ACCURATE RELIABILITY PREDICTION BASED ON SOFTWARE STRUCTURE

Swapna S. Gokhale  
Dept. of Computer Science and Engineering  
Univ. of Connecticut  
Storrs, CT 06269  
Email: ssg@engr.uconn.edu  
Phone: (860) 486-2772

ABSTRACT

Software reliability growth models (SRGMs) are inadequate to assess the reliability of modern, heterogeneous, component–based software systems since these models treat the system as a black box and model its input/output behavior without looking into its internal structure. Development of techniques to assess the reliability of a component–based software system (which may be assembled from a variety of components, some picked off–the–shelf, some developed in–house and some developed contractually), based on its structure is thus absolutely essential. Most of the prior efforts in the area of structure–based reliability assessment use the composite solution approach to predict the expected reliability of the system. While the composite approach produces an accurate reliability estimate, it does not explicitly relate system reliability to the reliabilities of the individual components and system structure. The hierarchical approach, on the other hand, produces a mathematical expression relating the system reliability to the reliabilities of its individual components and system structure. Such an expression facilitates sensitivity analysis, exploration of alternatives through optimization and identification of reliability bottlenecks. However, the reliability estimate obtained using the hierarchical approach is only an approximation of the one obtained using the composite approach. In this paper we develop an accurate hierarchical method to estimate the reliability of a software system based on its structure. The method incorporates second–order structural statistics, and hence provides an estimate that is closer to the one produced by the composite approach. Due to the improved accuracy in the reliability estimate afforded by our method, the hierarchical approach may be used with greater confidence for other purposes. We illustrate the use of our method with a case study.

1 Introduction

Structure–based software reliability assessment techniques are gaining increasing attention with the advent of component–based software development paradigm. These techniques are well suited to assess the reliability of modern software systems than the traditional software reliability growth models [1] due to a variety of reasons. These techniques enable us to: (i) relate system reliability to its structure and the individual component reliabilities, (ii) analyze the sensitivity of system reliability to the reliabilities of its components, (iii) explore alternatives to optimize various system parameters such as performance, reliability and cost, (iv) identify reliability bottlenecks, (v) assess system reliability earlier in the life cycle where maximum latitude exists to take corrective action if the system reliability does not meet the desired expectations, and (vi) assess the reliability of operational systems to identify components which provide maximum potential for reliability improvement.

Predominant research in the area of structure–based software reliability assessment was focused in the development of state–space models for reliability prediction [2, 3, 4, 5, 6]. The state–space models represent the structure of the software system as a control flow graph and estimate reliability analytically. The structure of the software system may be represented by a discrete time Markov chain (DTMC) [2, 7, 8], continuous time Markov chain (CTMC) [6], or a semi–Markov process [5]. The failure behavior of the individual components may be represented by their reliabilities [2], constant failure rates [6], or time–dependent failure rates. There are two approaches to solve the state–space models to predict the reliability of the system. In the “composite approach”, the state–space model representing the structure of the software system is combined with the failure model of its components into a single model. This combined model is then solved to predict system reliability. In the “hierarchical approach”, the assessment of system reliability is performed in two steps. In the first step, the state–space model is solved to obtain structural statistics of the system. These structural statistics include expected number of visits to each state, variance of the number of visits to each state, and the expected time spent in each state during a single run of the system. In the

KEY WORDS

Software structure, Software reliability
second step, the structural statistics are combined with the failure model of the components to predict the reliability of the system. For a given structural model and failure model, the composite method provides an accurate reliability estimate. On the other hand, the hierarchical method provides a reliability estimate which is an approximation of the estimate produced by the composite approach. However, despite the fact that the hierarchical approach produces only approximate predictions, it may be preferred over the composite approach, since the hierarchical approach also generates an analytical expression which relates the overall system reliability to the reliability of its individual components and structural statistics. Based on this analytical expression one can then:

- Analyze the sensitivity of system reliability to the reliabilities of its components.
- Analyze the sensitivity of system reliability to the structural statistics of the system.
- Rank the components of the system in the order of their importance from the system reliability perspective, and thus identify the components that are critical to the system. Additional resources may be devoted towards the reliability improvement of these components, so that a maximum improvement in the system reliability may be achieved for a given amount of resources.
- Determine the allocation of reliability to individual components so that the overall reliability target of the system is achieved.

Since the hierarchical approach provides only an approximate reliability estimate, reliability assessment using the hierarchical approach may lead us to believe that the system reliability is worse than it actually is. This may result in the dedication of more resources towards the improvement of system reliability than might be actually necessary, which may translate into budget and schedule overruns. Also identifying critical components based on an approximate reliability estimate may lead to incorrect conclusions (See Section 4 for an example). If decisions regarding the allocation of resources are based on these incorrect conclusions, it may result in the allocation of resources to components whose reliability improvement may not translate into maximum improvement in system reliability. Improving the accuracy of the reliability estimate obtained using the hierarchical approach is necessary, so that the other advantages offered by the hierarchical approach may be exploited with greater confidence, and is the focus of this paper.

In this paper, we present an accurate hierarchical method to predict system reliability based on its structure. In its present form, the hierarchical method produces an approximate reliability estimate because it ignores higher order structural statistics. The method developed in this paper includes second–order structural statistics, and hence provides a more accurate prediction. We assume that the structure of the system is modeled using a discrete time Markov chain (DTMC). To model the failure behavior of the individual components, we consider two possibilities, namely, probability of failure or reliability, and constant failure rate. The failure behavior of the components may be modeled using reliability or constant failure rate during the operational phase, when the detected faults are usually deferred to the next release (unless they are show stoppers). The methodology is based on the Taylor series expressions to determine the mean and the variance of the function of a random variable.

The layout of the paper is as follows: Section 2 presents Taylor series expressions to obtain the mean and the variance of the function of a random variable. Section 3 describes the accurate reliability prediction method. Section 4 compares the reliability estimate obtained using the accurate method with the reliability estimate obtained using the approximate method using a case study. Section 5 presents conclusions and directions for future research.

### 2 Taylor series expressions

In this section, we present expressions for the mean and variance of a function \( H(X) \) of a random variable \( X \), in terms of mean \( E[X] \), and variance \( Var[X] \) of \( X \). The interested reader is referred to [9] for a detailed derivation. The Taylor series approximation of \( H(X) \) about the mean \( E[X] \) is given by:

\[
H(x) = H(E[X]) + H’(E[X])(x - E[X]) + \frac{1}{2}H’’(E[X])(x - E[X])^2
\]  

(1)

Equation (1) suggests the following expressions for the mean and the variance of \( H(X) \):

\[
E[H(X)] = H(E[X]) + \frac{1}{2}H’’(E[X])Var[X]
\]  

(2)

\[
Var[H(X)] = [H’(E[X])]^2 Var[X]
\]  

(3)

### 3 Accurate reliability prediction method

In this section we describe the method for obtaining an accurate reliability estimate for a system whose structure is represented by an absorbing DTMC \(^1\). We consider two possibilities to represent the failure behavior of the components, namely, the probability of failure (or reliability), and constant failure rate. We assume that the system consists of \( n \) components, and has a single initial state denoted by 1, and a single absorbing or exit state denoted by \( n \). The

\(^1\) The structure of a terminating system is modeled by an absorbing DTMC. A terminating system is the one that operates on demand, and a single software run that corresponds to a terminating execution can be clearly identified.
structure of the system is given by the one–step transition probability matrix $P$ of the DTMC. The $(i, j)^{th}$ entry of matrix $P$ denotes the probability that the control is transferred to component $j$ upon the execution of component $i$. We let $R_i$ denote the reliability of component $i$, $\lambda_i$ denote the failure rate of component $i$, and $\tau_i$ denote the time spent per visit in component $i$.

During a single execution, the reliability of the system, denoted by the random variable $R$ is given by:

$$R = \prod_{i=1}^{n} R_i \times_{1,i}$$  \hspace{1cm} (4)

where $X_{1,i}$ denotes the number of visits to the transient state $i$ starting from state 1. The number of visits to the absorbing state $n$ is always 1, i.e., $X_{1,n} = 1$. We emphasize that since the number of visits to each component is a random variable (in general, except for the last component in the execution sequence), $R$ itself is a random variable.

The expected reliability of the system is given by:

$$E[R] = E\left[ \prod_{i=1}^{n} R_i \times_{1,i} \right] = \prod_{i=1}^{n} E[R_i \times_{1,i}]$$  \hspace{1cm} (5)

Thus to obtain the expected reliability of the system, we need to obtain $E[R_i \times_{1,i}]$, which is the expected reliability of component $i$ for a single run of the system.

We develop expressions to determine the expected reliability of a component during a single run of the system when the failure behavior of the components is represented by their reliabilities and constant failure rates.

### 3.1 Representing failure behavior by reliability

When the failure behavior of the components is represented by their reliabilities, from the Taylor series expression for the expected value of the function of a random variable (see Section 2), we have:

$$E[R_i \times_{1,i}] = R_i E[X_{1,i}] + \frac{1}{2}(R_i E[X_{1,i}])^2 Var[X_{1,i}]$$  \hspace{1cm} (6)

If we let $E[X_{1,i}] = m_{1,i}$ and $Var[X_{1,i}] = \sigma_{1,i}^2$, Equation (6) may be written as:

$$E[R_i \times_{1,i}] = R_i^{m_{1,i}} + \frac{1}{2}(R_i^{m_{1,i}})(\log R_i)^2 \sigma_{1,i}^2$$  \hspace{1cm} (7)

$m_{1,i}$ is the expected number of visits to state $i$ and $\sigma_{1,i}^2$ is the variance of the number of visits to state $i$. $m_{1,i}$ and $\sigma_{1,i}^2$ can be obtained from DTMC analysis [10, 11]. Recall, that DTMC represents the structural model of the software system.

Equation (5) can thus be written as:

$$E[R] = \prod_{i=1}^{n} \left( R_i^{m_{1,i}} + \frac{1}{2}(R_i^{m_{1,i}})(\log R_i)^2 \sigma_{1,i}^2 \right)$$  \hspace{1cm} (8)

Equation (8) incorporates the impact of second–order structural statistics which are captured by the variance of the number of visits to a component on the overall reliability of the system. Capturing of the second–order structural statistics will provide a more accurate estimate of system reliability that is closer to the estimate provided by a composite model. It should be noted that the only source of approximation in the model is the result of the Taylor series cut-off.

If we ignore the second-order structural statistics, as represented by the variance of the number of visits to a module, the expected reliability of component $i$ can be given by:

$$E[R_i^{X_{1,i}}] \approx R_i^{m_{1,i}}$$  \hspace{1cm} (9)

Hence the expected reliability of an system with $n$ components is given by:

$$E[R] \approx \prod_{i=1}^{n} R_i^{m_{1,i}}$$  \hspace{1cm} (10)

The component with the lowest value of $E[R_i^{X_{1,i}}]$ is the reliability bottleneck.

### 3.2 Representing failure behavior by constant failure rate

When the failure behavior of the components is represented by constant failure rates, the reliability of component $i$ for a single visit is given by:

$$R_i = e^{-\lambda_i \tau_i}$$  \hspace{1cm} (11)

The reliability of a component during a typical run is given by:

$$R_i^{X_{1,i}} = e^{-\lambda_i \tau_i X_{1,i}}$$  \hspace{1cm} (12)

In Equation (12), $\tau_i X_{1,i}$ represents the time spent in each component during a typical run of the system. Note that since $X_{1,i}$ is a random variable, the time spent in each component during a typical run of the system will be a random variable. Let $Y_i$ denote the random variable representing the time spent in component $i$ during a typical run of the system. Then $Y_i = \tau_i X_{1,i}$. The expected value of $Y_i$ and the variance of $Y_i$ are given by [11]:

$$E[Y_i] = E[\tau_i X_{1,i}] = \tau_i m_{1,i}$$  \hspace{1cm} (13)
The expected reliability of a component during a typical run is given by:

\[ E[R_i^{X_{1,i}}] = E[e^{-\lambda_i Y_i}] \]  

(15)

Based on Taylor series expressions for the mean of the function of a random variable (see Section 2) Equation (16) can be written as:

\[ E[e^{-\lambda_i Y_i}] = e^{-\lambda_i \tau_i m_{1,i}} + \frac{1}{2} \lambda_i^2 e^{-\lambda_i \tau_i m_{1,i}} \sigma_{1,i}^2 \tau_i \]  

(16)

Using Equation (16), Equation (5) can be written as:

\[ E[R] = \prod_{i=1}^{n} e^{-\lambda_i \tau_i m_{1,i}} + \frac{1}{2} \lambda_i^2 e^{-\lambda_i \tau_i m_{1,i}} \sigma_{1,i}^2 \tau_i \]  

(17)

Without incorporating the second order statistics, the expected reliability of the component would be given by:

\[ E[R_i^{X_{1,i}}] \approx e^{-\lambda_i \tau_i m_{1,i}} \]  

(18)

Thus, the expected reliability of the system without taking into consideration the second order statistics as captured by the variance in the number of visits is given by:

\[ E[R] \approx \prod_{i=1}^{n} e^{-\lambda_i \tau_i m_{1,i}} \]  

(19)

The component with the lowest value of \( E[e^{-\lambda_i Y_i}] \) is the reliability bottleneck.

We would like to note that when the structure of the system is represented by a discrete time Markov chain, and the failure behavior is represented by a constant failure rate, the composite method of solution is not analytically tractable. Hierarchical method of solution is the only feasible alternative in this case. As a result, improving the accuracy of the reliability estimate obtained from the hierarchical approach is even more important since the composite method is not available even when high accuracy predictions may be desired.

4 Illustrations

In this section, we illustrate the accurate reliability prediction method described in Section 3 using the example system shown in Figure 1. The system has 10 components and the structure of the system is represented by an absorbing DTMC shown in Figure 1. State 1 is the initial state of the DTMC, and state 10 is the exit or the absorbing state of the DTMC. For each component, its reliability, failure rate and the time spent per visit is known is summarized in Table 2.

The intercomponent transition probabilities for the system shown in Figure 1 are given in Table 1. Table 1 lists the non-zero entries of the transition probability matrix \( P \) for the system shown in Figure 1.

The mean \( m_{1,i} \), variance \( \sigma_{1,i}^2 \), \( E[R_i^{X_{1,i}}] \) (using Equation (7) and Equation (9)), for component \( i \) are summarized in Table 3. Using the accurate hierarchical method, the expected reliability of the system using Equation (8), is 0.4840. Using the approximate hierarchical approach, the expected reliability of the system computed using Equation (10) is 0.4721.

When the failure behavior of the components is represented by their reliabilities, the reliability of the system can also be obtained using a composite model shown in Figure 2. In Figure 2, states C and F correspond to the successful completion and abnormal termination of the system respectively. Analysis of the model shown in Figure 2,
Table 2. $R_i$, $\tau_i$ and $\lambda_i$ for each component

<table>
<thead>
<tr>
<th>Module #</th>
<th>$(R_i)$</th>
<th>$(\tau_i)$</th>
<th>$\lambda_i$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.9500</td>
<td>1.00</td>
<td>0.0523</td>
</tr>
<tr>
<td>2</td>
<td>0.9000</td>
<td>2.00</td>
<td>0.0537</td>
</tr>
<tr>
<td>3</td>
<td>0.8554</td>
<td>3.00</td>
<td>0.0531</td>
</tr>
<tr>
<td>4</td>
<td>0.9000</td>
<td>3.00</td>
<td>0.0361</td>
</tr>
<tr>
<td>5</td>
<td>0.9000</td>
<td>2.00</td>
<td>0.0537</td>
</tr>
<tr>
<td>6</td>
<td>0.9000</td>
<td>1.00</td>
<td>0.1064</td>
</tr>
<tr>
<td>7</td>
<td>0.9000</td>
<td>2.00</td>
<td>0.0537</td>
</tr>
<tr>
<td>8</td>
<td>0.9000</td>
<td>3.00</td>
<td>0.0361</td>
</tr>
<tr>
<td>9</td>
<td>0.9000</td>
<td>1.00</td>
<td>0.1064</td>
</tr>
<tr>
<td>10</td>
<td>0.9000</td>
<td>1.00</td>
<td>0.0523</td>
</tr>
</tbody>
</table>

Figure 2. Composite model of the system in Figure 1

gives the reliability of the system to be 0.5000.

From Table 3 it can be seen that based on the expected reliabilities computed for each component using Equation (9) (without taking into consideration the second-order structural statistics), components 3 and 5 have the lowest reliability, and would be identified as reliability bottlenecks. However, taking into consideration the second-order structural statistics as in the case of Equation (7), component 5 has lower reliability than component 3, and hence is the only reliability bottleneck. Though the difference in the reliabilities of components 3 and 5 after incorporating the effects of variance is not significant in this case, this example nevertheless highlights the fact that ignoring this effect can lead to misleading conclusions. Such mispredictions can be very costly, since this would imply expending resources on the reliability improvement of the components where the payoff is unlikely to be significant.

When the failure behavior of the components is modeled by a constant failure rates given in Table 2, the expected reliability of the system using the accurate hierarchical approach is 0.4772, and the expected reliability of the system using the approximate approach is 0.4650. As noted in Section 3.2, composite approach is simply infeasible in the case of this combination of structural and failure model. As a result, the accurate hierarchical approach would be additionally valuable in this case. The expected reliability for each component (similar to Table 3) may be computed in this case using Equations (16) and (18). However, we do not include this data due to space limitations.

Thus we see that the reliability prediction method described in this paper provides an estimate of the system reliability that is closer to the reliability estimate produced using the composite method. It is additionally important in the case of models where the composite solution approach is not feasible.

5 Conclusions and future research

In this paper we have developed an accurate hierarchical method for the reliability prediction of a software system based on its structure. The structure of the system was represented by a discrete time Markov chain (DTMC). The failure behavior of the components was represented using two possibilities, namely, the probability of failure or reliability, and constant failure rate. The method can also be used to identify reliability bottlenecks. We have illustrated the accurate reliability prediction method using a case study.

The method described in this paper relies on the assumption that the structure of the system, the time spent in each component per visit, the reliability and the failure rate of each component are known. However, more often than not, this information is not readily available, and needs to be extracted from some other sources. Our current research focuses on the development and validation of techniques to obtain the information for structure-based analysis from a variety of sources.

References


Table 3. \( m_{i,j}, \sigma_{i,j}^2, \) and \( E[R_i^{X_{1,i}}] \)

<table>
<thead>
<tr>
<th>Module #</th>
<th>( m_{1,i} )</th>
<th>( \sigma_{1,i}^2 )</th>
<th>( E[R_i^{X_{1,i}}] )</th>
<th>( E[R_i^{X_{1,i}}] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.0000</td>
<td>0.0000</td>
<td>0.9500</td>
<td>0.9500</td>
</tr>
<tr>
<td>2</td>
<td>0.9077</td>
<td>0.6444</td>
<td>0.9120</td>
<td>0.9088</td>
</tr>
<tr>
<td>3</td>
<td>0.9107</td>
<td>0.5499</td>
<td>0.8732</td>
<td>0.8674</td>
</tr>
<tr>
<td>4</td>
<td>0.4184</td>
<td>0.3928</td>
<td>0.9590</td>
<td>0.9569</td>
</tr>
<tr>
<td>5</td>
<td>1.3504</td>
<td>0.7185</td>
<td>0.8708</td>
<td>0.8674</td>
</tr>
<tr>
<td>6</td>
<td>0.2510</td>
<td>0.2319</td>
<td>0.9751</td>
<td>0.9739</td>
</tr>
<tr>
<td>7</td>
<td>0.6155</td>
<td>0.6261</td>
<td>0.9405</td>
<td>0.9372</td>
</tr>
<tr>
<td>8</td>
<td>0.8737</td>
<td>0.4225</td>
<td>0.9142</td>
<td>0.9121</td>
</tr>
<tr>
<td>9</td>
<td>0.3831</td>
<td>0.2462</td>
<td>0.9968</td>
<td>0.9604</td>
</tr>
<tr>
<td>10</td>
<td>1.0000</td>
<td>0.0000</td>
<td>0.9500</td>
<td>0.9500</td>
</tr>
</tbody>
</table>


