EFFICIENT STRATIFIED SAMPLING FOR FINANCIAL RISK SIMULATION

by

İsmail Başoğlu

B.S., Industrial Engineering, Yıldız Technical University, 2006

M.S., Industrial Engineering, Boğaziçi University, 2008

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Philosophy of Doctorate

Graduate Program in FBE Program for which the Thesis is Submitted

Boğaziçi University

2014

ACKNOWLEDGEMENTS

I would like to gratefully acknowledge the enthusiastic supervision of Assoc. Prof. Wolfgang Hörmann. His orientation, guidance, insightful criticisms and patient encouragement contributed very much for the realization of this thesis.

I would also like to acknowledge the support of TÜBİTAK, the unique research foundation in Turkey, with the Domestic Doctoral Study Scholarship Program.

During this work, I have collaborated with Assist. Prof. Halis Sak and Kemal Dinçer Dingeç, PhD., for whom I have great regard and I want to express that I thank much to those who have helped me with my work besides providing a stimulating environment. I also feel grateful to Prof. Refik Güllü for his valuable suggestions and academic mentorship during my study in Boğaziçi University.

Finally, I want to specially thank to my favorite supporters at every time: my beautiful wife Şeyma Boydak Başoğlu, my dear mum Mahmude Başoğlu, dear dad Hayri Başoğlu, and other family members, Nalan Gürşahbaz, Nazan Uyanık, Emrah Gürşahbaz, and Fatih Uyanık.

ABSTRACT

EFFICIENT STRATIFIED SAMPLING FOR FINANCIAL RISK SIMULATION

Monte Carlo simulation is frequently the only method available for computing financial risk, particularly under the realistic and complex portfolio models. The naive simulation generally leads to large confidence intervals on typical risk measures. Thus, to enhance the efficiency of the estimates, the necessity of variance reduction techniques becomes apparent. In this thesis, we discuss the efficient implementation of stratified sampling technique for Monte Carlo simulation problems. As the application field, we consider the risk evaluation of a linear asset portfolio. For given portfolio and the loss threshold, tail loss probability and conditional excess values are essential. To understand the general risk situation, one needs to efficiently estimate these values for multiple threshold levels in a single simulation. Stratified sampling is especially useful for such a task as the allocation fractions can be used as decision variables to minimize the overall error of all estimates. We develop an efficient simulation algorithm that combines optimal stratification and importance sampling to estimate multiple tail loss probabilities and conditional excess values for linear asset portfolios under the t-copula model. Two different classes of objective functions are proposed to represent the overall error. The first, including the mean squared and the mean squared relative error, allows for a simple closed-form solution. For the second class of error functions, including the maximal absolute and the maximal absolute relative error, a simple and fast heuristic is proposed. The application of the new method, called "OASIS: optimal allocation stratification and importance sampling", is explained for linear asset portfolios under the t-copula model and its performance is demonstrated with numerical examples.

ÖZET

FİNANSAL RİSK BENZETİMİ İÇİN ETKİN KATMANLI ÖRNEKLEME

Monte Karlo benzetimi, özellikle gerçekçi ve karmaşık portföy modelleri için finansal risk hesaplamasında kullanılabilecek tek yöntem olarak karşımıza çıkar. Sade simulasyon, tipik risk ölçüleri için genellikle geniş güven aralıkları oluşturur. Bu nedenle, hesaplanan değerlerin etkinliğini arttımak için varyans azaltma tekniklerine duyulan ihtiyaç ön plana çıkmaktadır. Bu tezde, Monte Karlo benzetim problemleri için katmanlı örnekleme tekniğinin etkin uygulaması incelenmektedir. Uygulama alanı olarak, doğrusal varlık portföyünün risk değerlemesi düşünülmüştür. Verilmiş bir portföy ve kayıp eşiği için, kuyruk kayıp olasılıkları ve koşullu kayıp değerleri önem taşımaktadır. Genel risk durumunu anlamak için bu değerlerin tek bir benzetimde birden farklı eşik değeri için hesaplanması gerekmektedir. Örneklem yerleşim fraksiyonlarının hesaplanan değerlerin toplam hatasının enküçüklemesinde karar değişkeni olarak kullanılabilmesi, katmanlı örneklemeyi bu iş için kullanışlı kılmaktadır. Çalışmada, t-kopula ile modellenen doğrusal varlık portföyleri için birden fazla kuyruk kayıp olasılığı ve koşullu kayıp değerini hesaplamak üzere, optimal katmanlandırma ve önem örneklemesinin birleşiminden oluşan etkin bir benzetim yöntemi geliştirilmiştir. Toplam hatayı temsil etmek üzere iki amaç fonkisyon sınıfı önerilmiştir. Birincisi, ortalama hata karesi ve ortalama göreceli hata karesini içermekte ve kapalı formül çözümlere imkan tanımaktadır. En büyük mutlak ve en büyük mutlak göreceli hatayı içeren ikinci amaç fonksiyon sınıfı için basit ve hızlı bir sezgisel çözüm yöntemi sunulmuştur. "Optimal yerleşimli katmanlandırma ve önem örneklemesi" adı verilen yeni yöntemin t-kopula ile modellenen doğrusal varlık portfoyleri için uygulaması izah edilmiş ve performansı sayısal örneklerle kanıtlanmıştır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$\boldsymbol{A}$	Premultiplier matrix
$Bias$	Bias of an estimator
$\boldsymbol{B}$	An arbitrary matrix
$\boldsymbol{b}$	An arbitrary vector
b_i	Strata boundaries in the unit interval
c	Piecewise constant function
$\boldsymbol{c}$	Vector of scaling factors
c_d	Scaling factor of the d-th logreturn
c_j	Coefficients for the diagonal elements of the variance covariance matrix $\boldsymbol{\Sigma}$
c_{jk}	Coefficients for the elements of the variance covariance matrix $\boldsymbol{\Sigma}$
C_f	Family of densities constructed over f with different allocation fractions
CEx	Conditional excess
$CVaR$	Conditional value-at-risk
D	Dimension of the random input
E	Expectation under the default distribution
E_f	Expectation under the specified density f
Eff	Efficiency of an estimator
ER	Efficiency ratio of an estimator
F	Cumulative distribution function of an arbitrary random variable
$F_{\chi^2_\nu}$	Cumulative distribution function of chi-squared distribution with degrees of freedom D
F_i	Cumulative distribution function of the i-th element of the random input vector
F_ν	Cumulative distribution function of the t-distribution with degrees of freedom ν
F_Γ	Cumulative distribution function of gamma distribution

f	Probability density function of the random input
f_{CE}	Cross-entropy optimal density
f_{IS}	Importance sampling density
f_{IS}^*	Optimal importance sampling density
f_{OAS}	The density of the optimally allocated stratified sample
g	A continuous function of the variance covariance matrix $\mathbf{\Sigma}$
G_d	Cumulative distribution function of the marginal distribution of the d-th stock
H_i	Sample set of stratum i
H_{i1}	Numerator sample set of stratum i
H_{i2}	Denominator sample set of stratum i
H_{ij}	Sample set of the j-th response in stratum i
H_{ij1}	Numerator sample set of the j-th response in stratum i
H_{ij2}	Denominator sample set of the j-th response in stratum i
I	Number of strata
I_d	Number of strata defined for the d-th dimension of the random input
i	Index of strata
i_d	Index of strata defined for the d-th dimension of the random input
J	Number of responses to be calculated in a single simulation
j	Index of responses
j^c	Index of the current move
j'	Buffer index
K	Number of iterations
k	Index of iterations, also used as a dummy index
l	Dummy index
$\boldsymbol{L}$	Lower triangular Cholesky factorization of the correlation matrix
$Loss$	Portfolio loss
M_i^k	Sample size accumulated in the i-th stratum in the k-th iteration

MSE	Mean-square error of an estimator
N	Total sample size in a simulation
N_i	Sample size allocated to the i-th stratum
N_i^k	Sample size allocated to the i-th stratum in the k-th iteration
N^k	Sample size used in the k-th iteration
N_p	Sample size used in the pilot study
$Normal\,(0,1)$	Standard normal distribution
$Normal\,(0,\mathbf{I}_D)$	D dimensional standard multinormal distribution
n	Index of random drawings
$\boldsymbol{p}$	Vector of strata probabilities
p_i	Probability of the i-th stratum
q	Simulation function
q_{IS}	Simulation function of importance sampling
$\mathbb{R}$	Set of real numbers
$q_{\boldsymbol{V}}$	Simulation function using linear transformation over basis $\boldsymbol{V}$
R	Radius of the hypersphere
$\boldsymbol{R}$	Correlation matrix of the stock logreturns in the multinormal model
$\mathbb{R}$	Set of real numbers
$\mathbb{R}^+$	Set of positive real numbers
r	Response function
r_0	Root of an equation
RE	Percentage relative error
$Return$	Portfolio return
S	Stratum index function or stratification function
$\boldsymbol{S}$	Stratification function
S_0	Initial amount of investment
$\hat{s}$	Sample standard deviation
$\hat{s}^2$	Sample variance
s_i	Standard deviation conditional on the i-th stratum
$\hat{s}_i$	Sample standard deviation of the i-th stratum

s_i^2	Variance of the simulation function conditional on the i-th stratum
$\hat{s}_{ij}^2$	Sample variance of the i-th stratum under the j-th parameter
$\hat{s}_{ijk}$	Covariance of the samples under the j-th and the k-th parameters in the i-th stratum
$\hat{s}_i^k$	Sample standard deviation of the i-th stratum calculated in the k-th iteration
$\hat{s}_{i1}^k$	Numerator sample standard deviation of the i-th stratum calculated in the k-th iteration
$\hat{s}_{i2}^k$	Denominator sample standard deviation of the i-th stratum calculated in the k-th iteration
$\hat{s}_{i12}^k$	Sample covariance of the numerator and the denominator samples of the i-th stratum calculated in the k-th iteration
$\hat{s}_{ij}^k$	Sample standard deviation of the j-th response in the i-th stratum calculated in the k-th iteration
$\hat{s}_{ij1}^k$	Numerator sample standard deviation of the j-th response in the i-th stratum calculated in the k-th iteration
$\hat{s}_{ij2}^k$	Deominator sample standard deviation of the j-th response in the i-th stratum calculated in the k-th iteration
$\hat{s}_{ij12}^k$	Sample covariance of the numerator and the denominator samples of the j-th response in the i-th stratum calculated in the k-th iteration
$\boldsymbol{T}$	Multi-t vector
T_d	d-th element of the multi-t vector $\boldsymbol{T}$
TM	Computation time of an estimator
TLP	Tail loss probability
U	Standard uniform variable
$\boldsymbol{U}$	Standard uniform vector
$\boldsymbol{u}$	Parameter vector of the original sampling density
U_d	d-th element of the standard uniform vector $\boldsymbol{U}$
$Unif\,(0,1)$	Standard uniform distribution
$Unif\,(par_1, par_2)$	Uniform distribution in the interval (par_1, par_2)
V	Variance under the default distribution

$\boldsymbol{V}$	Linear transformation matrix that holds stratification directions in the columns
$\boldsymbol{V}^*$	Optimal linear transformation matrix
$\bar{\boldsymbol{V}}$	The matrix that holds Δ orthogoal stratification directions in the columns
$\boldsymbol{v}$	Stratification direction
$\boldsymbol{v}_{init}$	Initial direction
$\boldsymbol{v}^*$	Optimal stratification direction
$\bar{\boldsymbol{v}}$	An arbitrary vector
$\boldsymbol{v}_d$	d-th orthogonal stratification direction
$\boldsymbol{v}_d^*$	d-th optimal orthogonal stratification direction
v_{dl}	The element of the linear transformation matrix in the d-th row and the l-th column
V_f	Variance under under the specified density f
VaR	Value-at-risk
var_d	Variance of the marginal distribution of he d-th stock
VR	Variance reduction factor
$\boldsymbol{w}$	Vector of the fraction of investments
w_d	Fraction of investment on the d-th stock
$\boldsymbol{X}$	Random input vector or the logreturn vector
X_d	Logreturn of the d-th stock in the portfolio
$\boldsymbol{X}_i$	Random input vector conditional to the i-th stratum
$\boldsymbol{X}^n$	n-th independent drawing of $\boldsymbol{X}$
$\boldsymbol{X}_i^n$	n-th independent drawing of $\boldsymbol{X}_i$
$\boldsymbol{X}_{IS}$	Random input vector following importance sampling density
$\boldsymbol{X}_{IS}^n$	n-th independent drawing of $\boldsymbol{X}_{IS}$
$\boldsymbol{x}$	An arbitrary real vector
Y	An arbitrary random variable
$\boldsymbol{Y}$	An arbitrary random vector
$\boldsymbol{Y}_d$	d-th element of the arbitrary random vector
y	Unknown expected value of the simulation function

y_i	Expected value of the simulation function conditional on the i-th stratum
$\hat{y}_1^k$	Estimator of the numerator in the k-th iteration
$\hat{y}_2^k$	Estimator of the denominator in the k-th iteration
$\hat{y}_i$	Sample mean of the i-th stratum
y_j	Unknown expected value of the simulation function under the j-th parameter
$\hat{y}_j$	An estimator of y_j
$\hat{y}_{j1}$	An estimator of y_{j1}
$\hat{y}_{j2}$	An estimator of y_{j2}
$\hat{\boldsymbol{y}}$	Vector of multiple estimates
$\hat{y}_{ij}$	Sample mean of the i-th stratum under the j-th parameter
$\hat{y}_i^k$	Sample mean of the i-th stratum in the k-th iteration
$\hat{y}_{i1}^k$	Numerator sample mean of the i-th stratum in the k-th iteration
$\hat{y}_{i2}^k$	Denominator sample mean of the i-th stratum in the k-th iteration
$\hat{y}_{ij}^k$	Sample mean of the j-th response in the i-th stratum in the k-th iteration
$\hat{y}_{ij1}^k$	Numerator sample mean of the j-th response in the i-th stratum in the k-th iteration
$\hat{y}_{ij2}^k$	Denominator sample mean of the j-th response in the i-th stratum in the k-th iteration
$\hat{y}_j^k$	Estimator of the j-th response in the k-th iteration
$\hat{y}_{j1}^k$	Estimator of the j-th numerator response in the k-th iteration
$\hat{y}_{j2}^k$	Estimator of the j-th denominator response in the k-th iteration
$\hat{y}^{AOA}$	Adaptive optimal allocation estimator of y
$\hat{y}^{IS}$	Importance sampling estimator of y
$\hat{y}^{NV}$	Naive Monte Carlo estimator of y
$\hat{y}^{OASIS}$	OASIS estimator of y
$\hat{y}_j^{OASIS}$	OASIS estimator of y_j
$\hat{y}^{STRS}$	Stratified sampling estimator of y

$\boldsymbol{Z}$	Standard multinormal vector
$\bar{\boldsymbol{Z}}$	Multinormal vector conditional on a stratum in Δ dimensional real number space
$\tilde{\boldsymbol{Z}}$	Multinormal vector conditional on a stratum in D dimensional real number space
z	An arbitrary real number
$\boldsymbol{z}$	An arbitrary real vector
z_d	d-th element of the arbitrary real vector $\boldsymbol{z}$
Z_d	d-th element of the standard multinormal vector
$\boldsymbol{Z}_{\boldsymbol{\Lambda}}$	Multinormal vector with zero mean and variance covariance matrix $\boldsymbol{\Lambda}$
α	Given probability or significance level
χ^2_ν	Chi-squared random variable with ν degrees of freedom
Δ	Dimension of the stratification vector
δ	Cross-entropy distance
Φ	Cumulative distribution function of standard normal distribution
ϕ	Probability density function of the standard normal distribution
γ^*	Optimal IS scale parameter for the t-copula model
η	Iteration count
$\mathbf{I}_D$	D dimensional identity matrix
λ	Lagrange multiplier
λ_d	d-th eigen value of $\boldsymbol{\Lambda}$
λ_j	j-th coefficient of the convex combination
$\boldsymbol{\Lambda}$	Variance covariance matrix or the correlation matrix of the t-copula model
$\boldsymbol{\mu}$	Mean vector of stock logreturns in multinormal model
$\boldsymbol{\mu}^*$	Optimal IS shift for the t-copula model
μ_d	d-th element of $\boldsymbol{\mu}$
ν	Degrees of freedom

O	Big O notation
$\boldsymbol{\pi}$	Vector of allocation fractions
$\boldsymbol{\pi}^c$	Current solution for the vector of allocation fractions
$\boldsymbol{\pi}^h$	Best heuristic solution for the vector of optimum allocation fractions
$\boldsymbol{\pi}^j$	Vector of optimum allocation fractions for the j-th response
π_i	Allocation fraction of the i-th stratum
π_i^j	Optimum allocation fraction in the i-th stratum for the j-th response
π_i^*	Optimal allocation fraction of the i-th stratum
π_i^k	Allocation fraction of the i-th stratum in the k-th iteration
Θ	The parameter space of the response function
$\boldsymbol{\theta}$	Vector of parameters used in the response function
$\boldsymbol{\theta}_j$	Parameter vector of the j-th response
ρ	Likelihood ratio of importance sampling
$\boldsymbol{\Sigma}$	Variance covariance matrix of the simulation responses
Σ_{jk}	The element of the variance covariance matrix $\boldsymbol{\Sigma}$ in the j-th row and in the k-th column
$\boldsymbol{\sigma}$	Volatility vector of stock logreturns in multinormal model
σ_d	d-th element of $\boldsymbol{\sigma}$
τ	Loss threshold
τ_j	j-th loss threshold
τ_{min}	Minimum of the loss thresholds
τ_{max}	Maximum of the loss thresholds
τ^*	Threshold for determining IS parameters in the multiresponse setting of OASIS
$\boldsymbol{\upsilon}$	Parameter vector of the importance sampling density
$\boldsymbol{\upsilon}^{CE}$	Cross-entropy optimal parameter vector
ω	Objective function that represents the overall error of the simulation
ω^*	Optimal objective value
ω^{**}	Suboptimal objective value

$\hat{\omega}$	Dummy objective value
ω^c	Current objective value
ω^h	Best objective value of the heuristic
ω'	Buffer objective value
$\hat{\omega}_j$	Objective function that represents the error of the j-th simulation response
ω_{MAXE}	Objective function that represents the maximum absolute error of all estimates
ω_{MAXR}	Objective function that represents the maximum absolute relative error of all estimates
ω_{MSE}	Objective function that represents the mean squared error of all estimates
ω_{MSR}	Objective function that represents the mean squared relative error of all estimates
ω^*_{MSR}	Minimum mean squared relative error of all estimates
ω_{SUM}	Objective function that sums the al elements in the variance covariance matrix $\mathbf{\Sigma}$
ξ_i	i-th stratum of the sampling domain
ζ	An arbitrary value in the unit interval

LIST OF ACRONYMS/ABBREVIATIONS

AOA	Adaptive optimal allocation
BFGS	Broyden-Fletcher-Goldfarb-Shanno algorithm
CDF	Cumulative distribution function
CE	Cross-entropy
CLT	Central limit theorem
GH	Generalized hyperbolic
iid	Independent and identically distributed
IS	Importance sampling
KNITRO	Nonlinear interior point trust region optimization
MAXE	Maximum absolute error
MAXR	Maximum absolute relative error
MRS	Multiresponse stratified sampling
MSE	Mean squared error
MSE	Mean squared relative error
NV	Naive Monte Carlo
OAS	Optimal allocation stratification
OASIS	Optimal allocation stratification and importance sampling
PAS	Proportional allocation stratification
PASIS	Proportional allocation stratification and importance sampling
PDF	Probability density function
PINV	Polynomial Inversion
STRS	Stratified sampling
VaR	Value-at-risk

1. INTRODUCTION

The essence of control in the financial world relies on how well the used models mimic reality and on the precision of the computational methods used. As the models get more realistic and thus complex, the variety of applicable computational methods decreases. Nevertheless, simulation is often one of the best alternatives as it leads to confidence intervals on the pin-point values. In fact, Monte Carlo simulation is widely used in estimating the profit and loss distribution of a portfolio and thus in computing risk measures that summarize this distribution (Glasserman, 2004).

To enhance the efficiency of Monte Carlo simulation, one should perform techniques to reduce the variance of Monte Carlo estimators to obtain narrower confidence intervals. Well-known variance reduction techniques are conditional Monte Carlo (Trotter and Tukey, 1956), antithetic variates (Hammersley and Morton, 1956), control variates[1] , Latin hypercube sampling (McKay *et al.*, 1979), importance sampling (Kahn and Marshall, 1953), and stratified sampling (Cochran, 1977). While the main objective of these methods are the same, their basics, ways of implementation and complexities are different. Moreover, each method has certain advantages and disadvantages when implemented to different type of problems.

This thesis is mainly focused on stratified sampling (STRS) and its combination with importance sampling (IS). In both methods, the minimal variance is obtained by directing sampling effort towards the most important regions of the sampling domain, i.e., the regions which contribute a larger share to the output variance. Therefore, they are most effective in rare event simulation problems where we need to observe an unlikely event in order to estimate the quantity of interest. In these type of simulation problems, naive Monte Carlo simulation yields a poor estimate since the rare event may occur only a few times or not at all.

[1]Boyle *et al.* (1997) state that the earliest application of control variates to option pricing is Boyle (1977).

In fact, estimation of financial risk measures via Monte Carlo is a typical rare event simulation problem. One value of interest is the tail loss probability, which is the probability that the loss of the portfolio falls below a certain threshold level. Value-at-risk, as a risk measure, is defined as the $1-\alpha$ quantile of the loss distribution for a given probability α. There are many publications proposing efficient methods for risk estimation under different portfolio models. Glasserman *et al.* (2002) develop an efficient method for simulating the tail loss probabilities in option portfolios using delta-gamma approximations. Kang and Shahabuddin (2005) and Basamboo *et al.* (2008) use importance sampling for the estimation of multi-factor portfolio credit risk. Although value-at-risk estimation is mentioned, the main objective of these studies is to reduce the variance in tail loss probability estimates, as the calculation of value-at-risk can be realized by interpolating the tail loss probabilities of several threshold values.

In that context, estimation of multiple values from a single simulation, i.e., multiresponse simulation is of practical importance. Such an objective can easily be realized using common random numbers (see e.g., Law, 2014). However, under this simple approach, all estimates yield large variances. Variance reduction techniques can be utilized to obtain more accurate estimates for multiple values. Among the ones known in the literature, stratified sampling can be especially useful for such a task as the sample allocation fractions can be used as decision variables to minimize the overall error of all estimates.

There are studies on the statistical estimation of multiple values in a population using stratified sampling (see e.g., Díaz-García and Cortez, 2006, 2008; Miller *et al.*, 2007; Khowaja *et al.*, 2012). However, the potential use of stratified sampling in multiresponse simulation is overlooked in the literature. The idea is first introduced in the progress of this thesis - and in Başoğlu *et al.* (2013) - for minimizing the maximum relative error of multiple tail loss probability estimates for a linear asset portfolio. Later, in Başoğlu and Hörmann (2014), the idea is generalized for different type of simulation problems with a wider class of objective functions that represent the overall error of

the simulation. These objective functions contain the mean squared (relative) error and the maximum absolute (relative) error of all estimates. The resulting method is called "Multiresponse Stratified Sampling" (MRS) and it exhibits a good performance in simulation problems for which stratification reaches a good variance reduction in the single response case. The analogy of MRS can also be used for minimizing the variance of single or multiple ratio estimators.

Stratified sampling - or MRS - can be combined with IS to enhance the efficiency of the estimates further. One of the objectives of the thesis is to understand why and how this combination works well. Glasserman (2004) points out that STRS can be considered as a form of IS since the total stratified sample follows different distributions under different sample allocations. In other words, STRS is an automatic implementation of IS restricted to a large family of sampling densities, which are constructed over the original density by changing the allocation fractions. Using this analogy, the reason why STRS and IS work well when they are combined is explained and demonstrated with a numerical example.

The main objective of this thesis is to develop a combined algorithm based on optimal allocation stratification (OAS) and IS - abbreviated as OASIS - that works efficiently for both estimating single and multiple estimates. To demonstrate the practical use and the efficiency of the OASIS algorithm, the problem of risk estimation in linear asset portfolios is considered. As for the values to be estimated, we consider tail loss probability and the conditional excess, i.e., the expected portfolio loss given that the loss exceeds a threshold level. Conditional excess has a ratio estimator and its variance can be minimized using the analogy of MRS.

In the context of financial simulation, the combination of IS and stratification was only considered in Glasserman *et al.* (1999) for option pricing and Glasserman *et al.* (2000) for the risk quantification of an asset portfolio. The former paper develops heuristics for optimal stratification of a single estimate whereas the latter considers the multiple estimates case but does not use optimal allocation for the stratification.

Neither of them tries a joint minimization of the errors for several estimates.

For a linear asset portfolio, a realistic assumption on the joint distribution of the logarithmic asset returns is the t-copula model. Copulas are a popular way of modeling dependence among financial variables and the necessity of using copulas is stressed by many authors (see e.g., Frey and McNeil, 2001; Embrechts *et al.*, 2002). Mashal *et al.* (2003) suggest that the t-copula fits empirically better than the Gaussian copula without assuming specific marginal distributions for asset returns. In a more recent work, Kole *et al.* (2007) apply goodness-of-fit tests to the t, Gaussian and Gumbel copula for the risk management of linear asset portfolios. The t-copula is reported to have a better fit than the Gaussian and Gumbel copulas because it captures the dependence in the tails better. According to these studies, the t-copula model is currently one of the most flexible and realistic models for the joint distribution of logarithmic asset returns.

Unfortunately, the t-copula model does not allow the simple calculation of tail loss probabilities. Therefore, Sak *et al.* (2010) developed an efficient IS method to estimate a single tail loss probability of a linear asset portfolio under the t-copula model. For estimating multiple tail loss probabilities and conditional excess values under the t-copula model, the OASIS algorithm combines the IS algorithm of Sak *et al.* (2010) and stratified sampling under optimal allocation. The optimal allocation fractions are determined by using the adaptive optimal allocation (AOA) algorithm suggested by Étoré and Jourdain (2010) which works iteratively. In each iteration, a portion of the sample size is allocated optimally based on the information gathered from the total sample allocated in previous iterations.

The rest of this thesis is organized as follows: Chapter 2 presents a literature review on importance sampling and stratified sampling, explaining some of the automatic implementations of these methodologies such as the cross-entropy approach (see e.g., Rubinstein and Kroese, 2008) and the AOA algorithm of Étoré and Jourdain (2010). In Chapter 3, typical strata structures for different types of random inputs are explained

in detail. Chapter 4 describes how to implement OAS for estimating multiple values in a single simulation. In this context, the MRS methodology of Başoğlu and Hörmann (2014) is explained. In Chapter 5, a discussion is held on how OAS and IS enhance the efficiency of the estimates when they are combined. Financial risk simulation is explained in Chapter 6 along with the implementation details of the OASIS algorithm for the t-copula model. Chapter 7 covers the numerical experiments on the developed OASIS algorithm. Finally, in Chapter 8, concluding remarks are provided.

Note that, in this thesis, vectors and matrices are set in bold to enhance readability.

2. VARIANCE REDUCTION FOR RARE EVENT SIMULATION

In this section, we mainly follow Glasserman (2004), Rubinstein and Kroese (2008), and Lemieux (2009) to give the general description of the variance reduction techniques which form the basis of this thesis. In Section 2.1, we describe general definition of the Monte Carlo problem along with the Naive Monte Carlo simulation. Then, in Section 2.2 and 2.3, we give the idea and the implementation details of IS and STRS. We remind that both of these techniques are capable of reducing the variance of the estimators in rare event simulations.

2.1. Naive Monte Carlo Simulation

Let $\boldsymbol{X} \in \mathbb{R}^D$ be a random vector with density f, and $q : \mathbb{R}^D \to \mathbb{R}$ be a measurable function[2] such that $E_f\left[q^2(\boldsymbol{X})\right] < \infty$. Our goal is to compute the unknown value

$$y = E_f\left[q\left(\boldsymbol{X}\right)\right] = \int\limits_{\boldsymbol{x}\in\mathbb{R}^D} q\left(\boldsymbol{x}\right) f\left(\boldsymbol{x}\right) d\boldsymbol{x}. \tag{2.1}$$

If the integrand in Equation 2.1 is simple, then we can calculate a closed-form solution for y. However, in most situations, the closed-form solution is not available. In these situations, an estimate for y can be obtained by Monte Carlo simulation. The naive Monte Carlo algorithm generates an independent and identically distributed (iid) random sample $\boldsymbol{X}^1, \ldots, \boldsymbol{X}^N$ from density f and returns the naive estimator

$$\hat{y}^{NV} = N^{-1} \sum_{n=1}^{N} q\left(\boldsymbol{X}^n\right),$$

which is unbiased.

[2]Henceforth, q will be called the simulation function.

The variance of the naive estimator is $V\left[\hat{y}^{NV}\right] = N^{-1}V_f\left[q(\boldsymbol{X})\right]$ and according to the central limit theorem (CLT), the asymptotic distribution of

$$\frac{\hat{y}^{NV} - y}{\sqrt{V_f\left[q\left(\boldsymbol{X}\right)\right]/N}} \Rightarrow Normal\left(0,1\right)$$

is standard normal. An unbiased estimator of $V_f\left[q(\boldsymbol{X})\right]$ is the sample variance of $q\left(\boldsymbol{X}^1\right), \ldots, q\left(\boldsymbol{X}^N\right)$, namely:

$$\hat{s}^2 = \left(N-1\right)^{-1}\sum_{n=1}^{N}\left(q\left(\boldsymbol{X}^n\right) - \hat{y}^{NV}\right)^2.$$

Using the sample standard deviation $\hat{s}$, we can construct a confidence interval for y:

$$\hat{y}^{NV} \pm \Phi^{-1}\left(1-\alpha/2\right)\frac{\hat{s}}{\sqrt{N}}, \tag{2.2}$$

where Φ^{-1} is the inverse of the cumulative distribution function (CDF) of standard normal distribution and $1-\alpha$ is the confidence level.

A simple way to reduce the variance of the naive Monte Carlo estimator $\hat{y}^{NV}$ is to increase the total sample size N. By this way, the error bound of the confidence interval in Equation 2.2 will decrease with order $\mathrm{O}\left(N^{-1/2}\right)$. However, since the expected time of the simulation algorithm will increase linearly in N, the efficiency of the naive estimator will not be affected by this approach.

To find more efficient estimators than the naive Monte Carlo estimator, one must reduce the variance of the estimator with a factor that must be greater than the factor of increase in the expected simulation time (Lemieux, 2009). The idea of variance reduction is to find another simulation function whose expectation is the same as $E_f\left[q\left(\boldsymbol{X}\right)\right]$ but whose variance is smaller than $V_f\left[q\left(\boldsymbol{X}\right)\right]$.

2.2. Importance Sampling

It is stated by several authors (see e.g, Koopman *et al.*, 2009; Durbin and Koopman, 2012) that IS was introduced first in Kahn and Marshall (1953) and Marshall (1956), and it was first described in the book of (Hammersley and Handscomb, 1964, Section 5.4). Rubinstein and Kroese (2008) describe IS as the most fundamental variance reduction technique, as it quite often leads to a dramatic variance reduction, in particular when estimating rare event probabilities.

IS involves choosing a new density f_{IS} such that $f_{IS}(\boldsymbol{x}) \neq 0$ for any $\boldsymbol{x} \in \mathbb{R}^D$ for which $f(\boldsymbol{x}) \neq 0$. Then, the expectation in Equation 2.1 can be represented as:

$$\begin{aligned} E_f\left[q\left(\boldsymbol{X}\right)\right] &= \int\limits_{\boldsymbol{x}\in\mathbb{R}^D} q\left(\boldsymbol{x}\right) f\left(\boldsymbol{x}\right) d\boldsymbol{x} = \int\limits_{\boldsymbol{x}\in\mathbb{R}^D} q\left(\boldsymbol{x}\right) \rho\left(\boldsymbol{x}\right) f_{IS}\left(\boldsymbol{x}\right) d\boldsymbol{x} \\ &= E_{f_{IS}}\left[q\left(\boldsymbol{X}\right) \rho\left(\boldsymbol{X}\right)\right] = E_{f_{IS}}\left[q_{IS}\left(\boldsymbol{X}\right)\right], \end{aligned}$$

where $q_{IS}\left(\boldsymbol{x}\right) = q\left(\boldsymbol{x}\right)\rho\left(\boldsymbol{x}\right)$ and $\rho\left(\boldsymbol{x}\right) = f\left(\boldsymbol{x}\right)/f_{IS}\left(\boldsymbol{x}\right)$ denotes the likelihood ratio or the Radon-Nikodym derivative of the two densities at $\boldsymbol{x}$. Then, one can generate an iid random sample $\boldsymbol{X}_{IS}^1, \ldots, \boldsymbol{X}_{IS}^N$ from density f_{IS} and evaluate the IS estimator

$$\hat{y}^{IS} = N^{-1} \sum_{n=1}^{N} q_{IS}\left(\boldsymbol{X}_{IS}^n\right) = N^{-1} \sum_{n=1}^{N} q\left(\boldsymbol{X}_{IS}^n\right) \rho\left(\boldsymbol{X}_{IS}^n\right),$$

which is asymptotically normal (see Glasserman, 2004, page 256). The IS estimator $\hat{y}^{IS}$ is unbiased if the IS density $f_{IS}\left(\boldsymbol{x}\right)$ has higher tails than $\left|q\left(\boldsymbol{x}\right) f\left(\boldsymbol{x}\right)\right|$ (see Rubinstein and Kroese, 2008, page 131).

The variance of the IS estimator is $V\left[\hat{y}^{IS}\right] = N^{-1} V_{f_{IS}}\left[q_{IS}\left(\boldsymbol{X}\right)\right]$ where:

$$V_{f_{IS}}\left[q_{IS}\left(\boldsymbol{X}\right)\right] = E_{f_{IS}}\left[q^2\left(\boldsymbol{X}\right) \frac{f^2\left(\boldsymbol{X}\right)}{f_{IS}^2\left(\boldsymbol{X}\right)}\right] - E_{f_{IS}}\left[q\left(\boldsymbol{X}\right) \frac{f\left(\boldsymbol{X}\right)}{f_{IS}\left(\boldsymbol{X}\right)}\right]^2. \tag{2.3}$$

It is easy to see that the subtracted expression in Equation 2.3 is equal to y^2. By the

change of the measure, the first expression in Equation 2.3 can be written as:

$$E_{f_{IS}}\left[q^2\left(\boldsymbol{X}\right)\frac{f^2\left(\boldsymbol{X}\right)}{f_{IS}^2\left(\boldsymbol{X}\right)}\right] = E_f\left[q^2\left(\boldsymbol{X}\right)\frac{f\left(\boldsymbol{X}\right)}{f_{IS}\left(\boldsymbol{X}\right)}\right] < \infty,$$

where the last inequality is a necessary condition to have a bounded variance for the IS estimator. This suggests that, preferably, the likelihood ratio $\rho\left(\boldsymbol{x}\right) = f\left(\boldsymbol{x}\right)/f_{IS}\left(\boldsymbol{x}\right)$ should be bounded (see Rubinstein and Kroese, 2008, page 133).

As a consequence, the variance of the IS estimator can be rewritten:

$$V\left[\hat{y}_{IS}\right] = N^{-1}\left(E_f\left[q^2\left(\boldsymbol{X}\right)\frac{f\left(\boldsymbol{X}\right)}{f_{IS}\left(\boldsymbol{X}\right)}\right] - y^2\right),$$

and it is smaller than the naive Monte Carlo estimator when:

$$E_f\left[q^2\left(\boldsymbol{X}\right)\frac{f\left(\boldsymbol{X}\right)}{f_{IS}\left(\boldsymbol{X}\right)}\right] \leq E_f\left[q^2\left(\boldsymbol{X}\right)\right].$$

2.2.1. Optimal Importance Sampling Density

The optimal choice of the IS density is

$$f_{IS}^*\left(\boldsymbol{x}\right) = \frac{\left|q\left(\boldsymbol{x}\right)f\left(\boldsymbol{x}\right)\right|}{\int\limits_{\boldsymbol{x}\in\mathbb{R}^D}\left|q\left(\boldsymbol{x}\right)f\left(\boldsymbol{x}\right)\right|d\boldsymbol{x}}, \tag{2.4}$$

which minimizes $V_{f_{IS}}\left[q_{IS}\left(\boldsymbol{X}\right)\right]$ (see Rubinstein and Melamed, 1998). In particular, if $q\left(\boldsymbol{x}\right)$ is non-negative, the importance sampling estimator yields zero variance under the optimal IS density given in Equation 2.4 (Rubinstein and Kroese, 2008).

Since the denominator in Equation 2.4 is unavailable, it is impossible to obtain f_{IS}^* in practice. Even if the analysis above gives us some intuition to guide us in our choice of the IS density f_{IS}, there is, generally, no way of constructing a probability density function that will achieve the largest variance reduction, or even to construct

one that will guarantee a variance reduction (Lemieux, 2009). Nevertheless, we might be able to choose an IS density that imitates the function $|q(\boldsymbol{x})f(\boldsymbol{x})|$.

In practice, the IS density can be selected in a process of trial and error guided by the following general rules:

- The IS density $f_{IS}(\boldsymbol{x})$ should mimic the behavior of $|q(\boldsymbol{x})f(\boldsymbol{x})|$.
- The IS density $f_{IS}(\boldsymbol{x})$ must have higher tails than $|q(\boldsymbol{x})f(\boldsymbol{x})|$.
- The likelihood ratio $\rho(\boldsymbol{x})$ should be bounded.

If f is a parametric density function, an approach to ease the process of identifying a good IS density is to restrict our attention to the same distribution family with different parameters. The parameters of the new distribution are chosen so that the variance in Equation 2.3 is reduced.

Once the choice of the IS density is restricted to the same distribution family, a practical solution is to choose the IS parameters such that the mode of the IS density coincides with the mode of the optimal IS density f_{IS}^* (see e.g., Glasserman *et al.*, 1999; Sak *et al.*, 2010). This approach finds a suboptimal but effective IS density with a less computational cost.

2.2.2. Cross-entropy Approach

An alternative approach is to choose the importance sampling density f_{IS} such that the cross-entropy (CE) distance between the optimal density f_{IS}^* in Equation 2.4 and f_{IS} is minimal (Rubinstein and Kroese, 2008). The CE distance between two densities f_{IS}^* and f_{IS} is given by:

$$\begin{aligned}\delta(f_{IS}^*, f_{IS}) &= E_{f_{IS}^*}\left[\ln \tfrac{f_{IS}^*(\boldsymbol{X})}{f_{IS}(\boldsymbol{X})}\right] = \int\limits_{\boldsymbol{x}\in\mathbb{R}^D} f_{IS}^*(\boldsymbol{x}) \ln \tfrac{f_{IS}^*(\boldsymbol{x})}{f_{IS}(\boldsymbol{x})} d\boldsymbol{x} \\ &= \int\limits_{\boldsymbol{x}\in\mathbb{R}^D} f_{IS}^*(\boldsymbol{x}) \ln f_{IS}^*(\boldsymbol{x})\, d\boldsymbol{x} - \int\limits_{\boldsymbol{x}\in\mathbb{R}^D} f_{IS}^*(\boldsymbol{x}) \ln f_{IS}(\boldsymbol{x})\, d\boldsymbol{x},\end{aligned} \tag{2.5}$$

where the first expression is independent of the choice of the IS density. We are interested in $f_{CE} = \arg\min \{f_{IS} : \delta(f_{IS}^*, f_{IS})\}$. Here, f_{CE} is called CE optimal density. If we are in an unconstrained function space, then $f_{CE} = f_{IS}^*$. However, if we restrict ourselves to a parametric family of densities, then the problem turns out to be a minimization problem in the parameter space (see Rubinstein and Kroese, 2008). In that case, the corresponding density function would be suboptimal in variance minimization.

In fact, in literature, most IS densities are obtained by changing the parameter of the original sampling density f (see e.g., Glasserman *et al.*, 1999, 2000, 2002; McLeish, 2010; Sak *et al.*, 2010; Başoğlu *et al.*, 2013). Let $\boldsymbol{u}$ denote the vector of original parameters which are subject to change in IS and let $\boldsymbol{v}$ be the vector of new parameters for the IS density. Then, the CE optimization problem turns into:

$$\boldsymbol{v}^{CE} = \arg\min \{\boldsymbol{v} : \delta(f_{IS}^*, f(.;\boldsymbol{v}))\},$$

where $\boldsymbol{v}^{CE}$ is called the CE optimal parameter vector. Since the first expression of the CE distance in Equation 2.5 is independent of $\boldsymbol{v}$, the CE optimal parameter vector can be found by:

$$\max_{\boldsymbol{v}} \int_{\boldsymbol{x}\in\mathbb{R}^D} f_{IS}^*(\boldsymbol{x}) \ln f(\boldsymbol{x};\boldsymbol{v})\, d\boldsymbol{x}. \tag{2.6}$$

Under the assumption that $q(\boldsymbol{x}) \geq 0$, the expression in Equation 2.6 is equivalent to:

$$\max_{\boldsymbol{v}} \int_{\boldsymbol{x}\in\mathbb{R}^D} q(\boldsymbol{x}) f(\boldsymbol{x};\boldsymbol{u}) \ln f(\boldsymbol{x};\boldsymbol{v})\, d\boldsymbol{x} = E_{f(.;\boldsymbol{u})}[q(\boldsymbol{x}) \ln f(\boldsymbol{x};\boldsymbol{v})].$$

The solution for this problem can be approximated by:

$$\max_{\boldsymbol{v}} \sum_{n=1}^{N_p} q(\boldsymbol{X}^n) \ln f(\boldsymbol{X}^n;\boldsymbol{v}),$$

using an iid pilot sample $\boldsymbol{X}^1, \ldots, \boldsymbol{X}^{N_p}$ of size N_p generated from the original density

$f(.;\boldsymbol{u})$ (see e.g., Rubinstein and Kroese, 2008).

2.3. Stratified Sampling

The earliest use of stratification is to increase the precision of population estimates. In statistical surveys, when sub-populations within an overall population vary, it is advantageous to sample each sub-population independently. Stratification is the process of dividing members of the population into homogeneous subgroups (strata) before sampling. The strata should be mutually exclusive (every element in the population must be assigned to only one stratum) and also collectively exhaustive (no population element can be excluded). Then, simple random sampling or systematic sampling is applied within each stratum. This often improves the representativeness of the sample by reducing sampling error and may produce a weighted mean that has less variability than the arithmetic mean of a simple random sample of the population.

As a variance reduction technique, it contains the ideas that are used in statistical sampling (Cochran, 1977). The main idea is to partition the sample space into I disjoint and covering subsets (strata) and estimate y using the information in each stratum.

Let ξ_i, $i = 1, \ldots, I$ be a partition of $\mathbb{R}^D$ into I strata and $p_i = \Pr\{\boldsymbol{X} \in \xi_i\}$ is known for $i = 1, \ldots, I$. We are interested in the estimation of

$$y = E_f\left[q\left(\boldsymbol{X}\right)\right] = \sum_{i=1}^{I} p_i E_f\left[q\left(\boldsymbol{X}\right)|\boldsymbol{X} \in \xi_i\right]$$

based on the latter equality. Let $\boldsymbol{X}_i$ denote the random vector that follows the conditional distribution of $\boldsymbol{X}$ given $\boldsymbol{X} \in \xi_i$ and

$$y_i = E_f\left[q\left(\boldsymbol{X}\right)|\boldsymbol{X} \in \xi_i\right] = E_f\left[q\left(\boldsymbol{X}_i\right)\right]$$

denote the expectation conditional on the i-th stratum.

The stratified Monte Carlo estimator is calculated using a total sample of size N. Let N_i be the amount of drawings allocated to stratum i such that $N = \sum_{i=1}^{I} N_i$, and $\boldsymbol{X}_i^n$, $n = 1, \ldots, N_i$ be the independent drawings of $\boldsymbol{X}_i$. Then, the unbiased Monte Carlo estimator $\hat{y}_i$ of the conditional expectation y_i is:

$$\hat{y}_i = N_i^{-1} \sum_{n=1}^{N_i} q\left(\boldsymbol{X}_i^n\right), \quad i = 1, \ldots, I,$$

and the stratified Monte Carlo estimator $\hat{y}^{STRS}$ of y is:

$$\hat{y}^{STRS} = \sum_{i=1}^{I} p_i \hat{y}_i.$$

The conditional expectation estimates $\hat{y}_i$, $i = 1, \ldots, I$ and $\hat{y}^{STRS}$ are asymptotically normal for large values of N_i and N, respectively (Glasserman, 2004).

In order to calculate the variance of the stratified Monte Carlo estimator, let s_i^2 denote the variance of $q(\boldsymbol{X})$ conditional on the i-th stratum, namely:

$$s_i^2 = V\left[q\left(\boldsymbol{X}_i\right)\right] = V_f\left[q\left(\boldsymbol{X}\right) | \boldsymbol{X} \in \xi_i\right].$$

Then, the variance of the stratified estimator is:

$$V\left[\hat{y}^{STRS}\right] = \sum_{i=1}^{I} p_i^2 \frac{s_i^2}{N_i}. \tag{2.7}$$

2.3.1. Optimal Allocation Stratification

To achieve a maximum efficiency with stratified sampling, two questions must be solved:

- How should we decompose $\mathbb{R}^D$ into I strata?

- For given strata, how should we select the sample sizes for the different strata?

Unfortunately, the answer of the first question depends both on the type of the random input $\boldsymbol{X}$ and on the simulation function q. To see this, we define the stratum index function $S(\boldsymbol{X}) = i$ if $\boldsymbol{X} \in \xi_i$. Using the conditional variance formula, the variance of $q(\boldsymbol{X})$ can be decomposed into two parts:

$$V_f[q(\boldsymbol{X})] = V[E_f[q(\boldsymbol{X})|S(\boldsymbol{X})]] + E[V_f[q(\boldsymbol{X})|S(\boldsymbol{X})]]. \tag{2.8}$$

The variance of the stratified estimator in Equation 2.7 uses only the conditional variances. Thus, it is only influenced by the second component of the decomposition in Equation 2.8. Therefore, we need to choose strata such that they maximize the first component of the decomposition in Equation 2.8. On the other hand, the strata should be computationally tractable, so that we can generate all conditional vectors $\boldsymbol{X}_i$, $i = 1, \ldots, I$ easily. This issue will be handled in detail in Chapter 3.

Once the strata are fixed, the second question can be answered by defining allocation fractions, $\pi_i = N_i/N$, the portion of the sample to be allocated in stratum i. We rewrite the variance of the stratified estimator in terms of allocation fractions:

$$V\left[\hat{y}^{STRS}\right] = \sum_{i=1}^{I} p_i^2 \frac{s_i^2}{N_i} = \frac{1}{N} \sum_{i=1}^{I} \frac{p_i^2 s_i^2}{\pi_i} \geq \frac{1}{N} \left(\sum_{i=1}^{I} p_i s_i \right)^2. \tag{2.9}$$

The lower bound given in Equation 2.9 is an immediate result of Jensen's inequality (1906), since:

$$\sum_{i=1}^{I} \frac{p_i^2 s_i^2}{\pi_i} = \sum_{i=1}^{I} \left(\frac{p_i s_i}{\pi_i} \right)^2 \pi_i \geq \left(\sum_{i=1}^{I} \left(\frac{p_i s_i}{\pi_i} \right) \pi_i \right)^2 = \left(\sum_{i=1}^{I} p_i s_i \right)^2.$$

For the choice of allocation fractions, we have two practical possibilities. The first one is proportional allocation, i.e. to choose allocation fractions π_i equal to p_i. With

proportional allocation stratification (PAS), the variance of the stratified estimator will be:

$$V\left[\hat{y}^{STRS}\right] = \frac{1}{N}\sum_{i=1}^{I}\frac{p_i^2 s_i^2}{p_i} = N^{-1}\sum_{i=1}^{I} p_i s_i^2 = N^{-1}E\left[V_f\left[q\left(\boldsymbol{X}\right)|S\left(\boldsymbol{X}\right)\right]\right]$$

which is clearly less than or equal to the variance of the naive estimator, since it uses only the second component of the variance decomposition in Equation 2.8. However, it is also greater than or equal to the lower bound given in Equation 2.9, again as a result of Jensen's inequality:

$$\sum_{i=1}^{I} p_i s_i^2 \geq \left(\sum_{i=1}^{I} p_i s_i\right)^2.$$

The second choice is the optimal allocation (also called Neyman allocation) stratification (OAS). We can find values π_i for which $V\left[\hat{y}^{STRS}\right]$ is minimized. To do this, we need to solve the optimization problem:

$$\begin{array}{ll} \min & \sum_{i=1}^{I}\pi_i^{-1}p_i^2 s_i^2 \\ \text{s.t.} & \sum_{i=1}^{I}\pi_i = 1, \quad \text{and} \quad \pi_i > 0, \quad i = 1,\ldots,I \end{array}$$

We can use a Lagrange multiplier λ and rewrite the problem as:

$$\begin{array}{ll} \min & \sum_{i=1}^{I}\pi_i^{-1}p_i^2 s_i^2 + \lambda\left(\pi_1 + \ldots + \pi_I - 1\right) \\ \text{s.t.} & \pi_i > 0, \quad i = 1,\ldots,I \end{array}$$

The gradient of the objective function with respect to $\boldsymbol{\pi} = (\pi_1,\ldots,\pi_I)'$ is:

$$\left(-\pi_1^{-2}p_1^2 s_1^2 + \lambda, \ldots, -\pi_I^{-2}p_I^2 s_I^2 + \lambda\right)$$

which is equal to 0 for:

$$\pi_i = \lambda^{-1/2} p_i s_i, \quad i = 1, \ldots, I$$

with $\lambda = \left(\sum_{i=1}^{I} p_i s_i \right)^2$, so that $\sum_{i=1}^{I} \pi_i = 1$. Thus, for fixed strata, the optimal allocation fractions are:

$$\pi_i^* = \frac{p_i s_i}{\sum_{l=1}^{I} p_l s_l}, \quad i = 1, \ldots, I \tag{2.10}$$

which, in fact, attains the lower bound given in Equation 2.9.

The main drawback of OAS is that the calculation of optimal fractions requires prior information on the conditional standard deviations s_i, $i = 1, \ldots, I$. Therefore, it is not possible to determine the optimal allocation fractions directly. A practical approach is to use the estimates $\hat{s}_i$ of conditional standard deviations s_i obtained through a pilot study with N_p replications. Then, in the main run, the remaining $N - N_p$ replications can be allocated using the estimates of the optimal allocation fractions in Equation 2.10. At the end of the simulation, the sample generated in the pilot study is combined with the sample generated in the main run. By this approach, no drawings are wasted.

Using this idea, Étoré and Jourdain (2010) propose an adaptive optimal allocation (AOA) algorithm that works iteratively. In each iteration, it modifies the proportion of further drawings by using conditional standard deviation estimates. These proportions converge to the optimal allocation fractions through the iterations. Étoré and Jourdain (2010) show that the stratified estimator of the AOA algorithm is asymptotically normal and its asymptotic variance is minimal. We use their algorithm with some minor modifications. In Section 2.3.2, we provide the implementation details of our version of the AOA algorithm.

2.3.2. The Modified Adaptive Optimal Allocation Algorithm

The AOA algorithm terminates in K iterations, each of which is denoted by the index $k = 1, \ldots, K$. The total sample size is denoted by N and N^k is the sample size used in iteration k. Let N_i^k be the size of the sample drawn in iteration k conditional on stratum i. Then, the following equation holds:

$$N = \sum_{k=1}^{K} N^k = \sum_{k=1}^{K} \sum_{i=1}^{I} N_i^k. \tag{2.11}$$

Étoré and Jourdain (2010) suggest using an increasing sequence of N^k, $k = 1, \ldots, K$. In their numerical experiments, they allocate 10%, 40% and 50% of the total sample size in $K = 3$ iterations, sequentially.

In the first iteration, the allocation fractions are selected as $\pi_i^1 = p_i$ and N_i^1, $i = 1, \ldots, I$ are determined proportionally. However, the sample size in each stratum must be an integer and they should satisfy (2.11). Therefore, small variations may occur between N_i^1 and the allocated sample sizes. Our first modification is that we do not try to satisfy $N^1 = \sum_{i=1}^{I} N_i^1$ and let $N_i^1 = \max\left\{\lceil p_i N^1 \rceil, 10\right\}$. Thus, the size of the sample allocated in the first and also in any iteration does not necessarily match exactly the aimed sample size. At the end of the first iteration, the standard deviation conditional on stratum i is estimated as $\hat{s}_i^1$ and the allocation fractions of the next iteration π_i^2 are calculated using the following general formula.

$$\pi_i^k = \frac{p_i \hat{s}_i^{k-1}}{\sum_{l=1}^{I} p_l \hat{s}_l^{k-1}}, \quad i = 1, \ldots, I, \quad k = 2, \ldots, K. \tag{2.12}$$

Finally, the allocation sizes of the next iteration are determined using:

$$N_i^k = \max\left\{\left\lceil \pi_i^k N^k \right\rceil, 10\right\}, \quad i = 1, \ldots, I, \quad k = 1, \ldots, K. \tag{2.13}$$

Étoré and Jourdain (2010) suggest allocating at least one drawing to each stratum

to ensure the asymptotic convergence of the stratified estimates. However, in our extensive experiments with tail loss simulation, we have observed that this often leads to suboptimal allocations in early iterations and, thus, to poor variance estimates and coverage probabilities for the calculated confidence intervals. Therefore, as a second modification, we increase the minimum allocation size to 10 as in Equation 2.13 to ensure better variance estimates and coverage probabilities.

At the end of iteration k, the conditional standard deviations s_i, $i = 1, \ldots, I$ are estimated with $\hat{s}_i^k$ by using also the drawings of earlier iterations. With this approach, the asymptotic optimality of the allocation sizes is guaranteed and no drawings are wasted. Each drawing made in stratum i is collected in set H_i. Let $\hat{y}_i^k$ be the mean of the sample that is collected in H_i at the end of iteration k. Then, the stratified estimator of the final iteration is calculated using:

$$\hat{y}^{AOA} = \sum_{i=1}^{I} p_i \hat{y}_i^K. \tag{2.14}$$

Let M_i^k be the size of the sample accumulated in H_i at the end of iteration k, namely $M_i^k = \sum_{k=1}^{K} N_i^k = M_i^{k-1} + N_i^k$, with $M_i^0 = 0$. Then, $\hat{y}^{AOA}$ is an unbiased estimator of y with variance:

$$V\left[\hat{y}^{AOA}\right] = \sum_{i=1}^{I} \frac{p_i^2 s_i^2}{M_i^K} \tag{2.15}$$

which can be estimated by replacing s_i with the conditional standard deviation estimates $\hat{s}_i^K$ of the final iteration.

The proofs for the asymptotic optimality of the allocation sizes and the asymptotic convergence of the stratified estimate are given in Étoré and Jourdain (2010).

The pseudo code of the modified AOA algorithm is given in Figure 2.1

Require: Simulation function $q : \mathbb{R}^D \to \mathbb{R}$; density function of the random input f; strata ξ_i of the sampling domain and respective probabilities p_i, $i = 1, \ldots, I$; number of iterations K; the aimed sample sizes in each iteration N^k, $k = 1, \ldots, K$

Ensure: Stratified estimator $\hat{y}^{AOA}$ and its variance $V\left[\hat{y}_{AOA}\right]$

set $M_i^0 = 0$ and $\pi_i^1 = p_i$ for $i = 1, \ldots, I$
for iteration $k = 1, \ldots, K$ **do**
  if $k \geq 2$ **then**
    calculate π_i^k for $i = 1, \ldots, I$ using Eq. 2.12
  end if
  for stratum index $i = 1, \ldots, I$ **do**
    calculate N_i^k using Eq. 2.13 and set $M_i^k = N_i^k + M_i^{k-1}$
    for drawing $n = 1, \ldots, N_i^k$ **do**
      generate $\boldsymbol{X}$ from density f conditional on $\boldsymbol{X} \in \xi_i$
      compute $q(\boldsymbol{X})$ add it to set H_i
    end for
    set $M_i^k = M_i^{k-1} + N_i^k$
    compute sample standard deviation $\hat{s}_i^k$ in set H_i
    if $k = K$ **then**
      compute sample mean $\hat{y}_i^k$ in set H_i
    end if
  end for
end for
compute and return $\hat{y}^{AOA}$ and $V\left[\hat{y}^{AOA}\right]$ using respectively Eq. 2.14 and 2.15

Figure 2.1. The modified AOA algorithm.

3. EFFICIENT STRATA STRUCTURES

In Section 2.3.1, we have mentioned that the decomposition of the sampling domain into strata is an essential issue in stratified sampling. We have stated that the strata should be chosen in such a way that the variance of the conditional expectations is large. We have also stated that it should be easy to generate random input from the conditional distribution in each stratum.

To simplify these objectives, we define the stratification function $\boldsymbol{S}(\boldsymbol{X})$, as a surjective mapping from $\mathbb{R}^D$ onto $\mathbb{R}^\Delta$ where $\Delta \leq D$, and stratify each element of $\boldsymbol{S}(\boldsymbol{X})$ independently. For this reason, we have the following requirements for the stratification function $\boldsymbol{S}(\boldsymbol{X})$: The elements of $\boldsymbol{S}(\boldsymbol{X})$ must be independent and their marginal distributions should be available, and it must be possible to sample $\boldsymbol{X}$ from its original distribution conditional on any stratum of $\boldsymbol{S}(\boldsymbol{X})$.

We rewrite the conditional variance formula of Equation 2.8

$$V_f\left[q\left(\boldsymbol{X}\right)\right] = V\left[E_f\left[q\left(\boldsymbol{X}\right)|\boldsymbol{S}\left(\boldsymbol{X}\right)\right]\right] + E\left[V_f\left[q\left(\boldsymbol{X}\right)|\boldsymbol{S}\left(\boldsymbol{X}\right)\right]\right]. \tag{3.1}$$

Among the possible choices of $\boldsymbol{S}$, we choose one for which $V\left[E_f\left[q\left(\boldsymbol{X}\right)|\boldsymbol{S}\left(\boldsymbol{X}\right)\right]\right]$ is large, as this component of the variance is removed by the stratified estimator. Moreover, as each element of $\boldsymbol{S}(\boldsymbol{X})$ is stratified independently, it is better to choose a projection to small dimensions ($\Delta = 1$, 2, or rarely 3) to avoid a large number of strata.

In Section 3.1, we explain how to stratify the random input $\boldsymbol{X}$ (or $\boldsymbol{S}(\boldsymbol{X})$) through uniform random variables. Then, we give possible projection forms for stratification under multivariate uniform, multivariate normal, and multivariate t-distributed random variables in Section 3.2, 3.3, and 3.4, respectively .

3.1. Stratification Through Uniform Random Variables

It is possible to stratify random variables through stratified uniform random variables using the inverse CDF of the marginal distributions. This also helps to define equiprobable strata for the subject random variable, since equiprobable strata are equivalent to equidistant strata for the uniform distribution.

Suppose we are given a decomposition of the unit interval into I intervals, namely $[b_{i-1}, b_i)$, $i = 1, \ldots, I$ with $b_0 = 0$ and $b_I = 1$. Using a standard uniform variable $U \sim Unif\,(0,1)$, the transformation below generates a uniform random variable Y in $[b_{i-1}, b_i)$.

$$Y = b_{i-1} + (b_i - b_{i-1})\, U. \tag{3.2}$$

A continuous random variable X can be stratified using its inverse CDF F^{-1} as long as it is available. Given stratum probabilities $\boldsymbol{p} = (p_1, \ldots, p_I)'$ that sum up to one, we can evaluate the stratum borders $b_0, \ldots, b_I$ for random variable X.

$$\begin{array}{lll} b_0 = \inf\{x : F(x) > 0\} & & \xi_1 = [b_0, b_1), \quad (b_1 = -\infty \Rightarrow \xi_1 = (b_0, b_1)) \\ b_i = F^{-1}\left(\sum_{l=1}^{i} p_l\right) & \Rightarrow & \xi_i = [b_{i-1}, b_i) \\ b_I = \sup\{x : F(x) < 1\} & & \xi_I = [b_{I-1}, b_I) \end{array}$$

In order to obtain a single draw from stratum ξ_i, we generate a uniform random variable Y in the interval $[F(b_{i-1}), F(b_i))$ with

$$Y = \left(\sum_{l=1}^{i-1} p_l\right) + p_i U.$$

and, then, evaluate and return $F^{-1}(Y)$.

The above procedure can be generalized for random input vectors $\boldsymbol{X}$ if the el-

ements of $\boldsymbol{X}$ are independent and the inverse CDF of their marginal distributions $F_1^{-1}, \ldots, F_D^{-1}$ are available. In that case, using a vector $\boldsymbol{U} = (U_1, \ldots, U_D)$ of iid uniform variables, the vector $\boldsymbol{X}$ can be generated by:

$$\left(F_1^{-1}(U_1), F_2^{-1}(U_2), \ldots, F_d^{-1}(U_D)\right)',$$

and it can be stratified by stratifying $\boldsymbol{U}$. The stratification of uniform vectors is explained in Section 3.2.

3.2. Stratifying the Multivariate Uniform Distribution

In this section, we assume a multivariate uniform input $\boldsymbol{U} = (U_1, \ldots, U_D)'$ where U_d, $d = 1, \ldots, D$ are iid standard uniform variables. Stratification of the multivariate uniform distribution is important as it enables the stratification of other type of random inputs as described in Section 3.1.

We consider the identity function $\boldsymbol{S}(\boldsymbol{U}) = \boldsymbol{U}$ as the stratification function. For the sake of simplicity, we assume that each dimension d of the unit hypercube $[0,1)^D$ is divided into I_d equiprobable strata. The strata can be denoted by:

$$\xi_{(i_1,\ldots,i_D)'} = \prod_{d=1}^{D} \left[\frac{i_d - 1}{I_d}, \frac{i_d}{I_d}\right), \quad \forall d : i_d = 1, \ldots, I_d$$

with identical probabilities $\prod_{d=1}^{D} I_d^{-1}$. We can generate a uniform vector $\boldsymbol{Y}$ in stratum $\xi_{(i_1,\ldots,i_D)'}$ as follows:

$$Y_d = \frac{i_d - 1 + Unif(0,1)}{I_d}, \quad d = 1, \ldots, D,$$

and by changing index vector $(i_1, \ldots, i_D)'$, we can sample from any stratum. Figure 3.1 gives the plot of a stratified uniform sample of size 50 in $[0,1)^2$ where both dimensions are stratified with five equiprobable strata and proportional allocation is used.

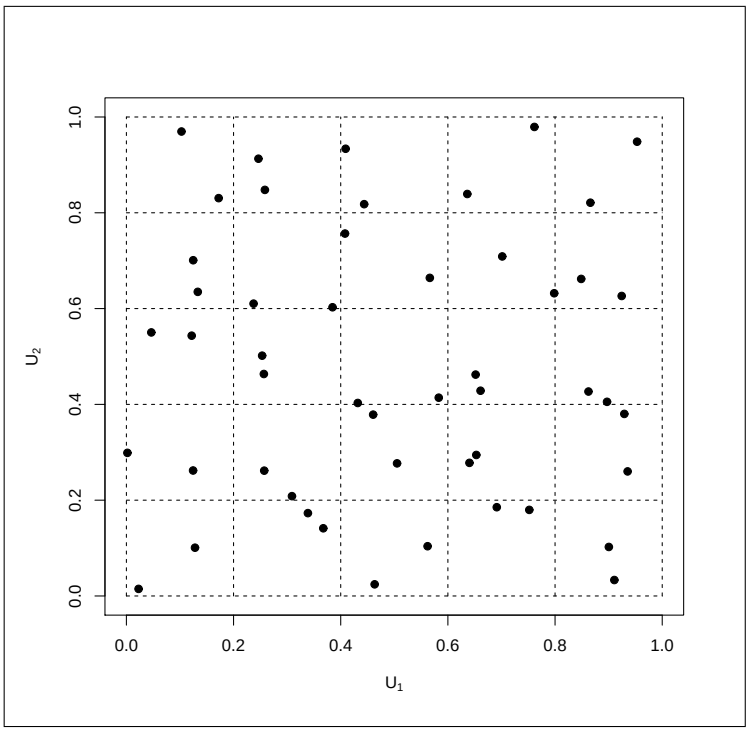

Figure 3.1. Stratification of multivariate uniforms in $[0,1)^2$ with five equiprobable strata over each dimension.

This type of stratification is easy to use, however it often needs too many strata to achieve reasonable variance reduction for practical problems with moderate or high dimension.

3.3. Stratifying Multinormal Distribution

In this section, we assume a multinormal input $\boldsymbol{Z} = (Z_1, \ldots, Z_D)'$ where Z_d, $d = 1, \ldots, D$, are iid standard normal variables. Multinormal vectors can be stratified with special strata structures due to special properties of the normal distribution.

Normal variables are typical random inputs in simulating stochastic models of financial variables, thus have highest practical importance. For other input variables with similar distributions (e.g., generalized hyperbolic or t-distribution), the random input can be stratified through standard normal variables using the CDF Φ of standard normal distribution and the inverse CDF of the specified distribution.

The following subsections consider two stratification functions for the multinormal input.

3.3.1. Directional Stratification

For a fixed direction $\boldsymbol{v} \in \mathbb{R}^D$ with $\|\boldsymbol{v}\| = 1$, we consider the linear projection $S(\boldsymbol{Z}) = \boldsymbol{v}'\boldsymbol{Z}$ of $\boldsymbol{Z}$ as stratification function, which also follows the standard normal distribution.

Given that $\boldsymbol{v}'\boldsymbol{Z} = z$, the conditional distribution of $\boldsymbol{Z}$ is also multinormal with mean vector $\boldsymbol{v}z$ and a variance covariance matrix $\boldsymbol{\Lambda} = \mathbf{I}_D - \boldsymbol{v}\boldsymbol{v}'$ where $\mathbf{I}_D$ denotes the D dimensional identity matrix. Since $\boldsymbol{v}\boldsymbol{v}'$ is orthogonal, we have $\boldsymbol{\Lambda}\boldsymbol{\Lambda}' = \boldsymbol{\Lambda}$. Thus, we do not have to compute a square-root of this matrix to sample $\boldsymbol{Z}$ from the conditional distribution given that $\boldsymbol{v}'\boldsymbol{Z} = z$.

Suppose the unit interval is partitioned into I disjoint intervals with borders $b_0 = 0, b_1, \ldots, b_{I-1}, b_I = 1$. We can generate $\boldsymbol{Z}$ conditional on $\Phi(\boldsymbol{v}'\boldsymbol{Z}) \in [b_{i-1}, b_i)$ by the steps shown in Figure 3.2.

Require: A stratum in the unit interval $[b_{i-1}, b_i)$;
Ensure: A random drawing of $\boldsymbol{Z}$ conditional on $\Phi(\boldsymbol{v}'\boldsymbol{Z}) \in [b_{i-1}, b_i)$
1: Generate $Y \sim Unif(b_{i-1}, b_i)$ using Equation 3.2
2: Compute $Z = \Phi^{-1}(Y)$
3: Generate standard multinormal vector $\boldsymbol{Z} \sim N(0, \mathbf{I}_D)$ independent of Y
4: Compute and return $\tilde{\boldsymbol{Z}} = \boldsymbol{v}Z + (\mathbf{I}_D - \boldsymbol{v}\boldsymbol{v}')\boldsymbol{Z}$

Figure 3.2. Generator for multinormal random vector with a stratified linear projection over direction $\boldsymbol{v} \in \mathbb{R}^D$.

For the last step in Figure 3.2, Jourdain *et al.* (2011) suggest to compute $\boldsymbol{v}Z + (\mathbf{I}_D - \boldsymbol{v}\boldsymbol{v}')\boldsymbol{Z}$ as $\boldsymbol{v}Z + \boldsymbol{Z} - \boldsymbol{v}(\boldsymbol{v}'\boldsymbol{Z})$ which requires $\mathrm{O}(D)$ operations rather than $\mathrm{O}(D^2)$. This algorithm requires the generation of one additional uniform number.

Figure 3.3 gives the plot of a stratified multinormal sample of size 100 in $\mathbb{R}^2$ where the sampling domain is stratified with 20 equiprobable strata over the direction $\boldsymbol{v} = \left(\sqrt{2}/2, \sqrt{2}/2\right)'$ and the sample is allocated proportionally.

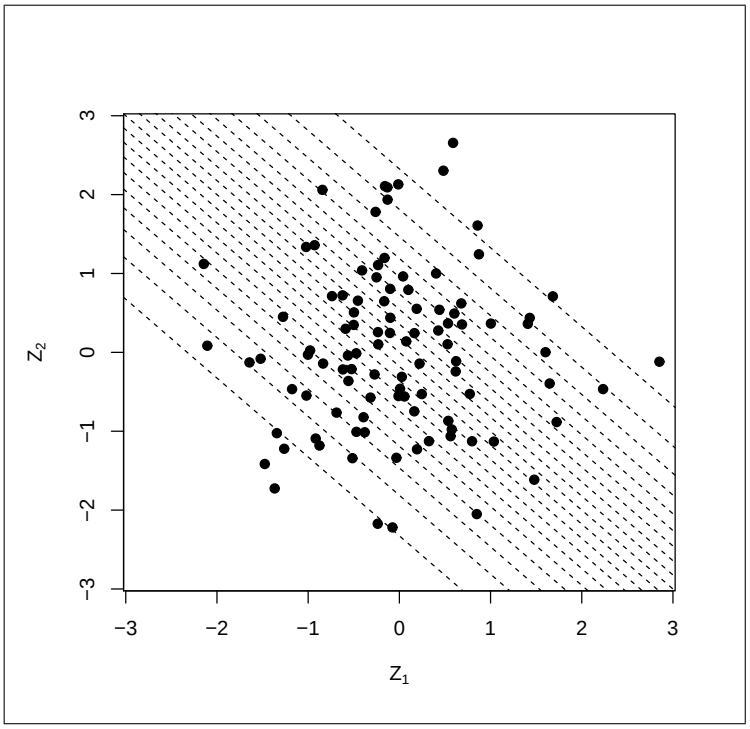

Figure 3.3. Stratification of multinormals in $\mathbb{R}^2$ with 20 equiprobable strata over direction $\boldsymbol{v} = \left(\sqrt{2}/2, \sqrt{2}/2\right)'$.

With the algorithm given in Figure 3.2, the sampling domain $\mathbb{R}^D$ is stratified by parallel hyperplanes orthogonal to direction $\boldsymbol{v}$. Now, suppose we are given a matrix $\bar{\boldsymbol{V}} \in \mathbb{R}^{D\times\Delta}$ of Δ orthogonal stratification directions, i.e., $\bar{\boldsymbol{V}} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_\Delta)$ and $\bar{\boldsymbol{V}}\bar{\boldsymbol{V}}' = \mathbf{I}_D$. Using the same approach, we can stratify $\mathbb{R}^D$ with parallel hyperplanes orthogonal to each direction. Thus, we can stratify the linear projection of the random input $\boldsymbol{Z}$ over the Δ dimensional subspace that is spanned by the columns of $\bar{\boldsymbol{V}}$.

The linear projection $\boldsymbol{S}(\boldsymbol{Z}) = \bar{\boldsymbol{V}}'\boldsymbol{Z} \in \mathbb{R}^\Delta$ follows the standard multinormal distribution and, given that $\bar{\boldsymbol{V}}'\boldsymbol{Z} = \boldsymbol{z}$, the conditional distribution of $\boldsymbol{Z}$ is also multinormal with mean vector $\bar{\boldsymbol{V}}\boldsymbol{z}$ and a variance covariance matrix $\boldsymbol{\Sigma} = \mathbf{I}_D - \bar{\boldsymbol{V}}\bar{\boldsymbol{V}}'$.

Suppose now the unit hypercube $[0,1)^\Delta$ is partitioned into strata $\xi_1, \ldots, \xi_I$ as

described in Section 3.2. We can generate $\boldsymbol{Z}$ conditional on $\left(\Phi\left(\boldsymbol{v}_1'\boldsymbol{Z}\right),\ldots,\Phi\left(\boldsymbol{v}_\Delta'\boldsymbol{Z}\right)\right)' \in \xi_i$ by the steps shown in Figure 3.4.

Require: A stratum in the unit hypercube ξ_i;

Ensure: A random drawing of $\boldsymbol{Z}$ conditional on $\left(\Phi\left(\boldsymbol{v}_1'\boldsymbol{Z}\right),\ldots,\Phi\left(\boldsymbol{v}_\Delta'\boldsymbol{Z}\right)\right)' \in \xi_i$

1: Generate multivariate uniform $\boldsymbol{Y} \in \xi_i \subseteq [0,1)^\Delta$ as described in Section 3.2.

2: Compute $\bar{\boldsymbol{Z}} = \left(\Phi^{-1}\left(Y_1\right),\ldots,\Phi^{-1}\left(Y_\Delta\right)\right)'$

3: Generate standard multinormal vector $\boldsymbol{Z} \sim N\left(0,\mathbf{I}_D\right)$ independent of $\boldsymbol{Y}$

4: Compute and return $\tilde{\boldsymbol{Z}} = \bar{\boldsymbol{V}}\bar{\boldsymbol{Z}} + \left(\mathbf{I}_D - \bar{\boldsymbol{V}}\bar{\boldsymbol{V}}'\right)\boldsymbol{Z}$

Figure 3.4. Generator for multinormal random vector with a stratified linear projection over the subspace spanned by the columns of $\boldsymbol{V} \in \mathbb{R}^{D\times\Delta}$.

Again, for the last step in Figure 3.2, replacing $\bar{\boldsymbol{V}}\bar{\boldsymbol{Z}} + \left(\mathbf{I}_D - \bar{\boldsymbol{V}}\bar{\boldsymbol{V}}'\right)\boldsymbol{Z}$ by $\bar{\boldsymbol{V}}\bar{\boldsymbol{Z}} + \boldsymbol{Z} - \bar{\boldsymbol{V}}\left(\bar{\boldsymbol{V}}'\boldsymbol{Z}\right)$ reduces the complexity of the algorithm from $\mathrm{O}\left(D^2\right)$ to $\mathrm{O}\left(D\right)$. This algorithm requires the generation of Δ additional uniform numbers.

In this thesis, we only use stratification over orthogonal directions. Jourdain *et al.* (2011) also investigate the possibility of stratification over different directions which are not necessarily orthogonal.

3.3.1.1. Linear Transformation. The complexity of the generators given in Figures 3.2 and 3.4 can be reduced to $\mathrm{O}\left(D\right)$ operations. However, for practically relevant simulation examples, the simulation function q is already of $\mathrm{O}\left(D^2\right)$ or a higher order. Linear transformation is another method that stratifies the multinormal input which is of $\mathrm{O}\left(D^2\right)$ but simpler, and does not require the generation of additional uniform variables. This method changes the basis of the generated multinormal input as the multinormal distribution is symmetric around the origin.

Suppose $\boldsymbol{V} \in \mathbb{R}^{D\times D}$ is an orthogonal matrix, i.e., $\boldsymbol{V}\boldsymbol{V}' = \mathrm{I}_D$. In the simulation algorithm, we can replace $q\left(\boldsymbol{Z}\right)$ by $q\left(\boldsymbol{V}\boldsymbol{Z}\right)$, since $\boldsymbol{V}\boldsymbol{Z}$ also follows the multivariate standard normal distribution and $E\left[q\left(\boldsymbol{Z}\right)\right] = E\left[q\left(\boldsymbol{V}\boldsymbol{Z}\right)\right]$. Here, $\boldsymbol{V}$ is called the linear transformation matrix.

If we are given Δ stratification directions $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_\Delta$, we can form a linear transformation matrix $\boldsymbol{V} = (\boldsymbol{v}_1, \ldots, \boldsymbol{v}_D)$ by generating remaining $D - \Delta$ columns arbitrarily without breaking the orthogonality rule. Once the linear transformation is applied, first Δ elements of $\boldsymbol{Z}$ will correspond to respective stratification directions. Thus, stratifying $Z_1, \ldots, Z_\Delta$ through uniform random variables will stratify the random input $\boldsymbol{V}\boldsymbol{Z}$ over directions $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_\Delta$. The pseudo code for the construction of the linear transformation matrix $\boldsymbol{V}$ is given in Figure 3.5.

Require: Stratification directions $\bar{\boldsymbol{V}} \in \mathbb{R}^{D \times \Delta}$

Ensure: Linear transformation matrix $\boldsymbol{V} \in \mathbb{R}^{D \times D}$

1: Define matrix $\boldsymbol{V} \in \mathbb{R}^{D \times D}$ with all entries equal to zero
2: Set the first Δ columns of $\boldsymbol{V}$ as $\bar{\boldsymbol{V}}$
3: **for** index $d = \Delta, \ldots, D-1$ **do**
4: Set $v_{D,d+1} = 1$
5: Define $\boldsymbol{B} \in \mathbb{R}^{d \times d}$ as the sub-matrix of $\boldsymbol{V}$ between the elements $v_{D-d,1}$ and $v_{D-1,d}$
6: Define $\boldsymbol{b} \in \mathbb{R}^d$ as the sub-vector of $\boldsymbol{V}$ between the elements $v_{D,1}$ and $v_{D,d}$
7: Find the unique solution of $\boldsymbol{B}'\bar{\boldsymbol{v}} + \boldsymbol{b} = 0$
8: Set the elements between $v_{D-d,d+1}$ and $v_{D-1,d+1}$ as $\bar{\boldsymbol{v}} \in \mathbb{R}^d$
9: Scale the column $d+1$ of $\boldsymbol{V}$ to the unit length.
10: **end for**
11: return $\boldsymbol{V}$

Figure 3.5. Generator for multinormal random vector with a stratified linear projection over the subspace spanned by the columns of $\boldsymbol{V} \in \mathbb{R}^{D \times \Delta}$.

The complexity of linear transformation is $\mathrm{O}(D^2)$, however it does not require the generation of any additional standard normal variables.

3.3.1.2. Optimal Stratification Directions. There are three different approaches to determine efficient stratification directions, i.e., the columns of the linear transformation matrix.

The first one is a standard approach based on the principle component construction of the variance covariance matrix. Suppose, $\boldsymbol{Z}_{\boldsymbol{\Lambda}}$ follows a multinormal distribution with zero mean vector and variance covariance matrix $\boldsymbol{\Lambda}$. The covariance of the random input may also be introduced as a part of the simulation function q by multiplying the standard multinormal vector with the Cholesky factor of $\boldsymbol{\Lambda}$. Instead, principle component construction uses eigenvalue factorization of the variance covariance matrix $\boldsymbol{\Lambda}$ to obtain a linear transformation matrix.

For a positive definite matrix $\boldsymbol{\Lambda} \in \mathbb{R}^{D\times D}$, there exist non-negative real eigenvalues $\lambda_1, ..., \lambda_D$ and orthogonal set of eigenvectors $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_D$ which satisfy $\boldsymbol{v}_d'\boldsymbol{v}_d = 1$ and $\boldsymbol{\Lambda}\boldsymbol{v}_d = \lambda_d \boldsymbol{v}_d$ for $d = 1, \ldots, D$. Then, we can generate $\boldsymbol{Z}_{\boldsymbol{\Lambda}}$ by:

$$\boldsymbol{Z}_{\boldsymbol{\Lambda}} = \sum_{d=1}^{D} \boldsymbol{v}_d \sqrt{\lambda_d} Z_d.$$

Here, the random variables $\lambda_d Z_d$ are called the principal components of the random vector $\boldsymbol{Z}_{\boldsymbol{\Lambda}}$ and the eigenvectors $\boldsymbol{v}_d$ are referred as the direction of the d-th principal component (see e.g., Kreinin *et al.*, 1998; Glasserman, 2004). In other words, $\sqrt{\lambda_d}$ is a scaling factor of a standard normal variable through the direction $\boldsymbol{v}_d$ to get $\boldsymbol{Z}_{\boldsymbol{\Lambda}}$. Thus, the share of the variance contribution of Z_d in $\boldsymbol{Z}_{\boldsymbol{\Lambda}}$ becomes proportional to λ_d. Suppose the eigenvalues and the respective eigenvectors of $\boldsymbol{\Lambda}$ are reordered so that $\lambda_1 \geq \ldots \geq \lambda_D$. Then, instead of stratifying each element of $\boldsymbol{Z}$, we can stratify just the first Δ elements of $\boldsymbol{Z}$. The main drawback of this approach is the fact that the principle component construction does not consider the variance of the simulation function q but only the variance coming from the variance covariance matrix of the input variables when finding effective directions.

The second approach determines efficient stratification directions with respect to the simulation function q. We again consider the conditional variance formula in Equation 3.1. We need to choose the directions such that $V\left[E\left[q\left(\boldsymbol{Z}\right)|\bar{\boldsymbol{V}}'\boldsymbol{Z}\right]\right]$ is maximized, which is equivalent to the minimization of $E\left[V\left[q\left(\boldsymbol{Z}\right)|\bar{\boldsymbol{V}}'\boldsymbol{Z}\right]\right]$, since the total variance $V\left[q\left(\boldsymbol{Z}\right)\right]$ is fixed.

In case of a single stratification direction, the above statement is in parallel with Jourdain *et al.* (2011): A good candidate for a stratification direction is the one that maximizes the "explained" component of the variance, or minimizes the "unexplained" part. Such a direction is the solution of the following problem.

$$\boldsymbol{v}^* = \underset{\boldsymbol{v}\in\mathbb{R}^D, \|\boldsymbol{v}\|=1}{\arg\min} \int_{-\infty}^{\infty} V\left[q\left(\boldsymbol{Z}\right)|\boldsymbol{v}'\boldsymbol{Z}=z\right]\phi\left(z\right)dz \tag{3.3}$$

Suppose that q is differentiable. Jourdain *et al.* (2011) suggest approximating the optimization problem in Equation 3.3:

$$\boldsymbol{v}^* = \underset{\boldsymbol{v}\in\mathbb{R}^D, \|\boldsymbol{v}\|=1}{\arg\min} \int_{-\infty}^{\infty} \nabla q\left(0\right) V\left[\boldsymbol{Z}|\boldsymbol{v}'\boldsymbol{Z}=z\right]\nabla' q\left(0\right)\phi\left(z\right)dz$$

by replacing $q\left(\boldsymbol{Z}\right)$ with its linear approximation $q\left(\boldsymbol{Z}\right) \approx q\left(0\right) + \nabla q\left(0\right)\boldsymbol{Z}$ around 0. The variance $V\left[\boldsymbol{Z}|\boldsymbol{v}'\boldsymbol{Z}=z\right] = \mathbf{I}_D - \boldsymbol{v}\boldsymbol{v}'$ is independent of z, thus the optimization problem reduces to:

$$\boldsymbol{v}^* = \underset{\boldsymbol{v}\in\mathbb{R}^D, \|\boldsymbol{v}\|=1}{\arg\min} \nabla q\left(0\right)\left(\mathbf{I}_D - \boldsymbol{v}\boldsymbol{v}'\right)\nabla' q\left(0\right) = \underset{\boldsymbol{v}\in\mathbb{R}^D, \|\boldsymbol{v}\|=1}{\arg\max} \left(\nabla q\left(0\right)\boldsymbol{v}\right)^2 \tag{3.4}$$

The optimal direction $\boldsymbol{v}^*$ of the optimization problem in Equation 3.4 is:

$$\boldsymbol{v}^* = \pm\frac{\nabla' q\left(0\right)}{\|\nabla q\left(0\right)\|} \tag{3.5}$$

for which a detailed proof is given in Jourdain *et al.* (2011).

Jourdain *et al.* (2011) suggest calculating the gradient at different points in order to produce multiple directions, however, the proposed approach provides non-orthogonal directions.

The third method produces Δ orthogonal stratification directions using the iterative algorithm of Imai and Tan (2006). They use their algorithm to optimally determine

the columns of the linear transformation matrix $\boldsymbol{V}$ which minimizes the effective directions of the multinormal input in underlying simulation function. The produced linear transformation matrix is then used to enhance the quasi-Monte Carlo method. We can use the columns of this linear transformation matrix as stratification directions.

For our use, the algorithm of Imai and Tan (2006) terminates in Δ steps. In the first iteration, we find the main stratification direction $\boldsymbol{v}_1^*$ by maximizing its variance contribution and, in the remaining steps $d = 2, \ldots, \Delta$, we solve the following optimization problem:

$$\begin{array}{ll} \max & \text{variance contribution of the } d-\text{th dimension} \\ \text{s.t.} & \boldsymbol{v}_d \in \mathbb{R}^D, \quad \|\boldsymbol{v}_d\| = 1, \quad \langle \boldsymbol{v}_l^*, \boldsymbol{v}_d \rangle = 0, \quad l = 1, \ldots, d-1, \end{array} \tag{3.6}$$

so that the orthogonality of $\boldsymbol{V}$ is preserved.

However, the variance contribution of the d-th dimension may not be expressed analytically for general simulation functions. Imai and Tan (2006) suggest approximating the objective function by linearizing the simulation function $q_{\boldsymbol{V}}(\boldsymbol{Z}) = q(\boldsymbol{V}\boldsymbol{Z})$ and, then, maximizing the variance contribution accordingly. They use the first-order Taylor series expansion around an arbitrary point $\boldsymbol{z}$:

$$q_{\boldsymbol{V}}(\boldsymbol{Z}) \approx q_{\boldsymbol{V}}(\boldsymbol{z}) + \sum\nolimits_{d=1}^{D} (Z_d - z_d) \left. \frac{\partial q_{\boldsymbol{V}}(\boldsymbol{Z})}{\partial Z_d} \right|_{\boldsymbol{Z}=\boldsymbol{z}},$$

which is linear in normal variables $Z_d - z_d$, $d = 1, \ldots, D$. Thus, the variance contributed by the d-th dimension is denoted by:

$$\left(\left. \frac{\partial q_V(\boldsymbol{Z})}{\partial Z_d} \right|_{\boldsymbol{Z}=\boldsymbol{z}} \right)^2,$$

and the optimization problem in Equation 3.6 can be reformulated as follows:

$$\begin{array}{ll} \max & \left(\left. \frac{\partial q_{\boldsymbol{V}}(\boldsymbol{Z})}{\partial Z_d} \right|_{\boldsymbol{Z}=\boldsymbol{z}} \right)^2 \\ \text{s.t.} & \boldsymbol{v}_d \in \mathbb{R}^D, \quad \|\boldsymbol{v}_d\| = 1, \quad \langle \boldsymbol{v}_l^*, \boldsymbol{v}_d \rangle = 0, \quad l = 1, \ldots, d-1 \end{array} \tag{3.7}$$

The main direction $\boldsymbol{v}_1$ is optimized by assuming a linearization at $\boldsymbol{z} = 0$. Imai and Tan (2006) consider higher order Taylor series expansions when finding a subsequent optimal direction. Then, in optimizing $\boldsymbol{v}_d$, it requires a d-th order expansion. As this is a very complex approach, they again suggest a first order approximation but at a different point. Thus, given an optimum $\boldsymbol{v}_l^*$, $l = 1, \ldots, d-1$, for optimizing the column $\boldsymbol{v}_d$, they propose to set $\boldsymbol{z} = (1, \ldots, 1, 0, \ldots, 0)'$ with $d-1$ leading ones.

After the linearization procedure, the maximization problem simplifies. In Imai and Tan (2006, 2009), the iterative maximization procedure is explained in detail.

Lemma 3.1. *If we linearize $q_{\boldsymbol{V}}(\boldsymbol{Z})$ around $\boldsymbol{z} = 0$, the optimal linear transformation matrix $\boldsymbol{V}^*$ of the maximization problem in Equation 3.7 has its first column equal to the optimal stratification direction in Equation 3.5 proposed by Jourdain* et al. *(2011).*

Proof. For the first column of the linear transformation matrix $\boldsymbol{V}$, the maximization problem in Equation 3.7 simplifies to:

$$\begin{aligned} \max \quad & \left(\frac{\partial q_{\boldsymbol{V}}(\boldsymbol{Z})}{\partial Z_1}\bigg|_{\boldsymbol{Z}=0} \right)^2 \\ \text{s.t.} \quad & \boldsymbol{v}_1 \in \mathbb{R}^D, \quad \|\boldsymbol{v}_1\| = 1 \end{aligned}$$

We can rewrite the objective function:

$$\left(\frac{\partial q_{\boldsymbol{V}}(\boldsymbol{Z})}{\partial Z_1}\bigg|_{\boldsymbol{Z}=0} \right)^2 = \left(\frac{\partial q(\boldsymbol{V}\boldsymbol{Z})}{\partial Z_1}\bigg|_{\boldsymbol{Z}=0} \right)^2 = \left(\frac{\partial q(\boldsymbol{V}\boldsymbol{Z})}{\partial \boldsymbol{V}\boldsymbol{Z}}\bigg|_{\boldsymbol{Z}=0} \frac{\partial \boldsymbol{V}\boldsymbol{Z}}{\partial Z_1}\bigg|_{\boldsymbol{Z}=0} \right)^2 = (\nabla q(0)\, \boldsymbol{v}_1)^2$$

and reformulate the optimization problem as in Equation 3.4. □

If $q(\boldsymbol{Z})$ is not everywhere differentiable, the procedures of this section can be implemented to the differentiable components of q (see e.g, Jourdain *et al.*, 2011). One can also implement Conditional Monte Carlo (Trotter and Tukey, 1956) priorly to smooth out q and obtain a differentiable simulation function.

3.3.2. Radial Stratification

The fact that the multinormal distribution is symmetric around the origin makes it possible to draw samples from this distribution with a stratified Euclidean norm. For a standard multinormal vector $\boldsymbol{Z} \in \mathbb{R}^D$, the Euclidean norm $\|\boldsymbol{Z}\|$ is the radius of the sphere on which $\boldsymbol{Z}$ falls and the squared norm of the vector $\|\boldsymbol{Z}\|^2$ is a chi-square distributed random variable with D degrees of freedom. The vector $\boldsymbol{Z}/\|\boldsymbol{Z}\|$ is uniformly distributed over the unit hypersphere and the vector $R\boldsymbol{Z}/\|\boldsymbol{Z}\|$ is uniformly distributed over the sphere which has radius R.

If we stratify the random variable χ^2_D through uniform random variables, the random vector $\tilde{\boldsymbol{Z}} = \sqrt{\chi^2_D}\boldsymbol{Z}/\|\boldsymbol{Z}\|$ will be stratified radially.

We consider the stratification function $\boldsymbol{S}(\boldsymbol{Z}) = F_{\chi^2_D}\left(\|\boldsymbol{Z}\|^2\right)$ where $F_{\chi^2_D}$ is the CDF of chi-squared distribution with D degrees of freedom. Suppose the unit interval is partitioned into I disjoint intervals with borders $b_0 = 0, b_1, \ldots, b_{I-1}, b_I = 1$. We can generate $\boldsymbol{Z}$ conditional on $\boldsymbol{S}(\boldsymbol{Z}) \in F_{\chi^2_D}\left(\|\boldsymbol{Z}\|^2\right)$ by following steps in Figure 3.6.

Require: A stratum in the unit interval $[b_{i-1}, b_i)$;
Ensure: A random drawing of $\boldsymbol{Z}$ conditional on $F_{\chi^2_D}\left(\|\boldsymbol{Z}\|^2\right) \in [b_{i-1}, b_i)$
1: Generate $Y \sim Unif\,(b_{i-1}, b_i)$ using Equation 3.2
2: Compute $R = \sqrt{F^{-1}_{\chi^2_D}(Y)}$
3: Generate standard multinormal vector $\boldsymbol{Z} \sim N(0, \mathbf{I}_D)$ independent of Y
4: Compute and return $\tilde{\boldsymbol{Z}} = R\boldsymbol{Z}/\|\boldsymbol{Z}\|$

Figure 3.6. Generator for a radially stratified multinormal random vector.

Figure 3.7 gives the plot of a radially stratified multinormal sample of size 50 in $\mathbb{R}^2$ where the sampling domain is stratified with 10 equiprobable strata and the sample is allocated proportionally.

Radial stratification can also be combined with directional stratification. Suppose the first Δ elements of the random vector $\boldsymbol{Z}$ is subject to directional stratification. The

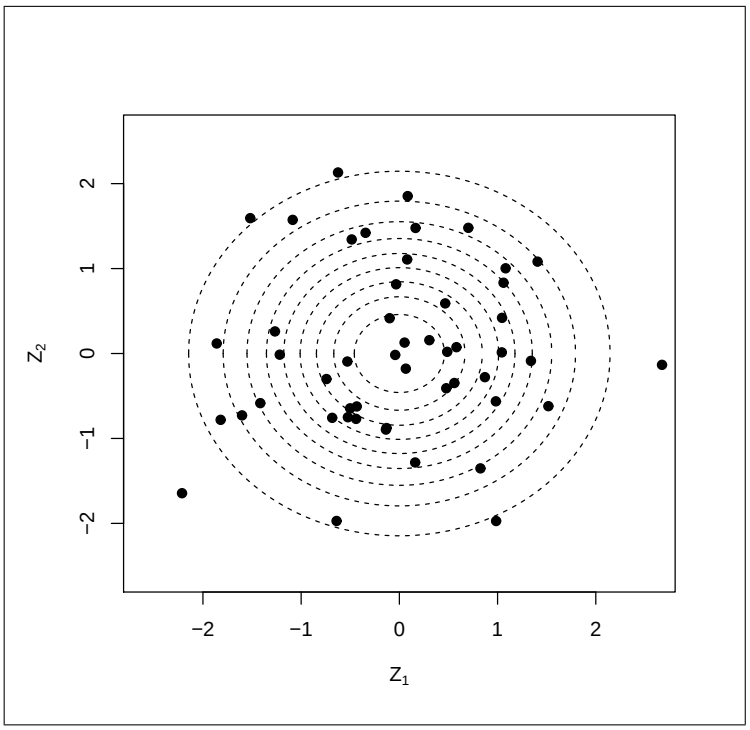

Figure 3.7. Radial stratification of multinormals in $\mathbb{R}^2$ with 10 equiprobable strata.

remaining $D - \Delta$ elements still can be radially stratified since the elements of $\boldsymbol{Z}$ are independent. Figure 3.8 is an illustration of the described stratification for $D = 3$ and $\Delta = 1$. The space is stratified through the direction that corresponds to the first element Z_1 of the input vector. Then, the remaining directions are radially stratified with circles, or namely hyperspheres on $D - \Delta = 2$ dimensional space.

A detailed explanation about radial stratification can be found in Glasserman (2004), page 227.

3.4. Stratifying Multivariate t-Distribution

Let $\boldsymbol{T} = (T_1, \ldots, T_D)'$ be a random vector, the elements of which are iid t-distributed random variables with ν degrees of freedom. The standard approach to generate multivariate t-distributed vector requires a standard multinormal random vector $\boldsymbol{Z} = (Z_1, \ldots, Z_D)'$ and an independent chi-squared random variable χ^2_ν with ν

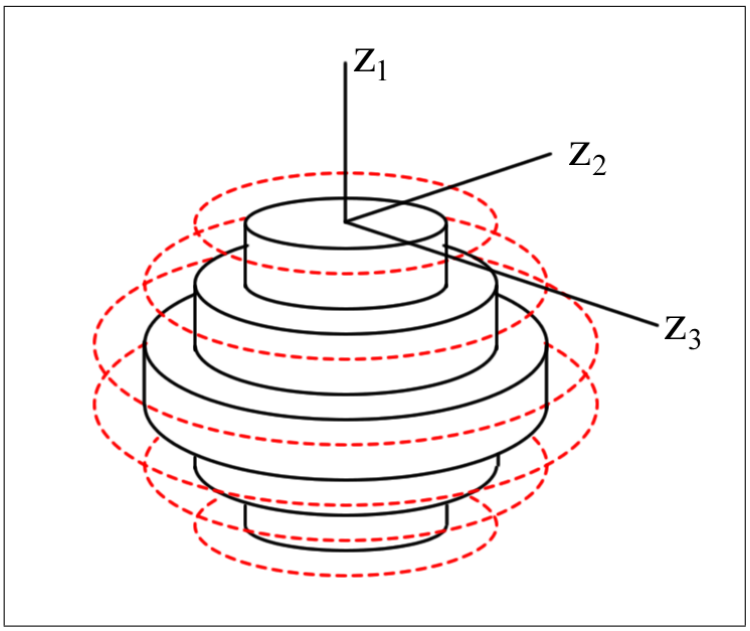

Figure 3.8. An illustration of $\mathbb{R}^3$ that is stratified over a single direction and the remaining directions are stratified radially.

degrees of freedom, such that:

$$\boldsymbol{T} = \boldsymbol{Z}\left(\chi^2_\nu/\nu\right)^{-1/2}.$$

Then, we can apply directional stratification to the multinormal input $\boldsymbol{Z}$ following the procedures described in Section 3.3.1. Simultaneously, chi-squared random variable χ^2_ν can be stratified through uniform random variables.

An alternative way to generate multivariate t-distributed vectors based on the polar method and numerical inversion is proposed by Hörmann and Sak (2010). That method requires only a standard multinormal random vector. The generated $\boldsymbol{T}$ vectors are highly correlated with the multinormal input. In this case, one can use any stratification technique for the multinormal distribution described in Section 3.3.

4. MULTIRESPONSE STRATIFIED SAMPLING

4.1. Introduction to MRS

We define the response function $r(\boldsymbol{\theta})$ as the expectation of a random valued function under parameter $\boldsymbol{\theta}$ where Θ is the set of possible parameter values. We assume that $r(\boldsymbol{\theta})$ can only be estimated via simulation for each $\boldsymbol{\theta} \in \Theta$. Moreover, we assume that the dimension of the random input is not changed by the parameters. Then, it is possible to simulate r for several $\boldsymbol{\theta}$ values in a single simulation. Such an objective can easily be realized using common random numbers (Law, 2014). However, under this simple approach, all $r(\boldsymbol{\theta})$ values are estimated with comparatively large variances. To obtain more accurate estimates for several $r(\boldsymbol{\theta})$ values, one can apply variance reduction methods such as antithetic variates (Myers and Montgomery, 2002), control variates (see e.g., Rubinstein and Marcus, 1985), and importance sampling (see e.g., Glasserman and Li, 2005; Sak and Hörmann, 2012).

A variance reduction method that can be very useful for the above problem is stratified sampling. This seems to be a fact overlooked in the literature. In the progress of this thesis, we have published two papers which are closely related to the use of stratified sampling in multiresponse simulation. The idea is first used in Başoğlu *et al.* (2013) to minimize the maximum relative error of tail loss probabilities in a linear asset portfolio. Also, Başoğlu and Hörmann (2014) show how stratification can be used to minimize the overall error of multiple estimates in general Monte Carlo simulation problems. The latter work introduces objective functions that measure the overall error and develops simple methods to minimize (approximately) these objective functions. For the optimization problem, the allocation fractions are used as decision variables. The optimal allocation fractions are then used in the sampling phase. The resulting method is called "multiresponse stratified sampling (MRS)" and it is applicable to simulation problems where the size of the random input is independent of the parameter space, and it exhibits a good performance for simulation problems for which

stratification reaches a good variance reduction in the single response case.

This chapter closely follows Başoğlu *et al.* (2013) and Başoğlu and Hörmann (2014). It explains how stratification and allocation fractions can be used to minimize the overall error of multiple estimates.

4.2. Minimizing the Overall Error

We define the response function $r(\boldsymbol{\theta}) = E\left[q\left(\boldsymbol{X}, \boldsymbol{\theta}\right)\right]$, where $r : \Theta \to \mathbb{R}$ and $\boldsymbol{X} \in \mathbb{R}^D$ follows a common distribution that is independent of $\boldsymbol{\theta}$. We assume that for each $\boldsymbol{\theta} \in \Theta$, $r(\boldsymbol{\theta})$ can only be estimated via simulation. If we can find an effective stratification function $\boldsymbol{S}\left(\boldsymbol{X}\right)$ that has a large contribution to $V\left[q\left(\boldsymbol{X}, \boldsymbol{\theta}\right)\right]$ and is computationally tractable (i.e., we can generate $\boldsymbol{X}$ conditional on strata defined for $\boldsymbol{S}\left(\boldsymbol{X}\right)$), then that stratification will effectively reduce the variance of the estimates.

Suppose we are given J points, $\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_J$, in Θ. Our goal is to estimate $y_j = r(\boldsymbol{\theta}_j)$ for $j = 1, \ldots, J$ in a single simulation using stratified sampling. Let $\hat{y}_{ij}$ be the mean conditional on stratum i estimated under parameter $\boldsymbol{\theta}_j$. Then, the stratified estimator of y_j is calculated as $\hat{y}_j = \sum_{i=1}^{I} p_i \hat{y}_{ij}$. Each of these estimates is unbiased and asymptotically normal Étoré and Jourdain (2010). Let $\hat{s}_{ij}^2$ be the variance of the sample in stratum i drawn under parameter $\boldsymbol{\theta}_j$. We also define $\hat{s}_{ijk}$ as the covariance of the samples in stratum i drawn under parameters $\boldsymbol{\theta}_j$ and $\boldsymbol{\theta}_k$. Given the total sample size, N, and the vector of allocation fractions $\boldsymbol{\pi} = (\pi_1, \ldots, \pi_I)'$, the elements of the sample variance covariance matrix $\boldsymbol{\Sigma}$ of $\hat{\boldsymbol{y}} = (\hat{y}_1, \ldots, \hat{y}_J)'$ can be calculated as a function of the allocation fractions:

$$\Sigma_{jj}\left(\boldsymbol{\pi}\right) = N^{-1} \sum_{i=1}^{I} \pi_i^{-1} p_i^2 \hat{s}_{ij}^2, \quad j = 1, \ldots, J, \tag{4.1}$$

$$\Sigma_{jk}\left(\boldsymbol{\pi}\right) = N^{-1} \sum_{i=1}^{I} \pi_i^{-1} p_i^2 \hat{s}_{ijk}, \quad j = 1, \ldots, J, \quad k = 1, \ldots, J, \quad j \neq k. \tag{4.2}$$

We define the objective function $\omega(\boldsymbol{\pi}) = g(\boldsymbol{\Sigma}(\boldsymbol{\pi}))$, where $g : \mathbb{R}^{J\times J} \to \mathbb{R}^{+} \cup \{0\}$ is a continuous function of the variance and covariance values. This objective function is used for representing the overall error of all J estimates, for example:

- $\omega_{MSE}(\boldsymbol{\pi}) = \sum_{j=1}^{J} \Sigma_{jj}(\boldsymbol{\pi})$, the mean squared error of all estimates,
- $\omega_{MSR}(\boldsymbol{\pi}) = \sum_{j=1}^{J} \hat{y}_j^{-2}\Sigma_{jj}\,(\boldsymbol{\pi})$, the mean squared relative error of all estimates,
- $\omega_{SUM}(\boldsymbol{\pi}) = \sum_{j=1}^{J}\sum_{k=1}^{J} \Sigma_{jk}\,(\boldsymbol{\pi})$, the sum of all elements in the variance covariance matrix,
- $\omega_{MAXE}(\boldsymbol{\pi}) = \max\{j : \Sigma_{jj}\,(\boldsymbol{\pi})\}$, the maximum of the absolute errors of all estimates,
- $\omega_{MAXR}(\boldsymbol{\pi}) = \max\{j : \hat{y}_j^{-2}\Sigma_{jj}\,(\boldsymbol{\pi})\}$, the maximum of the absolute relative errors of all estimates,

all of which are convex in $\boldsymbol{\pi}$. Therefore, we assume ω to be a convex function of $\boldsymbol{\pi}$ for practically relevant examples. Our objective is to solve the following optimization model:

$$\begin{aligned} \min \quad & \omega\,(\boldsymbol{\pi}) \\ \text{s.t.} \quad & \textstyle\sum_{i=1}^{I} \pi_i = 1 \quad \text{and} \quad \pi_i \geq 0, \quad i = 1, \ldots, I. \end{aligned} \tag{4.3}$$

The constraints of the model in Equation 4.3 form a convex and bounded feasible region. Thus, the mathematical model in Equation 4.3 becomes a convex programming problem and a local optimum found in the feasible solution set will also be a global optimum. If a closed-form optimal solution is unavailable, the optimal solution of the problem in Equation 4.3 can be found by using an interior-type method with trust regions (Byrd *et al.*, 2006). However, since we use the estimates of conditional variance and covariance values, $\hat{s}_{ij}^2$ and $\hat{s}_{ijk}$, the optimal solution of an instance will only be an estimate for the real optimal allocation fractions. Thus, a sub-optimal solution for the model in Equation 4.3 is enough in practice.

The above examples for the objective function $\omega(\boldsymbol{\pi})$ can be categorized in two

general classes. The first three examples are all linear functions of the elements of the variance-covariance matrix $\mathbf{\Sigma}$. It is possible to find closed-form solutions under this class of objective functions. The second class consists of the last two examples where we consider the maximum of the variances, Σ_{jj}, which are weighted with non-negative coefficients. We provide a heuristic method for the second class.

4.2.1. Minimizing a Linear Combination of the Elements of the Variance Covariance Matrix

As a first example, suppose, we want to minimize the mean squared relative error of all estimates. We consider the objective function

$$\omega_{MSR}(\boldsymbol{\pi}) = \sum_{j=1}^{J} \hat{y}_j^{-2} \Sigma_{jj}\,(\boldsymbol{\pi}) = N^{-1} \sum_{i=1}^{I} \pi_i^{-1} p_i^2 \sum_{j=1}^{J} \hat{y}_j^{-2} \hat{s}_{ij}^2,$$

which has a form that is similar to the variance of the stratified estimator given in Equation 2.9, only the expression of the conditional variances, s_i^2, is now replaced by $\sum_{j=1}^{J} \hat{y}_j^{-2} \hat{s}_{ij}^2$. The lower bound for $\omega_{MSR}(\boldsymbol{\pi})$ is:

$$\omega_{MSR}^*(\boldsymbol{\pi}) = N^{-1} \left(\sum_{i=1}^{I} p_i \left(\sum_{j=1}^{J} \hat{y}_j^{-2} \hat{s}_{ij}^2 \right)^{1/2} \right)^2$$

and, according to Equation 2.10, it can be attained if we choose the allocation fractions:

$$\pi_i^* = p_i \left(\sum_{j=1}^{J} \hat{y}_j^{-2} \hat{s}_{ij}^2 \right)^{1/2} \Bigg/ \sum_{l=1}^{I} p_l \left(\sum_{j=1}^{J} \hat{y}_j^{-2} \hat{s}_{lj}^2 \right)^{1/2}, \quad i = 1, \ldots, I. \tag{4.4}$$

Using the pilot sample, we estimate the response values, $\hat{y}_j$, $j = 1, \ldots, J$, and conditional variances, $\hat{s}_{ij}^2$, $i = 1, \ldots, I$, $j = 1, \ldots, J$. Then, we can use Equation 4.4 to determine the optimal allocation of the sample in the main simulation, so that the mean squared relative error of all estimates is minimized.

Clearly, the arguments of above remain valid if we replace s_i^2 of Equation 2.9 by arbitrary non-negative constants. Thus, we can generalize the previous example by assuming g to be a linear function of the elements of $\boldsymbol{\Sigma}(\boldsymbol{\pi})$.

Theorem 4.1. *Assume the objective function* $\omega(\boldsymbol{\pi}) = \sum_{j=1}^{J} \sum_{k=j}^{J} c_{jk}\Sigma_{jk}(\boldsymbol{\pi})$ *with real coefficients. We plug in the variance and the covariance values given in Equation 4.1 and 4.2 into this objective function and obtain:*

$$\omega(\boldsymbol{\pi}) = N^{-1} \sum_{i=1}^{I} \pi_i^{-1} p_i^2 \left(\sum_{j=1}^{J} c_{jj} \hat{s}_{ij}^2 + \sum_{j<k} c_{jk} \hat{s}_{ijk} \right).$$

For $\omega(\boldsymbol{\pi})$ *to be convex in the feasible region of Equation 4.3, a simple necessary condition is the non-negativity of* $\sum_{j=1}^{J} c_{jj}\hat{s}_{ij}^2 + \sum_{j<k} c_{jk}\hat{s}_{ijk}$ *for* $i = 1, \ldots, I$. *Under this assumption, the optimal solution that minimizes* $\omega(\boldsymbol{\pi})$ *is:*

$$\pi_i^* = \frac{p_i \left(\sum\limits_{j=1}^{J} c_{jj} \hat{s}_{ij}^2 + \sum\limits_{j<k} c_{jk} \hat{s}_{ijk} \right)^{1/2}}{\sum\limits_{l=1}^{I} p_l \left(\sum\limits_{j=1}^{J} c_{jj} \hat{s}_{lj}^2 + \sum\limits_{j<k} c_{jk} \hat{s}_{ljk} \right)^{1/2}}, \quad i = 1, \ldots, I. \tag{4.5}$$

As a first application of Theorem 4.1, we have already considered $\omega_{MSR}(\boldsymbol{\pi})$. As a second example, we consider minimizing the variance of a convex combination of all estimates, $\sum_{j=1}^{J} \lambda_j \hat{y}_j$ where $\lambda_j \geq 0$, $j = 1, \ldots, J$, are fixed and $\sum_{j=1}^{J} \lambda_j = 1$. This is of practical relevance since, in response surface applications, we require linear approximations of the intermediate values using the estimates of $r(\boldsymbol{\theta}_j)$, $j = 1, \ldots, J$ (Box and Wilson, 1951). Our objective function is, then, $\omega(\boldsymbol{\pi}) = \sum_{j=1}^{J} \sum_{k=1}^{J} \lambda_j \lambda_k \Sigma_{jk}(\boldsymbol{\pi})$. If we aim to minimize the variance of the average of all estimates, the coefficients λ_j become all equal and the objective function simplifies to the sum of all elements in the variance covariance matrix, $\omega_{SUM}(\boldsymbol{\pi}) = \sum_{j=1}^{J} \sum_{k=1}^{J} \Sigma_{jk}(\boldsymbol{\pi})$. For any case, the optimal solution can be found using Equation 4.5 and the estimates of the conditional variance and covariances.

A third example in this class of objective functions is related to the estimation of the ratio $r(\boldsymbol{\theta}_1)/r(\boldsymbol{\theta}_2)$. We obtain the stratified estimates, $\hat{y}_1$ and $\hat{y}_2$, using a single simulation. Then, we can estimate the ratio with $\hat{y}_1/\hat{y}_2$ which has bias (see e.g., Fishman, 1996, page 109):

$$Bias\left[\hat{y}_1/\hat{y}_2\right] = \hat{y}_1\hat{y}_2^{-3}\Sigma_{22}(\boldsymbol{\pi}) - \hat{y}_2^{-2}\Sigma_{12}(\boldsymbol{\pi}) + O\left(N^{-2}\right).$$

Here, the leading term is of order $O(N^{-1})$. It is possible to reduce the bias by subtracting the estimate of the leading term from the ratio estimate. However, the squared bias is of order $O(N^{-2})$, which is small compared to the variance. Thus, we rather consider reducing the variance that is approximated by:

$$V\left[\hat{y}_1/\hat{y}_2\right] \approx \hat{y}_1^2\hat{y}_2^{-4}\Sigma_{22}(\boldsymbol{\pi}) - 2\hat{y}_1\hat{y}_2^{-3}\Sigma_{12}(\boldsymbol{\pi}) + \hat{y}_2^{-2}\Sigma_{11}(\boldsymbol{\pi}). \tag{4.6}$$

The approximate variance in Equation 4.6 is of order $O(N^{-1})$ and found by using the multivariate Taylor series expansion of the variance of the ratio (see e.g., Glasserman, 2004). Equation 4.6 defines an objective function that is a linear function of the variance-covariance matrix. It is easy to show that it satisfies the sufficient condition for convexity stated in Theorem 4.1. Therefore, we can find optimal allocation fractions using Equation 4.5:

$$\pi_i^* = \frac{p_i\left(\hat{y}_1^2\hat{y}_2^{-4}\hat{s}_{i2}^2 - 2\hat{y}_1\hat{y}_2^{-3}\hat{s}_{i12} + \hat{y}_2^{-2}\hat{s}_{i1}^2\right)^{1/2}}{\sum_{l=1}^{I} p_l\left(\hat{y}_1^2\hat{y}_2^{-4}\hat{s}_{l2}^2 - 2\hat{y}_1\hat{y}_2^{-3}\hat{s}_{l12} + \hat{y}_2^{-2}\hat{s}_{l1}^2\right)^{1/2}}, \quad i = 1, \ldots, I. \tag{4.7}$$

where the conditional sample variances, $\hat{s}_{i1}^2$ and $\hat{s}_{i2}^2$, the conditional sample covariances, $\hat{s}_{i12}$, and the estimates, $\hat{y}_1$ and $\hat{y}_2$, are found using the pilot sample for $i = 1, \ldots, I$.

Note that the allocation fractions given in Equation 4.7 minimize the variance of a single ratio estimate. We can also use the MRS algorithm for minimizing the overall error of multiple ratio estimates.

4.2.2. Minimizing the Maximum of the Variances Weighted with Nonnegative Coefficients

We consider minimizing $\omega(\boldsymbol{\pi}) = \max\{j : c_j \Sigma_{jj}(\boldsymbol{\pi})\}$ for non-negative c_j values. We use the short notation $\omega_j(\boldsymbol{\pi})$ for functions $c_j \Sigma_{jj}(\boldsymbol{\pi})$, $j = 1, \ldots, J$. The objective function is non-differentiable at some points in the solution set; thus, the model in Equation 4.3 is replaced by:

$$\begin{aligned} \min \quad & \hat{\omega} \\ \text{s.t.} \quad & \hat{\omega} - \omega_j(\boldsymbol{\pi}) \geq 0, \quad j = 1, \ldots, J, \\ & \textstyle\sum_{i=1}^{I} \pi_i = 1, \quad \text{and} \quad \pi_i > 0, \quad i = 1, \ldots, I. \end{aligned} \tag{4.8}$$

Remark 4.1. *If the number of strata, I, is much larger than the number of estimates J, it is likely that the relative errors of all estimates are equal at the global optimum. Then, it might be better to transform the first set of inequalities to strict equations to enforce that all relative errors are equal. However, there might be none or uncountably many points satisfying this condition. Thus, we choose to proceed with the model in Equation 4.8.*

For the model in Equation 4.8, we develop a simple allocation heuristic which yields a satisfactory sub-optimal solution in a short period of time. The heuristic method calculates the respective optimum allocation fractions $\boldsymbol{\pi}^j \in \mathbb{R}^I$ which minimize the variances of the stratified estimates, $\boldsymbol{\Sigma}_{jj}(\boldsymbol{\pi})$, $j = 1, \ldots, J$,

$$\pi_i^j = p_i \hat{s}_{ij} \Big/ \sum\nolimits_{l=1}^{I} p_l \hat{s}_{lj}, \quad i = 1, \ldots, I,$$

and searches for the best solution in the convex hull of these points. In other words,

the allocation heuristic searches the optimal solution of the following model:

$$\begin{aligned} \min \quad & \hat{\omega} \\ \text{s.t.} \quad & \hat{\omega} - \omega_j(\boldsymbol{\pi}) \geq 0, \quad j = 1, \ldots, J, \quad \pi_i - \textstyle\sum_{j=1}^{J} \lambda_j \pi_i^j = 0, \quad i = 1, \ldots, I, \\ & \textstyle\sum_{j=1}^{J} \lambda_j = 1, \quad \text{and} \quad \lambda_j \geq 0, \quad j = 1, \ldots, J. \end{aligned} \tag{4.9}$$

Thus, the equality and the non-negativity constraints of the model in Equation 4.8 are automatically satisfied and the dimension of the problem is clearly reduced for practically relevant settings.

The main idea of the heuristic method is as follows: For every point in the convex hull, the objective value $\omega(\boldsymbol{\pi})$ is attained by one of the convex functions, say, $\omega_j(\boldsymbol{\pi})$. Then, we expect the objective value to decrease if we move towards the respective optimum solution $\boldsymbol{\pi}^j$. We stop moving towards $\boldsymbol{\pi}^j$ if we reach a point for which the objective value is attained by another function $\omega_k(\boldsymbol{\pi})$, $k \neq j$; then, we can move towards $\boldsymbol{\pi}^k$. In summary, for every point in the convex hull, the heuristic automatically determines a descent direction at the function evaluation. The size of the move is determined by the iteration number.

We continue with a more-detailed description of the heuristic algorithm. Assume a feasible solution $\boldsymbol{\pi}$ in the convex hull of $\boldsymbol{\pi}^j$, $j = 1, \ldots, J$. The objective value at $\boldsymbol{\pi}$ is

$$\omega(\boldsymbol{\pi}) = \max \left\{ j : \omega_j(\boldsymbol{\pi}) \right\}, \tag{4.10}$$

and the index of the function which attains this objective value is obtained by

$$j\left(\boldsymbol{\pi}\right) = \arg\max \left\{ j : \omega_j\left(\boldsymbol{\pi}\right) \right\}. \tag{4.11}$$

Assume further that ω is differentiable at $\boldsymbol{\pi}$. Then, the two functions, ω and $\omega_{j(\boldsymbol{\pi})}$ are equal in an open neighborhood of $\boldsymbol{\pi}$, and $\omega_{j(\boldsymbol{\pi})}$ decreases as we move towards $\boldsymbol{\pi}^j$. Hence, $\boldsymbol{\pi}^j - \boldsymbol{\pi}$ forms a simple descent direction for ω at $\boldsymbol{\pi}$.

The allocation heuristic algorithm starts with an initial feasible solution in the convex hull of $\boldsymbol{\pi}^j$, $j = 1, \ldots, J$. The average of vectors $\boldsymbol{\pi}^j$, $j = 1, \ldots, J$, is a simple candidate. The current and the best solution are denoted by $\boldsymbol{\pi}^c$ and $\boldsymbol{\pi}^h$, respectively. Therefore, we initialize $\boldsymbol{\pi}^h = \boldsymbol{\pi}^c = J^{-1} \sum_{i=1}^{J} \boldsymbol{\pi}^j$ and set the best and the current objective value $\omega^h = \omega^c = \omega(\boldsymbol{\pi}^c)$. We also determine the index of the function which attains the current objective value, $j^c = j(\boldsymbol{\pi}^c)$.

We expect an improvement in the objective value if we make a move towards $\boldsymbol{\pi}^{j^c}$. Let η denote the iteration count as the algorithm is run. When the algorithm is started, η is initialized to 1. We update the current solution according to the recursion $\boldsymbol{\pi}^c = \left(\eta \boldsymbol{\pi}^c + \boldsymbol{\pi}^{j^c}\right) / (\eta + 1)$. As η increases, the current solution $\boldsymbol{\pi}^c$ and, since ω is continuous, the current objective value ω^c converge.

After each move, we update the current objective value ω^c and set $j' = j(\boldsymbol{\pi}^c)$ as the index of the function which attains this value. At this point, we sequentially check the following two conditions:

- If $\omega^c \leq \omega^h$, we update the best solution $\boldsymbol{\pi}^h = \boldsymbol{\pi}^c$ and the best objective value $\omega^h = \omega^c$.
- If $j' \neq j^c$, then the current solution can be improved by moving in another direction. We set $j^c = j'$ and increase η by one.

Then, we return to the step where we update the current solution. The algorithm terminates when ω^c converges.

Remark 4.2. *In order to find the optimal solution in the convex hull, we have also implemented bisection method on the simple descent direction we have suggested. However, we have observed that the convergence becomes too slow around the non-differentiable points of* $\omega(\boldsymbol{\pi})$. *Comparatively, the heuristic provides a faster convergence but it is not necessarily monotone.*

Let us demonstrate how the heuristic works on a simple instance. Assume that

$I = J = 3$ and we have the following $\omega_j(\boldsymbol{\pi})$ for $j = 1, 2, 3$,

$$\omega_1(\boldsymbol{\pi}) = .01\pi_1^{-1} + .09\pi_2^{-1} + .36\pi_3^{-1},$$
$$\omega_2(\boldsymbol{\pi}) = .09\pi_1^{-1} + .25\pi_2^{-1} + .04\pi_3^{-1},$$
$$\omega_3(\boldsymbol{\pi}) = .16\pi_1^{-1} + .04\pi_2^{-1} + .16\pi_3^{-1}.$$

Since $\pi_3 = 1 - \pi_1 - \pi_2$, we can search the heuristic solution $\boldsymbol{\pi}^h$ in the subspace spanned by π_1 and π_2 with the restriction that $\pi_1 + \pi_2 < 1$. Figure 4.1 shows the contour plot of the objective function $\omega(\boldsymbol{\pi}) = \max\{\omega_1(\boldsymbol{\pi}), \omega_2(\boldsymbol{\pi}), \omega_3(\boldsymbol{\pi})\}$ in the subspace spanned by π_1 and π_2. The heuristic searches the sub-optimal solution in the convex hull of the respective minima, $\boldsymbol{\pi}^j$, $j = 1, 2, 3$. The heuristic terminates after 12 moves. The moves which result in an improvement in the objective value are emphasized.

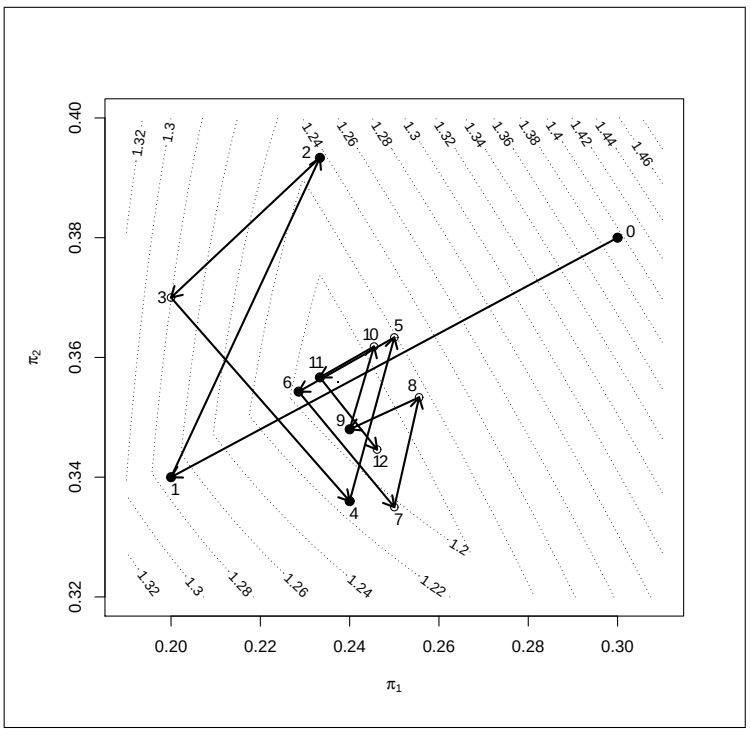

Figure 4.1. Contour plot of $\omega(\boldsymbol{\pi}) = \max\{j : \omega_j(\boldsymbol{\pi})\}$ for given instance.

The global minimum $\boldsymbol{\pi}^* = (.237, .356, .407)'$ is in the convex hull of the respective minima $\boldsymbol{\pi}^1 = (.1, .3, .6)'$, $\boldsymbol{\pi}^2 = (.3, .5, .2)'$ and $\boldsymbol{\pi}^3 = (.4, .2, .4)'$ with an objective value $\omega^* = \omega(\boldsymbol{\pi}^*) = 1.18$. The heuristic algorithm starts the search at $\boldsymbol{\pi}^c = (.3, .38, .32)$ and

terminates after 12 moves. The objective value improves at the 1st, 2nd, 4th, 6th, 9th and 11th moves. The heuristic solution found in the 11th move is $\boldsymbol{\pi}^h = (.233, .357, .41)'$ with an objective value $\omega^h = \omega(\boldsymbol{\pi}^h) = 1.188$, which is very close to the optimal objective value ω^*.

We give the pseudo code of the heuristic method in Figure 4.2.

Require: The functions ω_j of the model in Equation 4.8, $j = 1, \ldots, J$; optimal solutions $\boldsymbol{\pi}^j$ of each respective problem, $j = 1, \ldots, J$; an initial feasible solution $\boldsymbol{\pi}^c$ in the convex hull of $\boldsymbol{\pi}^j$

Ensure: A sub-optimal solution $\boldsymbol{\pi}^h$ for the model in Equation 4.8.

set $\varepsilon = \infty$, $\eta = 1$ and $\boldsymbol{\pi}^h = \boldsymbol{\pi}^c$
set $\omega^h = \omega^c = \omega(\boldsymbol{\pi}^c)$ and $j^c = j(\boldsymbol{\pi}^c)$ using respectively Equations 4.10 and 4.11
while $|\varepsilon|/\omega^h > \epsilon$ **do**
    update the current solution $\boldsymbol{\pi}^c = \left(\eta\boldsymbol{\pi}^c + \boldsymbol{\pi}^{j^c}\right)/(\eta + 1)$
    set $\omega' = \omega(\boldsymbol{\pi}^c)$ and $j' = j(\boldsymbol{\pi}^c)$ using respectively Equations 4.10 and 4.11
    if $\omega' \leq \omega^h$ **then**
        set $\varepsilon = \omega^h - \omega'$
        update the best solution $\boldsymbol{\pi}^h = \boldsymbol{\pi}^c$ and the best objective value $\omega^h = \omega'$
    end if
    if $j' \neq j^c$ **then**
        set $\eta = \eta + 1$ and $\varepsilon = \omega^c - \omega'$
        set $\omega^c = \omega'$ and $j^c = j'$
    end if
end while
return $\boldsymbol{\pi}^h$

Figure 4.2. A heuristic method for determining optimal allocation fractions for minimizing the maximum of $\omega_j(\boldsymbol{\pi})$ of J different estimates.

4.3. A Numerical Example

We now present an example where we demonstrate the practical aspects of the optimization models given in this section. MRS yields comparatively good results if stratification is also successful in the estimation of a single value, for example, the simulation function is of rare-event type or the random input has a small number of stratification variables $\boldsymbol{S}(\boldsymbol{X})$ that have a large contribution to the output variance. For this reason, we choose such a rare-event problem where the size of the random input is small.

We consider a simulation function where we can achieve successful variance reduction by applying stratification over a single direction.

$$q(\boldsymbol{Z}, \boldsymbol{\theta}) = \min\left\{\max\left\{(Z_1 + Z_2)^2 + \theta_1 Z_1, \theta_2\right\}, \theta_2 + \theta_3\right\}. \tag{4.12}$$

Here, Z_1 and Z_2 are independent standard normal variables. Our aim is to estimate $x_j = r(\boldsymbol{\theta}_j)$, $j = 1, \ldots, 6$, in a single simulation for $\boldsymbol{\theta}_j = (\theta_{j1}, \theta_{j2}, \theta_{j3})'$ vectors:

$$\begin{aligned} &\boldsymbol{\theta}_1 = (.1, 1.1, .722)', \quad \boldsymbol{\theta}_2 = (.1, 1.2, .688)', \quad \boldsymbol{\theta}_3 = (.2, 1.1, .291)', \\ &\boldsymbol{\theta}_4 = (.2, 1.2, .342)', \quad \boldsymbol{\theta}_5 = (.3, 1.1, .148)', \quad \boldsymbol{\theta}_6 = (.1, 1.2, .192)'. \end{aligned} \tag{4.13}$$

We divide the sampling domain into 100 equiprobable subsets with planes that are orthogonal to the direction $(\sqrt{2}/2, \sqrt{2}/2)'$. In other words, we stratify the random variable $Z_1 + Z_2$ with $I = 100$ equiprobable strata (see Figure 3.3 for a demonstration with 20 equiprobable strata).

We run MRS under the objective functions listed in the beginning of this section and also, for minimizing the variance of each single estimate, with the objective functions $\Sigma_{jj}(\boldsymbol{\pi})$, $j = 1, \ldots, 6$. This allows us to observe the minimal error reachable for each $\boldsymbol{\theta}_j$.

We remind that in all MRS runs, the pilot samples use the same random sequence. For each example, we also run naive Monte Carlo simulation using common random numbers and find the variance of multiple estimates in a single simulation. The efficiency results are presented as variance reduction factors, $VR[\hat{y}] = V\left[\hat{y}^{NV}\right]/V[\hat{y}]$ where $\hat{y}^{NV}$ and $\hat{y}$ denote the naive Monte Carlo and stratified estimators, respectively. The execution times of the methods are also reported.

The variance and the percentage relative error of all estimates obtained under each of these objective functions are listed in Table 4.1. Variance reduction factors are also given for performance comparison. The total sample size used in all simulations is $N = 10^6$ and $N_p = 10^5$ of the drawings are used in a pilot study to determine the optimal allocation fractions via Equation 4.5 or the allocation heuristic described in Section 4.2.2.

If we minimize the variance of a single estimate rather than focusing on the overall error, we observe in Table 4.1 that the variance for that estimate is reduced, however, we also see severe losses in the variance reduction factors for most of the other estimates. The last five objective functions aim to minimize the overall error of the simulation and, in those cases, we observe reasonably good variance reduction with only moderate losses compared to the maximum variance reduction factors. For the last two objective functions, we utilize the allocation heuristic to determine sub-optimal allocation fractions. The objective function ω_{MAXE} leads to very close variances. A similar result is also valid for relative error values under the objective function ω_{MAXR}. Thus, for this specific problem, we conclude that the allocation heuristic successfully determines the allocation fractions which minimize the maximum variance and the relative error, respectively.

Table 4.1. The variance and the percentage relative error of all estimates obtained under different objective functions.

Method	$\omega(\boldsymbol{\pi})$	$\boldsymbol{\theta}_j$ $\sim y_j$	$\boldsymbol{\theta}_1$ 1.385	$\boldsymbol{\theta}_2$ 1.462	$\boldsymbol{\theta}_3$ 1.226	$\boldsymbol{\theta}_4$ 1.340	$\boldsymbol{\theta}_5$ 1.166	$\boldsymbol{\theta}_6$ 1.281
Naive	-	$V[\hat{y}_j]$	1.14E-07	1.03E-07	2.00E-08	2.71E-08	5.30E-09	8.78E-09
1.80 sec.	-	$\%RE[\hat{y}_j]$	±0.04785	±0.04307	±0.02262	±0.02410	±0.01224	±0.01434
Multiresponse	$\min \Sigma_{11}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.08E-10	1.04E-10	2.26E-10	1.46E-10	5.90E-10	3.52E-10
Stratified	2.52 sec.	$\%RE[\hat{y}_j]$	±0.00147	±0.00137	±0.00240	±0.00177	±0.00409	±0.00287
Sampling		$VR[\hat{y}_j]$	1059	989	89	186	9	25
	$\min \Sigma_{22}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.35E-10	9.09E-11	5.44E-10	2.36E-10	1.01E-09	6.64E-10
	2.54 sec.	$\%RE[\hat{y}_j]$	±0.00164	±0.00128	±0.00373	±0.00225	±0.00534	±0.00394
		$VR[\hat{y}_j]$	849	1136	37	115	5	13
	$\min \Sigma_{33}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	7.39E-10	1.14E-09	1.07E-10	1.99E-10	1.50E-10	2.01E-10
	2.46 sec.	$\%RE[\hat{y}_j]$	±0.00385	±0.00453	±0.00166	±0.00207	±0.00206	±0.00217
		$VR[\hat{y}_j]$	155	90	187	136	35	44
	$\min \Sigma_{44}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.82E-10	2.47E-10	1.41E-10	1.20E-10	3.33E-10	2.11E-10
	2.52 sec.	$\%RE[\hat{y}_j]$	±0.00191	±0.00211	±0.00190	±0.00160	±0.00307	±0.00222
		$VR[\hat{y}_j]$	629	418	142	226	16	42
	$\min \Sigma_{55}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	8.10E-10	1.12E-09	1.31E-10	2.29E-10	8.78E-11	1.50E-10
	2.52 sec.	$\%RE[\hat{y}_j]$	±0.00403	±0.00449	±0.00183	±0.00222	±0.00158	±0.00187
		$VR[\hat{y}_j]$	141	92	153	119	60	59
	$\min \Sigma_{66}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	2.62E-10	3.50E-10	1.23E-10	1.49E-10	1.03E-10	1.20E-10
	2.51 sec.	$\%RE[\hat{y}_j]$	±0.00229	±0.00251	±0.00178	±0.00178	±0.00171	±0.00168
		$VR[\hat{y}_j]$	437	295	162	182	52	73
Multiresponse	$\min \omega_{SUM}(\boldsymbol{\pi}) = \sum_{j=1}^{J} \sum_{k=1}^{J} \Sigma_{jk}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.37E-10	1.49E-10	1.28E-10	1.34E-10	1.34E-10	1.33E-10
Stratified	2.50 sec.	$\%RE[\hat{y}_j]$	±0.00166	±0.00164	±0.00181	±0.00169	±0.00194	±0.00176
Sampling		$VR[\hat{y}_j]$	835	692	156	203	40	66
	$\min \omega_{MSE}(\boldsymbol{\pi}) = \sum_{j=1}^{J} \Sigma_{jj}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.32E-10	1.35E-10	1.35E-10	1.43E-10	1.22E-10	1.32E-10
	2.47 sec.	$\%RE[\hat{y}_j]$	±0.00163	±0.00156	±0.00186	±0.00175	±0.00186	±0.00176
		$VR[\hat{y}_j]$	864	763	148	190	43	67
	$\min \omega_{MSR}(\boldsymbol{\pi}) = \sum_{j=1}^{J} \hat{y}_j^{-2} \Sigma_{jj}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.37E-10	1.42E-10	1.32E-10	1.44E-10	1.18E-10	1.30E-10
	2.53 sec.	$\%RE[\hat{y}_j]$	±0.00165	±0.00160	±0.00184	±0.00176	±0.00182	±0.00174
		$VR[\hat{y}_j]$	837	726	151	188	45	68
	$\min \omega_{MAXE}(\boldsymbol{\pi}) = \max\{j : \Sigma_{jj}(\boldsymbol{\pi})\}$	$V[\hat{y}_j]$	1.31E-10	1.33E-10	1.36E-10	1.38E-10	1.31E-10	1.35E-10
	2.12 sec.	$\%RE[\hat{y}_j]$	±0.00162	±0.00155	±0.00187	±0.00172	±0.00192	±0.00178
		$VR[\hat{y}_j]$	874	776	147	197	41	65
	$\min \omega_{MAXR}(\boldsymbol{\pi}) = \max\{j : \hat{y}_j^{-2} \Sigma_{jj}(\boldsymbol{\pi})\}$	$V[\hat{y}_j]$	1.54E-10	1.69E-10	1.23E-10	1.46E-10	1.11E-10	1.26E-10
	2.14 sec.	$\%RE[\hat{y}_j]$	±0.00176	±0.00174	±0.00177	±0.00177	±0.00177	±0.00172
		$VR[\hat{y}_j]$	741	611	162	185	48	70

5. RELATIONS BETWEEN OAS AND IS

It is easier to use STRS than IS as determining an efficient IS density is a delicate operation. If it is not chosen carefully, IS can also increase the variance. STRS has its own complexity and restrictions depending on the type of the random input, the strata, and the way of generating conditional random vectors in each stratum. The strata construction (i.e., the stratification function $\boldsymbol{S}$, the number of strata, the location of strata boundaries) is a important operation which affects the estimator performance. On the other hand, for given strata, PAS never increases the output variance. Implementing stratified sampling with any other allocation (such as OAS) may be viewed as a form of importance sampling (Glasserman, 2004).

Jourdain (2009) classifies variance reduction techniques in two categories: those which guarantee a variance reduction however at a moderate level (e.g. antithetic variates, control variates, and conditional Monte Carlo) and those which may lead to a significant variance reduction but may also increase the variance in an improper implementation (e.g. importance sampling). Stratified sampling, as a variance reduction technique, is considered to be at the boundary of these two categories. This is due to two facts: A variance reduction is guaranteed under proportional allocation, yet other allocation rules may increase the variance as well as they are capable to decrease it.

The similarity of IS and OAS is that they both intend to make more sampling from the regions where $q(\boldsymbol{X})$ has a large variance. Thus, both methods are capable of reducing the variance in rare event simulations. Importance sampling intends to do this by a change of measure. On the other hand, STRS achieves an initial variance reduction by omitting the first component of the variance decomposition in (2.8) - or (3.1). Then, OAS applies a change of measure by using optimal allocation fractions. As a result, although they share a common objective, OAS has an exclusive variance component that is not being reduced by IS. However, practically, OAS needs a clever strata structure too many equiprobable strata to reach the efficiency of IS. Therefore,

compared to the lean implementations, a combination of these methods may yield further variance reduction. However, it is important to comprehend the reason why this combination works well and, in order to do that, we have to point out the weaknesses of both methods in practice.

As it was stated in Section 2.2.2, most IS densities are obtained by changing the parameter of the original sampling density f. However, even with the change of the parameter, the original densities are not generally capable of imitating closely the optimal IS density for practical simulation functions. Thus, only a limited variance reduction can be achieved. A solution to that is to choose a density from a larger family of distributions.

In fact, as Glasserman (2004) points out, OAS can be considered as a form of importance sampling which aims to find an optimal IS density from a large family of distributions. This family is generated by multiplying the original density f with a function c, that is constant in each stratum. In other words, for fixed strata ξ_i, $i = 1, \ldots, I$, the family of densities is:

$$C_f = \left\{ c\left(x\right) f\left(x\right) : c\left(x\right) = \sum_{i=1}^{I} \pi_i p_i^{-1} \mathbb{1}_{\{x \in \xi_i\}},\ \sum_{i=1}^{I} \pi_i = 1,\ \pi_i \geq 0,\ i = 1, \ldots, I \right\} \quad (5.1)$$

where the function $\mathbb{1}_{\{.\}}$ takes the value of one if the indicated event occurs, and zero otherwise. We define $\left\|\boldsymbol{p}\right\|_\infty = \max\left\{p_1, \ldots, p_I\right\}$ as the maximum of the stratum probabilities. As $\left\|\boldsymbol{p}\right\|_\infty$ converges to zero, the family C_f starts to contain good approximations of the optimal IS density f_{IS}^*. However, the number of strata required for a close approximation can be too large for practical situations where the difference between the original density f and the optimal IS density f_{IS}^* is excessive.

If we construct this family with a density, say a suboptimal IS density f_{IS}, which is similar to the optimal IS density, then good approximations of f_{IS}^* can be observed in $C_{f_{IS}}$ even with a small number of strata. We can demonstrate this with the following example.

Example: Suppose, we want to estimate $y = E_\phi [q(Z)]$ where $Z \sim N(0,1)$, ϕ is the probability density function (PDF) of standard normal distribution, and $q(x) = \{e^x - 3.6\}^+$. The true value of this expectation is:

$$E_\phi [q(Z)] = \int_{\ln 3.6}^{\infty} q(x)\,\phi(x)\,dx = -3.6\Phi(-\ln 3.6) + e^{0.5}\Phi(1 - \ln 3.6) \cong 0.2816$$

Then, the optimal IS density is:

$$f_{IS}^*(x) = \frac{q(x)\,\phi(x)}{\int_{-\infty}^{\infty} q(x)\,\phi(x)\,dx} \cong \begin{cases} (1.4168e^x - 5.1003)\,e^{-0.5x^2} & x > \ln 3.6 \\ 0 & x \leq \ln 3.6 \end{cases}$$

which attains its maximum at the point $x^* = 1.9828$. We consider a suboptimal IS density from the normal distribution family obtained by changing the location parameter. Thus, we choose the normal density with mean x^* and unit variance. Figure 5.1 shows the standard normal density $\phi(x)$, the shifted IS density $f_{IS}(x)$ and the optimal IS density $f_{IS}^*(x)$.

Now, let us assume the family of densities C_ϕ that is constructed from the original density ϕ with I strata. One density in this family is constructed with optimal allocation fractions π_i^* in (2.10). In other words, an optimally allocated stratified sample follows a density f_{OAS} in C_ϕ, which is constructed with π_i^* values.

$$f_{OAS}(x) = \phi(x) \sum_{i=1}^{I} \pi_i^* p_i^{-1} \mathbb{1}_{\{x \in \xi_i\}}$$

In fact, this density converges to the optimal IS density f_{IS}^* as $\|\boldsymbol{p}\|_\infty$ goes to zero. Figure 5.2 demonstrates this convergence for 100, 250, 500, and 1000 equiprobable strata. Notice that the rare event probability $\Pr\{q(Z) > 0\} = Pr\{Z > \ln 3.6\} \approx 0.1$. Thus, the rare event region is divided into 100 strata in the best case (Figure 5.2d).

In Figure 5.2, the density f_{OAS} of the optimally allocated stratified sample approaches to the optimal IS density. One needs to observe that the rare event occurs in

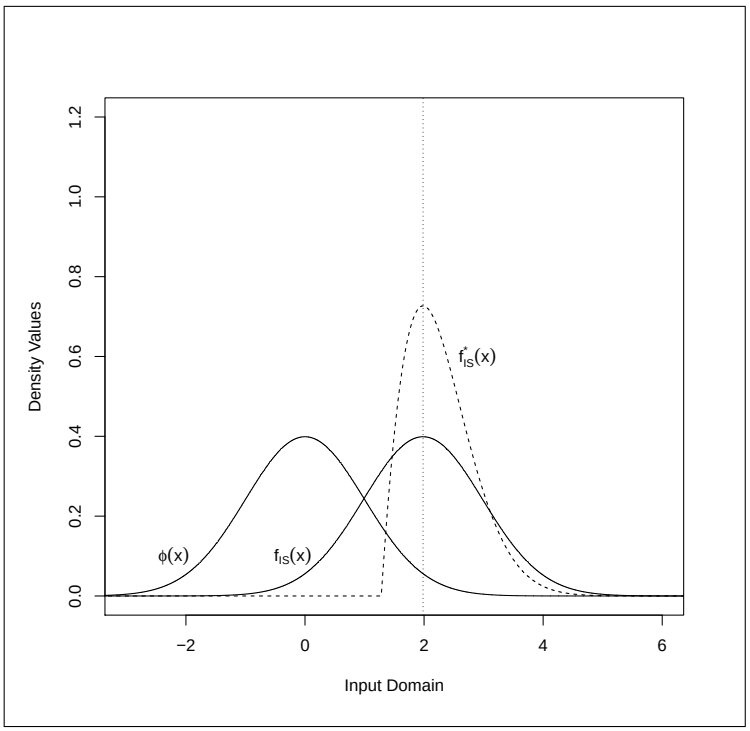

Figure 5.1. The original density $\phi(x)$, the shifted IS density $f_{IS}(x)$ and the optimal IS density $f^*_{IS}(x)$.

the right tail of the standard normal density and the constructed densities are obtained by multiplying the tail with piecewise $\pi^*_i p_i^{-1}$ constants. It is clearly seen in Figure 5.2a, concave and increasing parts of f^*_{IS} are approximated by piecewisely convex and decreasing functions. Moreover, the tails of ϕ shows a fast exponential decrease, and so does the right tail of constructed densities. Thus, especially far in the right tail, the convergence occurs very slowly and the constructed density remains below the optimal IS density f^*_{IS}.

Now, let us observe the same process if we construct the family given in Equation 5.1 from the density f_{IS}, the shifted normal density with mean $x^* = 1.9828$ and unit variance. Here, we calculate p_i values with respect to density f_{IS} and calculate the conditional standard deviations σ_i under the simulation function $q_{IS}(x) = q(x)\rho(x)$. In order to be fair, we have fixed the maximum number of strata to 100 when we construct the densities form density f^*_{IS}. In that case, 76 of the strata falls in the rare event region. Figure 5.3 shows the convergence of $f_{OAS}(x)$ under five, 10, 25, and 100

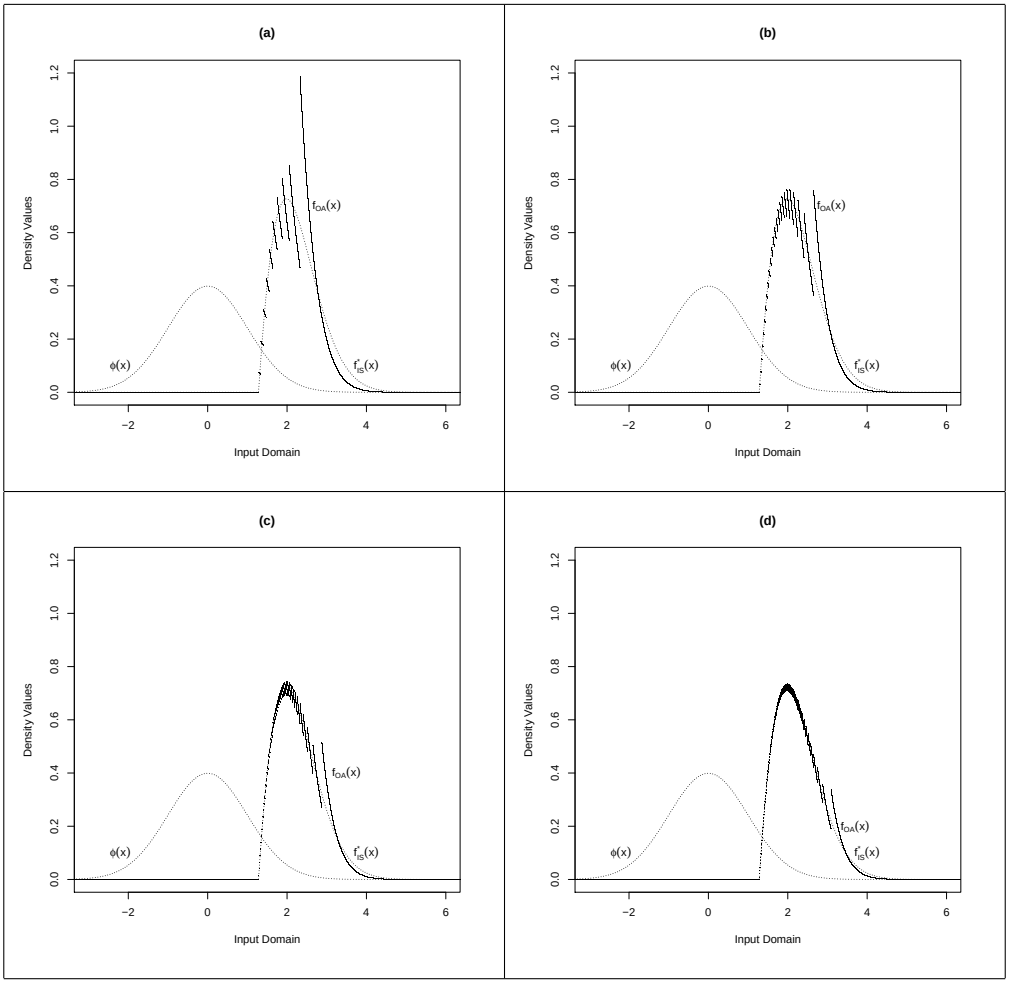

Figure 5.2. The densities $f_{OAS}(x)$ constructed as the product of the original density $\phi(x)$ and the piecewise constant density of π_i^* values under (a) 100, (b) 250, (c) 500, and (d) 1000 equiprobable strata respectively.

equiprobable strata.

From Figure 5.3, we again observe that the density of the optimally allocated stratified sample converges to the optimal IS density, yet the convergence occurs faster in $\|\boldsymbol{p}\|_\infty$. Since f_{IS} and the f_{IS}^* have a common mode and similar shape, the piecewise density constructed from f_{IS} approximates f_{IS}^* in the whole sampling domain including the far right tail. Thus, it can be expected that the variance of the stratified estimator obtained with this procedure is smaller.

For this toy example, we report also the variances for the naive Monte Carlo

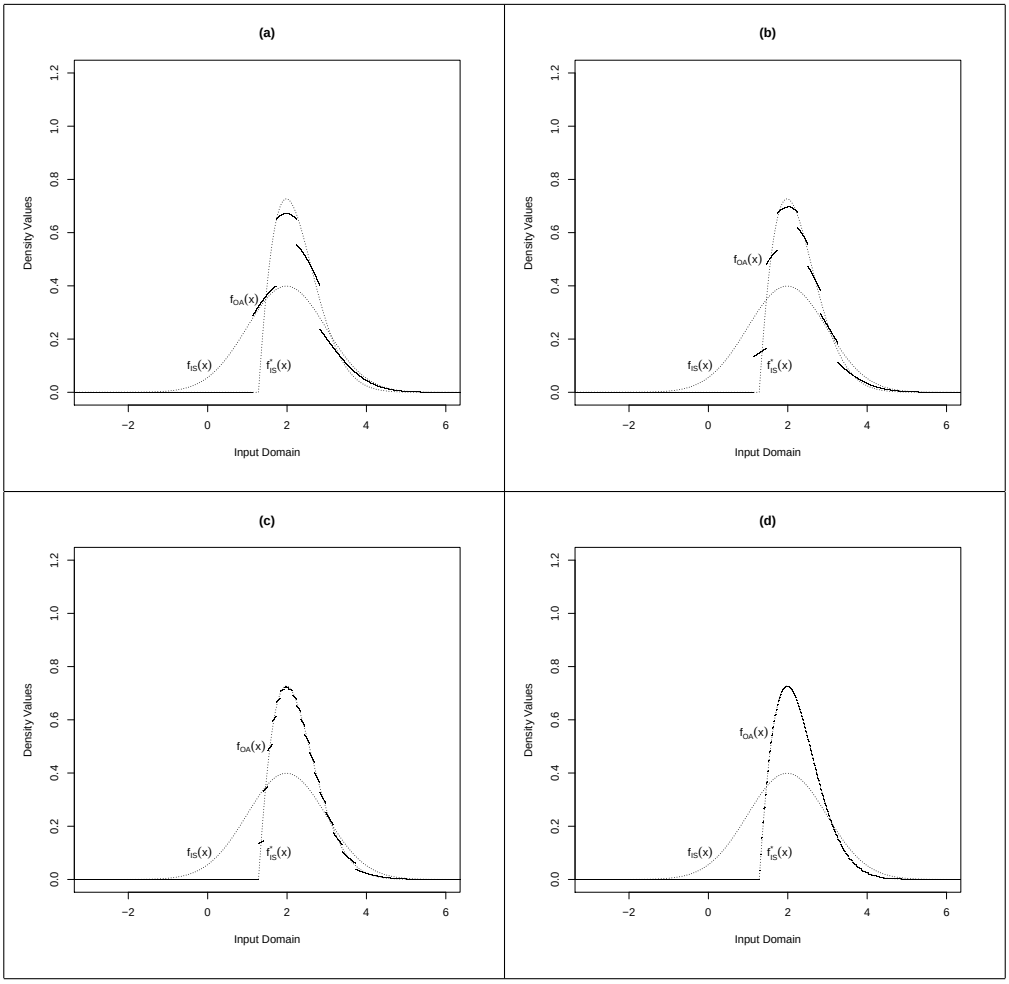

Figure 5.3. The densities $f_{OAS}(x)$ constructed as the product of the IS density $f_{IS}(x)$ and the piecewise constant density of π_i^* values under (a) five, (b) 10, (c) 25, and (d) 100 equiprobable strata respectively.

(NV), importance sampling with mean shift (IS), the modified AOA method with no IS and 1000 equiprobable strata (OAS) and the modified AOA method combined with IS and 100 equiprobable strata (OASIS). For the mean shift we use $x^* = 1.9828$. In each of these methods, the sample size is fixed to $N = 10^5$. Since the execution time of all algorithms are about the same, we will consider the variance of each estimator as an efficiency measure. Table 5.1 shows the estimates and their variances obtained with each method.

As expected, the IS with a mean shift yields a moderate variance reduction. Lean stratified sampling yields smaller variance, however due to the slow convergence

Table 5.1. Comparison of NV, IS, OAS, and OASIS algorithms.

	Estimate	Variance
NV	0.27970	2.17E-05
IS	0.28192	3.77E-07
OAS	0.28155	2.73E-09
OASIS	0.28159	8.53E-11

observed in the last stratum in Figure 5.2, a certain amount of variance still remains. About 97% of this variance is removed by combining IS and stratification, since f_{OAS} has a better convergence to f_{IS}^* when we generate the sample from a density that has similar patterns to f_{IS}^*.

For practically relevant examples where $D \geq 2$, it is not so simple to show that f_{OAS} approaches to f_{IS}^*. In fact, the density f_{OAS} of optimally allocated stratified sample does not always approach to f_{IS}. For the counter example, see Appendix A.

The combined method achieves better variance reduction compared to lean implementations of IS and STRS. In literature, both IS and STRS are considered as effective methods for rare event simulation. But, still they have deficiencies that can be removed by combining these two methods:

- IS remains weak since a practical choice of the IS density does not really succeed in imitating the optimal IS density. Yet it provides an initial variance reduction. The simultaneous stratification first reduces $V\left[E_{f_{IS}}\left[q_{IS}\left(\boldsymbol{X}\right)|S\left(\boldsymbol{X}\right)\right]\right]$ from $V\left[q_{IS}\left(\boldsymbol{X}\right)\right]$. Secondly, under the optimal allocation policy, the generated sample follows a distribution which imitates the optimal IS density better.
- Lean stratification requires a large number of complex strata to achieve successful variance reductions. This, sometimes, might lead to poor estimation of conditional standard deviations. A simultaneous IS helps to reduce the number of strata, thus, the complexity of stratification. Moreover, the conditional stan-

dard deviation estimates and, thus, the optimal allocation fractions become more accurate. Also, an equiprobable strata structure that is constructed under the IS density provides a clever strata structure for the original density, thus, enhances the efficiency of the stratified estimator.

6. OASIS FOR FINANCIAL RISK SIMULATION

6.1. Risk Measures

In finance, risk measures are used for quantifying the risk level of investments. Most of them are related with the loss distribution and the probabilistic analysis on the amount of loss given that we lose more than a specified percentage.

One value of interest is the tail loss probability (TLP), that is the probability that the portfolio loss exceeds a certain threshold level τ at the and of a fixed horizon.

$$TLP(\tau) = \Pr\{Loss > \tau\} = E\left[\mathbb{1}_{\{Loss>\tau\}}\right].$$

Estimating tail loss probabilities for many threshold values gives information about the general behavior of the tail distribution.

Value-at-risk (VaR), as a risk measure, is defined as the $1-\alpha$ quantile of the loss distribution for a given probability α.

$$VaR(1-\alpha) = \inf\{\tau : \Pr\{Loss > \tau\} < \alpha\}$$

This particular value provides a simple way of summarizing information about the tail distribution and is often interpreted as a reasonable worst-case loss level.

Any attempt to summarize a distribution in a single value is open to criticism, but VaR has a particular deficiency stressed by Artzner *et al.* (1999): combining two portfolios in a single portfolio may result in a VaR that is larger than the sum of the VaRs for the two original portfolios. This runs counter to the idea that diversification reduces risk.

Another value of interest is the conditional excess (CEx), i.e., the expected loss given that the loss exceeds a specified threshold τ:

$$CEx\left(\tau\right) = E\left[Loss|Loss > \tau\right] = \frac{E\left[Loss\mathbb{1}_{\{Loss>\tau\}}\right]}{E\left[\mathbb{1}_{\{Loss>\tau\}}\right]}. \tag{6.1}$$

Some risk measures are free of the shortcomings of VaR. An important example is the expected shortfall (ES), which is an alternative to VaR for being more sensitive to the shape of the tail loss distribution. The expected shortfall at $1-\alpha$ level is the expected loss of the portfolio in the worst α of the cases.

$$ES\left(\alpha\right) = E\left[Loss|Loss > VaR\left(1-\alpha\right)\right]$$

Expected shortfall is also a coherent risk measure and is frequently called with other names: Conditional value at risk, average value at risk, and expected tail loss. By satisfying the sub-additivity condition, conditional excess is regarded as a coherent risk measure (see Artzner *et al.*, 1999).

In our study, we mainly consider the problem of estimating tail loss probability and conditional excess values. Because with tail loss probabilities estimated for multiple threshold values, we can use spline interpolation to approximate the CDF of the loss distribution and root finding algorithms to obtain VaR values for arbitrary probabilities. If we have multiple conditional excess values as well, the same approach can be used to calculate expected shortfall, again for arbitrary probabilities.

6.2. Risk of Asset Portfolios

An asset portfolio is an investment in several assets (typically stocks) at the same time. The idea of holding a portfolio is that it is expected that this reduces the risk according to the well known principle: "Do not put all eggs into a single basket". It is very important to understand that we cannot expect that the returns of different stocks

are independent. It is therefore essential for a realistic stochastic model to model the joint distribution of the logarithmic returns (logreturns) of all stocks contained in the portfolio.

6.2.1. Multinormal Model

In the multinormal model, the logreturns are modeled with multinormal distribution. Assuming that the dependence across the assets are introduced within the model, the portfolio loss then can be represented as a function of the latent variable $\boldsymbol{Z}$ that follows standard multinormal distribution.

Consider a portfolio of D linear stocks. We assume that the stock log-returns follow a multinormal distribution with mean vector, $\boldsymbol{\mu}$, volatility vector, $\boldsymbol{\sigma}$, and correlation matrix $\boldsymbol{R}$. Then, the portfolio return is:

$$Return\,(\boldsymbol{Z}) = \sum_{d=1}^{D} w_d \exp\{\mu_d + \sigma_d \sum_{k=1}^{D} L_{dk} Z_k\},$$

where $\boldsymbol{w} = (w_1, \ldots, w_D)'$ holds the fraction of investments in each stock and $\boldsymbol{L}$ is the lower triangular Cholesky factorization of the correlation matrix $\boldsymbol{R}$. The loss of the portfolio is calculated as $Loss\,(\boldsymbol{Z}) = S_0(1 - Return\,(\boldsymbol{Z}))$ where S_0 is the initial amount of investment.

6.2.2. The t-Copula Model

In the t-Copula model, the logreturns are modeled with a copula and the latent variable $\boldsymbol{T}$ follows a multivariate T distribution.

Consider a portfolio of D linear stocks. The weight vector $\boldsymbol{w} = (w_1, \ldots, w_D)'$ holds the shares invested in stocks. The log-return vector $\boldsymbol{X} = (X_1, \ldots, X_D)'$ of the stocks is assumed to follow a t-copula with ν degrees of freedom. G_d denotes the CDF of the marginal distribution of the d-th log-return and F_ν denotes the CDF of a t-

distribution with ν degrees of freedom. Let σ_d denote the yearly volatility parameter and var_d denote the variance of the d-th marginal distribution. We define the scaling factor c_d for the d-th log-return:

$$c_d := \left[\sigma_d^2/(252\ var_d)\right]^{1/2}. \tag{6.2}$$

Then, the log-returns $X_d := c_d G_d^{-1}\left(F_\nu\left(T_d\right)\right)$, $d = 1, \ldots, D$ are generated using a multi-t distributed random vector $\boldsymbol{T} := (T_1, \ldots, T_D)'$ with ν degrees of freedom.

Suppose, we are given the correlation matrix $\boldsymbol{\Lambda}$ which represents the linear dependence structure of the stocks and let $\boldsymbol{L} \in \mathbb{R}^{D\times D}$ be the lower triangular Cholesky factor of $\boldsymbol{\Lambda}$ satisfying $\boldsymbol{L}\boldsymbol{L}' = \boldsymbol{\Lambda}$. We use the standard approach to generate $\boldsymbol{T}$ described in Section 3.4 using a standard multinormal vector $\boldsymbol{Z} \in \mathbb{R}^D$ and a chi-squared random variable Y with ν degrees of freedom. However, this time we introduce dependence across the elements of $\boldsymbol{T}$:

$$\boldsymbol{T} = \boldsymbol{L}\boldsymbol{Z}(Y/\nu)^{-1/2}. \tag{6.3}$$

The portfolio return is formulated as the weighted average of stock returns, that is:

$$Return\left(\boldsymbol{T}\right) := \sum_{d=1}^{D} w_d \exp\left(c_d G_d^{-1}\left(F_\nu\left(T_d\right)\right)\right), \tag{6.4}$$

with $\boldsymbol{T} = (T_1, ..., T_D)'$ vector generated according to Equation 6.3. The loss of the portfolio is calculated as

$$Loss\left(\boldsymbol{T}\right) = S_0(1 - Return\left(\boldsymbol{T}\right)) \tag{6.5}$$

where S_0 is the initial amount of investment.

Sak *et al.* (2010) propose an efficient IS algorithm in order to estimate tail loss probability $\Pr\{Loss(\boldsymbol{T}) \geq \tau\}$. The IS density is constructed by adding a shift $\boldsymbol{\mu}$ to the standard multi-normal vector $\boldsymbol{Z}$. In addition, the chi-squared random variable Y is considered as a gamma random variable with shape and scale parameters equal to $\nu/2$ and 2 respectively. Then, the algorithm sets a new value to the scale parameter. The shift vector and the scale values are selected so that the mode of the resulting IS density is equal to the mode of the zero-variance IS function (see e.g., Glasserman *et al.*, 1999). For determining the IS parameters, the algorithm utilizes an efficient quasi-Newton method, namely the constrained version of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) algorithm suggested by Byrd *et al.* (1995). Also, for generation of the log-returns, a fast polynomial inversion (PINV) algorithm is used (see Derflinger *et al.*, 2010). The numerical results show that the IS algorithm is fast and reliable for realistic marginal distributions such as the generalized hyperbolic distribution.

6.3. The OASIS Algorithm for the t-copula Model

As we have stated in Chapter 1, in the OASIS algorithm, we combine the IS with the OAS. Our main aim is to use the OASIS algorithm to minimize the overall error of multiple tail loss probability (or conditional excess) estimates. This section contains the extensions of the stratified importance sampling algorithm of Başoğlu *et al.* (2013) to different type of overall error functions given in Başoğlu and Hörmann (2014).

For the t-copula model, we suggest using the IS methodology of Sak *et al.* (2010) in the OASIS algorithm. Therefore, in Appendix B, we give the pseudo-code of their algorithm which returns the optimal IS parameters. As for OAS, the modified AOA algorithm is used (see Figure 2.1). The random input is stratified with a two-dimensional stratification. The multi-normal input $\boldsymbol{Z}$ is stratified over the direction of the IS shift $\boldsymbol{v} = \boldsymbol{\mu}^*/\|\boldsymbol{\mu}^*\|$. The gamma random variable Y, which was originally a chi-squared random variable with ν degrees of freedom, is also stratified.

6.3.1. Initialization

Stratification of a D-dimensional standard multi-normal vector $\boldsymbol{Z}$ along the direction $\boldsymbol{v}$ can be realized through stratifying its linear projection over that direction. This method requires $\mathrm{O}(D)$ operations and additional independent standard normal variables by the number of stratified directions (see Section 3.3.1). It might be sensible to use this approach in order to decrease the computational cost when D is large. However, the computational complexity of Equation 6.3 is already $\mathrm{O}(D^2)$. Therefore, we use the linear transformation of the multi-normal input (see Section 3.3.1.1) so that the first element Z_1 of the input vector $\boldsymbol{Z}$ will correspond to direction $\boldsymbol{v}$. In that case, Z_1 will be the only element to be stratified in $\boldsymbol{Z}$. This is achieved by premultiplying $\boldsymbol{Z}$ with an orthogonal matrix $\boldsymbol{V} \in \mathbb{R}^{D\times D}$ that has its first column equal to $\boldsymbol{v}$ and the remaining columns can be selected using the algorithm in Figure 3.5.

Computationally, it is not cheap to multiply the Cholesky matrix $\boldsymbol{L}$ with the transformed vector $\boldsymbol{VZ}$ each time we want to generate a multi-t vector. For this reason, we save it in matrix $\boldsymbol{A} = \boldsymbol{LV}$ and we replace Equation 6.3 by:

$$\boldsymbol{T} = (\boldsymbol{LV})\,\boldsymbol{Z}(Y/\nu)^{-1/2} = \boldsymbol{AZ}(Y/\nu)^{-1/2}. \tag{6.6}$$

At this point, the initialization of the OASIS algorithm ends. The pseudo-code of the initialization is given in Figure 6.1.

6.3.2. Main Run

For the main run, we set the total sample size to N and the sample sizes N^k for each iteration k such that $N = \sum_{k=1}^{K} N^k$.

For the IS part, we use the IS parameters $\boldsymbol{\mu}^*$ and γ^* calculated with algorithm in Figure B.1. In the OASIS algorithm, after the linear transformation, the first element

Require: Parameters of the t-copula model; portfolio loss threshold τ

Ensure: IS parameters $\boldsymbol{\mu}$ and γ; premultiplier matrix $\boldsymbol{A}$

1: compute Cholesky factor $\boldsymbol{L}$ of $\boldsymbol{\Sigma}$, i.e., $\boldsymbol{L}\boldsymbol{L}' = \boldsymbol{\Sigma}$

2: compute c_d, for $d = 1, \ldots, D$, using Equation 6.2

3: compute $\boldsymbol{\mu}^*$ and γ^* using Algorithm in Figure B.1

4: set $\boldsymbol{v} = \boldsymbol{\mu}^*/\boldsymbol{\mu}^*$

5: call the algorithm in Figure 3.5 with input $\boldsymbol{v}$ to construct linear transformation matrix $\boldsymbol{V}$

6: compute the premultiplier matrix $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{V}$

7: return $\boldsymbol{\mu}^*$, γ^* and $\boldsymbol{A}$

Figure 6.1. OASIS Algorithm for the t-copula model: Initialization.

Z_1 of the standard multi-normal input corresponds to the shift direction. Therefore, we add the shift $\|\boldsymbol{\mu}^*\|$ to Z_1. The remaining elements $Z_2, \ldots, Z_D$ are not changed. Similarly, the gamma random variable Y will be stratified under the IS scale parameter γ^*. The IS algorithm moves the sampling process to the main region where the portfolio returns are small. As a consequence, the variance of the simulation function decreases over the sampling domain, especially over the stratification directions. Therefore, for both Z_1 and Y, it is sufficient to use a moderate number of equiprobable strata. As we use two-dimensional stratification, we require the index sets $i_1 = 1, \ldots, I_1$ for the multi-normal input and $i_2 = 1, \ldots, I_2$ for the gamma random variable. Thus, the total number of equiprobable strata is $I = I_1 \times I_2$. The stratum index $i = 1, \ldots, I$ corresponds to the equiprobable interval $i_1 = \lceil i/I_1 \rceil$ of the standard normal distribution and the equiprobable interval $i_2 = i - (i_1 - 1)\, I_1$ of the gamma distribution. The stratum probabilities p_i, $i = 1, \ldots, I$ are simply equal to I^{-1}.

The stratified random variables are generated through independent and identically distributed standard uniform variables U_1 and U_2. We generate Z_1 conditional on the equiprobable interval i_1 using the inverse CDF of the standard normal distribution

Φ^{-1}:

$$Z_1 = \|\boldsymbol{\mu}^*\| + \Phi^{-1}\left(\left(i_1 - 1 + U_1\right)/I_1\right), \tag{6.7}$$

and the gamma random variable conditional on the equiprobable interval i_2 using the inverse CDF of the gamma distribution F_Γ^{-1}:

$$Y = F_\Gamma^{-1}\left(\left(i_2 - 1 + U_2\right)/I_2; \nu/2, \gamma^*\right), \tag{6.8}$$

under the shape parameter $\nu/2$ and the scale parameter γ^*. Unfortunately, the computational cost of F_Γ^{-1} is large. Therefore, we suggest using the PINV algorithm of Derflinger *et al.* (2010) to generate the stratified gamma random variable as well. Figure 6.2 gives the pseudo code of the generator of stratified multivariate t-distributed vector for the OASIS algorithm.

Require: Stratum index i

Ensure: A random drawing of $\boldsymbol{T}$ conditional on stratum i

1: set $i_1 = \lceil i/I_1 \rceil$ and $i_2 = i - (i_1 - 1)\, I_1$

2: generate $U_1 = U(0,1)$ and set Z_1 using Eq. 6.7

3: generate $Z_d \sim N(0,1)$, $d = 2, \ldots, D$ independently

4: generate $U_2 = U(0,1)$ independently and set Y using Eq. 6.8

5: compute and return $\boldsymbol{T}$ using Eq. 6.6 and the PINV algorithm

Figure 6.2. Generator for stratified multi-t vector for the OASIS algorithm.

The loss of the portfolio is calculated using Equation 6.5. The simulation function $\mathbb{1}_{\{Loss(\boldsymbol{T})>\tau\}}$ will be weighted with the IS ratio:

$$\rho\left(\boldsymbol{T}\right) = \rho\left(Z_1, Y\right) = e^{-Z_1\|\boldsymbol{\mu}^*\| + \|\boldsymbol{\mu}^*\|^2/2 - Y/2 + Y/\gamma^* + \ln(\gamma^*/2)\nu/2}, \tag{6.9}$$

and the weighted values generated in stratum i will be stored in set H_i.

At the end of iteration k, the sample standard deviation $\hat{s}_i^k$ in each set H_i is com-

puted. The allocation fractions for iteration k are determined by Equation 2.12 and, subsequently, the allocation of the drawings in iteration k is determined by Equation 2.13.

Finally, in the last iteration, the sample mean $\hat{y}_i^K$ in each set H_i is computed. The tail loss probability estimate and its variance are calculated using Equations 2.14 and 2.15. The pseudo code of OASIS for a single probability estimate is given in Figure 6.3.

Require: Parameters of the t-copula model; portfolio loss threshold τ

Ensure: Tail loss probability estimate $\hat{y}^{OASIS}$ and its variance $V\left[\hat{y}^{OASIS}\right]$

initialize with the algorithm in Figure 6.1
set $H_i = \emptyset$, $M_i^0 = 0$, and $\pi_i^1 = p_i$ for $i = 1, \ldots, I$
for iteration $k = 1, \ldots, K$ **do**
  calculate N_i^k using Eq. 2.12 and 2.13, and set $M_i^k = N_i^k + M_i^{k-1}$ for $i = 1, \ldots, I$
  for stratum index $i = 1, \ldots, I$ **do**
    for drawing $n = 1, \ldots, N_i^k$ **do**
      generate $\boldsymbol{T}$ using algorithm in Figure 6.2
      compute $\rho(\boldsymbol{T})\, \mathbb{1}_{\{Loss(\boldsymbol{T})>\tau\}}$ using Eq. 6.5 and 6.9, and add it to set H_i
    end for
    compute sample mean $\hat{y}_i^k$ and sample standard deviation $\hat{s}_i^k$ in set H_i
  end for
end for
compute and return $\hat{y}^{OASIS}$ and $V\left[\hat{y}^{OASIS}\right]$ using respectively Eq. 2.14 and 2.15

Figure 6.3. OASIS algorithm for the t-copula model: Single tail loss probability estimate.

In order to estimate the conditional excess $E\left[Loss\left(\boldsymbol{T}\right)|Loss\left(\boldsymbol{T}\right) > \tau\right]$, we follow Glasserman *et al.* (2002) and use the same IS density and the likelihood ratio when estimating the numerator and the denominator in Equation 6.1. In fact, we do not require an optimal IS density for conditional excess, since the variance that is not reduced by the IS algorithm is reduced by the subsequent stratification. Thus, we just use the IS density that is optimal for tail loss probability estimates.

The OASIS algorithm for the conditional excess is similar to the one for the tail loss probability except a few changes. The conditional excess has a ratio estimator (see Equation 6.1). Therefore, we need to evaluate two simulation functions for each drawing of the random input $\boldsymbol{T}$: $Loss\,(\boldsymbol{T})\,\mathbb{1}_{\{Loss(\boldsymbol{T})>\tau\}}$ and $\mathbb{1}_{\{Loss(\boldsymbol{T})>\tau\}}$. Both simulation functions will be weighted with the IS ratio in Equation 6.9 and the weighted values generated in stratum i will be stored respectively in sets H_{i1} and H_{i2}.

At the end of iteration k, the sample mean $\hat{y}_{i1}^k$ and the sample standard deviation $\hat{s}_{i1}^k$ in set H_{i1}, the sample mean $\hat{y}_{i2}^k$ and the sample standard deviation $\hat{s}_{i2}^k$ in set H_{i2}, and the sample covariance $\hat{s}_{i12}^k$ of sets H_{i1} and H_{i2} are computed. The estimators of the numerator and the denominator of Equation 6.1 are also calculated as $\hat{y}_1^k$ and $\hat{y}_2^k$ using Equation 2.14. The allocation fractions for iteration k are determined by plugging in these values in Equation 4.7 and, subsequently, the allocation of the drawings in iteration k is determined by Equation 2.13.

Finally, at the end of the final iteration, the conditional excess estimate is calculated as the ratio $\hat{y}_{i1}^K/\hat{y}_{i2}^K$. The variance of the ratio estimator is calculated using Equation 4.6. The pseudo code of OASIS for a single conditional excess estimate is given in Figure 6.4.

6.3.3. Multiresponse Setting

As we have stated previously, we are mainly interested in efficient estimation of tail loss probabilities (or conditional excess values) for several threshold values τ_j, $j = 1, \ldots, J$ in a single simulation.

In order to minimize the overall error of tail loss probability estimates, we utilize the OASIS algorithm and consider the optimization models described in Section 4. For the IS density, one can suggest using density mixtures as described in Sak and Hörmann (2012) since the optimal IS parameters are different for each threshold value. However, since the stratification of mixture distributions is too complicated, we use a single IS

Require: Parameters of the t-copula model; portfolio loss threshold τ

Ensure: Conditional excess estimate $\hat{y}^{OASIS}$ and its variance $V\left[\hat{y}^{OASIS}\right]$

initialize with the algorithm in Figure 6.1
set $H_{i1} = H_{i2} = \emptyset$, $M_i^0 = 0$, and $\pi_i^1 = p_i$ for $i = 1, \ldots, I$
for iteration $k = 1, \ldots, K$ **do**
  calculate N_i^k using Eq. 4.7 and 2.13, and set $M_i^k = N_i^k + M_i^{k-1}$ for $i = 1, \ldots, I$
  for stratum index $i = 1, \ldots, I$ **do**
    for drawing $n = 1, \ldots, N_i^k$ **do**
      generate $\boldsymbol{T}$ using algorithm in Figure 6.2
      compute $\rho(\boldsymbol{Z}, Y)\, Loss(\boldsymbol{T})\, \mathbb{1}_{\{Loss(\boldsymbol{T})>\tau\}}$ using Eq. 6.5 and 6.9, and add it to set H_{i1}
      compute $\rho(\boldsymbol{Z}, Y)\, \mathbb{1}_{\{Loss(\boldsymbol{T})>\tau\}}$ using Eq. 6.5 and 6.9, and add it to set H_{i2}
    end for
    compute sample mean $\hat{y}_{i1}^k$ and sample standard deviation $\hat{s}_{i1}^k$ of set H_{i1}
    compute sample mean $\hat{y}_{i2}^k$ and sample standard deviation $\hat{s}_{i2}^k$ of set H_{i2}
    compute sample covariance $\hat{s}_{i12}^k$ of sets H_{i1} and H_{i2}
    compute $\hat{y}_1^k$ and $\hat{y}_2^k$ using Equation 2.14
  end for
end for
compute and return $\hat{y}^{OASIS} = \hat{y}_1^K / \hat{y}_2^K$ and $V\left[\hat{y}^{OASIS}\right]$ using Eq. 4.6

Figure 6.4. OASIS algorithm for the t-copula model: Single conditional excess estimate.

density that is reasonably appropriate for all estimates. As stated in Section 6.3.2, we do not require an optimal IS density since the variance that is not reduced by the IS algorithm is reduced by the subsequent stratification. For that matter, we choose a threshold value between τ_{min} and τ_{max} and determine an IS density with respect to that threshold. We denote this threshold with $\tau^* = \zeta\tau_{max} + (1-\zeta)\tau_{min}$ where $\zeta \in [0, 1]$. Since the IS part of the OASIS algorithm is not too critical, we can obtain reasonably good results unless ζ is too close to zero or one. Once we choose τ^*, we utilize the initialization algorithm in Figure 6.1 to set the IS parameters.

For the stratification part, the strata structure and the generation of the stratified random input remains the same as in Figure 6.2. Assuming that we are in stratum i, $Loss(\boldsymbol{T})$ and the IS weights $\rho(\boldsymbol{T})$ are calculated using Equations 6.5 and 6.9, respectively. At this point, the weighted responses $\rho(\boldsymbol{T})\mathbb{1}_{\{Loss(\boldsymbol{T})>\tau_j\}}$ are calculated for each threshold value τ_j, $j = 1, \ldots, J$ and stored in respective sets H_{ij}. At the end of iteration k, the sample mean $\hat{y}_{ij}^k$ and the sample standard deviation $\hat{s}_{ij}^k$ of each set H_{ij} are computed. The tail loss probability estimates found at the end of iteration k are calculated as $\hat{y}_j^k = I^{-1}\sum_{i=1}^{I}\hat{y}_{ij}^k$ for $j = 1, \ldots, J$.

The allocation fractions π_i^1 of the first iteration are again selected proportionally as $p_i = I^{-1}$, $i = 1, \ldots, I$. To determine the allocation fractions π_i^k, $i = 1, \ldots, I$ for $k \geq 2$, we first choose an objective function which represents the overall error of the simulation (see Section 4). Then, we solve for the optimal allocation fractions π_i^k using the appropriate methodology.

- If the objective is to minimize the mean squared error of all estimates, we consider objective function $\omega_{MSE}(\boldsymbol{\pi}) = \sum_{j=1}^{J}\Sigma_{jj}(\boldsymbol{\pi})$. We use Equation 4.5 by setting $p_i = I^{-1}$ for $i = 1, \ldots, I$, all coefficients $c_{jj} = 1$ for $j = 1, \ldots, J$, and the remaining coefficients to zero.
- If the objective is to minimize the mean squared relative error of all estimates, we consider objective function $\omega_{MSR}(\boldsymbol{\pi}) = \sum_{j=1}^{J}\hat{y}_j^{-2}\Sigma_{jj}(\boldsymbol{\pi})$. We use Equation 4.5 by setting $p_i = I^{-1}$ for $i = 1, \ldots, I$, all coefficients $c_{jj} = \hat{y}_j^{-2}$ for $j = 1, \ldots, J$, and the remaining coefficients to zero.
- If the objective is to minimize the maximum absolute error of all estimates, we consider objective function $\omega_{MAXE}(\boldsymbol{\pi}) = \max\{j : \Sigma_{jj}(\boldsymbol{\pi})\}$. We call the heuristic algorithm in Figure 4.2 by setting $\omega_j(\boldsymbol{\pi}) = \sum_{i=1}^{I}\pi_i^{-1}\hat{s}_{ij}^2$ for $j = 1, \ldots, J$.
- If the objective is to minimize the maximum absolute relative error of all estimates, we consider objective function $\omega_{MAXR}(\boldsymbol{\pi}) = \max\{j : \hat{y}_j^{-2}\Sigma_{jj}(\boldsymbol{\pi})\}$. We again call the heuristic algorithm in Figure 4.2 by setting $\omega_j(\boldsymbol{\pi}) = \hat{y}_j^{-2}\sum_{i=1}^{I}\pi_i^{-1}\hat{s}_{ij}^2$ for $j = 1, \ldots, J$.

Require: Iteration index k; conditional estimates $\hat{y}_{ij}$, $\hat{s}_{ij}$ for $i = 1, \ldots, I$ and $j = 1, \ldots, J$ calculated in the previous iteration; the objective function that represents the overall error

Ensure: Optimal allocation fractions π_i^k.

if $k = 1$ **then**
  set $\pi_i^k = I^{-1}$ for $i = 1, \ldots, I$
else
  if objective function is $\omega_{MSE}(\boldsymbol{\pi})$ **then**
    set $\pi_i^k = \left(\sum_{j=1}^{J} \hat{s}_{ij}^2\right)^{1/2} \Big/ \sum_{l=1}^{I} \left(\sum_{j=1}^{J} \hat{s}_{lj}^2\right)^{1/2}$ for $i = 1, \ldots, I$
  else if objective function is $\omega_{MSR}(\boldsymbol{\pi})$ **then**
    compute $\hat{y}_j = I^{-1} \sum_{i=1}^{I} \hat{y}_{ij}$ for $j = 1, \ldots, J$
    set $\pi_i^k = \left(\sum_{j=1}^{J} \hat{y}_j^{-2} \hat{s}_{ij}^2\right)^{1/2} \Big/ \sum_{l=1}^{I} \left(\sum_{j=1}^{J} \hat{y}_j^{-2} \hat{s}_{lj}^2\right)^{1/2}$ for $i = 1, \ldots, I$
  else if objective function is $\omega_{MAXE}(\boldsymbol{\pi})$ **then**
    set $\omega_j(\boldsymbol{\pi}) = \sum_{i=1}^{I} \pi_i^{-1} \hat{s}_{ij}^2$ for $j = 1, \ldots, J$
    call the heuristic algorithm in Fig. 4.2 for π_i^k
  else if objective function is $\omega_{MAXR}(\boldsymbol{\pi})$ **then**
    compute $\hat{y}_j = I^{-1} \sum_{i=1}^{I} \hat{y}_{ij}$ and set $\omega_j(\boldsymbol{\pi}) = \hat{y}_j^{-2} \sum_{i=1}^{I} \pi_i^{-1} \hat{s}_{ij}^2$ for $j = 1, \ldots, J$
    call the heuristic algorithm in Fig. 4.2 for π_i^k
  end if
end if
return π_i^k

Figure 6.5. Optimal allocation fractions for multiple tail loss probability estimates.

The number of drawings from stratum i in iteration k is determined using π_i^k and Equation 2.13 for $i = 1, \ldots, I$ and $k = 1, \ldots, K$.

Finally, in the last iteration, we use the values $\hat{y}_{ij}^K$ and $\hat{s}_{ij}^K$ to estimate the tail loss probabilities

$$\hat{y}_j = I^{-1} \sum_{i=1}^{I} \hat{y}_{ij}^K, \; j = 1, \ldots, J, \tag{6.10}$$

their variances

$$V\left[\hat{y}_j\right] = I^{-2}\sum_{i=1}^{I}\left(\hat{s}_{ij}^{K}\right)^2 \Big/ M_i^K, \quad j = 1, \ldots, J, \tag{6.11}$$

and percentage relative errors

$$RE\left[\hat{y}_j\right] = 100 \times \hat{y}_j^{-1}\Phi^{-1}\left(0.975\right)V\left[\hat{y}_j\right]^{1/2}, \quad j = 1, \ldots, J. \tag{6.12}$$

The pseudo code of OASIS for multiple probability estimates is given in Algorithm 6.6.

In order to estimate multiple conditional excess values $E\left[Loss\left(\boldsymbol{T}\right)|Loss\left(\boldsymbol{T}\right) > \tau_j\right]$ for $j = 1, \ldots, J$, we use the same IS density and the likelihood ratio used for estimating multiple tail loss probabilities.

The OASIS algorithm for multiple conditional excess values is similar to the one for multiple tail loss probabilities except the changes regarding to the ratio estimator of conditional excess. We need to evaluate two simulation functions for each drawing of the random input $\boldsymbol{T}$ and each threshold τ_j: $Loss\left(\boldsymbol{T}\right)\mathbb{1}_{\{Loss(\boldsymbol{T})>\tau_j\}}$ and $\mathbb{1}_{\{Loss(\boldsymbol{T})>\tau_j\}}$ for $j = 1, \ldots, J$. Both simulation functions will be weighted with the IS ratio in Equation 6.9 and the weighted values generated in stratum i will be stored respectively in sets H_{ij1} and H_{ij2}. At the end of iteration k, the sample mean $\hat{y}_{ij1}^k$ and the sample standard deviation $\hat{s}_{ij1}^k$ in set H_{ij1}, the sample mean $\hat{y}_{ij2}^k$ and the sample standard deviation $\hat{s}_{ij2}^k$ in set H_{ij2}, and the sample covariance $\hat{s}_{ij12}^k$ of sets H_{ij1} and H_{ij2} are computed for $j = 1, \ldots, J$ and each stratum i. The estimators of the numerator and the denominator of Equation 6.1 are also calculated as $\hat{y}_{j1}^k$ and $\hat{y}_{j2}^k$ for each response j using Equation 2.14.

The allocation fractions π_i^1 of the first iteration are again selected proportionally as $p_i = I^{-1}$, $i = 1, \ldots, I$. To determine the allocation fractions π_i^k, $i = 1, \ldots, I$ for

Require: Parameters of the t-copula model; portfolio loss thresholds τ_j, $j = 1, \ldots, J$

Ensure: Tail loss probability estimates $\hat{y}_j^{OASIS}$, their variance $V\left[\hat{y}_j^{OASIS}\right]$ and relative errors $RE\left[\hat{y}_j^{OASIS}\right]$ for $j = 1, \ldots, J$

initialize with Algorithm 6.1 with respect to $\tau^* = \zeta\tau_{max} + (1-\zeta)\tau_{min}$
set $H_{ij} = \emptyset$ for $i = 1, \ldots, I$ and $j = 1, \ldots, J$
set $M_i^0 = 0$ and $\pi_i^1 = p_i$ for $i = 1, \ldots, I$
for iteration $k = 1, \ldots, K$ **do**
  for stratum index $i = 1, \ldots, I$ **do**
    determine π_i^k using algorithm in Figure 6.5
    calculate N_i^k using π_i^k and 2.13, and set $M_i^k = N_i^k + M_i^{k-1}$
    for drawing $n = 1, \ldots, N_i^k$ **do**
      generate $\boldsymbol{T}$ using algorithm in Figure 6.2
      for threshold $j = 1, \ldots, J$ **do**
        compute $\rho(\boldsymbol{T})\mathbb{1}_{\{Loss(\boldsymbol{T})>\tau_j\}}$ using Eq. 6.5 and 6.9, and add it to set H_{ij}
      end for
    end for
    for threshold $j = 1, \ldots, J$ **do**
      compute sample mean $\hat{y}_{ij}^k$ and sample standard deviation $\hat{s}_{ij}^k$ of set H_{ij}
    end for
  end for
  compute $\hat{y}_j^k = I^{-1}\sum_{i=1}^{I}\hat{y}_{ij}^k$ for $j = 1, \ldots, J$
end for
compute and return $\hat{y}_j^{OASIS}$, $V\left[\hat{y}_j^{OASIS}\right]$, and $RE\left[\hat{y}_j^{OASIS}\right]$ for $j = 1, \ldots, J$ using Eq. 6.10, 6.11, and 6.12

Figure 6.6. OASIS algorithm for the t-copula model: Multiple tail loss probability estimates.

$k \geq 2$, we calculate $\hat{y}_j = \hat{y}_{j1}/\hat{y}_{j2}$ for $j = 1, \ldots, J$ and

$$\hat{s}_{ij} = \hat{y}_{j1}^2\hat{y}_{j2}^{-4}\hat{s}_{ij2}^2 - 2\hat{y}_{j1}\hat{y}_{j2}^{-3}\hat{s}_{ij12} + \hat{y}_{j2}^{-2}\hat{s}_{ij1}^2, \quad i = 1, \ldots, I, \quad j = 1, \ldots, J \tag{6.13}$$

using the estimates calculated in the k-th iteration. Then, we choose an objective function which represents the overall error of the simulation (see Section 4) and solve for the optimal allocation fractions π_i^k using the algorithm in Figure 6.5.

The number of drawings from stratum i in iteration k is determined using π_i^k and Equation 2.13 for $i = 1, \ldots, I$ and $k = 1, \ldots, K$.

Finally, in the last iteration, we use the values $\hat{y}_{ij1}^K$, $\hat{y}_{ij2}^K$ to calculate the numerator $\hat{y}_{j1} = I^{-1} \sum_{i=1}^{I} \hat{y}_{ij1}^K$ and the denominator $\hat{y}_{j2} = I^{-1} \sum_{i=1}^{I} \hat{y}_{ij2}^K$ for all ratio estimators $j = 1, \ldots, J$. The conditional excess estimates are calculated as:

$$\hat{y}_j = \hat{y}_{j1}/\hat{y}_{j2}, \quad j = 1, \ldots, J. \tag{6.14}$$

We use $\hat{y}_{j1}$, $\hat{y}_{j2}$ together with the conditional standard deviation estimates $\hat{s}_{ij1}^K$, $\hat{s}_{ij2}^K$ and the conditional covariance estimates $\hat{s}_{ij12}^K$ to estimate the variance of the estimators of conditional excess values

$$V\left[\hat{y}_j\right] = I^{-2} \left(\sum_{i=1}^{I} \frac{\hat{y}_{j1}^2 \hat{y}_{j2}^{-4} \left(\hat{s}_{ij2}^K\right)^2 - 2\hat{y}_{j1} \hat{y}_{j2}^{-3} \hat{s}_{ij12}^K + \hat{y}_{j2}^{-2} \left(\hat{s}_{ij1}^K\right)^2}{M_i^K} \right), \quad j = 1, \ldots, J, \tag{6.15}$$

and the percentage relative errors can be calculated using Equation 6.12.

The pseudo code of OASIS for multiple conditional excess estimates is given in Algorithm 6.7.

Require: Parameters of the t-copula model; portfolio loss thresholds τ_j, $j = 1, \ldots, J$

Ensure: Conditional excess estimates $\hat{y}_j^{OASIS}$, their variance $V\left[\hat{y}_j^{OASIS}\right]$ and relative errors $RE\left[\hat{y}_j^{OASIS}\right]$ for $j = 1, \ldots, J$

initialize with Algorithm 6.1 with respect to $\tau^* = \zeta\tau_{max} + (1-\zeta)\tau_{min}$
set $H_{ij1} = H_{ij2} = \emptyset$ for $i = 1, \ldots, I$ and $j = 1, \ldots, J$
set $M_i^0 = 0$ and $\pi_i^1 = p_i$ for $i = 1, \ldots, I$
for iteration $k = 1, \ldots, K$ **do**
  for stratum index $i = 1, \ldots, I$ **do**
    determine π_i^k using algorithm in Figure 6.5
    calculate N_i^k using π_i^k and 2.13, and set $M_i^k = N_i^k + M_i^{k-1}$
    for drawing $n = 1, \ldots, N_i^k$ **do**
      generate $\boldsymbol{T}$ using algorithm in Figure 6.2
      for threshold $j = 1, \ldots, J$ **do**
        compute $\rho(\boldsymbol{T})\, Loss(\boldsymbol{T})\, \mathbb{1}_{\{Loss(\boldsymbol{T})>\tau_j\}}$ using Eq. 6.5 and 6.9, and add it to set H_{ij1}
        compute $\rho(\boldsymbol{T})\, \mathbb{1}_{\{Loss(\boldsymbol{T})>\tau_j\}}$ using Eq. 6.5 and 6.9, and add it to set H_{ij2}
      end for
    end for
    for threshold $j = 1, \ldots, J$ **do**
      compute sample mean $\hat{y}_{ij1}^k$ and standard deviation $\hat{s}_{ij1}^k$ of set H_{ij1}
      compute sample mean $\hat{y}_{ij2}^k$ and standard deviation $\hat{s}_{ij2}^k$ of set H_{ij2}
      compute sample covariance $\hat{s}_{ij12}^k$ of sets H_{ij1} and H_{ij2}
      compute $\hat{y}_{j1}^k$ and $\hat{y}_{j2}^k$ using Equation 2.14 and $\hat{y}_j = \hat{y}_{j1}^k / \hat{y}_{j2}^k$ for $j = 1, \ldots, J$
      compute $\hat{s}_{ij}$ using Equation 6.13
    end for
  end for
end for
compute and return $\hat{y}_j^{OASIS}$, $V\left[\hat{y}_j^{OASIS}\right]$, and $RE\left[\hat{y}_j^{OASIS}\right]$ for $j = 1, \ldots, J$ using Eq. 6.14, 6.15, and 6.12

Figure 6.7. OASIS algorithm for the t-copula model: Multiple conditional excess estimates.

7. NUMERICAL EXPERIMENTS

In this chapter, we present experiments and their results to measure the performance of the OASIS algorithm for the risk simulation for linear asset portfolios under the t-copula model. Some of these results are also presented in Başoğlu *et al.* (2013).

7.1. Preliminaries

To evaluate the performance of the OASIS algorithm, we need a measure of efficiency for its estimators. As explained in Section 2.1, the efficiency of an unbiased estimator is measured by its error bound and its expected computation time. In Lemieux (2009), the efficiency of an estimator $\hat{y}$ for a quantity y is defined as:

$$Eff\left(\hat{y}\right) = \left[MSE\left(\hat{y}\right) \times TM\left(\hat{y}\right)\right]^{-1} \tag{7.1}$$

where $MSE\left(\hat{y}\right) = V\left[\hat{y}\right] + Bias^2\left[\hat{y}\right]$ is the mean-square error of $\hat{y}$, $Bias\left[\hat{y}\right] = E\left[\hat{y}\right] - y$ is the bias of $\hat{y}$, and $TM\left(\hat{y}\right)$ is the expected computation time of $\hat{y}$. Since the methods that are used in the OASIS algorithm yield asymptotically unbiased estimators (or estimators with smaller bias compared to the mean squared error), the bias in Equation 7.1 can simply be ignored.

For a specified estimator, the efficiency ratio is defined as the ratio of the efficiencies of the naive estimator and the specified estimator. Namely:

$$ER\left(\hat{y}\right) = \frac{Eff\left(\hat{y}\right)}{Eff\left(\hat{y}^{NV}\right)} = \frac{V\left[\hat{y}^{NV}\right] TM\left(\hat{y}^{NV}\right)}{V\left[\hat{y}\right] TM\left(\hat{y}\right)}.$$

Thus, if the efficiency ratio is greater than one, then the specified estimator is considered to be more efficient than the naive estimator.

The performance results provided in this chapter are given in efficiency ratios.

Occasionally, the variance reduction factors $VRF(\hat{y}) = V\left[\hat{y}^{NV}\right]/V[\hat{y}]$ and the execution time $TM(\hat{y})$ of the algorithms are tabulated.

For our experiments, we use real world portfolios of D stocks ($D = 2$, 5, and 10) with equal weights ($w_d = D^{-1}$, $d = 1, \ldots, D$). The initial investment is fixed as $S_0 = 1$. We assume that the marginal distributions of the log-returns follow t or generalized hyperbolic distributions. As parameters of the marginal distributions, the fitted values for New York Stock Exchange (NYSE) data reported in Halulu (2012) are used. Halulu (2012) uses a two-step maximum likelihood procedure for estimating the parameters of the marginal distributions and, then, the copula parameters for the daily log-returns. Therefore, it is safe to consider that the numerical tests for variance reduction factors and execution times reflect the values that could be obtained for real world stock portfolios. We provide an instance of the fitted parameters of daily log-returns for $D = 5$ stocks under the t-copula model with generalized hyperbolic marginals, since the generalized hyperbolic distribution is considered to be the best fitting and the most flexible distribution due to its asymmetric density (see e.g. Aas and Haff, 2006; Behr and Potter, 2009). The fitted degrees of freedom of the t-copula is $\nu = 8.195$. The parameters of the marginal distributions are presented in Table 7.1 and the correlation matrix for the t-copula model is presented in Table 7.2. The abbreviations respectively stand for C: Citigroup Inc., CMS: CMS Energy Corp., F: Ford Motor Co., MO: Altria Group Inc, WMT: Wal-Mart Stores. We direct the reader to Halulu (2012) for more information on the other portfolio instances and parameter sets.

Table 7.1. Fitted parameters of generalized hyperbolic marginals for $D = 5$ stocks in NYSE under the t-copula model.

Stocks	lambda	alpha	delta	beta	mu
C	-0.602828	8.52771	0.014492	-0.533197	-0.000091
CMS	-1.331923	2.72759	0.019891	-2.573416	0.001388
F	-1.602705	3.26482	0.035139	1.456542	-0.001662
MO	-1.131092	15.13351	0.014771	-1.722396	0.001304
WMT	-0.955118	31.14005	0.015362	0.896576	-0.000238

Table 7.2. Fitted correlation matrix for $D = 5$ stocks in NYSE under the t-copula model with generalized hyperbolic marginals.

Stocks	C	CMS	F	MO	WMT
C	1	0.554	0.632	0.419	0.400
CMS	0.554	1	0.495	0.540	0.479
F	0.632	0.495	1	0.426	0.445
MO	0.419	0.540	0.426	1	0.443
WMT	0.400	0.479	0.445	0.443	1

The total sample size N used in all simulations is selected as 10^5, however due to the sample allocation in Equation 2.13, the OASIS algorithm might slightly exceed the aimed total sample size. Following the suggestions of Étoré and Jourdain (2010), the OASIS algorithm terminates in three iterations, using approximately 10%, 40%, and 50% of the total sample size in each iteration, sequentially.

For all implementations of the OASIS algorithm in this chapter, the number of strata for both the multi-normal and the gamma input are selected as 22. Increasing the number of strata promises for more variance reduction. However, the marginal contribution of adding a stratum eventually decreases and the variance reduction becomes saturated. Increasing the number of strata also decreases the number of drawings to be allocated in each stratum in the first iteration of the OASIS algorithm, and may result in poor estimates of the conditional standard deviations. Thus, we decide $I_1 = I_2 = 22$, so that the number of strata does not exceed 500. In the general use of the OASIS algorithm, the choice for the number of strata should be decided on the simulation function q and the size of the sample used in the first iteration of the OASIS algorithm.

All simulation experiments in this chapter are implemented using the R software with the most recent version 3.1.1.

7.2. The OASIS Algorithm: Single Estimates Case

Our first set of experiments measure the performance of the OASIS algorithm in the estimation of a single tail loss probability. Along with the OASIS algorithm, the naive simulation and the IS algorithm proposed in Sak *et al.* (2010) is implemented. The timing results (TM), variance reduction factors (VR) and the efficiency ratios (ER) for the IS and OASIS algorithms are presented in Table 7.3. For each instance, we select two threshold values τ such that the tail loss probabilities are approximately equal to 0.05 and 0.001, as the efficiency results highly depend on the probability of the rare event.

Table 7.3. Variance reduction factors, execution times in seconds, and the efficiency ratios of tail loss probability estimates for IS and OASIS under t and GH marginals.

		$\Pr\{Loss(\boldsymbol{T}) > \tau) \approx 0.05$						$\Pr\{Loss(\boldsymbol{T}) > \tau) \approx 0.001$					
		IS			OASIS			IS			OASIS		
Marginals	D	VR	TM	ER	VR	TM	ER	VR	TM	ER	VR	TM	ER
t	2	5.5	1.09	4.3	103.2	1.64	54.1	224.3	0.96	217.2	6533.2	1.45	4190.3
	5	9.0	3.11	6.4	95.7	3.24	65.3	277.8	2.67	228.9	1867.2	2.88	1426.3
	10	9.6	5.72	7.6	154.6	6.01	116.5	213.8	5.22	180.2	685.1	5.41	557.2
GH	2	6.2	1.04	5.2	274.4	1.56	151.3	178.8	1.14	131.7	3829.7	1.51	2130.4
	5	9.2	3.06	6.7	81.2	3.34	54.4	289.6	2.84	222.3	3429.4	3.14	2381.0
	10	9.6	5.64	7.1	183.3	5.85	130.7	285.5	5.89	197.8	4296.9	6.36	2756.5

Table 7.3 shows that the combined method, OASIS, leads to considerable variance reduction compared to the lean IS methodology. The execution time of the OASIS algorithm does not significantly differ from the execution time of the IS algorithm in any of the instances. Thus, we can conclude that the OASIS algorithm outperforms the IS methodology in the estimation of tail loss probabilities for practically relevant examples. Moreover, the efficiency ratio of the OASIS estimators are considerably large, which means that the OASIS algorithm can be used as an efficient tool for practical applications.

Under the same experimental design, we run all algorithms for estimating a single

conditional excess value. Following Glasserman *et al.* (2002), we use the same IS density that is used for estimating the tail loss probability in both IS and OASIS. For the OASIS algorithm, we use optimal allocations that minimize the variance of the ratio estimator of conditional excess (see Figure 6.4). The timing results (TM), variance reduction factors (VR) and the efficiency ratios (ER) for the IS and OASIS algorithms are presented in Table 7.4.

Table 7.4. Variance reduction factors, execution times in seconds, and the efficiency ratios of conditional excess values for IS and OASIS under t and GH marginals.

		$\Pr\{Loss(\boldsymbol{T}\} > \tau) \approx 0.05$						$\Pr\{Loss(\boldsymbol{T}\} > \tau) \approx 0.001$					
		IS			OASIS			IS			OASIS		
Marginals	D	VR	TM	ER	VR	TM	ER	VR	TM	ER	VR	TM	ER
t	2	8.5	0.94	7.9	34.3	1.36	22.0	542.2	0.80	589.6	1389.0	1.23	982.5
	5	15.1	2.65	12.7	42.5	3.09	30.7	203.9	2.43	189.6	374.0	2.81	300.8
	10	15.7	5.68	12.0	56.8	6.12	40.2	505.7	4.86	452.7	1056.8	5.23	879.0
GH	2	7.8	0.89	7.4	35.9	1.45	21.1	258.0	1.00	221.9	686.2	1.35	437.2
	5	12.8	2.90	27.4	38.9	3.31	73.0	497.9	2.69	409.1	1080.2	3.10	770.1
	10	12.6	5.17	10.1	51.0	5.66	37.3	405.6	5.81	288.3	949.5	6.24	628.5

In Table 7.4, we observe that the OASIS algorithm outperforms the lean IS methodology also in the estimation of conditional excess values. Compared to the estimation of a single tail loss probability, the difference in the efficiency ratios is not too much, especially in the extreme rare event case ($\Pr\{Loss(\boldsymbol{T}\} > \tau) \approx 0.001$). However, the difference is enough to make the OASIS algorithm preferable also for the estimation of a single conditional excess value.

Remark 7.1. *Our experiments on lean stratification showed that, compared to OASIS, we obtain worse results with the same strata size and strata structure. Introducing IS allows to use a small number of equiprobable strata, which reduces the complexity of the stratification procedure. Moreover, the conditional standard deviation estimates become more accurate even with a small sample size.*

7.3. The OASIS Algorithm: Multiple Estimates Case

In our second set of experiments, we consider computing tail loss probabilities at multiple threshold values in a single simulation. We use the algorithm in Figure 6.6 to estimate multiple tail loss probabilities for a portfolio of $D = 5$ stocks under the t-copula model with generalized hyperbolic marginals. For the marginal distributions, we use the parameters given in Table 7.1 and the correlation matrix of the t-copula is given in Table 7.2. The degrees of freedom is fixed to $\nu = 8.195$.

We consider $J = 10$ equidistant threshold values. The threshold values τ_1 and τ_{10} are chosen such that they produce loss probabilities approximately equal to 0.1 and 0.01 respectively. The IS parameters in the OASIS algorithm are selected for threshold value $\tau^* = 0.75\tau_1 + 0.25\tau_{10}$ and determined using Algorithm 6.1.

For the OASIS algorithm, we consider four different functions to represent the overall error of the simulation: mean squared error $\omega_{MSE}(\boldsymbol{\pi})$, mean squared relative error $\omega_{MSR}(\boldsymbol{\pi})$, the maximum absolute error $\omega_{MAXE}(\boldsymbol{\pi})$, and the maximum absolute relative error $\omega_{MAXR}(\boldsymbol{\pi})$. We run the OASIS algorithm for minimizing each of these functions, and also for minimizing the variance of each single estimate, with the objective functions $\Sigma_{jj}(\boldsymbol{\pi})$, $j = 1, \ldots, 10$. In those cases, the IS parameters are selected for the respective threshold value τ_j. This procedure allows us to observe the minimal error reachable for each τ_j and how much variance reduction is lost in other estimates. We remind that in all OASIS runs, the first iteration use the same random sequence, which means that the sample allocations of further iterations are based on the same conditional standard deviation estimates for each run. In each iteration of the OASIS algorithm, optimal allocation fractions of the next iteration is determined by using the algorithm in Figure 6.5.

In order to calculate efficiency ratios, we run naive Monte Carlo simulation using common random numbers and find the variance of multiple estimates in a single simulation. We also run the combined method under proportional allocation (PASIS) to

see the efficiency loss when the optimization procedure is removed.

The relative errors (RE) and the efficiency ratios (ER) of all estimates obtained under each of these objective functions are listed in Table 7.5. The execution time (TM) of the algorithms are also given.

Before commenting on Table 7.5, we present a plot of the overall error obtained under some of the objective functions used in the experiment. Figure 7.1a shows the logarithms of percentage relative errors of all tail loss probability estimates obtained with the naive simulation and the OASIS algorithm. The dashed line shows naive simulation (0) results whereas the dotted lines correspond to OASIS minimizing the variances of the first (1), the fifth (2), and the tenth (3) estimates. The remaining plots correspond to OASIS under objective functions (4) $\omega_{MSR}(\boldsymbol{\pi})$, (5) $\omega_{MAXR}(\boldsymbol{\pi})$ and (P) PASIS. In Figure 7.1b, the logarithms of absolute errors of tail loss probability estimates is shown. To reduce the overall relative error in tail loss probability estimates, the objective functions (6) $\omega_{MSE}(\boldsymbol{\pi})$, and (7) $\omega_{MAXE}(\boldsymbol{\pi})$ are considered.

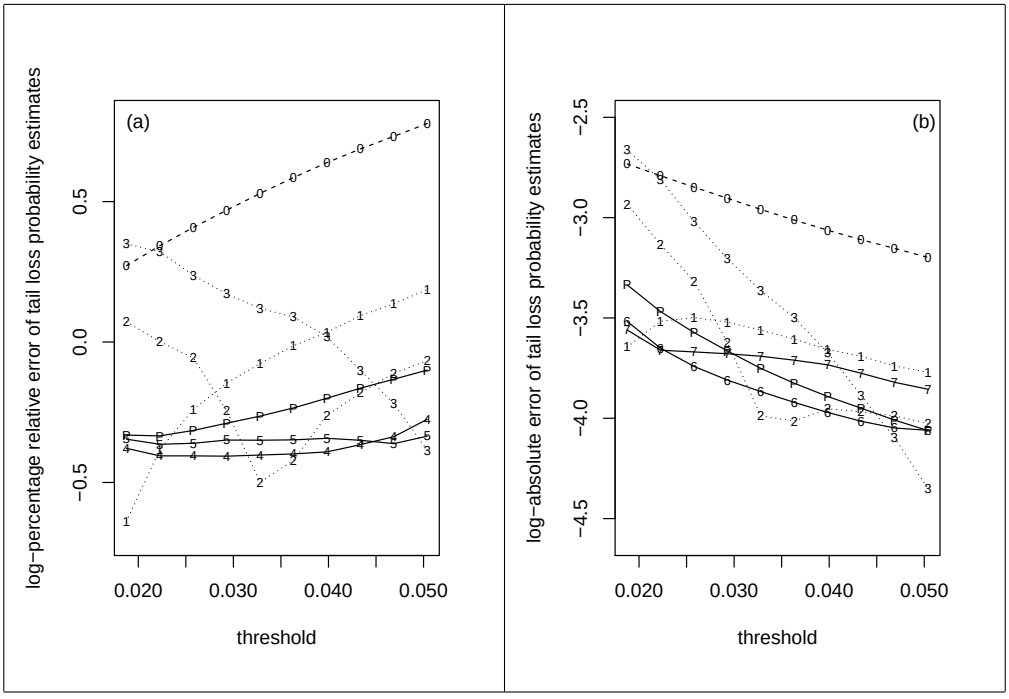

Figure 7.1. The plot diagram of (a) the log-percentage relative errors and (b) the log-absolute errors of tail loss probability estimates under different algorithms.

Table 7.5. Relative errors and the efficiency ratios of multiple tail loss probability estimates of the OASIS algorithm with respect to different objective functions.

		$\boldsymbol{\tau}_j$	$\boldsymbol{\tau}_1$	$\boldsymbol{\tau}_2$	$\boldsymbol{\tau}_3$	$\boldsymbol{\tau}_4$	$\boldsymbol{\tau}_5$	$\boldsymbol{\tau}_6$	$\boldsymbol{\tau}_7$	$\boldsymbol{\tau}_8$	$\boldsymbol{\tau}_9$	$\boldsymbol{\tau}_{10}$
	$\omega(\boldsymbol{\pi})$	$\sim y_j$	0.099	0.074	0.056	0.043	0.033	0.026	0.021	0.017	0.014	0.011
Naive	2.25 sec.	$RE[\hat{y}_j]$	1.87	2.20	2.55	2.93	3.36	3.82	4.31	4.83	5.36	5.93
OASIS	$\min \Sigma_{11}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	0.23	0.41	0.57	0.71	0.83	0.96	1.07	1.22	1.35	1.52
	4.15 sec.	$ER(\hat{y}_j)$	37.2	15.9	11.2	9.7	8.9	8.5	8.7	8.2	8.2	7.9
	$\min \Sigma_{22}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	0.57	0.25	0.40	0.57	0.70	0.80	0.92	1.05	1.17	1.31
	4.54 sec.	$ER(\hat{y}_j)$	5.8	42.7	21.9	14.4	12.4	11.8	11.2	10.8	10.8	10.5
	$\min \Sigma_{33}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	0.86	0.66	0.25	0.40	0.59	0.72	0.84	0.97	1.09	1.11
	4.51 sec.	$ER(\hat{y}_j)$	2.6	6.1	60.9	29.8	18.0	15.4	14.0	13.3	12.7	14.9
	$\min \Sigma_{44}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	1.01	0.89	0.60	0.26	0.39	0.55	0.67	0.76	0.87	0.98
	4.73 sec.	$ER(\hat{y}_j)$	1.8	3.3	10.0	72.4	38.5	25.3	21.2	20.1	18.8	17.9
	$\min \Sigma_{55}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	1.20	1.00	0.88	0.61	0.34	0.40	0.53	0.66	0.77	0.86
	5.35 sec.	$ER(\hat{y}_j)$	1.3	2.7	4.7	12.9	53.1	47.6	34.3	27.3	25.1	24.7
	$\min \Sigma_{66}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	1.33	1.19	1.05	0.90	0.58	0.38	0.45	0.55	0.66	0.75
	4.27 sec.	$ER(\hat{y}_j)$	1.1	1.9	3.4	6.1	19.2	57.4	50.4	40.4	35.1	32.8
	$\min \Sigma_{77}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	2.32	1.41	1.26	1.06	0.82	0.58	0.38	0.47	0.62	0.65
	4.23 sec.	$ER(\hat{y}_j)$	0.4	1.4	2.5	4.5	9.9	25.0	72.3	59.0	42.2	46.8
	$\min \Sigma_{88}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	2.01	1.55	1.40	1.23	1.09	0.89	0.62	0.32	0.37	0.51
	4.24 sec.	$ER(\hat{y}_j)$	0.5	1.2	1.9	3.4	5.7	10.6	27.3	124.9	118.7	75.1
	$\min \Sigma_{99}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	2.09	1.74	1.49	1.34	1.21	1.08	0.79	0.59	0.42	0.42
	4.35 sec.	$ER(\hat{y}_j)$	0.5	0.9	1.7	2.7	4.3	6.8	15.8	34.6	86.6	105.0
	$\min \Sigma_{1010}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	2.78	2.44	1.75	1.49	1.30	1.20	1.05	0.77	0.59	0.40
	4.34 sec.	$ER(\hat{y}_j)$	0.3	0.5	1.2	2.3	3.9	5.7	9.5	22.0	45.2	119.7
	$\min \omega_{MSE}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	0.35	0.32	0.35	0.37	0.41	0.48	0.52	0.59	0.68	0.78
	4.34 sec.	$ER(\hat{y}_j)$	16.6	27.6	31.1	36.7	37.2	35.7	36.6	36.1	33.9	30.8
	$\min \omega_{MSR}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	0.43	0.40	0.39	0.39	0.39	0.40	0.41	0.43	0.46	0.52
	4.35 sec.	$ER(\hat{y}_j)$	10.9	17.6	24.9	32.8	40.9	49.4	58.7	66.7	71.0	69.7
	$\min \omega_{MAXE}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	0.25	0.33	0.42	0.49	0.61	0.75	0.87	1.01	1.15	1.26
	4.95 sec.	$ER(\hat{y}_j)$	29.8	23.4	19.5	19.1	16.0	13.5	12.7	11.5	11.1	11.2
	$\min \omega_{MAXR}(\boldsymbol{\pi})$	$RE[\hat{y}_j]$	0.43	0.42	0.43	0.44	0.44	0.44	0.45	0.44	0.44	0.46
	4.74 sec.	$ER(\hat{y}_j)$	9.9	14.3	18.9	24.1	31.3	39.0	47.8	60.2	74.9	83.9
PASIS	3.89 sec.	$RE[\hat{y}_j]$	0.48	0.48	0.49	0.51	0.55	0.58	0.64	0.69	0.74	0.80
		$ER(\hat{y}_j)$	11.2	15.7	19.6	23.1	26.8	29.5	31.6	33.3	35.8	37.1

An immediate result of Table 7.5 is that the OASIS algorithm successfully reduces the overall error under the objective functions $\omega_{MSE}(\boldsymbol{\pi})$, $\omega_{MSR}(\boldsymbol{\pi})$, $\omega_{MAXE}(\boldsymbol{\pi})$, and $\omega_{MAXR}(\boldsymbol{\pi})$. Focusing on individual estimates may even result in a variance increase

in the other estimates. For instance, a variance minimization for the tenth estimate increases the variance of the first and the second estimates compared to naive simulation. On the other hand, a variance minimization for the first or the fifth estimate reduces the variance of other estimates to a moderate level (see Figure 7.1). If we choose to minimize the mean-squared relative error, we can easily determine optimal allocation fractions using Equation 4.5. However, by using the optimal allocation heuristic in Figure 4.2, we can further decrease the maximum relative error using the allocation heuristic and obtain similar relative errors, as can be seen in the last row of Table 7.5. Since the loss probabilities converge to zero as the threshold increases, it would be a better idea to reduce the relative error rather than the absolute error for all estimates. In fact, this can easily be observed in Figure 7.1b, minimizing the mean-squared or the maximum absolute error does not make much difference compared to the minimization of the variance of the first estimate. On the other hand, the combined algorithm under proportional allocation does not produce estimates as efficient as in the OASIS algorithm.

Under the same experimental design, we run all algorithms for estimating multiple conditional excess values. The optimal IS parameters in OASIS are selected for threshold value $\tau^* = 0.75\tau_1 + 0.25\tau_{10}$ and determined using Algorithm 6.1. For the OASIS algorithm, we use optimal allocations that minimize the overall error of the ratio estimators of conditional excess values.

The variance and the efficiency ratios (ER) of all estimates obtained under each of the objective functions are listed in Table 7.5. The execution time (TM) of the algorithms are also given.

Before commenting on Table 7.6, similar to Figure 7.1, we present a plot of the overall error obtained under some of the objective functions used in the experiment. Figure 7.2a shows the logarithms of percentage relative errors of all conditional excess estimates obtained with the naive simulation and the OASIS algorithm. The dashed line shows naive simulation (0) results whereas the dotted lines correspond to OASIS

Table 7.6. Variances and the efficiency ratios of multiple conditional excess estimates of the OASIS algorithm with respect to different objective functions.

		τ_j	τ_1	τ_2	τ_3	τ_4	τ_5	τ_6	τ_7	τ_8	τ_9	τ_{10}
		$\sim y_{j2}$	0.099	0.074	0.055	0.042	0.033	0.026	0.020	0.016	0.013	0.011
	$\omega(\boldsymbol{\pi})$	$\sim y_j$	0.033	0.037	0.042	0.046	0.051	0.055	0.060	0.064	0.069	0.073
Naive	2.56 sec.	$V[\hat{y}_j]$	3.3E-08	5.0E-08	7.4E-08	1.1E-07	1.6E-07	2.2E-07	3.1E-07	4.2E-07	5.6E-07	7.5E-07
OASIS	$\min \Sigma_{11}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	4.7E-10	9.2E-10	1.5E-09	2.2E-09	3.3E-09	4.6E-09	6.3E-09	8.4E-09	1.1E-08	1.4E-08
	6.52 sec.	$ER(\hat{y}_j)$	27.5	21.4	20.1	19.1	18.6	19.3	19.6	19.8	20.1	20.7
	$\min \Sigma_{22}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	2.0E-09	6.5E-10	1.1E-09	1.7E-09	2.4E-09	3.5E-09	5.1E-09	7.3E-09	8.7E-09	1.2E-08
	6.54 sec.	$ER(\hat{y}_j)$	6.6	30.3	27.6	24.7	25.9	25.0	23.9	22.9	25.2	25.1
	$\min \Sigma_{33}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	3.4E-09	2.1E-09	8.2E-10	1.3E-09	2.0E-09	2.7E-09	3.8E-09	5.2E-09	7.1E-09	9.4E-09
	6.48 sec.	$ER(\hat{y}_j)$	3.9	9.3	35.9	33.2	30.7	32.7	32.6	32.3	31.5	31.6
	$\min \Sigma_{44}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	5.4E-09	4.2E-09	2.5E-09	1.2E-09	1.5E-09	2.4E-09	2.9E-09	4.1E-09	5.7E-09	7.3E-09
	6.59 sec.	$ER(\hat{y}_j)$	2.4	4.7	11.8	34.6	41.4	37.0	41.4	39.9	38.5	40.0
	$\min \Sigma_{55}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	6.9E-09	7.2E-09	5.0E-09	3.3E-09	1.2E-09	1.6E-09	2.3E-09	3.3E-09	4.5E-09	6.0E-09
	7.77 sec.	$ER(\hat{y}_j)$	1.6	2.3	4.9	11.0	43.5	46.4	44.0	41.9	41.1	41.4
	$\min \Sigma_{66}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	2.4E-08	1.0E-08	6.6E-09	5.7E-09	3.3E-09	1.5E-09	1.8E-09	2.6E-09	3.7E-09	4.9E-09
	6.38 sec.	$ER(\hat{y}_j)$	0.5	1.9	4.5	7.6	19.4	58.2	69.3	64.5	61.3	61.6
	$\min \Sigma_{77}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.6E-08	1.2E-08	1.3E-08	7.8E-09	6.8E-09	4.4E-09	1.8E-09	2.0E-09	2.9E-09	4.0E-09
	6.35 sec.	$ER(\hat{y}_j)$	0.8	1.7	2.2	5.6	9.4	20.7	70.7	84.5	77.3	76.1
	$\min \Sigma_{88}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.5E-08	1.8E-08	1.7E-08	9.4E-09	8.9E-09	7.4E-09	4.8E-09	2.2E-09	2.8E-09	3.3E-09
	6.82 sec.	$ER(\hat{y}_j)$	0.8	1.0	1.6	4.3	6.7	11.3	24.4	71.3	74.5	86.2
	$\min \Sigma_{99}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	2.2E-08	3.7E-08	5.2E-08	1.4E-08	1.2E-08	1.1E-08	8.6E-09	6.1E-09	2.1E-09	2.8E-09
	6.80 sec.	$ER(\hat{y}_j)$	0.6	0.5	0.5	3.0	4.9	7.7	13.7	26.3	98.9	99.6
	$\min \Sigma_{10\,10}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.9E-08	2.5E-08	2.5E-08	2.6E-08	1.3E-08	1.2E-08	1.1E-08	8.5E-09	5.6E-09	2.6E-09
	6.44 sec.	$ER(\hat{y}_j)$	0.7	0.8	1.2	1.7	4.9	7.4	10.9	19.9	39.8	115.7
	$\min \omega_{MSE}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.4E-09	1.4E-09	1.4E-09	1.6E-09	1.8E-09	2.1E-09	2.5E-09	3.1E-09	4.0E-09	5.3E-09
	6.49 sec.	$ER(\hat{y}_j)$	9.4	14.3	21.1	26.9	35.0	42.4	49.3	54.0	55.7	55.9
	$\min \omega_{MSR}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.1E-09	1.1E-09	1.2E-09	1.5E-09	1.9E-09	2.4E-09	2.9E-09	3.7E-09	4.8E-09	6.4E-09
	6.51 sec.	$ER(\hat{y}_j)$	11.6	18.6	23.8	28.4	33.1	37.4	42.5	44.9	46.2	46.6
	$\min \omega_{MAXE}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	2.7E-09	2.8E-09	2.9E-09	2.9E-09	3.0E-09	3.2E-09	3.3E-09	3.4E-09	3.5E-09	4.0E-09
	6.77 sec.	$ER(\hat{y}_j)$	4.6	6.8	9.9	14.1	19.8	26.3	36.0	47.9	61.3	71.5
	$\min \omega_{MAXR}(\boldsymbol{\pi})$	$V[\hat{y}_j]$	1.0E-09	1.1E-09	1.5E-09	1.8E-09	2.2E-09	2.6E-09	3.0E-09	3.5E-09	4.0E-09	4.8E-09
	7.10 sec.	$ER(\hat{y}_j)$	11.6	15.8	18.3	22.4	26.3	31.6	37.9	43.8	50.8	56.5
PASIS	4.53 sec.	$V[\hat{y}_j]$	1.3E-09	1.5E-09	1.9E-09	2.6E-09	3.6E-09	4.8E-09	6.4E-09	8.6E-09	1.1E-08	1.5E-08
		$ER(\hat{y}_j)$	15.6	20.7	24.1	25.9	27.9	29.7	30.7	31.3	30.9	31.4

minimizing the variances of the first (1), the fifth (2), and the tenth (3) estimates. The remaining plots correspond to OASIS under objective functions (4) $\omega_{MSR}(\boldsymbol{\pi})$, (5) $\omega_{MAXR}(\boldsymbol{\pi})$, and (P) PASIS. In Figure 7.2b, the logarithms of absolute errors of conditional excess estimates is shown. To reduce the overall relative error in conditional excess estimates, the objective functions (6) $\omega_{MSE}(\boldsymbol{\pi})$, and (7) $\omega_{MAXE}(\boldsymbol{\pi})$ are considered.

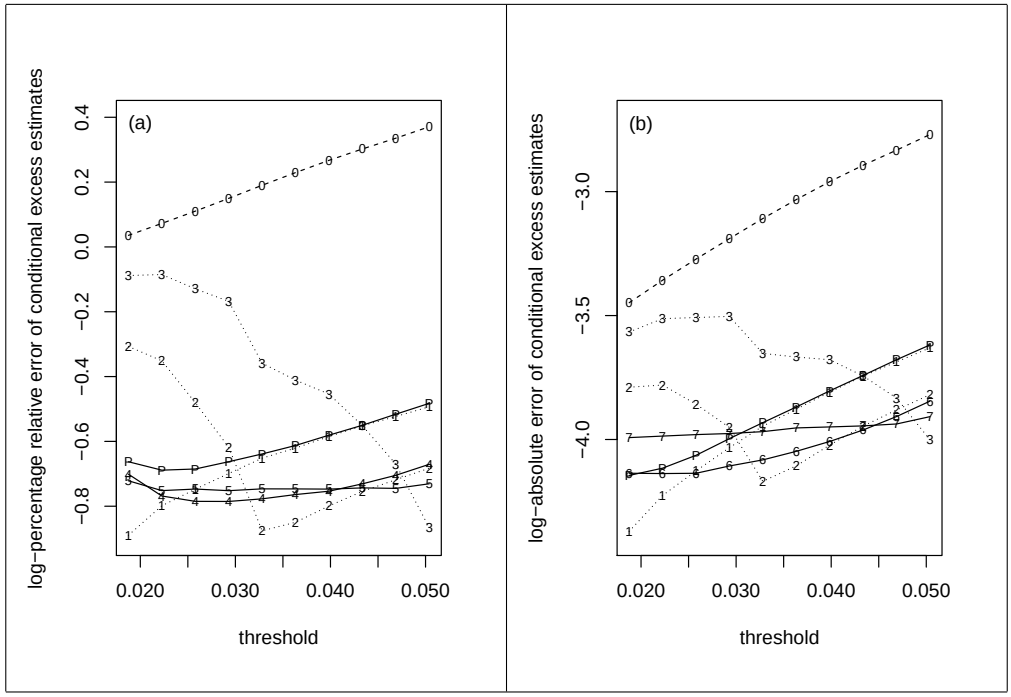

Figure 7.2. The plot diagram of (a) the log-percentage relative errors and (b) the log-absolute errors of conditional excess estimates under different algorithms.

The results in Table 7.6 are similar to the ones in Table 7.5. The OASIS algorithm successfully reduces the overall error under the objective functions $\omega_{MSE}(\boldsymbol{\pi})$, $\omega_{MSR}(\boldsymbol{\pi})$, $\omega_{MAXE}(\boldsymbol{\pi})$, and $\omega_{MAXR}(\boldsymbol{\pi})$. However, this time, focusing on individual estimates does not result in a variance increase in the other estimates. A variance minimization for any of the estimates reduces the variance of other estimates to a moderate level (see Figure 7.2). If we choose to minimize the mean-squared error, we can easily determine optimal allocation fractions using Equation 4.5. However, by using the optimal allocation heuristic in Figure 4.2, we can further decrease the maximum error using the allocation heuristic and obtain similar absolute errors, as can be seen in Figure 7.2b. The conditional excess values show an increasing trend as the threshold increases, thus, reducing the relative error rather than the absolute error for all estimates does not have a significant difference on the efficiency of the estimates. In fact, this can easily be observed in Figure 7.2a, minimizing the mean-squared or the maximum relative error does not make much difference compared to minimizing the mean-squared or the maximum absolute error. Again, the combined algorithm under proportional allocation does not produce estimates as efficient as in the OASIS algorithm.

7.4. Performance of the Optimal Allocation Heuristic

We make our third set of experiments to measure the performance of the optimal allocation heuristic. We have generated ten realistic random instances of the optimization problem in Equation 4.8 regarding to the simulation of the stock portfolios described in this section. We consider $D = 5$ stocks under the t-copula model with both t and the generalized hyperbolic marginals. The number of strata is equal to 484 and we have 10 different threshold values. For each model, we have generated four instances with an approximate sample size $N = 10^4$ and a fifth instance with an approximate sample size $N = 2 \times 10^6$, which yields more precise variance estimates. For each instance, we have used KNITRO solver to find the optimum objective value ω^* of the model in Equation 4.8 and the optimum objective value ω^{**} of the model in Equation 4.9. We also used the optimal allocation heuristic in Figure 4.2 to find the sub-optimal objective value ω^h. Table 7.7 shows these objective values with the sub-optimality results compared to ω^*.

Table 7.7. Optimal and suboptimal objective values, ω^* and ω^{**}, and the heuristic solution ω^h for five random instances of the model in Equation 4.8.

	t					GH				
	ω^*	ω^{**}		ω^h		ω^*	ω^{**}		ω^h	
Ins.	Value	Value	Subopt.	Value	Subopt.	Value	Value	Subopt.	Value	Subopt.
1	0.918	0.939	2.2%	0.944	2.8%	1.091	1.109	1.7%	1.112	1.9%
2	0.971	0.991	2.0%	0.997	2.6%	1.090	1.109	1.7%	1.122	2.9%
3	0.933	0.953	2.1%	0.962	3.0%	1.158	1.176	1.6%	1.180	1.9%
4	0.988	1.008	2.0%	1.015	2.7%	1.196	1.213	1.4%	1.215	1.6%
5	1.126	1.144	1.6%	1.147	1.9%	1.205	1.222	1.4%	1.236	2.6%

In Table 7.7, we can see that the objective values obtained by the optimal allocation heuristic, ω^h, are very close to the optimal objective values ω^*. Moreover, reducing the dimension of the problem as in Equation 4.9 is a good idea since the ω^{**} values are also very close to the ω^* values.

The optimal solutions obtained with the larger sample size (instance 5) are better estimators of the real optimal allocation fractions. In Table 7.7, we also observe that the optimal objective values obtained by using variance estimates with small sample size (rows 1 to 4) are clearly smaller than the objective values obtained with large sample size (bottom row). We can see that the sub-optimality results are insignificant compared to this deviation and therefore conclude that it is not necessary to solve the optimization problem in the OASIS algorithm more precise than the heuristic.

8. CONCLUSIONS

In the scope of this thesis, stratified sampling and its combined implementation with importance sampling is studied. As the application area, the risk estimation of linear asset portfolios is selected. In the literature, there are rare comments on the combination of these methods for financial applications, such as option pricing and portfolio risk estimation. In none of these studies, an optimal implementation of the combined methods for multiresponse simulation is considered.

In this thesis, a multiresponse simulation algorithm, OASIS, is developed as a combination of optimal allocation stratification and importance sampling. The algorithm is used for the estimation of multiple values in a single simulation, such as multiple tail loss probabilities or conditional excess values in linear asset portfolios. For the realistic t-copula portfolio model, the implementation of OASIS is explained in detail.

For practically relevant examples, the OASIS algorithm increased the efficiency of tail loss probability and conditional excess estimates under the t-copula model compared to the naive simulation and other benchmark methods in the literature. The variance of the estimates are substantially reduced without a significant increase in the execution time. Therefore, the OASIS method can provide highly efficient tail loss probability or conditional excess estimates for realistic stock portfolio models.

Moreover, OASIS enables the efficient estimation of tail loss probabilities or conditional excess values for multiple loss thresholds in a single simulation. The IS part of the method provides an initial variance reduction nearly in all estimates and the stratification methodology allows the minimization of the overall error of the simulation by considering allocation fractions as decision variables. In order to increase the efficiency of all estimators, two general objective functions are proposed to represent the overall error. In the first, we consider the minimization of the linear functions of the

variance-covariance matrix of the stratified estimates. Whereas, in the second class, we consider the minimization of the maximum of the variances which are weighted with non-negative coefficients. For these objective functions, nonlinear optimization models are introduced with allocation fractions as decision variables. A closed-form solution is developed for the first class of objective functions. For the second class, an optimal allocation heuristic is utilized to develop a sub-optimal solution. These solutions are used in the sampling phase to minimize quantities that represent the overall error of the simulation, such as the mean-squared (relative) error or the maximum absolute (relative) error.

With multiple tail loss probability estimates in a single simulation, one can use spline interpolation to approximate the CDF of the loss distribution and root finding algorithms to obtain VaR values for arbitrary probabilities. Along with multiple tail loss probability estimates, existence of multiple conditional excess values enable the calculation of expected shortfall values in a similar fashion.

With the numerical examples, the OASIS algorithm is shown to be an efficient and flexible method for simulation problems for which we can find efficient stratification functions. The idea of the OASIS algorithm can be used to minimize the overall error of an arbitrary simulation associated with multiple estimates. We remind that in our examples, the size of the random input is independent of the parameter space. Whether comparable results would be obtainable for all type of discrete event simulations is an area in need of further research. Nevertheless, OASIS can be promisingly useful for stochastic optimization and response surface estimation problems, as they require the evaluation of the simulation function for many different parameter values.

APPENDIX A: A COUNTER EXAMPLE

Consider the function $q(x) = \mathbb{1}_{\{e^x > 3.6\}}$. Then $E\left[q\left(Z\right)\right]$ is simply equal to $\Pr\{Z > \ln 3.6\} \approx 0.1$, and the optimal IS density is $f_{IS}^{*}\left(x\right) \approx 10\phi\left(x\right)\mathbb{1}_{\{x > \ln 3.6\}}$. The optimal allocation rule in Equation 2.10 will asymptotically force the sample to be allocated in the stratum that contains the root of $\ln 3.6$. The strata, for which the conditional variances are estimated as zero in preliminary iterations, will no longer be subject to significant allocations in further iterations, even if the conditional expectations are different than zero. Thus, the density f_{OAS} will not be approaching to the optimal IS density.

However, in the family of densities given in Equation 5.1, there exists a density which approaches to the optimal IS density as $\|\boldsymbol{p}\|_{\infty}$ goes to zero. In order to define that density, we divide the strata indices into two sets. In the first set P_0, we collect the indices of strata that has zero conditional variance and non-zero conditional expectation. The remaining indices belong to the other set, P_0^c. We define:

$$\pi_i^{IS} = \begin{cases} \frac{p_i\sigma_i}{\sum_{l\in P_0} p_l\sigma_l}, & i \in P_0 \\ 1, & i \in P_0^c \end{cases} \tag{A.1}$$

In the family of densities given in Equation 5.1, the density that is constructed with $\pi_i = \pi_i^{IS}$ given in Equation A.1 and it approaches to f_{IS}^{*} as $\|\boldsymbol{p}\|_{\infty}$ goes to zero.

APPENDIX B: OPTIMAL IS PARAMETERS FOR THE t-COPULA MODEL

We give the pseudo-code of the algorithm of Sak *et al.* (2010) which returns the optimal IS parameters for the t-copula model. For more details, we direct the reader to Sak *et al.* (2010).

Require: Vector of investment shares in each stock $\boldsymbol{w} = (w_1, \ldots, w_D)$; vector of scaling factors $\boldsymbol{c} = (c_1, \ldots, c_D)$ of the logreturns $d = 1, \ldots, D$; Cholesky factor $\boldsymbol{L}$ of correlation matrix Λ, threshold τ; parameters of the t-copula model

Ensure: Optimal mean shift $\boldsymbol{\mu}$ and scale parameter γ

1: compute direction $\boldsymbol{v}_{init} = L'(\boldsymbol{c} * \boldsymbol{w})$ ("$*$" denotes a component-wise product)

2: call constrained BFGS algorithm with initial direction $\boldsymbol{v}_{init}$, objective function as given in the algorithm in Figure B.2, and non-negativity constraints for all components of $\boldsymbol{v}$. Get optimal direction $\boldsymbol{v}^*$.

3: call algorithm in Figure B.2 direction $\boldsymbol{v}^*$ and get the optimal mean shift $\boldsymbol{\mu}^*$ and optimal scale parameter γ^*.

Figure B.1. Computation of optimal IS parameters for the t-copula model.

Require: Direction $\boldsymbol{v}$

Ensure: the mean shift $\boldsymbol{\mu}$ and scale parameter γ

1: Compute r_0 by solving $Return\,(r_0\boldsymbol{v}/\|\boldsymbol{v}\|) - \tau = 0$ numerically

2: Return objective function value $(\nu/2 - 1)\left(\ln\left((\nu-2)\,(1 + r_0^2\nu^{-1})^{-1}\right) - 1\right)$, the scale parameter $\gamma = 2(1 + r_0^2\nu^{-1})^{-1}$, and the mean shift $\boldsymbol{\mu} = r_0\left((\nu-2)\,(\nu + r_0^2)^{-1}\right)^{1/2}\boldsymbol{v}/\|\boldsymbol{v}\|$

Figure B.2. Computation of IS parameters for direction $\boldsymbol{v}$.

REFERENCES

Aas, K., and I. H. Haff, "The Generalized Hyperbolic Skew Student's t-distribution", *Journal of Financial Econometrics* Vol. 4, No. 2, pp. 275-309.

Artzner, P., F. Delbaen, J.-M. Eber, and D. Heath, 1999, "Coherent Measures of Risk", *Mathematical Finance*, Vol. 9, No. 3, pp. 203-228.

Bassamboo, A, S. Juneja, A. Zeevi, 2008, "Portfolio Credit Risk with Extremal Dependence: Asymptotic Analysis and Efficient Simulation", *Operations Research*, Vol. 56, No. 3, pp. 593-606.

Başoğlu, İ., W. Hörmann, and H. Sak, 2013, "Optimally Stratified Importance Sampling for Portfolio Risk with Multiple Thresholds", *Optimization*, Vol. 62, No. 11, pp. 1451-1471.

Başoğlu, İ., and W. Hörmann, 2014, "Efficient Stratified Sampling Implementations in Multiresponse Simulation", To be appeared in: *Proceedings of the 2014 Winter Simulation Conference*, Savannah, Georgia.

Behr, A., and U. Pötter, "Alternatives to the Normal Model of Stock Returns: Gaussian Mixture, Generalised Log-f and Generalised Hyperbolic Models", *Annals of Finance* Vol. 5, No. 1, pp. 49-68.

Boyle, P., 1977, "Options: A Monte Carlo Approach", *Journal of Financial Economics*, Vol. 4, No. 3, pp. 323-338.

Boyle, P., M. Broadie, and P. Glasserman, 1997, "Monte Carlo Methods for Security Pricing", *Journal of Economic Dynamics and Control*, Vol. 21, No. 8-9, pp. 1267-1321.

Box, G., and K. Wilson, 1951, "On the Experimental Attainment of Optimum Conditions", *Journal of the Royal Statistical Society Series*, Vol. B13, No. 1, pp. 138.

Byrd, R. H., P. Lu, J. Nocedal, and C. Zhu, 1995, "A Limited Memory Algorithm for Bound Constrained Optimization", *SIAM Journal on Scientific Computing*, Vol. 16, No. 5, pp. 1190-1208.

Byrd, R. H., J. Nocedal, and R. A. Waltz, 2006, "Knitro: An Integrated Package for Nonlinear Optimization", In *Large-Scale Nonlinear Optimization*, Edited by G. Pillo and M. Roma, *Nonconvex Optimization and Its Applications*, Vol. 83, pp. 3559, Springer US.

Cochran, W. G., 1977, *Sampling Techniques*, Second Edition, John Wiley & Sons, New York.

Derflinger, G., W. Hörmann, and J. Leydold, 2010, "Random Variate Generation by Numerical Inversion When Only the Density Function is Known", *ACM Transactions on Modeling and Computer Simulation*, Vol. 20, No. 18, pp. 1-25.

Díaz-García, J. A., L. U. Cortez, 2006, "Optimum Allocation in Multivariate Stratified Sampling: Multi-objective Programming", *Comunicación Técnica*, No. I-06-07/28-03-206, Guanajuato, México.

Díaz-García, J. A., L. U. Cortez, 2008, "Multi-objective Optimisation for Optimum Allocation in Multivariate Stratified Sampling", *Survey Methodology*, Vol. 34, No. 2, pp. 215-222.

Durbin, J., and S. J. Koopman, 2012, *Time Series Analysis by State Space Methods*, Second Edition, Oxford Statistical Science Series.

Embrechts, P., A. McNeil, D. Straumann, 2002, "Correlation and Dependence in Risk Management: Properties and Pitfalls." *Risk Management: Value at Risk and Beyond*, pp. 176-223.

Étoré, P., and B. Jourdain, 2010, "Adaptive Optimal Allocation in Stratified Sampling Methods", *Methodology and Computing in Applied Probability*, Vol. 12, No. 3, pp. 335-360.

Fishman, G., 1996, *Monte Carlo: Concepts, Algorithms, and Applications*, Springer-Verlag, New York.

Glasserman, P., P. Heidelberger, and P. Shahabuddin, "Asymptotically Optimal Importance Sampling and Stratification for Pricing Path Dependent Options", *Mathematical Finance*, Vol. 9, No. 2, pp. 117-152.

Glasserman, P., P. Heidelberger, and P. Shahabuddin, "Variance Reduction Techniques for Estimating Value-at-Risk", *Management Science*, Vol. 46, No. 10, pp. 1349-1364.

Glasserman, P., P. Heidelberger, and P. Shahabuddin, "Portfolio Value-at-risk with Heavy-tailed Risk Factors", *Mathematical Finance*, Vol. 12, No. 3, pp. 239-269.

Glasserman, P., 2004, *Monte Carlo Methods in Financial Engineering*, Springer-Verlag, New York.

Glasserman, P., and J. Li, 2005, "Importance Sampling for Portfolio Credit Risk", *Management Science*, Vol. 51, No. 11, pp. 16431656.

Halulu, S., 2012, *Quantifying the Risk of Portfolios Containing Stocks and Commodities*, Master's thesis, Boğaziçi University, İstanbul.

Hammersley, J. M., and D. C. Handscomb, 1964, *Monte Carlo Methods*, Methuen, London.

Hammersley, J. M., and K. W. Morton, 1956, "A New Monte Carlo Technique: Antithetic Variates", *Mathematical Proceedings of the Cambridge Philosophical Society*, Vol. 52, No. 3, pp. 449-475.

Hörmann, W., and H. Sak, 2010, "t-copula Generation for Control Variates", *Mathematics and Computers in Simulation*, 2010, Vol. 81, No. 4, pp. 782-790.

Imai, J., and K. S. tan, 2006, "A General Dimension Reduction Technique for Derivative Pricing", *Journal of Computational Finance*, Vol. 10, No. 2, pp. 129-155.

Imai, J., and K. S. tan, 2009, "An Accelerating Quasi-Monte Carlo Method for Option Pricing under the Generalized Hyperbolic Lévy Processes", *SIAM Journal on Scientific Computing*, Vol. 31, No. 3, pp. 2282-2302.

Jensen, J. L. W. V., 1906, "Sur les Fonctions Convexes et les Ingalits Entre les Valeurs Moyennes", *Acta Mathematica*, Vol. 30, No. 1, pp. 175-193.

Jourdain, B., 2009, "Adaptive Variance Reduction Techniques in Finance", *Radon Series on Computational and Applied Mathematics*, Vol. 8, pp. 1-18.

Jourdain, B., B. Lapeyre, and P. Sabino, 2011, "Convenient Multiple Directions of Stratification", *International Journal of Theoretical and Applied Finance* , Vol. 14, No. 6, pp. 867-897.

Kahn, H., and A. W. Marshall, 1953, "Methods of Reducing Sample Size in Monte Carlo Computations", *Journal of the Operations Research Society of America*, Vol. 1, No. 5, pp. 263-278.

Kang, W. and P. Shahabuddin, 2005, "Fast Simulation for Multifactor Portfolio Credit Risk in the t-copula Model", In: *Proceedings of the 2005 Winter Simulation Conference*, Edited by M. E. Kuhl, N. M. Steiger, F. B. Armstrong, an J. A. Joines, Orlando, Florida.

Khowaja, S., S. Ghufran, and M. J. Ahsan, 2012, "Multi-objective Optimization for Optimum Allocation in Multivariate Stratified Sampling with Quadratic Cost", *Journal of Statistical Computation and Simulation*, Vol. 82, No. 12, pp. 1789-1798.

Kole, E., K. Koedijk, and M. Verbeek, 2007, "Selecting Copulas for Risk Management", *Journal of Banking and Finance*, Vol. 31, No. 8, pp. 24052423.

Koopman, S. J., N. Shephard, and D. Creal, 2009, "Testing the Assumptions Behind Importance Sampling", *Journal of Econometrics*, Vol. 149, No. 1, pp. 2-11.

Kreinin, A., L. Merkoulovitch, D. Rosen, and M. Zerbs, 1998, "Principal Component Analysis in Quasi Monte Carlo Simulation", *Algo Research Quarterly*, Vol. 1, No.2, pp. 21-30.

Law, A. M., 2014, *Simulation Modeling & Analysis*, McGraw-Hill, New York.

Lemieux, C., 2009, *Monte Carlo and Quasi-Monte Carlo Sampling*, Springer Series in Statistics, New York.

Marshall, A., 1956, "The use of multi-stage sampling schemes in Monte Carlo computations", In: H. Meyer (Ed.) *Symposium on Monte Carlo Methods*, pp. 123-140, John Wiley & Sons, New York.

Mashal, R., M. Naldi, and A. Zeevi, 2003, "Comparing the Dependence Structure of Equity and Asset Returns", *Risk* No. 16, pp. 8287.

McKay, M. D., W. J. Conover, and R. J. Beckman, 1979, "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output from a Computer Code", *Technometrics*, Vol. 21, No. 2, pp. 239-245.

McLeish, D. L., 2010, "Bounded Relative Error Importance Sampling and Rare Event Simulation", *ASTIN Bulletin*, Vol. 40, No. 01, pp. 377-398.

Miller, T. J., J. R. Skalski, and J. N. Ianelli, 2007, "Optimizing a Stratifield Sampling Design When Faced with Multiple Objectives", *ICES Journal of Marine Science*, Vol. 64, No. 1, pp. 97109.

Myers, R., and D. Montgomery, 2002, *Response Surface Methodology*, Wiley, New York.

Rubinstein, R. Y., and D. P. Kroese, 2008, *Simulation and the Monte Carlo Method*, John Wiley & Sons, New Jersey.

Rubinstein, R. Y., and R. Marcus, 1985, "Efficiency of Multivariate Control Variates in Monte Carlo Simulation", *Operations Research* Vol. 33, No. 3, pp. 661677.

Rubinstein, R. Y., and B. Melamed, 1998, *Modern Simulation and Modeling*, Wiley.

Sak, H., W. Hörmann, and J. Leydold, 2010, "Efficient Risk Simulations for Linear Asset Portfolios in the t-copula model", *European Journal of Operational Research*, Vol. 202, No. 3, pp. 802-809.

Sak, H., and W. Hörmann, 2012, "Fast Simulations in Credit Risk", *Quantitative Finance*, Vol. 12, No. 10, pp. 15571569.

Trotter, H. F. , and J. W. Tukey, 1956, "Conditional Monte Carlo for Normal Samples", In: H. Meyer (Ed.) *Symposium on Monte Carlo Methods*, pp. 80-88, John Wiley & Sons, New York.