without CE procedurein the case of Weibull-type of increments, $m = 4$. We used $p_{k,j} = 1/(K+2)(m-k)$, for $j = 0, 1, ...K$ and $k = 1, 2, ..., m-1$ as the "standard" choice of the mixing probabilities.

b	Standard	CE	Method
150	7.977E-11	7.966E-11	Avg. Est.
	2.580E-12	7.642E-13	Avg. Std. Err.
	3.235%	0.959%	Avg. SE/Avg. Est. (%)
450	1.371E-18	1.372E-18	
	4.835E-20	1.071E-20	
	3.526%	0.781%	
750	6.086E-24	6.069E-24	
	2.209E-25	3.185E-26	
	3.630%	0.525%	

By failing to prepare, you are preparing to fail.

Benjamin Franklin

5

Stochastic Insurance-Reinsurance Networks: Modeling, Analysis and Efficient Monte Carlo

The financial crisis has been plaguing the world since its outburst in 2007. Since then, there has been extensive discussions on the significance of systemic risk within the financial system. And a vast amount of research has been devoted to this field. In the

modeling stream along this line of research, it remains particularly challenging to develop a *dynamic* model that encompasses *stylized features* on conventions such as contractual structure, network connectivity, payment / default settlement and netting mechanism, while still maintaining a comfortable level of analytical tractability. Simulation turns out to be a natural choice. Nevertheless, as the level of complexity of the model increases, it may not even be clear a posteriori how simulation techniques can be properly engineered to analyze some particular performance measures to gauge the level of systemic risks in the network under consideration. In this chapter we aim to provide a framework to *blend modeling and analysis (via simulation)* of risk networks in the financial world. We base our development particularly on an insurance / reinsurance application.

5.1 Motivations and Goals

We develop efficient simulation methodology for risk assessment in the context of multiple insurance and / or financial entities with correlated exposures to each others risks and to systematic market factors. We also introduce a modeling framework for insurance / reinsurance networks that evolves according to equilibrium settlements at the time of default of companies. These settlements are computed as the solution of an associated linear program at each time period. Our types of models are closely related to and, in fact, inspired by network models that have been analyzed in the literature in recent years, for example [29], [30], [3], [40] and [65], to name a few.

Our interest lies in efficiently computing the conditional expected amount of the losses in the entire system, given the failure of a selected set of market participants. We say a *market or system dislocation* occurs when a specific group of participants fails. Using our results and simulation procedures we aim at characterizing the features that dictate a significant change in the nature of the system's exposures given market dislocation. For

instance, if a specific set of market participants is not sufficiently capitalized to fulfill their obligations, what is the most likely reason for such a situation, a systemic shock in the market or a sequence of specific idiosyncratic events pertaining to the specific set of participants?

Because of the various levels of dependence present in our model, and the structure of rare-events of interest (involving several companies defaulting) it turns out that the design of efficient simulation procedures for rare events in our setting typically involves more than one jump, whereas most of the rare-event simulation literature dealing with heavy tailed models involves single-jump events. The challenge in this situation lies in the fact that we are conditioning on rare events (involving several market participants) whose occurrence could most likely be caused by several large jumps. Also, as it will become clear given the integer programming formulation that we provide in Theorem 5.5, obtaining the large deviations behavior involves dealing with a combinatorial problem.

Our goal is to provide a simulation framework that can be rigorously shown to achieve *strong optimality* properties (in terms of designing estimators with bounded coefficient of variation uniformly as the event of interest becomes increasingly rare), and yet it is simple to implement in practice. Our contributions can therefore be summarized as follows:

a) We propose a dynamic network model that allows to deal with counterparty default risks with a particular aim of capturing cascading losses at the time of company defaults by means of the solution of a *linear programming problem* that can be interpreted in terms of an equilibrium. This formulation allows us to define the evolution of reserve processes in the network throughout time, see Theorem 5.2 and Theorem 5.4.

b) The linear programming formulation and therefore the associated equilibrium of settlements at the time of default recognizes: 1) the correlations among the risk

factors, which are assumed to follow a linear factor model, 2) the contractual obligations among the companies, which are assumed to follow popular contracts in the insurance industry (such as stop-loss and proportional reinsurance retrocesion), and 3) the interconnectedness of the network. The equilibrium approach we adopted (see (5.5)) turns out to be closely related to the market clearing framework established in [40], see Subsection 5.2.3. Our approach, however, permits reinsurance companies to net against each other's losses in the wake of default.

c) Our model allows to obtain asymptotic results and a description of the asymptotic most likely way in which the default of a specific group of participants can occur. This description indicated is fleshed out explicitly, by means of an *integer programming problem* (a Knapsack problem with multiple knapsacks). Such a description emphasizes the impact of the interactions between the severity of the exogenous claims, their dependence structure, and the interconnectedness of the companies on the systemic risk landscape of the entire network under consideration, see Theorem 5.5 and Theorem 5.6 and Proposition 5.1.

d) We propose a class of strongly efficient estimators for computing the expected loss of the network at the time of dislocation conditioning on the event that a specific set of market participants fails to meet their obligations. In addition, these estimators allow to compute associated conditional distributions of the network exposures given the dislocation of a set of specific players. The estimation of these conditional distributions is performed with a computational cost (as measured by the number of simulation replications) that remains bounded even if the event of interest becomes increasingly rare, see Theorem 5.7.

We are aware of only a limited amount of research that provides a risk analytical framework in an integrated insurance-reinsurance market with heavy-tailed risks. The

work of [68] considers a simple two-node insurance-reinsurance network involving light-tailed claims. Our work, however, puts into consideration a more complex and general network that captures more stylized features of the insurance market in practice. This is also the first work to the best of our knowledge that constructs provably efficient estimators in the setting of heavy-tailed risk networks. We have formulated our results in terms of regularly varying distributions for simplicity. Deriving logarithmic asymptotics with basically the same qualitative conclusions under other types of tail distributions is straightforward (see e.g., [21]). Our asymptotic results are obtained with the intention of gaining qualitative insight in the form of approximations that are correct up to a constant in the regularly varying setting. The role of the simulation algorithms, then, is to endow these asymptotic approximations with a computational device that allows one to efficiently obtain quantitatively accurate results. Thus, the entire approach we use, namely analysis and efficient computation, must be thought as a coherent contribution.

Now, as the connections in the network increase, one must account for all possibilities in which failure can occur. We have aimed at laying out a program to obtain estimators that have uniform relative error, for a fixed network architecture, as the probability of a failure event becomes more and more rare. At the same time, we have settled for estimators that are relatively easy to implement with the indicated performance guarantee. When the networks have more connections, the relative variance (even though uniformly bounded as rare events of interest become more and more rare) could grow. The question of designing rare-event simulation algorithms in which both uniformity in the size of the network and the underlying large deviations parameter are ensured is certainly important but too open-ended at this point. We plan to investigate this avenue in future research.

We envision that our model and our computational approach, based on efficient simulation, can serve as a prototype for the analysis of other types of risk networks. The

philosophy behind our work is that in the presence of network risk models, the settlements and the evolution of the associated risk reserve processes should obey equilibrium constraints that dictate the cascading effect when default occurs. These constraints can effectively be modeled in terms of linear programs, which, coupled with a heavy-tailed linear factor model, allow to describe qualitatively the most likely way in which simultaneous defaults occur. Efficient simulation, in the form of provably efficient Monte Carlo estimators, should then be used to make more precise quantitative statements.

The rest of the chapter is organized as follows. In Section 5.2 we describe in detail our network model and discuss the associated linear programming formulation for the evolution of contract settlements in the event of company failures. The asymptotic analysis of the model is given in Section 5.3. In Section 5.4 we propose a dynamic simulation scheme that balances practicality and efficiency, accompanied by a rigorous efficiency analysis at the end of the section. Numerical experiments are given in Section 5.5 on a test network under various configurations and target sets. We also include in Section 5.6 the proofs of several useful results in our development.

5.2 The Network Model and Its Properties

In this section we provide a precise description of the model in light of the insurance setting. Specifically, we consider an insurance market with two types of companies:

1. *Insurance companies or Insurers* whose core business involves underwriting insurance policies and thereby providing protection to policy-holders. In turn, they receive premiums upfront from policy holders as a source of funding.

2. *Reinsurance companies or Reinsurers*, acting as "insurers of insurers", primarily sell reinsurance contracts to insurance companies, in exchange for collections of

reinsurance premiums to get funded.

In order to cover typical features of an insurance market with these two sets of participants, the model is set up to allow reasonable generalities regarding

1) *contractual specifications*, which include types of contracts traded among the participants, correlation structure among the contracts, and specific dynamics of the stochastic models governing the profit and loss from these contracts;

2) *network topology / architecture*, which specifies how the participants are connected to each other, and rules of how such connections are changed in time;

3) *settlement / clearing mechanisms*, which stipulate how the participants make / receive payments from their contracts, as well as how company defaults are settled.

We refer to the class of networks covered by our model as $\mathcal{N}_e$. Specifications covering feature 1) and 2) above will be introduced in Subsection 5.2.1 and Subsection 5.2.4; and a detailed description of the settlement mechanisms is provided in Subsection 5.2.2.

5.2.1 Contractual Specifications and Network Topology

Let us denote by $\mathcal{I} = \{1, 2, \ldots, K_I\}$ and $\mathcal{R} = \{1, 2, \ldots, K_R\}$, the set of vertices in $\mathcal{N}_e$ representing the insurance and reinsurance companies in the market, respectively. The letters $\mathcal{I}$ and $\mathcal{R}$ are adopted for obvious mnemonic convenience. We then endow the following claim structure to this insurance network.

Claim arrival and heavy-tailed claim structure. We consider a slotted time model. Claims arrive to each player I_i, $i = 1, \ldots, K$ exogenously at time $n = 1, 2, \ldots$ according

to the following dynamics

$$\widetilde{N}_i(n) = B_1(n) + B_2(n) + \cdots + B_{\overline{N}_n}(n), \tag{5.1}$$

for $i \in \mathcal{I}$, where $B_j(n)$ is a Bernoulli random variable for the j-th claim at the n-th period with success parameter $q_n > 0$. Here $\overline{N}_n$ is a fixed positive number representing the maximum number of claims at period n. In other words, the number of total claims, $\widetilde{N}_i(n)$, collected by I_i at time n follows a $Binomial\left(\overline{N}_n, q_n\right)$. We must ensure that $\mathbb{E}z^{\widetilde{N}_i(n)} < \infty$ for some $z > 1$. The correlation structure among the $B_j(n)$'s can actually be made arbitrary. We shall study the system during time periods $n \in \{1, 2, \ldots, M\}$ for $M < \infty$. Note that the methodology and results developed here can be extended immediately to finite-state Markov modulation.

We assume that claim sizes adopt a linear factor model with heavy-tailed structure. Let $V_{i,j}(n)$ be the size of the j-th claim that I_i receives during the n-th period, its structure is specified as follows:

$$V_{i,j}(n) = \sum_{h=1}^{d} \gamma_{i,h} Z_h(n) + \beta_i Y_{i,j}(n), \tag{5.2}$$

Here $\{Z_h\}_{h \le d}$ is a series of common factors, introducing dependence among the claims. In particular, I_i is exposed to Z_h if the factor loading, $\gamma_{i,h}$, is positive. In other words, we allow each claim that arrive exogenously to the insurance companies to be exposed to *multiple* common risks, each of them possibly affecting different groups of insurers in the network. The set of common factors $\{Z_h\}$ quantifies the "sectoral risk" that is shared by a subset of insurance companies in the network. For example, geographic risk in catastrophic insurance, demographic risk in life insurance, etc. On the other hand, $Y_{i,j}(n)$ is the factor individual to the i-th insurance participant and is independent of all

the common factors Z_h, $h \leq d$. And β_i is the factor loading of I_i associated with $Y_{i,j}$. Both the factors and the loadings are non-negative.

Factors are assumed to have heavy tails. In particular, they belong in the class of regularly varying distributions (see Definition 1.7 in Subsection 1.2.2). Specifically, we assume

$$Z_h(n) \in \mathcal{RV}(-\alpha_h^Z),\ Y_{i,j}(n) \in \mathcal{RV}(-\alpha_i).$$

The regularly varying class requires the random variable to basically possess polynomial decaying tails, and it encapsulates a number of practical distributions, including the well-known Pareto and t-distributions. Since we will be dealing with Pareto quite often throughout the chapter, we give the following formal definition. A random variable X is said to have Pareto distribution, $X \sim Pareto\,(\theta, \alpha)$, if

$$\mathbb{P}\,(X > x) = \left(\frac{\theta}{\theta + x}\right)^{\alpha},\ x > 0.$$

We also impose the following technical condition in case of identical regular variation indices:

Condition 5.1. *If two factors have the same regular variation indices, let* $\overline{F}_1$, $\overline{F}_2$ *be their tail distribution functions, respectively, then* $\lim_{t\to\infty} \overline{F}_1(t)/\overline{F}_2(t)$ *exists.*

Reserve and Premiums

Each company in $\mathcal{N}_e$ is funded by: 1) an initial reserve and 2) net premiums, defined as the difference between the total premiums collected and the total premiums paid out, if any, at each period. Denote the initial reserves for I_i and R_s by $u_i(0)$ and $u_s^R(0)$, respectively. Let C_i and q_i be the aggregate periodic insurance premiums received and reinsurance premiums paid by I_i, $i \in \mathcal{I}$. Therefore the net premium obtained by I_i at each time is given by $\overline{C}_i = C_i - q_i$. Furthermore, let Q_s be the aggregate premiums

collected from its reinsurance policy holders at each period, $s \in \mathcal{R}$. $u_i(0)$ and $u_s^R(0)$ along with the premiums $\overline{C}_i$ and q_i, constitute the capital base of the (re)insurance companies to fulfill their obligations. Let us further denote by $u_i(n)$ and $u_s^R(n)$ the level of reserve for I_i, $i \in \mathcal{I}$ and R_s, $s \in \mathcal{R}$, respectively, at the end of period n. If the reserves $u_i(n)$ or $u_s^R(n)$ is not sufficiently large to cover all the claims collected, then the company is forced to fail. Precise definitions of $\{u_i(n)\}_{i\in\mathcal{I}}$ and $\{u_s^R(n)\}_{s\in\mathcal{R}}$ will be given in (5.17) later in Subsection 5.2.4.

Contractual Links and Network Topology

Naturally, the effective claims received by the companies are contingent on the survival of its counterparty, which in turn is influenced by how the participants deal with each other in the network. It is therefore crucial to first set the rules that govern the connectivity of the network, which is summarized in the following assumption.

Assumption 5.1 (Contractual Links and Network Topology for $\mathcal{N}_e$)**.**

i) ***Insurer-Reinsurer:*** *Each insurer I_i enters into "quater-share" reinsurance contracts with more than one standing reinsurers. The proportion it reinsured with R_s, and therefore the contractual link between I_i and R_s, is summarized by the nonnegative vector $\{\omega_{i,s}\}_{i\in\mathcal{I},s\in\mathcal{R}}$, with $\sum_{s\in\mathcal{R}}\omega_{i,s} = 1$, $\forall i \in \mathcal{I}$. Each reinsurance contract between I_i and R_s is assumed to be of a stop-loss type, with a reinsurance deductible equal to $\overline{v}_i^s$. If $\omega_{i,s} > 0$, there is a directed edge from I_i to R_s in the graph representing a contractual presence in the network, highlighting the business link between these two companies.*

ii) ***Reinsurance re-routing:*** *If one or some of the multiple reinsurance counterparties of insurer I_i fails at some time n, the vector $\{\omega_{i,s}\}$ is re-weighted proportionally among the survival reinsurance counterparties of I_i after time n. And the edges are*

re-directed reflecting the renewed contractual links. If, however, all of I_i's reinsurance companies have failed, then I_i will remain exposed to the claim risks until the end of the time horizon $M < \infty$.

iii) ***Reinsurer-Reinsurer:*** *Each reinsurer $R_s, s \in \mathcal{R}$, cannot reinsure the exposure transferred from one reinsurer $R_{s_1}, s_1 \neq s$ to some other reinsurer $R_{s_2}, s_2 \neq s_1, s$ (i.e. there are only two 'hoops' in the reinsurance sequence). Moreover, R_s can only enter into a proportional reinsurance contract (retrocession) with other reinsurers, covering exposures that are directly transferred from the insurers. The proportions of retrocession from reinsurer R_{s_1} to R_{s_2} is specified by the vector $\{\omega^R_{s_1,s_2}\}_{s_1,s_2\in\mathcal{R}}$, with $\omega^R_{s,s} = 1 - \sum_{s'\neq s}\omega^R_{s,s'}$. If $\omega^R_{s_1,s_2} > 0$, there is an edge from R_{s_1} leading to R_{s_2} in the network graph. And we further define*

$$\mathcal{P}_{i,s_1,s_2} = \omega_{i,s_1}\omega^R_{s_1,s_2}, \tag{5.3}$$

the weight of the reinsurance connection between I_i and R_{s_2} via R_{s_1}.

iv) ***Network Coverage:*** *For each $s \in \mathcal{R}$, define*

$$\overline{inV}(R_s) \triangleq \big\{i \in \mathcal{I} : \omega_{i,s} > 0\big\} \cup \big\{s' \in \mathcal{R} : \omega_{s',s} > 0\big\}, \tag{5.4}$$

i.e., the vertices that have an incoming edge or arc from node R_s. We assume that

$$\bigcup_{s\in\mathcal{R}} \overline{inV}(R_s) = \mathcal{I}.$$

We need to point out that the results obtained in this chapter hold in greater generality than in the networks with activities stipulated by Assumptions 5.1-i) and iii), which are mainly made to facilitate the definitions of the proportions that are transferred back in

the event of failures of the participants; these quantities, to be defined momentarily, are denoted by ρ_{si} and $\widetilde{\rho}_{ss'}$. The motivation of Assumption 5.1-ii) is that, each insurance company has its own specialty and risk-profile, meanwhile each reinsurance company specializes in different domains of reinsurance coverage. The assumption describes an insurance market in which each insurer I_i has fixed preferences, as measured by the vector $\{\omega_{i,s}\}_{s\in\mathcal{R}}$, over the reinsurance providers that underwrite reinsurance contracts on the particular type of risks I_i wishes to hedge against. The reinsurers are willing and are allowed to exchange risks among each other in the form of a proportional insurance contracts that are tailored to their own risk preferences. Note also that Assumption 5.1-iv) is a very mild one. We are only interested in a group of reinsurance companies along with the group of insurance companies they cover.

An example of such a network is illustrated in Figure 5.1 below. Let $N_{e_1} \in \mathcal{N}_e$ be the particular network given in the figure. Note that in N_{e_1} *multiple* reinsurers share the reinsurance liabilities from the insurers, and successive reinsurance and retrocession transactions among the reinsurance companies creates a so-called *reinsurance-spiral* in the network, which could be a source of systemic risk hibernating therein (see [62] and [1]). It is important to emphasize that the assumptions stated above, permits the formulation of such a reinsurance spiral. However, the risk re-sharing activity is strictly regulated by Assumption 5.1-iii). The rule basically forbids the reinsurer to cede reinsurance coverage back to the reinsurance companies which initially seek protection on that particular coverage. Again the stipulation of no more than two 'hoops' in the retrocession sequence is imposed merely for the sake of expositional simplicity (and only affects the definitions of ρ_{si} and $\widetilde{\rho}_{ss'}$ to be introduced shortly). In fact, as long as the reinsurance contract ends up with a party other than the one that buys protection at the first place, or equivalently if the "hoops" do not create a "loop", the framework introduced in this chapter works.

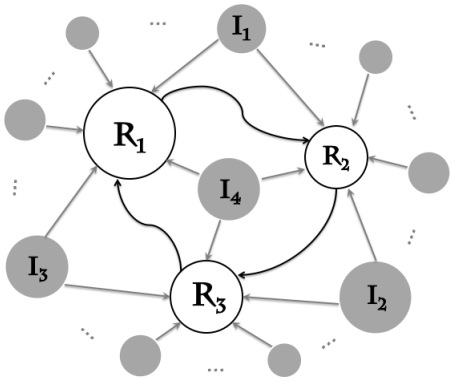


Figure 5.1: Network N_{e_1}. Each insurer enters into excess-of-loss reinsurance contracts with *multiple* reinsurers. A "reinsurance-spiral" among the reinsurance companies exists and is indicated by the "cycle" consisting of the curved lines.

5.2.2 Settlement Mechanism and Network Equilibrium

At the end of each period, each existing company in the network is faced with *the settlement of the claims* collected during the period. Due to the sophisticated contractual links among the companies, the state of the system at the end of period n is defined after a sequence of events that might involve a cascade of write-offs and settlements throughout the network at time n. In order to cope with these situations, we define the *equilibrium state* of the network at each period as follows.

Definition 5.1. *We say a network $N_e \in \mathcal{N}_e$ is in equilibrium state at time n, $1 \leq n \leq M$, if no companies in N_e are left unsettled from the failures, if any, of other companies in N_e that occur at time n.*

Note that, depending on the methods of settlements as well as the structure of the contractual links among the companies, there may or may not exist an equilibrium state for a given network. In the following assumption we make it clear how each counterparty of a ruined company gets settled at the time of such failure. We shall argue momentarily that, if companies in a network operating under Assumption 5.1 negotiate an arrangement under

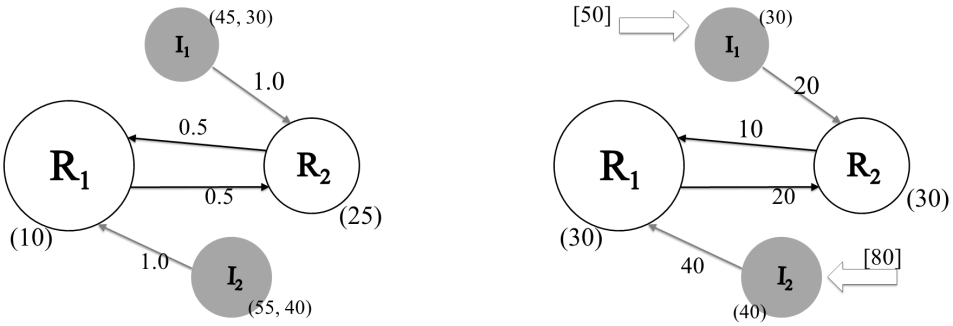
which the spillover loss at counterparty default (i.e., who gets how much) is distributed according to a reasonable mechanism (in the form of a linear program system), there exists a unique equilibrium state for the network at all times. We first specify the following assumption on the rules governing the allocation of spillover losses in the network system.

Assumption 5.2 (Rules for Spillover Loss Allocation)**.** *Upon the incident of R_s defaulting during period $n, n \leq M$, I_i gets partially settled by an amount proportional to its unsettled reinsurance exposure to R_s, if any, at period n; and $R_{s'}, s' \neq s$, gets settled by an amount proportional to its unsettled retrocession exposure to R_s, if any, at time n.*

In what follows, we shall denote by ρ_{si} the proportion of spillover loss that I_i gets if R_s fails, $i \in \mathcal{I}, s \in \mathcal{R}$, and similarly, denote by $\widetilde{\rho}_{ss'}$ the proportion that R'_s takes on in the event of the failure of R_s, $s, s' \in \mathcal{R}, s \neq s'$. Both ρ_{si} and $\widetilde{\rho}_{ss'}$ depend on the claims arriving to the network at the particular period when the failure of R_s occurs. We shall give the formal definitions shortly in (5.16). For now, we contend ourselves with the fact that both sets of proportions can be computed as soon as all the claims to the network system within a given period have been collected.

Nevertheless, having Assumption 5.2 alone turns out to be inadequate to secure a well-defined settlement mechanism in the event of a cascade of failures. Let us take a closer look using the following example.

Example 1. *Consider the simple network illustrated in Figure 5.2. Right after the claims have been collected, reinsurer R_1 does not have sufficient reserve base to buffer the size of the claims arrived at that period. A write-off procedure is therefore triggered. According to Assumption 5.2, R_2 will get an amount of the spillover loss from R_1 equal to $(10 - 30) \times (1/3) = -20/3$. With this allocation of contagion loss, R_2 is subsequently forced to fail because $25 - 20 - 20/3 = -5/3 < 0$. But we immediately ran into a dilemma if the recurrent spillover loss from R_2 is to be allocated to I_1 and R_1: should R_1, a bankrupt*



(a) Network Example: Initial Configurations (b) Network Example: Before Write-offs

Figure 5.2: (a): For each reinsurer the initial reserve levels are stated in the parentheses. For each insurer, the initial reserve as well as the reinsurance deductible are given in the parentheses next to the company. Transfer ratios are given next to the arrow representing the flow of contracts. (b): State of the network after all claims have been collected, before the write-offs. Bracketed numbers are the sizes of the claims. Numbers in parentheses are effective claims to the companies. And the rest is the transferred amount.

company, take on the spillover loss from R_2? If we allow this process to iterate by arguing that any failure/bankruptcy shall not be declared until all the subsequent cascading write-offs are settled, then a more precise write-off mechanism is called for to ensure a unique network state after all the contagion losses have been settled and received.

In order to address the afore-mentioned issue, we take an *equilibrium approach*. In particular, we require that, in addition to the principle stipulated in Assumption 5.2, the companies work out the spillover loss allocation at the end of each period according to the following single-period linear optimization problem, which we proceed to formulate now and interpret after we summarize that the equilibrium is well defined.

To streamline notations, let us suppress the time index and denote by u_i and u_s^R the levels of reserves at the beginning of the period for $I_i, i \in \mathcal{I}$ and $R_s, s \in \mathcal{R}$, respectively. Moreover, let L_i be the effective claims, net the reinsured amount before any settlement, retained by I_i. Similarly, let L_s^R be the effective reinsurance claims transferred to R_s

before any settlement. The mathematical definitions of L_i and L_s^R are provided later in (5.15). Note that both L_i and L_s^R are obtained after all claims at that period have been collected, but before any write-off/settlement has occurred. Define $\mathcal{I}^+ = \{l \in \mathcal{I} : u_l > 0\}$ and $\mathcal{R}^+ = \{v \in \mathcal{R} : u_v^R > 0\}$, the set of survival insurers and reinsurers, respectively. An equilibrium state for $\mathcal{N}_e$ corresponds to the state of the network after all companies mark write-offs and make settlements according to the optimal solution vector of the following linear optimization problem:

$$
\begin{aligned}
& [P^{(\kappa)}]: && (5.5)\\
\min \quad & \sum_{i\in\mathcal{I}^+} \pi_i^- + \xi \sum_{s\in\mathcal{R}^+} \psi_s^- \\
\text{s.t.} \quad & \pi_i^+ - \pi_i^- = u_i + \overline{C}_i - L_i - \sum_{s\in\mathcal{R}^+} \psi_s^- \cdot \rho_{si},\ \forall i \in \mathcal{I}^+ && (I)\\
& \psi_s^+ - \psi_s^- = u_s^R + Q_s - L_s^R - \sum_{s'\in\mathcal{R}^+, s'\neq s} \left(\psi_{s'}^- \cdot \widetilde{\rho}_{s's} - \kappa\psi_s^- \cdot \widetilde{\rho}_{ss'}\right),\ \forall s \in \mathcal{R}^+ && (II)\\
& \pi_i^+, \pi_i^-, \psi_s^+, \psi_s^- \geq 0.
\end{aligned}
$$

Here $\kappa \in [0, 1]$ is a parameter controlling the degree of *netting agreement* between each two reinsurance companies. When $\kappa = 0$, none of the contracts between two reinsurers are netted. And $\kappa = 1$ corresponds to a *fully netted* scenario, for example, when all the contracts between two reinsurers are fungible/exchangeable. Of course the netting parameter κ can be made arc dependent, but for simplicity we consider the situation where κ is identical throughout the network. We shall interpret the linear program shortly after we state the following results, which indicate desirable "stability" properties of the equilibrium state of the network underscored by the preceding linear program. We delay the proofs until later in Section 5.6.

Theorem 5.2. *The linear program* $[P^{(\kappa)}]$*, given in (5.5), has the following properties:*

1) It admits a unique optimal solution for any $\kappa \in [0,1]$*. Moreover, at this optimal solution, exactly one element in each pair,* $\left(\pi_i^+, \pi_i^-\right)$*, is equal to zero, for each* $i \in \mathcal{I}^+$*; and exactly one element in each pair* (ψ_s^+, ψ_s^-)*, is equal to zero, for each* $s \in \mathcal{R}^+$*.*

2) Given $\kappa \in [0,1]$*, the optimal solution is insensitive to the choice of* $\xi > 0$*.*

The previous result reveals that, at optimality, constraints (I) and (II) in (5.5) correspond to the negative reserves of the insurance and reinsurance companies, respectively, after the potentially cascading write-offs have passed through the network at the end of each period. It turns out that the equilibrium determined by $[P^{(\kappa)}]$ is also optimal to an optimization problem with more general objective functions.

Corollary 5.3. *Let* $\pi^- = \left(\ldots, \pi_i^-, \ldots\right)$*,* $i \in \mathcal{I}^+$*, and* $\psi^- = (\ldots, \psi_s^-, \ldots)$*,* $s \in \mathcal{R}^+$*. Let* $f(\pi^-, \psi^-)$ *be a function that is differentiable and non-decreasing with respect to its variables. And define* $[P_f^{(\kappa)}]$ *be the set of optimization problems with objective function* $f(\pi^-, \psi^-)$ *and with constraints identical to the ones in* $[P^{(\kappa)}]$*. Then the* $[P^{(\kappa)}]$*-optimal solution is also* $[P_f^{(\kappa)}]$*-optimal.*

Note that any objective function f that satisfies the condition specified in the previous result can be interpreted as a measure of the *incremental system loss* at the end of that particular period. The property of stabile optimality suggests that, the equilibrium state found by solving $[P^{(\kappa)}]$ is the best settlement solution to the system, as long as the companies in the network negotiate to minimize any sensible measure, f, of the incremental system loss.

Let us denote the optimal solution pairs to $P^{(\kappa)}$ by $\{\check{\pi}_i^+, \check{\pi}_i^-\}_{i\in\mathcal{I}}$ and $\{\check{\psi}_s^+, \check{\psi}_s^-\}_{s\in\mathcal{R}}$. At optimality, if $\check{\psi}_s^- > 0$ and $\check{\psi}_s^+ = 0$, constraint (II) in $P^{(\kappa)}$ guarantees that R_s has failed. And constraint (I) ensures that each insurer I_i receives the contagion loss of amount equal to $\check{\psi}_s^- \cdot \rho_{si}$. If the capital base of I_i is solid enough to weather the total spillover loss from

the reinsurers (which is represented by the amount $\sum_{s\in\mathcal{R}} \check{\psi}_s^-$), i.e., $u_i+\overline{C}_i > L_i+\sum_{s\in\mathcal{R}} \check{\psi}_s^-$, then I_i will remain solvent, in which case $\check{\pi}_i^+ > 0 = \check{\pi}_i^-$. If otherwise, then I_i fails, in which case $\check{\pi}_i^+ = 0$ and $\check{\pi}_i^- > 0$. As a result, the vectors $\{\check{\pi}_i^-\}_{i\in\mathcal{I}}$ and $\{\check{\psi}_s^-\}_{s\in\mathcal{R}}$ represent the loss at default for I_i and R_s, respectively, at the equilibrium state of the network.

Note that the preceding optimization problem would yield the same optimal solution if we impose the additional constraint that $\pi_i^+ \times \pi_i^- = 0, \forall i \in \mathcal{I}^+$, and $\psi_s^+ \times \psi_s^- = 0, \forall s \in \mathcal{R}^+$. Therefore, we can interpret the equilibrium state associated with the optimal solution vector to $[P^{(\kappa)}]$ as the equilibrium state of the network in which the weighted total loss of the network is minimized at the optimal objective value, equal to $\sum_{i\in\mathcal{I}^+} \check{\pi}_i^- + \xi \sum_{s\in\mathcal{R}^+} \check{\psi}_s^-$.

Example 2 (Example 1 (Con'd))**.** *Consider again the network given in Figure 5.2. Let* $\xi = 1$.

1) If we set $\kappa = 0$*, i.e., no netting is allowed for the default losses, and each contract has to be honored, the optimal solution to* $[P^{(\kappa=0)}]$ *becomes*

$$\check{\psi}_1^- = 30, \check{\psi}_2^- = 15, \check{\pi}_1^+ = 10, \check{\pi}_2^- = 5. \tag{5.6}$$

Note that the associated equilibrium state corresponds to increasing the negative reserve levels for R_1 *and* R_2 *before the write-offs both by* 10*. Since no netting agreement is in force, the write-off process continues until the levels of unsettled claims for both companies have reached the equilibrium levels.*

2) If, however, we set $\kappa = 1$*, i.e., allow maximal netting, the optimal solution to* $[P^{(\kappa=1)}]$ *is given by*

$$\check{\psi}_1^- = 55/3, \check{\psi}_2^- = 20/3, \check{\pi}_1^+ = 115/9, \check{\pi}_2^+ = 25/9.$$

Note that the equilibrium levels of unsettled claims for R_1 and R_2 are both lower than their negative reserves after absorbing the "first-degree" spillover losses from each other, i.e., $55/3 < 20+5\times 2/3$, and $20/3 < 5+20\times 1/3$. Eventually, under full netting agreement, R_1 only needs to transfer an amount equal to $5/3 = 20-55/3 = 20/3-5$ of its losses to R_2, and there is no need to take on any further losses back from R_2.

5.2.3 Connections to the Eisenberg-Noe ([40]) Formulation

Note that the optimal solution to $[P^{(\kappa=0)}]$ can be alternatively obtained using the approach given in [40]. In this subsection we use the particular network studied in Example 1 to discuss the connections between these two formulations.

The target output of the formulation in [40] is a so-called optimal payment or "clearing" vector, $\mathbf{p}$ which summarizes the equilibrium amount *paid out* by the market participants. For the insurance-reinsurance network we study in this chapter, in particular, we can write $\mathbf{p} = \big(\dots, p_i, \dots, p_s^R \dots\big)$, $i \in \mathcal{I}, s \in \mathcal{R}$. According to [40], this clearing payment vector can be obtained as the optimal solution to a particular optimization problem.

In order to put our model into the framework of [40], we need to create an extra "fictitious" vertex in our network, representing the "external" insureds who directly buy protection from the insurers. Let us denote by this extra node vertex $\mathcal{E}$. In the language of [40], the insurance market (at any single period) is then fully characterized by specifying $(\Pi, \overline{\mathbf{p}}, \mathbf{u})$. In particular, $\mathbf{u}$ is the vector of initial endowments of the participants, $\overline{\mathbf{p}}$ is the vector of *aggregate* nominal exposures to the participants, and Π is a square liability matrix specifying the amount (in proportions) of obligations between any two participants in the system, in which the element Π_{ij} is the proportion of the total obligations to participant i that is owed to participant j. The *clearing payment vector* $\mathbf{p}$ (for the period) is then

shown to be the solution to the following optimization problem:

$$\begin{aligned} [P(\Pi, \overline{\mathbf{p}}, \mathbf{u}, f)]: \quad & \max\ f(\mathbf{p}) \\ & \text{s.t. } \mathbf{p} \leq \Pi^T \mathbf{p} + \mathbf{u} \\ & \mathbf{0} \leq \mathbf{p} \leq \overline{\mathbf{p}}. \end{aligned} \tag{5.7}$$

where the objective function $f(\mathbf{p})$ can be taken as any increasing function in $\mathbf{p}$ to guarantee a unique optimal solution.

Now we illustrate how the equilibrium state for the network considered in Example 1 is derived using the program $[P(\Pi, \overline{\mathbf{p}}, \mathbf{u}, f)]$ above, for the particular period depicted in Figure 5.2. We define the *pairwise exposure matrices*, E^+ and E^-. In particular, each entry of E^+, $E^+_{i,j}$, represents the nominal exposure from i to j, or the nominal amount that i is supposed to pay j; and each entry of E^-, $E^-_{i,j}$, identifies the amount that i is expected to receive from j. For the network as presented in Figure 5.2, we have

$$E^+ = \begin{array}{c} \\ I_1 \\ I_2 \\ R_1 \\ R_2 \\ \mathcal{E} \end{array} \begin{array}{c} \begin{array}{ccccc} I_1 & I_2 & R_1 & R_2 & \mathcal{E} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 50 \\ 0 & 0 & 0 & 0 & 80 \\ 0 & 40 & 0 & 10 & 0 \\ 20 & 0 & 20 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}, \tag{5.8}$$

and

$$E^- = \begin{array}{c} \\ I_1 \\ I_2 \\ R_1 \\ R_2 \\ \mathcal{E} \end{array} \begin{array}{c} \begin{array}{ccccc} I_1 & I_2 & R_1 & R_2 & \mathcal{E} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 20 & 0 \\ 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}. \tag{5.9}$$

The aggregate exposure vector $\overline{\mathbf{p}}$ is then obtained by aggregating the individual exposures summarized in E^+ and E^-, via

$$\overline{\mathbf{p}} = e^T \left(E^+ - E^- \right) = (50, 80, 30, 30, 0)^T . \tag{5.10}$$

Note that in [40], the information of aggregate exposure $\overline{\mathbf{p}}$ is sufficient to pin down the equilibrium payment vector. However, as we shall reveal shortly, in order to transform the equilibrium payment vector obtained from $[P(\Pi, \overline{\mathbf{p}}, \mathbf{u}, f)]$ to the equilibrium reserve level identified by $[P^{(\kappa=0)}]$, one needs to explicitly construct E^+ and E^-.

Meanwhile, it is not hard to write down Π and $\mathbf{u}$ as follows,

$$\Pi = \begin{array}{c} \\ I_1 \\ I_2 \\ R_1 \\ R_2 \\ \mathcal{E} \end{array} \begin{array}{c} \begin{array}{ccccc} I_1 & I_2 & R_1 & R_2 & \mathcal{E} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 4/5 & 0 & 1/5 & 0 \\ 1/2 & 0 & 1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}, \quad \mathbf{u} = \begin{pmatrix} 45 \\ 55 \\ 10 \\ 25 \\ 0 \end{pmatrix}.$$

Note that the vector $\mathbf{u}$ for the insurance market we study is just the initial reserve at the beginning of a period. If we simply let $f(\mathbf{p}) = \mathbf{e}^T\mathbf{p}$, then the program, (5.7), yields the

unique optimal solution equal to

$$\mathbf{p} = \left(p_1, p_2, p_1^R, p_2^R, 0\right) = (50, 75, 25, 30, 0)\,.$$

We now demonstrate how the associated equilibrium end-of-period reserves can be obtained from the preceding optimal payment vector, $\mathbf{p}$, and how they can be shown to match the unique optimal solution of the linear program $[P^{(\kappa=0)}]$ in (5.5). The first step is to further break down the payment to the pairwise level. In order to do this, let us denote by $\widetilde{p_{ij}^-}$ the specific equilibrium payment made from company i to company j, defined via

$$\widetilde{p_{ij}^-} = p_i \Pi_{ij}.$$

Equivalently, the associated pairwise payment matrix, $\widetilde{\mathbf{p}}^-$, can be obtained using the following matrix operation,

$$\widetilde{\mathbf{p}}^- = \left[\mathbf{p} \,|\mathbf{p} \,|\mathbf{p} \,|\mathbf{p} \,|\mathbf{p}\right] \circ \Pi, \tag{5.11}$$

where the notation $\circ$ denotes matrix component-wise multiplication (i.e., if A and B are matrices of the same dimension, then $(A \circ B)_{i,j} = A_{i,j} \times B_{i,j}$). Moreover, define

$$\widetilde{\mathbf{p}}^+ = \left(\widetilde{\mathbf{p}}^-\right)^T, \tag{5.12}$$

i.e., $\widetilde{p_{ji}^+}$ denotes the amount of payment *received* by j from i, and $\widetilde{p_{ji}^+} = \widetilde{p_{ij}^-}$. For the

particular network example we are studying the matrix $\widetilde{\mathbf{p}}^-$ is given by

$$\widetilde{\mathbf{p}}^- = \begin{array}{c} \\ I_1 \\ I_2 \\ R_1 \\ R_2 \\ \mathcal{E} \end{array} \begin{array}{c} \begin{array}{ccccc} I_1 & I_2 & R_1 & R_2 & \mathcal{E} \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 50 \\ 0 & 0 & 0 & 0 & 75 \\ 0 & 20 & 0 & 5 & 0 \\ 15 & 0 & 15 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array},$$

or equivalently the non-zero elements of $\widetilde{\mathbf{p}}^-$ are

$$\widetilde{p}^-_{R_2,I_1} = 15, \widetilde{p}^-_{R_1,I_2} = 20, \widetilde{p}^-_{R_1,R_2} = 5, \widetilde{p}^-_{R_2,R_1} = 15,$$
$$\widetilde{p}^-_{I_1,\mathcal{E}} = p_1 = 50, \widetilde{p}^-_{I_2,\mathcal{E}} = p_2 = 75.$$

In order to obtain the resulting reserve levels from these payments it is necessary to compare them with the individual nominal exposures given by the matrices $E+$ and E^-. Therefore, let us define

$$\mathbf{G} = \min\left(\widetilde{\mathbf{p}}^+ - E^+, \mathbf{0}\right) + \min\left(\widetilde{\mathbf{p}}^- - E^-, \mathbf{0}\right),$$

where the minimum is performed component-wise (i.e., $\min(A, B) = C$ where $C_{ij} = \min(A_{ij}, B_{ij})$), and $\widetilde{\mathbf{p}}^+, \widetilde{\mathbf{p}}^-, E^+$ and E^- are given in (5.12), (5.11), (5.8) and (5.9). In other words, $\mathbf{G}$ summaries the *negative loss on each directional exposure* between two participants.

Consequently the relation between the optimal solutions to $[P(\Pi, \overline{\mathbf{p}}, \mathbf{u}, f)]$ and $[P^{(\kappa=0)}]$

is established via

$$\begin{aligned}\left(\pi_1^-, \pi_2^-, \psi_1^-, \psi_2^-\right)^T &= -\left(e^T\mathbf{G}\right)_{\{-\mathcal{E}\}} = (0, 5, 30, 15)^T \\ \left(\pi_1^+, \pi_2^+, \psi_1^+, \psi_2^+\right)^T &= \left(\mathbf{u} - (\mathbf{I} - \Pi)^T \mathbf{p}\right)_{\{-\mathcal{E}\}} = (10, 0, 0, 0)^T, \end{aligned} \tag{5.13}$$

where the subscript $_{\{-\mathcal{E}\}}$ denotes the associated vector without the element corresponding to the "fictitious" vertex $\mathcal{E}$. In summary, 15 out of the 20 nominal reinsurance exposure from I_1 to R_2 is honored by R_2, but I_1 is financially solid enough to weather this situation and pays the insureds the 50 in full, and eventually it is only able to cover 75 out of the 80 claims it received. I_2 is not so lucky because the 20 payment it receives from R_1 is not sufficient to prevent itself from failure. R_1 and R_2 settle with each other with payments of amount equal to 15 and 5, respectively. Note that the reserve levels obtained from the preceding operations coincide with the equilibrium reserve levels output from the linear program $[P^{(\kappa=0)}]$, see (5.6).

We need to point out, however, that the advantage of using the LP formulation in (5.5) is manifold.

a) It allows us to incorporate netting of default losses in a flexible way, which is not captured in the approach developed in [40]. For example, the mutual payment between R_1 and R_2 in the previous example can be reduced if certain level of netting is enforced in the settlement of default losses. Scenario 2) in Example 2 illustrates the benefit of allowing netting to the whole system: I_1 no longer defaults in this scenario, and all claims submitted from the insureds are honored.

b) Moreover, the output of the linear optimization problem $[P^{(\kappa)}]$ are the end-of-the-period reserve levels, which turn out to be the direct inputs to our dynamic reserve processes, see Theorem 5.4 below. In contrast, although the approach in [40] yields

an equivalent equilibrium state of the network at each stage (in the case when $\kappa = 0$), a few extra steps of calculation is required to transform the payment vector to the vector of reserve levels, as illustrated in the development leading to (5.13).

c) Recall that our ultimate goal is to efficiently evaluate the conditional spillover loss at system dislocation using simulation. An additional benefit of our LP formulation lies in the fact that some natural intuition on the large deviations description of the system can be derived out of the setup of the optimization problem, which we shall turn to shortly in the next section. Consequently, we believe the equilibrium approach adopted here is better suited for this dynamic network system we proposed in this insurance setting.

5.2.4 Effective Claims and Reserve Processes

Now we are in a good position to fill the gap and specify the rest of the model. Let $X_{i,j}(n)$, $W_{i,j}(n)$ be the effective claim size of the j-th claim ($1 \le j \le \widetilde{N}_i(n)$) arrived to I_i which is reinsured by R_s at period n, and the amount reinsured for this particular claim, respectively. The two quantities are defined via

$$\begin{aligned} X_{i,j}(n) &= \sum_{s\in\mathcal{R}} \omega_{i,s} \left(\min\left(V_{i,j}(n), \overline{v}_i^s\right) I\left(\tau_{R_s} > n-1\right) + V_{i,j}(n) I\left(\tau_{R_s} \le n-1\right)\right), \\ W_{i,j}(n) &= V_{i,j}(n) - X_{i,j}(n) = \sum_{s\in\mathcal{R}} \omega_{i,s} \max\left(0, V_{i,j}(n) - \overline{v}_i^s\right) I\left(\tau_{R_s} > n-1\right) \\ &= \sum_{s\in\mathcal{R}} \omega_{i,s} W_{i,j}^s(n), \end{aligned} \tag{5.14}$$

where $W_{i,j}^s(n) \triangleq \omega_{i,s} \max\left(0, V_{i,j}(n) - \overline{v}_i^s\right) I\left(\tau_{R_s} > n-1\right)$, and $\overline{v}_i^s \cdot \omega_{i,s}$ represents the reinsurance deductibles between I_i and R_s, and τ_{R_s} is the first time at which the reserve of R_s are non-positive. Note that the cap $\overline{v}_i^s$ loses effect as soon as R_s fails. At the same

time, any claim with size exceeding the cap $\overline{v}_i^s \cdot \omega_{i,s}$ is covered by R_s. The effective claims for insurer I_i and reinsurer R_s during period n are therefore

$$\begin{aligned} L_i(n) &= \sum_{j=1}^{\widetilde{N}_i(n)} X_{i,j}(n), i \in \mathcal{I}, \\ L_s^R(n) &= \sum_{t\in\mathcal{R}} \sum_{v\in\mathcal{I}} \sum_{l=1}^{\widetilde{N}_v(n)} W_{v,l}^t(n) \mathcal{P}_{v,t,s}, \ s \in \mathcal{R}, \end{aligned} \tag{5.15}$$

where $\mathcal{P}_{v,t,s}$ is defined in (5.3).

Based on Assumption 5.2, the allocation ratios of spillover losses at time n, $\{\rho_{si}(n)\}$ and $\{\widetilde{\rho}_{ss'}(n)\}$ are defined via

$$\begin{aligned} \rho_{si}(n) &\triangleq \frac{\sum_{j=1}^{\widetilde{N}_i(n)} W_{i,j}^s(n) \mathcal{P}_{i,s,s}}{L_s^R(n)} = \frac{\sum_{j=1}^{\widetilde{N}_i(n)} W_{i,j}^s(n) \mathcal{P}_{i,s,s}}{\sum_{t\in\mathcal{R}} \sum_{v\in\mathcal{I}} \sum_{l=1}^{\widetilde{N}_v(n)} W_{v,l}^t(n) \mathcal{P}_{v,t,s}}, i \in \mathcal{I}, \\ \widetilde{\rho}_{ss'}(n) &\triangleq \frac{\sum_{v\in\mathcal{I}} \sum_{j=1}^{\widetilde{N}_v(n)} W_{v,j}^{s'}(n) \mathcal{P}_{v,s',s}}{L_s^R(n)} = \frac{\sum_{v\in\mathcal{I}} \sum_{j=1}^{\widetilde{N}_v(n)} W_{v,j}^{s'}(n) \mathcal{P}_{v,s',s}}{\sum_{t\in\mathcal{R}} \sum_{v\in\mathcal{I}} \sum_{l=1}^{\widetilde{N}_v(n)} W_{v,l}^t(n) \mathcal{P}_{v,t,s}}, s' \in \mathcal{R}, s' \neq s. \end{aligned} \tag{5.16}$$

Let us index the single-period linear program $[P^{(\kappa)}]$, defined in (5.5), by n, i.e., $[P^{(\kappa)}(n)]$ is set-up by replacing the constraints and objectives with their time-n counterparts. Then at the end of each period, the system reaches the equilibrium state associated with the unique optimal solution to $[P^{(\kappa)}(n)]$. And the end-of-period reserves are determined by the unique optimal solution vectors $\{\check{\pi}_i^+(n), \check{\pi}_i^-(n)\}_{i\in\mathcal{I}^+(n)}$ and $\{\check{\psi}_s^+(n), \check{\psi}_s^-(n)\}_{s\in\mathcal{R}^+(n)}$, via

$$\begin{aligned} u_i(n) &= \check{\pi}_i^+(n) + \check{\pi}_i^-(n), i \in \mathcal{I}^+(n), \\ u_s^R(n) &= \check{\psi}_s^+(n) + \check{\psi}_s^-(n), s \in \mathcal{R}^+(n). \end{aligned} \tag{5.17}$$

Note that $u_i(n) = u_s^R(n) = 0$ if $i \notin \mathcal{I}^+(n)$ and $s \notin \mathcal{R}^+(n)$. The following result is a direct

implication of Theorem 5.2.

Theorem 5.4. *The stochastic processes,* $\{u_i(n)\}_{0\le n\le M}$, $i \in \mathcal{I}$, *and* $\{u_s^R(n)\}_{0\le n\le M}$, $s \in \mathcal{R}$, *given in (5.17) are well-defined.*

5.2.5 Conditional Spillover Loss at System Dislocation

Motivated by the insurance applications discussed in the previous section, we shall study the performance measure *Conditional Spillover Loss at System Dislocation* which is in the form of a conditional expectation. In simple words, it is the expected loss in the entire system conditioning on the failure of a subset of the network constituents. Before giving the formal definition we proceed to introduce a few more necessary notations.

Let A_I and A_R be subsets of $\mathcal{I}$ and $\mathcal{R}$, respectively; and set $A = A_I \cup A_R$. We define the following failure times associated with $\mathcal{N}_e$:

$$\begin{aligned}
\tau_i &= \inf\{n > 0 : u_i(n) \le 0\},\ i \in \mathcal{I},\\
\tau_{R_s} &= \inf\{n > 0 : u_s^R(n) \le 0\},\ s \in \mathcal{R},\\
\tau_{A_I} &= \max_{i\in A_I} \tau_i,\ \tau_{A_R} = \max_{s\in A_R} \tau_s,\\
\tau_A &= \tau_{A_I} \vee \tau_{A_R},
\end{aligned}$$

i.e., τ_A is the first time when all names in A have failed. Finally, if we define

$$D_i(A) \triangleq -\min\{u_i(\tau_A), 0\},$$

the *lost reserve at system dislocation* at time τ_A for I_i, we can therefore introduce the following formal definition of **Conditional Spillover Loss at System Dislocation**:

Definition 5.2. *The Conditional Spillover Loss at System Dislocation for the subset* $A = \{A_I \cup A_R\} \subseteq \{\mathcal{I} \cup \mathcal{R}\}$ *in time horizon* $[0, M]$ *is defined as*

$$CSD(A) = \mathbb{E}\left[\sum_{i\in\mathcal{I}} D_i(A)\Big|\tau_A \leq M\right]. \tag{5.18}$$

In words, the performance measure of the system, $CSD(A)$, measures the *contagion* (or *spillover*) impact of the collapse of the companies encoded by A to the entire system. The idea of such a measure is motivated by the so-called *Systemic Risk Index* or *Contagion Index*, following the terminology in [10], and studied in, for example [29] and [30]. The authors in [29] used a Cauchy copula to evaluate the Systemic Risk Index, which is also defined in terms a conditional expectation. Their simulation procedure does not necessarily meet any provable optimality property, and it appears to be suited to the case where conditioning event is the failure of *a single* player. Our work in this chapter aims to provide a provably efficient procedure that can capture multiple-jumps.

5.3 Asymptotic Description of the Network System

Having fixed the architecture of the network, we now embark on providing a qualitative characterization of the large deviations behavior of the system given $\{\tau_A \leq M\}$, i.e., the event of system dislocation caused by the set A occurring before the fixed horizon M. In the analysis that follows let us scale the initial reserves by b, and we later send b to infinity. Let $b > 0$ and assume that $u_i(0) = r_i b$ is the initial reserve for I_i, $i \in \mathcal{I}$, and let $u_s^R(0) = \widetilde{r}_s b, s \in \mathcal{R}$, where r_i and $\widetilde{r}_s$ are fixed positive constants. In what follows we will also make explicit the dependence of various model quantities on b.

Our plan is to first pin down the asymptotic description of the general network system portrayed in the previous section. As we shall reveal momentarily, this description can be

identified by solving another optimization problem. We then show that for some special network structure, a more in-depth characterization can in fact be obtained with care.

5.3.1 Large Deviations Description via An Integer Program

We shall demonstrate that the large deviations description for the network has a "multiple-regime" characterization. Depending on the tail structure of the claim size distributions, the failure of the system arises from different numbers of extreme shocks in the claims. This particular feature of the system inspires us to tailor a sequential algorithm for evaluating $CSD(A)$, for any given set A, which we shall describe in details in the next section.

It is interesting to realize that useful implications about the asymptotic behavior of the system can be obtained from the linear program $[P^{(\kappa)}]$ given in (5.5). To see this, recall that constraints (I) in (5.5) require, for each $i \in \mathcal{I}^+$ that,

$$\pi_i^+ - \pi_i^- = u_i + \overline{C}_i - L_i - \sum_{s \in \mathcal{R}^+} \psi_s^- \cdot \rho_{si}.$$

From the definitions in (5.15) and (5.14) as well as Assumption 5.1-ii), it's not hard to see that the effective claims L_i are capped from above if and only if all the reinsurance counterparties to I_i have not yet failed, and in that case $u_i + \overline{C}_i - L_i = \Theta_p(b)$, where the notation $\Theta_p(\cdot)$ is defined in Definition 1.2 in Subsection 1.2.1. Therefore, the intuition is that, $\mathbb{P}(\check{\pi}_i^- > 0) = \Theta(1)$ if and only if there exists $s \in \mathcal{R}^+$, such that *both* of the following are satisfied:

i) $\check{\psi}_s^- = \Theta_p(b)$,

ii) $\rho_{si} = \Theta_p(1)$.

In other words, both the default loss for R_s and the contractual link between I_i and

R_s need to be sufficiently large in order for I_i with $\Theta(1)$ probability. This can occur due to either of the following two possible cases:

a) $Z_h = \Theta(b)$, for some $1 \leq h \leq d$ such that $\gamma_{i,h} > 0$,

b) $Y_{i,j} = \Theta(b)$, for some $1 \leq j \leq N_i$.

The intuitions above are certainly helpful, for now we are able to restrict the enumeration of possible paths (leading to the event $\{\tau_A \leq M\}$) down to a much smaller subset. In fact, as we shall see shortly, the combinatorial task of singling out the cheapest route to the target event boils down to solving a *Knapsack problem with multiple constraints*.

Let us denote by Ξ the *factor exposure matrix* for the insurers in the network, which is an $|\mathcal{I}| \times (d + |\mathcal{I}|)$ matrix. Each column corresponds to a specific factor. We align the factors in such a way that the first d factors are the common factors, and the remaining $|\mathcal{I}|$ factors are the individual factors for the $|\mathcal{I}|$ insurers. Let Ξ_j^c be the j-th column of Ξ. In what follows we shall denote by U_j the factor, common or individual, corresponding to Ξ_j^c. On the other hand, the i-th row of Ξ, Ξ_i^r, represents the i-th insurance company. Define ν_{ij} to be the exposure of insurer I_i to factor U_j. In other words,

$$\nu_{ij} = \begin{cases} \gamma_{ij}, & \text{if } j \leq d \\ \beta_i, & \text{if } j = i + d, i \in \mathcal{I} \\ 0, & \text{otherwise.} \end{cases}$$

The entries of the matrix Ξ is therefore defined via

$$\Xi_{ij} = I\left(\nu_{ij} > 0\right). \tag{5.19}$$

Last but not least, define $\widetilde{\alpha}_j$ to be the regularly varying index of U_j, i.e., $\widetilde{\alpha}_j = \alpha_j^Z$ if $j \leq d$,

and $\widetilde{\alpha}_j = \alpha_i$ if $j = i + d$, $i \in \mathcal{I}$. The following result shows that, the large deviation description of the system is simply obtained by solving an *integer programming problem*, which is easily identified as a Knapsack type of problem with multiple knapsacks. We shall delay the proof of the theorem to the end of Section 5.4. We mention that a one dimensional Knakpsack formulation has also be used by [71] in the setting of heavy-tailed large deviations.

Theorem 5.5. *As $b \nearrow \infty$, we have*

$$\frac{\log \mathbb{P}\left(\tau_A(b) \leq M\right)}{\log b} \longrightarrow -\zeta, \tag{5.20}$$

where ζ is the optimal cost to the following integer programming problem:

$$\begin{aligned} [IP]: \quad \min \quad & \sum_{j=1}^{m} \widetilde{\alpha}_j x_j \\ s.t. \quad & \sum_{j=1}^{m} x_j \Xi_{i,j} \geq 1, \ \forall i \in A \\ & x_j \in \{0,1\}, \ 1 \leq j \leq m \end{aligned} \tag{5.21}$$

Remark 5.1. *For any $[IP]$-optimal solution $\mathbf{x}^* = (x_1^*, \ldots, x_m^*)^T$, x_j^* is interpreted as the "indicator of activation" which dictates the occurrence of a large factor U_j. In particular, if for fixed $i \in \mathcal{I}$, $x_{i+d}^* = 1$, then $Y_i = \Theta(b)$ in the large deviations description of the system; if $x_h^* = 1$, for some $h \leq d$, then $Z_h = \Theta(b)$ in the large deviations description of the system. For a survey of the algorithms to solve this Knapsack type of problems, we refer the readers to e.g. [54].*

There are several interesting features of this characterization.

1. The large deviations behavior of the network (conditioning on the event $\{\tau_A \leq M\}$)

is dictated only by a set of tail indices. Depending on the choice of A, the description of the most likely way leading to $\{\tau_A \leq M\}$ may change domains. For instance, the event $\{\tau_{A_1} \leq M\}$, where $A_1 = \{\text{AIG, Prudential}\}$, could most likely result from the occurrence of a few large common factors, while $\{\tau_{A_2} \leq M\}$, where $A_2 = \{\text{Lincoln Benefit, Northwestern Mutual}\}$, might occur most likely due to multiple phenomenal idiosyncrasies, or a mixture of extremal idiosyncratic and common shocks.

2. Local to each insurer I_i, large deviations is characterized by the so-called "single jump domain"; however on the network level, depending on the characteristics of the claim size distributions, the large deviations of the system might fall into the "multiple jump domain", in which more than one shocks are necessary for the rare event to occur.

An important albeit slightly counter-intuitive implication from Thereom 5.5 is that, the existence of the reinsurance companies does not alter the asymptotic description of the network system, in the sense that the most likely way leading to the failure of the subset A is identical to that of a network consisting stand-alone insurance companies that do not enter into any reinsurance contracts. We need to point out that this observation does not suggest the roles of the reinsurance companies as risk buffers are vulnerable and therefore flawed. Under market conditions in which moderately large claims arrive, the reinsurance companies function well as a centralized risk mitigator, and might successfully ward off the failure of some of its otherwise financially vulnerable insurance counterparties. Furthermore, we find this observation to be consistent with various empirical studies, which argue that reinsurance failure may not be a substantial source of systemic risk for the insurance industry, see for example [62], [1] and [69].

We could, however, further strengthen the roles of the reinsurance companies by enforcing a more stringent capital requirement for the reinsurers. In order to see this, let us

assume that

$$u_s^R(0) = \Theta\left(b^\rho\right), \rho > 1,$$

for all $s \in \mathcal{R}$, thereby demanding each reinsurer in the network to pledge more capital than the insurance companies (recall that $u_i(0) = \Theta(b)$ for $i \in \mathcal{I}$). The following result indicates that asymptotic description for the system with this modified assumption can be identified by solving a different integer programming problem.

Theorem 5.6. *Define*

$$\mathcal{R}(A) = \bigcup_{i \in A} \left\{ s \in \mathcal{R} : \sum_{r \in \mathcal{R}} \mathcal{P}_{i,r,s} > 0 \right\},$$

for $A \subseteq \mathcal{I}$, where $\mathcal{P}_{i,r,s}$ is defined in (5.3). In words, $\mathcal{R}(A)$ is the set of reinsurance counterparties of companies in A. Then we have, as $b \nearrow \infty$,

$$\frac{\log \mathbb{P}\left(\tau_A(b) \leq M\right)}{\log b} \longrightarrow -\widetilde{\zeta}(\rho) \tag{5.22}$$

where $\widetilde{\zeta}(\rho)$ is the optimal cost to the following integer programming problem:

$$\begin{aligned} [\widetilde{IP}^{(\rho)}]: \quad \min \quad & \sum_{j=1}^{m} \rho \widetilde{\alpha}_j x_j + \sum_{j=1}^{m} \widetilde{\alpha}_j y_j \\ s.t. \quad & \sum_{j=1}^{m} \Xi_{i,j} x_j \geq 1, \ \forall i \in \mathcal{R}(A) \\ & \sum_{j=1}^{m} \Xi_{l,j} \left(x_j + y_j\right) \geq 1, \ \forall l \in A \\ & x_j, y_j \in \{0,1\}, \ 1 \leq j \leq m \end{aligned} \tag{5.23}$$

We dispense ourselves with the formal proof of the result, which can be carried out in a similar fashion as the proof of Theorem 5.5. The basic intuition is that, since $u_s^R(0) =$

$\Theta\left(b^{\rho}\right)$, the corresponding spillover losses from reinsurer R_s, is of the same order, i.e., $\check{\psi}_s^- = \Theta\left(b^{\rho}\right)$ as a result of Lemma 5.2 given in the next subsection. Now for $i \in A$, as long as $\rho_{si} = o\left(b^{-(\rho-1)}\right)$, for all $s \in \mathcal{R}(i)$, $\mathbb{P}\left(\check{\pi}_i^- > 0\right) = o\left(1\right)$ and therefore I_i survives, with overwhelming probability, after all its counterparties have been brought down (by some other factors that I_i is not exposed to). From then on, it loses reinsurance protection and requires a factor of order $\Theta(b)$ to get ruined. If, however, the exposure between I_i and R_s, for some $s \in \mathcal{R}(i)$, is substantial enough such that $\rho_{si} = \Omega\left(b^{-(\rho-1)}\right)$, then I_i fails with overwhelming probability by the spillover loss passed on from the failure of R_s.

Remark 5.2. *In any* $[\widetilde{IP}^{(\rho)}]$*-optimal solution* $(\mathbf{x}^*, \mathbf{y}^*)$*,* x_j^* *and* y_j^* *are interpreted as the "strong" and "weak" activation indicators, respectively. If* $x_j^* = 1$*, then the corresponding factor* U_j *is among the factors that most likely lead to the failure of the counterparty set* $\mathcal{R}(A)$*, i.e.,* $U_j = \Theta\left(b^{\rho}\right)$*; if* $y_j^* = 1$*, then* U_j *is among the factors that result in the failure of some companies in* A *after they lost protections from their reinsurance counterparties, and in that case,* $U_j = \Theta\left(b\right)$*.*

5.3.2 Characterizing Asymptotic Behavior of A Special Network

The development in the previous subsection suggests that, for a general network defined in Section 5.2 one needs to explicitly solve the IP given by (5.21) to obtain an asymptotic description of the system. We shall demonstrate in this subsection that for some special network architecture, a more detailed characterization for the most likely way of the network hitting the event $\{\tau_A(b) \leq M\}$ is readily accessible, without even resorting to the optimization problem.

Consider an insurance-reinsurance network with *a single* reinsurance company, which we refer to as $R = R_1$. Let us write $K = K_I$, the number of insurers in the system. An

example of such a network is shown in Figure 5.3. Because the shape of such a network is in close resemblance of a star, in what follows we shall refer to it as the *star-shaped network*. Endowed with such a special structure, Assumption 5.1 can be greatly simplified. In particular, since there is only one reinsurer in business in the network, $\omega_{i,1} = 1$ and $\mathcal{P}_{i,1,1} = 1$, for all $i \in \mathcal{I}$. And there is apparently no retrocession activity in the star-shaped network. Furthermore, the reinsurance re-routing assumption becomes trivial: as soon as R fails, the remaining insurers no longer receive any reinsurance protection, and are subject to absorbing all potential claim risks from their policy holders.

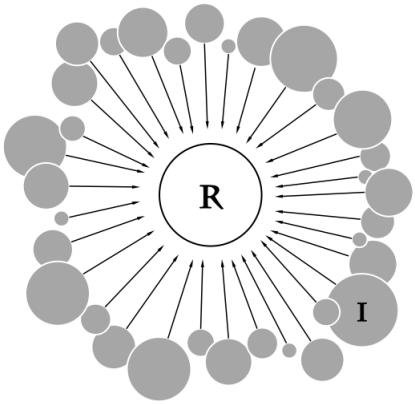


Figure 5.3: An example of a "star-shaped" network.

In addition to the star-shape topological simplification, the number of claims arrived to I_i at each time n is assumed to be *Poisson* with mean λ_i, i.e., $N_i(n) \sim Poisson\,(\lambda_i)$. And we further simplify the correlation structure among the claims by fixing the total number of common factors to be one, i.e., $d = 1$. Therefore under this specification, the exogenous claim size, V, the effective insurance claim size, X, and the effective reinsurance claim size, W, can be expressed in the following way:

$$\begin{aligned} V_{i,j}(n) &= \gamma_i Z(n) + \beta_i Y_{i,j}(n),\ 1 \le j \le N_i(n), \\ X_{i,j}(n) &= \min\left(V_{i,j}(n), \overline{v}_i\right) I\left(\tau_R > n-1\right) + V_{i,j}(n) I\left(\tau_R \le n-1\right), \end{aligned}$$

$$W_{i,j}(n) \;=\; V_{i,j}(n) - X_{i,j}(n),$$

for each $i \in \mathcal{I}$, $n \leq M < \infty$ and $1 \leq j \leq N_i(n)$. Here τ_R is the failure time of R to be defined shortly.

Note that for the star-shaped network, the equilibrium of the system and hence the payment / settlement to each company at each time is easily solved from the linear program in (5.5). In particular, let $\check{\psi}_1^-(n)$ be the optimal solution variable for $\psi_1^-(n)$ in (5.5), associated with the star-shaped network. It's not hard to convince ourselves that $\check{\psi}_1^-(n) = -\min\big(u(n), 0\big)$. Therefore we can express "feedback" allocation of unsettled claims from R to I_i at time n, denoted as Γ_i, defined via

$$\Gamma_i(n) = \check{\psi}_1^-(n) \cdot \rho_{1i} = -\min\big(u(n), 0\big) \times \frac{\sum_{j=1}^{N_i(n)} W_{i,j}(n)}{\sum_{l=1}^{K}\sum_{j=1}^{N_l(n)} W_{l,j}(n)}, \tag{5.24}$$

for $1 \leq n \leq M$. Let the initial reserve for R and I_i be $u(0) = rb$ and $u_i(0) = r_i b$, respectively, where $r, r_i > 0$ are some positive constants. We can therefore express the reserve processes for R and I_i, $i \in \mathcal{I}$, as

$$u(n) \;=\; u(n-1) + QI\left(\tau_R > n-1\right) - \sum_{i=1}^{K}\sum_{j=1}^{N_i(n)} W_{i,j}(n), \tag{5.25}$$

$$u_i(n) \;=\; u_i(n-1) + \overline{C}_i - \sum_{j=1}^{N_i(n)} X_{i,j}(n) - \Gamma_i(n), \tag{5.26}$$

for $1 \leq n \leq M$, where $Q = Q_1$ is the periodic reinsurance premiums R receives. Here the failure times τ_R and τ_i are formally defined as $\tau_R = \inf\{k > 0 : u(k) \leq 0\}$ and $\tau_i = \inf\{k > 0 : u_i(k) \leq 0\}$.

We now proceed to characterize the asymptotic behavior of the star-shaped network. Note first that, given the Poisson nature of the claim arrival process, the probability

$\mathbb{P}(\tau_A \leq M)$ is dominated by the probability of one or a few extremal claims. To see this, Note that

$$\begin{aligned}
\mathbb{P}(\tau_A(b) < M \wedge \tau_R(b)) &\leq \mathbb{P}(\tau_A(b) < \tau_R(b)) \\
&\leq \sum_{n=1}^{M} \mathbb{P}(u_i(n) < 0, \forall i \in A) \\
&= \sum_{n=1}^{M} \mathbb{P}\left(\bar{C}_i n - \sum_{k=1}^{n}\left(\sum_{j=1}^{N_i(k)} X_{i,j}(k)\right) + u_i(0) < 0, \forall i \in A\right) \\
&\leq \sum_{n=1}^{M} \prod_{i\in A} \mathbb{P}\left(\sum_{k=1}^{n} N_i(k)\overline{v}_i > \bar{C}_i n + u_i(0)\right) \\
&\leq \sum_{n=1}^{M} \prod_{i\in A} \mathbb{P}\left(\sum_{k=1}^{n} N_i(k) > \widehat{r} b\right), \qquad (5.27)
\end{aligned}$$

for some positive constant $\widehat{r}$ that depends only on the set A. In fact, we can pick for b large enough, $\widehat{r} = \min_{i\in A}\{r_i/(2\overline{v}_i)\}$. Hence the term $\mathbb{P}(\tau_A(b) < M \wedge \tau_R(b))$ decays at least exponentially in b. We can therefore conclude, with the aid of the following proposition, that

$$\mathbb{P}(\tau_A(b) \leq M) \sim \mathbb{P}(\tau_R(b) \leq \tau_A(b) \leq M) \qquad (5.28)$$

as $b \nearrow \infty$.

Proposition 5.1. *Let α and α_i be the indices of regularly variation for the single common factor and the i-th individual factor, respectively. Assume that the reserve levels are sufficiently large (i.e., b is large).*

(i) If

$$\alpha < \sum_{i\in A} \alpha_i, \qquad (5.29)$$

the event $\{\tau_A \leq M\}$ is caused with overwhelming probability (as $b \nearrow \infty$) by a large common factor.

(ii) If $\alpha > \sum_{i\in A}\alpha_i$, the event $\{\tau_A \leq M\}$ occurs with overwhelming probability (as $b \nearrow \infty$) in the following way: the occurrence of a single large individual factor from some insurer I_i in A first leads to the failure of R, after which insurers in A break down because of the occurrence of a series of additional individual factors, one from each of the insurers in $A\backslash\{i\}$.

(iii) If, however, $\alpha = \sum_{i\in A}\alpha_i$, the event $\{\tau_A \leq M\}$ can be caused, with probability bounded away from zero, either by the occurrence of a large common factor as in case (i), or by the sequence of events as described in case (ii) above.

In order to prove the proposition, we need the following results, the proofs of which are given in the Section 5.6.

Lemma 5.1. *Suppose $\{X_i\}_{i\geq 1}$ is a sequence of i.i.d. regularly varying random variables with index α; Z is regularly varying with index α_0 and is independent of the X_i's. And $N \sim Poisson(\lambda)$, independent of both Z and X_i's. Moreover, Condition 1 is in force for X_i and Z. Suppose further that $\psi : \mathbb{N} \to \mathbb{R}$ is a non-decreasing mapping which satisfies $\mathbb{E}\left[\psi(N)^{\alpha(1+\delta)}\right] < \infty$, for some $\delta > 0$. Then*

$$\mathbb{P}\left(\sum_{i=1}^{N} X_i + \psi(N)Z > b\right) \sim \mathbb{E}N\mathbb{P}\left(X_1 > b\right) + \mathbb{P}\left(Z > \frac{b}{\mathbb{E}\psi(N)}\right). \tag{5.30}$$

Lemma 5.2. *1) Suppose Z is a nonnegative regularly varying random variable with index $\alpha > 0$, and Y is a nonnegative random variable satisfying $\mathbb{E}\left[Y^{\alpha(1+2\epsilon)}\right] < \infty$ for some $\epsilon > 0$. Then*

$$\mathbb{P}\left(ZX > b + x | ZX > b\right) \longrightarrow \left(\frac{1}{1+x/b}\right)^{\alpha}.$$

2) Suppose X_i is nonnegative and regularly varying with index $\alpha_i > 0$, $i = 1, \ldots, K$. $X_{i,j}$ is the j-th independent copy of X_i. N_i is nonnegative random variable satisfying

$\mathbb{E}\left[N_i^{\alpha_i(1+2\epsilon')}\right] < \infty$ *for some* $\epsilon' > 0$. *And Condition 1 holds for* X_i *and* X_j, $i \neq j$. *Then*

$$\mathbb{P}\left(\sum_{i=1}^{K}\sum_{j=1}^{N_i} X_{i,j} > b + x \middle| \sum_{i=1}^{K}\sum_{j=1}^{N_i} X_{i,j} > b\right) \longrightarrow \left(\frac{1}{1+x/b}\right)^{\alpha_*},$$

where $\alpha_* = \min_{i=i}^{K} \alpha_i$.

Proof of Proposition 5.1. We shall study the probability $\mathbb{P}(\tau_R \leq \tau_A \leq M)$. Note that, if $\tau_R \leq M$, then there exist $1 \leq n \leq M$ and $1 \leq i \leq M$ such that

$$\max\left(\gamma_i N_i(n) Z_n, \sum_{j=1}^{N_i(n)} \beta_i Y_{i,j}(n)\right) + \sum_{k=1}^{n-1} N_i(k)\overline{v}_i > r_i b.$$

On the other hand, if there exist $1 \leq n \leq M$ and $1 \leq i \leq M$ such that

$$\max\left(\gamma_i N_i(n) Z_i, \sum_{j=1}^{N_i(n)} \beta_i Y_{i,j}(n)\right) > (r_i + r)\, b,$$

we would guarantee that $\tau_R \leq n \leq M$. Let $\delta \triangleq (r, \min_{i\in A} r_i) / (2KM)$, and define

$$\begin{aligned}
B_Z &= \left\{\exists n \leq M : \left(\sum_{i=1}^{K} \gamma_i N_i(n)\right) Z_n > K\delta b, \tau_A \geq \tau_R = n\right\}, \\
B_Y &= \left\{\exists n \leq M, i \leq K : \sum_{j=1}^{N_i(n)} \beta_i Y_{i,j}(n) > \delta b, \tau_A \geq \tau_R = n\right\} \\
&= \bigcup_{i\leq K}\left\{\exists n_i \leq M : \sum_{j=1}^{N_i(n_i)} \beta_i Y_{i,j}(n_i) > \delta b, \tau_A \geq \tau_R = n_i\right\} = \bigcup_{i\leq K} B_{Y,i},
\end{aligned}$$

where $B_{Y,i} \triangleq \left\{\exists n \leq M : \sum_{j=1}^{N_i(n)} \beta_i Y_{i,j}(n) > \delta b, \tau_A \geq \tau_R = n\right\}$, and the $B_{Y,i}$'s are disjoint

sets. Note that $\{\tau_R \leq \tau_A \leq M\} \subseteq B_Y \cup B_Z$. Further define the following probabilities:

$$p_Z = \mathbb{P}\left(\tau_R \leq \tau_A \leq M; B_Z\right) \text{ and } p_Y = \mathbb{P}\left(\tau_R \leq \tau_A \leq M; B_Y\right).$$

Note that

$$p_Z + p_Y - \mathbb{P}\left(B_Z \cap B_Y\right) \leq \mathbb{P}\left(\tau_R \leq \tau_A \leq M\right) \leq p_Z + p_Y.$$

And since $\mathbb{P}\left(B_Z \cap B_Y\right) = o\left(p_Z \vee p_Y\right)$, it suffices to compare p_Z and p_Y. The cases $p_Y = o\left(p_Z\right)$, $p_Z = o\left(p_Y\right)$ and $p_Z = \Theta\left(p_Y\right)$ correspond to case i), ii) and iii) in the proposition, respectively.

1) Analysis of p_Z.

From Lemma 5.2 we know

$$\left[\left(\sum_{i=1}^{K} \gamma_i N_i(n)\right) Z_n \middle| \left(\sum_{i=1}^{K} \gamma_i N_i(n)\right) Z_n > K\delta b\right] \sim \left(K\delta + K\delta W\right) b, \tag{5.31}$$

where $W \sim Pareto\left(1, \alpha\right)$. Intuitively, the overshoot, and hence the amount that is unable to be covered by the failed R, is asymptotically Pareto ($\approx \delta W b$). When R collapses, Assumption 1 is in place, and each I_i has to absorb a fraction of this unsettled exposure proportional to its current reserve level. Since in this case the shock is common to all the claims, the allocation to each player in set A is expected to be roughly proportional to $\gamma_i N_i(n)$, $i \in A$. To make this intuition precise, let A_0 be a strict subset of A. Note that

$$\begin{aligned}
&\mathbb{P}\Big(\tau_R < \tau_A \leq M | B_Z\Big) \\
&= \sum_{n=1}^{M-1} \mathbb{P}\big(\tau_R = n < \tau_A \leq M | B_Z\big) \\
&= \sum_{A_0 \subset A} \sum_{n=1}^{M-1} \mathbb{P}\Big(u_i(n) \geq 0, \forall i \in A_0 \Big| B_Z\Big) \mathbb{P}\Big(n = \tau_R < \tau_A \leq M | B_Z, u_i(n) \geq 0, \forall i \in A_0\Big)
\end{aligned}$$

$$
\begin{aligned}
&\leq \sum_{A_0 \subset A} \sum_{n=1}^{M-1} \Theta\left[\mathbb{P}\Big(\gamma_i N_i(n-1)\delta W b \leq u_i(n-1) + \overline{C}_i, \forall i \in A_0\Big)\right] \\
&\quad \times \mathbb{P}\Big(n = \tau_R < \tau_A \leq M | B_Z, u_i(n) \geq 0, \forall i \in A_0\Big) \\
&= o\left(1\right),
\end{aligned}
$$

where the third line follows by virtue of (5.31). The last equality holds because, for the first probability in the summand,

$$
\begin{aligned}
&\mathbb{P}\Big(\gamma_i N_i(n-1)\delta W b \leq u_i(n-1) + \overline{C}_i, \forall i \in A_0\Big) \\
=&\Theta\left[\prod_{i \in A_0} \mathbb{P}\left(W \leq \frac{r_i}{\gamma_i \delta \mathbb{E}\big(N_i(n-1)\big)}\right)\right] = \Theta\left(1\right),
\end{aligned}
$$

where we used Lemma 5.1. At the same time,

$$
\mathbb{P}\Big(n = \tau_R < \tau_A \leq M | B_Z, u_i(n) \geq 0, \forall i \in A_0\Big) = o(1)
$$

since we need a few more large factors in the remaining players in $A \setminus A_0$ in order to bring down those in set A. Therefore, let $\sigma_i \stackrel{\Delta}{=} r_i/2$, $i \in A$, we have

$$
\begin{aligned}
\mathbb{P}\Big(\tau_R \leq \tau_A \leq M | B_Z\Big) &= \Theta\Big(\mathbb{P}\big(\tau_R = \tau_A \leq M | B_Z\big)\Big) \\
&= \Theta\Big(\sum_{n=1}^{M} \mathbb{P}\big(\gamma_i N_i(n)\delta W b > \sigma_i b, \forall i \in A; \tau_R = n\big)\Big) \\
&= \Theta(1), \qquad (5.32)
\end{aligned}
$$

once again by virtue of (5.31) and Lemma 5.1. On the other hand, since

$$\mathbb{P}\left(\left(\sum_{i=1}^{K}\gamma_i N_i(1)\right)Z_1 \geq \delta b; \tau_A \geq 1 = \tau_R\right) \leq \mathbb{P}\left(B_Z\right) \leq \sum_{n=1}^{M}\mathbb{P}\left(\left(\sum_{i=1}^{K}\gamma_i N_i(n)\right)Z_n \geq \delta b\right), \tag{5.33}$$

along with (5.32) we conclude that

$$p_Z = \Theta\big(\mathbb{P}\left(B_Z\right)\big) = \Theta\big(b^{-\alpha}\big). \tag{5.34}$$

2) Analysis of p_Y.

The intuition is that, it is cheaper to bring down R by the occurrence of a large individual factor from some company, say I_i, in the set A than from outside A. From Lemma 5.2 we know that, for $1 \leq i \leq K$,

$$\left(\sum_{j=1}^{N_i(n)}\beta_i Y_{i,j}(n)\middle|\sum_{j=1}^{N_i(n)}\beta_i Y_{i,j}(n) > \delta b\right) \sim \left(\delta + \delta W_i\right) b, \tag{5.35}$$

where $W_i \sim Pareto(1, \alpha_i)$. Consider first the case if R is failed by some large individual factor from, say $I_l, l \notin A$, the same factor will create an overshoot of unsettled claims of size $\Theta(b)$. And spelled by Assumption 1, I_l will absorb $\Theta(1)$ proportion of the overshoot, large enough to fail I_l itself with $\Theta(1)$ probability. Whereas the remaining companies, $I_{l'}, l' \in A, l' \neq l$ will take on merely $\Theta(1/b)$ proportion of the unsettled claim, and hence will fail by this large individual factor from I_l with probability of size only $\Theta\left(b^{-\alpha_{l'}}\right), l' \in A, l \neq l'$. The probability of failing the remaining companies in A is of order $\Theta\left(b^{-\sum_{i\in A}\alpha_i}\right)$, leading to a total probability of $\Theta\left(b^{-\alpha_l - \sum_{i\in A}\alpha_i}\right)$. If, however, it is some individual factor from $I_i, i \in A$ that fails R in the first place, the probability of $\{\tau_A \leq M\}$ happening out of this scenario amounts to $\Theta\left(b^{-\sum_{i\in A}\alpha_i}\right)$.

We now proceed to make the previous argument more precise. First, we have, for any

$i \leq K$,

$$\mathbb{P}\big(\tau_i = \tau_R \leq M | B_{Y,i}\big) = \Theta\left[\mathbb{P}\big(\delta W_i b > \min_{i\leq K} r b\big)\right] = \Theta(1).$$

As soon as R fails, the remaining insurers no longer receive protection. Subsequently they face complete exogenous claims that are heavy-tailed. The event $E_{Y,i}$, $i \leq K$, defined via

$$E_{Y,i} \stackrel{\Delta}{=} \{\tau_A \leq \tau_R \leq M | B_{Y,i}, \tau_i = \tau_R \leq M\}$$

comes about out of the following two scenarios.

i) *Arrival of a large common factor.*

Similar to the analysis at the beginning of the proof, $E_{Y,i}$ is induced by the occurrence of a common factor if and only if there exists $\tau_R \leq n \leq M$, such that

$$\left(\sum_{l\in A\setminus\{i\}} \gamma_l N_l(n)\right) Z_n \geq \min_{l\in A\setminus\{i\}} r_l b/2,$$

the probability of which, by virtue of Lemma 5.1, is again $\Theta\left(b^{-\alpha}\right)$.

ii) *Individual factors.*

For each $l \in A \setminus \{i\}$, we require that there exists $\tau_R \leq n_l \leq M$, such that

$$\sum_{j=1}^{N_i(n_l)} \beta_l Y_{l,j}(n_l) \geq r_l b/2$$

which, again due to Lemma 5.1, independently has probability of order $\Theta\left(b^{-\alpha_l}\right)$. Therefore,

$$\mathbb{P}\left(E_{Y,i}\right) = \Theta\left(b^{-\sum_{l\in A\setminus\{i\}} \alpha_l}\right).$$

It remains to calculate $\mathbb{P}(B_{Y,i})$. Applying similar bounds as in (5.33), we have

$$\mathbb{P}\left(\sum_{j=1}^{N_i(1)}\beta_i Y_{i,j}(1)\geq \delta b, \tau_A\geq 1=\tau_R\right)\leq \mathbb{P}(B_{Y,i})\leq \sum_{n=1}^{M}\mathbb{P}\left(\sum_{j=1}^{N_i(n)}\beta_i Y_{i,j}(n)\geq \delta b\right).$$

Lemma 5.1 allows us to conclude that $\mathbb{P}(B_Y)=\Theta(b^{-\alpha_i})$. Consequently,

$$\begin{aligned} p_Y &= \sum_{i\leq K}\mathbb{P}\big(E_{Y,i}\big)\mathbb{P}\big(\tau_i=\tau_R\leq M|B_{Y,i}\big) \\ &= \Theta\left[\sum_{i\in A}\mathbb{P}\big(E_{Y,i}\big)\mathbb{P}\big(\tau_i=\tau_R\leq M|B_{Y,i}\big)\mathbb{P}(B_{Y,i})\right] \\ &= \begin{cases} \Theta\left[b^{-\left(\alpha+\min_{i\leq K}\alpha_i\right)}\right], & \text{Individual}\to\text{Common} \\ \Theta\left[b^{-\sum_{i\in A}\alpha_i}\right]. & \text{Individual}\to\text{Individual} \end{cases} \end{aligned} \tag{5.36}$$

And therefore the criteria given by (5.29) distinguishes p_Z from p_Y. Recall from the discussion at the beginning of the section that the probability $\mathbb{P}(\tau_A<M\wedge\tau_R)$ decays exponentially, it's immediate from (5.34) and (5.36) that

$$\mathbb{P}(\tau_A<M\wedge\tau_R)=o\big(\mathbb{P}(\tau_R\leq\tau_A\leq M)\big).$$

The result follows. □

5.4 Design of Efficient Simulation Algorithms for $\mathcal{N}_e$

The asymptotic analysis in the preceding section is useful in obtaining a qualitative description of the systemic risk landscape of the entire network. However, in order to achieve this one is required to fully solve a combinatorial problem. Moreover, the resulting asymptotic description is rather coarse. In this section we aim to achieve a more

precise quantitative assessment and make sharper evaluations of the embedded systemic risk throughout the network $\mathcal{N}_e$. We resort to the tool of Monte Carlo methods, and our goal is to propose an efficient simulation algorithm to evaluate the *conditional system dislocation* (5.18). We do this by designing an algorithm for the probability

$$q(b) = \mathbb{P}\left(\tau_A(b) \leq M\right)$$

instead. Estimators for (5.18) is a natural consequence.

5.4.1 Guidelines for Simulation Design

As pointed out in Subsection 1.2.3, the design of provably efficient simulation algorithms oftentimes relies on a careful asymptotic description of the system as a meaningful departing point. Therefore, constructing efficient estimators for the network system introduced in Section 5.2 will hinge on the insight from the large deviations analysis presented in the previous section.

Before we proceed, we require that our final estimator shall possess strong efficiency, an efficiency characteristics given in Definition 1.9 in Subsection 1.2.4. Given this notion of efficiency, our goal is to search for an estimator within the class of strongly efficient estimators that is *practically convenient*. Ideally, we hope the algorithm shares a uniform setup under various configurations of the system, and is easy to implement, without sacrificing too much efficiency. This translates to the search of a probability measure

$$\widetilde{P}(\cdot) \triangleq \mathbb{P}\left(\cdot|\widetilde{E}_n\right)$$

for some conditioning event $\widetilde{E}_n$ carefully "maneuvered" so that

1) Path sampling under $\widetilde{\mathbb{P}}$ is not complicated.

2) The behavior of the system under $\widetilde{\mathbb{P}}$, i.e., conditional on $\widetilde{E}_n$, is reasonably close to $\mathbb{P}_n^*$.

3) The associated estimator possesses the required notion of efficiency, in this setting in particular, strong efficiency.

And on top of these criteria we demand that

4) The algorithm requires minimum and uniform setup under various system configurations.

Considering the network model we study, it might be desirable to have the same estimator no matter how the claim structure varies that leads to different large deviations behavior (see Theorem 5.5 and Proposition 5.1). The bottom line is, within the class of strongly efficient estimators, one might be willing to sacrifice efficiency in exchange for convenience and flexibility.

5.4.2 A Mixture-based SDIS

Loosely speaking, large deviations behaviors of heavy-tailed systems are governed by the so-called "principle of large jumps" or "catastrophe principle", which declares that large deviations are triggered by one or a few components with immoderate magnitudes (see Subsection 1.2.2; also see [12] for an extended discussion). Recall from Section 5.2 that the reserve processes $u(n)$ and $u_i(n)$ are essentially heavy-tailed random walks whose increments are random sums of factors per se. The natural direction to pursue is therefore biasing the sampling distribution of the factors to be "locally" compatible with the large deviations rule of thumb stated above. The challenge is, however, how to judiciously pick the change of measure so that paths generated under such a measure can be sufficiently close to the most likely paths of the system that underscore both regimes (see Section

5.3). We need the following proposition in order to further connect the dots and achieve this goal. The essence of the result is of the same flavor as Proposition 1 in [17].

Proposition 5.2. *Given the network $\mathcal{N}_e$ defined in Section 5.2, define*

$$\delta_N \stackrel{\Delta}{=} \min_{i\in A} \frac{r_i}{2M\overline{N}_i \left(\sum_{h=1}^{d} \gamma_{i,h} + \beta_i\right)},$$

where $\overline{N}_i = \max_{k\le M} \widetilde{N}_i(k)$, $i \in \mathcal{I}$. Let $\mathcal{X}$ be the set of feasible solutions to the IP given in (5.21). And define

$$\mathcal{A}_{\delta_N}(b) \stackrel{\Delta}{=}$$
$$\bigcup_{x\in\mathcal{X}} \left\{ \bigcap_{i\in A} \left[\bigcup_{k\le M} \left(\left(\bigcup_{\substack{1\le h\le d \\ \gamma_{i,h}x_h>0}} \left\{ Z_h(k) \ge \delta_N b \right\} \right) \bigcup \left(\bigcup_{\substack{1\le l\le \widetilde{N}_i(k) \\ x_{i+d}=1}} \left\{ Y_{i,l}(k) \ge \delta_N b \right\} \right) \right) \right] \right\}$$

Then we have

i) $\mathcal{A}_{\delta_N}(b)$ is a superset of $\{\tau_A(b) \le M\}$, i.e.,

$$\mathcal{A}_{\delta_N}(b) \supseteq \{\tau_A(b) \le M\}. \tag{5.37}$$

ii) Conditioning on $N_i(k), i \in \mathcal{I}, k \le M$, we have, as $b \nearrow \infty$,

$$\frac{\log \mathbb{P}\big(\mathcal{A}_{\delta_N}(b)\big)}{\log b} \longrightarrow -\zeta,$$

where ζ is the optimal cost to $[IP]$ in (5.21).

Proof. i) Suppose there exists $i' \in A$, such that 1) $Z_h(k) < \delta_N b$ for all $h \le d$ such that $\gamma_{i',h}x_h > 0$, and for all $1 \le k \le M$, and 2) $Y_{i',l}(k) < \delta_N b$ for all $1 \le l \le \widetilde{N}_{i'}(k)$ and

for all $1 \leq k \leq M$, then we have, for any $n \leq M$,

$$\begin{aligned}
& u_{i'}(n) \\
\geq \quad & r_i b - \sum_{k=1}^{n} \left(\sum_{h=1}^{d} \gamma_{i',h} Z_h(k) \widetilde{N}_{i'}(k) + \sum_{l=1}^{\widetilde{N}_{i'}(k)} \beta_{i'} Y_{i',l}(k) \right) - \sum_{k=1}^{n} \sum_{s \in \mathcal{R}} \check{\psi}_s^-(k) \cdot \rho_{si'}(k) \\
\geq \quad & r_i b - \delta_N b \cdot n \overline{N}_{i'} \left(\sum_{h=1}^{d} \gamma_{i',h} + \beta_{i'} \right) - \sum_{k=1}^{n} \sum_{s \in \mathcal{R}} \check{\psi}_s^-(k) \cdot \rho_{si'}(k) \\
\geq \quad & r_i b/2 - \sum_{k=1}^{n} \sum_{s \in \mathcal{R}} \check{\psi}_s^-(k) \cdot \rho_{si'}(k),
\end{aligned}$$

where $\check{\psi}_s^-(k)$ is the optimal solution for $\psi_s^-(k)$, $s \in \mathcal{R}$ for the linear program $[P^\kappa(k)]$. Furthermore, the model setup ensures that at any point in time, each insurer *cannot* receive an allocation of the spillover losses from all of its reinsurance counterparties of an aggregate amount larger than the total amount it reinsures. In what follows, we shall refer to this observation as *limited spillover impact*. Therefore, we have

$$\sum_{k=1}^{n} \sum_{s \in \mathcal{R}} \check{\psi}_s^-(k) \cdot \rho_{si'}(k) \leq \sum_{k=1}^{n} \left(\sum_{h=1}^{d} \gamma_{i',h} Z_h(k) \widetilde{N}_{i'}(k) + \sum_{l=1}^{\widetilde{N}_{i'}(k)} \beta_{i'} Y_{i',l}(k) \right) \leq r_{i'} b/2.$$

And consequently $u_{i'}(n) \geq 0$, for all $n \leq M$, and this implies that $\{\tau_A(b) > M\}$. We have thus established (5.37).

ii) An equivalent expression for $\mathcal{A}_{\delta_N}(b)$ is given by

$$\mathcal{A}_{\delta_N}(b) = \bigcup_{\mathbf{x} \in \mathcal{X}} \left\{ \bigcup_{k \leq M} \left(\bigcap_{i \in A} \left(\bigcup_{1 \leq j \leq m, \Xi_{ij} x_j \geq 1} \left\{ U_j(k) \geq \delta_N b \right\} \right) \right) \right\},$$

where Ξ is the factor exposure matrix defined in (5.19), and $m = d + |\mathcal{I}|$ is the number of column of Ξ. Recall that $U_j = Z_h$ if $1 \leq j \leq d$, and $U_j = Y_i$ if $j = d + i$,

$i \in \mathcal{I}$. Let us further define

$$\mathcal{S}(\mathbf{x}) = \{j = d + i : i \in \mathcal{I}, x_j = 1\} \cup \{h \leq d : x_h = 1\}, \tag{5.38}$$

i.e., $\mathcal{S}(\mathbf{x})$ is the index set of active factors associated with $[IP]$-feasible solution $\mathbf{x}$.

For the lower bound, we note that

$$\begin{aligned}
\mathbb{P}\left(\mathcal{A}_{\delta_N}(b)\right) &\geq \mathbb{P}\left[\bigcap_{i\in A}\left(\bigcup_{1\leq j\leq m, \Xi_{ij}x_j^*\geq 1}\left\{U_j(1) \geq \delta_N b\right\}\right)\right] \\
&= \prod_{j\in\mathcal{S}(\mathbf{x}^*)} \mathbb{P}\big(U_j(1) \geq \delta_N b\big) \\
&\geq \mathbb{E}\left[\delta_N\right]^{-\widetilde{\alpha}^T\mathbf{e}} b^{-\widetilde{\alpha}^T\mathbf{x}^*} \geq \kappa_1 b^{-\widetilde{\alpha}^T\mathbf{x}^*},
\end{aligned}$$

for some positive constant κ_1, where $\mathbf{x}^*$ is an $[IP]$-optimal solution. Here the second inequality arises from Lemma 5.1.

And for the other direction, we utilize a union bound instead. In particular,

$$\mathbb{P}\left(\mathcal{A}_{\delta_N}(b)\right) \leq \sum_{\mathbf{x}\in\mathcal{X}}\sum_{n=1}^{M}\mathbb{P}\left[\bigcap_{i\in A}\left(\bigcup_{1\leq j\leq m, \Xi_{ij}x_j\geq 1}\left\{U_j(n) \geq \delta_N b\right\}\right)\right] \leq \kappa_2 b^{-\widetilde{\alpha}^T\mathbf{x}^*}, \tag{5.39}$$

for some positive constant κ_2, where x^* is again an optimal solution to $[IP]$. The result follows immediately after taking log for both the lower and upper bounds.

□

An immediate implication of the previous results is a sampling scheme that induces the occurrence of adequately large (of size at least δ_N) common or individual factors at each period might be sufficient to guarantee bounded relative error of the estimator. We in fact implemented this *state-independent* algorithm, and realized that a dynamic version of the

change of measure seems to be as easy to implement as the state-independent counterpart, but could further reduce the relative variance of the associated estimator. From the simulation perspective, the order of occurrence of the factors during each period deems irrelevant. Our strategy is therefore to view the factors as if they arrive sequentially. At each period, we can consider the random sums of the factors, as random walks themselves, thereby creating this "internal" layer of random walks. From this point on we can borrow apparatus from established *state-dependent* rare event simulation algorithms to aid the design of our importance sampling estimator. In particular, we shall exploit the idea developed in [34] (see also the survey paper [17]).

The key ingredient is a mixture based importance sampling distribution for the increments: with some probability $p(n)$, the increment is sampled conditioning on it being "large", and with probability $1-p(n)$, it's sampled as if it's a "normal" shock. Let X be the increment of the system, and without loss of generality suppose its density is given by $f(x)$, then the nth increment is drawn from the importance density $g_n(\cdot)$, defined as

$$g_n(x) = \left[p(n)\frac{I\big(x \in A_n(b)\big)}{\mathbb{P}\big(X_n \in A_n(b)\big)} + (1-p(n))\frac{I\left(x \in \overline{A_n(b)}\right)}{\mathbb{P}\left(X_n \in \overline{A_n(b)}\right)} \right] f(x), \tag{5.40}$$

where $A_n(b)$ specifies the region in which the increment is qualified to be a large shock. Note that the part in (5.40) corresponding to the "normal" jumps is necessary in order to conciliate the sensitivities of large deviations probabilities to the likelihood ratio of those paths that have more than one jumps of order $\Omega(b)$, a crucial observation pointed out by [12] (see also Example 4.1 in Chapter 4).

In the one dimensional random walk case, $A_n(b)$ is typically chosen to be proportional to the "distance to go" for the current position of the random walk, i.e., $A_n(b) = a(b - s_{n-1})$, for some $a \in (0,1)$ and $s_n = x_1 + \cdots + x_n$. In more general cases, $A_n(b)$ can be

derived from some "auxiliary" or "steering" processes other than the targeting process. A convenient choice of such an auxiliary process in our setting is obtained by "eliminating" the reinsurance participants $\mathcal{R}$ a priori and allocating the reserve process $u_s^R(n)$, $s \in \mathcal{R}$ proportionally to each $u_i(n)$, $i \in \mathcal{I}$. Equivalently, we pretend that the I_i's absorb full sized claims without reaching out to $\mathcal{R}$ to hedge risks. In principle, to recoup this higher risks taken by the insurers, the initial reserves $u_i(0)$'s, $i \in \mathcal{I}$ shall also be adjusted up accordingly, but we dispense ourselves with this adjustment in the auxiliary process. The benefit of doing so will be discussed after we outline the algorithm in the next subsection.

Effectively the auxiliary process consists of K_I random walks, dependent 1) explicitly upon the common factor $\{Z_h\}_{h \le d}$ and 2) implicitly on the presence of $\{R_s\}_{s \in \mathcal{R}}$. At the beginning of each period, we first sample the common factors for the current period in order to strip off the first layer of dependence among the claims; and then sequentially sample the remaining individual factors. The mixture sampling density (5.40) is used to sample each factor that corresponds to the survival companies in A, with the "distance to go" $A_n(b)$ properly defined in a dynamic way. We shall detail this choice in the next subsection. The resulting sampling scheme is easy to carry out, self-adjusting in nature, and saves the user the trouble of setting up the algorithm differently according to different network structures. Proposition 5.2 implies that the system simulated in this way is guaranteed to be within a moderate "distance" from the large deviations description of the system, which is sufficient to preserve strong efficiency of the associated estimator. Formally we have the following efficiency result, the proof of which is postponed after we have detailed the algorithm in the next subsection.

Theorem 5.7. *The adaptive importance sampling estimator $\widehat{q}_{Z,Y,N}$ (to be defined in (5.44) and (5.45) in the next subsection) is strongly efficient for estimating $q(b) = \mathbb{P}(\tau_A(b) \le M)$.*

If, in addition, $\alpha_i > 2$, for all $i \in \mathcal{I}$, and $\alpha_h^Z > 2$, for all $1 \le h \le d$, then the estimator

$$\widehat{h}_{Z,Y,N} \triangleq \sum_{i \in \mathcal{I}} \widehat{q}_{Z,Y,N} D_i(A)$$

is also strongly efficient for estimating $CSD(A) = \sum_{i \in \mathcal{I}} \mathbb{E}\big[D_i(A) I\left(\tau_A \le M\right)\big]$.

5.4.3 The Algorithm

We are now ready to carry out our plan and pinpoint the state-dependent importance sampling idea in details. We start by defining the auxiliary process via

$$\begin{aligned} S_i(n) &= \sum_{k=1}^{n} \sum_{j=1}^{\widetilde{N}_i(k)} V_{i,j}(k) - C_i n, \\ S_i^{(0)}(n+1) &= S_i(n) + \widetilde{N}_i(n+1) \sum_{h=1}^{d} \gamma_{i,h} Z_h(n+1) - C_i, \\ S_i^{(l)}(n+1) &= S_i^{(l-1)}(n+1) + \beta_i Y_{i,l}(n+1),\ 1 \le l \le \widetilde{N}_i(n+1), \end{aligned} \tag{5.41}$$

for each $i \in A$, where $V_{i,j}(k)$ is the claim size random variable defined in (5.2). We then summarize the details of our general SDIS algorithm for $\mathcal{N}_e$ as follows.

Description of The SDIS Algorithm

1) Solve the integer program, $[IP]$, given in (5.21). Recall that $\mathcal{X}$ is the set of feasible solutions to $[IP]$. Define

$$\mathcal{S} = \bigcup_{\mathbf{x} \in \mathcal{X}} \mathcal{S}(\mathbf{x}),\ \mathcal{S}^* = \bigcap_{\mathbf{x} \in \mathcal{X}} \mathcal{S}(\mathbf{x}), \tag{5.42}$$

where $\mathcal{S}(\mathbf{x})$ is defined in (5.38). In other words, $l \in \mathcal{S}$ if the l-th factor is active in *some* $[IP]$-feasible solutions, and $l \in \mathcal{S}^*$ if the l-th factor is active in *all* $[IP]$-optimal

solutions.

2) Sample $\widetilde{N}_i(n)$ from Binomial$\big(\overline{N}_n, q_n\big)$, for each $i \in \mathcal{I}, n \leq M$.

3) While $n \leq M$, at the beginning of period n, $1 \leq n \leq M$, let the survival companies in A be denoted as $A^{(n)}$. For each $1 \leq h \leq d$, let $f_{Z_h}(\cdot)$ be the density for the common factor Z_h. For $h \in \mathcal{S}$, given that $S_A(n-1) = \{S_i(n-1)\}_{i \in A^{(n)}} = s_A$, sample $Z_h(n)$ from the following mixture density

$$g_{h,n}(z|s_A) = \left[p_{Z_h}(n) \frac{I\big(z \geq ad_n(b, s_A)\big)}{\mathbb{P}\big(Z_h(n) \geq ad_n(b, s_A)\big)} + \big(1 - p_{Z_h}(n)\big) \frac{I\big(z < ad_n(b, s_A)\big)}{\mathbb{P}\big(Z_h(n) < ad_n(b, s_A)\big)} \right] f_{Z_h}(z),$$

for some positive choice the mixing probability $p_{Z_h}(n) \in (0, 1)$, where the "distance to go" d_n is defined as

$$d_n\left(b, S_A(n-1)\right) = \max\left(0, \min_{i \in A^{(n)}, \gamma_{i,h} > 0, h \in \mathcal{S}} \left(\frac{r_i b - S_i(n-1)}{d\gamma_{i,h} \widetilde{N}_i(n)} \right) \right).$$

For $h \notin \mathcal{S}$, sample $Z_h(n)$ from its original density. It is understood that $p_{Z_h}(n) = 0$ if $d_n\left(b, S_A(n-1)\right) \leq 0$, i.e., importance sampling is *switched off* when the auxiliary process hits the corresponding initial reserve level.

4) For each $i \in A^{(n)}$, if $\tau_i \leq n-1$, sample $Y_{i,l}(n)$, for each $1 \leq l \leq \widetilde{N}_i(n)$, from its original distribution. Otherwise, if $d + i \in \mathcal{S}$, given $S_i^{(l-1)}(n-1) = s$, sample $Y_{i,l}(n)$ from the mixture density given by

$$g_{i,n}^{(l)}(y|s) = \left[p_{i,j}(n) \frac{I\big(y > ad_{i,n}^{(l)}(b, s)\big)}{\mathbb{P}\big(Y_i(n) > ad_{i,n}^{(l)}(b, s)\big)} + \big(1 - p_{i,j}(n)\big) \frac{I\big(y \leq ad_{i,n}^{(l)}(b, s)\big)}{\mathbb{P}\big(Y_i(n) \leq ad_{i,n}^{(l)}(b, s)\big)} \right] f_{Y_i}(y),$$

for some positive mixing probability $p_{i,j}(n) \in (0,1)$, with the "distance to go" defined via

$$d_{i,n}^{(l)}(b, S_i^{(l-1)}(n-1)) = \max\left(0, \frac{r_i b - S_i^{(l-1)}(n-1)}{\beta_i}\right), \ \forall i : i \in A^{(n)} \text{ and } d+i \in \mathcal{S}.$$

And if $d+i \notin \mathcal{S}$, sample Y from its original density.

5) Given $Z_h(n), h \leq d$ and $Y_{i,l}(n)$ sampled in Step 3) and 4), update $S_i^{(l)}(n-1)$ by (5.41).

6) Set ρ_{si} and $\rho_{s's}$, $s, s' \in \mathcal{R}, i \in I$ according to (5.16).

7) Let the survival insurers and reinsurers at the beginning of period n be denoted as $\mathcal{I}^{+(n)}$ and $\mathcal{R}^{+(n)}$, respectively. Solve the single-period linear program $[P^\kappa]$ given in (5.5), with $\mathcal{I}^+$ and $\mathcal{R}^+$ replaced by $\mathcal{I}^{+(n)}$ and $\mathcal{R}^{+(n)}$, respectively. Let $\left(\check{\pi}_s^+(n), \check{\pi}_s^-(n), \check{\psi}_s^+(n), \check{\psi}_s^-(n)\right)$ be the optimal solution vector. Update the true reserve processes according to (5.17), i.e., $u_i(n) = \check{\pi}_s^+(n) + \check{\pi}_s^-(n)$ for each $i \in \mathcal{I}^{+(n)}$, and $u_s^R(n) = \check{\psi}_s^+(n) + \check{\psi}_s^-(n)$, for each $s \in \mathcal{R}^{+(n)}$.

8) Set $n = n+1$, and go to Step 3).

Remark 5.3. *In the algorithm above, we can further guide the choices of the mixing probabilities p_{Z_h} and $p_{i,j}$ by setting $p_{Z_h}(n) = \theta/(M-n+1)$ if $h \in \mathcal{S}^*$, and setting $p_{i,j}(n) = \theta'/\sum_{k=n}^{M} N_i(k)$ if $d+i \in \mathcal{S}^*$, where $\mathcal{S}^*$ is defined in (5.42), and θ, θ' are some positive constants independent of b. The choices are consistent with the asymptotic behaviors of the system in the sense that they*

1) reflects the large deviations description of the system, as specified by Theorem 5.5 (i.e., we endow a large value to the mixing probability if the associated factor is active in all $[IP]$-feasible solutions, and hence must be active in all $[IP]$-optimal solutions).

2) reflects the large deviations heuristics local to each company $i \in A$ (i.e., the choices $\theta/(M-n+1)$ and $\theta'/\sum_{k=n}^{M} N_i(k)$ are roughly proportional to the remaining chances that Z_h and Y_i are large).

It is, however, necessary to assign a small (bounded away from zero) probability to the mixing probability for which the associated factor is active in some but not all $[IP]$-optimal solutions. This is because paths in which these factors are large create a non-negligible contribution to the variance of the estimator. Therefore, if $h \in \mathcal{S}\backslash\mathcal{S}^*$, we set $p_{Z_h}(n) = \epsilon_Z \ll \theta/(M-n+1)$; and if $d+i \in \mathcal{S}\backslash\mathcal{S}^*$, we set $p_{i,j}(n) = \epsilon_Y \ll \theta'/\sum_{k=n}^{M} N_i(k)$, where both ϵ_Z and ϵ_Y are small positive constants.

Remark 5.4. *It is necessary to simulate all the claims within a period for I_i even if some intermediate claim causes its reserve to go below zero. This is because claims are assumed to be aggregated at the end of each period. However, the SDIS scheme should be switched off as soon as that insurer fails, and one shall continue with Crude Monte Carlo towards the end of that period.*

Before we state the formal expression of the estimator for $q(b)$, in light of the previous remark, let us define, with a slight abuse of notation, $n^{i,l}$ the moment immediately after the l-th individual factor for insurer I_i has been sampled at period n. And write

$$u_i\left(n^{i,l}\right) = u_i(n-1) - \sum_{j=1}^{l} X_{i,j}(n),$$

for $1 \leq l \leq \widetilde{N}_i(n), i \in \mathcal{I}$. Further define

$$\widetilde{\tau}_i = \inf_{k \leq M, l \leq \widetilde{N}_i(n)} \left\{ k^{i,l} : u_i\left(k^{i,l}\right) \leq 0 \right\}. \tag{5.43}$$

The Estimator

Define the local likelihood ratio of the aggregate claims between the original and change of measure as follows:

$$
\begin{aligned}
&\xi_{Z,Y,N}(n) \\
&= \left(\prod_{h\in\mathcal{S}} \frac{f_{Z_h}(Z_h(n))}{g_{h,n}\left(Z_h(n)|S_A(n-1)\right)}\right) \times \left(\prod_{i\in A^{(n)}\cap\mathcal{S}} \prod_{j=1}^{N_i(n)} I\left(\tau_i > n-1\right) \frac{f_{Y_i}\left(Y_{i,j}\right)}{g_{i,j}^{(j)}\left(Y_{i,j}|S_i^{(j-1)}\right)}\right) \\
&= \prod_{h\in\mathcal{S}} \Bigg[\frac{\mathbb{P}\left(Z_h(n) > ad_n(b, S_A(n-1))\right)}{p_{Z_h}(n)} I\big(Z_h(n) > ad_n(b, S_A(n-1))\big) \\
&\quad + \frac{\mathbb{P}\left(Z_h(n) \le ad_n(b, S_A(n-1))\right)}{1-p_{Z_h}(n)} I\big(Z_h(n) \le ad_n(b, S_A(n-1))\big)\Bigg] \\
&\quad \times \prod_{i:i\in A^{(n)}, d+i\in\mathcal{S}} \prod_{j=1}^{N_i(n)} I\left(\widetilde{\tau}_i > n^{i,j-1}\right) \\
&\quad \Bigg[\frac{\mathbb{P}\left(Y_i(n) > ad_{i,n}^{(j)}\left(b, S_i^{(j-1)}(n)\right)\right)}{p_{i,j}(n)} I\left(Y_i(n) > ad_{i,n}^{(j)}\left(b, S_i^{(j-1)}(n)\right)\right) \\
&\quad + \frac{\mathbb{P}\left(Y_i(n) \le ad_{i,n}^{(j)}\left(b, S_i^{(j-1)}(n)\right)\right)}{1-p_{i,j}(n)} I\left(Y_i(n) \le ad_{i,n}^{(j)}\left(b, S_i^{(j-1)}(n)\right)\right)\Bigg], \qquad (5.44)
\end{aligned}
$$

for $n \le M$. The estimator for the probability $q(b) = \mathbb{P}\left(\tau_A(b) \le M\right)$ is therefore given by

$$
\widehat{q}_{Z,Y,N} = \prod_{n=1}^{M} \xi_{Z,Y,N}(n) I\left(\tau_A \le M\right) = \prod_{n=1}^{M} \xi_{Z,Y,N}(n) I\left(A^{(M+1)} = \emptyset\right). \qquad (5.45)
$$

5.4.4 Proof of Theorem 5.5 and 5.7.

We first prove Theorem 5.7, which concludes our efficiency analysis of the algorithm, and then we finish the proof of Theorem 5.5 given in Section 5.3.

Proof of Theorem 5.7. Let $\widetilde{\mathbb{P}}\left(\cdot\right)$ be the probability measure induced by the proposed importance sampling distribution, and $\widetilde{\mathbb{E}}\left(\cdot\right)$ the associated expectation operator. Note that along a sample path generated under $\widetilde{\mathbb{P}}$ that eventually leads to the ruin of the set A

before time $M < \infty$, there exists (k_i, j, k) where $1 \leq k_i, k \leq M$, $1 \leq j \leq \widetilde{N}_i(k_i)$ such that at least one of the following cases occurs:

$$\begin{aligned} &1.\ Z_h(k) > ad_k\left(S_A(k-1)\right), \text{ for some } d \leq h, \\ &2.\ Y_{i,j}(k_i) > ad_{i,k_i}^{(j)}\left(b, S_i^{(j-1)}(k_i)\right), \end{aligned} \tag{5.46}$$

for all $i \in A$. Otherwise, we would obtain, for some $i \in A$,

$$S_i^{(l)}(n) - S_i^{(l-1)}(n) \leq \beta_i Y_{i,l}(n) \leq a\big(r_i b - S_i^{(l-1)}(n)\big), \text{ for } 1 \leq l \leq \widetilde{N}_i(n),$$

and

$$S_i^{(0)}(n) - S_i(n-1) \leq \widetilde{N}_i(n) \sum_{h=1}^{d} \gamma_{i,h} Z_h(n) \leq a\left(r_i b - S_i(n-1)\right),$$

for all $1 \leq n \leq M$. We want to use a telescopic sum over l, we therefore define $S_i^{(-1)}(n) = S_i(n-1)$, so that the previous two inequalities can be put together. As a result, we obtain,

$$\begin{aligned} S_i^{(j)}(n) &\leq ar_i b + a(1-a)r_i b + \cdots + a(1-a)^{j+\sum_{k=1}^{n-1}\left(\widetilde{N}_i(k)+1\right)} r_i b \\ &\leq r_i b\left(1 - (1-a)^{j+1+\sum_{k=1}^{n-1}\left(\widetilde{N}_i(k)+1\right)}\right) \\ &< r_i b, \end{aligned} \tag{5.47}$$

for some $i \in A$ for all $1 \leq n \leq M, -1 \leq j \leq \widetilde{N}_i(n)$. This implies $\tau_i > M$ and hence $\tau_A(b) > M$. Now, for each $i \in A$, let $n^* \in \{1, \ldots, M\}$ be the time at which a large factor (i.e., either (1) or (2) in (5.46) occurs). Furthermore, let $j_i^* = 0$ if such a large factor turns out to be any of the common factors (corresponding to the occurrence of (1) in (5.46)). Otherwise, we set $j_i^* \in \{1, \ldots, \widetilde{N}_i(n_i^*)\}$, corresponding to the index of the claim at which

(2) in (5.46) first occurs. It's not difficult to see from (5.47) that, if $(n_i^*, j_i^*) = (n, j)$,

$$S_i^{(j-1)}(n) \leq r_i b \left(1 - (1-a)^{\sum_{k=1}^{M}(\widetilde{N}_i(k)+1)}\right), i \in A.$$

Hence if $j = 0$,

$$\begin{aligned} d_n(b, S_A(n-1)) &= \max\left(0, \min_{i \in A^{(n)}, \gamma_{i,h}>0, h \in \mathcal{S}} \frac{r_i b - S_i(n-1)}{d\gamma_{i,h}\widetilde{N}_i(n)}\right) \\ &\geq \min_{i \in A^{(n)}, \gamma_{i,h}>0, h \in \mathcal{S}} \frac{(1-a)^{\sum_{k=1}^{M}(\widetilde{N}_i(k)+1)} r_i b}{d\gamma_{i,h}\widetilde{N}_i(n)}. \end{aligned} \tag{5.48}$$

And if $1 \leq j \leq N_i(n)$, for each $i \in \mathcal{S}$,

$$d_{i,n}^{(j)}\left(b, S_i^{(j-1)}\right) \geq \frac{r_i}{\beta_i}(1-a)^{\sum_{k=1}^{M}(\widetilde{N}_i(k)+1)} b. \tag{5.49}$$

Now, let $\widetilde{\Omega}(\mathcal{X})$ be the subset of all the sample paths generated under $\widetilde{\mathbb{P}}(\cdot)$ that contains large common factors or large individual factors (in the sense of (5.46)) matching the active factors corresponding to *any* $[IP]$-feasible solution in $\mathcal{X}$. It follows from (5.48) and (5.49) that those paths must be included on the event $\{\tau_A(b) \leq M\}$. Let the indicator $I\left((Z, Y, N) \in \widetilde{\Omega}(\mathcal{X})\right)$ be equal to one if the sample path encoded by the vector (Z, Y, N) belongs to $\widetilde{\Omega}(\mathcal{X})$, and zero otherwise. Further define

$$c_N = \min_{i \in A}\left[\frac{r_i}{\nu_i^*}(1-a)^{M(N_i^*+1)}\right], \tag{5.50}$$

where $N_i^* = \max_{k \leq M} \widetilde{N}_i(k)$, $\nu_i^* = \max\left(\max_{h \in \mathcal{S}} \gamma_{i,h} N_i^*, \max_{l: d+l \in \mathcal{S}} \beta_l\right)$, and let the set $\mathcal{A}_{c_N, \mathbf{x}}(b)$ be defined as

$$\mathcal{A}_{c_N, \mathbf{x}}(b) = \left\{\bigcup_{k \leq M}\left(\bigcap_{i \in A}\left(\bigcup_{1 \leq j \leq m, \Xi_{ij} x_j \geq 1}\left\{U_j(k) \geq c_N b\right\}\right)\right)\right\}, \tag{5.51}$$

for $\mathbf{x} \in \mathcal{X}$, where we have used the unified factor representation U introduced in Subsection 5.3.1 (see the paragraph before (5.19)). Let

$$\phi\left(\underline{\mathbf{p}}\right) = \prod_{k=1}^{M}\left[\prod_{h\in\mathcal{S}}\frac{1}{\min\left(p_{Z_h}(k), 1-p_{Z_h}(k)\right)}\prod_{i:i\in A, d+i\in\mathcal{S}}\left(\prod_{j=1}^{\widetilde{N}_i(k)}\frac{1}{\min\left(p_{Y_i}(k), 1-p_{Y_i}(k)\right)}\right)\right].$$

Then, we have

$$\widehat{q}_{Z,Y,N} I\left((Z,Y,N)\in\widetilde{\Omega}\left(\mathcal{X}\right)\right) \leq \max_{\mathbf{x}\in\mathcal{X}}\mathbb{P}\Big(\mathcal{A}_{c_N,\mathbf{x}}(b)\Big)\phi\left(\underline{\mathbf{p}}\right). \tag{5.52}$$

Now, once again by virtue of Lemma 5.1, we obtain, for any $\mathbf{x} \in \mathcal{X}$,

$$\begin{aligned}
\mathbb{P}\Big(\mathcal{A}_{c_N,\mathbf{x}}(b)\Big) &\leq \sum_{k\leq M}\mathbb{P}\left(\bigcap_{i\in A}\left(\bigcup_{1\leq j\leq m, \Xi_{ij}x_j\geq 1}\Big\{U_j(k)\geq c_N b\Big\}\right)\right)\\
&= \sum_{k\leq M}\left[\prod_{i:i\in A, d+i\in\mathcal{S}(\mathbf{x})}\mathbb{P}\left(Y_i\geq c_N b\right)\prod_{h\in\mathcal{S}(\mathbf{x})}\mathbb{P}\left(Z_h\geq c_N b\right)\right]\\
&\leq M\mathbb{E}\left(c_N\right)^{-\widetilde{\alpha}^T\mathbf{e}}\left[\prod_{i:i\in A, d+i\in\mathcal{S}(\mathbf{x})}\mathbb{P}\left(Y_i\geq b\right)\prod_{h\in\mathcal{S}(\mathbf{x})}\mathbb{P}\left(Z_h\geq b\right)\right]\\
&\leq \widetilde{K}_1\prod_{i:i\in A, d+i\in\mathcal{S}(\mathbf{x})}\mathbb{P}\left(Y_i\geq b\right)\prod_{h\in\mathcal{S}(\mathbf{x})}\mathbb{P}\left(Z_h\geq b\right)
\end{aligned} \tag{5.53}$$

for some positive constant $\widetilde{K}_1$ independent of N and b, where $\widetilde{\alpha}$ is defined in the paragraph following (5.19), and $\mathcal{S}(\mathbf{x})$ is defined in (5.38).

Meanwhile, on defining

$$\underline{c}_N(\mathbf{x}) = \left[\min_{i\in A}\left(\min_{l\in\mathcal{S}(\mathbf{x}), \Xi_{il}x_l\geq 1} r_i\Xi_{il}\right)\right]^{-1},$$

we have the following lower bound for $q(b) = \mathbb{P}\big(\tau_A(b) \geq M\big)$,

$$\begin{aligned}
\mathbb{P}\big(\tau_A(b) \geq M\big) &\geq \max_{\mathbf{x}\in\mathcal{X}} \mathbb{P}\left[\bigcap_{i\in A} \bigcup_{1\leq j\leq m, \Xi_{ij}x_j\geq 1} \Big\{U_j(1) \geq \underline{c}_N(\mathbf{x})b\Big\}\right] \\
&\geq \max_{\mathbf{x}\in\mathcal{X}} \left[\mathbb{E}\left(\underline{c}_N\right)^{-\widetilde{\alpha}^T\mathbf{e}} \prod_{i:i\in A, d+i\in\mathcal{S}(\mathbf{x})} \mathbb{P}\left(Y_i \geq b\right) \prod_{h\in\mathcal{S}(\mathbf{x})} \mathbb{P}\left(Z_h \geq b\right)\right] \\
&\geq \max_{\mathbf{x}\in\mathcal{X}} \left[\widetilde{K}_2 \prod_{i:i\in A, d+i\in\mathcal{S}(\mathbf{x})} \mathbb{P}\left(Y_i \geq b\right) \prod_{h\in\mathcal{S}(\mathbf{x})} \mathbb{P}\left(Z_h \geq b\right)\right], \qquad (5.54)
\end{aligned}$$

for some positive constant $\widetilde{K}_2$ independent of N and b, thanks to Lemma 5.1.

Let us further define $\overline{N}_A \stackrel{\Delta}{=} \max_{i\in A} N_i^*$. The way we choose the mixing probabilities (see Step 3) and Step 4) in the description of the algorithm in the previous subsection) leads us to the following bound for $\phi\left(\underline{\mathbf{p}}\right)$,

$$0 < \phi\left(\underline{\mathbf{p}}\right) \leq \left(1/p_*\right)^{M\left(\overline{N}_A+1\right)}, \qquad (5.55)$$

where

$$p_* \stackrel{\Delta}{=} \min\left(\min_{k\leq M}\big(p_Z(k), 1-p_Z(k)\big), \min_{i\in A, j\leq N_i(k)}\big(p_{Y_i}(k), 1-p_{Y_i}(k)\big)\right) > 0.$$

Now combining (5.53), (5.54) and (5.55) we conclude that the right hand side of (5.52) can be bounded from above by

$$C_N = \widetilde{K}_1\left(1/p_*\right)^{M\left(\overline{N}_A+1\right)}/\widetilde{K}_2. \qquad (5.56)$$

Consequently,

$$\frac{\widehat{q}_{Z,Y,N}}{\mathbb{P}\left(\tau_A(b) \leq M\right)} \leq 2C_N,$$

with positive constant C_N defined in (5.56). Recall that the number of claims N is Binomial, therefore

$$\widetilde{\mathbb{E}}\Big[\widehat{q}^2_{Z,Y,N}\Big] \leq 2\mathbb{E}\left(C_N^2\right) q^2(b) = O\Big[q^2(b)\Big].$$

And the result follows. □

Proof of Theorem 5.5. From the proof of Proposition 5.2 we know that $\mathcal{A}_{\delta_N}(b) \supseteq \{\tau_A(b) \leq M\}$. And from (5.39), we have

$$\mathbb{P}\left(\tau_A(b) \leq M\right) \leq \mathbb{P}\big(\mathcal{A}_{\delta_N}(b)\big) \leq \kappa_2 b^{-\widetilde{\alpha}^T \mathbf{x}^*},$$

where κ_2 is some positive constant independent of b, and $\mathbf{x}^*$ is an optimal solution to $[IP]$ given in (5.21). On the other hand, from the lower bound in (5.54), it's immediate that

$$\mathbb{P}\left(\tau_A(b) \leq M\right) \geq \widetilde{K}_2 b^{-\widetilde{\alpha}^T \mathbf{x}^*}.$$

Consequently the result follows. □

5.5 Numerical Examples

In this section we illustrate how to apply the simulation strategy described in the previous Section on a simple network consisting of three insurance companies along with one reinsurer, i.e., an example of the star-shaped network considered in Subsection 5.3.2. We assume the factors follow Pareto distributions. In particular,

$$\mathbb{P}\left(Z > z\right) = \left(\frac{\theta}{\theta + z}\right)^{\alpha}, \text{ and } \mathbb{P}\left(Y_i > y\right) = \left(\frac{\theta_i}{\theta_i + y}\right)^{\alpha_i}, i = 1, 2, 3.$$

Model parameters are given in the following table:

Table 5.1: Values of model parameters in numerical examples.

	I_1	I_2	I_3	R	Z
λ	4.0	8.0	16.0		
γ	0.8	0.4	0.2		
β	1.0	1.0	1.0		
θ	100	100	100		100
r	0.8	0.4	0.2	$0.6 \times (0.8 + 0.4 + 0.2)$	

In addition, the premium C and q are set according to the mean aggregate claim sizes $\mathbb{E}X$ and $\mathbb{E}W$, respectively, properly loaded up by an adjustment coefficient equal to 0.5. We take the horizon to be $M = 12$. In other words, claims are aggregated on a monthly basis, and we are evaluating system dislocation in a one-year horizon. We test our simulation strategy with two target sets, $A_1 = \{3\}$, and $A_2 = \{2, 3\}$. For each of target set, we consider the following scenarios, which include all incidents of system configurations discussed in Section 5.3:

1. $\alpha = 2.1, \alpha_1 = 4.9, \alpha_2 = 5.2, \alpha_3 = 6.3$.

2. $\alpha = 6.1, \alpha_1 = 3.9, \alpha_2 = 2.2, \alpha_3 = 3.3$.

3. $\alpha = 3.4, \alpha_1 = 2.1, \alpha_2 = 2.8, \alpha_3 = 2.3$.

The simulation results are demonstrated in Table 5.2 and Table 5.3 below. Each estimate is based on an average over 10^6 replications of the procedure described in the previous section. We report the mean estimate of the probability $q(b) = \mathbb{P}(\tau_A(b) \leq M)$, standard error as a percentage of the probability estimate, as well as the estimate of the Conditional Spillover Loss at System Dislocation of the set A, $CSD(A)$. For moderate values of b we compare our estimates against crude Monte Carlo in order to verify that our implementations are correct. The cost per replication of our importance sampling estimator and that of crude Monte Carlo are very comparable.

From the resulting tables we have a few noteworthy remarks. First of all, the relative stable ratio between the standard error and the mean of the estimates is in line with the

Table 5.2: Numerical results with scenarios 1-3 with $A = \{3\}$.

Secnario # 1.	$b = 10^7$	$b = 10^8$	$b = 10^9$
$\widehat{q}(s.e./\widehat{q}(\%))$	$2.06 \times 10^{-8}\,(0.573\%)$	$1.61 \times 10^{-10}\,(0.574\%)$	$1.30 \times 10^{-12}\,(0.588\%)$
95% $C.I.$	$(2.04, 2.09) \times 10^{-8}$	$(1.60, 1.63) \times 10^{-10}$	$(1.29, 1.32) \times 10^{-12}$
$\widehat{D}(A)(s.e./\widehat{D}(A)(\%))$	$0.2043\,(3.230\%)$	$0.0161\,(1.704\%)$	$1.272 \times 10^{-3}\,(1.684\%)$
95%$C.I.$	$(0.1913, 0.2172)$	$(0.0155, 0.0166)$	$(1.230, 1.314) \times 10^{-3}$
$\widehat{CSD}$	9.902×10^6	9.952×10^7	9.771×10^8
Scenario # 2.	$b = 10^5$	$b = 10^6$	$b = 10^7$
$\widehat{q}(s.e./\widehat{q}(\%))$	$1.72 \times 10^{-8}\,(6.832\%)$	$9.52 \times 10^{-12}\,(3.704\%)$	$4.91 \times 10^{-15}\,(3.492\%)$
95% $C.I.$	$(1.50, 1.94) \times 10^{-8}$	$(0.88, 1.02) \times 10^{-11}$	$(4.58, 5.25) \times 10^{-15}$
$\widehat{D}(A)(s.e./\widehat{D}(A)(\%))$	$6.453 \times 10^{-4}\,(8.115\%)$	$4.415 \times 10^{-6}\,(8.057\%)$	$2.399 \times 10^{-8}\,(6.991\%)$
95%$C.I.$	$(5.427, 7.480) \times 10^{-4}$	$(3.717, 5.112) \times 10^{-6}$	$(2.070, 2.728) \times 10^{-8}$
$\widehat{CSD}$	3.752×10^4	4.636×10^5	4.884×10^6
Scenario # 3.	$b = 10^6$	$b = 10^7$	$b = 10^8$
$\widehat{q}(s.e./\widehat{q}(\%))$	$9.75 \times 10^{-8}\,(1.459\%)$	$5.03 \times 10^{-10}\,(1.438\%)$	$2.55 \times 10^{-12}\,(1.428\%)$
95% $C.I.$	$(0.95, 1.00) \times 10^{-7}$	$(4.89, 5.17) \times 10^{-10}$	$(2.48, 2.62) \times 10^{-12}$
$\widehat{D}(A)(s.e./\widehat{D}(A)(\%))$	$0.0787\,(3.261\%)$	$4.195 \times 10^{-3}\,(4.915\%)$	$2.027 \times 10^{-4}\,(2.958\%)$
95%$C.I.$	$(0.0736, 0.0837)$	$(3.791, 4.599) \times 10^{-3}$	$(1.910, 2.145) \times 10^{-4}$
$\widehat{CSD}$	8.068×10^5	8.335×10^6	7.951×10^7

strong efficiency of the algorithm. In other words, as b increases, it's not necessary to increase the number of replications in order to achieve the same relative accuracy. On the other hand, there is some discernible performance differential across various system configurations, for example, the relative error experiences a deterioration moving from Scenario 1 to Scenario 2. This relates to Remark 5.3. Our explanation is as follows. Recall from Section 5.4 that the dynamic importance sampling scheme is *switched off* as soon as the auxiliary processes hit the initial reserve levels. Under a network setup such as Scenario 2, we know from Section 5.3 that the individual factors of insurer I_3 are most likely the "trouble-makers". However, since at each aggregation period, our uniform algorithm set-up ensures that the common factor is sampled first, before all the individual factors, one or several large common factors thus sampled will very likely inflate the auxiliary process rather quickly, which in turn handicaps the ensuing chances for importance sampling of individual factors; in particular those corresponding to I_3. To put it a different way, sample paths generated from our sampling scheme, although they deviate

Table 5.3: Numerical results with scenarios 1-3 with $A = \{2, 3\}$.

Secnario # 1.	$b = 10^7$	$b = 10^8$	$b = 10^9$
$\widehat{q}(s.e./\widehat{q}(\%))$	$1.03 \times 10^{-8}\,(2.961\%)$	$8.01 \times 10^{-11}\,(2.367\%)$	$6.46 \times 10^{-13}\,(2.898\%)$
95% $C.I.$	$(0.97, 1.09) \times 10^{-8}$	$(7.64, 8.38) \times 10^{-11}$	$(6.10, 6.83) \times 10^{-13}$
$\widehat{D}(A)(s.e./\widehat{D}(A)(\%))$	$0.1906\,(5.063\%)$	$0.0148\,(3.900\%)$	$1.164 \times 10^{-3}\,(3.509\%)$
95%$C.I.$	$(0.1717, 0.2095)$	$(0.0137, 0.0159)$	$(1.084, 1.244) \times 10^{-3}$
$\widehat{CSD}$	1.857×10^7	1.847×10^8	1.801×10^9
Scenario # 2.	$b = 10^5$	$b = 10^6$	$b = 10^7$
$\widehat{q}(s.e./\widehat{q}(\%))$	$9.78 \times 10^{-11}\,(2.90\%)$	$1.09 \times 10^{-16}\,(1.91\%)$	$3.13 \times 10^{-22}\,(1.57\%)$
95% $C.I.$	$(0.92, 1.03) \times 10^{-10}$	$(1.05, 1.13) \times 10^{-16}$	$(3.04, 3.23) \times 10^{-22}$
$\widehat{D}(A)(s.e./\widehat{D}(A)(\%))$	$1.069 \times 10^{-5}\,(4.287\%)$	$1.231 \times 10^{-10}\,(4.151\%)$	$3.664 \times 10^{-15}\,(4.196\%)$
95%$C.I.$	$(0.9787, 1.158) \times 10^{-5}$	$(1.131, 1.331) \times 10^{-10}$	$(3.363, 3.966) \times 10^{-15}$
$\widehat{CSD}$	1.092×10^5	1.134×10^6	1.169×10^7
Scenario # 3.	$b = 10^6$	$b = 10^7$	$b = 10^8$
$\widehat{q}(s.e./\widehat{q}(\%))$	$6.64 \times 10^{-11}\,(5.272\%)$	$2.80 \times 10^{-14}\,(4.249\%)$	$1.03 \times 10^{-17}\,(4.539\%)$
95% $C.I.$	$(5.96, 7.33) \times 10^{-11}$	$(2.57, 3.04) \times 10^{-14}$	$(0.94, 1.12) \times 10^{-17}$
$\widehat{D}(A)(s.e./\widehat{D}(A)(\%))$	$5.538 \times 10^{-5}\,(6.326\%)$	$2.282 \times 10^{-7}\,(4.971\%)$	$8.144 \times 10^{-10}\,(5.475\%)$
95%$C.I.$	$(4.852, 6.225) \times 10^{-5}$	$(2.060, 2.505) \times 10^{-7}$	$(7.270, 9.018) \times 10^{-10}$
$\widehat{CSD}$	8.337×10^5	8.138×10^6	7.923×10^7

from the large deviations description by an acceptable distance to still guarantee bounded relative error, seem to stray a bit farther away from the most likely characterization than those under other configurations. A similar argument explains the trailing performance in Scenario 3 in Table 5.3. A quick and simple solution is to weight the factors corresponding to the "trouble-makes" substantially more than the rest of the other factors. The asymptotically optimal waiting requires explicitly computing the asymptotic conditional distributions of each factor's contribution to the rare event. Since, as we saw in our later sections, this becomes difficult due to the dependence induced another approach could be to use cross-entropy or another adaptive technique as illustrated in [25]. Table 5.2, corresponding to Scenario 2, is produced by assigning a very small weight (equal to 1/100) to the factors that should not contribute to the rare event. Similar improving results have been obtained for Scenario 3 in Table 5.3.

Table 5.4: Comparison of results in Scenario 2, $A = \{3\}$, without/with IS for Z_n switched off.

Before	$b = 10^5$	$b = 10^6$	$b = 10^7$
$\widehat{q}(s.e./\widehat{q}(\%))$	$1.72 \times 10^{-8}\,(6.832\%)$	$9.52 \times 10^{-12}\,(3.704\%)$	$4.91 \times 10^{-15}\,(3.492\%)$
95% *C.I.*	$(1.50, 1.94) \times 10^{-8}$	$(0.88, 1.02) \times 10^{-11}$	$(4.58, 5.25) \times 10^{-15}$
$\widehat{D}(A)(s.e./\widehat{D}(A)(\%))$	$6.453 \times 10^{-4}\,(8.115\%)$	$4.415 \times 10^{-6}\,(8.057\%)$	$2.399 \times 10^{-8}\,(6.991\%)$
95%*C.I.*	$(5.427, 7.480) \times 10^{-4}$	$(3.717, 5.112) \times 10^{-6}$	$(2.070, 2.728) \times 10^{-8}$
$\widehat{CSD}$	3.752×10^4	4.636×10^5	4.884×10^6
After	$b = 10^5$	$b = 10^6$	$b = 10^7$
$\widehat{q}(s.e./\widehat{q}(\%))$	$1.76 \times 10^{-8}\,(3.153\%)$	$1.08 \times 10^{-11}\,(1.856\%)$	$5.13 \times 10^{-15}\,(1.849\%)$
95% *C.I.*	$(1.65, 1.87) \times 10^{-8}$	$(1.04, 1.12) \times 10^{-11}$	$(4.95, 5.32) \times 10^{-15}$
$\widehat{D}(A)(s.e./\widehat{D}(A)(\%))$	$8.109 \times 10^{-4}\,(7.695\%)$	$5.076 \times 10^{-6}\,(3.435\%)$	$2.261 \times 10^{-8}\,(3.236\%)$
95%*C.I.*	$(6.886, 9.332) \times 10^{-4}$	$(4.734, 5.417) \times 10^{-6}$	$(2.118, 2.405) \times 10^{-8}$
$\widehat{CSD}$	4.610×10^4	4.690×10^5	4.405×10^6

5.6 Proofs of Technical Results

Proof of Lemma 5.1. First of all,

$$\mathbb{P}\left(\sum_{i=1}^{N} X_i > b\right) \sim \mathbb{E}N\mathbb{P}\left(X > b\right)$$

results from the well-known properties of subexponential family (see Chapter IX, Lemma 2.2 in [7]), and

$$\mathbb{P}\left(\psi(N)Z > b\right) \sim \mathbb{P}\left(Z > b/\mathbb{E}\psi(N)\right) \tag{5.57}$$

due to Breiman's Theorem (see for example [63]). It remains to show that, if $Y_1 \in \mathcal{RV}(\alpha_1)$, $Y_2 \in \mathcal{RV}(\alpha_2)$ for $\alpha_1, \alpha_2 > 0$, and $\beta, \gamma \geq 0$,

$$\mathbb{P}\big(\beta Y_1 + \gamma Y_2 > b\big) \sim \mathbb{P}\left(\beta Y_1 > b\right) + b\mathbb{P}\left(\gamma Y_2 > b\right),$$

as $b \nearrow \infty$.

The result is trivial if $\beta, \gamma = 0$. Without loss of generality, suppose $\beta, \gamma > 0$. One direction is elementary. For the upper bound, first consider the case where the indices of regularly variation are different, i.e., $\alpha_1 \neq \alpha_2$. Without loss of generality, suppose $\alpha_1 < \alpha_2$. Fix

$\delta \in (0, 1/2)$, note that

$$\mathbb{P}\big(\beta Y_1 + \gamma Y_2 > b\big) \leq \mathbb{P}\big(\beta Y_1 > (1-\delta)b\big) + \mathbb{P}\big(\gamma Y_2 > (1-\delta)b\big) + \mathbb{P}\big(\beta Y_1 > \delta b\big)\mathbb{P}\big(\gamma Y_2 > \delta b\big),$$

Since $\alpha_1 < \alpha_2$, we have $\mathbb{P}\left(\beta Y_1 > (1-\delta)b\right)/\mathbb{P}\left(\gamma Y_2 > b\right) \to 0$, as $b \nearrow \infty$. Therefore

$$\limsup_{b\nearrow\infty} \frac{\mathbb{P}\big(\beta Y_1 + \gamma Y_2 > b\big) - \mathbb{P}\left(\beta Y_1 > b\right)}{\mathbb{P}\left(\gamma Y_2 > b\right)} \leq \limsup_{b\nearrow\infty} \frac{\mathbb{P}\left(\gamma Y_2 > (1-\delta)b\right)}{\mathbb{P}\left(\gamma Y_2 > b\right)} = 1, \tag{5.58}$$

as a result of the property of regular variation.

Now consider the case where $\alpha_1 = \alpha_2 = \alpha$. Let $L_1(\cdot), L_2(\cdot)$ be the slowly varying functions associated with the tail distributions of Y_1, Y_2, respectively. That is, $\mathbb{P}\left(Y_1 > t\right) = t^{-\alpha}L_1(t)$ and $\mathbb{P}\left(Y_2 > t\right) = t^{-\alpha}L_2(t)$. Condition 1 implies that the limit $r = \lim_{t\nearrow\infty} L_1(t)/L_2(t)$ exists. There are two cases:

i) $r < \infty$. Note that

$$\frac{\mathbb{P}\left(\beta Y_1 > (1-\delta)b\right) - \mathbb{P}\left(\beta Y_1 > b\right)}{\mathbb{P}\left(\gamma Y_2 > b\right)} \leq \frac{L_1\left(b/\gamma\right)}{L_2\left(b/\gamma\right)} \frac{L_1\big((1-\delta)b/\beta\big) - L_1\left(b/\beta\right)}{L_1\left(b/\gamma\right)} \to 0,$$

as $b \nearrow \infty$. The upper bound (5.58) follows.

ii) $r = \infty$. In this case consider instead the ratio

$$\frac{\mathbb{P}\big(\beta Y_1 + \gamma Y_2 > b\big) - \mathbb{P}\left(\gamma Y_2 > b\right)}{\mathbb{P}\left(\beta Y_1 > b\right)}.$$

□

Proof of Lemma (5.2). Part 1) is a direct consequence of Breiman's Theorem ([63]). For

part 2), define

$$\mathcal{L}_* \triangleq \{1 \leq i \leq K : \alpha_i = \alpha_*\}.$$

Denote by L_i the slowly varying function associated with the tail distribution function of X_i. Note that by virtue of (5.57), for any $\epsilon > 0$, there exists $b_0 > 0$, such that for $b > b_0$, we have

$$\mathbb{P}\left(\sum_{i=1}^{K}\sum_{j=1}^{N_i} X_{i,j} > b + x \middle| \sum_{i=1}^{K}\sum_{j=1}^{N_i} X_{i,j} > b\right) \leq \frac{\sum_{i=1}^{K} \mathbb{E}N_i \mathbb{P}\left(X_i > b + x\right)}{\sum_{i=1}^{K} \mathbb{E}N_i \mathbb{P}\left(X_i > b\right)} (1 + \epsilon). \quad (5.59)$$

Now, dividing both the denominator and the nominator of the expression on the right hand side by $\mathbb{P}\left(X_{l_*} > b\right)$, $l_* \in \mathcal{L}_*$, the index of any component that has the minimum index α_*, we obtain

$$\frac{\sum_{i=1}^{K} \mathbb{E}N_i \Big(\mathbb{P}\left(X_i > b + x\right)/\mathbb{P}\left(X_{l_*} > b\right)\Big)}{\sum_{i=1}^{K} \mathbb{E}N_i \Big(\mathbb{P}\left(X_i > b\right)\Big)/\mathbb{P}\left(X_{l_*} > b\right)\Big)} = \frac{\sum_{i=1}^{K} \mathbb{E}N_i \left(b + x\right)^{-\alpha_i} b^{\alpha_*} \Big(L_i\left(b + x\right)/L_{l_*}\left(b\right)\Big)}{\sum_{i=1}^{K} \mathbb{E}N_i b^{-(\alpha_i - \alpha_*)} \Big(L_i\left(b\right)/L_{l_*}\left(b\right)\Big)}.$$

Recall that Condition 1 stipulates the existence of the limit $r_i = \lim_{b \nearrow \infty} \big(L_i(b)/L_{l_*}(b)\big), i = 1, 2, \ldots, K$. Also, $L_i(b + x)/L_i(b) \to 1$ as $b \nearrow \infty$, $i = 1, \ldots, K$, thanks to the properties of slowly varying functions $L_i(\cdot)$. As a result, the right hand side of (5.59) is of order

$$\frac{\sum_{i \in \mathcal{L}_*} r_i \mathbb{E}N_i O\left[\big(b/(b + x)\big)^{-\alpha_*}\right]}{\sum_{i \in \mathcal{L}_*} r_i \mathbb{E}N_i} = O\left[\left(\frac{1}{1 + x/b}\right)^{\alpha_*}\right].$$

The other direction is obtained similarly. □

Proof of Theorem 5.2. 1) Uniqueness of Optimality.

i) [Existence of $[P]$-Optimality.] Throughout the proof we shall denote by $\mathbf{e}$ vector of ones, $\mathbf{0}$ vector of zeros, and e_j a vector with all entries zero except for the j-th position, which

has entry one. The dimensions of the matrices and vectors are self-manifest depending on the contexts they appear in. Let us introduce the following matrix notations. Let $\widetilde{\mathbf{u}}$ be an $|\mathcal{I}^+| \times 1$ vector with the i-th entry given by $u_i + \overline{C}_i$, and $\widetilde{\mathbf{u}}^R$ an $|\mathcal{R}^+| \times 1$ vector with the s-th entry given by $u_s^R + Q_s$. Define ϱ to be an $|\mathcal{I}^+| \times |\mathcal{R}^+|$ matrix with the (s,i)-th entry given by ρ_{si}, and define $\widetilde{\varrho}$ to be an $|\mathcal{R}^+| \times |\mathcal{R}^+|$ matrix, with zero diagonals and the (s,s')-th entry being $\widetilde{\rho}_{s's}$, $s' \neq s$. Furthermore, denote by $\vartheta_{\mathcal{R}}$ the diagonal matrix with the s-th diagonal entry being $\sum_{s' \neq s} \widetilde{\rho}_{ss'}$, $s \in \mathcal{R}$. We can therefore express the linear program $[P^{(\kappa)}]$ in the following matrix form:

$$
\begin{aligned}
[P'^{(\kappa)}]: \quad \min \quad & \mathbf{e}^T \pi^- + \xi \mathbf{e}^T \psi^- \\
\text{s.t.} \quad & \pi^+ - \pi^- = \widetilde{\mathbf{u}} - \mathbf{L} - \varrho \psi^- && (\varphi) \\
& \psi^+ - (I + \kappa \vartheta_{\mathcal{R}} - \widetilde{\varrho})\, \psi^- = \widetilde{\mathbf{u}}^R - \mathbf{L}^R && (\eta) \\
& \pi^+, \pi^- \geq \mathbf{0} \\
& \psi^+, \psi^- \geq \mathbf{0}.
\end{aligned}
$$

Here φ and ψ are the dual variables associated with the first two sets of constraints in $[P'^{(\kappa)}]$. Therefore, the dual of $[P'^{(\kappa)}]$ can be formulated as

$$
\begin{aligned}
[D'^{(\kappa)}]: \quad \max \quad & \varphi^T (\widetilde{\mathbf{u}} - \mathbf{L}) + \eta^T \left(\widetilde{\mathbf{u}}^R - \mathbf{L}^R\right) \\
\text{s.t.} \quad & \varphi \leq \mathbf{0} && (\pi^+) \\
& -\varphi \leq \mathbf{e} && (\pi^-) \\
& \eta \leq \mathbf{0} && (\psi^+) \\
& \varphi^T \varrho - \eta^T (I + \kappa \vartheta_{\mathcal{R}} - \widetilde{\varrho}) \leq \xi \mathbf{e}, && (\psi^-)
\end{aligned} \tag{5.60}
$$

Clearly we have $-1 \leq \varphi \leq 0$ and φ_i is bounded, for each $i \in \mathcal{I}^+$. Note that the matrices

ϱ and $\widetilde{\varrho}$ satisfy:

a) $\mathbf{e}^T \varrho + \mathbf{e}^T \widetilde{\varrho} = \mathbf{e}$.

b) $\mathbf{e}^T \left(\kappa \vartheta_{\mathcal{R}} - \widetilde{\varrho}\right) \leq \mathbf{0}$, for $0 \leq \kappa \leq 1$.

Both properties are direct implications from Assumption 5.1-i) and Assumption 5.1-iii). Property b) implies that the matrix $\widetilde{\varrho}$ is sub-stochastic. Along with the fact that the matrix $I + \kappa\vartheta_{\mathcal{R}}$ has spectral radius smaller than one, we obtain the invertibility of the matrix $(I + \kappa\vartheta_{\mathcal{R}} - \widetilde{\varrho})$. On the other hand, it is obvious by virtue of Property b) above that $\mathbf{e}^T \left(I + \kappa\vartheta_{\mathcal{R}} - \widetilde{\varrho}\right) > \mathbf{0}$. Therefore, the vector $\varphi^T \varrho - \xi\mathbf{e} \leq \mathbf{0}$ preserves signs after left multiplying the inverse of the matrix $(I + \kappa\vartheta_{\mathcal{R}} - \widetilde{\varrho})$. Consequently we obtain

$$\left(\varrho^T \varphi - \xi \mathbf{e}^T\right) \times \left(I + \kappa\vartheta_{\mathcal{R}} - \widetilde{\varrho}^T\right)^{-1} \leq \eta \leq \mathbf{0}.$$

As a result the dual problem $[D'^{(\kappa)}]$ is bounded, and since apparently $\eta = \mathbf{0}$ and $\varphi = \mathbf{0}$ is $[D'^{(\kappa)}]$-feasible, the dual $[D'^{(\kappa)}]$ has finite optimal objective value and optimality of $[P^{(\kappa)}]$ follows as a consequence of strong duality (see e.g., [11], Chapter 4). ii) [Uniqueness of $[P^{(\kappa)}]$-optimality.] Let us define

$$\mathbf{d} = \left(d_{\pi^+}, d_{\pi^-}, d_{\psi^+}, d_{\pi^-}\right)^T,$$

i.e., $\mathbf{d}$ is the direction variable corresponding to the $[P^{(\kappa)}]$-solution vector given by $(\pi^+, \pi^-, \psi^+, \psi^-)^T$. And write

$$\mathbf{A} = \begin{bmatrix} \mathbf{0}^T & \mathbf{e}^T & \mathbf{0}^T & \xi\mathbf{e}^T \\ \mathbf{I} & -\mathbf{I} & \mathbf{0} & \varrho \\ \mathbf{0} & \mathbf{0} & \mathbf{I} & -\left(I + \kappa\vartheta_{\mathcal{R}} - \widetilde{\varrho}\right) \end{bmatrix}.$$

It suffices to show that the auxiliary linear program indexed by j, $1 \leq j \leq 2\left(|\mathcal{I}| + |\mathcal{R}^+|\right)$,

$$\begin{array}{lll} [\widehat{P}^{(\kappa)}_{(j)}]: \quad \min & \mathbf{0}^T\mathbf{d} & \\ \text{s.t.} & \mathbf{A}\mathbf{d} = \mathbf{0} & (\mathbf{y}) \\ & \mathbf{d} \geq e_j & (\delta) \end{array}$$

is infeasible for all j. Equivalently, we show that the associated duals $[\widehat{D}^{(\kappa)}_{(j)}]$, given by

$$\begin{array}{lll} [\widehat{D}^{(\kappa)}_{(j)}]: \quad \max & \delta^T e_j & \\ \text{s.t.} & \mathbf{A}^T\mathbf{y} + \delta = \mathbf{0} & (\mathbf{d}) \\ & \delta \geq \mathbf{0} & \end{array}$$

is unbounded for any j, $1 \leq j \leq 2\left(|\mathcal{I}| + |\mathcal{R}^+|\right)$. Indeed, note that if we set $\widehat{\mathbf{y}} = \left(-a, -\mathbf{e}_{|\mathcal{I}|}, -\mathbf{e}_{|\mathcal{R}|}\right)^T$, and $\widehat{\delta} = \left(\mathbf{e}_{|\mathcal{I}|}, (a-1)\,\mathbf{e}_{|\mathcal{I}|}, \mathbf{e}_{|\mathcal{R}|}, a\xi\mathbf{e}_{|\mathcal{R}|} - \kappa\mathbf{e}^T\vartheta_{\mathcal{R}}\right)^T$, the pair $\left(\widehat{\mathbf{y}}, \widehat{\delta}\right)$ is easily shown to be $[D^{(\kappa)}_{(j)}]$-feasible , provided that

$$a > \max\left(1, \kappa/\xi\right),$$

using the property $\mathbf{e}^T\varrho + \mathbf{e}^T\widetilde{\varrho} = \mathbf{e}$. In the meantime, it yields a positive objective value no matter where the index j is. Therefore, the pair $\left(k\widehat{\mathbf{y}}, k\widehat{\delta}\right)$, $\forall k > 0$, is also $[\widehat{D}^{(\kappa)}_{(j)}]$-feasible. The unboundedness of $[\widehat{D}^{(\kappa)}_{(j)}]$ follows. Consequently we conclude that there exists no zero-cost direction for *any* $[P^{(\kappa)}]$-feasible solutions. We have therefore established that $[P^{(\kappa)}]$ entails a unique optimal solution, and that this optimal solution is *non-degenerate*.

2) Insensitivity of Optimality to ξ.

Fix $\kappa \in [0,1]$, let $\left(\check{\pi}^+, \check{\pi}^-, \check{\psi}^+, \check{\psi}^-\right)$ and $(\check{\varphi}, \check{\eta})$ be the optimal solution pair to $[P^{\kappa}]$, when

$\xi = \xi_1 > 0$. The strategy is to construct a feasible solution pair to the primal, $[P^{'(\kappa)}]$ and the dual, $[D^{'(\kappa)}]$ that satisfies complementary slackness, from the solution pair associated with $\xi = \xi_1$, when ξ is changed to $\xi_2 > 0$, $\xi_2 \neq \xi_1$. In order to do so, we first set $\varphi^* = \check{\varphi}$. Then, define

$$t(\xi_2) = \xi_2 \mathbf{e} - (\varphi^*)^T \varrho \geq \xi_2 \mathbf{e} > \mathbf{0},$$

and let $t_s(\xi_2)$ be the s-th element of $t(\xi_2)$, $s \in \mathcal{R}^+$. Now, set $\eta_s^* = \check{\eta}_s / t_s(\xi_2)$. The pair $\left(\check{\pi}^+, \check{\pi}^-, \check{\psi}^+, \check{\psi}^-\right)$ and (φ^*, η^*) is then $[P'^{\kappa}]$-feasible and $[D^{'(\kappa)}]$-feasible, when $\xi = \xi_2$. Moreover, it's not hard to convince ourselves that it satisfies complementary slackness. Therefore $\left(\check{\pi}^+, \check{\pi}^-, \check{\psi}^+, \check{\psi}^-\right)$ is the unique optimal solution to $[P^{'(\kappa)}]$ when $\xi = \xi_2 > 0$ as well. The result follows due to the arbitrariness of ξ_1 and ξ_2. □

Proof of Corollary 5.3. Let $\check{\nu} = \left(\check{\pi}^+, \check{\pi}^-, \check{\psi}^+, \check{\psi}^-\right)$ be the optimal solution to $[P^{(\kappa)}]$. For notational convenience let us define $\widetilde{I} = (I + \kappa \vartheta_{\mathcal{R}} - \widetilde{\varrho})$. Note that the Lagrangian of $[P_f^{(\kappa)}]$ evaluated at $\check{\nu}$ is given by

$$\begin{aligned} L(\check{\nu}, \mu) =& f\left(\check{\pi}^-, \check{\psi}^-\right) + \mathbf{x}^T \left[\widetilde{\mathbf{u}} - \mathbf{L} - \left(\varrho \check{\psi}^- - \check{\pi}^+ + \check{\pi}^-\right)\right] \\ &+ \mathbf{y}^T \left[\widetilde{\mathbf{u}}^R - \mathbf{L}^R - \check{\psi}^+ + \widetilde{I} \check{\psi}^-\right] - \left(\mathbf{z}_{\pi^+}^T \check{\pi}^+ + \mathbf{z}_{\pi^-}^T \check{\pi}^- + \mathbf{z}_{\psi^+}^T \check{\psi}^+ + \mathbf{z}_{\psi^-}^T \check{\psi}^-\right), \end{aligned}$$

where $\mu = (\mathbf{x}, \mathbf{y}, \mathbf{z})$. Here $\mathbf{x}, \mathbf{y}$ and $\mathbf{z} = (\mathbf{z}_{\pi^+}, \mathbf{z}_{\pi^-}, \mathbf{z}_{\psi^+}, \mathbf{z}_{\psi^-})$ are the Lagrange multipliers. The plan is to search for a specific set of Lagrange multipliers, corresponding to each choice of f, such that the resulting vector $\mu^f = \left(\mathbf{x}^f, \mathbf{y}^f, \mathbf{z}^f\right)$ is feasible to the Lagrange dual problem, and the associated solution pair $\left(\check{\nu}, \mu^f\right)$ achieves zero duality gap, which then leads to the $[P_f^{(\kappa)}]$-optimality of $\check{\nu}$, for any f.

We construct such a dual solution vector from the *Karush-Kuhn-Tucker* (KKT) conditions. Note that if $\left(\check{\nu}, \mu^f\right)$ enforces a zero duality gap, the following conditions must

hold (see e.g., [26], Chapter 5):

$$
\begin{aligned}
f'_{\pi^+} - \mathbf{x}^f - \mathbf{z}^f_{\pi^+} &= 0 \\
f'_{\pi^-} + \mathbf{x}^f - \mathbf{z}^f_{\pi^-} &= 0 \\
f'_{\psi^+} - \mathbf{y}^f - \mathbf{z}^f_{\psi^+} &= 0 \qquad (5.61) \\
f'_{\psi^-} - \varrho^T \mathbf{x}^f + \widetilde{I}^T \mathbf{y}^f - \mathbf{z}^f_{\psi^-} &= 0 \qquad (5.62) \\
z_{\pi^+_i}\pi^+_i = z_{\pi^-}\pi^-_i = z_{\psi^+}\psi^+_s = z_{\psi^-}\psi^-_s &= 0,\ i \in \mathcal{I}^+, s \in \mathcal{R}^+ \\
\check{\pi}^+ - \check{\pi}^- &= \widetilde{\mathbf{u}} - \mathbf{L} - \varrho\check{\psi}^- \\
\check{\psi}^+ - \widetilde{I}\check{\psi}^- &= \widetilde{\mathbf{u}}^R - \mathbf{L}^R. \\
\mathbf{z}^f &\geq 0
\end{aligned}
$$

Guided by these conditions we can construct the multipliers in the following way.

i) For each $i \in \mathcal{I}^+$,

   (a) if $\check{\pi}^+_i = 0, \check{\pi}^-_i > 0$, set

$$x^f_i = 0,\ z^f_{\pi^+_i} = 0,\ \text{and}\ z^f_{\pi^-_i} = f'_{\pi^-_i} \geq 0;$$

   (b) if $\check{\pi}^-_i = 0, \check{\pi}^+_i > 0$, then set

$$x^f_i = -f'_{\pi^-_i},\ z^f_{\pi^-_i} = 0,\ \text{and}\ z^f_{\pi^+_i} = f'_{\pi^-_i} \geq 0.$$

ii) Define $\mathcal{D} = \{s \in \mathcal{R}^+, \check{\psi}^-_s > 0\}$. For each $s \in \mathcal{D}$, set $z^f_{\psi^-_s} = 0$, and for each $s \in \overline{\mathcal{D}} = \mathcal{R}^+ \backslash \mathcal{D}$, set $z^f_{\psi^+_s} = 0$.

iii) Note that (5.61) and (5.62) can be expressed as

$$\mathbf{y}^f = -\mathbf{z}^f_{\psi^-},\ \widetilde{I}^T\mathbf{z}^f_{\psi^+} + \mathbf{z}^f_{\psi^-} = f'_{\psi^-} - \varrho^T\mathbf{x}^f. \tag{5.63}$$

Now, without loss of generality we can assume that the index $s \in \mathcal{R}^+$ are aligned such that the first $|\mathcal{D}|$ are all those belonging to $\mathcal{D}$, and the remaining ones belonging to $\overline{\mathcal{D}}$. Let $\mathbf{z}^f_{\psi^+_{\mathbf{D}}}$ be the vector consisting of the first $|\mathcal{D}|$ elements of $\mathbf{z}^f_{\psi^+}$, and $\mathbf{z}^f_{\psi^-_{\overline{\mathbf{D}}^R}}$ be the vector containing the last $|\overline{\mathcal{D}}|$ elements of $\mathbf{z}^f_{\psi^-}$. Define $\widehat{\mathbf{z}}^f_{\psi} = \left[\mathbf{z}^f_{\psi^+_{\mathbf{D}}}; \mathbf{z}^f_{\psi^-_{\overline{\mathbf{D}}}}\right]$, and note that $\mathbf{z}^f_{\psi^-_{\mathbf{D}}} = \mathbf{z}^f_{\psi^+_{\overline{\mathbf{D}}}} = 0$. Furthermore, we can write

$$\mathbf{z}^f_{\psi^+} = P_{\mathcal{D}} \times \widehat{z}^f_{\psi},\ \mathbf{z}^f_{\psi^-} = (I - P_{\mathcal{D}})\,\widehat{z}^f_{\psi},$$

where $P_{\mathcal{D}}$ is an $|\mathcal{R}^+| \times |\mathcal{R}^+|$ diagonal matrix, with the first $|\mathcal{D}|$ diagonal elements equal to one, and the remaining components being zero. It's not hard to recognize that the matrix given by

$$\widetilde{I}^T P_{\mathcal{D}} + (I - P_{\mathcal{D}}) = I + \kappa\vartheta_{\mathcal{R}} P_{\mathcal{D}} - \widetilde{\varrho}^T P_{\mathcal{D}}$$

is invertible, because $I + \kappa\vartheta_{\mathcal{R}} P_{\mathcal{D}}$ has spectral radius smaller than one, and $\widetilde{\varrho}^T P_{\mathcal{D}}$ is sub-stochastic. Therefore, from (5.63) we can set

$$\widehat{\mathbf{z}}^f_{\psi} = \left(\widetilde{I}^T P_{\mathcal{D}} + I - P_{\mathcal{D}}\right)^{-1}\left(f'_{\psi^-} - \varrho^T\mathbf{x}^f\right).$$

Note that $\widehat{\mathbf{z}}^f_{\psi} \geq \mathbf{0}$ because f is increasing in ψ^-_s, $s \in \mathcal{R}^+$, and the multiplier $\mathbf{x}^f$ constructed in i) is non-positive.

Consequently, the vector of multipliers $\mu^f = \left(\mathbf{x}^f, \mathbf{y}^f, \mathbf{z}^f\right)$ constructed from the procedures

above is a feasible solution to the Lagrange dual of $[P_f^\kappa]$. Moreover, it's easy to see that $L\left(\check{\nu}, \mu^f\right) = f\left(\check{\pi}^-, \check{\psi}^-\right)$, i.e., the primal-dual pair, $\left(\check{\nu}, \mu^f\right)$, leads to a zero-duality gap. Strong duality guarantees the $[P_f^\kappa]$-optimality of $\check{\nu}$. The proof is complete. □

Bibliography

[1] Systemic risk in insurance: An analysis of insurance and financial stability. Special Report of The Geneva Association Systemic Risk Working Group, 2010.

[2] R. Adler, J. Blanchet, and J.C. Liu. Efficient simulation of high excursions of gaussian random fields. *Annals of Applied Probability*, To Appear.

[3] H. Amini, R. Cont, and A. Minca. Stress testing the resilience of financial networks. *International Journal of Theoretical and Applied Finance*, 14, 2011.

[4] V. Anantharam, P. Heidelberger, and P. Tsoucas. Analysis of rare events in continuous time marked chains via time reversal and fluid approximation. *IBM Research Report, REC 16280*, 1990.

[5] P. Arbenz and W. Gander. A survey of direct parallel algorithms for banded linear systems. Technical Report 221, Department Informatik,ETH Zurich, 1994.

[6] S. Asmussen. *Applied Probability and Queues*. Wiley, 1987.

[7] S. Asmussen. *Ruin Probabilities*. World Scientific, River Edge, NJ, 2000.

[8] S. Asmussen and P. Glynn. *Stochastic Simulation: Algorithms and Analysis*. Springer-Verlag, New York, NY, USA, 2008.

[9] S. Asmussen and R. Y. Rubinstein. Steady-state rare events simulation in queueing models and its complexity properties. pages 429 – 466, 1995.

[10] O. D. Bandt and P. Hartmann. Systemic risk: A survey. volume 35 of *Working Paper Series*. European Central Bank, Frankfurt, Germany, 2000.

[11] D. Bertsimas and J. N. Tsitsiklis. *Introduction to Linear Optimization*. Athena Scientific, Nashua, U.S.A, 1997.

[12] S. Asmussen K. Binswanger and B. Hojgaard. Rare events simulation for heavy-tailed distributions. *Bernoulli*, 6:303–322, 1997.

[13] J. Blanchet. Optimal sampling of overflow paths in jackson networks. *forthcoming*, 2009.

[14] J. Blanchet, Joshua C.C. Chan, and D.P. Kroese. Asymptotics and fast simulation for tail probabilities of the maximum and minimum of sums of lognormals. working paper, 2010.

[15] J. Blanchet and P. Glynn. Efficient rare-event simulation for the maximum of a heavy-tailed random walk. *Annals of Applied Probability.*, 18:1351–1378, 2008.

[16] J. Blanchet, P. Glynn, and J. C. Liu. Fluid heuristics, lyapunov bounds and efficient importance sampling for a heavy-tailed g/g/1 queue. *QUESTA*, 57:99–113, 2007.

[17] J. Blanchet and H. Lam. State-dependent importance sampling for rare-event simulation: An overview and recent advances. Submitted to Surveys in Operations Research and Management Sciences, 2011.

[18] J. Blanchet, K. Leder, and P. Glynn. Lyapunov functions and subsolutions for rare event simulation. *Preprint*, 2009.

[19] J. Blanchet, K. Leder, and Y. Shi. Analysis of a splitting estimator for rare event probabilities in jackson networks. *Stochastic Systems*, 1:306–339, 2011.

[20] J. Blanchet and C. Li. Efficient rare event simulation for heavy-tailed compound sums. *ACM TOMACS*, 21(2):Article 9, 2011.

[21] J. Blanchet, J. Li, and M. Nakayama. A conditional monte carlo for estimating the failure probability of a network with random demands. In J. Himmelspach K. P. White S. Jain, R. R. Creasey and M. Fu, editors, *Proceedings of the 2011 Winter Simulation Conference*, 2011.

[22] J. Blanchet and J. Liu. Efficient simulation and conditional functional limit theorems for ruinous heavy-tailed random walks. *Stochastic Processes and Their Applications*, 2011.

[23] J. Blanchet and J. C. Liu. State-dependent importance sampling for regularly varying random walks. *Advances in Applied Probability*, 40:1104–1128, 2008.

[24] J. Blanchet and M. Mandjes. Rare event simulation for queues. In G. Rubino and B. Tuffin, editors, *Rare Event Simulation Using Monte Carlo Methods*, pages 87–124. Wiley, West Sussex, United Kingdom, 2009. Chapter 5.

[25] J. Blanchet and Y. Shi. Efficient rare event simulation for heavy-tailed systems via cross entropy. In S. Jain, R. R. Creasey, J. Himmelspach, K. P. White, and M. Fu, editors, *Proceedings of the 2011 Winter Simulation Conference*. IEEE Press, 2011.

[26] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, Cambridge, UK, 2004.

[27] L. Breiman. On some limit theorems similar to the arc-sin law. *Theory of Probability and its Applications.*, 10:323–331, 1965.

[28] J. C. C. Chan, P. W. Glynn, and D. P. Kroese. A comparison of cross-entropy and variance minimization strategies. *Journal of Applied Probability*, 48, 2011.

[29] R. Cont and A. Moussa. Too interconnected to fail: contagion and systemic risk in financial networks. Financial Engineering Report 2009-04, Columbia University, 2009.

[30] R. Cont, A. Moussa, and Edson Bastos e Santos. The brazilian financial system: network structure and systemic risk analysis. Working Paper, 2010.

[31] T. Dean and P. Dupuis. Splitting for rare event simulation: A large deviation approach to design and analysis. *Stochastic Processes and Their Applications*, 119(2):562–587, February 2009.

[32] A. Dembo and O. Zeitouni. *Large deviations techniques and applications*. Springer, New York, second edition, 1998.

[33] P. Dupuis and R. S. Ellis. The large deviation principle for a general class of queueing systems I. *Transactions of the American Mathematical Society*, 347:2689 – 2751, 1995.

[34] P. Dupuis, K. Leder, and H. Wang. Importance sampling for sums of random variables with regularly varying tails. *ACM TOMACS*, 17, 2006.

[35] P. Dupuis, A. Sezer, and H. Wang. Dynamic importance sampling for queueing networks. *Ann. Appl. Probab.*, 17:1306–1346, 2007.

[36] P. Dupuis, A. Sezer, and H. Wang. Subsolutions of an isaacs equation and efficient schemes for importance sampling. *Mathematics of Operations Research*, 32:1–35, 2007.

[37] P. Dupuis and H. Wang. Importance sampling, large deviations, and differential games. *Stoch. and Stoch. Reports*, 76:481–508, 2004.

[38] P. Dupuis and H. Wang. Subsolutions of an Isaacs equation and efficient schemes of importance sampling. *Mathematics of Operations Research*, 32:723–757, 2007.

[39] P. Dupuis and H. Wang. Importance sampling for jackson networks. *Queueing Systems.*, 62(1-2):113–157, 2009.

[40] L. Eisenberg and T. Noe. Systemic risks in financial systems. *Management Science*, 47:236–249, 2001.

[41] P. Embrechts and C. Goldie. On convolution tails. *Stochastic Processes and their Applications*, 13:263–278, 1982.

[42] S. Foss and D. Korshunov. Heavy tails in multi-server queue. *Queueing Systems*, 52:31–48, 2006.

[43] M. J. J. Garvels and D. P. Kroese. A comparison of restart implementations. In *Proceedings of the Winter Simulation Conference*, pages 601–609. IEEE Press, 1998.

[44] P. Glasserman, P. Heidelberger, P. Shahabuddin, and T. Zajic. A large deviations perspective on the efficiency of multilevel splitting. *IEEE Transactions on Automatic Control*, 43(12):1666–1679, 1998.

[45] P. Glasserman, P. Heidelberger, P. Shahabuddin, and T. Zajic. Multilevel splitting for estimating rare event probabilities. *Operations Research*, 47:585 – 600, 1999.

[46] P. Glasserman and S. Kou. Analysis of an importance sampling estimator for tandem queues. *ACM TOMACS*, 5:22–42, 1995.

[47] T. Harris. *The Theory of Branching Processes*. Springer-Verlag, New York, 1963.

[48] H. Hult, F. Lindskog, T. Mikosch, and G. Samordnitsky. Functional large deviations for multivariate regularly varying random walks. *Annals of Applied Probability*, 15:2651–2680, 2005.

[49] I. Ignatiouk-Robert. Large deviations of Jackson networks. *Annals of Applied Probability*, 10:962–1001, 2000.

[50] S. Juneja and V. Nicola. Efficient simulation of buffer overflow probabilities in jackson networks with feedback. *ACM Trans. Model. Comput. Simul.*, 15(4):281–315, 2005.

[51] S. Juneja and P. Shahabuddin. Simulating heavy-tailed processes using delayed hazard rate twisting. *ACM TOMACS*, 12:94–118, 2002.

[52] S. Juneja and P. Shahabuddin. Rare event simulation techniques: An introduction and recent advances. In S. G. Henderson and B. L. Nelson, editors, *Simulation*, Handbooks in Operations Research and Management Science, pages 291–350. Elsevier, Amsterdam, The Netherlands, 2006. Chapter 2.

[53] H. Kahn and T.E. Harris. Estimation of particle transmission by random sampling. *National Bureau of Standard Applied Mathematics Series.*, 12:27–30, 1951.

[54] H. Kellerer, U. Pferschy, and D. Pisinger. *Knapsack Problems*. Springer-Verlag, Berlin-Heidelberg, 2004.

[55] D. Kroese and V. Nicola. Efficient simulation of a tandem jackson network. *ACM Trans. Model. Comput. Simul.*, 12:119–141, 2002.

[56] D. P. Kroese, R. Y. Rubinstein, and P. W. Glynn. The cross-entropy method for estimation. In V. Govindaraju and C. R. Rao, editors, *Handbook of Statistics*, volume 31. Elsevier, 2010.

[57] K. Majewski and K. Ramanan. How large queues build up in a Jackson network. *To Appear in Mathematics of Operations Research*, 2008.

[58] M.Villen-Altamirano and J. Villen-Altamirano. Restart: A method for accelerating rare even simulations. In J.W. Colhen and C.D. Pack, editors, *Proceedings of the 13th International Teletraffic Congress. In Queueing, performance and control in ATM*, pages 71–76. Elsevier Science Publishers, 1991.

[59] V. Nicola and T. Zaburnenko. Efficient importance sampling heuristics for the simulation of population overflow in jackson networks. *ACM Trans. Model. Comput. Simul.*, 17(2), 2007.

[60] S. Parekh and J. Walrand. Quick simulation of rare events in networks. *IEEE Transactions of Automatic Control*, 34:54–66, 1989.

[61] E. J. G. Pitman. Subexponential distribution functions. *J. Austral. Math. Soc. Ser. A.*, 29:337 – 347, 1980.

[62] Swiss Re. Reinsurance - a systemic risk? Sigma, 2003.

[63] S. I. Resnick. *Heavy Tail Phenomena: Probabilistic and Statistical Modeling*. New York, 2006.

[64] P. Robert. *Stochastic Networks and Queues*. Springer-Verlag, Berlin, 2003.

[65] L. C. G. Rogers and L. A. M. Veraat. Failure and rescue in an interbank network. Working Paper, 2011.

[66] R. Y. Rubinstein and D. P. Kroese. *The Cross-Entropy Method*. Springer, New York, NY, 2004.

[67] A. Schwartz and A. Weiss. *Large Deviations for Performance Analysis*. Chapman and Hall, London, 1995.

[68] A. D. Sezer. Modeling of an insurance system and its large deviations analysis. *Journal of Computational and Applied Mathematics*, 235(3):535 – 546, 2010.

[69] I. van Lelyveld, F. Liedorp, and M. Kampman. An empirical assessment of reinsurance risk. *Journal of Financial Stability*, 7(4):191 – 203, 2011.

[70] M. Villén-Altamirano and J. Villén-Altamirano. Restart: a straightforward method for fast simulation of rare events. In *Winter Simulation Conference*, pages 282–289, 1994.

[71] B. Zwart, S. Borst, and M. Mandjes. Exact asymptotics for fluid queues fed by multiple heavy-tailed on-off flows. *The Annals of Applied Probability*, 14:903 – 957, 2004.