1 Introduction

In computer science more than in the physical sciences, an experimenter must decide what data to gather and analyze, sometimes without the benefit of guidance, experience or easily available intuition. How to obtain general models from experiments or measurements made in a particular environment os by nomeans clear. This chapter discusses the current research in the area of experimental analysis of computer system dependability. The discussion centers around methodologies, major developments and major directions of research.

Experimental evaluation of the dependability of the system can be performed at different phases of the system's life. In the *design phase*, CAD environments are used to evaluate the design via simulation, including simulated fault injection. Such fault injection tests the effectiveness of fault-tolerant mechanisms and evaluates system dependability, providing timely feedback to system designers. Simulation, however, requires accurate input parameters and validation of output results. Although the parameter estimates can be obtained from past measurements, this is often complicated by design and technology changes.

In the *prototype phase*, the system runs under controlled workload conditions. In this stage, controlled physical fault injection is used to evaluate the system behavior under faults, including the detection coverage and recovery capability of various fault tolerance machanisms. Fault injection on the real system can provide information about the failure process, from fault occurrence to system recovery, including error latency, propagation, detection and recovery (which may involve reconfiguration). But this type of fault injection can only study artificial faults; it cannot provide certain important dependability measures, such as mean time between failures (MTBF) and availability.

In the *operational phase*, a direct measurement-based approach can be used to measure systems in the field under real workloads. The collected data contains a large amount of information about naturally occurring errors/failures. Analysis of this data can provide understanding of actual error/failure characteristics and insight into analytical models. Although measurement-based analysis is useful for evaluating the real system, it is limited to detected errors. Further, conditions in the field can vary widely, casting doubt on the statistical validity of the results. Thus, all three approaches — simulated fault injection, physical fault injection, and measurement-based analysis — are required for accurate dependability analysis.

We present an overview of the relevant statistical techniques used in this area. The techniques cover the estimation of parameters and confidence intervals; distribution characterization, including function fitting; and multivariate analysis methods, including clustering analysis, correlation analysis and factor analysis. These techniques can be used in the analysis of data obtained from fault injection or from operational systems.

2 Statistical Techniques

In this section, several statistical techniques are introduced which are commonly used to analyze the data collected from dependability experiments. Ths discussion is not intended to be comprehensive. For a comprehensive study of statistical methods, refer to advanced texts in statistics, like [7] and [2].

2.1 Parameter Estimation

The most important characteristics of a random valiable are its distribution, mean and variance. In practice, means and variances are usually unknown parameters. Thus, how to estimate these parameters from data needs to be addressed.

2.1.1 Point estimation

Point estimation is used in experimental analysis. Examples include the estimation of mean time between failures (MTBF) from field data. Each fault injection and each failure occurrence can be treated as an experiment. In this analysis, each emperiment is assumed to be independent.

Given a collection of n experimental outcomes, $x_1, x_2, \dots x_n$, of a random variable X, each x_i can be considered as a value of a random variable X_i. These X_i's are independent of each other and identical to

X in distribution. The set $\{X_1, X_2, \ldots X_n\}$ is called a *random sample* of X. Our purpose is to estimate the value of some patameter θ (θ can be $E[X]$ or $\mathrm{Var}[X]$) of X using a function of $X_1, X_2, \ldots X_[n]$. The function used to estimate θ, $\tilde{\theta} = \tilde{\theta}(X_1, X_2, \ldots X_n)$, is called an *estimator* of θ, and $\tilde{\theta}(x_1, x_2, \ldots x_n)$ is called a *point estimate* of θ.

An estimator $\tilde{\theta}$ is *unbiased* if $E[\tilde{\theta}] = \theta$. The unbiased estimator that has the minimum variance $\mathrm{Var}(\tilde{\theta}) = E[(\tilde{\theta} - \theta)^2]$ is said to be the *unbiased minimum variance estimator.* It can be shown that the mean of the sample, $\overline{X} = \frac{1}{n} \sum_{i=1}^{n} X_i$ is the unbiased minimum variance estimator of the population mean μ, and the sample variance $S^2 = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \overline{X})^2$ is, under some mild conditions, an unbiased minimum variance estimator of the population variance σ^2. If an estimator $\tilde{\theta}$ converges in probability to θ, that is, $\lim_{n\to\infty} P(|\, \tilde{\theta}(X_1, X_2, \ldots X_n) - \theta \,| \geq \varepsilon) = 0$ for each $\varepsilon > 0$, it is said to be *consistent.*

2.1.2 Method of maximum likelihood

If the functional form of the probability distribution function (p.d.f) of the variable is known, the *method of maximum likelihood* is a good approach to parameter estimation. It is based on the assumption that the observed sample is the most likely to occur among all possible samples. The method usually produces minimum-variance consistent estimators; however, the estimator may be biased if the sample size is too small.

Assuming X has a p.d.f. $f(x|\theta)$, where θ is an unknown parameter, the joint p.d.f. of the sample $\{X_1, X_2, \ldots X_n\}$, $L(\theta) = \prod_{i=1}^{n} f(x_i|\theta)$ is called the *likelihood function* of θ. If $\tilde{\theta}(x_1, x_2, \ldots x_n)$ is the point estimate of θ that maximizes $L(\theta)$, then $\tilde{\theta}(\{X_1, X_2, \ldots X_n\})$ is the *maximum-likelihood estimator* of θ.

In many practical cases, approximate functional forms of the empirical distributions can be obtained, and then the maximum-likelihood estimator can be easily constructed by deriving the likelihood function against the unknown parameters.

2.1.3 Method of moments

Sometimes it is difficult to find maximum-likelihood estimators in closed form. One example is the p.d.f. of the gamma distribution $G(\alpha, \theta), g(x) = \frac{1}{\Gamma(\alpha)\theta^\alpha} \chi^{\alpha-1} e^{-x/\theta}$. The estimation of α and θ is complicated because of the gamma function $\Gamma(\alpha)$. The gamma distribution, however, is useful for characterizing interval times in the real world. E.g. the software time to error for an operating system [9] fits a multistage gamma distribution. In such cases, the *method of moments* can be used if an analytical relationship is found between the moments of the variable and the parameters to be estimated.

To explain the method of moments, the concepts of *sample moment* and *population moment* is introduced. The kth ($k = 1, 2, \ldots$) sample moment of the random variable X is defined as $m_k = \frac{1}{n} \sum_{i=1}^{n} X_i^k$ where $X_1, X_2, \ldots X_n$ are a sample of X. The kth population moment of X is just $E[X^k]$.

Suppose there are k parameters to be estimated. The method of moments is to set the first k sample moments equal to the first k population moments, which are expressed as the unknown parameters, and then solve these k equations for the unknown parameters. The method usually gives simple and consistent estimators, and avoids the difficulties with the p.d.f. of the sample distribution as it uses the expected value that often has simpler functional form.

2.1.4 Interval estimation

So far, only point estimation of unknown parameters was discussed. The estimate may deviate from the actual parameter value. To obtain an estimate with a high confidence, it is necessary to contruct an interval estimate such that the interval includes the actual parameter with a high probability. Given an estimator $\tilde{\theta}$, if $P(\tilde{\theta} - e_1 < \theta < \tilde{\theta} + e_2) = \beta$, the random interval $(\tilde{\theta} - e_1, \tilde{\theta} + e_2)$ is said to be the $100 \times \beta$ *percent confidence interval* for θ, and β is called the *confidence coefficient* (the probability that the confidence interval contains θ).

Confidence Intervals for Means. In the following discussion, the sample mean $\overline{X}$ is used as an estimator for the population mean. As mentioned before, it is the unbiased minimum variance linear estimator for μ. We first consider the case in which the sample size is large. By the central limit theorem, $\overline{X}$ is asymptotically normally distributed, no matter what the polulation distribution is. Thus, when the sample size n is reasonably large (usually 30 or above, sometimes 50 or more if the population is badly skewed with extreme outliners), $Z = (\overline{X} - \mu)/(S/\sqrt{n})$ can be approximately treated as a standard normal variable. To obtain a $100 \times \beta$ percent confidence interval for μ, we can find a number $z_{\alpha/2} from the$N(0,1) distribution table such that $P(Z > z_{\alpha/2}) = \alpha/2$, where $\alpha = 1 - \beta$. Then we have $P(-z_{\alpha/2} < \frac{\overline{X}-\mu}{S/\sqrt{n}} < z_{\alpha/2}) = 1-\alpha$. Thus, the $100(1-\alpha)$ percent confidence interval for μ is approximately $\overline{X} - z_{\alpha/2}\frac{S}{\sqrt{n}} \leq \mu \leq \overline{X} + z_{\alpha/2}\frac{S}{\sqrt{n}}$.

If the sample size is too small (smaller than 30), the above approximation may be poor. In this case, we consider two commonly used distributions: normal and exponential. If the population distribution is normal, the random variable $T = (\overline{X} - \mu)/(S/\sqrt{n})$ has a student t distribution with $n-1$ degrees of freedom. By repeating the same approach performed above with a t distribution table, the $100(1-\alpha)$ percent conficence interval for μ is $\overline{X} - t_{n-1;\alpha/2}\frac{S}{\sqrt{n}} \leq \mu \leq \overline{X} + t_{n-1;\alpha/2}\frac{S}{\sqrt{n}}$ where $t_{n-1;\alpha/2}$ is a number such that $P(T > t_{n-1;\alpha/2}) = \alpha/2$.

If the population distribution is exponential, it can be shown that $\chi^2 = 2n\overline{X}/\mu$ has a chi-square distribution with $2n$ degrees of freedom. Thus, the chi-square distribution table should be used. Because the chi-square distribution is not symmetrical about the origin, we need to find two numbers, $x^2_{2n;1-\alpha/2}$ and $x^2_{2n;\alpha/2}$ such that $P(\chi^2 > x^2_{2n;1-\alpha/2}) = \alpha/2$ and $P(\chi^2 < x^2_{2n;\alpha/2}) = \alpha/2$. The obtained $100(1-\alpha)$ percent confidence interval for μ is $\frac{2n\overline{X}}{x^2_{2n;1-\alpha/2}} < \mu < \frac{2n\overline{X}}{x^2_{2n;\alpha/2}}$.

Confidence Intervals for Variances. The estimation of confidence interval for variances is more complcated, because the sample variance cannot be simply approximated by an unique (such as normal) distribution regardless of the population distribution. However, irrespective to the population distribution, $\lim_{n\to\infty} \mathrm{Var}[S^2] = 0$. Thus, a good confidence interval can be expected as long as the sample size n is large. Next we focus on the two most common distributions: normal and exponential.

If X is normally distributed, the sample variance S^2 can be used to construct the confidence interval. It is known that the random variable $(n-1)S^2/\sigma^2$ has a chi-square distribution with $n-1$ degrees of freedom. To determine a $100(1-\alpha)$ percent confidence interval for σ^2, we follow the procedure for constructing the confidence interval for the mean to find the numbers $x^2_{n-1;1-\alpha/2}$ and $x^2_{n-1;\alpha/2}$. The confidence interval is then given by $\frac{(n-1)S^2}{x^2_{n-1;1-\alpha/2}} < \mu < \frac{(n-1)S^2}{x^2_{n-1;\alpha/2}}$.

If X is exponentially distributed, we can exploit the fact that in this case $\sigma^2 = \mu^2$. Therefore the confidence interval for the mean can be used after taking its square: $\left(\frac{2n\overline{X}}{x^2_{2n;1-\alpha/2}}\right)^2 < \sigma^2 < \left(\frac{2n\overline{X}}{x^2_{2n;\alpha/2}}\right)^2$

Confidence Intevals for Proportions. Often we need to estimate the confidence interval for a proportion or percentage whose underlying distribution is completely unknown. For example, we may want to estimate the confidence interval for the detection coverage after fault injection experiments. In general, given n Bernoulli trials with the probability of success on each trial being p and the number of successes is being Y, how do we find a confidence interval for p?. If n is large (particularly when $mp \geq 5$ and $n(1-p) \geq 5$ [5]), Y/n has an approximately normal distribution, $N(\mu, \sigma^2)$ with $\mu = p$ and $\sigma^2 = p(1-p)/n$. Note that Y/n is the sample mean, which is an estimate for μ or p. Therefore the $100(1-\alpha)$ percent confidence interval, as we have shown earlier, is $fracYn \pm z_{\alpha/2}\sqrt{p(1-p)/n}$.

This result can be used to determine the number of fault injections required to achieve a given confidence interval for an estimated fault detection coverage. Let n represent the number of fault injections and Y the number of fault detected. Assume that all faults have the same detection coverage, approximately p. Now we wish to estimate p with the $100(1-\alpha)$ percent confidence interval being e. From the previous result, $e = z_{\alpha/2}\sqrt{p(1-p)/n}$. Solving this equation for n gives us $n = frac(z_{\alpha/2})^2 p(1-p)e^2$, where n is the number of fault injections required to achieve the desired confidence interval in estimating p.

2.2 Distribution Characterization

Mean and variance are important parameters that summarize data by single numbers. Probability distribution provides more information about data. Analysis of distributions can help one understand the data in detail and arrive at simple conclusions regarding the underlying models. For example, if the time to failure and the recovery time for a system are both exponential, then the model is clearly a Markov model. We now discuss determination of the distribution functions in this subsection.

2.2.1 Empirical Distribution

Given a sample of X, the simplest way to obtan an empirical distribution of X is to plot a histogram of the observations. The range of the sample space is divided into a number of subranges called *buckets*. The lengths of the buckets are usually the same. Assume that we have k buckets, separated by $x_0, x_1, \ldots x_k$, for the given sample of size n. In each bucket, there are y_i instances. Clearly, the sample size n is $\sum_{i=1}^{k} y_i$. Then, y_i/n is an estimation of the probability that X takes a value in bucket i. The histogram is an *empirical probability distribution function* (p.d.f.) of X or the *empirical distribution*. It is trivial to construct the *empirical cumulative distribution function* (c.d.f) from the histogram as well.

The key problem in plotting histograms is determining the bucket size. A small size may lead to such a large variation among thebuckets that the characterization of the distribuition cannot be identified. A large size may lose details of the distribution. Given a data set, it is possible to obtain very different distribution shapes by using different bucket sizes. One rule of thumb is that if any bucket has less than five instances, the bucket size should be increased or a variable bucket size should be used.

2.2.2 Distribution function fitting

Analytical distribution functions are useful in analytical modeling and simulations. Thus, it is often desirable to fit an analytical function to a given empirical distribution. Function fitting relies on knowledge of statistical distribution functions. Given an empirical distribution, step 1 is to make a good guess of the closest distribution function(s) by observing the shape of the empirical distribution. Step 2 is to use a statistical program package to obtain the parameters for a guessed function by trying to fit it to the empirical distribution. Step 3 is to perform a significance test of the goodness-of-fit to see if the fitted function is acceptable. If not, we have to go back to step 1 to try a different function.

Now we discuss step 3 — the significance test. Assume that the given empirical c.d.f. is $F_k(x)$, and the hypothesized c.d.f. is $F(x)$ obtained from step 2. Our task is to test that $F_k(x) \approx F(x)$ within an acceptable tolerance limit. Two commonly used goodness-of-fit test methods are introduced here:

Chi-Square Test. The chi-square test assumes the distribution under consideration can be approximated by a multinomial distribution. Let $p_i = F(x_i) - F(x_{i-1}), i = 1, 2, \ldots, k$ where p_i is the probability that an instance falls into bucket i. If we define $X_1, X_2, \ldots X_k$ in such a way that $P[x_{i-1} \leq X_i < x_i] = p_i, i = 1, 2, \ldots, k$ then $X_1, X_2, \ldots X_k$ have a multinomial distribution that is equivalent to the original distribution $F(x)$. Thus, for a sample size of n, the expected instances falling into bucket i is np_i by the above distribution. The sum of error squares divided by the expected numbers $q_{k-1} = \sum\limits_{i=1}^{k} \frac{(y_i - np_i)^2}{np_i}$ is a measure of the closeness of the observed number of instances, y_i, to the expected number of instances, np_i, in bucket i. If q_{k-1} is small, we tend to accept the fitting of $F(x)$ to $F_k(x)$.

The smallness can be measures in terms of statistical significance if we treat q_{k-1} as a particular value of the random variable Q_{k-1}. It can be shown that if n is large ($np_i \geq 1$), Q_{k-1} has an approximate chi-square distribution with $k-1$ degrees of freedom. Therefore if our hypothesis is true, we expect that q_{k-1} falls onto an acceptable range of Q_{k-1}, so that the event is likely to occur. The boundary value $\chi^2_\alpha(k-1)$ of the acceptable range is chosen such that $P[Q_{k-1} > \chi^2_\alpha(k-1)] = \alpha$ where α is the *significance level* of the test. Its usual value is 0.05 or 0.1. We reject the hypothesis if $q_{k-1} > \chi^2_\alpha(k-1)$.

Kolmogorov-Smirnov Test. The Kolmogorov-Smirnov test is a nonparametric method in that it assumes no particular distribution for the variable in consideration. The method uses the empirical c.d.f., instead of the empirical p.d.f., to perform the test, which is more stringent than the chi-square test. The Kolmogorov-Smirnov statistic is defined as $D_k = sup_x[F_k(x) - F(x)]$ where sup_x represents the least upper bound of all pointwise differences $F_k(x) - F(x)$. In calculation, we can choose the midpoint between x_{i-1} and x_i, for $i = 1, 2, \ldots, k$, to obtain the maximum value of $F_k(x_i) - F(x_i)$. It is seen that D_k is a measure of the closeness of the empirical and hypothesized distribution functions. It can be derived that D_k submits to a distribution whose c.d.f. values are given by the table of Kolmogorov-Smirnov Acceptance Limits [5]. Thus, given a significance level α, we can find the critical value d_k from the table such that $P[D_k > d_k] = \alpha$. The hypothesis is rejected if the calculated value of D_k is smaller than the critical value d_k.

2.2.3 Multivariate Analysis

In reality, measurements are usually made on more than one variable. For example, a computer workload measurement may include usages on the CPU, memory, disk and network. A computer faulure measurement may collect data on multiple components. *Multivariate analysis* is the application of methods that deal with multiple variables. These methods, which include cluster analysis, correlation analysis and factor analysis (to be discussed), identify and quantify simultaneous relationships among multiple variables and are valuable tools for computer systems analysis.

Cluster analysis. *Cluster analysis* is helpful in identifying patterns in data.Specifically, it can help in reading a large number of points plotted in n-dimensional space into a few identifiable states called *clusters.* A common use is in characterizing workload states in computer systems by identifying the points in a resource usage plot that are "similar" by some measure and grouping them into a cluster. Assume we have a sample of p workload variables. We call each instance in the sample a *point* characterized by p values. In this case, the measure of similarity is the Euclidean distance. Let $x_i = (x_{i1}, x_{i2}, \ldots x_{ip})$ denote the ith point of the sample. The Euclidean distance between points i and j, $d_{ij} = |x_i - x_j| = (\sum_{l=1}^{p} (x_{il} - x_{jl})^2)^{1/2}$ is usually used as a similarity measure between points i and j.

There are several different clustering algorithms. The goal of these algorithms is to achieve small *within-cluster* variation relative to the *between-cluster* variation. A commonly used clustering algorithm is the *k-means* algorithm. The algorithm partitions a sample with p dimensions and n points into k clusters, $C_1, C_2, \ldots, C_k$. The mean, or centroid, of C_j is denoted by $\overline{x}_j$. The error component of the portion is defined as $E = \sum_{j=1}^{k} \sum_{x_i \in C_j} |x_i - \overline{x}_j|^2$. The goal of the k-means algorithm is to find a partition that minimizes E.

The clustering procedure is as follows. Start with k groups, each of which consists of a single point. Each new point is added to the group with the closest centroid. After a point is added to a group, the mean of that group is adjusted to take the new point into account. After a partition is formed, search for another partition with smaller E by moving points from one cluster to another cluster until no transfer of a point results in a reduction in E.

A problem associated with clustering algorithms is the presence of outliers in the sample. Outliers can be an order of magnitude greater than any of the other points (which are usually more than 95 percent) of the sample and can be scattered over the sample space. As a result, the generated clusters may not characterize the features of the sample well. For example, most generated clusters contain only one or two outliers, with all other points groupable into only a few clusters. One way to deal with this problem is to specify in the algorithm the minimum number of points that form a cluster, typically 0.5 percent of the sample size. Another way is to define an upper bound for the radius (maximum distance between the centroid and any point in a cluster) of any generated cluster. A recommended range for the upper bound is 1.0 to 1.5 standard deviations of the sample.

Correlation analysis. The correlation coefficient, $Cor(X_1, X_2)$, between the random variables X_1 and X_2 is defined as $Cor(X_1, X_2) = \frac{E[(X_1-\mu_1)(X2-\mu_2)]}{\sigma_1 \sigma_2}$ where μ_1 and μ_2 are the means of X_1 and X_2, and σ_1 and σ_2 are the standard deviations of X_1 and X_2, respectively. If we use ρ to denote the correlation coefficient, then ρ satisfies $-1 \leq \rho \leq 1$. The correlation coefficient is a measure of the linear relationship between two variables. When $|\rho| = 1$, we have $X_1 = aX_2 + b$, where $b > 0$ if $\rho = 1$, or $b < 0$ if $\rho = -1$. In this extreme case, there is an exact linear relationship between X_1 and X_2. When $|\rho| \neq 1$, there is no exact linear relationship between X_1 and X_2. In this case, ρ measures the goodness of the linear relationship $X_1 = aX_2 + b$ between X_1 and X_2. Usually, a ρ value of 0.5 or above is considered reasonably high. *Correlation analysis* can be used to quantify error or workload dependency between two components in a system.

Given random variables, X_1, X_2 and X_3, and correlation coefficients between each pair, ρ_{12}, ρ_{23} and ρ_{13}, we know these variables are related to each other by ρ_{12}, ρ_{23} and ρ_{13}. Since X_1 is related to X_2, and X_2 is related to X_3, a partial dependence between X_1 and X_3 may be due to X_2. The partial correlation coefficient defined below quantifies this partial dependence: $\rho_{13,2} = \frac{\rho_{13}-\rho_{13}\rho_{23}}{\sqrt{(1-\rho_{12}^2)(1-\rho_{23}^2)}}$. The partial correlation coefficient can be considered as a measure of the common relationship among the three variables.

If a random variable, X, is defined on time series, the correlation coefficient can be used to quantify the time serial dependence in the sample data of X. Given a time window $\Delta t > 0$, the *autocorrelation coefficient* of X on the time series t is defined as $Autocor(X, \Delta t) = Cor(X(t), X(t + \Delta t))$ where t is defined on the discrete values $(\Delta t, 2\Delta t, 3\Delta t, \ldots)$. In this case, we treat $X(t)$ and $X(t+\Delta t)$ as two different random variables, and the autocorrelation coefficient is actually the correlation coefficient between the two variables. That is, $Autocor(X, \Delta t)$ measures the time serial correlation of X with a window Δt.

Factor analysis. The limitation of correlation analysis is that the correlation coefficient can only quantify dependency between two variables. However, dependency may exist within a group of two or more variables or even among all variables. The correlation coefficient cannot provide information about this multiple dependency. *Factor analysis* is a statistical technique to quantify multiway dependency among variables. The method attempts to find a set of unobserved common factors that link together the observed variables. Consequently, it provides insight into the underlying structure of the data. For example, in a distributed system, a disk crash can account for failures on those machines whose operations depend on a set of critical data on the disk. The disk state can be considered to be a common factor for failures on these machines.

Let $X = (x_0, x_1, \ldots x_p)$ be a normalized random vector. We say that the k-factor model holds for X if X can be written in the form $X = \Lambda F + E$ where $\Lambda = (\lambda_{ij})(i = 1, 2, \ldots, p; j = 1, 2, \ldots, k)$ is a matrix of constants called *factor loadings*, and $F = (f_1, \ldots, f_k)$ and $E = (e_1, \ldots, e_p)$ are random vectors. The elements of F are called *common factors*, and the elements of E are called *unique factors* (error terms). These factors are unobservable variables. It is assumed that all factors (both common factors and unique factors) are independent of each other and that the common factors are normalized.

Each variable x_i $(i = 1, 2, \ldots, p)$ can then be expressed as $x_i = \sum_{j=1}^{k} \lambda_{ij} f_j + e_i$ and its variance can be written $\sigma_i^2 = \sum_{j=1}^{k} \lambda_{ij} + \psi_i$ as where ψ_i is the variance of e_i. Thus, the variance of x_i can be split into two parts. The first part $h_i^2 = \sum_{j=1}^{k} \lambda_{ij}$ is called the *commonality.* It represents the variance of x_i that is shared with the other variables via the common factors. In particular $\lambda_{ij} = Cor(x_i, f_j)$ represents the extent to which x_1 depends on the jth common factor. The second part, ψ_i, is called the *unique variance.* It is due to the unique factor e_i and explains the variability in x_i not shared with the other variables.

2.2.4 Importance Sampling

Importance sampling has long been used to reduce sampling size while keeping estimates obtained from the sample at a high level of confidence [6]. Specifically, it allows the analysis of rare events (such as failures or errors) with a high degree of accuracy when Monte Carlo techniques are used for numerical solutions. The method has been recently used to reduce the number of runs in Monte Carlo simulations for evaluating computer dependability [3],[1]. In the following, we first give an overview of the method and then discuss its applications in the Monte Carlo simulation of discrete-time Markov chains (DTMCs).

Overview of the method. Assume that a random variable X has p.d.f. $f(x)$ and that $Y = h(X)$ is a function of X. Our goal is to estimate the expected value of Y, $\theta = E[Y] = E[h(X)] = \int_{-\infty}^{\infty} h(x)f(x)dx$ through sampling.That is,we generate a sample $\{x_0, x_1, \ldots x_n\}$ according to $f(x)$, therefore generating $\{y_0, y_1, \ldots y_n\}$.We then calculate $\tilde{\theta} = \overline{Y} = \frac{1}{n} \sum_{i=1}^{n} y_i = \frac{1}{n} \sum_{i=1}^{n} h(x_i)$.

It may be very expensive to generate a statistically significant sample of X. For example,if $y_i =$

$h(x_i) = 0$ for most generated x_i, we may need an extremely large size of sample to estimate θ with a high level of confidence. However,if we can, in sampling,select the rare x_i's that are considered important for estimating θ much more frequently while keeping the estimate unbiased,the sample size will be greatly reduced. This is the essence of the importance sampling method.

To do importance sampling, we change the p.d.f. of X from $f(x)$ to $g(x)$ such that the x's of importance in our parameter estimation have higher occurrence probabilities in g(x). We use X' to represent the variable that has p.d.f. g(x').By the definition of θ, we have $\theta = \int_{-\infty}^{\infty} h(x)f(x)dx = \int_{-\infty}^{\infty} h(x)\frac{f(x)}{g(x)}g(x)dx = \int_{-\infty}^{\infty} h(x)\Lambda(x)g(x)dx$ where $\Lambda(x) = \frac{f(x)}{g(x)}$ is called the likelihood ratio. Let $Y' = h(X)\Lambda(X)$. Then we have $\theta = \int_{-\infty}^{\infty} y'g(x)dx = E[Y']$.

Thus, instead of sampling from $f(x)$ to estimate the expected value of Y, the experiment is changed to sampling from $g(x')$ to estimate the expected value of Y'. That is, we generate a sample $\{x_0, x_1, \ldots x_n\}$ according to $g(x')$, therefore generating $\{y'_1, y'_2, \ldots, y'_n\}$, and then calculate $\tilde{\theta} = \frac{1}{n}\sum_{i=1}^{n} y'_i = \frac{1}{n}\sum_{i=1}^{n} h(x'_i)\Lambda(x'_i)$.

The variance of the above estimator is $Var(Y') = E[(Y'-\theta)^2] = \int_{-\infty}^{\infty} \frac{h(x)f(x)^2}{g(x)}g(x)dx - \theta^2$. To achieve the minimum variance, we should have $g(x) = \frac{h(x)f(x)}{\theta}$. But θ is the unknown parameter to estimate. A heuristic is that the shape of $g(x)$ should follow the shape of $h(x)f(x)$ as closely as possible.

Applications in DTMC simulation. In many cases, the operation of a computer system can be modeled by a discrete-time Markov chain (DTMC) [10]. If the built DTMC is very large (such that it exceeds the available storage) or the functional simulation (simulation of the execution of machine instructions, algorithms, etc.) is used above a DTMC, the Monte Carlo simulation method is perhaps the only feasible way to solve the model. In dependability models, system failures are usually rare events with extremely small probabilities. To obtain statistically significant results, large simulation runs are required, which can be very time consuming. In such a case, importance sampling can reduce simulation runs, usually by orders of magnitude.

Assume we have a DTMC $\{Y_s, s \geq 0\}$ with a set of states $\{S_1, S_2, \ldots, S_m\}$ and a transition matrix $[p_{ij}]$. For each simulation run, we have a path $x_i = S_{i0}, S_{i1}, \ldots, S_{ik}$. The occurrence probability of path x_i [4] is $P(x_i) = p_{i0}p_{i_0 i_1}\cdots p_{i_{k-1}i_k}$ where each p_{ij} is an element in $[p_{ij}]$. All possible paths constitute the probability space of a random variable $X = \{x_1, x_2, x_3, \ldots\}$.

To reduce simulation runs,we change the transition probability matrix from $[p_{ij}]$ to $[p'_{ij}]$ such that the paths that are important in our dependability evaluation are more likely to be sampled. After the change, the occurrence probability of path x_i is $P'(x_i) = p'_{i0}p'_{i_0 i_1}\cdots p'_{i_{k-1}i_k}$.

Assume the dependability measure to evaluate is $\theta = E[h(X)]$. Then θ can be estimated using a sample, $\{x_0, x_1, \ldots x_n\}$, obtained from simulations by $\tilde{\theta} = \frac{1}{n}\sum_{i=1}^{n} h(x_i)\Lambda(x_i)$ where $\Lambda(x_i) = \frac{P(x_i)}{P'(x_i)} = \frac{p_{i0}p_{i_0 i_1}\cdots p_{i_{k-1}i_k}}{p'_{i0}p'_{i_0 i_1}\cdots p'_{i_{k-1}i_k}}$.

The remaining question is how to determine $[p'_{ij}]$. Several heuristics categorized as *failure biasing* have been proposed in the literature [8],[4]. Here we introduce one of the commonly used heuristics. Assume that in state S_i, transitions out of the state go either to a set of failure states, F (e.g. the system suffers one more component failure), or to a set of recovery states, R (e.g. the system recovers from a component failure). (S_i itself can be treated as either in F or in R.) It is obvious that we have $\sum_{j\in F} p_{ij} + \sum_{j\in R} p_{ij} = 1$.

Define a parameter b such that p'_{ij}'s satisfy $\sum_{j\in F} p'_{ij} = b$, $\sum_{j\in F} p'_{ij} = 1-b$. Then we determine each p'_{ij} in state S_i by

$$p'_{ij} = \begin{cases} b\frac{p_{ij}}{\sum_{k\in F} p_{ik}} & j \in F \\ (1-b)\frac{p_{ij}}{\sum_{k\in R} p_{ik}} & j \in R \end{cases}$$

The parameter b is usually chosen to be 0.5 [4]. Since the sum of the original probabilities to failure states is often very small, then the selection of b can significantly increase these probabilities, making these transitions much more likely to occur in simulations.

References

[1] A. E. Conway and A. Goyal: *Monte Carlo Simulation of Computer System Availability/Reliability Models*, Proceedings of the Seventeenth Symposium on Fault-Tolerant Computing, Pittsburgh, Pennsylvania, 1987, pp. 230-235.

[2] W. R. Dillon and M. Goldstein: *Multivariate Analysis*, J. Wiley & Sons, New York, 1984.

[3] A. Goyal, P. Heidelberger and P. Shahabuddin: *Measure Specific Dynamic Importance Sampling for Availability Simulations*, Proceedings of the 1987 Winter Simulation Conference, pp. 351-357.

[4] A. Goyal, P. Shahabuddin, P. Heidelberger, V.F. Nicola and P. W. Glynn: *A unified framework for simulating Markovian models of highly reliable systems*, IEEE Transactions on Computers C-41, 1992, pp. 36-51.

[5] R. V. Hogg and E. A. Tanis: *Probability and Statistical Inference*, 2nd. ed., Macmillan, New York, 1993.

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

[7] M. G. Kendall: *The Advanced Theory of Statistics*, Oxford University Press, 1977.

[8] E. E. Lewis and F. Bohm: *Monte Carlo simulation of Markov unreliability models*, Nuclear Engineering and Design, No. 77, 1986, pp. 49-62.

[9] D. Tang and R. K. Iyer: *Dependability Measurement and Modeling of a Multicomputer System*, IEEE Trans. on Computers, vol.42, no. 1 (Jan 1993), pp. 62-75.

[10] K. S. Trivedi, A. Goyal, W. C. Carter, E. de Souza e Silva and S. S. Lavenberg: *The System Availability Estimator*, Proceedings of the Sixteenth International Symposium on Fault-Tolerant Computing, IEEE Computer Society Press, 1986, pp. 84-89.