New variance reduction methods in Monte Carlo rare event simulation

Leslie Murray

PEDECIBA INFORMÁTICA
INSTITUTO DE COMPUTACIÓN, FACULTAD DE INGENIERÍA
UNIVERSIDAD DE LA REPÚBLICA
MONTEVIDEO, URUGUAY

TESIS DE DOCTORADO EN INFORMÁTICA

New variance reduction methods in Monte Carlo rare event simulation

Leslie Murray

lmurray@fing.edu.uy

Agosto de 2014

Supervisor: Héctor Cancela

Orientadores de tesis: Héctor Cancela y Gerardo Rubino

Tribunal: Nicolás Stier y Gustavo Guerberoff (revisores),

Ernesto Mordecki, Antonio Mauttone y Alvaro Martín

I dedicate this work to R.M., one of the persons I loved most in life.

Acknowledgements

Works like this thesis are possible because of people like Héctor Cancela and Gerardo Rubino, always ready to support and to help, far beyond their formal duties. Because of these two persons, writing this thesis has been, above all, a great pleasure to me. I have also learned from them that everyday work can be lived with pleasure. Professional honesty, ethics and passion for the work, are also things that I learned from Héctor Cancela and Gerardo Rubino. For all this, my sincere gratitude for them both.

I also want to say that this work would not have been possible without the valuable contribution of people like the jury members, who agree to participate, offering their knowledge, ability, and above all, their time. My sincere thanks then to Nicolás Stier, Gustavo Guerberoff, Ernesto Mordecki, Alvaro Martín and Antonio Mautone.

A special thanks to the Postgraduate Committee and to María Inés Sánchez Grezzi, secretary of the PEDECIBA, who were commissioned, for a long time, of all the documentation and formalities related to my doctorate.

And I would also like to deeply thank all those with whom I have shared my visits to the Facultad de Ingeniería UDELAR. In the offices, in lunches, sharing coffees or just talking anywhere, a lot of people made me feel at home. Many of them have become true friends, surely, for the rest of my life. They are the visible face of a country that I love very much. I look forward to sharing with them lots of visits in the future.

Resumen

Para sistemas que proveen algún tipo de servicio mientras están operativos y dejan de proveerlo cuando fallan, es de interés determinar parámetros como, por ejemplo, la probabilidad de encontrar el sistema en falla en un instante cualquiera, el tiempo medio transcurrido entre fallas, o cualquier medida capaz de reflejar la capacidad del sistema para proveer servicio. Las determinaciones de estas medidas de seguridad de funcionamiento se ven afectadas por diversos factores, entre ellos, el tamaño del sistema y la rareza de las fallas. En esta tesis se estudian algunos métodos concebidos para determinar estas medidas sobre sistemas grandes y altamente confiables, es decir sistemas formados por gran cantidad de componentes, en los que las fallas del sistema son eventos raros.

Ya sea en forma directa o indirecta, parte de las las expresiones que permiten determinar las medidas de interés corresponden a la probabilidad de que el sistema se encuentre en algún estado de falla. De un modo u otro, estas expresiones evaluan la fracción —ponderada por la distribución de probabilidad de las configuraciones del sistema— entre el número de configuraciones en las que el sistema falla y la totalidad de las configuraciones posibles. Si el sistema es grande el cálculo exacto de estas probabilidades, y consecuentemente de las medidas de interés, puede resultar inviable. Una solución alternativa es estimar estas probabilidades mediante simulación. Uno de los mecanismos para hacer estas estimaciones es la simulación de tipo Monte Carlo, cuya versión más simple es la simulación en crudo o estándar. El problema es que si las fallas son raras, el número de iteraciones necesario para estimar estas probabilidades mediante simulación estándar con una precisión aceptable, puede resultar desmesuradamente grande.

En esta tesis se analizan algunos métodos existentes para mejorar la simulación estándar en el contexto de eventos raros, se hacen análisis de varianza y se prueban los métodos sobre una variedad de modelos. En todos los casos la mejora se consigue a costa de una reducción de la varianza del estimador con respecto a la varianza del estimador estándar. Gracias a la reducción de varianza es posible estimar la probabilidad de ocurrencia de eventos raros con una precisión aceptable, a partir de un número razonable de iteraciones. Como parte central del trabajo se proponen dos métodos nuevos, uno relacionado con Spliting y otro relacionado con Monte Carlo Condicional.

Splitting es un método de probada eficiencia en entornos en los que se busca evaluar desempeño y confiabilidad combinados, escasamente utilizado en la simulación de sistemas altamente confiables sobre modelos estáticos (sin evolución temporal). En

su formulación básica Splitting hace un seguimiento de las trayectorias de un proceso estocástico a través de su espacio de estados y multiplica su número ante cada cruce de umbral, para un conjunto dado de umbrales distribuidos entre los estados inicial y final. Una de las propuestas de esta tesis es una adaptación de Splitting a un modelo estático de confiabilidad de redes. En el método propuesto se construye un proceso estocástico a partir de un tiempo ficticio en el cual los enlaces van cambiando de estado y se aplica Splitting sobre ese proceso. El método exhibe elevados niveles de precisión y robustez.

Monte Carlo Condicional es un método clásico de reducción de varianza cuyo uso no está muy extendido en el contexto de eventos raros. En su formulación básica Monte Carlo Condicional evalúa las probabilidades de los eventos de interés, condicionando las variables indicatrices a eventos no raros y simples de detectar. El problema es que parte de esa evaluación incluye el cálculo exacto de algunas probabilidades del modelo. Uno de los métodos propuestos en esta tesis es una adaptación de Monte Carlo Condicional al análisis de modelos Markovianos de sistemas altamente confiables. La propuesta consiste en estimar las probabilidades cuyo valor exacto se necesita, mediante una aplicación recursiva de Monte Carlo Condicional. Se estudian algunas características de este modelo y se verifica su eficiencia en forma experimental.

Abstract

For systems that provide some kind of service while they are operational and stop providing it when they fail, it is of interest to determine parameters such as, for example, the probability of finding the system failed at any moment, the mean time between failures, or any measure that reflects the capacity of the system to provide service. The determination of these measures —known as dependability measures— is affected by a variety of factors, including the size of the system and the rarity of failures. This thesis studies some methods designed to determine these measures on large and highly reliable systems, i.e. systems formed by a large number of components, such that systems' failures are rare events.

Either directly or indirectly, part of the expressions for determining the measures of interest correspond to the probability that the system is in some state of failure. Somehow, this expressions evaluate the ratio —weighted by the probability distribution of the systems' configurations— between the number of configurations in which the system fails and all possible configurations. If the system is large, the exact calculation of these probabilities, and consequently of the measures of interest, may be unfeasible. An alternative solution is to estimate these probabilities by simulation. One mechanism to make such estimation is Monte Carlo simulation, whose simplest version is crude or standard simulation. The problem is that if failures are rare, the number of iterations required to estimate this probabilities by standard simulation, with acceptable accuracy, may be extremely large.

In this thesis some existing methods to improve the standard simulation in the context of rare events are analyzed, some variance analyses are made and the methods are tested empirically over a variety of models. In all cases the improvement is achieved at the expense of reducing the variance of the estimator with respect to the standard estimator's variance. Due to this variance reduction, the probability of the occurrence of rare events, with acceptable accuracy, can be achieved in a reasonable number of iterations. As a central part of this work, two new methods are proposed, one of them related to Splitting and the other one related to Conditional Monte Carlo.

Splitting is a widely used method in performance and performability analysis, but scarcely applied for simulating highly reliable systems over static models (models with no temporal evolution). In its basic formulation Splitting keeps track of the trajectories of a stochastic process through its state space and it splits or multiplies the number of them at each threshold cross, for a given set of thresholds distributed between the initial and the final state. One of the proposals of this thesis is an adaptation of Splitting to a static network reliability model. In the proposed method, a fictitious

time stochastic process in which the network links keep changing their state is built, and Splitting is applied to this process. The method shows to be highly accurate and robust.

Conditional Monte Carlo is a classical variance reduction technique, whose use is not widespread in the field of rare events. In its basic formulation Conditional Monte Carlo evaluates the probabilities of the events of interest, conditioning the indicator variables to not rare and easy to detect events. The problem is that part of this assessment includes the exact calculation of some probabilities in the model. One of the methods proposed in this thesis is an adaptation of Conditional Monte Carlo to the analysis of highly reliable Markovian systems. The proposal consists in estimating the probabilities whose exact value is needed, by means of a recursive application of Conditional Monte Carlo. Some features of this model are discussed and its efficiency is verified experimentally.

Contents

List of Figures

List of Tables

CHAPTER 1

Introduction

Contents

Abstract

This chapter describes the basic ideas that support this thesis. The concepts of *rare events* and *dependability* are introduced by means of two very simple models. The publications of the author directly related with this thesis are commented and, in the last section, the structure of the thesis, with a brief comment about every chapter, is described.

1.1 Rare Events and Dependability

The concept of *rare event* simply refers to events whose probability of occurrence is "very low", where the meaning of "very low" is absolutely context–dependent. *Rare event* analysis is a subject of interest in areas like accident risk assessment, especially on accidents that can lead to the loss of lives, provision of services of high need such as communications or energy supply systems, and risk of ruin in insurance or investment companies, to name a few. It is, therefore, hard to say which probability values give an event the character of rare. Anyway, in areas like the ones mentioned, it is not far from the truth to say that events whose probability is in the order of $1\text{E}{-}09$ can be considered *rare events*.

However, *rare event* analysis is not only restricted to the determination of a probability value. It also includes the analysis of parameters and measures that somehow reflect systems' performance and *dependability* when these measures are directly related to the occurrence of *rare events*. In the latter case, the determination of performance or *dependability* measures can be as hard as the determination of the probability itself.

An illustrating example is a system that provides some kind of service while it is *up*, and quits providing it (it goes *down*) at the occurrence of a failure. Suppose that the probability of a failure occurrence is extremely low. Accepting that the system is *up* at time $t = 0$, a value that may be of interest is the probability that the system is still *up* at time $t > 0$, but it may also by of interest the mean time elapsed since the system is *up* at $t = 0$ and the first time it goes *down*. The first case is nothing but a probability determination, the second one consists in finding the expected value of

a time period. But both of them are affected by a *rare event* that, in this problem, is the systems' failure.

Rare event analysis attempts to give support in both, the design of new systems and the analysis of existing ones.

Events probabilities and *dependability* measures are usually given by a mathematical expression. When the size of a system is large, the complexity of these mathematical expressions can be very high and it may not be possible to compute them in a reasonable time. The size of the systems is, therefore, the first stumbling block in the determination of a *dependability* measure. An alternative solution is to resign the exact calculation, and to make an estimation instead. Standard simulation is the first method at hand but, if the number of replications does not exceed several times the inverse of the probability of the *rare event*, the estimation may not succeed or may not be possible at all. And, if the probability of the *rare event* is extremely low, the number of replications needed may be huge or directly unattainable.

Figures 1.1 and 1.2 show two *toy* models useful to highlight the drawbacks of standard simulation when the estimation is affected by the *rare event* problem. Each of the models belong to a class that will be described in more detail later in this thesis.

Figure 1.1 depicts a network —known as Bridge network— composed by four nodes connected by five links. The links can be *failed*, what is the same as being removed from the network, or *operational*, what is equivalent to connect the nodes at its extremes. The purpose of this network is to guarantee the existence of at least one path formed by *operational* links between the two marked nodes $\mathbf{s}$ and $\mathbf{t}$. Every link is *operational* with probability r and *failed* with probability $1-r$. The value of interest is the probability Q, that it does not exist a path formed by *operational* links between the nodes $\mathbf{s}$ and $\mathbf{t}$. The probability Q is known as the network *unreliability*, whereas $R = 1 - Q$, the probability that it does exist a path formed by *operational* links between $\mathbf{s}$ and $\mathbf{t}$, is called network *reliability*. Standard simulation consists in building N independent settings such that, in each one of them, every link is randomly set to *operational* or *failed* according to the links' probability distribution (Bernoulli, where the 1 is called *operational* and the 0 is called *failed*). There will be, therefore, N random settings —called samples— in which the nodes $\mathbf{s}$ and $\mathbf{t}$ may be connected or not, by a path of *operational* links. The standard estimation of Q, called $\widehat{Q}$, is the proportion of these N samples for which the nodes $\mathbf{s}$ and $\mathbf{t}$ are not connected.

To understand the limitations of standard simulation, consider the following analysis. Call $N_Q \leq N$ the number of samples for which the nodes $\mathbf{s}$ and $\mathbf{t}$ are not connected, and $N_R \leq N$, the number of samples for which $\mathbf{s}$ and $\mathbf{t}$ are connected. Clearly, $N_Q + N_R = N$, and $\widehat{Q} = N_Q/N$. It will be shown in Section 2.2 that $\widehat{Q}$ is an unbiased estimator of Q, i.e. $\mathbb{E}\{\widehat{Q}\} = Q$, but before this, an analysis of the standard method leads to Law of Large Numbers, that in this problem takes the form:

$$\lim_{N\to\infty} \mathbb{P}\left\{\left|\widehat{Q} - Q\right| < \varepsilon\right\} = 1, \ \ \forall \varepsilon > 0, \tag{1.1}$$

stating that the probability that $\widehat{Q}$ equals Q gets close to 1 as N increases to infinity. This indicates that an estimation made out of N_1 samples is likely to be better (closer to the real value) than an estimation made out of $N_2 < N_1$ samples. So, the use of large values of N is recommended, and increasing the value of N is always useful.

It is also of interest to measure how far the estimator $\widehat{Q}$ is from the real value Q. Accepting

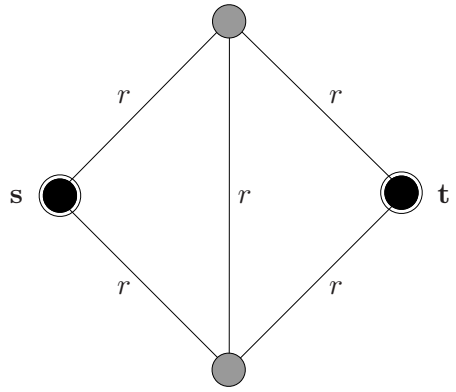


Figure 1.1: Bridge network

that the variance of the estimator is: $\mathbb{V}\{\widehat{Q}\} = \mathbb{E}\{(\widehat{Q} - Q)^2\} = \mathbb{E}\{\widehat{Q}^2\} - Q^2 = \sigma^2/N$, where σ^2 is the variance of the variable whose expected value is to be estimated, the Central Limit Theorem takes the form:

$$\lim_{N\to\infty} \mathbb{P}\left\{\frac{\widehat{Q} - Q}{\sigma}\sqrt{N} < z\right\} = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{z} e^{-y^2/2}dy \tag{1.2}$$

indicating that the asymptotic behaviour of the —properly scaled— error, follows the Standard Normal distribution. If N is large enough to consider that the difference $\widehat{Q} - Q$ is indeed normally distributed, expression (1.2) can be transformed into:

$$\mathbb{P}\left\{\widehat{Q} - z_{\alpha/2}\,\frac{\sigma}{\sqrt{N}} \le Q \le \widehat{Q} + z_{\alpha/2}\,\frac{\sigma}{\sqrt{N}}\right\} = 1 - \alpha$$

where $-z_{\alpha/2}$ and $+z_{\alpha/2}$ are two values such that a Standard Normal variable is between them with probability $1 - \alpha$. This expression defines a Confidence Interval, as it states that the exact value, Q, will lie inside the interval $\left[\widehat{Q} - z_{\alpha/2}\,\sigma/\sqrt{N}, \widehat{Q} + z_{\alpha/2}\,\sigma/\sqrt{N}\right]$ with probability $1 - \alpha$. A typical example of this kind of Confidence Interval is:

$$\mathbb{P}\left\{\widehat{Q} - 1.96\,\frac{\sigma}{\sqrt{N}} \le Q \le \widehat{Q} + 1.96\,\frac{\sigma}{\sqrt{N}}\right\} = 95\%$$

indicating that, with probability of 95%, the difference $|\widehat{Q} - Q|$ will be, at most, $1.96\,\sigma/\sqrt{N}$. Thus, this is not only a formula to compute how far the estimator could be from the estimated value, but also an indication that there are two ways to make the estimator be closer to the estimated value: increasing the number of samples, N, and decreasing the variance, σ^2, of the variable whose expectation is subject to the estimation.

This result must be accepted with some care because of two reasons. One of them is that ordinary simulations do not use "real" random numbers to build up the random settings, but pseudo–random numbers instead. This may produce loss of independence among samples. And the other reason is that in real implementations the number of samples, N, is directly related to the compu-

tational effort of the simulation, therefore, its value may not always be large enough.

After the preceding observations it immediately arises the question about how small the value of N can be to guarantee that the estimator will still be close to the real value. There is no categorical answer to this question. Going back to the network problem, it is clear that if Q is low —and even worse if Q it is extremely low— the number of samples, N, may not be as large as to make $N_Q > 0$. If $N_Q = 0$, what means that none of the N sampled settings reach a condition in which there is no path of *operational* links between the nodes $\mathbf{s}$ and $\mathbf{t}$, the standard estimator will take a deceptive value of 0, and expressions (1.1) and (1.2) will be far from being truth. Based on the ideas presented so far, the variance of the estimator, that is an indication of how dispersed the values of $\widehat{Q}$ are, may be considered also an indicator of how likely the estimator is to be close, or not, to the estimated value. It will be shown in Section 2.2 that, calling $\widehat{Q}_{N_i}$ the estimator obtained out of N_i samples, then, $\mathbb{V}\{\widehat{Q}_{N_1}\} \leq \mathbb{V}\{\widehat{Q}_{N_2}\}$ if $N_1 > N2$. It is also true that, if after changing one or more parameters of the network, but keeping fixed the value of N, the value Q_1 goes to $Q_2 < Q_1$, then $\mathbb{V}\{\widehat{Q}_2\} < \mathbb{V}\{\widehat{Q}_1\}$. So, an isolated value of a variance is not enough to decide how accurate an estimator is.

To evaluate the accuracy of the estimator, an indicator that is frequently used is the *coefficient of variation*, that in Monte Carlo estimation is defined as the ratio of the standard deviation of the estimator to the exact value estimated: $\mathbb{V}\{\widehat{Q}\}^{1/2}/Q$. This relation will be analyzed in more detail in Section 2.2. Now it is interesting to see, in Table 1.1, the evolution of the values it takes for different values of Q.

Table 1.1: Standard Simulation on the Bridge network in Figure 1.1

N	r	Q	$\widehat{Q}$	$\mathbb{V}\{\widehat{Q}\}$	$\mathbb{V}\{\widehat{Q}\}^{1/2}/Q$
1E+08	0.9	2.1520E−02	2.1534E−02	2.11E−10	0.07%
1E+08	0.99	2.0195E−04	2.0127E−04	2.02E−12	0.70%
1E+08	0.999	2.0020E−06	2.0600E−06	2.00E−14	7.07%
1E+08	0.9999	2.0002E−08	2.0000E−08	2.00E−16	70.71%
1E+08	0.99999	2.0000E−10	—	2.00E−18	707.10%

The *coefficient of variation*, seen as a *relative error*, increases as the *unreliability* Q decreases. There is a limit beyond which it is extremely unlikely that, in one of the sampled settings, the nodes $\mathbf{s}$ and $\mathbf{t}$ are disconnected. This example illustrates this effect numerically; when Q is smaller than 2.0002E−08, a sample size of 1E+08 is not enough to obtain an estimation.

A similar problem arises in the model shown in Figure 1.2. It is a continuous time Markov chain that represents a system composed of three components, each of which fails with rate λ. If one of the components is *failed*, it is repaired with rate μ If two components are *failed*, they are both —simultaneously— repaired with rate μ. If all the components fail, there is no possible reparation. There are four states associated with the number of components already *failed*: $\mathbf{u}$, where no component is *failed*; $\mathbf{d}$, where all components are *failed*, and the remaining ones, indicating one or two components *failed*, as it follows clearly from the associated graph. The system is considered *up* when it is in state $\mathbf{u}$ and *down* when it reaches state $\mathbf{d}$. It is of interest the probability γ that, after starting at state $\mathbf{u}$ in time $t = 0$, the system hits state $\mathbf{d}$ in time $t = t_{\mathbf{d}}$, conditioned to the fact that state $\mathbf{u}$ has not been visited in the period $(0, t_{\mathbf{d}}]$.

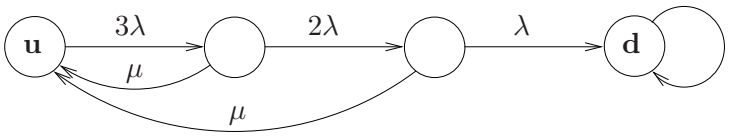


Figure 1.2: Continuous time Markov chain

If the components fail with a low (or very low) rate λ and they are repaired with a high (or very high) rate μ, the event whose probability γ is of interest is clearly a *rare event* and the associated problems of the standard *Monte Carlo* estimation, shown in Table 1.2, are essentially the same as the ones commented in the problem of the Bridge network.

Table 1.2: Standard Simulation on the Markov chain in Figure 1.2

N	λ, μ	γ	$\widehat{\gamma}$	$\mathbb{V}\{\widehat{\gamma}\}$	$\mathbb{V}\{\widehat{\gamma}\}^{1/2}/\gamma$
1E+09	1,1000	1.9940E−06	2.0220E−06	1.99E−15	2.24%
1E+08	1,1000	1.9940E−06	1.9500E−06	1.99E−14	7.08%
1E+07	1,1000	1.9940E−06	1.6000E−06	1.99E−13	22.39%
1E+06	1,1000	1.9940E−06	1.0000E−06	1.99E−12	70.82%
1E+05	1,1000	1.9940E−06	—	1.99E−11	223.94%

When the number of samples N is many orders of magnitude higher than the inverse of the estimated value, the ratios $\mathbb{V}\{\widehat{Q}\}^{1/2}/Q$ and $\mathbb{V}\{\widehat{\gamma}\}^{1/2}/\gamma$ —seen as *relative errors*— indicate that the estimators are somehow close to the estimated values. But, if the estimated values become smaller, they do it faster than the corresponding standard deviations, reason why as the event of interest becomes more rare the *relative errors* become higher. Another fact, also highlighted in the experiments, is that there are lower bounds for the value of N, beyond which the estimations are not possible.

This thesis is focused on some methods aimed at improving the standard simulation in the context of rare events, by means of variance reduction. As the main contribution of this work, two new methods are introduced, one of them applies to the estimation of network *unreliability* Q, on large and highly reliable networks; the other one is intended to estimate the value of γ on highly *dependable* systems modelled by large continuous time Markov chains.

1.2 Publications

The ideas behind this thesis were explored by the author, for the first time, in [Murray 2007a] and [Murray 2007b], where five *Monte Carlo* methods were reviewed, analyzed and applied to the *Diameter Constrained Reliability*, which is an extension of the previously introduced network *reliability* model. Four of these methods, namely, *Dagger*, *Permutation Monte Carlo*, and *Cross-Entropy* over *Crude* and *Permutation*, are variance reduction techniques thought of as an improvement to the crude or standard simulation. Both articles explain the methods in detail and compare their efficiency in tests made on selected network models.

After this first approach to *Monte Carlo* methods, the work continued on the line of the network *reliability* problem, with a proposal in which a classical method, called *Creation Process*, was improved by the application of *Splitting*, a variance reduction technique that had not been used in this context before. The proposal and the results obtained were presented in [Murray 2008a], [Murray 2008b], [Murray 2008c], [Murray 2010] and recently published in [Murray 2013b].

After the development of *Splitting* on the *Creation Process*, another classical technique called *Conditional Monte Carlo*, was recalled and applied by the author to the *dependability* analysis of Markovian systems. This problem has been the subject of research for many authors and different solutions have been proposed, most of them derived from *Importance Sampling* and, to a lesser extent, from *Splitting*. The *Conditional Monte Carlo* method presented in this thesis is an efficient attempt to reduce the variance of the standard estimation in the Markovian systems simulation. This proposal was introduced in [Murray 2012] and [Murray 2013a].

The following list corresponds to the publications just cited in this section. There are listed here with the only purpose of making the section self contained. These references can also be found in the Bibliography, at the end of this document.

[Murray 2007a] *Comparación entre Cinco Métodos de Monte Carlo para Estimar la Confiabilidad Diámetro Acotada de Redes de Comunicaciones*. Technical Report INCO 07-07, Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay, 2007. ISSN: 0797-6410.

[Murray 2007b] *Comparison of Five Monte Carlo Methods to Estimate the Network Diameter Constrained Reliability*. In Proceedings of the XXXII Latin-American Conference on Informatics (CLEI), San José, Costa Rica, 2007.

[Murray 2008a] *Monte Carlo Splitting Technique for Source-Terminal Network Reliability Estimation*. In Proceedings of the XXXIV Latin-American Conference on Informatics (CLEI), Santa Fé, Argentina, 2008.

[Murray 2008b] *Splitting in Source–Terminal Network Reliability Estimation*. In Proceedings of the 7th International Workshop on Rare Event Simulation, pages 57–68, Rennes, France, September 2008.

[Murray 2008c] *Splitting in the Simulation of the Network Creation Process*. Technical Report INCO 08-21, Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay, 2008. ISSN:0797-6410.

[Murray 2010] *Network Reliability Evaluation by Splitting a Link Creation Process*. In Proceedings of the ALIO–INFORMS Joint International Meeting, Buenos Aires, Argentina, June 2010.

[Murray 2012] *On Conditional Monte Carlo in Rare Event Probability Estimation*. In Proceedings of the 9th International Workshop on Rare Event Simulation, pages 57–68, Trondheim, Norway, September 2012.

[Murray 2013a] *A conditional Monte Carlo with intermediate estimations for computing MTTF of Markovian systems*. In Proceedings of the Sixth Latin–American Symposium on Dependable Computing (LADC), Rio de Janeiro, Brazil, 2013.

[Murray 2013b] *A Splitting algorithm for network reliability estimation*. IIE Transactions, vol. 45, no. 2, pages 177–189, 2013.

1.3 Structure of the thesis

In this chapter, the basic ideas behind this thesis were introduced, the concepts of *rare events* and *dependability* were explained and the publications of the author, directly related with this thesis, were showed. The rest of the work is divided into six more chapters, whose contents are as follows.

Chapter 2 introduces some measures of *dependability*, discusses the standard simulation and shows its drawbacks. Then it briefly describes *Importance Sampling*, *Splitting* and *Conditional Monte Carlo*.

Chapter 3 describes two variants of *Splitting* known, respectively, as *Fixed Splitting* and *Fixed Effort*. For these two variants it proves their unbiasedness and determines the variance of the corresponding estimators in a model with some constraints.

Chapter 4 introduces *Splitting/CP*, an original proposal of this thesis in which *Splitting* is applied to improve a well–known method called *Creation Process*. The proposed method is shown to be particularly efficient in the *reliability* estimation of highly reliable networks, i.e. networks for which failure is a *rare event*.

Chapter 5 shows the variance reduction capacity of *Conditional Monte Carlo* by means of many examples on different settings. This chapter behaves as a background for Chapter 6, in which another proposal of this thesis, derived from *Conditional Monte Carlo*, is introduced.

Chapter 6 introduces *Conditional Monte Carlo with Intermediate Estimations* (*CMIE*), a simulation method proposal in which *Conditional Monte Carlo* is applied recursively. The proposed method is shown to be particularly efficient in the context of highly reliable Markovian systems.

Chapter 7 presents the concluding remarks, highlights the main contributions and discusses possible research lines to continue the work of this thesis.

CHAPTER 2

Rare Event Simulation

Contents

Abstract

This chapter describes the setting of this thesis. In the first section some measures of *dependability* are introduced. After this, the standard simulation method is presented, discussing also the drawbacks that make it necessary to develop more efficient methods. Next, *Importance Sampling* is briefly described, and finally *Splitting* and *Conditional Monte Carlo* are introduced, as these two methods are the main subject of the issues around which the thesis is developed. *Splitting* and *Conditional Monte Carlo* will be revisited and treated in more detail in the following chapters.

2.1 Measures of Dependability

As all man–made systems are designed to provide some kind of service, it is usually of interest to measure their ability to provide such service. *Dependability* measures are intended to reflect this ability. However, due to some systems' particular features, the determination of these measures is not always straightforward.

When the *dependability* measures are given by a mathematical expression, the first interest should be to check whether they can be computed on models that represent the systems. In many cases, the price to pay in order to obtain a good representation of the systems, is to have a large size model. Then, the complexity of these mathematical expressions can become very high, being impossible to compute them in a reasonable time. An alternative solution is to simulate the systems in order to obtain an estimation of the *dependability*, instead of its exact value. However, the fact that the systems are highly dependable, meaning that their ability to provide service is extremely high, is a serious drawback in standard simulation. The higher the *dependability*, the lower the accuracy of the standard estimator.

The accuracy of an estimator is measured by its variance. Standard simulation must therefore be improved by the so–called variance reduction techniques in order to make simulations more

efficient. Due to variance reduction, accurate estimations can be achieved in a reasonable amount of time.

The main concern of this thesis is the problem of estimating *dependability* measures for two types of system, communication networks and Markovian systems. In both cases, large and highly dependable systems are considered.

The communication network model used in this thesis, is focused to the *static reliability* approach. The model (described in more detail in Chapter 4) is based on an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes and $\mathcal{E}$ the set of links. Nodes never fail, they are in an *operational* state all the time, while links fail independently and can only be found in one of two states, *operational* or *failed*. The network *reliability* R (*unreliability* Q) is defined as the probability that a set of terminal nodes $\mathcal{K} \subseteq \mathcal{V}$ is connected (not connected) in the sub–graph containing all the nodes in $\mathcal{V}$ but only the *operational* links (as for a link being *failed* is the same as being removed from $\mathcal{G}$).

Call network *instance* the setting in which a possible value (*operational* or *failed*) is sampled for every one of the links in $\mathcal{E}$. There will be, therefore, $2^{|\mathcal{E}|}$ different *instances*. To compute the exact value of the *reliability* R, or the *unreliability* Q, the most straightforward method is to generate all possible *instances* and, for each one of them, to check whether the set of terminal nodes $\mathcal{K} \subseteq \mathcal{V}$ is connected or not. The proportion of all instances for which the terminal nodes are connected, weighted by their probability values, is R and the proportion for which terminal nodes are not connected, weighted by their probability values, is Q. If $\mathcal{E}$ is large, or even moderately large, it is impossible to compute R or Q in a reasonable time.

The other model on which this thesis is focused, is a continuous time Markov chain $\{X(t), t \geq 0\}$, with discrete state space S. The state space S is partitioned into two subspaces: U, where the system is *up*, and D, where the system is *down* (the subspace D can be collapsed into a single absorbing state $\mathbf{d}$). The model operates as follows: $X(t)$ starts at some initial state $\mathbf{u} \in U$ (a state in which the system is fully *operational*) in time $t = 0$, and it stops either when it comes back to state $\mathbf{u}$ in time $\tau_{\mathbf{u}}$ or when it hits state $\mathbf{d}$ in time $\tau_{\mathbf{d}}$. In the mean time the process moves through states in which the system is partially damaged (not as much as to be in state $\mathbf{d}$) but it keeps providing service.

For this type of (ergodic) Markovian model, *dependability* measures can be classified into two groups: *steady–state* and *transient*. The latter are those metrics evaluated at some point in time t, or over some finite interval, typically of the form $[0, t]$. The two main transient metrics are the *reliability* at time t and the *availability* at time t. The *reliability* at time t, denoted $R(t)$, measures the "continuity" of service: $R(t) = \mathbb{P}\{X(s) \in U,\ 0 \leq s \leq t\}$. That is, $R(t)$ is the probability that the system operates as specified during the whole interval $[0, t]$. The *availability* at time t is the probability that the system is working at time t, $A(t) = \mathbb{P}\{X(t) \in U\}$.

A more complex transient metric is the *interval availability* on $[0, t]$, denoted $IA(t)$. It is defined as the fraction of the interval during which the system is up, that is:

$$IA(t) = \frac{1}{t} \int_{s=0}^{t} 1_{\{X(s) \in U\}} ds,$$

where $1_{\{e\}}$ is the indicator of event e. Since $IA(t)$ is a random variable, in practice, other specific

related metrics are used. For instance, its expectation, called the *expected interval availability* on $[0, t]$:

$$\mathbb{E}\{IA(t)\} = \frac{1}{t} \int_{s=0}^{t} A(s)ds,$$

or its distribution (much harder to compute).

Steady-state measures are defined on the system in equilibrium, that is, on a stationary version of the stochastic process modelling it. The main one is the *asymptotic availability*, $A(\infty) = \lim_{t\to\infty} A(t)$. See that:

$$A(\infty) = \lim_{t\to\infty} IA(t) \quad \text{a.s.}$$

The expected time from the beginning until the first system failure is another important *dependability* metric, called the *mean time to failure*, in short MTTF. Using the variable τ_d, MTTF = $\mathbb{E}\{\tau_d\}$. Also $R(t) = \mathbb{P}\{\tau_d > t\}$, and thus, MTTF = $\int_{t=0}^{\infty} R(t)dt$. Using the fact that X is Markovian:

$$\text{MTTF} = \frac{\mathbb{E}\{\min(\tau_{\mathbf{d}}, \tau_{\mathbf{u}})\}}{\gamma},$$

where $\gamma = \mathbb{P}\{\tau_{\mathbf{d}} < \tau_{\mathbf{u}}\}$. Both terms, $\mathbb{E}\{\min(\tau_{\mathbf{d}}, \tau_{\mathbf{u}})\}$ and γ, can be estimated using regenerative properties of X.

The main issues on which this thesis is focused are the development of variance reduction techniques aimed at the achievement of accurate estimations of the *reliability* R (*unreliability* Q) of large and highly reliable networks and the estimation of $\gamma = \mathbb{P}\{\tau_{\mathbf{d}} < \tau_{\mathbf{u}}\}$ for large and highly dependable markovian systems.

2.2 Standard Monte Carlo

In many *dependability* analysis, the problem reduces to the determination of the expectation of some random variable X. Sometimes X is an event indicator random variable, in the sense that it equals 1 if the event occurs, and 0 if the event does not occur. But there are cases in which X takes values different than 0 and 1, and its expectation is still the value of interest. The most simple simulation method to estimate the expected value of X is the one known as naive, crude or *Standard Monte Carlo*. This method was briefly and informally introduced in Section 1.1. Here it will be presented in more detail and the drawbacks due to the *rare event* problem will be formally analyzed.

Let X be a continuous random variable with probability density function $f(x)$ and expectation E:

$$E = \mathbb{E}\{X\} = \int_{\infty}^{+\infty} x f(x)\, dx$$

If X is a discrete random variable that takes values x_i, $i = 0, \cdots, n$, respectively with probability

$p(x_i)$, its expectation, E, is:

$$E = \mathbb{E}\{X\} = \sum_{i=0}^{n} x_i \, p(x_i)$$

In both cases, the variance of X is: $\sigma^2 = \mathbb{V}\{X\} = \mathbb{E}\{(X-E)^2\} = \mathbb{E}\{X^2\} - E^2$.

Although the probability distributions are not always available, it is often necessary to know the value of E. Crude, naive or *Standard Monte Carlo* is the simplest simulation method, thought of as to obtain an estimator $\widehat{E}$ of E, hoping that $\widehat{E}$ and E will be close enough. The estimation begins with the sampling process, which consists in sampling a set of independent values $X^{(i)}$, $i = 1, \cdots, N$. These values are collected using a random number generator and an algorithm designed to produce values according to the probability distribution of X (whether it is continuous as if it is discrete). The samples, sometimes called copies of X, can be considered random variables —in most cases independent— with the same probability distribution as X. The *Standard Monte Carlo* estimator is:

$$\widehat{E} = \frac{1}{N} \sum_{i=0}^{N} X^{(i)}.$$

It is simple to show that $\widehat{E}$ is an unbiased estimator of E:

$$\mathbb{E}\{\widehat{E}\} = \mathbb{E}\left\{\frac{1}{N}\sum_{i=0}^{N} X^{(i)}\right\} = \frac{1}{N}\sum_{i=0}^{N} \mathbb{E}\left\{X^{(i)}\right\} = \frac{1}{N}\sum_{i=0}^{N} E = E$$

while the variance of $\widehat{E}$ is:

$$\mathbb{V}\{\widehat{E}\} = \mathbb{V}\left\{\frac{1}{N}\sum_{i=0}^{N} X^{(i)}\right\} = \frac{1}{N^2}\sum_{i=0}^{N} \mathbb{V}\left\{X^{(i)}\right\} = \frac{1}{N^2}\sum_{i=0}^{N} \sigma^2 = \frac{\sigma^2}{N}.$$

Suppose that X is an indicator random variable, that is, X equals 1 if the event of interest occurs and equals 0 otherwise. In this case X is a Bernoulli random variable with parameter $E = \mathbb{E}\{\widehat{E}\}$ and variance $E(1-E)$. The *coefficient of variation*, usually accepted as an indication of the *relative error* of the estimation $\widehat{E}$, is:

$$RE(\widehat{E}) = \frac{\mathbb{V}\{\widehat{E}\}^{1/2}}{\mathbb{E}\{\widehat{E}\}} = \frac{(E(1-E)/N)^{1/2}}{E} = \left(\frac{1-E}{EN}\right)^{1/2} \approx \frac{1}{(EN)^{1/2}}. \tag{2.1}$$

This expression clearly highlights the drawbacks of standard simulation in *rare event* analysis. If X is the indicator of a *rare event*, the value of E will be extremely low, and this can make the *relative error* to be extremely large. Moreover, if E tends to zero, N being fixed, the *relative error* will tend to infinity. The only way to work around this problem is by increasing the number of samples, N. On the other hand, even if E is not extremely low, small values of N will produce the same effect, in the sense of making the *relative error* grow boundlessly.

In cases in which a variance reduction technique is applied, the variable subject of analysis is not a Bernoulli random variable and its variance, $\mathbb{V}\{\widehat{E}\}$, is other than $E(1-E)$. The expression of $\mathbb{V}\{\widehat{E}\}$ is the one that, ultimately, defines the properties of the variance reduction technique

employed, as it says how much the *relative error* reduces, compared to $1/(EN)^{1/2}$.

The exact value E is, of course, (almost) never available in real cases, it is actually the value the simulation attempts to find. That is why in expression (2.1) the estimator $\widehat{E}$ must be used instead. Something similar occurs with the variance, therefore, an unbiased estimator like the following, must be used in its place:

$$\widehat{\mathbb{V}}\{\widehat{E}\} = \frac{1}{N-1}\left[\frac{1}{N}\left(\sum_{i=1}^{N} {X^{(i)}}^2\right) - \widehat{E}^2\right].$$

This variance estimation and the estimator $\widehat{E}$, are the ones used to evaluate the *relative error* (2.1) in the experimental part of this thesis.

The context in which this thesis is framed is the estimation, by simulation, of probabilities and parameters directly related to events whose occurrence is extremely rare. If the indicator random variable X equals 1 when the event of interest occurs, and 0 otherwise, the expectation $E = \mathbb{E}\{X\}$ is, therefore, an indication of how likely the occurrence of the event of interest is. So far it's been shown how *Standard Monte Carlo* performs in this context, attempting to make an estimation $\widehat{E}$ of E. But the problem could be approached similarly on the variable $Y = 1 - X$. Variable Y equals 1 when the event of interest does not occur, and 0 otherwise. Thus, $F = \mathbb{E}\{Y\} = 1 - E$ is an indication of how likely the event of interest is not to occur, and an estimation $\widehat{F}$ of F could be as useful as $\widehat{E}$. However, in the context of *rare events*, $E \approx 0$ and $F \approx 1$. It will be shown in the rest of this section that, in the analysis of highly reliable systems, mostly if the interest is focused on the comparison of accuracy and precision among different methods, there are some benefits in using E instead of F.

Suppose that the decision is to estimate E. The most direct and well known definition of the *relative error* is δ/E, where $\delta = |E - \widehat{E}|$ is the absolute error. But E is, of course, unknown, and the only estimator at hand is the *Standard Monte Carlo* estimator $\widehat{E}$. Then, using the absolute error δ is clearly impossible and the standard deviation, $SD = \mathbb{V}\{\widehat{E}\}^{1/2}$, is used instead. See that SD is the same when estimating F or E (the variances, $E(1-E)/N$ and $F(1-F)/N$, are clearly equal because $E = 1-F$). Then, observe that the *relative error* $RE(\widehat{E})$ is greater than the *relative error* $RE(\widehat{F})$. More precisely, $SD \approx RE(\widehat{F}) \ll RE(\widehat{E})$. Then, controlling $RE(\widehat{E})$ guarantees that both errors are controlled. The use of $RE(\widehat{F})$ for controlling the estimation may lead to very poor accuracy on the estimation of E, which is extremely annoying (think of critical systems where E is the probability of a critical failure). For instance, let the true values be $E = 10^{-10} = 1 - F$, unknown, and pretend that the simulation produces an estimation $\widehat{E} = 10^{-9} = 1-\widehat{F}$. The absolute error is $\delta = 9 \times 10^{-10}$, and $RE(\widehat{F}) = 9 \times 10^{-10}/(1 - 10^{-10}) \approx 9 \times 10^{-10}$, but $RE(\widehat{E}) = 9$. Then, after an apparently good precision in the estimation of F there is a catastrophic 900% error on the estimation of E.

There is another risk in using $RE(\widehat{F})$ when $F \approx 1$. As SD is the same when estimating F or E, it follows that $RE(\widehat{F}) = RE(\widehat{E})\, E/F \approx RE(\widehat{E})\, E$. Thus, even when both estimates, $\widehat{F}$ and $\widehat{E}$, are at the same "distance" of the corresponding true value, $RE(\widehat{E})$ is about $1/E$ times larger that $RE(\widehat{F})$. Let again be a true value $E = 10^{-10}$ —reasonable in *rare event* simulation— and let $RE(\widehat{E}) = 10^{-3}$, then $RE(\widehat{F}) = 10^{-13}$, that could be negligible and confused with 0, unless the number of digits involved in the computation is high enough. Then, if the interest is focused

on "how close, or how far" the estimation is from the true value when E is a very low value, both *relative errors* contain the same information, $RE(\widehat{E})$ in a significant and mensurable number, and $RE(\widehat{F})$ in a negligible number.

2.3 Importance Sampling

In standard simulation, as in any *Monte Carlo* method intended to make estimations of some parameter, streams of random numbers are necessary to build up the random settings. Random numbers are generated according to the probability distributions associated with the different components of the simulated system. These probability distributions are the ones that, ultimately, define the behaviour of the simulation. Such distributions are closely related to the accuracy of the simulation, formally measured by the variance of the estimator. From a practical point of view, the probability distributions involved define how rare is that the event of interest occurs in one —or more— of the random settings built by the simulation.

There is a well known method called *Importance Sampling* based on the idea of changing the probability distributions of the model in order to make the event of interest be "frequent" instead of "rare" [Fishman 1996]. The estimation made after the change is biased and has to be corrected in order to recover the estimation corresponding to the system before the changes. After this, the problems due to the rarity are vanished and the variance of the estimator is reduced.

This section briefly introduces the basis of *Importance Sampling* and highlights its features in a very simple example.

Let X be a random variable with probability density function $f(x)$, and $\phi(X)$ a function of X. For simplicity the method will be introduced only for the case in which X is continuous, but the ideas apply as well and are easily extended to the discrete case. The expectation, E_f, of $\phi(X)$ is:

$$E_f = \mathbb{E}_f\{\phi(X)\} = \int_{-\infty}^{+\infty} \phi(x) f(x)\, dx \tag{2.2}$$

where the subscript f is a reference to the probability density function of X. The corresponding *Standard Monte Carlo* estimator is:

$$\widehat{E}_f = \frac{1}{N} \sum_{i=0}^{N} \phi(X^{(i)}) \tag{2.3}$$

where the copies $\phi(X^{(i)}),\ i = 1, \cdots, N$ are sampled according to the probability density function $f(x)$. Suppose that the standard estimation is not efficient, either because it is difficult to sample from $f(x)$ or just because the event of interest almost never occurs. Suppose also that, as a consequence of such inefficiency, the variance of $\widehat{E}_f$ is large. The solution proposed by *Importance Sampling* is to sample according to a new probability density function $h(x)$, instead of $f(x)$. Provided that $h(x)$ is not equal to 0 unless in those points for which $f(x)\phi(x)$ equals 0, expression (2.2) can be transformed as follows:

$$\mathbb{E}_f\{\phi(X)\} = \int_{-\infty}^{+\infty} \phi(x)\ f(x)\ dx = \int_{-\infty}^{+\infty} \phi(x)\ \frac{f(x)}{h(x)}\ h(x)\ dx = \mathbb{E}_h\left\{\frac{\phi(X)f(X)}{h(X)}\right\},$$

suggesting that the estimation in (2.3) should be transformed accordingly, that is:

$$\widehat{E}_h = \frac{1}{N}\sum_{i=0}^{N} \frac{\phi(X^{(i)})f(X^{(i)})}{h(X^{(i)})} = \frac{1}{N}\sum_{i=0}^{N} \phi(X^{(i)})\ R(X^{(i)})$$

where the copies $\phi(X^{(i)}f(X^{(i)})/h(X^{(i)}),\ i = 1, \cdots, N$ are sampled according to the probability density function $h(x)$. The expected value of $\phi(X)$ under the probability density function $f(x)$ equals the expected value of $\phi(X)f(X)/h(X)$ —seen as a random variable— under the probability density function $h(x)$. The factor $R(x) = f(x)/h(x)$ is called *likelihood ratio*. In the new sampling scheme, the copies $X^{(i)}$ are generated from the distribution $h(x)$ and then, $\phi(X^{(i)})$ and $R(X^{(i)})$ are computed. The process of changing the distribution from which the copies are sampled, is known as *change of measure*.

Accepting that the target is still the estimation of the expected value of $\phi(X)$, either $\widehat{E}_f$ or $\widehat{E}_h$ solve the problem, because the expectation of both is $\mathbb{E}\{\phi(X)\}$. The key to justify the use of one or the other is the comparison of the variances before and after the *change of measure*, that is:

$$\mathbb{V}_f\{\phi(X)\} = 1/N\left\{\int_{-\infty}^{+\infty} \phi(x)^2\ f(x)\ dx - \mathbb{E}_f^2\{\phi(X)\}\right\} \text{ and}$$
$$\mathbb{V}_h\{\phi(X)\ R(X)\} = 1/N\left\{\int_{-\infty}^{+\infty} \phi(x)^2 R(x)^2\ h(x)\ dx - \mathbb{E}_h^2\{\phi(X)\ R(X)\}\right\}.$$

Indeed a variance reduction takes place if:

$$\mathbb{V}_f\{\phi(X)\} - \mathbb{V}_h\{\phi(X)\ R(X)\} = \int_{-\infty}^{+\infty} [1 - R(x)]\ \phi(x)^2\ f(x)\ dx \geq 0.$$

This last expression is only to check whether some selected distribution $h(x)$ produces a variance reduction or not, but is not a formula to find a distribution $h(x)$ to achieve a desired variance reduction. The task of finding an appropriate $h(x)$ or, to say it in terms of *Importance Sampling*, to produce an adequate *change of measure*, is not straightforward. Many research lines have attempted to solve this problem, resulting in several methods [L'Ecuyer 2007b, L'Ecuyer 2011a, L'Ecuyer 2011b, Ridder 2005, Rubinstein 2004].

Further analysis leads to an interesting conclusion. Suppose that $\phi(X) \geq 0$ and that the *change of measure* is such that the distribution $h(x)$ is the following:

$$h(x) = \frac{\phi(x)\ f(x)}{\mathbb{E}_f\{\phi(X)\}}. \tag{2.4}$$

In this case, the variance of the estimator $\widehat{E}_h$ is given by:

$$\begin{aligned}
\mathbb{V}_h\{\phi(X)\ R(X)\} &= 1/N \left\{ \int_{-\infty}^{+\infty} \phi(x)^2 R(x)^2\ h(x)\ dx - \mathbb{E}_h^2\{\phi(X)\ R(X)\} \right\} \\
&= 1/N \left\{ \int_{-\infty}^{+\infty} \phi(x)^2 R(x)^2\ h(x)\ dx - \mathbb{E}_f^2\{\phi(X)\} \right\} \\
&= 1/N \left\{ \int_{-\infty}^{+\infty} \phi(x)^2 R(x)\ f(x)\ dx - \mathbb{E}_f^2\{\phi(X)\} \right\} \\
&= 1/N \left\{ \int_{-\infty}^{+\infty} \phi(x)^2 \frac{\mathbb{E}_f\{\phi(X)\}}{\phi(x)}\ f(x)\ dx - \mathbb{E}_f^2\{\phi(X)\} \right\} \\
&= 1/N \left\{ \mathbb{E}_f\{\phi(X)\} \int_{-\infty}^{+\infty} \phi(x)\ f(x)\ dx - \mathbb{E}_f^2\{\phi(X)\} \right\} \\
&= 1/N \left\{ \mathbb{E}_f\{\phi(X)\}\ \mathbb{E}_f\{\phi(X)\} - \mathbb{E}_f^2\{\phi(X)\} \right\} \\
&= 0
\end{aligned}$$

This seems to be an amazing result, as it implies that any sample obtained after applying this *change of measure* results in the exact value, meaning also that a single sample is enough to compute the desired measure. However, this result comes from a clearly impossible *change of measure* since to be achieved (see (2.4)) it is necessary to have the value $\mathbb{E}_f\{\phi(X)\}$, which is actually the value that the whole simulation is intended to obtain. Knowing the value of $\mathbb{E}_f\{\phi(X)\}$ makes it absolutely unnecessary to perform the estimation.

Anyway, despite this *change of measure* —call it the optimal— is clearly impossible, it is the base around which some variants have been developed. In one of them, called Zero-Variance [L'Ecuyer 2007b, L'Ecuyer 2011a, L'Ecuyer 2011b], different mechanisms attempt to find a *change of measure* that is somehow "close" to the optimal one, leading to an estimation that is also "close" to the exact value. Another variant, known as Cross-Entropy [Ridder 2005, Rubinstein 2004], uses a metric (the Kullback-Leibler distance) and chooses a *change of measure* that transforms the original probability distribution into a new one that is the closest to the optimal one under this metric.

In the rest of this section, a very simple but illustrating example is presented. The function whose expectation is of interest is $\phi(X) = X$, and the corresponding probability density function is:

$$f(x) = \begin{cases} 0 & x \leq d,\ d > 0, \\ \lambda\, e^{-\lambda(x-d)} & x > d. \end{cases}$$

Figure 2.1(a) shows $f(x)$, that is a shifted exponential distribution. Sampling from $f(x)$ is very simple, actually $X^{(i)} = d - \log U/\lambda,\ i = 1, \cdots, N$, where U is a sample from a uniformly distributed random variable in $(0, 1)$. The expectation of X is $d + 1/\lambda$ and the variance, $\mathbb{V}_f\{X\} = 1/\lambda^2$. Suppose that, for some purpose, this variance is high, and it is necessary to reduce it in order to estimate accurately the expectation of $\phi(X) = X$, from a "few" samples. It is possible to make this estimation by means of *Importance Sampling*.

Let the following $h(x)$ be the probability density function used in the application of *Impor-*

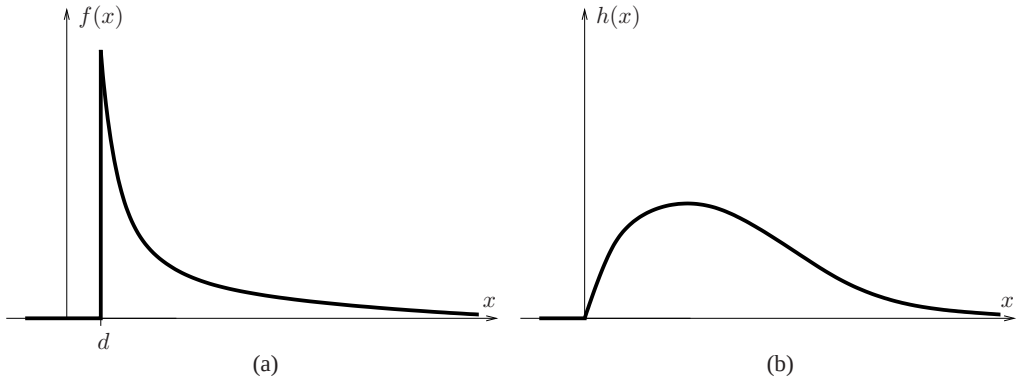


Figure 2.1: The functions involved in the *Importance Sampling* example

tance Sampling:

$$h(x) = \begin{cases} 0 & x \leq 0 \\ \lambda^2 x\, e^{-\lambda x} & x > 0 \end{cases}$$

Figure 2.1(b) shows $h(x)$. Sampling from $h(x)$ is still very simple because it is the distribution of the sum of two exponentially distributed random variables with parameter λ. Therefore, $X^{(i)} = \log U_1/\lambda - \log U_2/\lambda,\ i = 1, \cdots, N$, where U_1 and U_2 are samples from a uniformly distributed random variable in $(0, 1)$.

The *likelihood ratio* ratio is:

$$R(x) = f(x)/h(x) = \begin{cases} 0 & x \leq d, \\ e^{\lambda d}/\lambda x & x > d, \end{cases}$$

and the variance, after the *change of measure*, $\mathbb{V}_h\{X\ R(X)\} = 1/\lambda^2(\lambda d{+}1)[e^{\lambda d} - (\lambda d{+}1)]$. With an illustrative purpose, the values $\lambda = 0.1$ and $d = 0.1$ yield the following result: $\mathbb{V}_h\{X\ R(X)\} = \mathbb{V}_f\{X\}/19,736.03$. Then, at the expense of sampling two uniformly distributed values, instead of one, every time, estimating the expectation of $\phi(X)$ is much more accurate after the *change of measure*.

2.4 Splitting

The evolution of a Markov random process $\{X(t),\, t \geq 0\}$ corresponds to different trajectories in its state space. Rare events are associated with regions of the state space that trajectories reach with very low probability. The guiding principle of *Splitting* is to partition the state space into many sub spaces, and to recursively multiply or split trajectories as soon as they get into sub spaces that are somehow closer to the region of occurrence of the *rare event*.

The estimation made after splitting or multiplying trajectories is biased and has to be corrected in order to recover the estimation corresponding to the original system. Proceeding this way, the most promising trajectories are privileged and the variance of the estimator is reduced.

In the rest of this section *Splitting* is briefly introduced. In Chapters 3 and 4 some particular

features and applications are presented.

Let $\{X(t), t \geq 0\}$ be a Markov process with discrete state space $\mathcal{X}$, and let $\mathcal{X}_A$ and $\mathcal{X}_B$ be two disjoint regions in $\mathcal{X}$. Assume $X(0) \notin \mathcal{X}_B$. A quantity that is frequently of interest in different performance and *dependability* problems, is the probability γ that, starting at $t = 0$, process $\{X(t)\}$ enters $\mathcal{X}_B$ without having entered $\mathcal{X}_A$ before. If τ_A is the instant when $\{X(t)\}$ enters $\mathcal{X}_A$ the first time (or comes back to $\mathcal{X}_A$ if $X(0) \in \mathcal{X}_A$) and τ_B is the instant when $\{X(t)\}$ enters $\mathcal{X}_B$ the first time, then $\gamma = \mathbb{P}\{\tau_B < \tau_A\}$. Regions $\mathcal{X}_A$ and $\mathcal{X}_B$ may be defined implicitly via an importance function $h : \mathcal{X} \to \mathbb{R}$ as: $\mathcal{X}_A = \{x \in \mathcal{X} : h(x) \leq \ell_0\}$ and $\mathcal{X}_B = \{x \in \mathcal{X} : h(x) \geq \ell\}$, where ℓ_0 and ℓ are two values in $\mathbb{R}$ (usually $\ell_0 = 0$ and $\ell > 0$). Hence, if $h(X(0)) = 0$, τ_A is the first time $\{h(X(t))\}$ down–crosses ℓ_0, whereas τ_B is the first time $\{h(X(t))\}$ up–crosses ℓ.

Splitting [Garvels 2000, Glasserman 1996, L'Ecuyer 2007a, L'Ecuyer 2009, Villén-Altamirano 1991] is a variance reduction technique aimed at making accurate estimations of γ when $\{\tau_B < \tau_A\}$ is a *rare event*. In a *Splitting* application, the state space of $\{h(X(t))\}$ is partitioned by a set of real values $\ell_0 = 0 < \ell_1 < \ell_2 < \cdots < \ell_m = \ell$, as shown in Figure 2.2(a). Given this partition, for $i \geq 1$, $\tau_i = \inf\{t > 0 : h(X(t)) = \ell_i > h(X(t^-))\}$ and $\tau_0 = \inf\{t > 0 : h(X(t)) = \ell_0\}$.

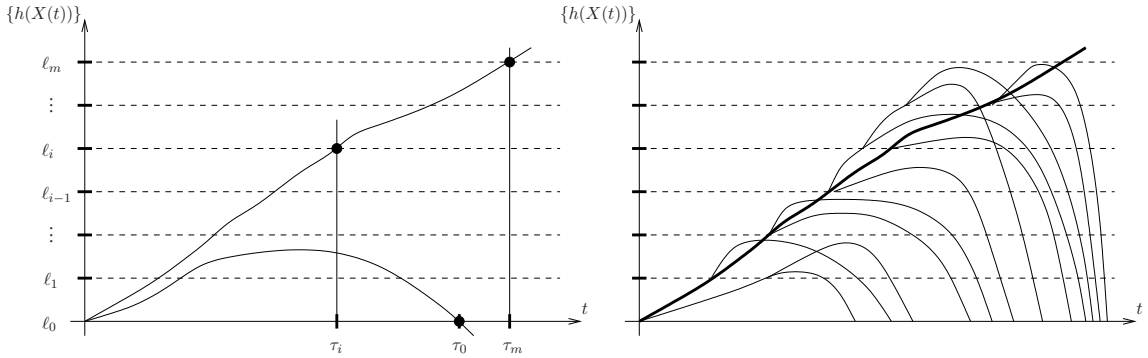


Figure 2.2: Sample replications over the state space of $\{h(X(t))\}$

The event $D_i = \{\tau_i < \tau_0\}$, $i = 1, 2, \cdots, m$, is an indication that $\{h(X(t))\}$ has up–crossed *threshold* ℓ_i without having entered the region under *threshold* $\ell_0 = 0$. It is clear that $D_m \subset D_{m-1} \subset \cdots \subset D_2 \subset D_1$, where $D_m = \{\tau_m < \tau_0\} = \{\tau_B < \tau_A\}$ is the event whose probability $\gamma_m = \gamma$ is the quantity of interest. Hence,

$$\gamma_m = \mathbb{P}\{D_m\} = \underbrace{\mathbb{P}\{D_m|D_{m-1}\}}_{p_m} \underbrace{\mathbb{P}\{D_{m-1}|D_{m-2}\}}_{p_{m-1}} \cdots \underbrace{\mathbb{P}\{D_2|D_1\}}_{p_2} \underbrace{\mathbb{P}\{D_1\}}_{p_1} = \prod_{i=1}^{m} p_i.$$

The *Splitting* estimation of γ_m is based on this expression. If an estimator $\widehat{p}_i$ is obtained for every p_i, the estimation of γ_m is $\widehat{\gamma}_m = \prod_{i=1}^{m} \widehat{p}_i$. Unbiasedness of this estimator has been recently proved in quite general settings by [Amrein 2011].

The process to obtain the estimators $\widehat{p}_i$ is as follows. If N_0 replications of $\{h(X(t))\}$ are launched from $t = 0$, and R_1 of them reach (up–cross) *threshold* ℓ_1, $\widehat{p}_1 = R_1/N_0$ is

an unbiased (*Crude Monte Carlo*) estimator of p_1. *Splitting multiplies* (*splits*) the replications that up-crosses ℓ_1, saving their state at the crossing point, and launching from there a number $N_1 > R_1$ of new replications, towards ℓ_2. If R_2 of them reach *threshold* ℓ_2, $\widehat{p}_2 = R_2/N_1$ is an estimator of p_2. Proceeding iteratively, the estimators $\widehat{p}_i,\ i = 2, \cdots, m$, are R_i/N_{i-1}. If *threshold* ℓ_m is reached by at least one replication of $\{h(X(t))\}$, the final estimation is: $\widehat{\gamma}_m = \prod_{i=1}^{m} \widehat{p}_i = R_1/N_0\ R_2/N_1\ R_3/N_2\ \cdots\ R_m/N_{m-1}$. This mechanism is shown in Figure 2.2(b). By the action of *splitting* or *multiplying*, the method privileges the replications for which the event $D_m = \{\tau_m < \tau_0\}$ is still likely to occur. According to the number of new copies that any replication of $\{h(X(t))\}$ is *multiplied by* or *split into*, *Splitting* supports two main implementation variants: *Fixed Splitting*, where the number of new copies started from every hitting point is a constant, $N_i/R_i = \alpha_i > 1\ \forall i$, and *Fixed Effort*, where the number of trajectories created at every hitting point is adjusted so as to let the total number of replications started from every *threshold* (*effort*) be a constant, $N_i = F_i\ \forall i$. There is no general agreement about the variance reduction benefits of each option. *Fixed Effort* avoids the risk of a combinatorial explosion on the number of trajectories (which is possible in *Fixed Splitting*) providing, therefore, a closer control on the execution times. *Fixed Effort* is the variant selected to support the experimental part of this thesis (see Section 4.6).

The variance of the *Splitting* estimator was analysed by [Garvels 1998] who, under the assumption that the $\widehat{p}_i$'s are all independent and identically distributed, proved that the value of $\mathbb{V}\{\widehat{\gamma}_m\}$ can be approximated by $\gamma_m^2 m^2 (1 - \gamma_m^{1/m})/(\gamma_m^{1/m} S)$, where $S = F_1 + F_2 + \cdots + F_{m-1}$. [L'Ecuyer 2007a] analysed a simplified *Fixed Effort* setting where $F_0 = F_1 = \cdots = F_{m-1} = F$ and the $\widehat{p}_i$'s are independent random variables such that $p_i = p = \gamma_m^{1/m}$. In this setting the variance $\mathbb{V}\{\widehat{\gamma}_m\}$ can be approximated by $m\gamma_m^{2-(1/m)}/F$.

2.5 Conditional Monte Carlo

From basic probabilistic analysis it is simple to show that if some random variable can be conditioned to the values of some other random variable, the variance after conditioning is lower than the variance of the variable before conditioning. The fact of conditioning the variable of interest to the observed values of some other variable, somehow reduces the state space of the original variable and yields a variance reduction.

In the rest of this section *Conditional Monte Carlo* is briefly introduced. In Chapters 5 and 6 a more insightful analysis and some applications are presented.

Let V be a random variable and θ an event on the state space of V. The probability $\gamma = \mathbb{P}\{\theta\}$ can be determined by means of the indicator random variable I:

$$I = \begin{cases} 1 & \text{if } \theta \text{ occurs,} \\ 0 & \text{otherwise,} \end{cases}$$

because $\mathbb{E}\{I\} = \mathbb{P}\{\theta\}$.

Suppose that $V = f(X_1, X_2, \cdots, X_n)$, or just $V = (X_1, X_2, \cdots, X_n)$, where $\{X_i\}_{i=1}^n$ is a set of random variables. If the value of any of these variables is fixed, e.g. $X_k = x_k$, the expected value of I conditioned to x_k is the fixed value $\mathbb{E}\{I | X_k = x_k\}$. As X_k is a random variable,

$\mathbb{E}\{I|X_k\}$ is a random variable too, because its value depends on the value of X_k. Moreover:

$$\mathbb{E}\{\mathbb{E}\{I|X_k\}\} = \mathbb{E}\{I\} = \mathbb{P}\{\theta\}$$

what means that $\mathbb{E}\{I|X_k\}$ and I are two random variables whose expectation is the value of interest, γ. See that the possible values of I are only 0 or 1 whereas —in general— the possible values of $\mathbb{E}\{I|X_k\}$ are real numbers between 0 and 1 ($\mathbb{E}\{I|X_k\}$ is the expectation of a variable that only takes the values 0 or 1). If these two variables are available, in the sense that it is possible to generate samples according to their probability distributions, γ can be estimated making estimations of either $\mathbb{E}\{I\}$ or $\mathbb{E}\{\mathbb{E}\{I|X_k\}\}$. It is interesting to see which one of these methods yield a more efficient estimator.

From a simple variance analysis:

$$\mathbb{V}\{I\} = \mathbb{E}\{\mathbb{V}\{I|X_k\}\} + \mathbb{V}\{\mathbb{E}\{I|X_k\}\} \tag{2.5}$$

thus,

$$\mathbb{V}\{I\} \geq \mathbb{V}\{\mathbb{E}\{I|X_k\}\} \tag{2.6}$$

because there are no negative terms in (2.5). This expression is the key to analyze the accuracy of the estimators of $\mathbb{E}\{I\}$ and $\mathbb{E}\{\mathbb{E}\{I|X_k\}\}$.

If $I^{(i)}$ and $\mathbb{E}\{I|X_k^{(i)}\},\ i = 1, \cdots$ are, respectively, independent samples of I and $\mathbb{E}\{I|X_k\}$, the following standard estimators:

$$\widehat{\gamma}_1 = \frac{1}{N_1}\sum_{i=1}^{N_1} I^{(i)} \qquad \text{and} \qquad \widehat{\gamma}_2 = \frac{1}{N_2}\sum_{i=1}^{N_1} \mathbb{E}\{I|X_k^{(i)}\} \tag{2.7}$$

are, respectively, unbiased estimators of the expected values of I and $\mathbb{E}\{I|X_k\}$ and, consequently, of γ. The first one is the *Crude Monte Carlo* estimator. The second one is a variant, from now on called *Conditional Monte Carlo* estimator. The variances of both are, respectively, $\mathbb{V}\{\widehat{\gamma}_1\} = \mathbb{V}\{I\}/N_1$ and $\mathbb{V}\{\widehat{\gamma}_2\} = \mathbb{V}\{\mathbb{E}\{I|X_k\}\}/N_2$. Therefore, due to expression (2.6), if $N_1 = N_2$, $\mathbb{V}\{\widehat{\gamma}_1\} \geq \mathbb{V}\{\widehat{\gamma}_2\}$, what means that $\widehat{\gamma}_2$ is more accurate than $\widehat{\gamma}_1$. A quantitative measure of the accuracy increase is the ratio $\mathbb{V}\{\widehat{\gamma}_1\}/\mathbb{V}\{\widehat{\gamma}_2\}$ or, if $N_1 = N_2$, $\mathbb{V}\{I\}/\mathbb{V}\{\mathbb{E}\{I|X_k\}\}$.

However, in order to decide which one of these estimators is more efficient, the variance comparison is not enough, it is also necessary to compare the computational efforts required to obtain the samples $I^{(i)}$ and $\mathbb{E}\{I|X_k^{(i)}\}$. To get a sample $I^{(i)}$ it is necessary to build a sample of $V^{(i)}$, what requires the sampling of $X_1^{(i)}$, $X_2^{(i)}$, $\cdots$, and $X_n^{(i)}$, and then, given the values of these components, to set $I^{(i)}$ to either 1 or 0. The generation of the samples $\mathbb{E}\{I|X_k^{(i)}\}$ may not always be easy. First, it is necessary to sample a value of X_k, and then to compute the expectation of I conditioned to it. This computation is the most critical and —sometimes— difficult step in the *Conditional Monte Carlo* estimation.

Given some measure of the trade-off between accuracy and computational effort (*speedup*, for instance), it is possible to determine which one of the estimators, $\widehat{\gamma}_1$ or $\widehat{\gamma}_2$, is more efficient.

CHAPTER 3

Variance Analysis of Splitting

Contents

Abstract

This chapter is devoted to two important characteristics of the *Splitting* estimator, namely, its bias and its variance. The determination of these values is not straightforward in the general case. However, under certain assumptions, unbiasedness can be easily proven and the variance can be given by a closed expression. In this chapter, *Fixed Splitting* and *Fixed Effort* are described in detail and they are analyzed for the particular case in which the estimators $\widehat{p}_i,\ i = 1, 2, \cdots, i$, are mutually independent. Under this assumption, both, *Fixed Splitting* and *Fixed Effort* estimators, are proved to be unbiased and their exact variances are determined.

3.1 Basic Fixed Splitting Setting

Consider a *Fixed Splitting* setting with m levels or *thresholds*, based on the ideas introduced in Section 2.4. A number of independent trajectories of a random process are started from *threshold* 0 and they are simulated until they either reach *threshold* 1 or they come back to down–cross *threshold* 0, in which case they are ended. The trajectories actually reaching *threshold* 1 are —all of them— split into a fixed number of new trajectories, and all of them are simulated until either they reach *threshold* 2 or down–cross *threshold* 0, in which case they are ended. The process continues iteratively until one or more of the trajectories started from *threshold* $m-1$ reach *threshold* m or down–cross *threshold* 0, in which case they are ended. The fact of having at least one trajectory up–crossing threshold m makes the *Splitting* estimation possible, as indicated in Section 2.4.

The following definitions apply to the setting under analysis (as the expression "*down–crossing* threshold 0" comes up all the time, in the following description it will be replaced by the term "*ending*"):

γ_i**:** the probability that a process trajectory up-crosses *threshold* i without *ending*.

p_i**:** the probability that a process trajectory up-crosses *threshold* i without *ending*, given that this trajectory has crossed *threshold* $i-1$ without *ending*.

n_1**:** the number of independent trajectories started from *threshold* 0 towards *threshold* 1.

n_i**:** the number of trajectories that every trajectory reaching *threshold* $i-1$ is split into.

F_i**:** the total number of independent trajectories started from *threshold* $i-1$ towards *threshold* i.

R_i**:** the total number of trajectories that up-cross *threshold* i without *ending*.

Due to the nesting of the consecutive up-crossing events, the value of γ_m —usually the magnitude of interest— is the following:

$$\gamma_m = p_1 p_2 \cdots p_{m-1} p_m. \tag{3.1}$$

However, as the probabilities $p_i,\ i = 1, 2, \cdots, m$ are unknown (otherwise the *Splitting* application would be unnecessary), they have to be estimated. Such estimations are achieved by means of a *Crude Monte Carlo* per level: $\widehat{p}_i = R_i/F_i\ \forall i = 1, 2, \cdots m$.

3.2 Fixed Splitting Estimation

The basic *Splitting* proposal is to estimate γ_m as:

$$\begin{aligned}
\widehat{\gamma}_m &= \widehat{p}_1 \widehat{p}_2 \cdots \widehat{p}_{m-1} \widehat{p}_m \\
&= \frac{R_1}{F_1}\frac{R_2}{F_2}\cdots\frac{R_m}{F_m} \\
&= \frac{R_1}{(n_1)}\frac{R_2}{(R_1 n_2)}\cdots\frac{R_m}{(R_{m-1} n_m)} \\
&= \frac{1}{n_1 n_2 \cdots n_m} R_m.
\end{aligned}$$

It will be shown that this estimation is unbiased.

Trajectories starting from *threshold* $i-1$ towards *threshold* i can be modelled by independent Bernoulli random variables with parameter p_i, with the following meaning: a value of 1 indicates that the trajectory reaches *threshold* i without *ending*, a value of 0 indicates that the trajectory *ends* before reaching *threshold* i. Figure 3.1 shows some trajectories of a *Splitting* replication in terms of the Bernoulli random variables and also shows the probabilities involved in the following calculation. The Bernoulli random variables are indexed in order to indicate the *threshold* they start from and also the trajectory they belong to. As their pattern is: $1_{i_1 i_2 \cdots i_m}$, its meaning follows clearly from Figure 3.1.

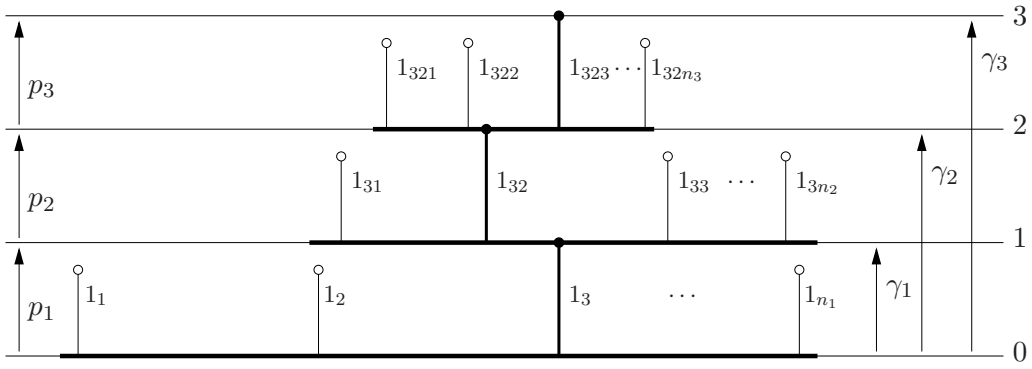


Figure 3.1: A "pictorial" view of the model in terms of the Bernoulli random variables

Given that in the case under analysis the random variables $\widehat{p}_i,\ i = 1, 2, \cdots, i$, are mutually independent, it is clear that the Bernoulli random variables $\mathtt{1}_{i_1}, \mathtt{1}_{i_1 i_2}, \cdots \mathtt{1}_{i_1 i_2 \cdots i_m}$, are also mutually independent.

Considering the independent n_1 trajectories starting from *threshold* 0 towards *threshold* 1, modelled by the Bernoulli variables $\mathtt{1}_{i_1},\ i_1 = 1, 2, \cdots, n_1$, with parameter p_1, the number R_1 of trajectories up-crossing *threshold* 1 without *ending* will be the number of these Bernoulli variables assuming a value of 1:

$$R_1 \ = \ \sum_{i_1=1}^{n_1} \mathtt{1}_{i_1}.$$

Each one of the successful R_1 trajectories is cloned or split into n_2 new trajectories towards *threshold* 2. Thus:

$$R_2 \ = \ \sum_{i_1=1}^{n_1} \mathtt{1}_{i_1} \left(\sum_{i_2=1}^{n_2} \mathtt{1}_{i_1 i_2} \right) \ = \ \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \mathtt{1}_{i_1} \mathtt{1}_{i_1 i_2}.$$

Finally:

$$\widehat{\gamma}_m \ = \ \frac{1}{n_1 n_2 \cdots n_m} R_m \ = \ \frac{1}{n_1 n_2 \cdots n_m} \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \cdots \sum_{i_m=1}^{n_m} \mathtt{1}_{i_1} \mathtt{1}_{i_1 i_2} \cdots \mathtt{1}_{i_1 i_2 \cdots i_m}. \tag{3.2}$$

Considering that the variables $\mathtt{1}_{i_1}, \mathtt{1}_{i_1 i_2}, \cdots \mathtt{1}_{i_1 i_2 \cdots i_m}$, are mutually independent and the fact that $\mathbb{E}\{\mathtt{1}_{i_1}\} = p_1$, $\mathbb{E}\{\mathtt{1}_{i_1 i_2}\} = p_2$, $\cdots$, and $\mathbb{E}\{\mathtt{1}_{i_1 i_2 \cdots i_m}\} = p_m$:

$$\begin{aligned} \mathbb{E}\{\widehat{\gamma}_m\} \ &= \ \frac{1}{n_1 n_2 \cdots n_m} \sum_{i_1=1}^{n_1} \sum_{i_2=1}^{n_2} \cdots \sum_{i_m=1}^{n_m} \mathbb{E}\{\mathtt{1}_{i_1} \mathtt{1}_{i_1 i_2} \cdots \mathtt{1}_{i_1 i_2 \cdots i_m}\} \\ &= \ \frac{1}{n_1 n_2 \cdots n_m} \, n_1 n_2 \cdots n_m \, p_1 p_2 \cdots p_m \end{aligned} \tag{3.3}$$

$$= \quad p_1 p_2 \cdots p_m \;=\; \gamma_m,$$

what proves that $\widehat{\gamma}_m$ is an unbiased estimator.

3.3 Variance of the Fixed Splitting Estimator

In this section the variance of the *Fixed Splitting* estimator is computed. The determination —adapted here to the model in which this section is based— is due to Glasserman et al. in [Glasserman 1996].

Let $\mathtt{S}_m$ be a sample of all the Bernoulli variables, up to the level m. Based on $\mathtt{S}_m$, the variance of an $m+1$ *threshold Splitting* estimator can be expressed as:

$$\begin{aligned}
\sigma^2_{m+1} &= \mathbb{V}\{\widehat{\gamma}_{m+1}\} \\
&= \mathbb{V}\{\mathbb{E}\{\widehat{\gamma}_{m+1}|\mathtt{S}_m\}\} + \mathbb{E}\{\mathbb{V}\{\widehat{\gamma}_{m+1}|\mathtt{S}_m\}\} \\
&= \mathbb{V}\left\{p_{m+1}\widehat{\gamma}_m\right\} + \mathbb{E}\{\mathbb{V}\{\widehat{\gamma}_m\widehat{p}_{m+1}|\mathtt{S}_m\}\} \\
&= p^2_{m+1}\sigma^2_m + \mathbb{E}\{\widehat{\gamma}^2_m\mathbb{V}\{\widehat{p}_{m+1}\}\} \\
&= p^2_{m+1}\sigma^2_m + \mathbb{E}\left\{\widehat{\gamma}^2_m \, \frac{p_{m+1}(1-p_{m+1})}{\left(\sum_{i_1=1}^{n_1} 1_{i_1} \sum_{i_2=1}^{n_2} 1_{i_1 i_2} \cdots \sum_{i_m=1}^{n_m} 1_{i_1 i_2 \cdots i_{m-1} i_m}\right) n_{m+1}}\right\} \\
&= p^2_{m+1}\sigma^2_m + \mathbb{E}\left\{\widehat{\gamma}^2_m \, \frac{p_{m+1}(1-p_{m+1})}{(\widehat{\gamma}_m n_1 n_2 \cdots n_m)\, n_{m+1}}\right\} \\
&= p^2_{m+1}\sigma^2_m + \mathbb{E}\left\{\widehat{\gamma}_m \, \frac{p_{m+1}(1-p_{m+1})}{n_1 n_2 \cdots n_m n_{m+1}}\right\} \\
&= p^2_{m+1}\sigma^2_m + \mathbb{E}\left\{\widehat{p}_1\widehat{p}_2\cdots\widehat{p}_m \, \frac{p_{m+1}(1-p_{m+1})}{n_1 n_2 \cdots n_m n_{m+1}}\right\} \\
&= p^2_{m+1}\sigma^2_m + \frac{p_{m+1}(1-p_{m+1})}{n_1 n_2 \cdots n_m n_{m+1}}\mathbb{E}\left\{\widehat{p}_1\widehat{p}_2\cdots\widehat{p}_m\right\} \\
&= p^2_{m+1}\sigma^2_m + \frac{p_1 p_2 \cdots p_m p_{m+1}(1-p_{m+1})}{n_1 n_2 \cdots n_m n_{m+1}}.
\end{aligned}$$

Solving this recurrence, the variance σ^2_m can be written as:

$$\begin{aligned}
\sigma^2_m &= \sum_{j=1}^{m}\left(\prod_{i=j+1}^{m} p_i^2\right)\frac{p_1 p_2 \cdots p_j(1-p_j)}{n_1 n_2 \cdots n_j} \\
\sigma^2_m &= (p_1 p_2 \cdots p_m)^2 \sum_{j=1}^{m} \frac{1-p_j}{(p_1 n_1)(p_2 n_2)\cdots(p_j n_j)} \\
&= \gamma^2_m \sum_{j=1}^{m} \frac{1-p_j}{(p_1 n_1)(p_2 n_2)\cdots(p_j n_j)}.
\end{aligned}$$

3.4 Variance of the Fixed Splitting Estimator (a different approach)

The following determination, starting from a standard variance expression, is a different way of developing the variance of the *Fixed Splitting* estimator.

$$\begin{aligned}
\mathbb{V}\{\widehat{\gamma}_m\} &= \mathbb{E}\{\widehat{\gamma}_m^2\} - \mathbb{E}\{\widehat{\gamma}_m\}^2 \\
&= \mathbb{E}\left\{\left(\frac{1}{n_1 n_2 \cdots n_m}\sum_{i_1=1}^{n_1}\sum_{i_2=1}^{n_2}\cdots\sum_{i_m=1}^{n_m} \mathtt{1}_{i_1}\mathtt{1}_{i_1 i_2}\cdots\mathtt{1}_{i_1 i_2 \cdots i_m}\right)^2\right\} - p_1^2 p_2^2 \cdots p_m^2 \\
&= \frac{1}{n_1^2 n_2^2 \cdots n_m^2}\mathbb{E}\left\{\left(\sum_{i_1=1}^{n_1}\sum_{i_2=1}^{n_2}\cdots\sum_{i_m=1}^{n_m} \mathtt{1}_{i_1}\cdots\mathtt{1}_{i_1 i_2 \cdots i_m}\right)^2\right\} - p_1^2 p_2^2 \cdots p_m^2. \qquad (3.4)
\end{aligned}$$

The square of a sum is composed by: the sum of the square of every one of its terms, plus two times the sum of the product of all the pairs of terms (considering only once pairs of the form (a, b) and (b, a)).

Using the fact that $\mathtt{1}_{i_1}, \mathtt{1}_{i_1 i_2}, \cdots \mathtt{1}_{i_1 i_2 \cdots i_m}$, are mutually independent plus the fact that, for a Bernoulli random variable $\mathtt{1}_x$ with parameter p_x, $\mathbb{E}\{\mathtt{1}_x^2\} = \mathbb{E}\{\mathtt{1}_x\} = p_x$, and using also expression (3.3), the expected value of the first part of the square of the sum reduces to:

$$\mathbb{E}\left\{\sum_{i_1=1}^{n_1}\sum_{i_2=1}^{n_2}\cdots\sum_{i_m=1}^{n_m}(\mathtt{1}_{i_1}\mathtt{1}_{i_1 i_2}\cdots\mathtt{1}_{i_1 i_2 \cdots i_m})^2\right\} = n_1 n_2 \cdots n_m p_1 p_2 \cdots p_m.$$

The remaining part requires a combinatorial analysis that can be better understood graphically. The target is to determine the mean of a sum, in which all the terms has the form:

$$\underbrace{(\mathtt{1}_{i_1}\mathtt{1}_{i_1 i_2}\cdots\mathtt{1}_{i_1 i_2 \cdots i_m})}_{(C_i)}\underbrace{(\mathtt{1}_{j_1}\mathtt{1}_{j_1 j_2}\cdots\mathtt{1}_{j_1 j_2 \cdots j_m})}_{(C_j)}$$

where

- both *chains* C_i and C_j are the product of a sequence of Bernoulli variables each one of them "pointing" to *thresholds* $0, 1, \cdots, m$.
- The elements of *chains* C_i and C_j must be completely different or:
  - may have the same variable in the first position, being different the rest of them,
  - may have the same variables in the first and second position, being different the rest of them,
  - $\vdots$
  - may have the same variables in the first, second, third, $\cdots$ position, being different only the last one.

A simple example with *thresholds* 1, 2, and 3 is shown in Figure 3.2. In this simple setting $n_1 = 6$, $n_2 = 3$ and $n_3 = 2$, being the corresponding probabilities p_1, p_2 and p_3. Figure 3.2

(a) shows that there is a number $n_1n_2n_3$ of *chains* modelling $n_1n_2n_3$ process trajectories that are likely to reach *threshold* 3. In fact, when any of the variables belonging to these *chains* take the value 0, the modelled trajectory immediately stops (i.e., it does not exist beyond this 0) and the variables that come after, are never sampled. However from the combinatorial point of view all the *chains* must be considered complete. The challenge is to count the number of all possible pairs of *chains*, accepting that the expected value of all these pairs are terms in the square of the sum in (3.4).

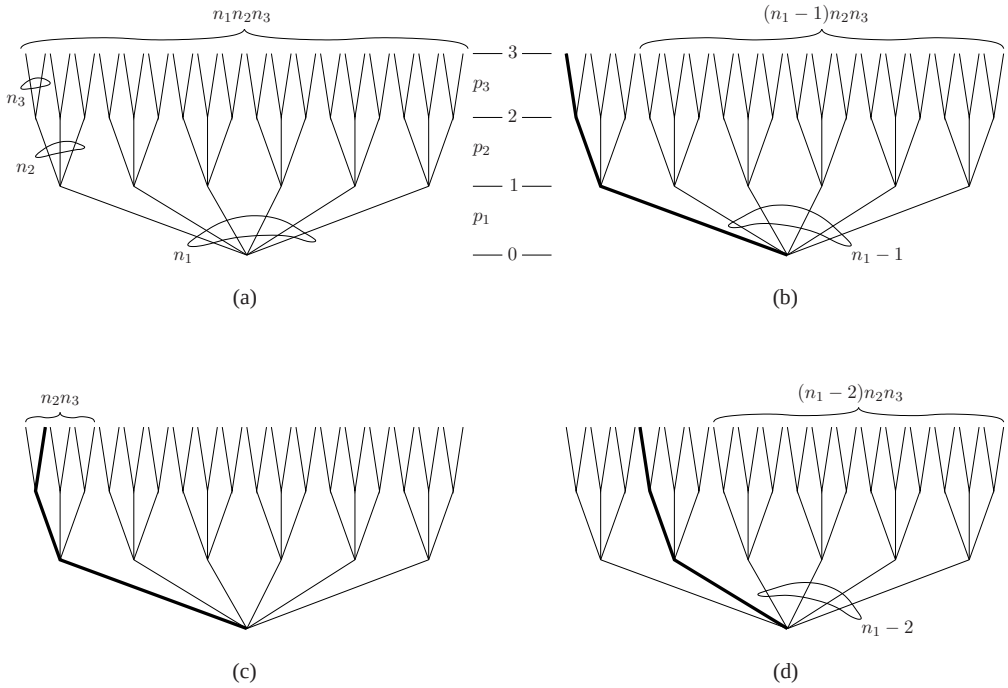


Figure 3.2: Bernoulli Variables in a Simple Three Thresholds Setting

Proceeding in stages:

1. *Chains* C_i and C_j that are completely different.
   Figure 3.2 (b) shows a *chain* C_i in thick line. There are $(n_1-1)n_2n_3$ possible *chains* C_j to complete the pair. Figure 3.2 (c) shows another C_i in thick line, for which it counts the same number $(n_1-1)n_2n_3$ of possible *chains* C_j. It is clear that there are n_2n_3 *chains* C_i (see Figure 3.2 (c)) for which the referred $(n_1-1)n_2n_3$ possible C_j complete the pair. Then, the number of pairs of completely different *chains* detected so far is:

$$n_2n_3(n_1-1)n_2n_3 = (n_1-1)n_2^2n_3^2.$$

   The expected value of each one of them is:

$$\mathbb{E}\{C_i\, C_j\} = \mathbb{E}\{\mathtt{1}_{i_1}\mathtt{1}_{i_1i_2}\mathtt{1}_{i_1i_2i_3}\ \mathtt{1}_{j_1}\mathtt{1}_{j_1j_2}\mathtt{1}_{j_1j_2j_3}\} = p_1^2p_2^2p_3^2$$

and therefore, the contribution of this set of pairs to the expression (3.4):

$$(n_1 - 1)n_2^2 n_3^2\, p_1^2 p_2^2 p_3^2 \cdot$$

However, starting the same counting process (as it has been done so far) from the thick line *chain* in Figure 3.2 (d), the number of pairs detected will be:

$$(n_1 - 2)n_2^2 n_3^2 \cdot$$

Finally, the number of pairs of completely different *chains* is:

$$(n_1 - 1) + (n_1 - 2) + \cdots + (\underbrace{n_1 - (n_1 - 1)}_{1}) \cdot$$

The total contribution of all of them to the expression (3.4) is:

$$[(n_1 - 1) + (n_1 - 2) + \cdots + (n_1 - (n_1 - 1))]n_2^2 n_3^2\, p_1^2 p_2^2 p_3^2 \cdot$$

Considering that $[(n_1-1)+(n_1-2)+\cdots+(n_1-(n_1-1))] = n_1(n_1-1)/2$, the expression reduces to:

$$\frac{n_1(n_1 - 1)n_2^2 n_3^2}{2}\, p_1^2 p_2^2 p_3^2 \cdot$$

2. *Chains* C_i and C_j that have the same variable in the first position, being different the rest of them.
   Based on the result obtained so far, this calculation is quite straightforward, because there will be as many identical sets as "first variables to share". In the example of Figure 3.2, there are $n_1 = 6$ identical sets, all of them composed by two *chains* having as the first element one of the $n_1 = 6$ variables in the lowest level. The remaining variables, i.e. the ones that start at *threshold* 1, deserve the same analysis that has been done in the previous item.

   The expected value of these pairs of *chains* is:

   $$\mathbb{E}\{C_i\, C_j\} = \mathbb{E}\{\mathtt{1}_{i_1}\mathtt{1}_{i_1 i_2}\mathtt{1}_{i_1 i_2 i_3}\ \mathtt{1}_{i_1}\mathtt{1}_{j_1 j_2}\mathtt{1}_{j_1 j_2 j_3}\} = \mathbb{E}\{\mathtt{1}_{i_1}^2\mathtt{1}_{i_1 i_2}\mathtt{1}_{i_1 i_2 i_3}\ \mathtt{1}_{j_1 j_2}\mathtt{1}_{j_1 j_2 j_3}\} = p_1 p_2^2 p_3^2,$$

   and the total number of them is:

   $$n_1 \left[\frac{n_2(n_2 - 1)n_3^2}{2}\right].$$

   Consequently, the contribution to the expression (3.4) of pairs that have the same variable in the first position and are different in the rest, is:

   $$\frac{n_1 n_2(n_2 - 1)n_3^2}{2}\, p_1 p_2^2 p_3^2 \cdot$$

3. *Chains* C_i and C_j that have the same variables in the first, second, $\cdots$ position, being different only the last one.

Considering that, for the current problem, the last variables are the ones that start from *threshold* 2, and proceeding the same way than in the prior item, the expected value of each one of these pairs of *chains* is:

$$\mathbb{E}\{C_i\, C_j\} = \mathbb{E}\{1_{i_1} 1_{i_1 i_2} 1_{i_1 i_2 i_3}\; 1_{i_1} 1_{i_1 i_2} 1_{j_1 j_2 j_3}\} = \mathbb{E}\{1_{i_1}^2\, 1_{i_1 i_2}^2 1_{i_1 i_2 i_3}\; 1_{j_1 j_2 j_3}\} = p_1 p_2 p_3^2.$$

The total number of them is:

$$n_1 n_2 \left[\frac{n_3(n_3-1)}{2}\right],$$

and the total contribution to the expression (3.4) of pairs that have the same variables in the first positions and are different only in the last one is:

$$\frac{n_1 n_2 n_3(n_3-1)}{2}\, p_1 p_2 p_3^2.$$

Based on the analysis of the simple example of Figure 3.2, it is possible to generalize the result for the case of m *thresholds* and for generic numbers $n_1, n_2, \cdots, n_m$:

$$\begin{aligned}
\mathbb{V}\{\widehat{\gamma}_m\} &= \frac{1}{n_1^2 n_2^2 \cdots n_m^2}\mathbb{E}\left\{\left(\sum_{i_1=1}^{n_1}\sum_{i_2=1}^{n_2}\cdots\sum_{i_m=1}^{n_m} 1_{i_1} 1_{i_1 i_2}\cdots 1_{i_1 i_2 \cdots i_m}\right)^2\right\} - p_1^2 p_2^2 \cdots p_m^2 \\
&= \frac{1}{n_1^2 n_2^2 \cdots n_m^2}\left\{n_1 n_2 \cdots n_m p_1 p_2 \cdots p_m + 2\left[\frac{n_1(n_1-1)n_2^2\cdots n_m^2}{2}\, p_1^2 p_2^2 \cdots p_m^2 + \right.\right. \\
&\quad \left.\left.\frac{n_1 n_2(n_2-1)\cdots n_m^2}{2}\, p_1 p_2^2 \cdots p_m^2 + \cdots + \frac{n_1 n_2 \cdots n_m(n_m-1)}{2}\, p_1 p_2 \cdots p_m^2\right]\right\} \\
&\quad - p_1^2 p_2^2 \cdots p_m^2 \\
&= \frac{p_1 p_2 \cdots p_m}{n_1 n_2 \cdots n_m}[1 + (n_1-1)n_2 n_3 \cdots n_m p_1 p_2 p_3 \cdots p_m + (n_2-1)n_3 \cdots n_m p_2 p_3 \cdots \\
&\quad \cdots p_m + (n_3-1)\cdots n_m p_3 \cdots p_m + \cdots + (n_m-1)p_m] - p_1^2 p_2^2 \cdots p_m^2 \\
&= \frac{p_1 p_2 \cdots p_m}{n_1 n_2 \cdots n_m}[(p_1 p_2 \cdots p_m)(n_1 n_2 \cdots n_m) + (1-p_1)(p_2 p_3 \cdots p_m)(n_2 n_3 \cdots n_m) + \\
&\quad \cdots + (1-p_{m-1})(p_m)(n_m) + (1-p_m)] - p_1^2 p_2^2 \cdots p_m^2 \\
&= \frac{p_1^2 p_2^2 \cdots p_m^2}{n_1 n_2 \cdots n_m}\left[\frac{(1-p_1)}{p1} n_2 n_3 \cdots n_m + \frac{(1-p_2)}{p1 p_2} n_3 \cdots n_m + \cdots + \frac{(1-p_{m-1})}{p_1 p_2 p_{m-1}} n_m \right. \\
&\quad \left. + \frac{(1-p_m)}{p_1 p_2 \cdots p_{m-1} p_m}\right] \\
&= p_1^2 p_2^2 \cdots p_m^2\left[\frac{(1-p_1)}{(p_1)(n_1)} + \frac{(1-p_2)}{(p_1 p_2)(n_1 n_2)} + \cdots + \frac{(1-p_{m-1})}{(p_1 p_2 \cdots p_{m-1})(n_1 n_2 \cdots n_{m-1})} \right. \\
&\quad \left. + \frac{(1-p_m)}{(p_1 p_2 \cdots p_m)(n_1 n_2 \cdots n_m)}\right]
\end{aligned}$$

$$
\begin{aligned}
&= p_1^2 p_2^2 \cdots p_m^2 \sum_{i=1}^{m} \frac{(1-p_i)}{(p_1 n_1)\cdots(p_i n_i)} \\
&= \gamma_m^2 \sum_{i=1}^{m} \frac{(1-p_i)}{(p_1 n_1)\cdots(p_i n_i)}.
\end{aligned}
$$

That is exactly the same expression obtained in the previous section.

3.5 Basic Fixed Effort Setting

Consider now a *Splitting* setting with m levels or *thresholds*, where γ_i, p_i, n_1, n_i, F_i and R_i, are the same as defined in Section 3.1.

As in *Fixed Splitting*, there is a value of n_i per level in *Fixed Effort*. But now the effort per level, $F_i = n_i \times R_{i-1}$, is a fixed value. Thus, n_i is a random variable whose values have to be adjusted so as to let F_i be fixed.

The magnitude of γ_m can be calculated just like in (3.1):

$$\gamma_m = p_1 p_2 \cdots p_{m-1} p_m, \tag{3.5}$$

where the probabilities $p_i,\ i = 1, 2, \cdots, m$ can be estimated by *Crude Monte Carlo* per level as: $\widehat{p}_i = R_i/F_i\ \forall i = 1, 2, \cdots m$.

3.6 Fixed Effort Estimation

The basic *Splitting* proposal is to estimate γ_m as:

$$
\begin{aligned}
\widehat{\gamma}_m &= \widehat{p}_1 \widehat{p}_2 \cdots \widehat{p}_{m-1} \widehat{p}_m \\
&= \frac{R_1 R_2 \cdots R_m}{F_1 F_2 \cdots F_m}.
\end{aligned}
$$

It will be shown that this estimation is unbiased.

Trajectories starting from *threshold* $i-1$ towards *threshold* i can be thought of as independent Bernoulli random variables with parameter p_i, just like in Section 3.1.

Consider F_1 independent trajectories starting from *threshold* 0 towards *threshold* 1, modelled by the Bernoulli variables $1_{1i},\ i = 1, 2, \cdots, F_1$, with parameter p_1. Now all the Bernoulli random variables starting at *threshold* 1 are considered as a set, independently of the starting point. Therefore, the subscript '$1i$' has two parts: a number —in this case, number 1— to indicate that the referred variable models a trajectory that goes from *threshold* 0 to *threshold* 1, and the index $i,\ i = 1, 2, \cdots, F_1$. The same nomenclature applies for the following *thresholds*. The number R_1 of trajectories up-crossing *threshold* 1 without *ending* will be the number of these Bernoulli variables assuming a value of 1:

$$R_1 = \sum_{i=1}^{F_1} 1_{1i}.$$

Accepting that at least one of these trajectories actually reaches *threshold* 1 (i.e., accepting $R_1 \geq 1$), the successful R_1 trajectories will be cloned into n_2 trajectories each, starting from *threshold* 1 towards *threshold* 2. The value of n_2 is adjusted so as to let $F_2 = n_2 \times R_1$. Then, the number of trajectories up-crossing *threshold* 2 will be:

$$R_2 \;=\; \sum_{i=1}^{F_2} \mathtt{1}_{2i},$$

under the same subscript convention.

Finally:

$$\begin{aligned}
\widehat{\gamma}_m &= \frac{R_1 R_2 \cdots R_m}{F_1 F_2 \cdots F_m} \\
&= \frac{1}{F_1 F_2 \cdots F_m} \sum_{i=1}^{F_1} \mathtt{1}_{1i} \sum_{i=1}^{F_2} \mathtt{1}_{2i} \cdots \sum_{i=1}^{F_m} \mathtt{1}_{mi} \\
&= \frac{1}{F_1 F_2 \cdots F_m} \sum_{i=1}^{F_1} \sum_{i=1}^{F_2} \cdots \sum_{i=1}^{F_m} \mathtt{1}_{1i} \mathtt{1}_{2i} \cdots \mathtt{1}_{mi}.
\end{aligned}$$

Considering that the variables $\mathtt{1}_{1i}, \mathtt{1}_{2i}, \cdots, \mathtt{1}_{mi}$, are mutually independent and also the fact that $\mathbb{E}\{\mathtt{1}_{1i}\} = p_1$, $\mathbb{E}\{\mathtt{1}_{2i}\} = p_2$, $\cdots$, and $\mathbb{E}\{\mathtt{1}_{mi}\} = p_m$:

$$\begin{aligned}
\mathbb{E}\{\widehat{\gamma}_m\} &= \frac{1}{F_1 F_2 \cdots F_m} \sum_{i=1}^{F_1} \sum_{i=1}^{F_2} \cdots \sum_{i=1}^{F_m} \mathbb{E}\{\mathtt{1}_{1i} \mathtt{1}_{2i} \cdots \mathtt{1}_{mi}\} \\
&= \frac{1}{F_1 F_2 \cdots F_m} F_1 F_2 \cdots F_m \, p_1 p_2 \cdots p_m \\
&= p_1 p_2 \cdots p_m.
\end{aligned}$$

3.7 Variance of the Fixed Effort Estimator

The following variance determination for the *Fixed Effort* estimator has been developed, as part of this thesis, following similar steps as in the determination of Glasserman et al. in [Glasserman 1996], but introducing the necessary changes in the model to let it perform as *Fixed Effort* instead of *Fixed Splitting*.

$$\begin{aligned}
\sigma^2_{m+1} &= \mathbb{V}\{\widehat{\gamma}_{m+1}\} \\
&= \mathbb{V}\{\mathbb{E}\{\widehat{\gamma}_{m+1} | \mathtt{S}_m\}\} + \mathbb{E}\{\mathbb{V}\{\widehat{\gamma}_{m+1} | \mathtt{S}_m\}\} \\
&= \mathbb{V}\{p_{m+1} \widehat{\gamma}_m\} + \mathbb{E}\{\mathbb{V}\{\widehat{\gamma}_m \widehat{p}_{m+1} | \mathtt{S}_m\}\} \\
&= p^2_{m+1} \sigma^2_m + \mathbb{E}\{\widehat{\gamma}^2_m \mathbb{V}\{\widehat{p}_{m+1}\}\} \\
&= p^2_{m+1} \sigma^2_m + \mathbb{E}\left\{\widehat{\gamma}^2_m \, \frac{p_{m+1}(1 - p_{m+1})}{F_{m+1}}\right\}
\end{aligned}$$

$$
\begin{aligned}
&= p_{m+1}^2\sigma_m^2 + \frac{p_{m+1}(1-p_{m+1})}{F_{m+1}}\mathbb{E}\left\{\widehat{\gamma}_m^2\right\} \\
&= p_{m+1}^2\sigma_m^2 + \frac{p_{m+1}(1-p_{m+1})}{F_{m+1}}(\sigma_m^2+\gamma_m^2) \\
&= \left[p_{m+1}^2 + \frac{p_{m+1}(1-p_{m+1})}{F_{m+1}}\right]\sigma_m^2 + \frac{p_{m+1}(1-p_{m+1})}{F_{m+1}}\gamma_m^2 \\
&= \left[p_{m+1}^2 + \frac{p_{m+1}(1-p_{m+1})}{F_{m+1}}\right]\sigma_m^2 + \frac{p_{m+1}(1-p_{m+1})}{F_{m+1}}\frac{\gamma_{m+1}^2}{p_{m+1}^2} \\
&= \left(p_{m+1}^2 + V_{m+1}\right)\sigma_m^2 + V_{m+1}\frac{\gamma_{m+1}^2}{p_{m+1}^2} \\
&= p_{m+1}^2\left(1+E_{m+1}^2\right)\sigma_m^2 + E_{m+1}^2\gamma_{m+1}^2, \qquad (3.6)
\end{aligned}
$$

where $V_{m+1} = p_{m+1}(1-p_{m+1})/F_{m+1}$ is the variance of the estimator $\widehat{p}_{m+1}$ and $E_{m+1}^2 = V_{m+1}/p_{m+1}^2$ is the square of the *relative error* (*coefficient of variation*) at level $m+1$.

Solving the recurrence in expression (3.6), the variance σ_m^2 can be written as:

$$
\begin{aligned}
\sigma_m^2 &= \gamma_m^2\sum_{j=1}^{m}E_j^2\prod_{i=j+1}^{m}(1+E_i^2) \qquad (3.7) \\
&= \gamma_m^2\sum_{j=1}^{m}\frac{(1-p_j)}{p_jF_j}\prod_{i=j+1}^{m}\left(1+\frac{(1-p_i)}{p_iF_i}\right). \qquad (3.8)
\end{aligned}
$$

3.8 Variance of the Fixed Effort Estimator (a different approach)

This variance determination for the *Fixed Effort Splitting* estimator, is a proposal of this thesis. Given that the number of trajectories started from every *threshold* is fixed, the variances of the individual estimators, $\widehat{p}_i,\ i = 1, 2, \cdots, m$, have a compact expression: $\mathbb{V}\{\widehat{p}_i\} = p_i(1-p_i)/F_i$. There is a formula due to Goodman in [Goodman 1962] that allows to calculate the variance of the product of independent random variables, in terms of the individual variances of the terms that are multiplied. This formula is, therefore, useful in this case. The variance of the *Splitting* estimator, as the product of its individual components, is the following:

$$
\mathbb{V}\{\widehat{\gamma}_m\} = \mathbb{V}\{\widehat{p}_1\widehat{p}_2\cdots\widehat{p}_m\}.
$$

The formula proposed in [Goodman 1962], applied to this product, considering also that $p_i = \mathbb{E}\{\widehat{p}_i\}$ and $E_i^2 = \mathbb{V}\{\widehat{p}_i\}/\mathbb{E}\{\widehat{p}_i\}^2 = (1-p_i)/(p_iF_i)$, becomes:

$$
\begin{aligned}
\mathbb{V}\{\widehat{\gamma}_m\} &= \prod_{i=1}^{m}p_i^2\left(\sum_{i_1}E_{i_1}^2 + \sum_{i_1<i_2}E_{i_1}^2E_{i_2}^2 + \sum_{i_1<i_2<i_3}E_{i_1}^2E_{i_2}^2E_{i_2}^2 + \cdots + E_1^2E_2^2E_3^2\cdots E_m^2\right) \\
&= \gamma_m^2\left[E_1^2\left(1+\sum_{1<i_1}E_{i_1}^2 + \sum_{1<i_1<i_2}E_{i_1}^2E_{i_2}^2 + \cdots + E_2^2E_3^2\cdots E_m^2\right) + E_2^2\left(1+\right.\right.
\end{aligned}
$$

$$\left. \sum_{2<i_1} E_{i_1}^2 + \sum_{2<i_1<i_2} E_{i_1}^2 E_{i_2}^2 + \cdots + E_3^2 E_4^2 \cdots E_m^2 \right) + \cdots + E_{m-1}^2 \left(1 + E_m^2\right) + E_m^2 \Bigg]$$

$$\begin{aligned}
&= \gamma_m^2 \left[E_1^2 (1+E_2)(1+E_3)\cdots(1+E_m) + E_2^2 (1+E_3)(1+E_4)\cdots(1+E_m) + \right. \\
&\quad \left. E_{m-1}^2 (1+E_m^2) + E_m^2 \right] \\
&= \gamma_m^2 \sum_{j=1}^{m} E_j^2 \prod_{i=j+1}^{m} (1+E_i^2) \\
&= \gamma_m^2 \sum_{j=1}^{m} \frac{(1-p_j)}{p_j F_j} \prod_{i=j+1}^{m} \left(1 + \frac{(1-p_i)}{p_i F_i}\right).
\end{aligned}$$

That is exactly the same expression obtained in the previous section.

CHAPTER 4

Splitting on Network Reliability Estimation

Contents

Abstract

This chapter introduces an original proposal of this thesis in which *Splitting* is applied to improve a well–known method called *Creation Process*, used in network *reliability* estimation. The resulting proposal, called here *Splitting/CP*, is particularly appropriate in the case of highly reliable networks, i.e. networks for which failure is a rare event. The chapter introduces the basis of *Splitting/CP* and presents a set of computational experiments based on network topologies taken from the literature. The results of these experiments show that *Splitting/CP* is accurate, efficient and robust, being therefore a valid alternative to the best known methods used in network *reliability* estimation.

4.1 Introduction

A network model based on a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes and $\mathcal{E}$ the set of links, augmented with a stochastic behaviour representing the probabilistic structure of failures in nodes and links, is suitable for network *reliability* analysis. In the model used in this chapter nodes never

fail, they are in an *operational* state all the time, while links fail independently and can only be found in one of two states, *operational* or *failed*. Based on this model, the network *reliability* R (unreliability Q) is defined as the probability that a set of terminal nodes $\mathcal{K} \subseteq \mathcal{V}$ is connected (not connected) in the sub–graph containing all the nodes in $\mathcal{V}$ but only the *operational* links (as for a link being *failed* is the same as being removed from $\mathcal{G}$).

This model, although very simple, has been employed in a wide number of application settings. Among other cases, many examples can be found in the evaluation and the topology design of communication networks, mobile ad hoc and tactical radio networks, evaluation of transport and road networks, etc. [Cook 2007, Günnec 2007, Li 2004, Lin 2006, Marotta 2010, Marseguerra 2005, Taboada 2008].

The exact computation of either R or Q is an NP–hard problem [Provan 1983], so that for large networks their values can only be estimated. *Monte Carlo* simulation methods are then often used to analyze these models. However, as shown in previous chapters, when Q is extremely small, the accuracy of *Standard Monte Carlo* methods collapses.

If $\widehat{Q}$ is an unbiased *Monte Carlo* estimator of Q, its *relative error* can be defined as $\mathbb{V}\{\widehat{Q}\}^{1/2}/\mathbb{E}\{\widehat{Q}\}$, expression known as the *coefficient of variation*, also called square root of the normalized variance. This *relative error* —the measure of accuracy in this context— grows, sometimes boundlessly, when the network *unreliability* Q goes to zero.

Much research has been focused on reducing the variance of *Monte Carlo* estimators of Q in order to reduce their *relative error* [Cancela 1995, Cancela 2003, Cancela 2008, Easton 1980, Elperin 1991, Fishman 1986, Hui 2003, Hui 2005, Karp 1983, Kumamoto 1977, Kumamoto 1980, Lomonosov 1994, Ross 1994]. A complete review of these methods can be seen in [Cancela 2009].

Splitting is a variance reduction technique that successfully increases the accuracy of *Monte Carlo* methods in *rare event* probability estimation [Garvels 2000, Glasserman 1996, L'Ecuyer 2007a, Villén-Altamirano 1991]. Although *Splitting* has been much used in performance and performability analysis, it has been scarcely applied for simulating highly reliable systems. Some applications in this context are due to [Villén-Altamirano 2007] and [Villén-Altamirano 2010], where the *reliability* and availability estimations of repairable systems are analysed. Other recent results on this subject have been published by [Botev 2010] (which tackles some static systems) and [Kroese 2013]. In the present work, *Splitting* is adapted to estimate static network *reliability* measures. This application of *Splitting* permits to increase the accuracy of a *Monte Carlo* method based on the so–called *Creation Process* [Elperin 1991].

The improvement achieved by the application of *Splitting* to the *Creation Process* also affects the computational efficiency, making it possible to deal with the estimation of extremely small unreliabilities in a reasonable time. Actually, the resulting method, called in this thesis *Splitting/CP*, performs in the order of the best–known algorithms proposed for the estimation of the *unreliability* of highly reliable networks. Another feature that has been subject to empirical analysis in this work is the high degree of robustness of *Splitting/CP* when the network *unreliability* varies from high or moderate values down to extremely low values.

The ideas presented in this chapter have resulted in the following publications: [Murray 2008a, Murray 2008b, Murray 2008c, Murray 2010, Murray 2013b].

The rest of the chapter is organized as follows. Section 4.2 introduces network models and

gives an outline of the *Creation Process*. Section 4.3 presents the application of *Splitting* to the *Creation Process*. Implementation details and experimental results are given in Sections 4.4, 4.5, 4.6 and 4.7. Section 4.8 shows an empirical analysis of the robustness of the resulting method. Conclusions and some ideas for the future work can be found at the end of this thesis, in Section 7.2.

4.2 Network Reliability Modelling

An undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is a set of v absolutely reliable vertices or nodes and $\mathcal{E}$ a set of e independent unreliable edges or links, is a frequently used model in communication network *reliability* analysis. The state of the links is modelled by independent binary random variables $X_i,\ i = 1, 2, \cdots, e$, such that: $X_i = 1$ when the i^{th} link is *operational* and $X_i = 0$ when the i^{th} link is *failed*. For a link to be *failed* means "to be removed from $\mathcal{G}$", whereas to be *operational* means "to perfectly perform the tasks it was committed to". Hence, $r_i = \mathbb{P}\{X_i = 1\}$ is the single link *reliability* and $q_i = 1 - r_i = \mathbb{P}\{X_i = 0\}$ is the single link *unreliability*. The state of all the links is modelled by the vector $\mathbf{X} = (X_1, X_2, \cdots, X_e)$.

The network is *operational* (*failed*) when some *structure function* $\phi(\mathbf{X})$ equals 1 (equals 0). The network *reliability* is defined as $R = \mathbb{P}\{\phi(\mathbf{X}) = 1\}$, and the *unreliability* as $Q = \mathbb{P}\{\phi(\mathbf{X}) = 0\}$. Typically, the network is considered *operational* and, as a consequence, $\phi(\mathbf{X}) = 1$, when some subset of nodes $\mathcal{K} \subseteq \mathcal{V}$ is connected in the sub–graph containing all the nodes in $\mathcal{V}$ but only the *operational* links. This definition leads to the concept of $\mathcal{K}$–terminal connectivity, that includes two special cases: *s–t* connectivity, for the case where $\mathcal{K} = \{s, t\}$, being s and t two nodes in $\mathcal{V}$, and all–terminal connectivity, when $\mathcal{K} = \mathcal{V}$.

Crude Monte Carlo estimations of network *reliability* and *unreliability* can be computed, respectively, by the unbiased estimators:

$$\widehat{R} = \frac{1}{N}\sum_{i=1}^{N}\phi(\mathbf{X}^{(i)}) \quad \text{and} \quad \widehat{Q} = \frac{1}{N}\sum_{i=1}^{N}(1 - \phi(\mathbf{X}^{(i)})),$$

where $\mathbf{X}^{(i)},\ i = 1, \cdots, N$ are independent samples of $\mathbf{X}$.

As a consequence of, say, changes in the single reliabilities r_i, due to which the network becomes more reliable, the *relative error* $\mathbb{V}\{\widehat{Q}\}^{1/2}/\mathbb{E}\{\widehat{Q}\}$ grows because $\mathbb{E}\{\widehat{Q}\}$ goes to 0 faster than $\mathbb{V}\{\widehat{Q}\}^{1/2}$ (observe this in a *Crude Monte Carlo* estimation: $\mathbb{V}\{\widehat{Q}\}^{1/2} = (Q(1-Q)/N)^{1/2}$ and $\mathbb{E}\{\widehat{Q}\} = Q$). In the case of highly reliable networks, accurate estimations require a very small value of $\mathbb{V}\{\widehat{Q}\}$ that can only be achieved with a very large sample size, N.

In the *Creation Process* [Elperin 1991] the link states are supposed to evolve in time, being all of them *failed* at time $t = 0$, and becoming *operational* at times $\tau_i,\ i = 1, 2, \cdots, e$, exponentially distributed with parameters λ_i. The network then turns into a stochastic dynamic graph $\mathcal{G}(t) = (\mathcal{V}, \mathcal{F}(t))$, $\mathcal{F}(t) \subseteq \mathcal{E}$, $t \geq 0$, corresponding to the stochastic process $\{\mathbf{X}(t) = (X_1(t), X_2(t), \cdots, X_e(t)), t \geq 0\}$. In this model $X_i(t) = 1$ (i^{th} link *operational*) if $t \geq \tau_i$ and $X_i(t) = 0$ (i^{th} link *failed*) if $t < \tau_i$. Times τ_i are called "repair times". Since the probability that the i^{th} link becomes *operational* at time t or earlier is $\mathbb{P}\{\tau_i \leq t\} = 1 - e^{-\lambda_i t}$, the choice of $\lambda_i = -\ln(q_i)$ makes the probability that the i^{th} link is *operational* at $t = 1$ be exactly

the single link *reliability* r_i. As a consequence, the probability that the network is *operational* (resp. *failed*) at $t = 1$ is R (resp. Q). In symbols, $R = \mathbb{E}\{\phi(\mathbf{X}(1))\}$ and $Q = \mathbb{E}\{1 - \phi(\mathbf{X}(1))\}$.

If the repair times are arranged as a sequence, they can be seen as the trajectory of a stochastic process. This approach makes the *Creation Process* subject to the application of *Splitting*, as it will be introduced in the following section.

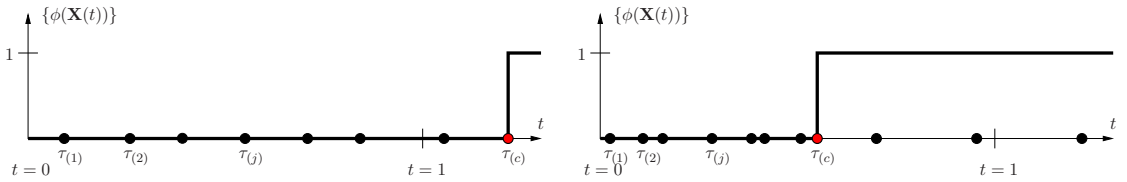
4.3 Splitting/CP

At every single replication of the *Creation Process*, there is one exponentially distributed repair time per link. Once these times are sampled, they may be arranged as a sequence $\mathcal{T} = \{\tau_{(1)}, \cdots, \tau_{(j)}, \cdots, \tau_{(e)}\}$, where $\tau_{(1)} \leq \cdots \tau_{(j)} \leq \cdots \tau_{(e)}$ ($\mathcal{T}$ is the *order statistics* of $\{\tau_1, \cdots, \tau_e\}$). However, to make sequences $\mathcal{T}$ subject to the application of *Splitting*, it is better to sample the times $\tau_{(i)}$ according to the well–known sampling method that will be described next.

Let $\lambda(\tau_{(i)})$ be the parameter of the Exponential random variable from which the time $\tau_{(i)}$ has been sampled (a random variable). Let $\mathcal{T}^i = \{\tau_{(1)}, \tau_{(2)}, \cdots, \tau_{(i)}\}$, $1 \leq i \leq e$, with $\mathcal{T}^0 = \emptyset$, be a partially sampled sequence, and $\Lambda^i = \{\lambda(\tau_{(1)}), \lambda(\tau_{(2)}), \cdots, \lambda(\tau_{(i)})\}$ the parameters associated with the repair times in $\mathcal{T}^i$ (thus, $\Lambda^e = \{\lambda_1, \cdots, \lambda_e\}$). Let $\bar{\Lambda}^i$ be the set of parameters that are not in Λ^i, i.e. the parameters of the links whose repair time has not been sampled yet. Calling $S^i = \sum_{j:j\leq i} \lambda(\tau_{(j)})$ and $\bar{S}^i = \sum_{j:j>i} \lambda(\tau_{(j)})$, then, once $\mathcal{T}^i$ is already sampled:

- $\tau_{(i+1)} = \tau_{(i)} + \Delta$, where Δ can be sampled from an Exponential random variable with parameter $\bar{S}^i$.
- A link with parameter λ_j, whose repair time has not been sampled yet, has a probability $\lambda_j/\bar{S}^i$ to be the next one in the sequence and can, therefore, be sampled accordingly.

Every sequence $\mathcal{T}$ determines a path of the stochastic process $\{\phi(\mathbf{X}(t))\}$. Two replications of $\{\phi(\mathbf{X}(t))\}$ are shown in Figures 4.1(a) and 4.1(b). In both of them there is a particular repair time $\tau_{(c)}$ such that $\phi(\mathbf{X}(t)) = 0$ if $t < \tau_{(c)}$ and $\phi(\mathbf{X}(t)) = 1$ if $t \geq \tau_{(c)}$.



(a) (b)

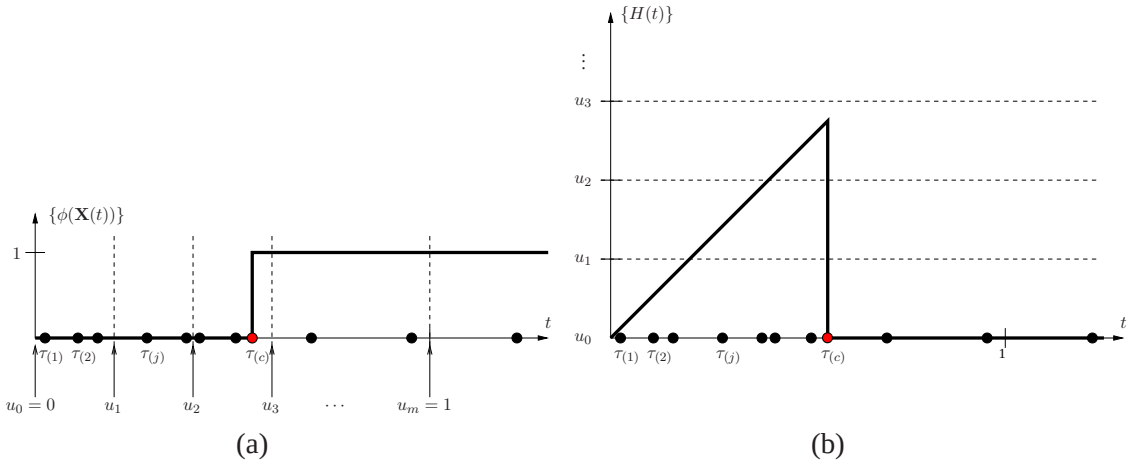
Figure 4.1: Two replications of the stochastic process $\{\phi(\mathbf{X}(t))\}$ as a function of a sequence $\mathcal{T}$

Event $E = \{\phi(\mathbf{X}(1)) = 0\}$ occurs if the network becomes *operational* after $t = 1$, therefore $Q = \mathbb{P}\{E\}$.

In the basic simulation of the *Creation Process*, the estimator $\widehat{Q}$ is the ratio between the number of successful events E and the total number of replications or, equivalently, the ratio between the number of sequences for which $\tau_{(c)} \geq 1$ and the total number of sampled sequences.

A proposal of this thesis is to see the sequences $\mathcal{T} = \{\tau_{(1)}, \tau_{(2)}, \cdots, \tau_{(e)}\}$ as the replications of a random process, and to apply *Splitting* to the simulation of this process. To do this, the interval $[0, 1]$ has to be partitioned by a set of *thresholds* $u_0 = 0 < u_1 < u_2 < \cdots < u_m = 1$, as shown in Figure 4.2(a). This partition defines $E_i = \{\phi(\mathbf{X}(u_i)) = 0\},\ k = 1, 2, \cdots, m$, as the indicator events that the network is still *failed* at $t = u_i$. In view of the ideas of Section 2.4, $Q = \mathbb{P}\{E_m\}$ and $\widehat{Q} = \prod_{i=1}^{m} \widehat{p}_i$. The estimators $\widehat{p}_i$ can be obtained separately, according to the following mechanism: start one or more sequences $\mathcal{T}$ from $t = 0$ and then (i) cancel the ones for which event E_1 does not occur and (ii) split the ones for which event E_1 occurs. Proceed the same way with all the new sequences started from u_1, i.e. cancel or split at *threshold* u_2. Keep repeating the mechanism until *threshold* $u_m = 1$ is reached. Finally, $\widehat{p}_i = R_i/N_{i-1},\ k = 1, 2, \cdots, m$, where R_i is the number of sequences crossing *threshold* u_i and N_{i-1} the total number of sequences actually launched from *threshold* u_{i-1}. The resulting estimator is $\widehat{Q} = R_1/N_0\ R_2/N_1\ R_3/N_2\ \cdots\ R_m/N_{m-1}$.

The preceding ideas are the basis of the proposed method, called here *Splitting/CP* (Splitting on the *Creation Process*). *Thresholds* are intuitively defined in terms of time, while most *Splitting* methods in the literature define them in terms of a level function over the random process's state space. However, a function like $\{H(t) = t\, I_{\{\phi(\mathbf{X}(t))=0\}}, t \geq 0\}$, where $I_{\{e\}} = 1$ if event e occurs and $I_{\{e\}} = 0$ otherwise, can be used to see that there is a formal equivalence between both approaches.



(a) (b)

Figure 4.2: The Same Replication on the Stochastic Processes $\{\phi(\mathbf{X}(t))\}$ and $\{H(t)\}$

This function maps the space of sequences $\mathcal{T} = \{\tau_{(1)}, \cdots, \tau_{(j)}, \cdots, \tau_{(e)}\}$ into the state space of the random process $\{H(t)\}$ (see Figure 4.2(b)). The estimators $\widehat{Q}$ obtained in both cases, either directly on $\{\phi(\mathbf{X}(t))\}$ or by the application of *Splitting* on $\{H(t)\}$, are identical.

4.4 Implementation Guidelines

An important issue in *Splitting/CP* is how to split trajectories every time a *threshold* is crossed. Suppose that $\mathcal{T} = \{\tau_{(1)}, \cdots, \tau_{(x)}, \tau_{(x+1)}\}$ is *under construction*, with $\tau_{(x)} < u_i < \tau_{(x+1)}$ (recall that $\tau_{(x+1)} = \tau_{(x)} + \Delta$, where Δ is a sample of an Exponential random variable with parameter

$\bar{S}^{x+1}$). Suppose also that the network is still *failed* at time $\tau_{(x+1)}$, meaning that event E_i has actually occurred. Trajectory $\mathcal{T}$ must therefore be split at $t = u_i$. New statistically equivalent values $\tau_{(y)}$ (as many as necessary) can be sampled in place of $\tau_{(x+1)}$. That is $\tau_{(y)} = u_i + \Delta$, sampling a new value of Δ, and a new repaired link, every time. Then, the values of $\tau_{(y)}$ define new trajectories starting from *threshold* u_i.

A problem arising in many *Splitting* implementations is the considerable computational effort necessary to simulate the process since any *threshold* is crossed until the trajectory eventually "dies". In the most general case, after any *threshold* cross, the process may follow an *up* and *down* evolution before the final condition is reached (i.e. before it falls below ℓ_0, in terms of Section 2.4). In *Splitting/CP* this problem does not arise. The only wasted effort for any "dying" trajectory is the one devoted to take it closer to the next *threshold* but, if the next *threshold* is not reached, the trajectory is just discarded, with no additional effort. This is shown in Figure 4.2(a). After process $\{\phi(\mathbf{X}(t))\}$ crosses *threshold* u_2, three more links are sampled, the third of which in time $\tau_{(c)}$. If the third link would have been any other one, *threshold* u_3 would have been crossed. Therefore, with a computational effort of three (possibly a few more) sampled links, *threshold* u_3 could have been crossed, whereas after a computational effort of exactly three sampled links, the trajectory immediately stops and no additional effort is needed.

Efficiency and accuracy of *Splitting* depend on the number of *thresholds*, m. But the optimal value of m is hard to find and, therefore, the choice of this number is not straightforward. Actually, there is no general procedure to set the value of m, only some recommendations and guidelines derived from the analysis of some specific models are available. One of these recommendations comes from [Villén-Altamirano 2002] who proved that, in the RESTART variant, the optimal value of m is $(\ln Q)/(\ln 0.5) - 1$. Other contributions on this issue are due to [Garvels 2000] and [L'Ecuyer 2007a] who, after the analysis of very simple *Fixed Effort* settings, concluded that $m = -(\ln Q)/2$ maximizes the efficiency of the *Splitting* estimator.

Even when the models used by [Villén-Altamirano 2002], [Garvels 2000] and [L'Ecuyer 2007a] do not entirely match the *Splitting/CP* model, the values of m that they proposed can be used as a starting point in a set of iterations or pilot runs, in order to find the most appropriate value of m in a *Splitting/CP* simulation. The lack of a value of Q for the very first pilot run can be solved using, in place, an upper bound (Q_U), a lower bound (Q_L) or any value Q_i between these bounds ($Q_L < Q_i < Q_U$). Upper and lower bounds on Q can usually be found by means of several network models analysis. Once the first pilot run is done, an estimation of Q is available for the next pilot runs.

[Villén-Altamirano 2006] showed that the RESTART variant is very robust against changes in the values of m. Given this conclusion, the selection of m does not seem to be worth much effort, because any value out of a wide set should work as well. This fact was experimentally verified during the development of Section 4.5 where, after a few pilot runs in every experiment, a suitable value of m was found. It is interesting to remark that, in most of the experiments, the value of m finally selected was quite close to the expression $(\ln Q)/(\ln 0.5) - 1$ proposed in [Villén-Altamirano 2002].

4.5 Experimental Setting

A set of benchmark network topologies to be used in the experimental phase of this chapter is next introduced. These topologies have been widely used in the network *reliability* literature for computational studies and benchmarking purposes, and they have been chosen in order to support direct comparison of the behaviour of *Splitting/CP* with results from other papers.

The referred network topologies are the following:

Easton–Wong Network, shown in Figure 4.3. This network is composed by three types of links: Horizontal (q_H), Vertical (q_V) and Diagonal (q_D). Two versions of it were used: EW-1 with single link unreliabilities $q_H = 0.02$, $q_V = 0.01$ and $q_D = 0.001$, and EW-2 with $q_H = 0.005$, $q_V = 0.01$ and $q_D = 0.0005$, both of them performing the all–terminal connectivity model.

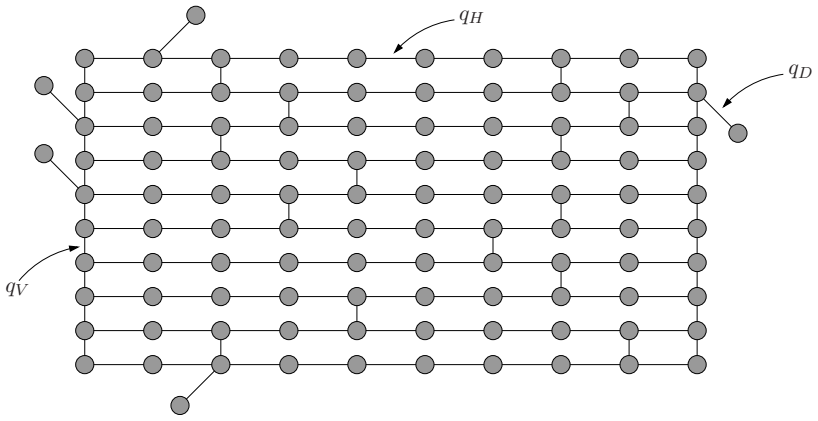


Figure 4.3: Easton–Wong Network

Dodecahedron Network, shown in Figure 4.4. This topology was used in two different versions, one of them with equi–reliable links and the other one with two types of links. In the latter case some of the links (dashed lined, depicting the minimum spanning tree) have a "low" *unreliability* q_L, and the rest (solid lined, resembling wireless backup links) have a slightly "higher" *unreliability* q_H. Based on this network, the *s–t* and the all–terminal connectivity models were implemented (the dark color nodes are s and t).

6×6–Grid Network, shown in Figure 4.5. In this case the $\mathcal{K}$–terminal connectivity, for the case of equi–reliable links, was implemented.

C_i **Networks,** $i = 10, 15, 20, 25, 30, 40, 50$**,** also known as *complete networks*, for which only the all–terminal connectivity, with equi–reliable links, was implemented.

All the experiments were performed in a cluster Sun Fire X2250 Server, with processor Quad Core Intek Xeon Processor Model L5420 (2.50GHz, 1333 MHz, 50W), RoHS–5.

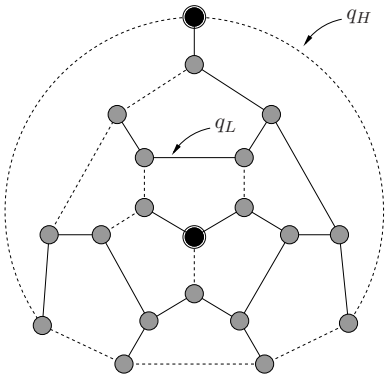


Figure 4.4: Dodecahedron Network

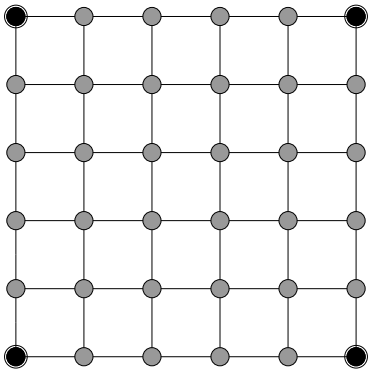


Figure 4.5: 6×6–Grid Network

4.6 Numerical Comparisons

In this section *Splitting/CP* is compared with many well–known methods used in network *reliability* estimation, namely: RVR [Cancela 2003], Sequential Construction and Destruction [Easton 1980], Creation, Destruction and Merge Process [Elperin 1991], Bound-Based Sampling [Fishman 1986, Kumamoto 1977], Leap–Evolve and Tree–Merge [Hui 2003], Cross Entropy over Merge, Permutation and *Crude Monte Carlo* [Hui 2005], Total Hazard [Jun 1992], Failure Sets Method [Karp 1983], and Dagger [Kumamoto 1980].

To make the comparisons, different experiments, designed on the basis of previous papers (see Section 4.5), have been performed. These experiments consist of running simulations over selected network topologies and obtaining an *unreliability* estimation together with measures of precision and efficiency. Actually the most interesting results of the experiments are not the estimations themselves, but the precision and efficiency of the methods, instead. This way, the results of the *Splitting/CP* procedure are compared to those published in the referred papers.

If $\widehat{Q}$ is the network *unreliability* estimator obtained by the method under analysis in time t (either *Splitting/CP* or any of the other ones), with expectation $\mathbb{E}\{\widehat{Q}\} = Q$ and variance $\mathbb{V}\{\widehat{Q}\}$, and $\widehat{Q}_c$ is the corresponding estimator obtained by *Crude Monte Carlo* in time t_c, the following notation applies for the results shown in this section:

- RE $= \mathbb{V}\{\widehat{Q}\}^{1/2}/\mathbb{E}\{\widehat{Q}\}$, the *Relative Error* ($RE \times 100$ in the cases where it is referred to as a percentage).
- VR $= \mathbb{V}\{\widehat{Q}_c\}/\mathbb{V}\{\widehat{Q}\}$, the *Variance Ratio*, that shows the precision improvement of the method under analysis over *Crude Monte Carlo* for runs that share a common parameter such as the sample size or the number of replications.
- W $= (\mathbb{V}\{\widehat{Q}_c\} \times t_c)/(\mathbb{V}\{\widehat{Q}\} \times t)$, the *Speedup*, also referred to as the *Precision Gain*, as it shows the precision improvement of the method under analysis over *Crude Monte Carlo* given a fixed computational time or, alternatively, the time improvement for a given precision.

The determination of either W or VR requires a reliable *Crude Monte Carlo* estimation of Q. Such estimation can only be obtained when the sample size N is considerably larger than $1/Q$. If

$N \leq 1/Q$, the number of samples for which the network state is *failed* tends to 0, and so do the crude estimator $\widehat{Q}_c$, its variance $\mathbb{V}\{\widehat{Q}_c\}$ and the product $\mathbb{V}\{\widehat{Q}_c\} \times t_c$. Nevertheless, the true value of $\mathbb{V}\{\widehat{Q}_c\}$ is $Q \times (1-Q)/N$ and, therefore, $\mathbb{V}\{\widehat{Q}_c\} \times t_c = Q \times (1-Q)/(N/t_c)$. As (N/t_c) is a constant, the product $\mathbb{V}\{\widehat{Q}_c\} \times t_c$ is a constant as well. Finally, as the product $\mathbb{V}\{\widehat{Q}_c\} \times t_c$ is independent of the sample size N and, as long as the true value of Q (or, at least, a precise estimation) is known, it suffices to determine the constant (N/t_c) from a not necessarily too long run, and then to make $\mathbb{V}\{\widehat{Q}_c\} \times t_c = Q \times (1-Q)/(N/t_c)$.

In every one of the following experiments, a set of pilot runs was performed to select the best value of m. All the experiments report the value of m that has been used.

The *Splitting/CP* method was programmed in CWEB [Knuth 1994], using the gcc compiler. The variant selected in all cases was *Fixed Effort*. Simulations proceeded as follows: K trials were done, being each one of them a single *Fixed Effort* simulation in which F trajectories were launched. Every one of the K trials reported an estimate $\widehat{Q}_i$. As the exact variance of the estimator $\widehat{Q}$ is unknown, an unbiased estimation $\widehat{\mathbb{V}}\{\widehat{Q}\}$ is reported in its place (actually, in all the experiments the values of RE, VR and W were obtained using $\widehat{\mathbb{V}}\{\widehat{Q}\}$ in place of $\mathbb{V}\{Q\}$). Based on the data collected from the experiments, the final *Splitting/CP* estimate, and the variance estimation, were calculated as:

$$\widehat{Q} = \frac{1}{K}\sum_{i=1}^{K} \widehat{Q}_i \qquad \widehat{\mathbb{V}}\{\widehat{Q}\} = \frac{1}{K-1}\left[\frac{1}{K}\left(\sum_{i=1}^{K} \widehat{Q}_i^2\right) - \widehat{Q}^2\right].$$

The sample size of the *Splitting/CP* simulations is accepted to be $N = K \times F$.

In the rest of this section, the results of the experiments are shown. As said before, the most important ones are those that reflect the behaviour of the different methods in the sense of precision (RE, VR) and efficiency (t, W). However, in all the experiments, some estimates $\widehat{Q}$ are also reported, as they give information about the rarity of the failure events.

4.6.1 RE and W on the Easton–Wong Network *all*–terminal connectivity

Table 4.1 shows RE and W for the Easton–Wong Network. Results for Merge Process, Destruction Process and Sequential Destruction are taken from the work of [Elperin 1991]. The sample size N was 10^4 ($K = 20$ and $F = 500$) for all methods and the number of *thresholds* m was 5 for EW-1 and 12 for EW-2.

The *relative errors* of *Splitting/CP* are higher than those of the other methods. However, in the *Speedup* evaluation *Splitting/CP* outperforms the other techniques. This indicates that the *Splitting/CP* execution time per replication (related to *Crude Monte Carlo*) is lower than the other methods, and that the overall trade–off precision–computational time is also better for *Splitting/CP*. The performance improvement of *Splitting/CP* is more significant when the network is more reliable.

4.6.2 W on the Dodecahedron Network *s–t* connectivity

Simulations were executed on the Dodecahedron Network with equi–reliable links, spanning five *unreliability* values ranging from 0.50 which is very high, up to 0.02 which is a low —but not an extremely low— value. The results are shown in Table 4.2, where the *Speedup* W is presented as a

Table 4.1: Easton–Wong Network, *all*–terminal connectivity

		Network	
Measure	*Method*	EW–1	EW–2
$\widehat{Q}$	Merge Process	4.38E−02	5.38E−03
	Splitting/CP	4.38E−02	5.47E−03
RE	Destruction Process	0.9%	1.6%
	Merge Process	0.6%	1.2%
	Splitting/CP	1.7%	2.5%
W	Destruction Proces	<1	1.5
	Merge Process	3.4	5.4
	Sequential Destruction	2.4	7.6
	Splitting/CP	4.1	10.6

function of the network *unreliability*. The sample size N was 10^4 ($K = 20$ and $F = 500$) and for *Splitting/CP*, the number of *thresholds* m was set to 2, 5, 6, 11 and 16 respectively. Values for all previously published methods are taken from the work of [Elperin 1991], except for the RVR-SP measures that come from the article by [Cancela 2003].

Some methods —like Failure Sets Method and, particularly, RVR–SP— exhibit extremely high values of W in some particular cases. Failure Sets Method stands extremely high for a Single Link *unreliability* of 0.05 while RVR–SP is, by far, the best in all cases indicating that, besides being a very efficient method it fits particularly well to the Dodecahedron Network topology. Except for these particular cases, *Splitting/CP* is in the order of the average of all other methods. As expected, the *Speedup* of *Splitting/CP* grows together with the network *reliability*.

Table 4.2: Dodecahedron Network, *s*–*t* connectivity

		Single Link unreliability $q_i, i = 1, 2, \cdots, 30$				
Measure	*Method*	0.50	0.20	0.10	0.05	0.02
$\widehat{Q}$	Merge Process	7.10E−01	3.58E−02	2.82E−03	2.88E−04	1.67E−05
	RVR–SP	7.10E−01	–	2.88E−03	2.95E−04	1.70E−05
	Splitting/CP	7.18E−01	3.77E−02	2.80E−03	2.96E−04	1.69E−05
W	Destruction Process	<1	<1	<1	1.30	8.20
	Merge Process	0.67	2.00	8.80	55.70	495.00
	Dagger	1.56	–	1.81	1.91	–
	Sequential Construction	0.68	–	1.40	2.71	–
	Bound based sampling	0.56	–	12.30	136.00	–
	Failure Sets Method	0.05	–	0.30	3,714.40	–
	Total Hazard	0.12	–	6.63	250.10	386.00
	RVR–SP	54.60	–	2,040.00	25,100.00	507,000.00
	Splitting/CP	0.74	1.85	12.25	27.89	333.41

4.6.3 RE and t on the Dodecahedron Network *all*–terminal connectivity

Results are shown in Table 4.3. The RVR–SP measures are taken from the work of [Cancela 2008], and the other results are taken from the article by [Hui 2003], where a set of methods were compared on the Dodecahedron Network in two versions, with different values for q_L and q_H. The sample size N was 10^6 ($K = 2000$ and $F = 500$) for all the experiments and the number of *thresholds* for *Splitting/CP*, 16 for the case of $\{q_L = 10^{-2}, q_H = 10^{-3}\}$ and 25 for $\{q_L = 10^{-2}, q_H = 10^{-6}\}$. The execution time, t, consigned in the third row of the table, is measured in seconds.

Splitting/CP appears to be one of the fastest methods, only exceeded by RVR. From the accuracy point of view (RE) *Splitting/CP* does not behave as well as the rest, but considering accuracy together with execution time, *Splitting/CP* behaves around the average of the rest of the methods.

Table 4.3: Dodecahedron Network, *all*–terminal connectivity

		Single Link Unreliability	
Measure	*Method*	q_H =1E−03 q_L =1E−02	q_H =1E−06 q_L =1E−02
$\widehat{Q}$	Exact Determination	7.90E−07	7.04E−10
	Merge Process	7.92E−07	7.03E−10
	Leap Evolve (0.15)	7.89E−07	7.05E−10
	Leap Evolve (0.25)	7.89E−07	7.05E−10
	Tree Merge (1+)	7.85E−07	7.04E−10
	Tree Merge (2+)	7.91E−07	7.04E−10
	RVR–SP	7.91E−07	7.04E−10
	Splitting/CP	7.84E−07	6.96E−10
RE	Exact Determination	–	–
	Merge Process	0.15%	0.14%
	Leap Evolve (0.15)	0.19%	0.24%
	Leap Evolve (0.25)	0.29%	0.68%
	Tree Merge (1+)	0.60%	0.13%
	Tree Merge (2+)	0.14%	0.000061%
	RVR–SP	0.13%	0.07%
	Splitting/CP	0.63%	0.71%
t	Exact Determination	1488	1488
	Merge Process	813	812
	Leap Evolve (0.15)	108	47
	Leap Evolve (0.25)	43	32
	Tree Merge (1+)	53	53
	Tree Merge (2+)	82	82
	RVR–SP	4	2
	Splitting/CP	31	40

4.6.4 RE on the 6×6–Grid Network $\mathcal{K}$–terminal connectivity

Table 4.4 shows the results. The RVR–SP measures are taken from the work of [Cancela 2008]; the other method's results are taken from the article by [Hui 2005], where a set of techniques,

improved by Cross Entropy, were compared based on simulations on the 6×6–Grid Network, in two different equi–reliable conditions. The sample size N was 10^6 ($K = 2000$ and $F = 500$) and the number of *thresholds* m, 25 for $q_i = 10^{-3}$ and 49 for $q_i = 10^{-6}$.

In this experiment *Splitting/CP* yields a better performance than the methods supported by Cross Entropy, except for the Merge Process and for RVR.

It is worth noting that both variants associated to Crude Monte Carlo simulation (either direct or supported by Cross Entropy) are not even able to make a reasonable estimation of the unreliability, $\widehat{Q}$. Clearly a number of replications $N{=}10^6$ is extremely low for unreliabilities in the order of 10^6 and even worse for 10^{12}. Concerning the relative error, *Splitting/CP* yields a better performance than most methods supported by Cross Entropy, except for the Merge Process and for RVR.

Table 4.4: 6×6–Grid Network, $\mathcal{K}$–terminal connectivity

		Single Link unreliability $q_i, i = 1, 2, \cdots, 6$	
Measure	*Method*	1E−03	1E−06
$\widehat{Q}$	CE–Merge Process	4.00E−06	4.00E−12
	Merge Process	4.01E−06	4.00E−12
	CE–Permutation Monte Carlo	4.02E−06	4.00E−12
	Permutation Monte Carlo	3.95E−06	4.01E−12
	CE–Crude Monte Carlo	1.30E−06	8.60E−14
	Crude Monte Carlo	6.00E−06	–
	RVR–SP	4.02E−06	4.01E−12
	Splitting/CP	4.02E−06	4.00E−12
RE	CE–Merge Process	0.15%	0.15%
	Merge Process	0.17%	0.18%
	CE–Permutation Monte Carlo	1.18%	1.28%
	Permutation Monte Carlo	2.03%	2.10%
	CE–Crude Monte Carlo	48.91%	90.87%
	Crude Monte Carlo	40.82%	–
	RVR–SP	0.24%	0.22%
	Splitting/CP	0.48%	0.66%

CE-. . . *indicates "Cross Entropy over . . ."*

4.6.5 RE, VR and W on the Complete Networks all–terminal connectivity

Table 4.5 shows the results of the experiments obtained with *Splitting/CP* and also results taken from the work of [Elperin 1991] and [Cancela 2008]. Simulations were carried out on the suite of complete networks $C_i, i = 10, 15, 20, 25, 30, 40, 50$ for the all–terminal connectivity, with equi–reliable links, $q_i = 0.55$. The sample size N was 10^4 ($K = 20$ and $F = 500$) and the number of *thresholds* m, respectively, 5, 13, 14, 21, 24, 28 and 27.

This experiment shows that the efficiency of *Splitting/CP* declines as the network graph becomes more "dense" (in terms of proportion of edges in relation to nodes). The Graph Density, defined as $2|\mathcal{E}|/(|\mathcal{V}|(|\mathcal{V}|-1))$, equals 1.00 for the complete networks, 0.16 for the Dodecahedron, 0.10 for the 6×6–Grid Network and 0.02 for the Easton–Wong network.

Table 4.5: Complete Networks, all–terminal connectivity, with $q_i = 0.55$

Measure	*Method*	*Network* C_{10}	C_{15}	C_{20}	C_{25}	C_{30}	C_{40}	C_{50}
$\widehat{Q}$	Merge Process	4.56E−02	3.46E−03	2.32E−04	1.47E−05	8.89E−06	–	–
	RVR–SP	4.59E−02	3.47E−03	2.33E−04	1.47E−05	8.86E−06	2.99E−09	9.49E−12
	Splitting/CP	4.76E−02	3.38E−03	2.44E−04	1.47E−05	8.48E−07	3.27E−09	9.51E−12
RE	Merge Process	0.54%	0.57%	0.53%	0.50%	0.47%	–	–
	RVR–SP	0.11%	0.11%	0.05%	0.03%	0.05%	0.02%	0.07%
	Splitting/CP	2.20%	2.21%	2.72%	3.76%	3.74%	4.10%	5.47%
VR	Merge Process	7.00E+01	8.87E+02	1.52E+04	2.70E+05	5.08E+07	–	–
	RVR–SP	2.72E+02	5.54E+03	1.10E+05	2.11E+06	4.27E+07	1.69E+09	6.47E+12
	Splitting/CP	3.99E+00	6.32E+01	5.28E+02	4.83E+03	8.86E+04	1.66E+07	3.51E+09
W	Merge Process	2.10E+01	1.69E+02	3.28E+03	4.72E+04	7.30E+06	–	–
	RVR–SP	8.16E+02	2.16E+03	1.03E+04	7.44E+04	7.67E+05	1.54E+07	3.43E+10
	Splitting/CP	0.95E+00	7.26E+00	5.04E+01	3.10E+02	4.51E+03	5.80E+05	8.86E+07

4.7 Splitting/CP Efficiency

In this section *Splitting/CP* is subject to simulation analysis on the Dodecahedron Network, for the *s–t* connectivity model, in order to observe the accuracy of the estimations as a function of the network *unreliability*. The results reported are the *relative error* and the execution time, t, of the simulations (measured in seconds). The experiments were repeated for four different values of the sample size N (with the values of K, 80, 125, 200 and 2000, and the values of F, respectively, 1250, 8000, 50000 and 50000). For the single link unreliabilities of 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6} and 10^{-7}, the number of *thresholds* m was set, respectively, to 10, 17, 18, 26, 30, 36 and 39. Results are shown in Table 4.6.

As expected, the *relative error* decreases in an inverse proportion to the square root of the sample size (the variance is inversely proportionally to the sample size), while the execution time grows linearly with the sample size. It is interesting to see that the evolutions of the *relative error* and the execution time, as a function of the sample size, are similar for all network *reliability* values.

On the other hand, the increment of the execution times together with the network *reliability* is due to the increasing number of *thresholds*, what leads to a growth in the number of operations involved in the whole simulation. Also, the *relative error* grows as the network becomes more reliable. It is important to notice that both quantities grow very slowly; while the link unreliabilities change over 6 orders of magnitude, and the network *unreliability* over 18 orders of magnitude, execution times and *relative errors* increase less than one order of magnitude (actually they are multiplied by a factor of about 3 or 4). Moreover, this increment becomes smaller as the network *reliability* grows higher, showing a high robustness of the method with regard to the rarity of the

events of interest. This feature will be further explored in the following section.

Table 4.6: Dodecahedron Network, *s–t* connectivity

Measure	*Sample Size, N*	*Single Link* unreliability $q_i, i = 1, 2, \cdots, 30$						
		1E−01	1E−02	1E−03	1E−04	1E−05	1E−06	1E−07
$\widehat{Q}$	1E+08	2.88E−03	2.06E−06	2.00E−09	2.00E−12	1.99E−15	2.00E−18	2.00E−21
RE	1E+05	1.28%	1.92%	2.75%	2.88%	3.14%	3.42%	3.39%
	1E+06	0.46%	0.71%	0.86%	0.88%	0.99%	1.00%	1.12%
	1E+07	0.14%	0.21%	0.27%	0.28%	0.33%	0.34%	0.36%
	1E+08	0.04%	0.07%	0.09%	0.09%	0.10%	0.11%	0.11%
t	1E+05	1.96	2.99	3.58	4.26	5.13	6.20	7.20
	1E+06	19.64	29.99	35.80	40.34	51.28	61.87	69.66
	1E+07	195.83	299.10	358.26	406.02	512.28	620.11	734.97
	1E+08	1,961.33	2,994.20	3,560.01	4,078.22	5,099.93	6,196.06	6,903.56

4.8 Empirical Analysis of Robustness

A fundamental issue concerning the robustness of *Splitting/CP* estimations is now considered. As the *unreliability* is the failure probability of the whole network, when networks are highly reliable such a failure is a *rare event*. One usual way to model this problem is to consider the unreliabilities of the links as polynomials of some parameter ε, where such parameter is a measure of rarity. In the particular case of equi–reliable links, the single link *unreliability*, q, is an appropriate measure of rarity: $\varepsilon = q$. In this section, robustness of the *Splitting/CP* estimations will be analyzed for networks with equi–reliable links (although it is also easy to do it in the general polynomial case).

Let $\widehat{Q}$ be the estimate obtained from a single *Splitting/CP* run. Using the Central Limit Theorem approximation, the corresponding Confidence Interval, at confidence level $\eta = 1 - \alpha$, is:

$$CI(\eta) = \left[\widehat{Q} - z_\eta \mathbb{V}\{\widehat{Q}\}^{1/2}, \widehat{Q} + z_\eta \mathbb{V}\{\widehat{Q}\}^{1/2}\right],$$

where $z_\eta = \Phi^{-1}(1 - \alpha/2) = \Phi^{-1}((1 + \eta)/2)$, and Φ is the standard normal cumulative distribution. If $Q = \mathbb{E}\{\widehat{Q}\}$ is the true network *reliability* value, the expectation of the following indicator function:

$$I_\eta = \begin{cases} 1 & \text{if } Q \in CI(\eta), \\ 0 & \text{otherwise}, \end{cases}$$

should not be less than η for a robust estimation, no matter the value of ε. However, in real simulations I_η is usually not independent of ε. Actually, the size of $CI(\eta)$ is closely related to the value of ε.

A network *unreliability* estimation method is said to be robust if, when $\varepsilon \to 0$, the mean of the indicator function I_η is still higher or equal than η or, from the simulation point of view, the average of a significant number of samples of I_η does not fall under η. To make an empirical evaluation of the *Splitting/CP* robustness for a given value of η, R runs of *Splitting/CP* were performed for

different values of ε. If $1_\eta^{(i)}$ is the sample obtained in the i–th run, the Coverage Factor is defined as:

$$CF(\varepsilon,\eta) = \frac{1}{R}\sum_{i=1}^{R} 1_\eta^{(i)}.$$

This factor denotes the proportion of times that, for a given measure of rarity ε, the real value Q lies inside a Confidence Interval of confidence level η.

In the next experiment, an empirical Coverage Factor $CF(\varepsilon,\eta)$ is determined. The networks selected for the experiment were C_{10} (all–terminal connectivity) and the Dodecahedron (*s*–*t* connectivity). In the case of C_{10}, the true value of the *unreliability* is known by means of the following recursive formula [Colbourn 1987]:

$$Q_n = \sum_{j=1}^{n-1} \binom{j-1}{n-1} q^{j(n-j)}(1-Q_j),$$

where Q_n is the *unreliability* of an equi–reliable network C_n, with $Q_1 = 0$ and $Q_2 = q$ (the single link *unreliability*). This formula was implemented on the *Maple* program. In the case of the Dodecahedron, the true value of *unreliability* was determined by means of an exact algorithm.

The confidence levels selected were 90%, 95% and 99%, and the measures of rarity $\varepsilon = 0.5$, 0.1, 0.05, 0.01 and 0.005 for the C_{10} Network and 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5} and 10^{-6} for the Dodecahedron. In the order just described, the number of *thresholds* m for the C_{10} network were 2, 9, 12, 20 and 23, and for the Dodecahedron, 3, 7, 10, 13, 17 and 20. The number of runs was $R = 2000$ for every confidence level. The results are shown in Table 4.7 and Table 4.8.

Table 4.7: Coverage Factor of a network *unreliability Splitting/CP* estimation for the C_{10} Network (*all*–terminal connectivity) with equi–reliable links, $q = \varepsilon$

	Single Link unreliability q = *Measure of Rarity* ε				
Measure	0.5	0.1	0.05	0.01	0.005
Q	1.955082E−02	1.000000E−08	1.953125E−11	1.000000E−17	1.953125E−20
$CF_{90\%}$	89.75%	89.90%	91.00%	90.25%	90.50%
$CF_{95\%}$	94.95%	95.00%	95.05%	94.60%	95.25%
$CF_{99\%}$	98.45%	99.15%	98.60%	98.80%	98.85%

It is clear that for both networks the empirical Coverage Factor agrees with the confidence level, no matter how reliable the networks are. This can be seen as an empirical verification of the unbiased character of the *Splitting/CP* estimator. But the most important conclusion derived from this experiment is the high degree of robustness of the *Splitting/CP* estimator.

The rest of this section shows the results obtained by performing a test proposed by [Schruben 1980] to explore the coverage of confidence intervals in more detail. This test applies to the current problem in the following way. As stated, for every estimate $\widehat{Q}$ there is a confidence interval $CI(\eta)$ of width $z_\eta \mathbb{V}\{\widehat{Q}\}^{1/2}$ and confidence level η. So, the true value Q is supposed to lie inside this interval with probability η. For the same collected data used in this estimation, $CI(\eta^*)$ is a new confidence interval of width $z_{\eta^*}\mathbb{V}\{\widehat{Q}\}^{1/2}$ and confidence level η^*, wide enough to let the

Table 4.8: Coverage Factor of a network *unreliability Splitting/CP* estimation for the Dodecahedron Network (*s*–*t* connectivity) with equi–reliable links, $q = \varepsilon$

	Single Link unreliability $q =$ *Measure of Rarity* ε					
Measure	1E−01	1E−02	1E−03	1E−04	1E−05	1E−06
Q	2.879601E−03	2.061890E−06	2.006006E−09	2.000600E−12	2.000060E−15	2.000006E−18
$CF_{90\%}$	90.30%	90.55%	89.35%	89.85%	89.75%	90.40%
$CF_{95\%}$	95.10%	95.40%	94.85%	94.80%	94.85%	95.25%
$CF_{99\%}$	99.15%	99.10%	98.75%	99.05%	99.00%	99.15%

true value Q lie inside of it. Then:

$$\begin{aligned} z_{\eta^*}\mathbb{V}\{\widehat{Q}\}^{1/2} &= |\widehat{Q} - Q| \\ z_{\eta^*} &= \frac{|\widehat{Q} - Q|}{\mathbb{V}\{\widehat{Q}\}^{1/2}} \\ (1+\eta^*)/2 &= \Phi\left(\frac{|\widehat{Q} - Q|}{\mathbb{V}\{\widehat{Q}\}^{1/2}}\right) \\ \eta^* &= 2\,\Phi\left(\frac{|\widehat{Q} - Q|}{\mathbb{V}\{\widehat{Q}\}^{1/2}}\right) - 1. \end{aligned}$$

It is clear that η^* is a random variable uniformly distributed in $[0, 1]$, because its cumulative distribution function is:

$$F_{\eta^*}(\eta) = \mathbb{P}\{\eta^* \leq \eta\} = \mathbb{P}\{Q \in CI(\eta)\} = \eta.$$

The test proposed by [Schruben 1980] consists of producing a large number of samples of η^* and to see if these samples follow an uniform distribution. One way to do this is graphically, by tracing an estimation of the cumulative distribution function, $\bar{F}_{\eta^*}(\eta)$, with the following interpretation: the shape of $F_{\eta^*}(\eta)$ is, of course, the straight line (with 45° inclination) $F_{\eta^*}(\eta) = \eta$. If at some value η it happens that $\bar{F}_{\eta^*}(\eta) < \eta$ then the coverage is lower than expected and likely to lead to erroneous conclusions; on the other hand, $\bar{F}_{\eta^*}(\eta) > \eta$ means that the confidence interval is too conservative and that the estimator $\hat{Q}$ is not efficient in the sense that the desired coverage can be achieved with a smaller sample size.

The networks selected for this test are two versions of the Dodecahedron, with the extreme values of rarity 10^{-1} and 10^{-6}. The number of *thresholds* were 3 and 20, respectively, and the number of runs $R = 2000$ in both cases. The results are shown in Figures 4.6 and 4.7.

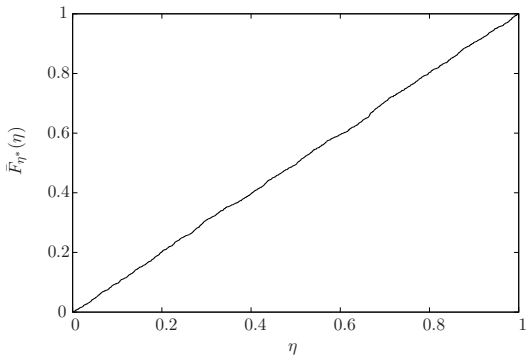


Figure 4.6: Schruben Test for the Dodecahedron Network (*s–t* connectivity) with equi–reliable links, $q = \varepsilon = 10^{-1}$

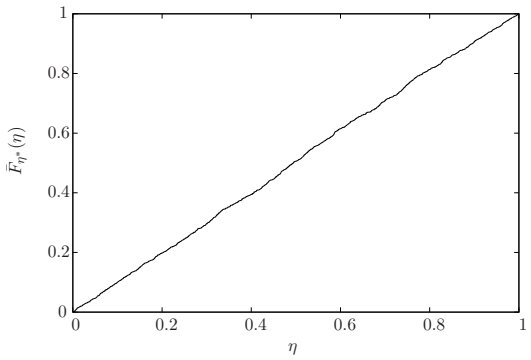


Figure 4.7: Schruben Test for the Dodecahedron Network (*s–t* connectivity) with equi–reliable links, $q = \varepsilon = 10^{-6}$

For both tests $\bar{F}_{\eta^*}(\eta)$ is tightly close to the cumulative distribution function of a uniform distribution. These results are further indication that the type of confidence interval selected for the *Splitting/CP* estimator is appropriate and robust.

CHAPTER 5

Conditional Monte Carlo on Different Settings

Contents

Abstract

The variance reduction capacity of *Conditional Monte Carlo* is shown by means of many examples on different settings. The examples illustrate the potential of *Conditional Monte Carlo* as a variance reduction technique and behave as a background for Chapter 6 in which an original proposal of this thesis, consisting in an application of *Conditional Monte Carlo* to a Markov chain model, is introduced.

5.1 The determination of π

Let $V = (X, Y)$ be a random vector, where X and Y are two independent random variables such that $X \sim Unif[0, 1]$ and $Y \sim Unif[0, 1]$. Let $\theta = \{X^2 + Y^2 \leq 1\}$ be an event on the state space of V. As the state space of V is a square of size 1, event θ is the set of points of that square that belong to a circle of radius 1, centered in one of the corners of the square.

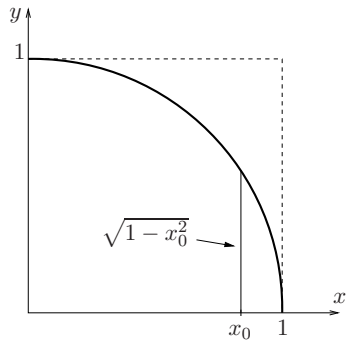


Figure 5.1: Event $\theta = \{X^2 + Y^2 \leq 1\}$

The probability of θ can be determined as:

$$\begin{aligned}
\mathbb{P}\{\theta\} &= \int_0^1 \int_0^{\sqrt{1-x^2}} f(x,y)\, dy\, dx \\
&= \int_0^1 \left\{ \int_0^{\sqrt{1-x^2}} 1\, dy \right\} dx \\
&= \int_0^1 \left\{ y\Big|_0^{\sqrt{1-x^2}} \right\} dx \\
&= \int_0^1 \sqrt{1-x^2}\, dx \\
&= \frac{x}{2}\sqrt{1-x^2} + \frac{1}{2}\arcsin x\Big|_0^1 \\
&= \frac{\pi}{4}.
\end{aligned}$$

Let I be a random variable defined as:

$$I = \begin{cases} 1 & \text{if } X^2 + Y^2 \leq 1, \\ 0 & \text{otherwise.} \end{cases}$$

Therefore, $\mathbb{E}\{I\} = \mathbb{P}\{\theta\} = \pi/4$.

Now the variance $\mathbb{V}\{I\}$ is determined. As I is a Bernoulli random variable,

$$\mathbb{V}\{I\} = \mathbb{E}\{I\}\,(1 - \mathbb{E}\{I\}) = \frac{\pi}{4}\,(1 - \frac{\pi}{4}) = 0.1685.$$

Figure 5.1 shows that, given a value of x, say x_0, the probability that the associated point (x_0, y_0) lies inside the circle is $\sqrt{1-x_0^2}$. Formally speaking, $\mathbb{P}\{\theta|X = x_0\} = \sqrt{1-x_0^2}$ or, in terms of I, $\mathbb{E}\{I|X = x_0\} = \sqrt{1-x_0^2}$. Thus, $\mathbb{E}\{I|X\} = \sqrt{1-X^2}$ and, clearly, $\mathbb{E}\{\mathbb{E}\{I|X\}\} = \mathbb{E}\{I\}$.

The variance of $\mathbb{E}\{I|X\}$ is now determined:

$$\begin{aligned}\mathbb{V}\{\mathbb{E}\{I|X\}\} &= \mathbb{E}\{\mathbb{E}\{I|X\}^2\} - \mathbb{E}\{\mathbb{E}\{I|X\}\}^2 \\ &= \mathbb{E}\{(1-X^2)\} - \left(\frac{\pi}{4}\right)^2 \\ &= \int_0^1 (1-x^2)\,dx - \left(\frac{\pi}{4}\right)^2 \\ &= \frac{2}{3} - \left(\frac{\pi}{4}\right)^2 \\ &= 0.0498.\end{aligned}$$

A measure of accuracy increase for this problem is:

$$\frac{\mathbb{V}\{I\}}{\mathbb{V}\{\mathbb{E}\{I|X\}\}} = \frac{0.1685}{0.0498} = 3.38$$

Suppose now that the interest is to obtain the value of $\mathbb{P}\{\theta\} = \mathbb{E}\{I\}$ by simulation. One of the alternatives is the standard simulation, that can be achieved from a set of N independent samples or copies of variable I, i.e. $I^{(i)},\ i = 1, \cdots, N$. Calling this *Standard* estimator $\widehat{I}_s$, it follows that:

$$\widehat{I}_s = \frac{1}{N}\sum_{i=1}^{N} I^{(i)} \qquad \text{and} \qquad \mathbb{V}\{\widehat{I}_s\} = \frac{\mathbb{V}\{I\}}{N}$$

The other alternative is to simulate given a set of N independent samples or copies of variable X, and to build the samples $\mathbb{E}\{I|X^{(i)}\},\ i = 1, \cdots, N$. Calling this *Conditional Monte Carlo* estimator $\widehat{I}_c$, it follows that:

$$\widehat{I}_s = \frac{1}{N}\sum_{i=1}^{N} \mathbb{E}\{I|X^{(i)}\} \qquad \text{and} \qquad \mathbb{V}\{\widehat{I}_s\} = \frac{\mathbb{V}\{\mathbb{E}\{I|X\}\}}{N}$$

Then, the ratio $\mathbb{V}\{\widehat{I}_s\}/\mathbb{V}\{\widehat{I}_s\}$ will still be 3.38.

5.2 Sum of two independent random variables

Let $V = (X, Y)$ be a random vector, where X and Y are two independent random variables such that $X \sim exp(\lambda_1)$ and $Y \sim exp(\lambda_2)$. Let $\theta = \{X + Y \leq L\}$ be an event on the state space of V. As the state space of V is the positive numbers sector of the $\mathbb{R}^2$ plane, event θ defines the right–angled triangle of sides L shown in Figure 5.2.

If $\lambda_1 \neq \lambda_2$, the probability of θ is:

$$\begin{aligned}\mathbb{P}\{\theta\} &= \int_0^L \int_0^{L-x_0} f(x,y)\,dy\,dx \;=\; \int_0^L \int_0^{L-x_0} f(x)f(y)\,dy\,dx \\ &= \int_0^L f(x)\left\{\int_0^{L-x_0} f(y)\,dy\right\}dx\end{aligned}$$

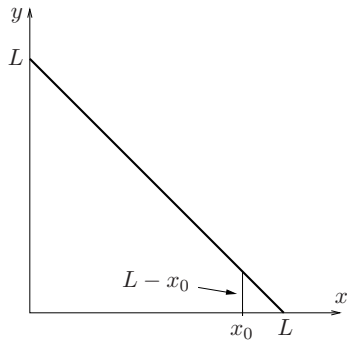


Figure 5.2: Event $\theta = \{X + Y \leq L\}$

$$
\begin{aligned}
&= \int_0^L \lambda_1 e^{-\lambda_1 x} \left\{ \int_0^{L-x_0} \lambda_2 e^{-\lambda_2 y}\, dy \right\} dx \\
&= \frac{\lambda_2 - \lambda_1 - \lambda_2 e^{-\lambda_1 L} + \lambda_1 e^{-\lambda_2 L}}{\lambda_2 - \lambda_1},
\end{aligned}
$$

whereas, if $\lambda_1 = \lambda_2 = \lambda$, the probability of θ is:

$$\mathbb{P}\{\theta\} = 1 - e^{-\lambda L} - \lambda L\, e^{-\lambda L}.$$

Let I be a random variable defined as:

$$I = \begin{cases} 1 & \text{if } X + Y \leq L, \\ 0 & \text{otherwise.} \end{cases}$$

It is clear that $\mathbb{E}\{I\} = \mathbb{P}\{\theta\}$. As I is a Bernoulli random variable, its variance is:

$$\mathbb{V}\{I\} = \mathbb{E}\{I\}\,(1 - \mathbb{E}\{I\}).$$

Table 5.1 shows $\gamma = \mathbb{E}\{I\}$ and $\mathbb{V}\{I\}$ for a set of values of λ_1, λ_2 and L.

Table 5.1: Analysis of I for the sum of two independent random variables

λ_1	λ_2	L	$\gamma = \mathbb{E}\{I\}$	$\mathbb{V}\{I\}$
1.00	1.00	1.00	0.26424112	0.19441775
0.50	1.00	1.00	0.15481812	0.13084947
0.10	1.00	1.00	0.03550058	0.03424029
0.10	1.00	0.50	0.01047071	0.01036108
0.10	1.00	0.10	0.00048212	0.00048189
0.01	1.00	0.01	0.00000050	0.00000050

Figure 5.2 shows that, given a value of $x \leq L$, say x_0, the probability that the associated point (x_0, y_0) lies inside the triangle is $\int_0^{L-x_0} f(y)\, dy = 1 - e^{-\lambda_2 (L - x_0)}$. Formally speaking,

$\mathbb{P}\{\theta|X = x_0\} = 1 - e^{-\lambda_2(L-x_0)}$ or, in terms of I, $\mathbb{E}\{I|X = x_0\} = 1 - e^{-\lambda_2(L-x_0)}$. Thus, $\mathbb{E}\{I|X\} = 1 - e^{-\lambda_2(L-X)}$ and, clearly, $\mathbb{E}\{\mathbb{E}\{I|X\}\} = \mathbb{E}\{I\}$.

If $\lambda_1 \neq \lambda_2$, the variance of $\mathbb{E}\{I|X\}$ can be determined as follows:

$$\begin{aligned}
\mathbb{V}\{\mathbb{E}\{I|X\}\} &= \mathbb{E}\{\mathbb{E}\{I|X\}^2\} - \mathbb{E}\{\mathbb{E}\{I|X\}\}^2 \\
&= \mathbb{E}\left\{\left(1 - e^{-\lambda_2(L-X)}\right)^2\right\} - \left(\frac{\lambda_2 - \lambda_1 - \lambda_2 e^{-\lambda_1 L} + \lambda_1 e^{-\lambda_2 L}}{\lambda_2 - \lambda_1}\right)^2 \\
&= \left(1 - e^{-\lambda_1 L}\right) - 2\lambda_1 \frac{\left(e^{-\lambda_1 L} - e^{-\lambda_2 L}\right)}{\lambda_2 - \lambda_1} + \lambda_1 \frac{\left(e^{-\lambda_1 L} - e^{-2\lambda_2 L}\right)}{2\lambda_2 - \lambda_1} - \\
&\quad \left(\frac{\lambda_2 - \lambda_1 - \lambda_2 e^{-\lambda_1 L} + \lambda_1 e^{-\lambda_2 L}}{\lambda_2 - \lambda_1}\right)^2,
\end{aligned}$$

whereas, if $\lambda_1 = \lambda_2 = \lambda$, the variance of $\mathbb{E}\{I|X\}$ is:

$$\mathbb{V}\{\mathbb{E}\{I|X\}\} = 1 - 2\lambda L\, e^{-\lambda L} - e^{-2\lambda L}.$$

Table 5.2 shows $\gamma = \mathbb{E}\{\mathbb{E}\{I|X\}\}$ and $\mathbb{V}\{\mathbb{E}\{I|X\}\}$ and Table 5.3 shows a measure of accuracy increase, $\mathbb{V}\{I\}/\mathbb{V}\{\mathbb{E}\{I|X\}\}$. Both tables refer to the set of values of λ_1, λ_2 and L used in previous tables.

Table 5.2: Analysis of $\mathbb{E}\{I|X\}$ for the sum of two independent random variables

λ_1	λ_2	L	$\gamma = \mathbb{E}\{\mathbb{E}\{I\|X\}\}$	$\mathbb{V}\{\mathbb{E}\{I\|X\}\}$
1.00	1.00	1.00	0.26424112	0,12890583
0.50	1.00	1.00	0.15481812	0.04926338
0.10	1.00	1.00	0.03550058	0.01507841
0.10	1.00	0.50	0.01047071	0.00276384
0.10	1.00	0.10	0.00048212	0.00003064
0.01	1.00	0.01	0.00000050	3.31×10^{-9}

Table 5.3: Accuracy increase for the sum of two independent random variables

λ_1	λ_2	L	γ	$\mathbb{V}\{I\}/\mathbb{V}\{\mathbb{E}\{I\|X\}\}$
1.00	1.0	1.00	0.26424112	1.51
0.50	1.0	1.00	0.15481812	2.66
0.10	1.0	1.00	0.03550058	2.27
0.10	1.0	0.50	0.01047071	3.75
0.10	1.0	0.10	0.00048212	15.73
0.01	1.0	0.01	0.00000050	150.64

5.3 Analysis of a simple network (case 1)

Let $V = (X_1, X_2, X_3, X_4, X_5)$ be a random vector, where $\{X_i\}_{i=1}^5$ is a set of independent Bernoulli random variables with parameters r_1, r_2, r_3, r_4, r_5 ($X_i = 1$ with probability r_i, and $X_i = 0$ with probability $q_i = 1 - r_i$). These variables model the states of links 1, 2, 3, 4 and 5 of the Bridge Network shown in Figure 5.3 in the following way: $X_i = 1$ means that link i is *operational* and, therefore, the nodes at its extremes are connected, whereas $X_i = 0$ means that link i is *failed*, what is the same as removing it from the network. Any pair of nodes is considered connected if there is a path of *operational* links between them.

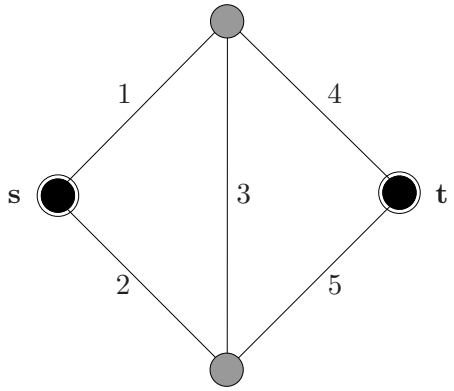


Figure 5.3: Bridge Network

Let θ be an event on the state space of V defined as follows: $\theta = \{$the set of values of $(X_1, X_2, X_3, X_4, X_5)$ such that nodes $\mathbf{s}$ and $\mathbf{t}$ are not connected$\}$. From basic network analysis (inclusion–exclusion principle to the mincuts), the probability of θ is:

$$\begin{aligned}\mathbb{P}\{\theta\} \;=\; & q_1q_2 + q_4q_5 + q_1q_3q_5 + q_2q_3q_4 - q_1q_2q_4q_5 - q_1q_2q_3q_5 - \\ & q_1q_2q_3q_4 - q_1q_3q_4q_5 - q_2q_3q_4q_5 + 2\, q_1q_2q_3q_4q_5. \end{aligned} \tag{5.1}$$

Let I be a random variable defined as:

$$I = \begin{cases} 1 & \text{if } \theta \text{ occurs,} \\ 0 & \text{otherwise.} \end{cases}$$

It is clear that $\mathbb{E}\{I\} = \mathbb{P}\{\theta\}$. In network analysis I is usually known as the *structure function*. As I is a Bernoulli variable, its variance is:

$$\mathbb{V}\{I\} = \mathbb{E}\{I\}\,(1 - \mathbb{E}\{I\}).$$

Table 5.4 shows $\gamma = \mathbb{E}\{I\}$ and $\mathbb{V}\{I\}$ for a set of values of q, in the particular case where $q_i = q\ \forall i$.

Suppose now that the value of one of the components of V, say X_1, is fixed. There are two possible values for X_1, 0 or 1 so, conditioned to X_1, $\mathbb{E}\{I\}$ will assume one of two values:

Table 5.4: Analysis of I for the simple network (case 1)

q	$\gamma = \mathbb{E}\{I\}$	$\mathbb{V}\{I\}$
1	1.00000E+00	0.00000E+00
0.1	2.15200E−02	2.10569E−02
0.01	2.01950E−04	2.01909E−04
0.001	2.00200E−06	2.00199E−06
0.0001	2.00020E−08	2.00020E−08
0.00001	2.00002E−10	2.00002E−10
0.000001	2.00000E−12	2.00000E−12

$\mathbb{E}\{I|X_1 = 0\}$ with probability q_1 and $\mathbb{E}\{I|X_1 = 1\}$ with probability r_1. Then:

$$\begin{aligned} \mathbb{E}\{\mathbb{E}\{I|X_1\}\} &= \mathbb{E}\{I|X_1 = 0\} \times q_1 + \mathbb{E}\{I|X_1 = 1\} \times r_1 \\ &= \mathbb{E}\{I|X_1 = 0\} \times q_1 + \mathbb{E}\{I|X_1 = 1\} \times (1 - q_1). \end{aligned} \tag{5.2}$$

The determinations of $\mathbb{E}\{I|X_1 = 0\}$ and $\mathbb{E}\{I|X_1 = 1\}$ are made by means of basic network analysis (inclusion–exclusion principle to the mincuts), on the Bridge Network transformed accordingly to the possible values of X_1 (see Figure 5.4).

$$\begin{aligned} \mathbb{E}\{I|X_1 = 0\} &= q_2 + q_4q_5 + q_3q_5 - q_2q_4q_5 - q_2q_3q_5 - q_3q_4q_5 + q_2q_3q_4q_5, \\ \mathbb{E}\{I|X_1 = 1\} &= q_2q_3q_4 + q_4q_5 - q_2q_3q_4q_5. \end{aligned}$$

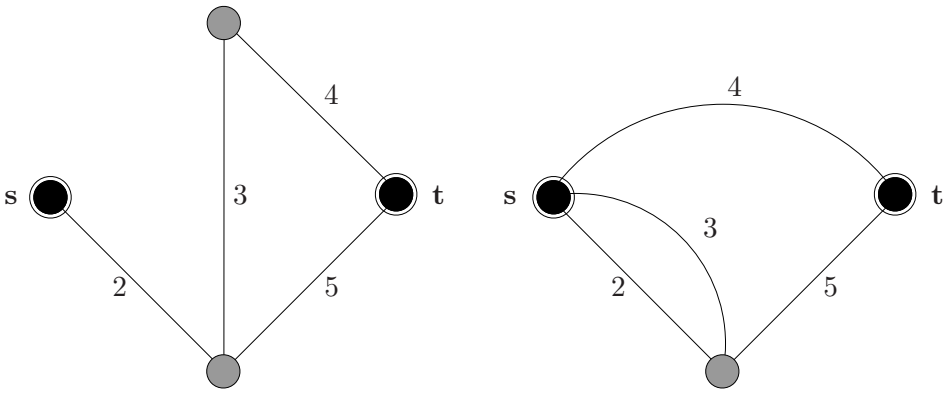


Figure 5.4: Bridge Network transformed according with $X_1 = 0$ and $X_1 = 1$

Replacing $\mathbb{E}\{I|X_1 = 0\}$ and $\mathbb{E}\{I|X_1 = 1\}$ in expression (5.2), the resulting value of $\mathbb{E}\{\mathbb{E}\{I|X_1\}\}$ is —as expected— the same as the one obtained in (5.1) for $\mathbb{P}\{\theta\}$.

As the variable $\mathbb{E}\{I|X_1\}$ is a discrete random variable, with two values whose probabilities

are known, the variance analysis is quite simple:

$$\mathbb{V}\{\mathbb{E}\{I|X_1\}\} \;=\; \mathbb{E}\{\mathbb{E}\{I|X_1\}^2\} - \mathbb{E}\{\mathbb{E}\{I|X_1\}\}^2,$$

where:

$$\begin{aligned}\mathbb{E}\{\mathbb{E}\{I|X_1\}^2\} &= \mathbb{E}\{I|X_1=0\}^2 \times q_1 + \mathbb{E}\{I|X_1=1\}^2 \times (1-q_1)\\ &= (q_2 + q_4q_5 + q_3q_5 - q_2q_4q_5 - q_2q_3q_5 - q_3q_4q_5 + q_2q_3q_4q_5)^2 \times q_1 +\\ &\quad (q_2q_3q_4 + q_4q_5 - q_2q_3q_4q_5)^2 \times (1-q_1)\end{aligned}$$

and:

$$\begin{aligned}\mathbb{E}\{\mathbb{E}\{I|X_1\}\}^2 &= (q_1q_2 + q_4q_5 + q_1q_3q_5 + q_2q_3q_4 - q_1q_2q_4q_5 - q_1q_2q_3q_5 -\\ &\quad q_1q_2q_3q_4 - q_1q_3q_4q_5 - q_2q_3q_4q_5 + 2\,q_1q_2q_3q_4q_5)^2.\end{aligned}$$

Table 5.5 shows $\gamma = \mathbb{E}\{\mathbb{E}\{I|X_1\}\}$ and $\mathbb{V}\{\mathbb{E}\{I|X_1\}\}$ for a set of values of q, in the particular case where $q_i = q\ \forall i$.

Table 5.5: Analysis of $\mathbb{E}\{I|X_1\}$ for the simple network (case 1)

q	$\gamma = \mathbb{E}\{\mathbb{E}\{I\|X_1\}\}$	$\mathbb{V}\{\mathbb{E}\{I\|X_1\}\}$
1	1.00000E+00	0.00000E+00
0.1	2.15200E−02	1.01506E−03
0.01	2.01950E−04	1.00910E−06
0.001	2.00200E−06	1.00099E−09
0.0001	2.00020E−08	1.00010E−12
0.00001	2.00002E−10	1.00001E−15
0.000001	2.00000E−12	1.00000E−18

A measure of accuracy increase for this problem is shown in Table 5.6, for the set of values of q used in previous tables.

Table 5.6: Accuracy increase in the simple network (case 1)

q	γ	$\mathbb{V}\{I\}/\mathbb{V}\{\mathbb{E}\{I\|X_1\}\}$
1	1.00000E+00	—
0.1	2.15200E−02	20.74
0.01	2.01950E−04	200.09
0.001	2.00200E−06	2,000.01
0.0001	2.00020E−08	20,000.00
0.00001	2.00002E−10	200,000.00
0.000001	2.00000E−12	2,000,000.00

5.4 Analysis of a simple network (case 2)

The same network is now analyzed fixing the value of two variables instead of one, say X_1 and X_5. The possible combination of values of X_1 and X_5 with the corresponding probabilities are:

$$\begin{aligned}
\mathbb{P}\{X_1 = 0, X_5 = 0\} &= q_1 \times q_5, \\
\mathbb{P}\{X_1 = 0, X_5 = 1\} &= q_1 \times r_5, \\
\mathbb{P}\{X_1 = 1, X_5 = 0\} &= r_1 \times q_5, \\
\mathbb{P}\{X_1 = 1, X_5 = 1\} &= r_1 \times r_5.
\end{aligned}$$

For each combination the Bridge Network takes a different form, as shown in Figure 5.5.

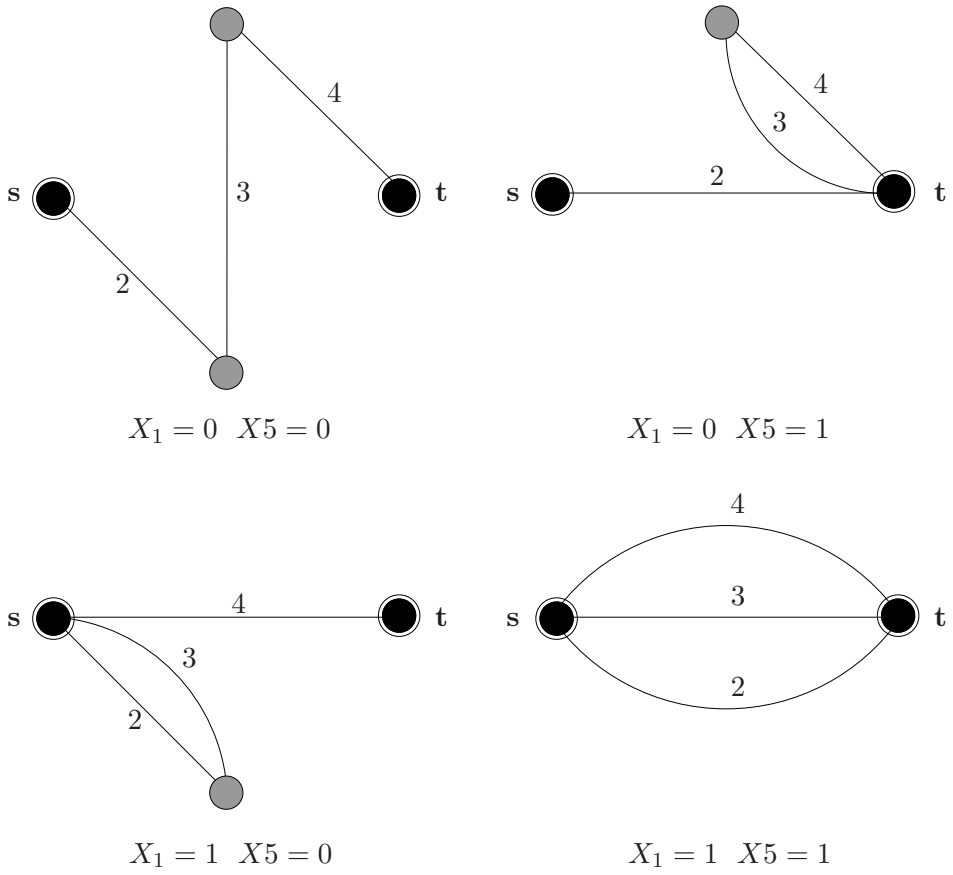


Figure 5.5: Bridge Network transformed according with the values of X_1 and X_5

$\mathbb{E}\{I\}$ conditioned to the four possible values of the pair X_1, X_5 results in the following expression:

$$\begin{aligned}
\mathbb{E}\{\mathbb{E}\{I|X_1, X_5\}\} &= \mathbb{E}\{\mathbb{E}\{I|X_1 = 0, X_5 = 0\}\} \times q_1 \times q_5 + \\
&\quad \mathbb{E}\{\mathbb{E}\{I|X_1 = 0, X_5 = 1\}\} \times q_1 \times r_5 +
\end{aligned}$$

$$\begin{aligned} & \mathbb{E}\{\mathbb{E}\{I|X_1=1,X_5=0\}\}\times r_1\times q_5+\\ & \mathbb{E}\{\mathbb{E}\{I|X_1=1,X_5=1\}\}\times r_1\times r_5\\ = \ & \mathbb{E}\{\mathbb{E}\{I|X_1=0,X_5=0\}\}\times q_1\times q_5+\\ & \mathbb{E}\{\mathbb{E}\{I|X_1=0,X_5=1\}\}\times q_1\times (1-q_5)+\\ & \mathbb{E}\{\mathbb{E}\{I|X_1=1,X_5=0\}\}\times (1-q_1)\times q_5+\\ & \mathbb{E}\{\mathbb{E}\{I|X_1=1,X_5=1\}\}\times (1-q_1)\times (1-q_5), \end{aligned}$$

where, after a basic network analysis (inclusion–exclusion principle to the mincuts) on each of the networks of Figure 5.5.

$$\begin{aligned} \mathbb{E}\{\mathbb{E}\{I|X_1,X_5\}\} \ = \ & (q_2+q_3+q_4-q_2q_3-q_3q_4-q_2q_4-q_2q_3q_4)\times q_1\times q_5+\\ & (q_2)\times q_1\times (1-q_5)+\\ & (q_4)\times (1-q_1)\times q_5+\\ & (q_2q_3q_4)\times (1-q_1)\times (1-q_5). \end{aligned}$$

This last expression agrees with the value of $\mathbb{P}\{\theta\}$ in (5.1) and it is the basis to determine the variance of $\mathbb{E}\{I|X_1,X_5\}$ in the following way:

$$\mathbb{V}\{\mathbb{E}\{I|X_1,X_5\}\} \ = \ \mathbb{E}\{\mathbb{E}\{I|X_1,X_5\}^2\}-\mathbb{E}\{\mathbb{E}\{I|X_1,X_5\}\}^2,$$

where:

$$\begin{aligned} \mathbb{E}\{\mathbb{E}\{I|X_1,X_5\}^2\} \ = \ & (q_2+q_3+q_4-q_2q_3-q_3q_4-q_2q_4-q_2q_3q_4)^2\times q_1\times q_5+\\ & (q_2)^2\times q_1\times (1-q_5)+\\ & (q_4)^2\times (1-q_1)\times q_5+\\ & (q_2q_3q_4)^2\times (1-q_1)\times (1-q_5) \end{aligned}$$

and

$$\begin{aligned} \mathbb{E}\{\mathbb{E}\{I|X_1,X_5\}\}^2 \ = \ & (q_1q_2+q_4q_5+q_1q_3q_5+q_2q_3q_4-q_1q_2q_4q_5-q_1q_2q_3q_5-\\ & q_1q_2q_3q_4-q_1q_3q_4q_5-q_2q_3q_4q_5+2\,q_1q_2q_3q_4q_5)^2. \end{aligned}$$

Table 5.7 shows $\gamma=\mathbb{E}\{\mathbb{E}\{I|X_1\}\}$ and $\mathbb{V}\{\mathbb{E}\{I|X_1\}\}$ for a set of values of q, in the particular case where $q_i=q\ \forall i$.

A measure of accuracy increase for this problem is shown in Table 5.8, for the set of values of q used in previous tables.

5.5 Remarks on Conditional Monte Carlo

As seen in all preceding examples, the exact calculation of γ usually involves a whole set of variables $\{X_i\}_{i=1}^n$. Depending on the type and the size of the problem, this exact calculation may be either impossible or too hard. However, if the exact calculation of γ becomes easier after

Table 5.7: Analysis of $\mathbb{E}\{I|X_1, X_5\}$ for the simple network (case 2)

q	$\gamma = \mathbb{E}\{\mathbb{E}\{I\|X_1, X_5\}\}$	$\mathbb{V}\{\mathbb{E}\{I\|X_1, X_5\}\}$
1	1.00000E+00	0.00000E+00
0.1	2.15200E−02	2.06131E−03
0.01	2.01950E−04	2.02742E−06
0.001	2.00200E−06	2.00297E−09
0.0001	2.00020E−08	2.00030E−12
0.00001	2.00002E−10	2.00003E−15
0.000001	2.00000E−12	2.00000E−18

Table 5.8: Accuracy increase in the simple network (case 2)

q	γ	$\mathbb{V}\{I\}/\mathbb{V}\{\mathbb{E}\{I\|X_1, X_5\}\}$
1	1.00000E+00	—
0.1	2.15200E−02	10.22
0.01	2.01950E−04	99.59
0.001	2.00200E−06	999.51
0.0001	2.00020E−08	9,999.50
0.00001	2.00002E−10	99,999.50
0.000001	2.00000E−12	999,999.50

fixing the value of one or more of the variables $\{X_i\}_{i=1}^n$, *Conditional Monte Carlo* can be a useful simulation option to produce accurate estimations of γ.

In the first step of *Conditional Monte Carlo*, the variables to be fixed are sampled and then, given their sampled values, γ is calculated. The original problem of *only calculation* is transformed into *estimation + exact calculation*. The difficulty level of these two problems, and the size of each one of them are extremely related to the precision and the efficiency increase of *Conditional Monte Carlo* over *Crude Monte Carlo*.

The Bridge Network case is an illustrating example. The exact computation of γ, given by expression (5.1), does not need to use the random values (0 or 1) of the variables X_i, but only their probabilities. On the other hand, the *Crude Monte Carlo* estimation of γ, where all the variables are considered by means of their random values is another another possible determination in which there is no exact calculation at all. *Conditional Monte Carlo* estimation is an intermediate option between these two, as it is not pure simulation, but not exact calculation either. In *Conditional Monte Carlo* the expectation of I can be conditioned to the value of k of the n variables, with $k = 0, \cdots, n$. In Example 3, $k = 1$, whereas in Example 4, $k = 2$. The case of $k = 0$ is the pure exact calculation (expression (5.1)). If $k = 1$ the value of γ is estimated sampling the value of one variable and then, making an exact calculation on a network formed by four links. If $k = 2$, two variables are sampled and and then, an exact calculation is made on a network formed by three links. As k grows, *exact calculation* becomes smaller and *estimation* becomes larger, explaining why the variance of the corresponding estimate grows together with k.

The reason why *Conditional Monte Carlo* yields a variance reduction comes from expression (2.5). However, this fact can also be shown intuitively. See, for example, the case of the Bridge

Network with $k = 1$. Once the selected random variable is sampled, say X_j, the *unreliability* of a network formed by four links is calculated exactly. Such exact calculation is equivalent to a weighted average made over the 2^4 possible configurations, i.e., over "all" the possible states of the four links network. But in a *Crude Monte Carlo* approach, not "all" these 2^4 possible configurations are considered every time X_j is sampled. In other words, the exact calculation of *Conditional Monte Carlo* covers "all" the possibilities in situations in which *Crude Monte Carlo* only covers "some" possibilities.

5.6 Conditional Monte Carlo on an M/M/1 Queueing System

In this section —and in the following ones, until the end of the chapter— an application of Conditional Monte Carlo on an M/M/1 queueing system is studied. It is a simple and straightforward application of the basic ideas of Conditional Monte Carlo [Kroese 2013, Ross 2006] over a well–known queueing system model. It is included just to illustrate the ideas behind *Conditional Monte Carlo* and also because it is a model of general interest.

Figure 5.6 shows the evolution of the number of customers, Q, in a trial of an M/M/1 queueing system (Q is the number of waiting customers plus the customer eventually being served, i.e., the total number of customers in the system). *Arrivals* and *departures* occur according with the arrival and service rates, respectively, λ and μ.

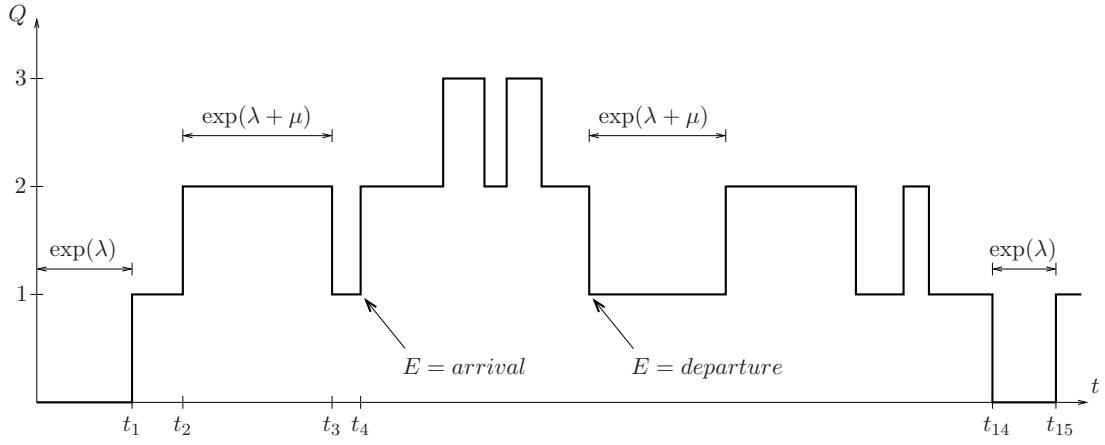


Figure 5.6: A trial of an M/M/1 Queueing System

In order to simulate the M/M/1 queueing system, some variables will be defined.

E**:** a discrete random variable that takes values on the events of the M/M/1 queueing system (*arrivals* and *departures*), with the following probability distribution:

$$\mathbb{P}\{E = arrival\} \quad = \quad \begin{cases} 1 & \text{if } Q = 0 \\ \dfrac{\lambda}{\lambda + \mu} & \text{otherwise} \end{cases}$$

$$\mathbb{P}\{E = departure\} \quad = \quad \begin{cases} 0 & \text{if } Q = 0 \\ \dfrac{\mu}{\lambda + \mu} & \text{otherwise} \end{cases}$$

δ**:** a random variable that takes values on the times between events of the M/M/1 queueing system, according to the value of Q:

$$\delta \sim \begin{cases} \exp(\lambda) & \text{if } Q = 0, \\ \exp(\lambda + \mu) & \text{otherwise.} \end{cases}$$

The M/M/1 queueing system can be simulated with the following code (the operation $v \leftarrow V$ means: *"a sample of the random variable V is saved into the variable v"*):

```
01  Q = 0 ;  t = 0
02  if Q = 0
03     t = t + exp(λ)
04     Q = Q + 1
05  else
06     t = t + exp(λ + μ)
07     e ← E
08     if e = arrival
09        Q = Q + 1
10     else
11        Q = Q − 1
12  if stopping conditions are satisfied
13     exit
14  else
15     go to line 02
```

The execution of this code produces —among others— two sequences of values. One of them composed by the list of events (*arrivals* and *departures*) occurring throughout the simulation. In the replication shown in Figure 5.6 this sequence is:

$$x = \{arrival, arrival, departure, \cdots\}$$

The other one is composed by the times between events, actually, the exponentially distributed times sampled in every iteration. In the replication shown in Figure 5.6 this sequence is:

$$y = \{t_1, t_2 - t_1, t_3 - t_2, \cdots\}$$

The elements of both sequences are related, *one–to–one*, in the following way: the first element in y is the occurrence time of the first event in x, the second element in y is the elapsed time between the fisrt and the second events in x, and so on.

Each of these sequences can be seen as the replication of a random process. The first one is a replication of process X, a *random walk* restricted to non negative values:

$$X = \{X_n,\ n \geq 0\},$$

where

$$X_n = \left\{ \begin{array}{ll} arrival & \text{if } Q = 0 \\ arrival & \text{with probability } \lambda/(\lambda+\mu) \\ departure & \text{with probability } \mu/(\lambda+\mu) \end{array} \right\} \text{ otherwise,} \qquad n = 1, 2, \cdots .$$

The second one is a replication of process Y, a sequence of two —interleaved— exponentially distributed random variables:

$$Y = \{Y_n,\ n \geq 0\},$$

where

$$Y_n \sim \left\{ \begin{array}{ll} \exp(\lambda) & \text{if } Q = 0, \\ \exp(\lambda+\mu) & \text{otherwise,} \end{array} \right. \qquad n = 1, 2, \cdots .$$

Some parameters of the M/M/1 queueing system, like the number of customers at time t, or the average time that a customer spends in the system, etc., can be seen as a function of these two random processes. In particular, events θ defined on the state spaces of these parameters, and their corresponding indicator random variables I, can also be seen as functions of processes X and Y.

5.6.1 Busy Period

A time interval in which the server is permanently busy, is called a Busy Period (*BP*). In the replication shown in Figure 5.6, $[t_1, t_{14}]$ is a *BP*. Within a *BP*, $Q > 0$, reason why X and Y

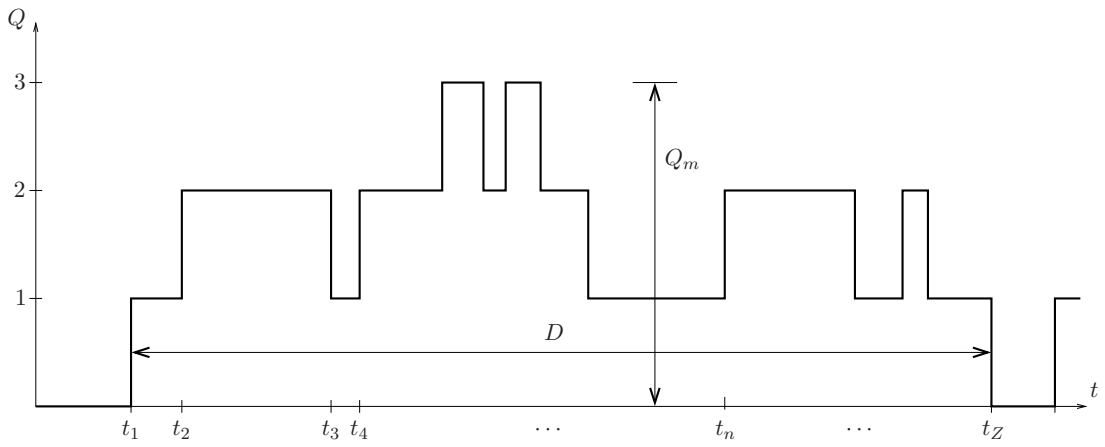


Figure 5.7: M/M/1 *Busy Period*

become:

$$\begin{array}{rcll} X & = & \{X_n,\ 1 < n \leq Z\} & X_n = \left\{ \begin{array}{ll} arrival & \text{with probability } \lambda/(\lambda+\mu), \\ departure & \text{with probability } \mu/(\lambda+\mu), \end{array} \right. \\ Y & = & \{Y_n,\ 1 < n \leq Z\} & Y_n = \exp(\lambda+\mu). \end{array}$$

Variables R, D and Q_m (see Figure 5.7) can be defined for a *BP*:

$$\begin{aligned} R &= \#\text{ events within the } BP = Z - 1, \\ D &= \text{ duration of the } BP, \\ Q_m &= \text{ maximum number of customers in the } BP. \end{aligned}$$

An M/M/1 simulation in which N *BP*'s occur, produces N replications of all the variables just defined: $X^{(i)}, Y^{(i)}, R^{(i)}, D^{(i)}, Q_m^{(i)},\ i = 1, \cdots, N$. In the following examples the estimation of the probability of events defined on the state space of two of theses variables —respectively, D and Q_m— will be analyzed in the context of *Crude* and *Conditional Monte Carlo*.

5.6.2 Duration of a Busy Period

Let θ be the event $D \geq T$, where T is some fixed real value, and let I be the indicator random variable of event θ. *Crude* or *standard* simulation to estimate $\gamma = \mathbb{P}\{\theta\}$ can be done as follows:

1. Simulate the M/M/1 queueing system up to some —long enough— time t_{max}.
2. Identify the replications $X^{(i)},\ i = 1, \cdots, N$ to conclude that N *BP*'s occurred.
3. For every *BP* detected, check the corresponding replication $Y^{(i)}$ to see whether the duration of the *BP* exceeds T or not.
4. Estimate $\gamma = \mathbb{P}\{\theta\}$ as the proportion of *BP*'s for which their duration exceeds T.

This estimation, that is actually an estimation of $\mathbb{E}\{I\}$, is based on the randomness of both sequences, in other words, it is a function of random processes X and Y. In the next section this estimation will be conditioned to a fixed replication of process X. Hence, γ will be estimated by $\mathbb{E}\{\mathbb{E}\{I|X\}\}$.

5.6.3 Duration of a Busy Period - Conditioning to process X

Figure 5.8 illustrates this problem. Pretend that Figure 5.8 (a) is a *BP* detected in the M/M/1 system simulation. It is necessary to know the probability that this *BP* lasts longer than T, subject to the sequence of *arrivals* and *departures* shown. The duration of this *BP* is D_1, that is clearly larger than T. However this measure is not relevant, as the replications shown in Figures 5.8 (b) and (c) also share the same sequence of *arrivals* and *departures*, but their durations are different, being one of them larger, and the other one shorter than T. The probability that any *BP* that share this sequence of *arrivals* and *departures* lasts longer than T, covers any of the replications shown in Figures 5.8 (a), (b) and (c). Actually, this probability covers "all" possible replications that share this sequence of *arrivals* and *departures*. Compared to the *Crude Monte Carlo* estimation, this probability is like averaging all the possible *BP*'s with this sequence of *arrivals* and *departures* (infinite, in this case). In other words, one sample of *Conditional Monte Carlo* is equivalent to infinite *Crude Monte Carlo* samples.

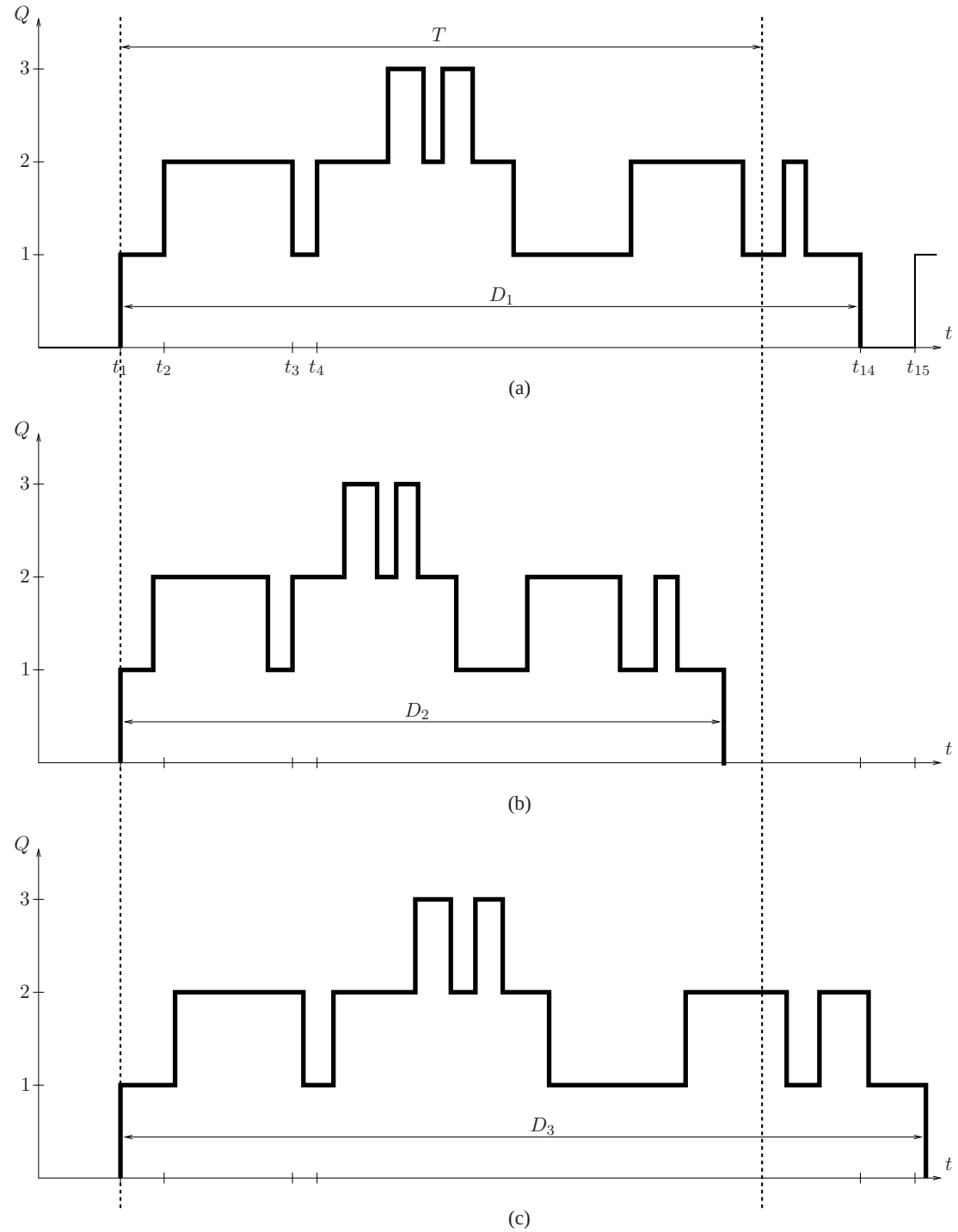


Figure 5.8: Three *BP*'s with the same sequence $X^{(i)}$, but different sequence $Y^{(i)}$

In summary, the estimation of $\gamma = \mathbb{P}\{\theta\}$, as a *Conditional Monte Carlo* estimation obtained by means of $\mathbb{E}\{I|X\}$, can be done as follows:

1. Simulate the M/M/1 queueing system up to some —long enough— time t_{max}.
2. Identify the replications $X^{(i)},\ i = 1, \cdots, N$ to conclude that N *BP*'s occurred.
3. For every *BP* detected, calculate the probability that the sum of the R times between events within the *BP* is larger than T.
4. Estimate $\gamma = \mathbb{P}\{\theta\}$ as the average of the probabilities calculated in item 3.

The most difficult part of this estimation is the probability calculation of item 3. The duration of the *BP*'s detected troughout the simulation is the sum of R exponentially distributed times with rate $(\lambda + \mu)$. Then, due to expression (5.4) in Section 5.6.6, the duration of a *BP* with R times between events, is distributed according to the Erlang distribution with parameters $(R, \lambda + \mu)$.

Table 5.9 shows the evolution of the *speedup*, $(\mathbb{V} \times t)_{Cr}/(\mathbb{V} \times t)_{Cond}$, for a set of values of T.

Table 5.9: *Speedup* of the estimation of the probability of $BP > T$, given the sequence $X^{(i)}$

λ	μ	T	$\widehat{\gamma}$	$(\mathbb{V} \times t)_{Cr}/(\mathbb{V} \times t)_{Cond}$
0.01	1.00	2.00	1.38E−01	26.61
0.01	1.00	4.00	1.98E−02	24.85
0.01	1.00	6.00	2.94E−03	40.27
0.01	1.00	8.00	4.52E−04	78.74
0.01	1.00	10.00	7.13E−05	161.79
0.01	1.00	12.00	1.15E−05	325.11
0.01	1.00	14.00	1.91E−06	773.67
0.01	1.00	16.00	3.20E−07	1,365.76

5.6.4 Maximum in a Busy Period

Let θ be the event $Q_m \geq M$, where M is some fixed integer value, and let I be the indicator random variable of event θ. *Crude* or *standard* simulation to estimate $\gamma = \mathbb{P}\{\theta\}$ can be done as follows:

1. Simulate the M/M/1 queueing system up to some —long enough— time t_{max}.
2. Identify the replications $X^{(i)},\ i = 1, \cdots, N$ to conclude that N *BP*'s occurred.
3. For every *BP* detected, check the corresponding piece of sequence in $X^{(i)}$ to see whether the value of Q within the *BP* exceeds M or not.
4. Estimate $\gamma = \mathbb{P}\{\theta\}$ as the proportion of *BP*'s for which the maximum value of Q exceeds M.

This estimation, that is actually an estimation of $\mathbb{E}\{I\}$, is based on the randomness of both sequences, in other words, it is a function of random processes X and Y. In the next section this estimation will be conditioned to a fixed replication of process Y. Hence, γ will be estimated by $\mathbb{E}\{\mathbb{E}\{I|Y\}\}$.

5.6.5 Maximum in a Busy Period - Conditioning to process Y

Figure 5.9 illustrates this problem. Pretend that Figure 5.9 (a) is a *BP* detected in the M/M/1 system simulation. It is necessary to know the probability that the number of customers Q within this *BP* is larger or equal than M, subject to the sequence of times between events shown. In the replication shown in 5.9 (a), Q is clearly larger than M. However this measure is not relevant, as the replications shown in Figures 5.9 (b) and (c) also share the same sequence of *arrivals* and *departures*, but in the case (b) Q is not larger than M while in case (c), it is. The probability that for any *BP* that share this sequence of times between events, Q is larger than M, covers any of the replications shown in Figures 5.8 (a), (b) and (c). Actually, this probability covers "all" possible replications that share this sequence of times between events. Compared to the *Crude Monte Carlo* estimation, this probability is like averaging all the possible *BP*'s with this sequence of. In other words, one sample of *Conditional Monte Carlo* is equivalent to "many" *Crude Monte Carlo* samples.

In summary, the estimation of $\gamma = \mathbb{P}\{\theta\}$, as a *Conditional Monte Carlo* estimation obtained by means of $\mathbb{E}\{I|Y\}$, can be done as follows:

1. Simulate the M/M/1 queueing system up to some —long enough— time t_{max}.
2. Identify the replications $X^{(i)},\ i = 1, \cdots, N$ to conclude that N *BP*'s occurred.
3. For every *BP* detected, calculate the probability that the combination of *arrivals* and *departures* is such that value of Q within the *BP* exceeds M.
4. Estimate $\gamma = \mathbb{P}\{\theta\}$ as the average of the probabilities calculated in the item 3.

The most difficult part of this estimation is the probability calculation of item 3. For a given replication $Y^{(i)}$, i.e. given the times between events (*arrivals* and *departures*) the evolution of Q within a BP follow trajectories of the type of the *Dyck Paths* (see Section 5.6.7).

Table 5.10 shows the evolution of the *speedup*, $(\mathbb{V} \times t)_{Cr}/(\mathbb{V} \times t)_{Cond}$, for a set of values of M.

Table 5.10: *Speedup* of the estimation of the probability of $Q > M$, given the sequence $Y^{(i)}$

λ	μ	M	$\widehat{\gamma}$	$(\mathbb{V} \times t)_{Cr}/(\mathbb{V} \times t)_{Cond}$
0.05	1.00	2.00	4.76E−02	0,90
0.05	1.00	3.00	2.38E−03	1,75
0.05	1.00	4.00	1.19E−04	3,45
0.05	1.00	5.00	5.97E−06	8,15
0.05	1.00	6.00	3.01E−07	16,16
0.05	1.00	7.00	1.53E−08	33,10
0.05	1.00	8.00	9.11E−10	98,90

5.6.6 Sum of Exponentials

When the simulation is conditioned to processes X, the combination of *arrivals* and *departures* is dispensable. The only values to care about, in every *BP*, are the times between events, specifically

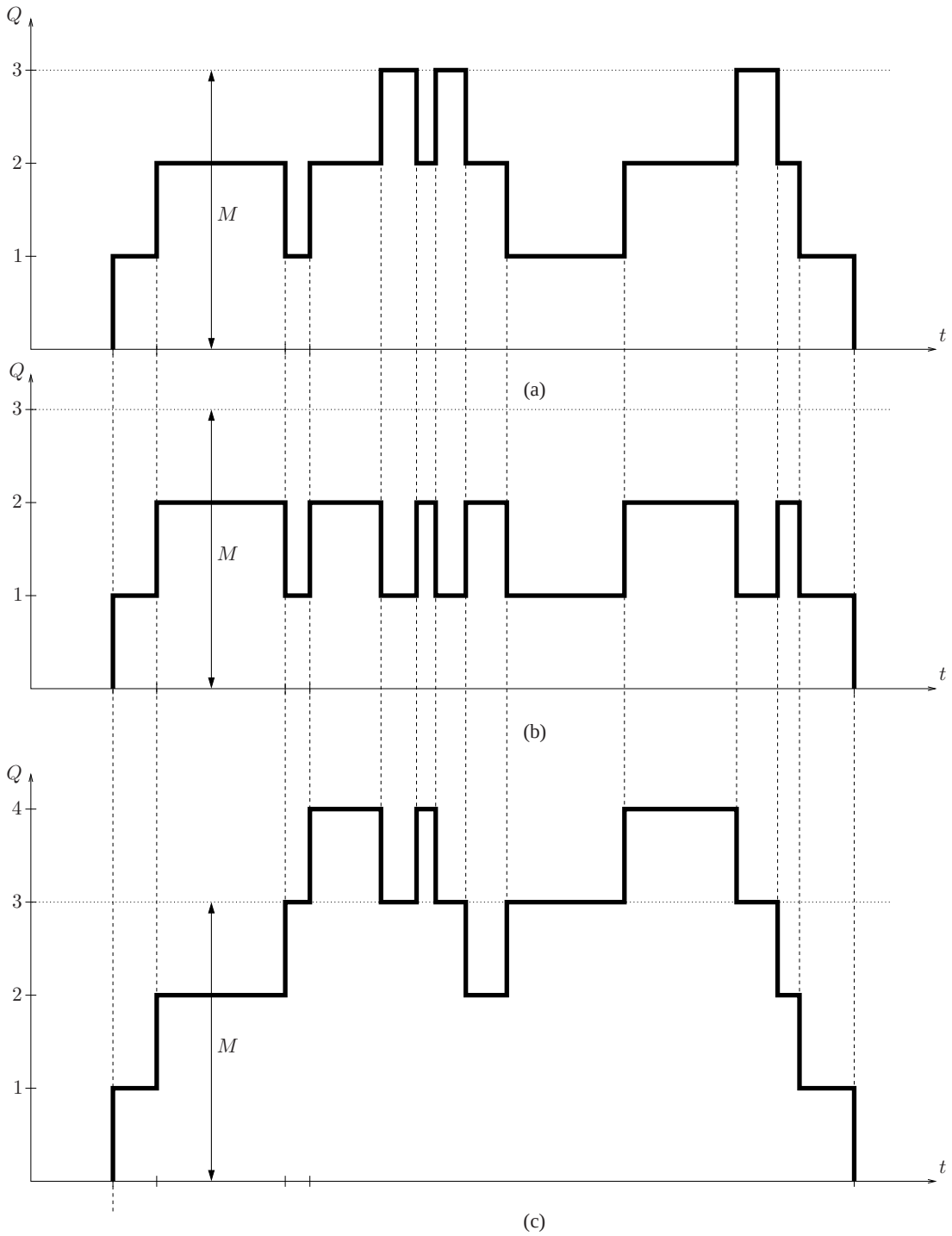


Figure 5.9: Three *BP*'s with the same sequence $Y^{(i)}$, but different sequence $X^{(i)}$

the sum of all these times within the *BP*. In Figures 5.8 (a), (b) and (c), this sum is, respectively, D_1, D_2 and D_3. The aim is to determine the probability that a *BP* with the same combination of

arrivals and *departures*, but with any other set of times between events, lasts longer than T. In general, the aim is to determine the proportion of paths for which some detected *BP* lasts longer than some fixed value, given the same combination of *arrivals* and *departures* but a different set of times between events.

Numerically, the problem is to find the probability that, for any *BP*, the sum of the times between events (all of them exponentially distributed with rate $\lambda + \mu$, as shown in (5.4)) is larger than some fixed value, T. As R is the number of exponentially distributed times, their sum is distributed according to the Erlang distribution, for which the cdf is:

$$1 - \sum_{n=0}^{R-1} \frac{e^{-(\lambda+\mu)\,t}\,((\lambda+\mu)\,t)^n}{n!} \tag{5.3}$$

Thus, the probability that such sum is larger than T is:

$$\sum_{n=0}^{R-1} \frac{e^{-(\lambda+\mu)\,T}\,((\lambda+\mu)\,T)^n}{n!} \tag{5.4}$$

5.6.7 Dyck Paths

For every sequence of instants at which *arrivals* and *departures* occur in a *BP*, there are —in general— many different sequences of *arrivals* and *departures*. Each one of them can define a *BP* of the same duration but with a different combination of *arrivals* and *departures*. The number of possible sequences of *arrivals* and *departures*, for any given sequence of instants, can be determined by means of a combinatorial analysis based on the concept of *Dyck Path*, a subject named after the German mathematician Walther Franz Anton von Dyck (1856–1934), who introduced it around 1880. In this section, this concept and some related problems, taken from [Flajolet 2009], are applied to the analysis of sequences of *arrivals* and *departures*.

Let $S = \{S_n : n = 0, 1, \cdots\}$ be a random process, on the non–negative integers, that starts at $S_0 = 0$ such that, at the beginning it goes from $S_0 = 0$ to $S_1 = 1$ with probability 1 and then, at every step, it increases 1 with probability p or decreases 1 with probability $q = 1 - p$. The process ends any time it returns to 0.

A *cycle of duration Z* is a replication that starts at $S_0 = 0$ and returns to 0 in exactly Z steps. C_Z is the set of all the cycles of duration Z:

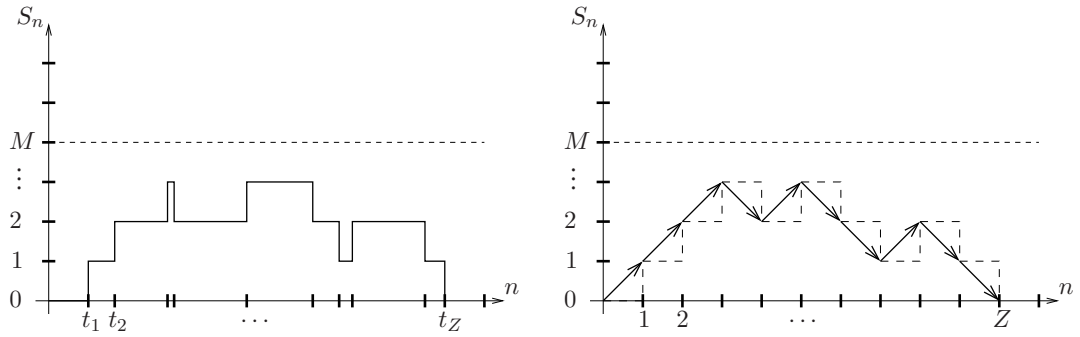
$$C_Z = \{S : S_0 = 0 \wedge S_Z = 0 \wedge S_n > 0\, \forall n : 1 < n < Z\}.$$

A cycle $S \in C_Z$ is said to be less than $M > 0$, and it is written $S < M$, if $S_n < M\ \forall n$, $0 \leq n \leq Z$. P_Z is the probability that a replication in C_Z is less than $M > 0$:

$$P_Z = \mathbb{P}(S < M | S \in C_Z).$$

Some remarks:

- For every cycle in C_Z, Z is even, and there will be $h = Z/2$ upward steps and h downward steps. Therefore, the probability of all possible trajectories in C_Z is $p^{h-1}q^h$ (the probability



of the initial step, from $S_0 = 0$ to $S_1 = 1$, is 1).

- As the probability of all the trajectories in C_Z is the same, the values of p and q are not necessary in the determination of P_Z. This value can be obtained as the ratio between the number, $N_{Z,M}$, of trajectories in C_Z that are less than M and the number, N_Z, of all the trajectories in C_Z.

- The determination of P_Z is only of interest if $1 \leq M \leq h$, because if $M = 0$, $P_Z = 0$, while if $M > h$, $P_Z = 1$.

If the problem is modified, and the replications are allowed to "touch" 0 before reaching Z, the corresponding set of *cycles of duration Z* becomes:

$$C_Z^0 = \{S : S_0 = 0 \wedge S_Z = 0 \wedge S_n \geq 0 \, \forall n : 1 < n < Z\}.$$

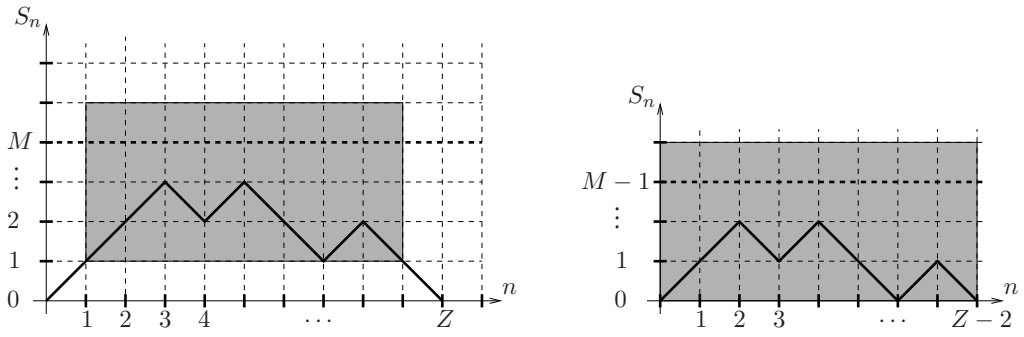
In this case the number of possible trajectories, i.e. the cardinality of C_Z^0, is the *Catalan Number* in $h = Z/2$, that is:

$$N_Z^0 = \binom{2h}{h} \frac{1}{h+1}, \qquad h = Z/2.$$

A cycle $S \in C_Z^0$ is said to be less than $M > 0$, and it is written $S < M$, if $S_n < M \; \forall n$, $0 \leq n \leq Z$. The number of trajectories in C_Z^0 that are less than $M > 0$, with $1 \leq M \leq h$, is:

$$\begin{aligned} N_{Z,M}^0 &= \sum_{s=1}^{M} \left(\frac{2}{M+1}\right) \sin^2\left(\frac{s\pi}{M+1}\right) \left(2\cos\left(\frac{s\pi}{M+1}\right)\right)^{2h} \\ &= \frac{2^{2h+1}}{M+1} \sum_{s=1}^{M} \sin^2\left(\frac{s\pi}{M+1}\right) \cos^{2h}\left(\frac{s\pi}{M+1}\right). \end{aligned}$$

With the aid of the following figure it is possible analyze the first problem ($S_n > 0$, $0 < n < Z$) in terms of the last one ($S_n \geq 0$, $0 < n < Z$):



N_Z (the cardinality of C_Z) can be expressed in terms of N_Z^0 (the cardinality of C_Z^0), just replacing the value of h (because Z has to be changed by $Z-2$), then:

$$N_Z = \binom{2h}{h}\frac{1}{h+1}, \qquad h = (Z-2)/2.$$

Proceeding in the same way, $N_{Z,M}$ can be obtained in terms of $N_{Z,M}^0$, replacing M by $M-1$:

$$N_{Z,M} = \frac{2^{2h+1}}{M}\sum_{s=1}^{M-1}\sin^2\left(\frac{s\pi}{M+1}\right)\cos^{2h}\left(\frac{s\pi}{M+1}\right), \qquad h = (Z-2)/2.$$

The case of $M=0$ must be avoided. However, this case is not of interest. Finally:

$$\begin{aligned} P_Z &= \mathbb{P}(S < M | S \in C_Z) = \frac{N_{Z,M}}{N_Z}, \\ \bar{P}_Z &= \mathbb{P}(S \geq M | S \in C_Z) = 1 - \frac{N_{Z,M}}{N_Z}. \end{aligned}$$

CHAPTER 6

Conditional Monte Carlo on Markovian Systems

Contents

Abstract

Conditional Monte Carlo with Intermediate Estimations (*CMIE*) is a simulation method proposal of this thesis to estimate the *reliability* of highly reliable Markovian systems accurately. In this chapter the basis of *CMIE* is introduced, the unbiasedness of the corresponding estimator is proven, and its variance is shown to be lower than the variance of the standard estimator. Some guidelines on the choice of the intermediate states are given and a modification to the basic scheme, in order to be applied to large multicomponent systems, is proposed. To illustrate the performance of the method, some experimental results are shown.

6.1 Introduction

The dynamics of many systems can be modelled by a continuous time homogeneous Markov chain X, irreducible, on a finite state space S (see [Cancela 2002, Goyal 1992, Juneja 2001] or Chapter 6 in [Rubino 2009]). Typical examples are systems in which the states can be classified

into two subspaces: one of them, D, in which the system is completely *failed* (*down*), and another one, U, in which the system is *operational* (*up*), assuming that at some states in U the system is fully *operational* whereas at some other the system is partially damaged but still operating, i.e. delivering service. In this thesis they are of interest highly reliable systems whose behaviour, in terms of these subspaces, is such that they strongly tend to remain at a state in U where the system is fully *operational*, they rarely leave this state moving within some other states in U due to the occurrence of failures, and they reach the subspace D with an extremely low probability.

The estimation of *dependability* measures on these type of system has been addressed extensively with the aid of *Importance Sampling* [Botev 2013, Cancela 2002, Glynn 1989] and its variants, Zero-Variance [L'Ecuyer 2007b, L'Ecuyer 2011a, L'Ecuyer 2011b] and Cross-Entropy [Ridder 2005, Rubinstein 2004].

In the rest of this chapter this problem is approached by means of a novel application of *Conditional Monte Carlo*, a classic variance reduction technique, already introduced in Chapter 5, that has not given rise to many lines of research in the field of *rare event* estimation. The method proposed is aimed at the estimation of the probability $\gamma = \mathbb{P}\{\tau_D < \tau_{\mathbf{u}}\}$, where the times $\tau_{\mathbf{u}}$ and τ_D are defined as follows. The state space of the Markov chain is partitioned as $S = U \cup D$ such that in U the system is *up* and in D the system is *down*. The process X starts at some initial state $\mathbf{u} \in U$. Define $\tau_{\mathbf{u}}$ as the *return time to* $\mathbf{u}$, that is, $\tau_{\mathbf{u}} = \inf\{t > 0\colon X(t) = \mathbf{u} \text{ and } X(t^-) \neq \mathbf{u}\}$, and τ_D as the *hitting time of* D, that is, $\tau_D = \inf\{t > 0\colon X(t) \in D\}$.

The simplest and most basic dependability metric is the Mean Time To Failure, MTTF, defined as the expected life–time of the system, i.e. the mean time until the system enters the subset D:

$$\mathrm{MTTF} = \mathbb{E}\{\tau_D\}.$$

Goyal et al. have proved in [Goyal 1992] that the MTTF can be written as:

$$\mathrm{MTTF} = \mathbb{E}\{\min(\tau_D, \tau_{\mathbf{u}})\}/\gamma.$$

Since this work is focused on the estimation of γ, all D can be collapsed into a single state $\mathbf{d}$ made absorbing. As before, event $\{\tau_{\mathbf{d}} < \tau_{\mathbf{u}}\}$ means that X gets absorbed at $\mathbf{d}$ before coming back to $\mathbf{u}$.

For systems with a large (or infinite) number of states, the exact computation of γ is not feasible. An alternative solution is to employ *Monte Carlo* simulation, in which case the main concern is to attain a good estimation precision, what means to obtain a low variance estimator. In the standard *Monte Carlo* simulation, N replications are started from $\mathbf{u}$ and they are stopped either when they get absorbed at state $\mathbf{d}$ or when they arrive back at $\mathbf{u}$. The standard estimation of γ is the number of those trajectories that are absorbed at $\mathbf{d}$, divided by N. This estimation can be achieved by working with the discrete time Markov chain Y, canonically embedded in X at X's jump times.

However, the fact that $\gamma \ll 1$ is a serious drawback for standard simulation, because acceptable values of the estimator's variance can only be achieved at the expense of a very high number of replications, N. *Monte Carlo* methods must, therefore, be improved and adapted to address efficiently the *rare event* case.

A solution proposal to this problem, to be introduced next, consists in an application of *Conditional Monte Carlo* in which the rare event of interest, namely the visit of state $\mathbf{d}$, is conditioned

on the values of some random variable in the model.

The rest of the chapter is organized as follows. Section 6.2 shows a basic application of *Conditional Monte Carlo* on Markovian systems. Section 6.3 —the core of this proposal—, introduces modifications to the basic *Conditional Monte Carlo* algorithm, in order to make it usable and efficient. Sections 6.4, 6.5 and 6.6 discuss some properties and features of the proposed method. Section 6.7 shows how to apply it to the particular case of Markovian multicomponent systems. Some experimental results are included in Sections 6.6 and 6.7, and a comparison with *Splitting* is shown in Section 6.8.

6.2 Conditional Monte Carlo algorithm

There are different simulation methods to estimate value of γ. In the crude or standard simulation, N_1 replications start at state $\mathbf{u}$ and they are simulated until they either come back to $\mathbf{u}$, in time $\tau_{\mathbf{u}}$, or hit state $\mathbf{d}$, in time $\tau_{\mathbf{d}}$. Let I be the indicator random variable of the event $\{\tau_{\mathbf{d}} < \tau_{\mathbf{u}}\}$:

$$I = \begin{cases} 1 & \text{w.p.} \quad \gamma, \\ 0 & \text{w.p.} \quad 1-\gamma. \end{cases} \tag{6.1}$$

Then $\gamma = \mathbb{E}\{I\}$, and the standard estimator of γ is:

$$\widehat{\gamma}_s = \frac{1}{N_1} \sum_{j=1}^{N_1} I^{(j)}, \tag{6.2}$$

where $I^{(j)}, j = 1, 2, \ldots, N_1$ are N_1 independent values sampled from distribution (6.1).

Let $C = \{\mathbf{d}, k, \mathbf{u}\}$, where k is any state in the Markov chain, other than $\mathbf{d}$ or $\mathbf{u}$, and let X_C be a random variable defined as *the first state in C, hit by a replication started at* $\mathbf{u}$. The probability distribution of X_C is, therefore:

$$X_C = \begin{cases} \mathbf{d} & \text{w.p.} \quad p_{\mathbf{d}}, \\ k & \text{w.p.} \quad p_k, \\ \mathbf{u} & \text{w.p.} \quad p_{\mathbf{u}}. \end{cases}$$

See that $p_{\mathbf{d}} \leq \gamma$, because γ is the probability that any replication that starts at $\mathbf{u}$ reaches $\mathbf{d}$ before coming back to $\mathbf{u}$, whereas $p_{\mathbf{d}}$ is the probability that any replication that starts at $\mathbf{u}$ reaches $\mathbf{d}$ "through a path not containing k", before coming back to $\mathbf{u}$. Similarly, $p_{\mathbf{u}} \leq 1 - \gamma$.

The expectation of I, conditioned on the values of X_C, is given by the following expressions: $\mathbb{E}\{I \mid X_C = \mathbf{d}\} = 1$, $\mathbb{E}\{I \mid X_C = k\} = \gamma_k$ and $\mathbb{E}\{I \mid X_C = \mathbf{u}\} = 0$ (γ_k is the probability that a replication that starts at state k, hits state $\mathbf{d}$ before it hits state $\mathbf{u}$). Thus, $\mathbb{E}\{I \mid X_C\}$ is a random variable with the following probability distribution:

$$\mathbb{E}\{I \mid X_C\} = \begin{cases} 1 & \text{w.p.} \quad p_{\mathbf{d}}, \\ \gamma_k & \text{w.p.} \quad p_k, \\ 0 & \text{w.p.} \quad p_{\mathbf{u}}, \end{cases} \tag{6.3}$$

and the following expectation:

$$\mathbb{E}\{\mathbb{E}\{I \mid X_C\}\} = \mathbb{E}\{I\} = 1 \times p_{\mathbf{d}} + \gamma_k \times p_k + 0 \times p_{\mathbf{u}} = \gamma.$$

The expected value of both random variables, I and $\mathbb{E}\{I \mid X_C\}$, is γ. As a consequence, another estimator of γ —namely, a *Conditional Monte Carlo* estimator— is:

$$\widehat{\gamma}_c = \frac{1}{N_1} \sum_{j=1}^{N_1} \mathbb{E}\{I \mid X_C^{(j)}\} \tag{6.4}$$

where $\mathbb{E}\{I \mid X_C^{(j)}\}, j = 1, 2, \ldots$ are N_1 independent random variables sharing distribution (6.3). The samples $\mathbb{E}\{I \mid X_C^{(j)}\}$ are obtained in two steps: first, $X_C^{(j)}$ is sampled and then, the corresponding value $\mathbb{E}\{I \mid X_C^{(j)}\}$ is computed. In this introductory example the only three possible values to be sampled are $\{\mathbf{u}, k, \mathbf{d}\}$, whereas the exact values associated with them are, respectively, $\{1, \gamma_k, 0\}$.

If the set C includes more intermediate states besides k, the method applies as well. If, for example, $C = \{\mathbf{d}, 1, 2, \ldots, n, \mathbf{u}\}$, the distribution of $\mathbb{E}\{I \mid X_C\}$ becomes:

$$\mathbb{E}\{I \mid X_C\} = \begin{cases} 1 & \text{w.p.} \quad p_{\mathbf{d}}, \\ \gamma_1 & \text{w.p.} \quad p_1, \\ \gamma_2 & \text{w.p.} \quad p_2, \\ \vdots & \\ \gamma_n & \text{w.p.} \quad p_n, \\ 0 & \text{w.p.} \quad p_{\mathbf{u}}, \end{cases} \tag{6.5}$$

where γ_i is the probability that a replication that starts at state i hits state $\mathbf{d}$ before it hits state $\mathbf{u}$. Now:

$$\begin{aligned} \mathbb{E}\{\mathbb{E}\{I \mid X_C\}\} = \mathbb{E}\{I\} &= 1 \times p_{\mathbf{d}} + \sum_{i=1}^{n} \gamma_i \, p_i + 0 \times p_{\mathbf{u}} \\ &= \gamma_0 \times p_0 + \sum_{i=1}^{n} \gamma_i \, p_i \\ &= \sum_{i=0}^{n} \gamma_i \, p_i = \gamma, \end{aligned}$$

where the notation $\gamma_0 = 1$ and $p_0 = p_{\mathbf{d}}$ is included for simplicity. The estimator given in Expression (6.4) remains valid, with the only difference of sampling from the distribution (6.5) instead of (6.3).

Figure 6.1 depicts the set of probabilities so far defined and shows the nomenclature used to refer to them in the rest of this chapter (as $\gamma_{\mathbf{u}} = 0$, the term $p_{\mathbf{u}} \times \gamma_{\mathbf{u}}$ equals 0, reason why it is shown in Figure 6.1 but does not belong to any further expression).

For any given set $C = \{\mathbf{d}, 1, 2, \ldots, n, \mathbf{u}\}$, call $\widetilde{C} = C \setminus \{\mathbf{d}, \mathbf{u}\}$, i.e. the subset formed only by the intermediate states, that is, $\widetilde{C} = \{1, 2, \ldots, n\}$.

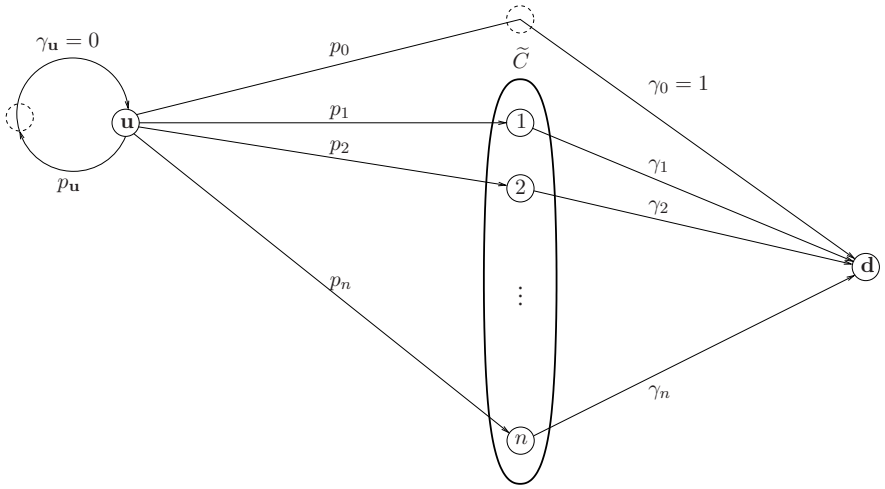


Figure 6.1: The set of probabilities used in all calculations.

The variance of the *Conditional Monte Carlo* estimator is now computed:

$$\mathbb{V}\{\mathbb{E}\{I \mid X_C\}\} = \mathbb{E}\{\mathbb{E}\{I \mid X_C\}^2\} - \mathbb{E}\{\mathbb{E}\{I \mid X_C\}\}^2 \; = \; \sum_{i=0}^{n} p_i \gamma_i^2 - \gamma^2.$$

Then, the variance of the estimator in (6.4) is:

$$\mathbb{V}\{\widehat{\gamma}_c\} = \frac{1}{N_1}\left(\sum_{i=0}^{n} p_i \gamma_i^2 - \gamma^2\right). \tag{6.6}$$

On the other hand, the variance of the standard estimator given in (6.2) is known to be:

$$\begin{aligned} \mathbb{V}\{\widehat{\gamma}_s\} &= \frac{1}{N_1}\left(\gamma - \gamma^2\right) \\ &= \frac{1}{N_1}\left(\sum_{i=0}^{n} p_i \gamma_i - \gamma^2\right). \end{aligned} \tag{6.7}$$

Comparing expressions (6.6) and (6.7) and considering that $\gamma_i \leq 1,\ i = 0, \ldots, n$, because all these values are probabilities, it is clear that:

$$\sum_{i=0}^{n} p_i \gamma_i^2 \leq \sum_{i=0}^{n} p_i \gamma_i,$$

what means that the variance of the *Conditional Monte Carlo* estimator given in (6.6), is never larger than the *Standard Monte Carlo* estimator variance given in (6.7). This is of course, a general fact on *Conditional Monte Carlo* methods, but it is worth making it explicit in our context.

Figure 6.1 shows a path that goes from $\mathbf{u}$ to $\mathbf{d}$ without hitting any state in $\widetilde{C}$. However,

depending on which states are selected to form the set $\widetilde{C}$, there may not exist such a path. The efficiency of the simulation in both cases is quite different.

In the latter case, when there is no path from $\mathbf{u}$ to $\mathbf{d}$ without hitting a state in $\widetilde{C}$, the terms to accumulate in expression (6.4) are γ_1, γ_2, ..., γ_n, with probabilities p_1, p_2, ..., p_n, respectively. As the probabilities p_1, p_2, ..., p_n, are all higher than γ, the values to accumulate in (6.4) can be found with less computational effort than the effort needed to find a 1 for expression (6.2) in the standard simulation.

On the other hand, if there is a path that goes from $\mathbf{u}$ to $\mathbf{d}$ without hitting any state in $\widetilde{C}$, one of the terms to accumulate in expression (6.4) is $\gamma_0 = 1$ with probability p_0, where $0 < p_0 < \gamma$. Therefore, the computational effort to find a 1 for expression (6.4) will be higher than the necessary effort to find a 1 in the standard simulation.

Finally, if there is no path from $\mathbf{u}$ to $\mathbf{d}$ without hitting a state in $\widetilde{C}$, *Conditional Monte Carlo* simulation can achieve the same accuracy level as crude or standard simulation, with a smaller number of replications.

6.3 Conditional Monte Carlo with Intermediate Estimations

The main problem in the use of *Conditional Monte Carlo*, as it was introduced so far, is the fact that the values γ_1, γ_2, ..., γ_n are unknown, and may be as hard to evaluate as the calculation of γ itself. To work around this problem, these values will be now replaced by estimators.

In this section it will be shown that after replacing γ_1, γ_2, ..., γ_n by estimators, the method is still unbiased. This replacement is the core of the proposal introduced in this chapter and the basis of the so–called *Conditional Monte Carlo with Intermediate Estimations* (*CMIE*) method. At the end of this section, the variance of the *CMIE* estimator will be determined.

To address the following calculation, it is better to express $\widehat{\gamma}$ in terms of the random vector $\bar{I} = (I_0, I_1, \ldots, I_{n+1})$, whose components are dependent binary random variables such that one and only one has value 1, distributed as follows:

$$\bar{I} = \begin{cases} (1,0,0,\ldots,0,0) & \text{w.p.} \quad p_{\mathbf{d}}, \\ (0,1,0,\ldots,0,0) & \text{w.p.} \quad p_1, \\ (0,0,1,\ldots,0,0) & \text{w.p.} \quad p_2, \\ \qquad\vdots & \\ (0,0,0,\ldots,1,0) & \text{w.p.} \quad p_n, \\ (0,0,0,\ldots,0,1) & \text{w.p.} \quad p_{\mathbf{u}}. \end{cases} \tag{6.8}$$

Then, the standard estimator is:

$$\begin{aligned} \widehat{\gamma}_s &= \frac{1}{N_1}\sum_{j=1}^{N_1} I_0^{(j)} \times \gamma_0 + I_1^{(j)} \times \gamma_1 + I_2^{(j)} \times \gamma_2 + \ldots + I_n^{(j)} \times \gamma_n + I_{n+1}^{(j)} \times 0 \\ &= \frac{1}{N_1}\sum_{j=1}^{N_1}\sum_{k=0}^{n} I_k^{(j)} \times \gamma_k \end{aligned} \tag{6.9}$$

where $\gamma_0 = 1$.

In (6.9), the samples $I_0^{(j)}$, $I_1^{(j)}$, ..., $I_n^{(j)}$ are obtained by the simulation, whereas the values $\gamma_1, \gamma_2, \ldots, \gamma_n$ must be calculated. However, if such calculation is too hard, or simply impossible, these values can be replaced by standard estimators. In order to do this, every time the simulation reaches a state $i \in \widetilde{C}$, N_2 independent replications must be started at i and simulated until they either reach $\mathbf{d}$ (and then 1 is accumulated) or $\mathbf{u}$ (and then 0 is accumulated). Once these N_2 replications started at i are completed, a standard estimator $\widehat{\gamma}_i$ can be evaluated and used in place of γ_i. To compute these estimations, define the set of Bernoulli random variables $\{J_i\}_{i=1}^n$, with the following probability distribution:

$$J_i = \begin{cases} 1 & \text{w.p.} \quad \gamma_i, \\ 0 & \text{w.p.} \quad 1-\gamma_i, \end{cases} \qquad i = 1, 2, \ldots, n. \tag{6.10}$$

The samples of J_i are obtained from the actual simulation of the Markov chain, which is the same as sampling them from distribution (6.10) ($J_0 = 1$ w.p. 1). Then, if γ_k is replaced by the estimator $\widehat{\gamma}_k$ in (6.9), the standard estimator is transformed into the *CMIE* estimator $\widehat{\gamma}_{cie}$:

$$\begin{aligned} \widehat{\gamma}_{cie} &= \frac{1}{N_1} \sum_{j=1}^{N_1} \left(\sum_{k=0}^{n} I_k^{(j)} \times \frac{1}{N_2} \sum_{i=1}^{N_2} J_k^{(j,i)} \right) \\ &= \frac{1}{N_1} \sum_{j=1}^{N_1} \left(\frac{1}{N_2} \sum_{k=0}^{n} I_k^{(j)} \sum_{i=1}^{N_2} J_k^{(j,i)} \right) \\ &= \frac{1}{N_1 N_2} \sum_{j=1}^{N_1} \sum_{k=0}^{n} \sum_{i=1}^{N_2} I_k^{(j)} J_k^{(j,i)}. \end{aligned}$$

It is simple to show that $\widehat{\gamma}_{cie}$ is unbiased:

$$\begin{aligned} \mathbb{E}\{\widehat{\gamma}_{cie}\} &= \frac{1}{N_1 N_2} \mathbb{E}\left\{ \sum_{j=1}^{N_1} \sum_{k=0}^{n} \sum_{i=1}^{N_2} I_k^{(j)} J_k^{(j,i)} \right\} \\ &= \frac{1}{N_1 N_2} \sum_{j=1}^{N_1} \sum_{k=0}^{n} \sum_{i=1}^{N_2} \mathbb{E}\left\{ I_k^{(j)} J_k^{(j,i)} \right\} \\ &= \frac{1}{N_1 N_2} \sum_{j=1}^{N_1} \sum_{k=0}^{n} \sum_{i=1}^{N_2} \mathbb{E}\left\{ I_k^{(j)} \right\} \mathbb{E}\left\{ J_k^{(j,i)} \right\} \\ &= \frac{1}{N_1 N_2} \sum_{j=1}^{N_1} \sum_{k=0}^{n} \sum_{i=1}^{N_2} p_k \gamma_k \\ &= \sum_{k=0}^{n} p_k \gamma_k \; = \; \gamma. \end{aligned}$$

In order to determine the variance of $\widehat{\gamma}_{cie}$, let $\bar{I}^{(x)}$ be any possible replication of $\bar{I}$, what means

that $I_0^{(x)}$, $I_1^{(x)}$, ..., $I_n^{(x)}$, are the components of this replication. Using the variance decomposition formula, the variance of the estimator can be written as:

$$\mathbb{V}\{\widehat{\gamma}_{cie}\} = \underbrace{\mathbb{V}\{\mathbb{E}\{\widehat{\gamma}_{cie} \mid \bar{I}^{(x)}\}\}}_{A} + \underbrace{\mathbb{E}\{\mathbb{V}\{\widehat{\gamma}_{cie} \mid \bar{I}^{(x)}\}\}}_{B}.$$

Terms A and B are now analysed separately.

$$\begin{aligned}
A &= \mathbb{V}\left\{\mathbb{E}\left\{\frac{1}{N_1 N_2}\sum_{k=0}^{n}\sum_{j=1}^{N_1}\sum_{i=1}^{N_2} I_k^{(x)} J_k^{(j,i)}\right\}\right\} \\
&= \mathbb{V}\left\{\frac{1}{N_1 N_2}\sum_{k=0}^{n}\sum_{j=1}^{N_1}\sum_{i=1}^{N_2} I_k^{(x)}\ \mathbb{E}\{J_k^{(j,i)}\}\right\} \\
&= \mathbb{V}\left\{\frac{1}{N_1 N_2}\sum_{k=0}^{n}\sum_{j=1}^{N_1}\sum_{i=1}^{N_2} I_k^{(x)}\gamma_k\right\} \\
&= \mathbb{V}\left\{\frac{1}{N_1}\sum_{j=1}^{N_1}\sum_{k=0}^{n} I_k^{(x)}\gamma_k\right\}.
\end{aligned}$$

For any given x only one of the values $I_0^{(x)}$, $I_1^{(x)}$, ..., $I_n^{(x)}$ equals 1 and the rest equal 0. As $I_i^{(x)} = 1$ w.p. p_i (see (6.8)), the term $\sum_{k=0}^{n} I_k^{(x)}\gamma_k$ equals γ_i w.p. p_i. To evaluate the variance, note that the randomness in the expression between braces is only due to $I_k^{(x)}$. Thus, $\sum_{k=0}^{n} I_k^{(x)}\gamma_k$ is a random variable on the state space $\{\gamma_0, \gamma_1, \ldots, \gamma_n\}$, with probabilities $\{p_0, p_1, \ldots, p_n\}$. Then:

$$\begin{aligned}
A &= \frac{1}{N_1}\left(\sum_{k=0}^{n} p_k\gamma_k^2 - \left(\sum_{k=0}^{n} p_k\gamma_k\right)^2\right) \\
&= \frac{1}{N_1}\left(\sum_{k=0}^{n} p_k\gamma_k^2 - \gamma^2\right).
\end{aligned}$$

For the remaining term,

$$\begin{aligned}
B &= \mathbb{E}\left\{\mathbb{V}\left\{\frac{1}{N_1 N_2}\sum_{k=0}^{n}\sum_{j=1}^{N_1}\sum_{i=1}^{N_2} I_k^{(x)} J_k^{(j,i)}\right\}\right\} \\
&= \mathbb{E}\left\{\frac{1}{N_1^2 N_2^2}\sum_{k=0}^{n}\sum_{j=1}^{N_1}\sum_{i=1}^{N_2}\left(I_k^{(x)}\right)^2\ \mathbb{V}\left\{J_k^{(j,i)}\right\}\right\} \\
&= \mathbb{E}\left\{\frac{1}{N_1^2 N_2^2}\sum_{k=0}^{n}\sum_{i=1}^{N_2} N_1 \times I_k^{(x)}\ \gamma_k(1-\gamma_k)\right\}
\end{aligned}$$

$$
\begin{aligned}
&= \mathbb{E}\left\{\frac{1}{N_1 N_2^2}\sum_{k=0}^{n} I_k^{(x)} \sum_{i=1}^{N_2} \gamma_k(1-\gamma_k)\right\} \\
&= \mathbb{E}\left\{\frac{1}{N_1 N_2^2}\sum_{k=0}^{n} I_k^{(x)} N_2 \times \gamma_k(1-\gamma_k)\right\} \\
&= \frac{1}{N_1 N_2}\sum_{k=0}^{n} \mathbb{E}\left\{I_k^{(x)}\right\} \gamma_k(1-\gamma_k) \\
&= \frac{1}{N_1 N_2}\sum_{k=0}^{n} p_k\, \gamma_k(1-\gamma_k) \\
&= \frac{1}{N_1 N_2}\left(\sum_{k=0}^{n} p_k\gamma_k - \sum_{k=0}^{n} p_k\gamma_k^2\right) \\
&= \frac{1}{N_1 N_2}\left(\gamma - \sum_{k=0}^{n} p_k\gamma_k^2\right).
\end{aligned}
$$

Then:

$$
\mathbb{V}\{\widehat{\gamma}_{cie}\} = A + B \;=\; \frac{1}{N_1}\left(\sum_{k=0}^{n} p_k\gamma_k^2 - \gamma^2\right) + \frac{1}{N_1 N_2}\left(\gamma - \sum_{k=0}^{n} p_k\gamma_k^2\right). \tag{6.11}
$$

The term A is the value of the variance of the *Conditional Monte Carlo* estimator when the values $\gamma_1, \gamma_2, \ldots, \gamma_n$ are known exactly (see (6.6)). The term B is the increment of variance due to the fact that the values $\gamma_1, \gamma_2, \ldots, \gamma_n$ are replaced by estimators.

6.4 Multiple Sets of Intermediate States

The key to the application of *Conditional Monte Carlo* to Markov chains (as described in Section 6.2) —call it *pure Conditional Monte Carlo*— is the knowledge of the probabilities $\gamma_1, \gamma_2, \ldots, \gamma_n$. The lack of these values makes it necessary to use estimators instead (as described in Section 6.3). This technique is the heart of the *CMIE* method proposed in this thesis. As shown, the estimators $\widehat{\gamma}_1, \widehat{\gamma}_2, \ldots, \widehat{\gamma}_n$ can be obtained by standard simulation started every time one of the intermediate states $1, 2, \ldots, n$ is reached. But these values can be estimated more accurately, applying the same *Conditional Monte Carlo* method recursively, in the following way.

Suppose that two sets of intermediate states, $\widetilde{C}_1$ and $\widetilde{C}_2$, are defined, instead of one, as shown in Figure 6.2. Assume that $\widetilde{C}_1 \cap \widetilde{C}_1 = \emptyset$ and $\mathbf{u}, \mathbf{d} \notin \widetilde{C}_1, \widetilde{C}_2$. Suppose that the process starts just as in the case in which $\widetilde{C}_1$ is the only set of intermediate states. Then, once a state $i \in \widetilde{C}_1$ is reached, N_2 replications are started at state i, and they are simulated until they either hit a state in $\widetilde{C}_2$, go back to $\mathbf{u}$, or gets absorbed at $\mathbf{d}$. This can be considered the second recursive level of the simulation. It is intended to obtain the values $\gamma'_1, \gamma'_2, \ldots, \gamma'_{n_1}$, which indicate the probability that each of these N_2 replications get absorbed at $\mathbf{d}$. Proceeding this way, these probabilities can be estimated by this recursive level of *Conditional Monte Carlo* simulation that makes use of $\widetilde{C}_2$ as the set of intermediate states. These estimations are, in fact, more accurately than the ones

obtained by means of standard simulation in the case of only one set of intermediate states. It is simple to extend this mechanism to more recursive levels (with more sets of intermediate states).

In Appendix A the variance analysis is extended to the case of two sets of intermediate states, $\widetilde{C}_1$ and $\widetilde{C}_2$, as decribed in the previous paragraphs. The probabilities involved are shown in Figure 6.2. The result obtained is the following:

$$\begin{aligned}\mathbb{V}\{\widehat{\gamma}_{cie}\} =& \frac{1}{N_1}\left(\sum_{l=0}^{n_1} p_l\,{\gamma'_l}^2 - \gamma^2\right) + \\ & \frac{1}{N_1}\sum_{l=0}^{n_1} p_l\left(1/N_2\left(\sum_{k=0}^{n_2} p_{lk}\gamma_k^2 - {\gamma'_l}^2\right) + \frac{1}{N_2 N}\left(\gamma'_l - \sum_{k=0}^{n_2} p_{lk}\,\gamma_k^2\right)\right).\end{aligned}$$

Given this expression, it is possible to prove that the variance obtained in a model with two sets of intermediate states, $\widetilde{C}_1$ and $\widetilde{C}_2$, is lower than or equal to the variance obtained in a model with the single intermediate set of states $\widetilde{C}_1$. This statement is proven in Appendix B.

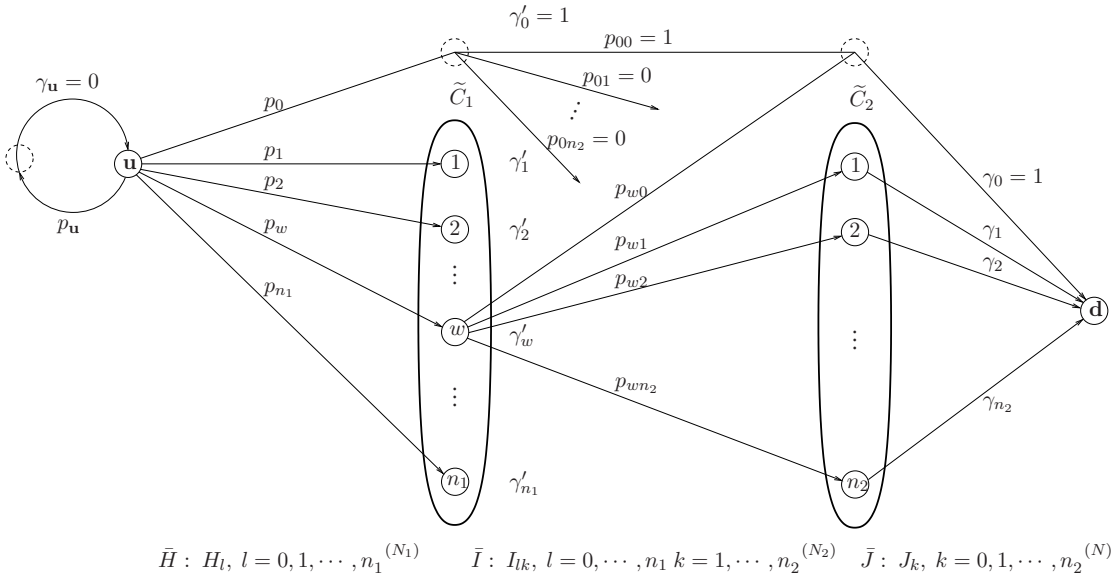


Figure 6.2: The case of two sets of Intermediate States, $\widetilde{C}_1$ and $\widetilde{C}_2$

6.5 Comparative Analysis of Variances

The variance of the *CMIE* estimator, for the case of only one set of intermediate states, was derived in Section 6.3. In this section, this variance is compared to the variance of other estimators.

The variance of the *Standard Monte Carlo* estimator, as shown in (6.7), is:

$$\mathbb{V}\{\widehat{\gamma}_s\} = \frac{1}{N_1}\left(\gamma - \gamma^2\right).$$

The variance of the *pure Conditional Monte Carlo* estimator derived in (6.6), in which the exact values of the probabilities $\gamma_i,\ i = 0, \ldots, n$, are used, is:

$$\mathbb{V}\{\widehat{\gamma}_c\} = \frac{1}{N_1}\left(\sum_{i=0}^{n} p_i\gamma_i^2 - \gamma^2\right),$$

and the variance of the proposed estimator just derived in (6.11), in which the probabilities $\gamma_i,\ i = 0, \ldots, n$, are estimated by standard simulation, is:

$$\mathbb{V}\{\widehat{\gamma}_{cie}\} = \frac{1}{N_1}\left(\sum_{k=0}^{n} p_k\gamma_k^2 - \gamma^2\right) + \frac{1}{N_1 N_2}\left(\gamma - \sum_{k=0}^{n} p_k\gamma_k^2\right).$$

As $N_2 \to \infty$, then $\mathbb{V}\{\widehat{\gamma}_{cie}\} \to \mathbb{V}\{\widehat{\gamma}_c\}$. Clearly, if the number of replications used in the estimation of the probabilities $\gamma_i,\ i = 0, \ldots, n$ is infinite, the estimators converge to the corresponding exact values, and the method becomes the *pure Conditional Monte Carlo*.

At the end of Section 6.2 it has been shown that $\mathbb{V}\{\widehat{\gamma}_c\} \leq \mathbb{V}\{\widehat{\gamma}_s\}$, meaning that the accuracy of *pure Conditional Monte Carlo* is never less than the accuracy of *Standard Monte Carlo*. It is clear that $\mathbb{V}\{\widehat{\gamma}_c\} \leq \mathbb{V}\{\widehat{\gamma}_{cie}\}$. It is pending to prove that $\mathbb{V}\{\widehat{\gamma}_{cie}\} \leq \mathbb{V}\{\widehat{\gamma}_s\}$, meaning that the proposed estimator is never less accurate than *Standard Monte Carlo* estimator.

$$\begin{aligned}
\mathbb{V}\{\widehat{\gamma}_{cie}\} &= \frac{\sum_{k=0}^{n} p_k\gamma_k^2 - \gamma^2}{N_1} + \frac{\gamma - \sum_{k=0}^{n} p_k\gamma_k^2}{N_1 N_2} \\
&\leq \frac{\sum_{k=0}^{n} p_k\gamma_k^2 - \gamma^2}{N_1} + \frac{\gamma - \sum_{k=0}^{n} p_k\gamma_k^2}{N_1} \\
&\leq \frac{\gamma - \gamma^2}{N_1} \\
&\leq \mathbb{V}\{\widehat{\gamma}_s\}.
\end{aligned}$$

The inequality holds, no matter the values of N_1 and N_2. This means that the proposed estimator, $\widehat{\gamma}_{cie}$, is never less accurate than *Standard Monte Carlo* estimator, $\widehat{\gamma}_s$, even for a low number of replications N_1 and N_2.

Finally, the three variances involved are related as follows:

$$\mathbb{V}\{\widehat{\gamma}_c\} \leq \mathbb{V}\{\widehat{\gamma}_{cie}\} \leq \mathbb{V}\{\widehat{\gamma}_s\},$$

which means that *CMIE* is always more accurate than crude or *Standard Monte Carlo*, but never as accurate as *pure Conditional Monte Carlo*, in which the exact values $\gamma_i,\ i = 0, \ldots, n$, are used.

6.6 Intermediate States Analysis

The variance reduction capacity of *CMIE* depends on the choice of the set of intermediate states. In this section two properties of the sets of intermediate states are analyzed. The first one concerns

the number of states of such sets. It will be shown that after adding a new state to an existing set, the variance of the estimator never increases and, therefore, a variance reduction may be expected. The second property is related to the comparison between two particular sets of intermediate states, namely cuts, each of them considered individually, i.e. one at a time. It will be shown that the one that yields the highest variance reduction is the one composed of states that are somehow closer to the initial state, $\mathbf{u}$. These two properties are consistent and behave as a guideline to make the choice of the sets of intermediate sates. For simplicity these properties will be analysed in the case of *pure Conditional Monte Carlo*.

Let $\widehat{\gamma}_{c_n}$ be the *pure Conditional Monte Carlo* estimator obtained when the set of intermediate states is composed of n states, and let $\widehat{\gamma}_{c_{n+1}}$ be the same estimator when the set of intermediate states is composed of $n+1$ sates, obtained by the addition of one state to the first considered set. To address the first property, define $V_n = N_1 \times \mathbb{V}\{\widehat{\gamma}_{c_n}\}$ and $V_{n+1} = N_1 \times \mathbb{V}\{\widehat{\gamma}_{c_{n+1}}\}$. It will be proven that $V_n - V_{n+1} \geq 0$.

Let $p_k,\ k = 0, \ldots, n$, be the probability that a replication started at state $\mathbf{u}$ reaches state k, before any other state in C, in the model with n intermediate states and let $p'_k,\ k = 0, \ldots, n, n+1$, be the same probabilities in the model with $n+1$ intermediate states. Then:

$$V_n = \sum_{k=0}^{n} p_k \gamma_k^2 - \gamma^2$$
$$\text{and } V_{n+1} = \sum_{k=0}^{n+1} p'_k \gamma_k^2 - \gamma^2.$$

Clearly $p'_k \leq p_k,\ k = 0, \ldots, n$, because as p_k is the probability that starting at $\mathbf{u}$ the process reaches k before any other state in C, after C grows by the addition of one more state, some of the paths from $\mathbf{u}$ to k before the addition may now include the new state and, therefore, such paths will not go from $\mathbf{u}$ to k, before reaching states in C, anymore.

The target, γ, can thus be written in the two following ways:

$$\sum_{k=0}^{n} p_k \gamma_k = \sum_{k=0}^{n+1} p'_k \gamma_k = \gamma,$$

and thus,

$$\sum_{k=0}^{n} (p_k - p'_k)\gamma_k = p'_{n+1}\gamma_{n+1}. \tag{6.12}$$

Considering that any replication that starts at $\mathbf{u}$, either ends at some state in C, comes back to $\mathbf{u}$ or gets absorbed at $\mathbf{d}$ (see Figure 6.1), it is clear that:

$$p_{\mathbf{u}} + \sum_{k=0}^{n} p_k = p'_{\mathbf{u}} + \sum_{k=0}^{n+1} p'_k = 1,$$

leading to:

$$(p_{\mathbf{u}} - p'_{\mathbf{u}}) + \sum_{k=0}^{n} (p_k - p'_k) = p'_{n+1}. \tag{6.13}$$

Calling $r_k = (p_k - p'_k) \geq 0,\ k = \mathbf{u}, 0, 1, \ldots, n$, Equation (6.12) can be written as follows:

$$\sum_{k=0}^{n} r_k \gamma_k = p'_{n+1}\gamma_{n+1} \tag{6.14}$$

and Equation (6.13) as:

$$r_{\mathbf{u}} + \sum_{k=0}^{n} r_k = p'_{n+1}. \tag{6.15}$$

Recall that we have to prove that $V_n - V_{n+1} \geq 0$. This difference takes the following form:

$$\begin{aligned} V_n - V_{n+1} &= \sum_{k=0}^{n} p_k \gamma_k^2 - \sum_{k=0}^{n+1} p'_k \gamma_k^2 \\ &= \sum_{k=0}^{n} r_k \gamma_k^2 - p'_{n+1}\gamma_{n+1}^2. \end{aligned}$$

The only case of interest is when p'_{n+1} is strictly positive, because if $p'_{n+1} = 0$ both sets of intermediate states are the same and $V_n = V_{n+1}$. Assuming that $p'_{n+1} > 0$,

$$\frac{V_n - V_{n+1}}{p'_{n+1}} = \sum_{k=0}^{n} \frac{r_k}{p'_{n+1}} \gamma_k^2 - \gamma_{n+1}^2. \tag{6.16}$$

Think of a random variable W with the following probability distribution:

$$W = \begin{cases} 0 & \text{w.p.} \quad r_{\mathbf{u}}/p'_{n+1}, \\ 1 & \text{w.p.} \quad r_0/p'_{n+1}, \\ \gamma_1 & \text{w.p.} \quad r_1/p'_{n+1}, \\ \vdots & \\ \gamma_n & \text{w.p.} \quad r_n/p'_{n+1}. \end{cases}$$

This is in fact a probability distribution because the terms $r_{\mathbf{u}}$, r_0, ..., r_n and p'_{n+1} are all positive and $r_{\mathbf{u}}/p'_{n+1} + \sum_{k=0}^{n} r_k/p'_{n+1} = 1$, due to (6.15). The expectation of this random variable (see (6.14)) is:

$$\begin{aligned} \mathbb{E}\{W\} &= \frac{r_{\mathbf{u}}}{p'_{n+1}} \times 0 + \sum_{k=0}^{n} \frac{r_k}{p'_{n+1}} \gamma_k \\ &= \gamma_{n+1}, \end{aligned}$$

and, therefore, (6.16) gives its variance. This completes the proof, because a variance is always non negative:

$$\frac{V_n - V_{n+1}}{p'_{n+1}} = \sum_{k=0}^{n} \frac{r_k}{p'_{n+1}} \gamma_k^2 - \gamma_{n+1}^2 \geq 0,$$

and thus,

$$V_n - V_{n+1} \geq 0.$$

Next it will be considered the case in which the intermediate states compose a cut in the Markov chain. Then, the variances of two estimators obtained using two different cuts — performing separately— are compared.

For any two states i and j in the state space S of a Markov chain, $\widetilde{C}$ is called a cut *between* i and j if there are three disjoint subsets S_i, $\widetilde{C}$ and S_j such that: (i) $S = S_i \cup \widetilde{C} \cup S_j$, (ii) $i \in S_i$ and $j \in S_j$ and (iii) every path from a state in S_i to a state in S_j contains one state in $\widetilde{C}$. A cut $\widetilde{C}$ without any mention to the states i and j is an implicit reference to the case in which these states are $\mathbf{u}$ and $\mathbf{d}$.

It is possible to define a cut *between* one state i and another cut $\widetilde{C}_j$ and also a cut *between* two cuts, $\widetilde{C}_i$ and $\widetilde{C}_j$. In the first case the definition must apply for all the pairs formed by i and the states in $\widetilde{C}_j$. In the second case the definition must apply for all the pairs formed by one state in $\widetilde{C}_i$ and one in $\widetilde{C}_j$.

Now, the variances of two estimators, $\widehat{\gamma}_{c_1}$ and $\widehat{\gamma}_{c_2}$, are compared. $\widehat{\gamma}_{c_1}$ is the *pure Conditional Monte Carlo* estimator obtained by a simulation for which the only cut selected is $\widetilde{C}_1$, while $\widehat{\gamma}_{c_2}$ is the same, when the only cut selected is $\widetilde{C}_2$. $\widetilde{C}_1$ is a cut between $\mathbf{u}$ and $\widetilde{C}_2$, whereas $\widetilde{C}_2$ is a cut between $\widetilde{C}_1$ and $\mathbf{d}$.

Let $V_1 = N_1 \times \mathbb{V}\{\widehat{\gamma}_{c_1}\}$ and $V_2 = N_1 \times \mathbb{V}\{\widehat{\gamma}_{c_2}\}$. In order to compare the variances, V_1 and V_2 must be written in terms of the same set of probabilities. The comparison will show that V_2 can never be less than V_1. The probabilities involved are shown in Figure 6.3.

Writing V_1 as follows,

$$\begin{aligned} V_1 &= p_1 {\gamma'_1}^2 + \cdots + p_{n_1} {\gamma'_{n_1}}^2 - \gamma^2 \\ &= p_1 \underbrace{(p_{11}\gamma_1 + p_{12}\gamma_2 + \cdots + p_{1n_2}\gamma_{n_2})^2}_{x_1} + \cdots + p_{n_1} \underbrace{(p_{n_1 1}\gamma_1 + p_{n_1 2}\gamma_2 + \cdots + p_{n_1 n_2}\gamma_{n_2})^2}_{x_{n_1}} - \gamma^2, \end{aligned}$$

and V_2 as,

$$\begin{aligned} V_2 &= p'_1\gamma_1^2 + \cdots + p'_{n_2}\gamma_{n_2}^2 - \gamma^2 \\ &= (p_1 p_{11} + p_2 p_{21} + \cdots + p_{n_1} p_{n_1 1})\gamma_1^2 + \cdots + (p_1 p_{1n_2} + p_2 p_{2n_2} + \cdots + p_{n_1} p_{n_1 n_2})\gamma_{n_2}^2 - \gamma^2 \\ &= p_1 \underbrace{(p_{11}\gamma_1^2 + p_{12}\gamma_2^2 + \cdots + p_{1n_2}\gamma_{n_2}^2)}_{y_1} + \cdots + p_{n_1} \underbrace{(p_{n_1 1}\gamma_1^2 + p_{n_1 2}\gamma_2^2 + \cdots + p_{n_1 n_2}\gamma_{n_2}^2)}_{y_{n_1}} - \gamma^2, \end{aligned}$$

it results:

$$V_2 - V_1 = p_1(y_1 - x_1) + \cdots + p_{n_1}(y_{n_1} - x_{n_1}). \tag{6.17}$$

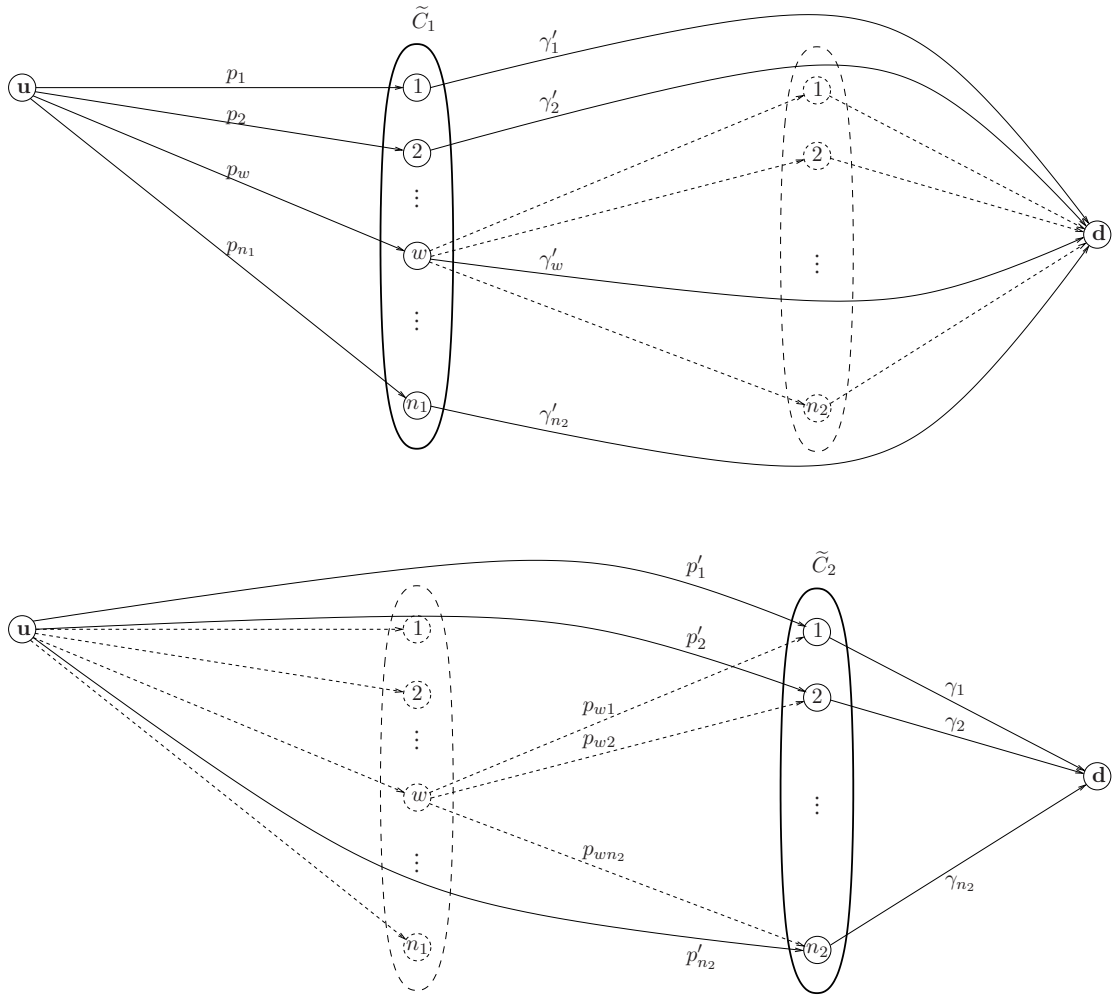


Figure 6.3: Probabilities for variance comparison using one of two different sets, $\widetilde{C}_1$ or $\widetilde{C}_2$

It will be shown that $V_2 - V_1 \geq 0$.

Let Γ_1 (defined only with auxiliary purpose) be the following random variable:

$$\Gamma_1 = \begin{cases} \gamma_1 & \text{w.p.} \quad p_{11}, \\ \gamma_2 & \text{w.p.} \quad p_{12}, \\ \vdots & \\ \gamma_{n_2} & \text{w.p.} \quad p_{1_{n_2}}. \end{cases} \tag{6.18}$$

The expected value of Γ_1 is:

$$\mathbb{E}\{\Gamma_1\} = p_{11}\gamma_1 + p_{12}\gamma_2 + \cdots + p_{1n_2}\gamma_{n_2} = \gamma'_1,$$

and its variance:

$$\mathbb{V}\{\Gamma_1\} = (p_{11}\gamma_1^2 + p_{12}\gamma_2^2 + \cdots + p_{1n_2}\gamma_{n_2}^2) - (p_{11}\gamma_1 + p_{12}\gamma_2 + \cdots + p_{1n_2}\gamma_{n_2})^2.$$

This variance —that, as any variance, is not negative— is numerically equal to the term $y_1 - x_1$. Similarly $y_2 - x_2$, $\cdots$, $y_{n_1} - x_{n_1}$ are, respectively, the variances of the variables Γ_2, $\cdots$, Γ_{n_1}, whose definitions are similar to (6.18).

Going back to (6.17), it follows that V_2 can never be less than V_1, because:

$$V_2 - V_1 = p_1 \underbrace{(y_1 - x_1)}_{\geq 0} + \cdots + p_{n_1} \underbrace{(y_{n_1} - x_{n_1})}_{\geq 0} \geq 0, \tag{6.19}$$

or what is the same, that the variance of the estimator $\widehat{\gamma}_{c_2}$ can never be less than the variance of the estimator $\widehat{\gamma}_{c_1}$, no matter the number of states in each of the cuts. If it is possible to find a cut $\widetilde{C}_0$ between $\mathbf{u}$ and $\widetilde{C}_1$, the variance of $\widehat{\gamma}_{c_0}$ will be lower than or equal to the variance of $\widehat{\gamma}_{c_1}$. The cut $\widetilde{C}_a$ formed by the states adjacent to $\mathbf{u}$ leaves no room for another cut (between $\widetilde{C}_a$ and $\mathbf{u}$) and, therefore, there is no cut that produces an estimator with lower variance than $\widehat{\gamma}_{c_a}$.

A similar conclusion can be derived from the fact proven in the first part of this section. Actually, if the addition of one state to an existing set of intermediate states yields a variance that is lower than, or at worst equal to, the variance before the addition, the set that yields the least variance is the one composed of all the states: $C = S$. However, from the implementation point of view, the set $C = S$ is equivalent to the set formed by the adjacent states to $\mathbf{u}$, $\widetilde{C}_a$ (because, if $C = S$, for any replication started at the initial state $\mathbf{u}$, the only reachable states will be the adjacent ones).

In the case of two or more sets of intermediate states, the choice of the second, and the consecutive ones, must be somehow similar to the choice of the first one with respect to the initial state $\mathbf{u}$. Whenever possible, the second cut (between the existing one and state $\mathbf{d}$) must be formed by the adjacent states to the existing cut. However, this is not straightforward and must be analyzed for every particular model.

Figure 6.4 shows a continuous time Markov chain used by Juneja and Shahabuddin in [Juneja 2001]. The system has 2 components of class A and 4 components of class B. The components can only be *operational* or *failed*. The state is the pair (N_A,N_B), where N_i indicates the number of *failed* components in class i. Failure rates of classes A and B are, respectively, $\varepsilon/2$ and ε. The system fails if all components of all classes fail. Group repair (all *failed* components of a class are repaired simultaneously) begins if two components of the same class fail. Group repair rates for both classes are equal to 1. There is one repair-person in the system, and class A gets preemptive priority over class B.

In the rest of this section *CMIE* is tested on the model just introduced, and in the next section it is compared, by means of different models, with many other methods used in the estimation of γ, namely: FB, SFB and SFBP, used in [Cancela 2002] and BFB, SBLR, ZVA(v_0), ZVA(v_1), ZVA(v_2), and ZVA(v_3), used in [L'Ecuyer 2011b] (all of them derived from *Importance Sampling*). To make the comparisons, different experiments, are performed. These experiments consist of running simulations over selected models and obtaining an estimation of γ together with measures of precision and efficiency. Actually the most interesting results of the experiments are

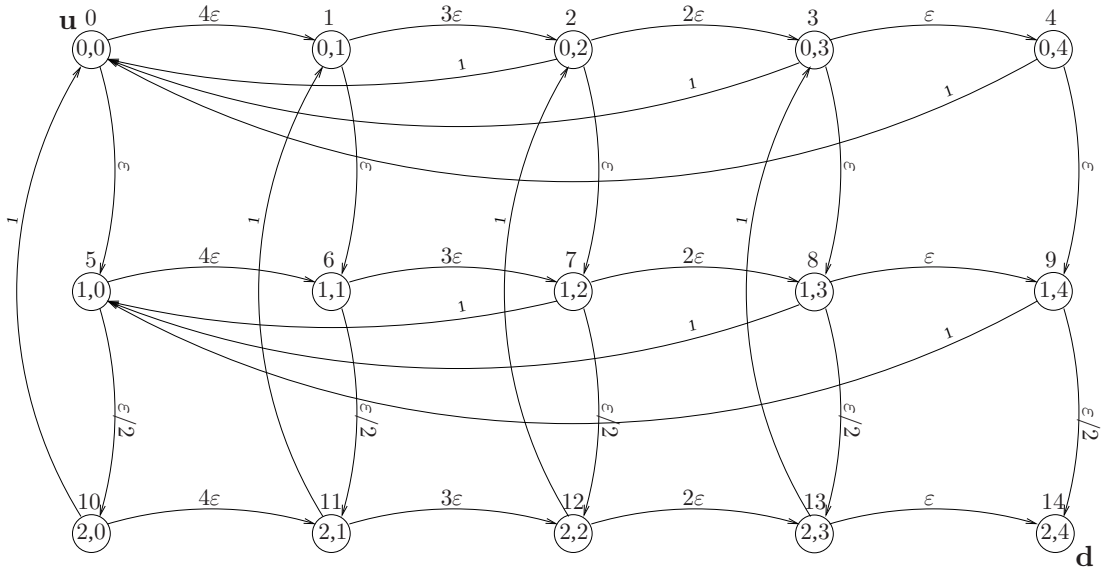


Figure 6.4: Continuous time Markov chain used in the experimental variance analysis

not the estimations themselves, but the precision and efficiency of the methods, instead. This way, the results of the *CMIE* procedure are compared to those published in the referred papers.

If $\widehat{\gamma}$ is the estimator obtained by the method under analysis in time t (either *CMIE* or any other one), with expectation $\mathbb{E}\{\widehat{\gamma}\} = \gamma$ and variance $\mathbb{V}\{\widehat{\gamma}\}$, and $\widehat{\gamma}_c$ is the corresponding estimator obtained by *Crude Monte Carlo* in time t_c, the following parameter is used in the results shown hereafter:

$\mathbb{V}\{\widehat{\gamma}_c\}/\mathbb{V}\{\widehat{\gamma}\}$: the *Variance Ratio*, that shows the precision improvement of the method under analysis over *Crude Monte Carlo* for runs that share a common parameter such as the sample size or the number of replications.

The *CMIE* method was programmed in the `C` language, using the `gcc` compiler. Simulations proceeded as follows: N_1 replications were started at state $\mathbf{u}$, and they were multiplied as they reached an intermediate state of the first set. The created replications kept on being multiplied recursively as they reached the consecutive sets of intermediate states, until all the trajectories finally end, either at state $\mathbf{d}$ or at state $\mathbf{u}$. Every one of the N_1 replications reported an estimate $\gamma^{(j)},\ j = 1, \ldots, N_1$. Given these values, the estimator $\widehat{\gamma}_{cie}$ and an unbiased estimator of its variance were calculated as follows:

$$\widehat{\gamma}_{cie} = \frac{1}{N_1}\sum_{j=1}^{N_1}\gamma^{(j)}, \qquad \widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\} = \frac{1}{N_1 - 1}\left[\frac{1}{N_1}\left(\sum_{j=1}^{N_1}{\gamma^{(j)}}^2\right) - \widehat{\gamma}_{cie}^2\right]. \tag{6.20}$$

In the case of the crude or standard estimation, N_1 replications were started at state $\mathbf{u}$ and they were simulated until they finally end, either at state $\mathbf{d}$ or at state $\mathbf{u}$. Every one of the N_1

replications reported an estimate $\gamma^{(j)} \in \{0,1\},\ j = 1,\ldots,N_1$. The estimator $\widehat{\gamma}_s$ and the estimator of its variance were also calculated by means of (6.20).

At this point in the introduction and development of CMIE it might be perceived that, somehow, it resembles *Splitting*. Actually, there is a formal equivalence between both methods that will be approached in Section 6.8. For now suffice it to say that, if the sets of intermediate states are cuts in the graph of the Markov chain, both methods are in fact equivalent. However the way CMIE is defined is such that *Splitting* is a particular case, mostly because of two reasons, (i) in CMIE it is possible to go straight from state $\mathbf{u}$ to state $\mathbf{d}$ avoiding the intermediate states (see Figure 6.1) and (ii) because, as it will be seen in Section 6.7, in CMIE it is possible to condition to different events, other than hitting intermediate states, what can make very difficult, o even impossible, to find an importance function to translate these events into thresholds in the state space of the Markov chain.

Tables 6.1, 6.2 and 6.3 show the estimator $\widehat{\gamma}_{cie}$, and an estimator of its variance, $\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\}$, obtained by *CMIE* simulations over the model of Figure 6.4, for different sets of intermediate states. Table 6.3 also shows the ratio $\mathbb{V}\{\widehat{\gamma}_c\}/\mathbb{V}\{\widehat{\gamma}\}$ when *CMIE* and standard simulation run the same execution time. The sets, namely $\widetilde{C}_1$, $\widetilde{C}_2$ or $\widetilde{C}_3$, are all cuts and they are referred to, making use of the numbers placed above each state in Figure 6.4.

Table 6.1: Results of the experiments for the model in Figure 6.4, $\varepsilon = 0.01$

$\widetilde{C}_1$	$\widehat{\gamma}_{cie}$	$\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\}$
1–5	6.18E−06	6.28E−14
2–6–10	6.23E−06	6.88E−14
3–7–11	6.36E−06	6.90E−14
4–8–12	6.34E−06	1.53E−13
9–13	6.93E−06	6.56E−12

The results in Table 6.1 show that the cut that attains the lowest variance is the one formed by the adjacent states to $\mathbf{u}$. The variances of the estimators whose associated cut is close to state $\mathbf{u}$ are also low and quite similar. But, when the cuts are "far" from state $\mathbf{u}$, the variance greatly increases. In these experiments the number of replications started at state $\mathbf{u}$ was 10,000 and the number of replications launched from the intermediate states was also 10,000.

Table 6.2: Results of the experiments for the model in Figure 6.4, $\varepsilon = 0.01$

$\widetilde{C}_1$	$\widetilde{C}_2$	$\widetilde{C}_3$	$\widehat{\gamma}_{cie}$	$\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\}$
1–5	—	—	4.00E−06	4.00E−12
1–5	2–6–10	—	6.19E−06	6.31E−14
1–5	2–6–10	3–7–11	6.13E−06	2.27E−15

The experiment whose results are in Table 6.2 show that as the number of cuts increases, the variance of the estimator $\widehat{\gamma}_{cie}$ decreases. In these experiments the number of replications started at state $\mathbf{u}$ where 10,000 and the number of replications launched from the intermediate states was 100 for all cases.

Table 6.3: Results of the experiments for the model in Figure 6.4, $\varepsilon = 0.001$

$\widetilde{C}_1$	$\widetilde{C}_2$	$\widetilde{C}_3$	$\widehat{\gamma}_{cie}$	$\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\}$	$\widehat{\mathbb{V}}\{\widehat{\gamma}_s\}/\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\}$
4–8–12	—	—	6.53E−09	4.60E−20	87
3–7–11	9–13	—	6.38E−09	7.75E−21	516
3–7–11	4–8–12	9–13	6.44E−09	1.28E−21	3125

The experiments in Table 6.3 are included to briefly show the variance reduction capacity of the *CMIE* method. These experiments were done in the following way. Four simulations were executed, one using the crude or standard method, and three more using *CMIE* with respectively one, two and three cuts as shown in Table 6.3 (chosen after some pilot runs to select sets performing efficiently). The number of replications launched from intermediate states was 1,000 for all cases; the number of replications starting at state $\mathbf{u}$ was adjusted so that the total execution time of each of the four simulations was $t = 500$ sec. This time was fixed in advance and equal for all methods in order to have a fair comparison of the accuracy that was obtained by the different experiments. The fourth column shows estimations of γ that are quite similar, which is reasonable since they are all unbiased estimators. The fifth column shows how the variance decreases at the addition of cuts (while the simulation time is fixed), and the sixth column shows the ratios between the standard estimator (which is the same for the three experiments) and the variances of the corresponding *CMIE* estimators, illustrating the *precision gain* of *CMIE* over the crude or standard method.

6.7 Application to Large Multicomponent Systems

Sometimes the state space S of the Markov chain is extremely large and, therefore, the choice of intermediate states is hard to be done explicitly. Typical examples of these models are those of multicomponent systems. Think, for instance, of a system formed by components of five classes, with five components per class, each of which can be either *operational* or *failed*. Regardless of the rules and criteria that define the *up* and *down* states of the whole system, the model is a Markov chain with a potential state space of 2^{25} states. The idea of *CMIE* fits better to these models if it is adapted in the following way.

As seen so far, the replications start at some initial state $\mathbf{u}$ and they stop as soon as they hit one of the so–called intermediate states (or the one of the states $\mathbf{u}$ or $\mathbf{d}$). When the replications stop at some state, other than $\mathbf{u}$ or $\mathbf{d}$, the probability of reaching state $\mathbf{d}$ before state $\mathbf{u}$ is computed (or estimated) and accumulated in order to make the *Monte Carlo* estimation. Formally speaking, the computed values are samples of the probability of interest, γ, conditioned to the fact of hitting that intermediate state. It is possible, however, to generalize the type of event on which the target event can be conditioned to, and to gather the samples conditioned on those different facts, other than hitting intermediate states.

In (6.4), Section 6.2, the indicator random variable I is conditioned on $X_C^{(j)}$, which is the value of the random variable X_C in the j^{th} replication. X_C takes values in a space formed by the intermediate states. The values of X_C can be seen as a function of the path $\pi_{\mathbf{u}}$ followed by the trajectory started at state $\mathbf{u}$. These values are the states at which these paths end, and they make a

partition in the domain of the variable $\mathbb{E}\{I \mid X_C\}$, because there will be one value of $\mathbb{E}\{I \mid X_C\}$, called γ_i (see Figure 6.1), for every path that ends at state i.

But this partition, and consequently the values of γ_i, can be obtained by the application of different functions that take the path $\pi_{\mathbf{u}}$ as their argument. In the following subsections three possible alternatives for these functions are introduced and, then, they are tested in Section 6.7.4.

6.7.1 Forward Steps

In the models used so far, there is only one (initial) state $\mathbf{u}$, in which the system is *up*, and one (target) state $\mathbf{d}$, in which the system is *down*. Suppose that the system is such that it is possible to define consecutive cuts in the graph, distributed between $\mathbf{u}$ and $\mathbf{d}$, with a separation of a single step jump between them. After the first one step jump from state $\mathbf{u}$, the simulation reaches one of the states in the first set. Once the simulation is at one of the states in the first set, it can either move to the second set (wherever it is) or come back to state $\mathbf{u}$. If it moves forward to the second set, it should reach one of its states after a one step jump. If it goes back to state $\mathbf{u}$, the replication terminates.

After the simulation reaches one state in the second set, it can either move unto the third one (wherever it is) or backwards to state $\mathbf{u}$. If it moves unto the third set, it should reach one of its states after a one step jump. If it moves backwards to state $\mathbf{u}$, the simulation must be kept on until it goes definitely back to $\mathbf{u}$ or finally reaches one state in the third set. And the same holds for the subsequent sets of intermediate states.

The implementation of this variant of *CMIE* makes it necessary to detect whether the simulation moves forwards (to the target state) or backwards (to the initial state). As the system is composed by a number of components, each of which can only be *operational* or *failed*, and accepting that as the components fail the simulation gets closer to the target state, whereas when the components are repaired the simulation gets closer the initial state, the number of *operational* components is a reasonable measure of the distance to the target state. Thus, the increase or decrease of this distance is an indicator of whether the simulation moves forward or backward.

It has been shown that increasing the number of sets of intermediate states, a variance reduction of the *CMIE* estimator may be expected (see Appendix B). It is clear, however, that as the number of sets of intermediate states increases the computational effort —and, therefore, the simulation time—. If the state space S of the Markov chain is extremely large, to place the consecutive sets of intermediate states one step apart may cause to have a very high number of intermediate states and, consequently, a huge number of recursive calls. A solution to this problem is to consider the sets of intermediate states more apart from each other, so as to make the recursive calls only after moving $D \geq 1$ steps closer to the target state, every time. Proceeding this way, wherever the simulation starts, it must keep moving forwards and backwards until it either comes back to $\mathbf{u}$, or moves D steps towards $\mathbf{d}$. If $\mathbf{u}$ is reached, the replication terminates; if the simulation moves D steps closer to $\mathbf{d}$, a new replication (recursive call) is launched.

6.7.2 Consecutive Failures

In the *Forward Steps* implementation the simulation stops, and a number of new replications start, whenever it gets D steps closer to the target state $\mathbf{d}$. It does not matter whether, before getting

D steps closer to $\mathbf{d}$, the simulation moves in zigzag, forward and backward. A replication will stop, and a number of new replications will start, whenever any state visited by the simulation is D steps closer to the state $\mathbf{d}$, compared to the state from which the replication starts.

For a replication that starts at some state i, there are many ways (paths) to get D steps closer to the target state $\mathbf{d}$. One of them corresponds to the case in which D consecutive failures occur. If the system is composed of more than D components, there will be many different ways in which D consecutive failures may occur, all of them rarer than the case in which the D steps are completed after a zigzag of forward and backward steps. Thus, the indicator random variable I can be conditioned on such a sequence of D consecutive failures (D consecutive forward steps).

6.7.3 Measure of Rarity

Let $\pi_{i,j}$ be a path that starts at state i and ends at state j, without hitting state $\mathbf{u}$. If this path is composed of the sequence of states i, k, ..., l, j, the probability that the simulation goes through it, is:

$$\mathbb{P}\{\pi_{i,j}\} = p_{ik} \times \ldots \times p_{lj} \tag{6.21}$$

where p_{xy} is the probability of going from state x to the neighbour state y, no matter if this jump corresponds to a fail or a repair.

In order to apply *CMIE*, the indicator variable I can be conditioned on the event $\mathbb{P}\{\pi_{i,j}\} \leq B$, where B is a fixed value. At every single step jump, a new individual value p_{xy} has to be multiplied by the previous transition probabilities, making expression (6.21) grow by one term every time. The simulation must stop, and a number of new replications start, as soon as the value of the product falls below the value of the bound B.

In highly reliable systems, most of the probabilities p_{xy} are likely to be low. Then, the product of the sequence $p_{ik} \times \ldots \times p_{lj}$ could be extremely low. To avoid numerical precision problems, logarithms can be applied in the following way:

$$-\log(\mathbb{P}\{\pi_{i,j}\}) = -\log(p_{ik}) - \ldots - \log(p_{lj})$$

translating the condition $\mathbb{P}\{\pi_{i,j}\} \leq B$ into $-\log(\mathbb{P}\{\pi_{i,j}\}) \geq W$ (where $W = -\log(B)$). It is clear that, for all x, y, $-\log(p_{xy}) \geq 1$ and, consequently, $-\log(\mathbb{P}\{\pi_{i,j}\}) \geq 1$. In the models of interest, failures are usually rare whereas repairs are not. Then, for a transition from x to y, the value $-\log(p_{xy})$ will be high if the transition is a failure and it will be low if it is a repair. Finally, $-\log(\mathbb{P}\{\pi_{i,j}\})$ will be high if the product contains one or more failures or, in other words, if the probability of going through path $\mathbb{P}\{\pi_{i,j}\}$ is low. The value $-\log(\mathbb{P}\{\pi_{i,j}\})$ performs as a measure of rarity of the path that the simulations goes through, and the inequality $-\log(\mathbb{P}\{\pi_{i,j}\}) \geq W$ indicates that the path is promising in the sense of getting close to the final target state $\mathbf{d}$.

6.7.4 Experimental Comparison

The three implementations proposed to apply *CMIE* to large systems are now subject to an experimental comparison. The results in all the experiments are the estimation $\widehat{\gamma}_{cie}$ and the product

$\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\} \times t$, for different models taken from published papers. All the values reported in the tables were obtained with the formulas indicated in (6.20).

The model used in the first set of experiments was used by Cancela et. al. in [Cancela 2002] and it represents a computer that is composed of a multiprocessor, a dual disk controller, two RAID disk drives, two fans, two power supplies, and one dual interprocessor bus. When a component in a dual fails, the subsystem is reconfigured into a simplex. This tandem computer system requires all subsystems, one fan, and one power supply for it to be *operational*. The failure rates are 5ε, 2ε, 4ε, 0.1ε and 3ε for the processors, the disk controller, the disks, the fans, the power supplies and the bus respectively, with $\varepsilon = 10^{-5}$ failures/hour. There is only one repairman and the repair rates are 30 repairs/hour for all the components, except for the bus, which has repair rate equal to 15 repairs/hour. In the experiments shown in Table 6.4, the multiprocessor and the disks have two units each, and only one is needed for the system to be working. FB, SFB and SFBP are all *Importance Sampling* methods used in [Cancela 2002].

Table 6.4: Example of a *tandem computer*, first version, in [Cancela 2002]

Method	$\widehat{\gamma}_{cie}$	$\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\} \times t$
FB	1.33E−06	3.37E−14
SFB	1.27E−06	2.06E−15
SFBP	1.27E−06	2.20E−15
Forward Steps	1.21E−06	4.20E−13
Consecutive Failures	1.19E−06	3.93E−13
Measure of Rarity	1.20E−06	5.38E−13

Table 6.5 shows the results obtained for the same system, but with with a four-unit multiprocessor (only one of the four processors is required to have an *operational* system), and with each RAID being composed of 5 drives, only 3 of which are required for the system to be *operational*.

Table 6.5: Example of a *tandem computer*, second version, in [Cancela 2002]

Method	$\widehat{\gamma}_{cie}$	$\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\} \times t$
FB	1.24E−07	1.88E−15
SFB	—	1.57E−16
SFBP	1.25E−07	9.05E−17
Forward Steps	1.19E−07	5.55E−14
Consecutive Failures	1.30E−07	6.17E−14
Measure of Rarity	1.24E−07	1.11E−14

The third system used, also taken from [Cancela 2002], consists of a replicated database in which there are four sites, and each site has a whole copy of the database, on a RAID disk cluster. All clusters are identical, with the same redundancies (7-out-of-9), and with failure rate (for each disk) equal to $\varepsilon = 10^{-2}$. There is one repairman per class, and the repair rate is 1. The system is considered *up* if there is at least one copy of the database working. Results are shown in Table 6.6.

Measure of Rarity is efficient only if failure and repair rates are considerably different. When

this is not the case, the measure of rarity increases significantly at both, failures and repairs and, as a consequence, an increase of such measure is not an indication that the systems is moving towards the target event. In the case of the replicated database, failure and repair rates are, respectively, 10^{-2} and 1. Compared to the rates of the other systems analyzed, these rates are considerably close. This is the reason why *Measure of Rarity* is not computed in Table 6.6.

Table 6.6: Example of a *replicated database* in [Cancela 2002]

Method	$\widehat{\gamma}_{cie}$	$\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\} \times t$
FB	8.54E−13	8.65E−25
SFB	1.16E−12	3.93E−23
SFBP	8.87E−13	2.37E−25
Forward Steps	8.04E−13	4.41E−26
Consecutive Failures	8.10E−13	4.18E−23
Measure of Rarity	—	—

In the second set of experiments the models are the ones used by L'Ecuyer et. al. in [L'Ecuyer 2011b]. In the first case (Example 5 in [L'Ecuyer 2011b]), the system is composed of two sets of processors with two processors per set, two sets of disk controllers with two controllers per set, and six clusters of disks with four disks per cluster. The failure rates for processors, controllers, and disks are 5×10^{-5}, 2×10^{-5} and 2×10^{-5}, respectively. The repair rate is 1 for each type of component. In each disk cluster, data is replicated, which means that the failure of a single disk does not provoke a system's failure. The system is *operational* if all data is accessible from both processor types, meaning that at least one processor of each type, one controller of each set, and three disks of each cluster are *operational*. Results are shown in Table 6.7. BFB, SBLR, ZVA(v_0), ZVA(v_1), ZVA(v_2), and ZVA(v_3) are all *Importance Sampling* methods used in [L'Ecuyer 2011b].

Table 6.7: Example 5 in [L'Ecuyer 2011b] (Exact Value 5.60E−05)

Method	$\widehat{\gamma}_{cie}$	$\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\} \times t$
BFB	—	4.93E−07
SBLR	—	1.17E−03
ZVA(v_0)	—	6.21E−11
ZVA(v_1)	—	3.90E−11
ZVA(v_2)	—	4.80E−11
Forward Steps	5.59E−05	3.08E−11
Consecutive Failures	5.51E−05	2.83E−11
Measure of Rarity	5.28E−05	9.88E−11

The last example is the one referred to as Example 6 in [L'Ecuyer 2011b]. The system is composed of 20 types of components numbered from 0 to 19, with 4 components of each type. All repair rates are assumed to be 1, but component's failure rates differ: type–i components have failure rate $\lambda_i = (1 + i/10)\varepsilon$ for $0 \leq i \leq 9$ and $\lambda_i = i\varepsilon^2/10$ for $10 \leq i \leq 19$, where $\varepsilon = 10^{-3}$. The system is *failed* whenever a total of 7 components are *failed*. Results are shown in Table 6.8.

Table 6.8: Example 6 in [L'Ecuyer 2011b]

Method	$\widehat{\gamma}_{cie}$	$\widehat{\mathbb{V}}\{\widehat{\gamma}_{cie}\} \times t$
BFB	3.10E−11	9.35E−17
SBLR	—	—
ZVA(v_3)	3.00E−11	1.26E−22
Forward Steps	3.03E−11	1.74E−23
Consecutive Failures	2.93E−11	4.28E−22
Measure of Rarity	2.38E−11	8.70E−21

All the *CMIE* estimations can be considered in the same order of precision and efficiency of the other methods to which the comparisons has been made.

6.8 CMIE vs. Splitting

If the sets of intermediate states are cuts, there is a formal equivalence between *CMIE* and *Splitting* [Garvels 2000, Glasserman 1996, L'Ecuyer 2007a, L'Ecuyer 2009, Villén-Altamirano 1991]. In both methods trajectories are managed similarly, but the interpretation of them and, consequently, the way the calculation is made, differs. The results of both estimates coincide, but for certain *Splitting* models for which the determination of the function of importance is particularly difficult, the application of *CMIE* is extremely simple.

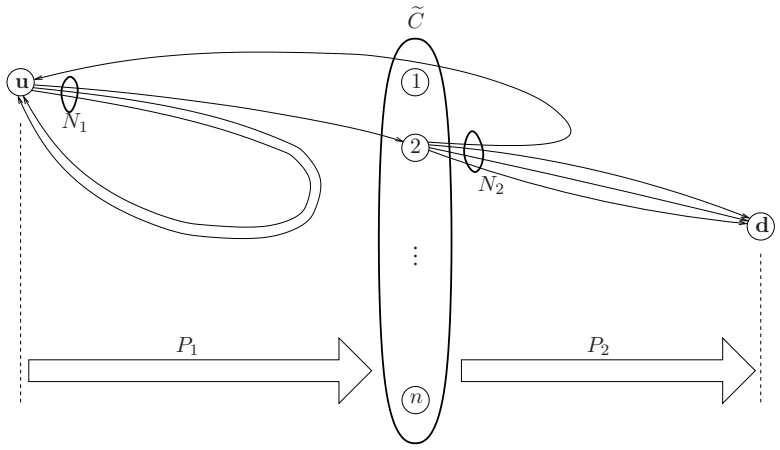


Figure 6.5: Some trajectories in a *CMIE* vs. *Splitting* comparison

When the set $\widetilde{C}$ is a cut, the *CMIE* estimator takes the form:

$$\widehat{\gamma}_{cie} \;=\; \frac{1}{N_1}\sum_{j=1}^{N_1}\left(\sum_{k=1}^{n} I_k^{(j)} \times \frac{1}{N_2}\sum_{i=1}^{N_2} J_k^{(j,i)}\right)$$

$$= \frac{1}{N_1 N_2} \sum_{j=1}^{N_1} \sum_{k=1}^{n} \sum_{i=1}^{N_2} I_k^{(j)} J_k^{(j,i)}. \tag{6.22}$$

A different analysis on the same model shows that, any path starting at state $\mathbf{u}$ has a probability, say P_1, to reach —any state of— the set $\widetilde{C}$ before coming back to $\mathbf{u}$. In the same way, a path starting from any state in the set $\widetilde{C}$ has a probability, say P_2, to reach state $\mathbf{d}$ before coming back to $\mathbf{u}$. The set $\widetilde{C}$ can be seen as a *bound* or *threshold* in the paths going from $\mathbf{u}$ to $\mathbf{d}$ and, therefore, *Splitting* can be applied in the estimation of γ. This *Splitting* estimation takes the form: $\widehat{\gamma} = \widehat{P}_1 \times \widehat{P}_2$, where $\widehat{P}_1$ and $\widehat{P}_2$ are, respectively, standard estimators of P_1 and P_2, as in any ordinary *Splitting* application. Figure 6.5 shows part of a set of replications, some of which start at $\mathbf{u}$ and goes forward to $\widetilde{C}$, and some others that start at $\widetilde{C}$ and goes forward to $\mathbf{d}$. According to this approach, the estimators of P_1 and P_2 are:

$$\widehat{P}_1 = \frac{1}{N_1} \sum_{j=1}^{N_1} \sum_{k=1}^{n} I_k^{(j)},$$

$$\widehat{P}_2 = \frac{\displaystyle\sum_{j=1}^{N_1} \sum_{k=1}^{n} I_k^{(j)} \sum_{i=1}^{N_2} J_k^{(j,i)}}{N_2 \displaystyle\sum_{j=1}^{N_1} \sum_{k=1}^{n} I_k^{(j)}},$$

and the *Splitting* estimator is:

$$\widehat{\gamma}_{spl} = \widehat{P}_1 \times \widehat{P}_2 = \frac{1}{N_1 N_2} \sum_{j=1}^{N_1} \sum_{k=1}^{n} I_k^{(j)} \sum_{i=1}^{N_2} J_k^{(j,i)} = \widehat{\gamma}_{cie}. \tag{6.23}$$

This leads to the conclusion that, if the set $\widetilde{C} = \{1, 2, \ldots, n\}$ is a cut in the graph of the Markov chain, *CMIE* and *Splitting* (based on a single level set) produce the same estimation. In other words, *Splitting* with a single level set $\widetilde{C}$ is the particular case of *CMIE* in which the set $\widetilde{C}$ is a cut in the graph of the Markov chain.

In a basic *Splitting* model there are *bounds* or *thresholds* between the initial and the final state, just like the set $\widetilde{C}$ in Figure 6.5. The consecutive probabilities P_1, P_2, ... need to be estimated somehow. One of them is the probability of reaching the final state from the *threshold* that is immediately before. In systems like the ones introduced in Section 6.7, there are usually more than one final state scattered through all the Markov chain, some of which may be located between *thresholds*. This feature requires a particular effort to design a *Splitting* function of importance, while the application of *CMIE* is straightforward.

Another feature that may cause complications in a basic *Splitting* model is failure propagation. Sometimes a particular failure may cause the simultaneous occurrence of a set of other failures, with a given probability. In a basic *Splitting* model this translates into crossing more than one *threshold* simultaneously, what makes necessary to modify the basic approach according to system

under analysis. *CMIE* is not affected by failure propagation.

6.9 Computational Cost

Let $\pi_{\mathbf{u},k}$ be a path that starts at $\mathbf{u}$ and reach $k \in C$ before coming back to $\mathbf{u}$. Let $L(\pi_{\mathbf{u},k})$ be the length of $\pi_{\mathbf{u},k}$, measured by the number of transitions. Considering that —in general— there is more than one path that starts at $\mathbf{u}$ and reaches $k \in C$ before coming back to $\mathbf{u}$, each one of them with some given probability, $L(\pi_{\mathbf{u},k})$ is a discrete random variable. Call $l_{\mathbf{u},k} = \mathbb{E}\{L(\pi_{\mathbf{u},k})\}$.

Similarly, $l_{k,\mathbf{d}} = \mathbb{E}\{L(\pi_{k,\mathbf{d}})\}$ is the mean number of transitions of a path that starts at state $k \in C$ and hits state $\mathbf{d}$ before coming back to $\mathbf{u}$.

Calling T, the number of transitions of the whole simulation:

$$
\begin{aligned}
T &= \sum_{j=1}^{N_1}\left(I_0^{(j)} L(\pi_{\mathbf{u},\mathbf{d}}^{(j)}) + \sum_{k=1}^{n} I_k^{(j)}\left(L(\pi_{\mathbf{u},k}^{(j)}) + \sum_{i=1}^{N_2} J_k^{(j,i)} L(\pi_{k,\mathbf{d}}^{(i)})\right)\right) \\
&= \sum_{j=1}^{N_1}\left(I_0^{(j)} L(\pi_{\mathbf{u},\mathbf{d}}^{(j)}) + \sum_{k=1}^{n}\left(I_k^{(j)} L(\pi_{\mathbf{u},k}^{(j)}) + \sum_{i=1}^{N_2} I_k^{(j)} J_k^{(j,i)} L(\pi_{k,\mathbf{d}}^{(i)})\right)\right) \\
\mathbb{E}\{T\} &= \sum_{j=1}^{N_1}\left(\mathbb{E}\{I_0^{(j)}\}\mathbb{E}\{L(\pi_{\mathbf{u},\mathbf{d}}^{(j)})\} + \sum_{k=1}^{n}\left(\mathbb{E}\{I_k^{(j)}\}\mathbb{E}\{L(\pi_{\mathbf{u},k}^{(j)})\} + \right.\right. \\
&\qquad \left.\left. \sum_{i=1}^{N_2}\mathbb{E}\{I_k^{(j)}\}\mathbb{E}\{J_k^{(j,i)}\}\mathbb{E}\{L(\pi_{k,\mathbf{d}}^{(i)})\}\right)\right) \\
&= \sum_{j=1}^{N_1}\left(p_0\, l_{\mathbf{u},\mathbf{d}} + \sum_{k=1}^{n}\left(p_k\, l_{\mathbf{u},k} + \sum_{i=1}^{N_2} p_k\, \gamma_k\, l_{k,\mathbf{d}}\right)\right) \\
&= N_1\left(p_0\, l_{\mathbf{u},\mathbf{d}} + \sum_{k=1}^{n}\left(p_k\, l_{\mathbf{u},k} + N_2 p_k\, \gamma_k\, l_{k,\mathbf{d}}\right)\right) \\
&= N_1\, p_0\, l_{\mathbf{u},\mathbf{d}} + \sum_{k=1}^{n} N_1\, p_k\, \left(l_{\mathbf{u},k} + N_2\, \gamma_k\, l_{k,\mathbf{d}}\right)
\end{aligned}
$$

CHAPTER 7

Conclusions

Contents

Abstract

This chapter presents the concluding remarks of this thesis. First, the main contributions are briefly highlighted. Then, two sections are dedicated to the conclusions regarding the two main method families proposed and studied in this thesis, *Splitting/CP* and *CMIE*. The last section is devoted to discuss possible research lines to continue this work.

7.1 Main Contributions

This thesis has explored some *Monte Carlo* methods designed to reduce the variance of the estimator in a context of *rare events*. As a result, two methods have been proposed.

The first one, *Splitting* on the *Creation Process* (Splitting/CP), applies to the *reliability* estimation on a static model of a communication network for which the links are extremely reliable. The other one, *Conditional Monte Carlo with Intermediate Estimations* (*CMIE*), is intended to estimate *dependability* parameters on Markovian systems for which failure is associated with states that are visited with extremely low probability.

Both methods were studied empirically by doing several numerical experiments over many well known benchmark models. The results of these experiments show that the performances of the two methods are in the order of those of the best known methods. In the case of *CMIE* some of its properties were demonstrated and, besides, its variance was given a closed form.

Before addressing each of the two proposed methods, *Splitting* and *Conditional Monte Carlo* were studied on a number of basic models. In the particular case of *Splitting*, the variance was determined for two different settings, in one of them (*Fixed Splitting*) the result matches the one that was already published, in the other setting (*Fixed Effort*) the variance determination is a contribution since, as far as we know, it is not in the literature.

Ultimately, the main contribution of this thesis is to give new evaluation methods and useful ideas for simulating systems subject to *rare events*.

7.2 Splitting on Network Reliability Estimation

The first of the proposed methods —named in this thesis, *Splitting/CP*— applies to estimate the *reliability* of highly reliable communications networks. *Splitting/CP* is as an improvement of the *Creation Process*, a type of simulation in which the model evolves through a fictitious time t. At the beginning, all the links are *failed* and they become *repaired* after random times that are proportional to their single *reliability*. If the *structure function*, $\phi(\mathbf{X})$, equals 1 earlier than $t = 1$, the replication ends and the network is considered *operational*. Otherwise the replication ends and the network is considered *failed*. The proposal of this thesis is to partition the period $0 \leq t \leq 1$ by means of *thresholds* and to split or multiply every one of the network's temporal evolutions, just like "ordinary" applications of *Splitting* do to the trajectories on the state space.

All the work and, consequently, most of the conclusions around this proposal has been supported empirically. In the first set of experiments, the accuracy and the computational efficiency of *Splitting/CP* were compared to the results of different methods taken from the literature. This comparison shows that, except from RVR [Cancela 2003], *Splitting/CP* performs in the order of most of the well–known methods to which it was compared, namely Creation, Destruction and Merge Process, Sequential Construction and Destruction, Dagger, Bound–based sampling, Failure Sets Method, Total Hazard, Leap–Evolve and Tree–Merge and Cross Entropy over Merge, Permutation and *Crude Monte Carlo*. Experiments also lead to the conclusion that *Splitting/CP* behaves more efficiently on sparse graph networks (like the Easton–Wong or the 6×6–Grid) than on dense graph networks (like the Dodecahedron or the complete networks).

In the second set of experiments, simulations were conducted on the Dodecahedron network, with increasing *reliability*. The *relative error* $\mathbb{V}\{\widehat{Q}\}^{1/2}/\mathbb{E}\{\widehat{Q}\}$ of the *Splitting/CP* estimator $\widehat{Q}$ is very small and stays quite stable for widely varying values of failure rarity. There is actually a very small growth when *reliability* approaches 1, but a change in sixteen orders of magnitude of *unreliability* translates into less than one order of magnitude increase of *relative error* (and the growing step becomes smaller when the network *reliability* grows). This suggests that the method can attain very small *relative error* values, no matter how reliable the network is.

In the last set of experiments the robustness of *Splitting/CP* was empirically analyzed. The Coverage Factor, defined as the proportion of times that the exact value lies inside a confidence interval, was determined for three different confidence levels: 90%, 95% and 99%. Such a Coverage Factor was calculated for different (increasing) values of the system *reliability*, varying the single link *unreliability* as the measure of rarity ε. The tests were performed on C_{10} (the ten nodes complete network), and the Dodecahedron, varying the measure of rarity so as to let the whole network *unreliability* span from 2×10^{-3} down to 2×10^{-21}. All the Coverage Factors were significantly tight to the confidence level (all of them within $\pm 1\%$ of the confidence level). In addition, a test proposed by Schruben was also applied; this test considers the full coverage function (not just its value at some of its levels), and allows for a more strict evaluation of the appropriateness of the confidence interval. The results obtained showed an excellent behaviour of the *Splitting/CP* method, where the computed confidence intervals with respect the normality assumptions for an extremely wide range of *reliability* values. These results show that the *Splitting/CP* estimator is extremely robust.

7.3 Conditional Monte Carlo on Markovian Systems

The second method proposed in this thesis, referred to as *Conditional Monte Carlo with Intermediate Estimations* (*CMIE*), is a variance reduction technique developed to enhance simulations in the context of *rare events* on markovian systems. *CMIE* is based on *Conditional Monte Carlo*, a classical variance reduction technique, whose use is not widespread in the field of *rare events*. *CMIE* was conceived to estimate the probability of visiting the failure state before coming back to the initial state (accepted as the state in which the system is *up*). The application of ordinary *Conditional Monte Carlo* requires the knowledge of some exact probabilities in the model. To overcome the fact that this probabilities are likely to be unknown, the proposal of *CMIE* is to estimate them, for which it is necessary to launch the method, recursively, from some selected states called intermediate states.

From this point of view, *Splitting* can be considered the particular case of *CMIE* in which the events are implicitly defined by *thresholds* in the state space of the Markov chain. However, the way in which the target probability is recursively computed in *CMIE* is simpler than the *Splitting* algorithm, which needs to determine the probabilities of crossing each *threshold* conditioned to the previous cross, and has to keep track of the number of times each *threshold* is crossed.

Another advantage of *CMIE* over *Splitting*, comes up in systems in which there are more than one target state and/or fault propagation. The presence of more than one target state is a drawback in the determination of *thresholds* (cuts in the graph). Due to the presence of fault propagation, multiple *thresholds* crosses may occur. A particular effort then is required to adapt *Splitting* to these particular settings, whereas the *CMIE* implementations are straightforward and do not differ with respect to ones in which there is only one target state and there is no fault propagation.

Based on the variance determination made for *CMIE*, some properties of the intermediate sets were derived. One of them, referred to the number of states in a single set, says that the addition of states to an existing set never increases the variance and, therefore, a variance reduction may be expected. The other one is similar in the sense that the addition of sets of states never increases the variance and, therefore, a variance reduction may be expected. It is to remark that the last of these two properties was proven for the particular case in which the sets of intermediate states are cuts with no intersection states between them.

CMIE was adapted to the case of large multicomponent systems. In these settings the number of states is usually extremely high and explicit selection of intermediate states may not be easy, except for selections that can be made regarding the number of —*failed* or *operational*— components. But it is also possible, and in this case useful, to condition to different events, other that just visiting the intermediate states. Three variants were proposed on this regard: *Forward Steps*, in which the event occurs if the systems moves a number of D steps closer to the target state, *Consecutive Failures*, where the event occurs if a sequence of D failures (with no repairs in between) occur and, *Measure of Rarity*, in which the event occurs if, after setting a measure of rarity for the paths that the systems takes through the graph, a path is such that the measure exceeds some bound B.

Both methods, *CMIE* with explicit intermediate states selection and *CMIE* adapted to large multicomponent systems, were empirically tested over the models used in different papers.

First, a 14 states continuous time Markov chain used by Juneja and Shahabuddin in

[Juneja 2001] was used to make an experimental verification of the properties derived for the sets of intermediate states. The sets of intermediate states were determined explicitly. As expected, Table 6.1 shows that the set with the lowest variance associated is the closest cut to the initial state and Table 6.2 shows that, as the number of sets of intermediates states increases, the variance of the estimator decreases. Table 6.3 shows the variance reduction capacity of *CMIE* compared to the variance of a standard estimation in three scenarios selected with, respectively, one, two and three sets of intermediate states. As expected, the largest variance reduction (measured as a ratio to the standard estimator variance) is achieved in the case of three sets of intermediate states.

In the second set of experiments the variant of *CMIE* adapted to large multicomponent systems, was tested on a series of experiments taken from some other papers. These tests were used to compare the product $variance \times execution\ time$ of *CMIE* against the other methods', all of which are derived from *Importance Sampling*. The product $variance \times execution\ time$ of *CMIE* was above the product of some of the other methods and below the product of some others. However, the differences are narrow enough to consider that the methods perform in the same order of efficiency.

CMIE can be easily extended to other type of *rare event* problem like, for instance, network *reliability* estimation.

7.4 Open Research Lines

Concerning *Splitting/CP*, one possible line of future work is to make a more insightful analysis of the asymptotic behaviour of accuracy and robustness. The experimental results suggest that the method might be close to conditions like Bounded Relative Error and Bounded Normal Approximation. However, this topic merits a more detailed study, attempting to prove this more formally.

Further analysis may include the number of *thresholds* in *Splitting/CP*. Certainly, there must be an optimal number of *thresholds* for every graph topology and every *reliability* value. Some observations were pointed out in this concern. However, this determination is critical, as it has a direct impact on the efficiency of the method. A few guidelines were given in this thesis in order to make an initial attempt. These guidelines are supported by recommendations made for a general case, making use of upper and lower bounds. But a more formal analysis on this respect would be worthwhile.

Another *Splitting/CP* issue that should be developed in more detail is the trade–off between accuracy and execution time. An estimation is globally efficient if a desired accuracy level can be obtained in a reasonable time. The work in this thesis has been focused, mostly, on increasing the accuracy. But this increasing is, obviously, achieved at the expense of the execution time. A more detailed analysis of the trade–off between accuracy and execution time is, therefore, a useful line of work.

Reagarding *CMIE*, it is also of interest to analyse the asymptotic behaviour and to see how close or how far it is to have Bounded Relative Error and/or Bounded Normal Approximation.

A topic to improve in *CMIE* is the selection of the intermediate states sets. For the case of only one set, it was proven that, from the accuracy point of view, the best selection is the closest cut to the initial state. In the case of two sets, the selection of the second set is tied to the graph topology and is not straightforward. This topic deserves further attention in order to generalize the

mechanism, not only for two but also for more than two sets of intermediate states.

Like in *Splitting/CP*, and becasue of similar reasons, an important issue to work on is the trade–off between accuracy and execution time. In the case of only one set of intermediate states, this analysis may give support to decide the most appropriate "location" for such set; in the case of more than one set it may help to decide not only the "locations", but also the number of sets to use.

All these lines of work, either for *Splitting/CP* or *CMIE*, would be greatly simplified if closed form expressions for the variance were given. However, there are alternatives (eg. the use of bounds) for cases —like *Splitting/CP*— in which these expressions seem to be very difficult to obtain.

APPENDIX A

Variance in the case of two sets of Intermediate States

In this appendix, the variance of the *CMIE* estimator, for the case of two intermediate states, $\widetilde{C}_1$ and $\widetilde{C}_2$, is developed. The probabilities used are shown in Figure 6.2.

$$\widehat{\gamma}_{cie} = \frac{1}{N_1 N_2 N} \sum_{l=0}^{n_1} \sum_{k=0}^{n_2} \sum_{r=1}^{N_1} \sum_{j=1}^{N_2} \sum_{i=1}^{N} H_l^{(r)} I_{lk}^{(r,j)} J_k^{(r,j,i)}.$$

Let $\bar{H}^{(x)}$ be any possible replication of $\bar{H}$, what means that $H_0^{(x)}$, $H_1^{(x)}$, ..., $H_{n_1}^{(x)}$, are the components of this replication of $\bar{H}$. Use these value to condition the estimator in order to calculate its variance:

$$\mathbb{V}\{\widehat{\gamma}_{cie}\} = \underbrace{\mathbb{V}\{\mathbb{E}\{\widehat{\gamma}_{cie} \mid \bar{H}^{(x)}\}\}}_{A} + \underbrace{\mathbb{E}\{\mathbb{V}\{\widehat{\gamma}_{cie} \mid \bar{H}^{(x)}\}\}}_{B}.$$

$$\begin{aligned}
A &= \mathbb{V}\left\{\mathbb{E}\left\{\frac{1}{N_1 N_2 N} \sum_{l=0}^{n_1} \sum_{k=0}^{n_2} \sum_{r=1}^{N_1} \sum_{j=1}^{N_2} \sum_{i=1}^{N} H_l^{(x)} I_{lk}^{(r,j)} J_k^{(r,j,i)}\right\}\right\} \\
&= \mathbb{V}\left\{\frac{1}{N_1 N_2 N} \sum_{l=0}^{n_1} \sum_{k=0}^{n_2} \sum_{r=1}^{N_1} \sum_{j=1}^{N_2} \sum_{i=1}^{N} H_l^{(x)} \mathbb{E}\left\{I_{lk}^{(r,j)} J_k^{(r,j,i)}\right\}\right\} \\
&= \mathbb{V}\left\{\frac{1}{N_1 N_2 N} \sum_{l=0}^{n_1} \sum_{k=0}^{n_2} \sum_{r=1}^{N_1} \sum_{j=1}^{N_2} \sum_{i=1}^{N} H_l^{(x)} \mathbb{E}\left\{I_{lk}^{(r,j)}\right\} \mathbb{E}\left\{J_k^{(r,j,i)}\right\}\right\} \\
&= \mathbb{V}\left\{\frac{1}{N_1 N_2 N} \sum_{l=0}^{n_1} \sum_{k=0}^{n_2} \sum_{r=1}^{N_1} \sum_{j=1}^{N_2} \sum_{i=1}^{N} H_l^{(x)} p_{lk} \gamma_k\right\} \\
&= \mathbb{V}\left\{\frac{1}{N_1 N_2 N} \sum_{l=0}^{n_1} \sum_{k=0}^{n_2} \sum_{r=1}^{N_1} \sum_{j=1}^{N_2} \sum_{i=1}^{N} H_l^{(x)} p_{lk} \gamma_k\right\} \\
&= \mathbb{V}\left\{\frac{1}{N_1} \sum_{l=0}^{n_1} \sum_{k=0}^{n_2} \sum_{r=1}^{N_1} H_l^{(x)} p_{lk} \gamma_k\right\} \\
&= \mathbb{V}\left\{\frac{1}{N_1} \sum_{r=1}^{N_1} \sum_{l=0}^{n_1} H_l^{(x)} \sum_{k=0}^{n_1} p_{lk} \gamma_k\right\}
\end{aligned}$$

$$= \mathbb{V}\left\{\frac{1}{N_1}\sum_{r=1}^{N_1}\sum_{l=0}^{n_1} H_l^{(x)}\,\gamma'_l\right\}.$$

For any given x, only one of the values $H_0^{(x)}$, $H_1^{(x)}$, ..., $H_{n_1}^{(x)}$, equals 1, and the rest equals 0. As $H_i^{(x)} = 1$ w.p. p_i, the term $\sum_{l=0}^{n_1} H_l^{(x)}\gamma'_l$ equals γ'_i w.p. p_i. To evaluate the variance, randomness of the expression between braces is only due to $H_l^{(x)}$, thus, $\sum_{l=0}^{n_1} H_l^{(x)}\gamma'_l$ is a random variable on the state space $\{\gamma'_0, \gamma'_1, \ldots, \gamma'_{n_1}\}$, with probabilities $\{p_0, p_1, \ldots, p_{n_1}\}$. Then:

$$A \;=\; \frac{1}{N_1}\left(\sum_{l=0}^{n_1} p_l\,\gamma_l'^2 - \left(\sum_{l=0}^{n_1} p_l\,\gamma'_l\right)^2\right) \;=\; \frac{1}{N_1}\left(\sum_{l=0}^{n_1} p_l\,\gamma_l'^2 - \gamma^2\right),$$

and for the remaining term:

$$\begin{aligned}
B &= \mathbb{E}\left\{\mathbb{V}\left\{\frac{1}{N_1 N_2 N}\sum_{l=0}^{n_1}\sum_{k=0}^{n_2}\sum_{r=1}^{N_1}\sum_{j=1}^{N_2}\sum_{i=1}^{N} H_l^{(x)}\,I_{lk}^{(r,j)}\,J_k^{(r,j,i)}\right\}\right\}\\
&= \mathbb{E}\left\{\frac{1}{N_1^2}\sum_{l=0}^{n_1}\sum_{r=1}^{N_1}\left(H_l^{(x)}\right)^2\,\mathbb{V}\left\{\frac{1}{N_2 N}\sum_{k=0}^{n_2}\sum_{j=1}^{N_2}\sum_{i=1}^{N} I_{lk}^{(r,j)}\,J_k^{(r,j,i)}\right\}\right\}\\
&= \mathbb{E}\left\{\frac{1}{N_1^2}\sum_{l=0}^{n_1}\sum_{r=1}^{N_1} H_l^{(x)}\,\mathbb{V}\left\{\frac{1}{N_2 N}\sum_{k=0}^{n_2}\sum_{j=1}^{N_2}\sum_{i=1}^{N} I_{lk}^{(r,j)}\,J_k^{(r,j,i)}\right\}\right\}\\
&= \mathbb{E}\left\{\frac{1}{N_1^2}\sum_{l=0}^{n_1}\sum_{r=1}^{N_1} H_l^{(x)}\,\mathbb{V}\left\{\widehat{\gamma}'_l\right\}\right\}\\
&= \frac{1}{N_1^2}\sum_{l=0}^{n_1}\sum_{r=1}^{N_1}\sum_{i=1}^{N}\mathbb{E}\left\{H_l^{(x)}\right\}\,\mathbb{V}\left\{\widehat{\gamma}'_l\right\}\\
&= \frac{1}{N_1}\sum_{l=0}^{n_1} p_l\,\mathbb{V}\left\{\widehat{\gamma}'_l\right\} \qquad\qquad (A.1)\\
&= \frac{1}{N_1}\sum_{l=0}^{n_1} p_l\,\left(\frac{1}{N_2}\left(\sum_{k=0}^{n_2} p_{lk}\gamma_k^2 - \gamma_l'^2\right) + \frac{1}{N_2 N}\left(\gamma'_l - \sum_{k=0}^{n_2} p_{lk}\,\gamma_k^2\right)\right).
\end{aligned}$$

Expression (A.1) can be used to derive the case of more than two sets of intermediate states.

$$\mathbb{V}\{\widehat{\gamma}_{cie}\} \;=\; A + B$$

$$
\begin{aligned}
&= \frac{1}{N_1}\left(\sum_{l=0}^{n_1} p_l\, {\gamma'_l}^2 - \gamma^2\right) + \frac{1}{N_1}\sum_{l=0}^{n_1} p_l\, \mathbb{V}\left\{\widehat{\gamma'_l}\right\} \\
&= \frac{1}{N_1}\left(\sum_{l=0}^{n_1} p_l\, {\gamma'_l}^2 - \gamma^2\right) + \\
&\quad \frac{1}{N_1}\sum_{l=0}^{n_1} p_l\, \left(\frac{1}{N_2}\left(\sum_{k=0}^{n_2} p_{lk}\gamma_k^2 - {\gamma'_l}^2\right) + \frac{1}{N_2 N}\left(\gamma'_l - \sum_{k=0}^{n_2} p_{lk}\, \gamma_k^2\right)\right)
\end{aligned}
$$

APPENDIX B

Variance comparison between the cases of one and two sets of Intermediate States

In this appendix it is shown that if a second set of intermediate states $\widetilde{C}_2$ is added to a model with a single set $\widetilde{C}_1$, the variance of the resulting estimator is less than or equal the variance in the case with only $\widetilde{C}_1$. The proof is based on a comparison between the formula obtained in Appendix A, for two sets, and the variance expression shown in (6.11), for one set.

Referring to Figure 6.2, omitting the set $\widetilde{C}_2$ and considering that $\widetilde{C}_1$ is the only set of intermediate states, the variance of the *CMIE* estimator, shown in (6.11), can be written as:

$$\begin{aligned}
\mathbb{V}\{\widehat{\gamma}_{cie}\} &= \frac{1}{N_1}\left(\sum_{l=0}^{n_1} p_l {\gamma'_l}^2 - \gamma^2\right) + \frac{1}{N_1 N_2}\left(\gamma - \sum_{l=0}^{n_1} p_l {\gamma'_l}^2\right) \\
&= \frac{1}{N_1}\left(\sum_{l=0}^{n_1} p_l {\gamma'_l}^2 - \gamma^2\right) + \frac{1}{N_1 N_2}\left(\sum_{l=0}^{n_1} p_l \gamma'_l - \sum_{l=0}^{n_1} p_l {\gamma'_l}^2\right) \\
&= \frac{1}{N_1}\left(\sum_{l=0}^{n_1} p_l {\gamma'_l}^2 - \gamma^2\right) + \frac{1}{N_1 N_2}\sum_{l=0}^{n_1} p_l(\gamma'_l - {\gamma'_l}^2).
\end{aligned}$$

Considering now Figure 6.2 with both sets of intermediate states, $\widetilde{C}_1$ and $\widetilde{C}_2$, the variance of the *CMIE* estimator, developed in Appendix A, is:

$$\begin{aligned}
\mathbb{V}\{\widehat{\gamma}_{cie}\} &= \frac{1}{N_1}\left(\sum_{l=0}^{n_1} p_l\, {\gamma'_l}^2 - \gamma^2\right) + \\
&\quad \frac{1}{N_1}\sum_{l=0}^{n_1} p_l \left(\frac{1}{N_2}\left(\sum_{k=0}^{n_2} p_{lk}\gamma_k^2 - {\gamma'_l}^2\right) + \frac{1}{N_2 N}\left(\gamma'_l - \sum_{k=0}^{n_2} p_{lk}\,\gamma_k^2\right)\right) \\
&= \frac{1}{N_1}\left(\sum_{l=0}^{n_1} p_l\, {\gamma'_l}^2 - \gamma^2\right) + \\
&\quad \frac{1}{N_1 N_2}\sum_{l=0}^{n_1} p_l \left(\sum_{k=0}^{n_2} p_{lk}\gamma_k^2 - {\gamma'_l}^2 + \frac{1}{N}\left(\gamma'_l - \sum_{k=0}^{n_2} p_{lk}\,\gamma_k^2\right)\right).
\end{aligned}$$

An attempt to prove that the latter variance is lower than or equal the prior one, translates into

proving that:

$$\sum_{k=0}^{n_2} p_{lk}\gamma_k^2 - {\gamma'_l}^2 + \frac{1}{N}\left(\gamma'_l - \sum_{k=0}^{n_2} p_{lk}\,\gamma_k^2\right) \leq \gamma'_l - {\gamma'_l}^2$$

$$\sum_{k=0}^{n_2} p_{lk}\gamma_k^2 + \frac{1}{N}\left(\gamma'_l - \sum_{k=0}^{n_2} p_{lk}\,\gamma_k^2\right) \leq \gamma'_l$$

$$\frac{1}{N}\left(\gamma'_l - \sum_{k=0}^{n_2} p_{lk}\,\gamma_k^2\right) \leq \gamma'_l - \sum_{k=0}^{n_2} p_{lk}\gamma_k^2.$$

Note that

$$\gamma'_l - \sum_{k=0}^{n_2} p_{lk}\gamma_k^2 = \sum_{k=0}^{n_2} p_{lk}\gamma_k - \sum_{k=0}^{n_2} p_{lk}\gamma_k^2 = \sum_{k=0}^{n_2} p_{lk}(\underbrace{\gamma_k - \gamma_k^2}_{\geq 0}) \geq 0,$$

which, considering that $N \geq 1$, completes the proof.

It is simple to extend this proof to show that if additional cut is added to a model with Z sets of intermediate states, the variance in the model with $Z+1$ sets will be lower than or equal the variance in the model with Z intermediate states.

Bibliography

[Amrein 2011] M. Amrein and H. R. Künsch. *A Variant of Importance Splitting for Rare Event Estimation: Fixed Number of Successes*. ACM Trans. Model. Comput. Simul., vol. 21, no. 2, pages 13:1–13:20, February 2011. 18

[Botev 2010] Z. I. Botev and D. P. Kroese. *Efficient Monte Carlo Simulation via the Generalized Splitting Method*. Statistics and Computing, pages 1–16, 2010. DOI: 10.1007/s11222-010-9201-4. 34

[Botev 2013] Z. I. Botev, P. L'Ecuyer and B. Tuffin. *Markov chain importance sampling with applications to rare event probability estimation*. Statistics and Computing, vol. 23, no. 2, pages 271–285, 2013. 74

[Cancela 1995] H. Cancela and M. El Khadiri. *A Recursive Variance–Reduction Algorithm for Estimating Communication–Network Reliability*. IEEE Transactions on Reliability, vol. 44, no. 4, pages 595–602, December 1995. 34

[Cancela 2002] H. Cancela, G. Rubino and B. Tuffin. *MTTF estimation using Importance Sampling on Markov models*. Monte Carlo Methods and Applications, vol. 8, no. 4, pages 321–341, 2002. xiii, 73, 74, 88, 94, 95

[Cancela 2003] H. Cancela and M. El Khadiri. *The Recursive Variance–Reduction Simulation Algorithm for Network Reliability Evaluation*. IEEE Transactions on Reliability, vol. 52, no. 2, pages 207–212, June 2003. 34, 40, 42, 100

[Cancela 2008] H. Cancela, M. El Khadiri and G. Rubino. *A New Simulation Method Based on the RVR Principle for the Rare Event K–Network Reliability Problem*. Technical Report INCO 08–18, Facultad de Ingeniería, Universidad de la República, 2008. 34, 43, 44

[Cancela 2009] H. Cancela, M. El Khadiri and G. Rubino. *Rare events analysis by Monte Carlo techniques in static models*. In G. Rubino and B. Tuffin, editors, Rare Event Simulation using Monte Carlo Methods, chapter 7, pages 145–170. John Wiley & Sons, 2009. 34

[Colbourn 1987] C. J. Colbourn. The combinatorics of network reliability. Oxford University Press, Inc., New York, NY, USA, 1987. 47

[Cook 2007] J. L. Cook and J. E. Ramírez-Marquez. *Two–terminal reliability analyses for a mobile ad hoc wireless network*. Reliability engineering & systems safety, vol. 92, no. 6, pages 821–829, 2007. 34

[Easton 1980] M. Easton and C. Wong. *Sequential destruction method for Monte Carlo evaluation of system reliability*. IEEE Transactions on Reliability, vol. 29, no. 1, pages 27–32, 1980. 34, 40

[Elperin 1991] T. Elperin, I. B. Gertsbakh and M. Lomonosov. *Estimation of Network Reliability Using Graph Evolution Models*. IEEE Transactions on Reliability, vol. 40, no. 5, pages 572–581, Dec 1991. 34, 35, 40, 41, 42, 44

[Fishman 1986] G. S. Fishman. *A Monte Carlo sampling plan for estimating network reliability*. Operational Research, vol. 34, pages 581–594, 1986. 34, 40

[Fishman 1996] G. S. Fishman. Monte Carlo: Concepts, algorithms, and applications. Springer Series in Operations Research. Springer–Verlag, New York, 1996. 14

[Flajolet 2009] P. Flajolet and R. Sedgewick. Analytic combinatorics. Cambridge University Press, New York, NY, USA, 2009. 70

[Garvels 1998] M. J. J. Garvels and D. P. Kroese. *A Comparison of RESTART Implementations*. In Proceedings of the 1998 Winter Simulation Conference, pages 601–609. IEEE Press, 1998. 19

[Garvels 2000] M. J. J. Garvels. *The Splitting Method in Rare Event Simulation*. PhD thesis, Faculty of mathematical Science, University of Twente, The Netherlands, 2000. 18, 34, 38, 96

[Glasserman 1996] P. Glasserman, P. Heidelberger, P. Shahabuddin and T. Zajic. *Splitting for Rare Event Simulation: Analysis of Simple Cases*. In Proceedings of the 1996 Winter Simulation Conference, pages 302–308, San Diego, California, 1996. IEEE Computer Society Press. 18, 24, 30, 34, 96

[Glynn 1989] P. W. Glynn and D. L. Iglehart. *Importance Sampling for Stochastic Simulations*. Management Science, vol. 35, no. 11, pages 1367–1392, November 1989. 74

[Goodman 1962] L. A. Goodman. *The Variance of the Product of K Random Variables*. Journal of the American Statistical Association, vol. 57, no. 297, pages 54–60, March 1962. 31

[Goyal 1992] A. Goyal, P. Shahabuddin, P. Heidelberger, V. F. Nicola and P. W. Glynn. *A Unified Framework for Simulating Markovian Models of Highly Dependable Systems*. IEEE Transactions on Computers, vol. 41, no. 1, pages 36–51, 1992. 73, 74

[Günnec 2007] D. Günnec and F. S. Salman. *Assessing the Reliability and the Expected Performance of a Network under Disaster Risk*. In Proceedings of the International Network Optimization Conference, INOC 2007, Spa, Belgium, 2007. 34

[Hui 2003] K-P Hui, N. Bean, M. Kraetzl and D. P. Kroese. *The tree cut and merge algorithm for estimation of network reliability*. Probability in the Engineering and Informational Sciences, vol. 17, no. 1, pages 23–45, 2003. 34, 40, 43

[Hui 2005] K-P Hui, N. Bean, M. Kraetzl and D. P. Kroese. *The Cross–Entropy Method for Network Reliability Estimation*. Annals of Operations Research, vol. 134, pages 101–118(18), February 2005. 34, 40, 43

[Jun 1992] C. H. Jun and S. M. Ross. *System Reliability by Simulation: Random Hazards Versus Importance Sampling*. Probability in the Engineering and Informational Sciences, vol. 6, pages 119–126, 1992. 40

[Juneja 2001] S. Juneja and P. Shahabuddin. *Fast Simulation of Markov Chains with Small Transition Probabilities*. Management Science, vol. 47, no. 4, pages 547–562, April 2001. 73, 88, 102

[Karp 1983] R. M. Karp and M. G. Luby. *A new Monte Carlo method for estimating the failure probability of an n–component system*. Technical report, Computer Science Division, University of California, 1983. 34, 40

[Knuth 1994] D. E. Knuth and S. Levy. The cweb system of structured documentation: Version 3.0. Addison–Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1994. 41

[Kroese 2013] D. P. Kroese, T. Taimre and Z. I. Botev. Handbook of Monte Carlo Methods. Wiley Series in Probability and Statistics. Wiley, 2013. 34, 62

[Kumamoto 1977] H. Kumamoto, K. Tanaka and K. Inoue. *Efficient Evaluation of System Reliability by Monte Carlo Method*. IEEE Transactions on Reliability, vol. R–26, no. 5, pages 311–315, Dec. 1977. 34, 40

[Kumamoto 1980] H. Kumamoto, K. Tanaka, K. Inoue and E. J. Henley. *Dagger–Sampling Monte Carlo for System Unavailability Evaluation*. IEEE Transactions on Reliability, vol. R–29, no. 2, pages 122–125, June 1980. 34, 40

[L'Ecuyer 2007a] P. L'Ecuyer, V. Demers and B. Tuffin. *Rare Events, Splitting, and quasi-Monte Carlo*. ACM Trans. Model. Comput. Simul., vol. 17, no. 2, April 2007. 18, 19, 34, 38, 96

[L'Ecuyer 2007b] P. L'Ecuyer and B. Tuffin. *Effective Approximation of Zero-Variance Simulation in a Reliability Setting*. In Proceedings of the 2007 European Simulation and Modeling Conference, pages 48–54, Ghent, Belgium, 2007. EUROSIS. 15, 16, 74

[L'Ecuyer 2009] P. L'Ecuyer, F. Le Gland, P. Lezaud and B. Tuffin. *Splitting Techniques*. In G. Rubino and B. Tuffin, editors, Rare Event Simulation using Monte Carlo Methods, chapter 3, pages 39–61. John Wiley & Sons, 2009. 18, 96

[L'Ecuyer 2011a] P. L'Ecuyer, G. Rubino, S. Saggadi and B. Tuffin. *Approximate Zero-Variance Importance Sampling for Static Network Reliability Estimation*. IEEE Transactions on Reliability, vol. 8, no. 4, pages 590–604, 2011. 15, 16, 74

[L'Ecuyer 2011b] P. L'Ecuyer and B. Tuffin. *Approximating Zero-Variance Importance Sampling in a Reliability Setting*. Annals of Operations Research, vol. 189, no. 1, pages 277–297, 2011. xiii, 15, 16, 74, 88, 95, 96

[Li 2004] Yan Li. *Measuring Individual Residence's Accessible Probability by using Geographical Information Systems*. In Proceedings of the 2nd Symposium on Transportation Network Reliability (INSTR 2004), volume 3, pages 239–244, Christchurch, New Zealand, 2004. 34

[Lin 2006] L. Lin and M. Gen. *A Self–controlled Genetic Algorithm for Reliable Communication Network Design*. In IEEE Congress on Evolutionary Computation, pages 640–647, 2006. 34

[Lomonosov 1994] M. Lomonosov. *On Monte Carlo Estimates in Network Reliability*. Probability in the Engineering and Informational Sciences, vol. 8, pages 245–264, 1994. 34

[Marotta 2010] A. Marotta, H. Cancela, V. Peralta and R. Ruggia. *Reliability Models for Data Integration Systems*. In Hoang Pham, Javier Faulin, Angel A. Juan, Sebastián Martorell and José-Emmanuel Ramírez-Márquez, editors, Simulation Methods for Reliability and Availability of Complex Systems, Springer Series in Reliability Engineering, pages 123–144. Springer London, 2010. 34

[Marseguerra 2005] M. Marseguerra, E. Zio, L. Podofillini and D. W. Coit. *Optimal design of reliable network systems in presence of uncertainty*. IEEE Transactions on Reliability, vol. 54, no. 2, pages 243–253, 2005. 34

[Murray 2007a] L. Murray. *Comparación entre Cinco Métodos de Monte Carlo para Estimar la Confiabilidad Diámetro Acotada de Redes de Comunicaciones*. Technical Report INCO 07-07, Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay, 2007. ISSN: 0797-6410. 5

[Murray 2007b] L. Murray and H. Cancela. *Comparison of Five Monte Carlo Methods to Estimate the Network Diameter Constrained Reliability*. In Proceedings of the XXXII Latin-American Conference on Informatics (CLEI), San José, Costa Rica, 2007. 5

[Murray 2008a] L. Murray and H. Cancela. *Monte Carlo Splitting Technique for Source-Terminal Network Reliability Estimation*. In Proceedings of the XXXIV Latin-American Conference on Informatics (CLEI), Santa Fé, Argentina, 2008. 6, 34

[Murray 2008b] L. Murray, H. Cancela and G. Rubino. *Splitting in Source-Terminal Network Reliability Estimation*. In Proceedings of the 7th International Workshop on Rare Event Simulation, pages 57–68, Rennes, France, September 2008. 6, 34

[Murray 2008c] L. Murray, H. Cancela and G. Rubino. *Splitting in the Simulation of the Network Creation Process*. Technical Report INCO 08-21, Facultad de Ingeniería, Universidad de la República, Montevideo, Uruguay, 2008. ISSN: 0797-6410. 6, 34

[Murray 2010] L. Murray, H. Cancela and G. Rubino. *Network Reliability Evaluation by Splitting a Link Creation Process*. In Proceedings of the ALIO-INFORMS Joint International Meeting, Buenos Aires, Argentina, June 2010. 6, 34

[Murray 2012] L. Murray, H. Cancela and G. Rubino. *On Conditional Monte Carlo in Rare Event Probability Estimation*. In Proceedings of the 9th International Workshop on Rare Event Simulation, pages 57–68, Trondheim, Norway, September 2012. 6

[Murray 2013a] L. Murray. *A conditional Monte Carlo with intermediate estimations for computing MTTF of Markovian systems*. In Proceedings of the Sixth Latin-American Symposium on Dependable Computing (LADC), Rio de Janeiro, Brazil, 2013. 6

[Murray 2013b] Leslie Murray, Héctor Cancela and Gerardo Rubino. *A Splitting algorithm for network reliability estimation*. IIE Transactions, vol. 45, no. 2, pages 177–189, 2013. 6, 34

[Provan 1983] J. S. Provan and M. O. Ball. *The complexity of counting cuts and of computing the probability that a graph is connected*. SIAM Journal on Computing, vol. 12, pages 777–788, 1983. 34

[Ridder 2005] A. Ridder. *Importance Sampling Simulations of Markovian Reliability Systems Using Cross-Entropy*. Annals of Operations Research, vol. 134, pages 119–136, 2005. 15, 16, 74

[Ross 1994] S. M. Ross. *A New Simulation Estimator of System Reliability*. Journal of Applied Mathematics and Stochastic Analysis, vol. 7, no. 3, pages 331–336, 1994. 34

[Ross 2006] S. M. Ross. Simulation, fourth edition. Academic Press, Inc., Orlando, FL, USA, 2006. 62

[Rubino 2009] G. Rubino and B. Tuffin. Rare event simulation using monte carlo methods. Wiley, 2009. 73

[Rubinstein 2004] R. Y. Rubinstein and D. P. Kroese. The cross-entropy method: A unified approach to combinatorial optimization, Monte Carlo simulation and machine learning. Information Science and Statistics. Springer, 2004. 15, 16, 74

[Schruben 1980] L. W. Schruben. *A Coverage Function for Interval Estimators of Simulation response*. Management Science, vol. 26, no. 1, pages 18–27, 1980. 47, 48

[Taboada 2008] H. A. Taboada, J. Espiritu and D. W. Coit. *MOMS–GA: A Multi–objective Multi–State Genetic Algorithm for System Reliability Optimization Design Problems*. IEEE Transactions on Reliability, vol. 57, no. 1, pages 182–191, 2008. 34

[Villén-Altamirano 1991] M. Villén-Altamirano and J. Villén-Altamirano. *RESTART: A Method for Accelerating Rare Events Simulations*. In Proceedings of the 13th International Teletraffic Congress, pages 71–76. North–Holland, 1991. 18, 34, 96

[Villén-Altamirano 2002] M. Villén-Altamirano and J. Villén-Altamirano. *Analysis of restart simulation: Theoretical basis and sensitivity study.* European Transactions on Telecommunications, pages 373–385, 2002. 38

[Villén-Altamirano 2006] M. Villén-Altamirano and J. Villén-Altamirano. *On the efficiency of RESTART for multidimensional state systems*. ACM Transactions on Modeling and Computer Simulation, vol. 16, pages 251–279, July 2006. 38

[Villén-Altamirano 2007] J. Villén-Altamirano. *Importance Functions for RESTART Simulation of Highly–Dependable Systems*. Simulation, vol. 83, pages 821–828, December 2007. 34

[Villén-Altamirano 2010] J. Villén-Altamirano. *RESTART simulation of non–Markov consecutive–k–out–of–n: F repairable systems*. Reliability Engineering & System Safety, vol. 95, no. 3, pages 247–254, 2010. 34