Journal of the Chinese Statistical Association
Vol. 45, (2007)18–37

ASSESSING SOFTWARE RELIABILITY BY UNREVEALED PROPORTION ESTIMATION IN STRATIFIED SAMPLING

Mark C. K. Yang[1], Anne Chao[2] and Y. C. Chen[3]
[1]**Department of Statistics, University of Florida**
[2]**Institute of Statistics, National Tsing Hua University**
[3]**Chia-Nan University of Pharmacy and Science**

ABSTRACT

Suppose there is an unknown number of species in an area of interest. A sample is collected and all the animals are identified according to their species. The purpose is to estimate the proportion θ of the animals that belong to undiscovered species. Point and interval estimators for θ are derived when the area is stratified into K strata and observations are taken from each stratum. Use of proportional allocation is shown to be more efficient than simple random sampling. This model fits very well in software reliability estimation under beta testing, i.e., the software faults are detected by complaints from the users. Many difficult situations in software testing and reliability are overcome by this model.

Key words and phrases: undiscovered species; proportional allocation; software reliability; software maintenance; beta testing.
JEL classification: C13, C42, C88.

1. Introduction

Improving software reliability by testing is one of the most important components in software development. It has been estimated that in many projects, the efforts used

in testing can be around 50% of the total developmental cost (Myers, 1976). However, the use of the testing information to predict its future reliability is one of the major challenges in software engineering, because most reliability estimation theory developed for hardware cannot be applied. In typical hardware reliability estimation, a random sample from the hardware devices is collected, tested and their performance is used to make inference on other devices manufactured the same way. But in software, if bugs are found in testing, they will be removed after testing. Thus the released software is no long the same software tested. How do we assess the reliability of the software after debugging?

One estimator, the failure rate estimation based on a randomly selected testing sample from the users' inputs domain (called the operational profile), can be justified by statistical theory. It is the proportion estimation from a simple random sample. However, this method can be used only to estimate the reliability of the current software. Any debugging will make the traditional binomial type estimation invalid. Thus, reliability can be established only when testing is without failure (Miller et al., 1992). This is, of course, quite unlikely.

Most software reliability model uses testing history to predict the reliability, but there is no consensus on the right models for connecting the past testing history to the future reliability. For example, in the Musa's basic execution or Goel-Okumoto model, discussed in Musa, et al. (1987) and Goel and Okumoto (1984), also known as the Jelinski and Moranda model (1972) in statistical literature, assumes that every fault has the same failure rate. This model has also been used by many authors such as Dalal and Mallows (1988) and Yamada et al. (1993). The Littlewood-Verrall model (Littlewood and Verrall 1973, Littlewood 1981) relaxed the equal failure rate assumption by assuming that the fault failure rates are a sample from a gamma prior distribution. Neither the equal failure rate nor the gamma prior model can be justified in practice. Thus, some researchers would rather fit the testing history (cumulative faults encountered versus time) to a simple smooth curve and then extrapolate the fitted curve to the future. Among them are Musa's logarithmic Poisson (LP) model, Weibull curve by Yamada et al. (1993) and many other curves discussed in Chapter 11 of Musa, et al. (1987). Again, the suitability of a curve can be checked only in the fitted portion.

Its extension to the unknown future can be dangerous. It often happens that several curves can fit the existing data equally well, but their extensions to the future disagree greatly. Other models such as time series model can be found in some recent literature, e.g., Singpurwalla and Wilson (1999).

The above discussion indicates that it is difficult, if not impossible, to ascertain a highly reliable software through pre-release testing, although high reliability is a typical requirement for software performance. However, if the software is in use, we can use the information gathered from users to revise the software. Continual revision is a common practice in software maintenance. The information gathered this way is equivalent to use users as testers. It is called operational testing, or beta testing. Beta testing not only has the merit of testing with a very large sample, but also has an operational profile is more realistic than the one used by a laboratory tester. Podgurski et al. (1993) also have noticed the merit of using operational testing to estimate reliability, but their method can be used only to estimate the current reliability, not the reliability after revision.

In beta testing, let Ω denote the operational profile, i.e., Ω contains all the possible inputs to the software. To simplify notation, we use real numbers to represent the elements in Ω. For $x \epsilon \Omega$, let $b(x)$ denote the consequence after x being executed by the software. We define $b(x) = 0$ if the output is correct and $b(x) = j$ if the jth "fault" is encountered. The term "fault" as used here represents a defect component that will cause some type of failure or incorrect output with certain inputs following the definition of Musa, et al. (1987, p. 236). Sometimes "bug" , "defect" or "error" is also used for the term "fault." The definitions can be different, but it is difficult to make them precise (Frankl et al., 1998, §2.2). They are approximations to a similar phenomenon and cause no difference in the mathematical aspect of software reliability. Note that the jth fault cannot be numbered beforehand. It is just a symbol for one fault. If two inputs, x_1 and x_2, encounter the same fault, then $b(x_1) = b(x_2)$. After testing, we may find many faults. There may also be faults not found. It would be impractical and unnecessary to worry all of them, because some of them may cause little trouble to future users. What we need to know is the reliability after all the known faults being removed, i.e., the proportion of x for which $b(x)$ is not known. This

is the same as to estimate the proportion of unrevealed species first studied by Good (1953). This analogy has been noticed by Goudie (1990), Chao et al. (1993), and Yang and Chao (1995). However, their discussion is still limited to the traditional testing such as cleanroom (Mills, et al., 1987) for small sample. The purpose of this paper is to apply this idea to large sample beta testing.

Since each user has his/her own area of interest, it is unrealistic to assume that every user's inputs consist a random sample from the entire operational profile. Thus, the input domain Ω should be partitioned into disjoint strata, also called bins in some software literature. Let there be K strata, denoted by Ω_1, Ω_2, ..., Ω_K. Moreover, we assume that when a user's input is in stratum Ω_k, it is a random sample from Ω_k. Thus, what we need to estimate is the unrevealed proportion in the population after stratified sampling. We will first develop its theory through the traditional notation in statistics literature, i.e., $b(x)$ is the "species" x belongs to.

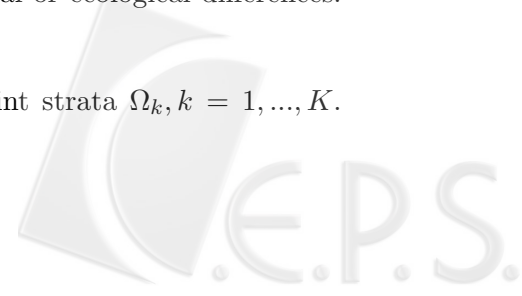
2. Estimating unrevealed proportion with stratified sampling

Without stratification, a simple version of this problem can be demonstrated in wildlife study. Suppose an area is surveyed and m species of insects were identified. What proportion of the insects that belong to some undiscovered species? We define this proportion to be θ. Note that θ is a random variable, not a parameter. Good (1953) showed that S/n is a good estimate of θ, where n is the total number of observed animals and S is the number of species that have been observed only once, i.e, S is the number of singletons.

Note that θ can also be interpreted as the probability that the next observation belongs to a new species. This interpretation is exactly what we mean software reliability, the probability that the next user will encounter an unrevealed fault.

The application in wildlife survey with stratified sampling is also easy to see because there is no reason to take simple random sample when the area of interest can be conveniently divided into strata according to their geological or ecological differences. We start with the estimation of θ.

Let the population Ω can be partitioned into K disjoint strata $\Omega_k, k = 1, ..., K$.



Let N_k be the size of Ω_k and the population size $N = \sum_{k=1}^{K} N_k$. Define the strata proportions as

$$\alpha_k = N_k/N \tag{1}$$

Let there be M (unknown) species in the population and the proportion of species i in stratum k be q_{ki}. Apparently, $\sum_{i=1}^{M} q_{ki} = 1$ for all $k = 1, 2, ..., K$, and the proportion of species i in the whole population is

$$q_i = \sum_{i=1}^{K} \alpha_k q_{ki}. \tag{2}$$

Throughout this paper we assume that all the N_k are large and the α_k's are known. Suppose n_k observations are taken from stratum k and the total sample size is $n = n_1 + n_2 + ... + n_k$. Let

$$X_k(i) = \text{ the number of species } i \text{ discovered in stratum } k. \tag{3}$$

Then the unrevealed proportion is

$$\theta = \sum_{i=1}^{M} q_i I[X_k(i) = 0, \text{ for all } k = 1, ..., K], \tag{4}$$

where $I[A]$ is the indicator function which takes value 1 if A is true and 0 otherwise. Since

$$E[X_k(i) = 0 \text{ for all } k = 1, 2., , , K] = \prod_{l=1}^{K} (1 - q_{lk})^{n_l},$$

we have,

$$E[\theta] = \sum_{k=1}^{K} \alpha_k \sum_{i=1}^{M} q_{ki} \prod_{l=1}^{K} (1 - q_{lk})^{n_l}. \tag{5}$$

To estimate θ in (4), we define all the species that appear only once in all the strata as the inter-stratum singletons. This number is denoted by S. Since a inter-stratum singleton appears only once, it must have occurred in one and only one of the strata. Let S_k denote the number of inter-stratum singletons that appear in stratum k. Apparently, $S = \sum_{k=1}^{K} S_k$. If we let

$$\hat{\theta} = \sum_{k=1}^{K} \frac{\alpha_k S_k}{n_k}, \tag{6}$$

then the expected value of $\hat{\theta}$ becomes

$$\begin{aligned} E[\hat{\theta}] &= \sum_{k=1}^{K} \frac{\alpha_k}{n_k} \sum_{i=1}^{M} \binom{n_k}{1} q_{ki}(1-q_{ki})^{n_k-1} \prod_{l\neq k}^{K} (1-q_{li})^{n_l} \\ &= \sum_{k=1}^{K} \alpha_k \sum_{i=1}^{M} q_{ki} \prod_{l=1}^{K} (1-q_{li})_l^n / (1-q_{ki}) \\ &\approx E[\theta] \end{aligned} \tag{7}$$

The last step of (7) is based on the assumption $(1-q_{ki})^{n_k-1} \approx (1-q_{ki})^{n_k}$, because if q_{ki} is large, it should have been discovered and this i will not appear in the summation of (4). Thus, $\hat{\theta}$ is nearly an unbiased estimator of θ. The mean square error between $\hat{\theta}$ and θ can be shown as follows:

$$\begin{aligned} e^2 \equiv E(\hat{\theta}-\theta)^2 &\approx \sum_{k=1}^{K} \frac{\alpha_k^2}{n_k} \sum_i q_{ki}(1-q_{ki})^{n_k-1} \prod_{l\neq k}^{K} (1-q_{li})^{n_l} \\ &+ \sum_{k=1}^{K} \alpha_k^2 \sum_i q_{ki}^2 \prod_{l=1}^{K} (1-q_{li})^{n_l} \\ &- \sum_{k=1}^{K} \frac{\alpha_k^2}{n_k} \sum\sum_{i<j} q_{ki}q_{kj}(1-q_{ki}-q_{kj})^{n_k-2} \prod_{l\neq k}^{K} (1-q_{li}-q_{lj})^{n_l} \\ &+ 2\sum\sum_{k<k'} \alpha_k\alpha_{k'} \sum_i q_{ki}q_{k'i} \prod_{l=1}^{K} (1-q_{li})^{n_l}. \end{aligned} \tag{8}$$

The justification of (8) is in Appendix A. Though e^2 involves many parameters, it can be estimated by

$$\hat{e}^2 = \sum_{k=1}^{K} \frac{\alpha_k^2}{n_k}\left[\frac{f_1^{(k)}}{n_k} + \frac{2f_2^{(k)}}{n_k} - \left(\frac{f_1^{(k)}}{n_k}\right)\right]^2 + 2\sum\sum_{k<k'} \frac{\alpha_k\alpha_{k'}}{n_k n_{k'}} f_{11}^{kk'}, \text{ where,} \tag{9}$$

$$f_1^{(k)} = \sum_{i=1}^{M} I[\text{species } i \text{ is a inter-stratum singleton and appears in stratum } k] = S_k$$

$$f_2^{(k)} = \sum_{i=1}^{M} I[\text{species } i \text{ is a inter-stratum doubleton and appears in stratum } k]$$

$$f_{11}^{(kk')} = \sum_{i=1}^{M} I[\text{species } i \text{ is a inter-stratum doubleton and appears in stratum } k \text{ and } k']$$

and a doubleton means that a species appeared exactly twice. The justification is again given in Appendix A. Moreover, we have established the asymptotic normality of $\hat{\theta} - \theta$. Thus, a $(1-\alpha)$ 100% confidence interval for θ is approximately

$$\hat{\theta} \pm z_{\alpha/2}\hat{e}, \tag{10}$$

where $z_{\alpha/2}$ is the $\alpha/2$ upper quantile of a standard normal distribution. Since the derivation of the asymptotic normality is quite complex, we show its main steps in Appendix B. To see how the asymptotic results work for practically acceptable sample sizes, a simulation study is presented in Section 4.

3. Sample allocation

Given the total sample size n, how should one allocate the sample sizes n_k in each stratum? Since (8) depends on the unknown parameters q_{ki}, it is unlikely that one can construct the optimal allocation with minimum mean square error. Since α_k's are known, we can use the proportional allocation by letting $n_k = n\alpha_k$. We will show that this allocation produces a smaller mean square error than a simple random sample of the same sample size. Using proportional allocation, (8) becomes

$$\begin{aligned} e_p^2 &\equiv E[\hat{\theta}_p - \theta_p]^2 \\ &= \sum_{i=1}^{M}\prod_{l=1}^{K}(1-q_{li})^{n_l}\{\frac{1}{n}\sum_{k=1}^{K}\alpha_k q_{ki} + (\sum_{k=1}^{K}\alpha_k q_{ki})^2\} \\ &= \sum_{i=1}^{M} q_i \prod_{l=1}^{K}(1-q_{li})^{n_l}\{\frac{1}{n} + q_i\} \end{aligned}$$

For simple random sampling, all the strata are lumped together and the estimate of θ is

$$\hat{\theta}_s = \text{number of singletons}/n \tag{11}$$

and

$$e_s^2 \equiv E[\hat{\theta}_s - \theta_s]^2 = \sum_{i=1}^{M} q_i(1-q_i)^n(\frac{1}{n} + q_i). \tag{12}$$

It can be shown that $e_s^2 \geq e_p^2$ (in Appendix C). Since θ is a random variable, to make the mean square error $E[(\tilde{\theta} - \theta)^2]$ small for any estimator $\tilde{\theta}$ is not the only criterion to judge different sampling schemes. For example, if we do not take any observation, we have $\tilde{\theta} = \theta = 1$. Though $E[(\tilde{\theta} - \theta)^2] = 0$, this is obviously not a good sampling strategy. We also want $E[\theta] \approx E[\tilde{\theta}]$ small. It has been shown, also in Appendix C, that $E[\hat{\theta}_s] \geq E[\hat{\theta}_p]$. Thus, it is fair to conclude that the stratified sampling with proportional allocation is better than a non-stratified random sampling.

4. Simulation

To see how the asymptotic theory works for a sample of reasonable size in beta testing, we have done the following simulation study. One main variable in simulating software testing and debugging process is the fault distribution. Without loss of generality, we may arrange the faults in a monotonic decreasing order according to their failure rates. Five types of decreasing rates, exponential, Zipf's law, constant, random, and Adams IBM data (Adams, 1984), were used by Yang and Chao (1995) to cover large variations of fault distributions. We feel that Adams' data in Table 1 is the closest to most real fault distributions; there are many small faults that are difficult to detect. However, due to the difficulty in estimating each failure rate accurately, the Adams' data set uses a constant failure rate in each category as an approximation. To check the robustness of this approximation, we have replaced them by exponentially decreasing failure rates in the simulation. For example, the last category in Table 1 originally contains 75 faults with equal failure rate $q_i = 3.2 \times 10^{-6}$. They are replaced by $q_i = \alpha \times \lambda^i$ with the middle point $q_{38} = 3.2 \times 10^{-6}$ and the total $\sum_{i=1}^{75} = 75 \times 3.2 \times 10^{-6}$. The results are very similar to the data in the original table. In our simulation, we assume that the operational profile can be divided into four strata as shown in the second row of Table 2. Some faults are assigned into the common portion of the program. The notation $A/2$ in Table 2 means that 1/2 of the faults in Table 1 are assigned to the common operational part of the software. The fourth row shows that strata 1 and 3 share a common portion that contains a quarter $(A/4)$ of the faults in Table 1, and the next row indicates that strata 1, 2 and 4 share another quarter $(A/4)$ of the faults

in Table 1. The same rules apply to the next two rows. Since the numbers of faults in Table 1 are not all even numbers, a random assignment is used when they are split into two. For example, 7 is split into 3 and 4. The same rule applies when the faults are split into four parts.

Since operational testing can absorb a large number of testing cases, $n = 10^4$, 10^5 and 10^6 were used. Many other combinations of the usage proportion $\{\alpha_i, i = 1, ..., 4\}$ were also used, but the results were similar. Only the result for $\alpha_1 = \alpha_2 = \alpha_3 = \alpha_4 = 0.25$ is presented in Table 4. Proportional sampling is assumed, i,e., the reporting from each stratum is proportional to the stratum size. The results are based on 3,000 simulation trials for each n. The results in Table 4 show that the bias $E[\hat{\theta}] - E[\theta]$ is small as expected and the estimated root mean square error (ERMSE, see Table 3 for definition) is close to the true root mean square error (RMSE). The normal approximation works the best if $Z \equiv (\hat{\theta} - \theta)/\sqrt{\hat{e}^2}$ is approximately a standard normal distribution with the mean $(E[Z]) = 0$, variance $(V[Z]) = 1$, skewness $(S[Z]) = 0$ and kurtosis $(K[Z]) = 3$. For small θ or large n, the skewness and kurtosis are not close to their ideal values. This does not contradict the asymptotic normality because in the proof we require that the species proportions decrease as the sample size increases to avoid the no error degenerating case at a large sample (relative to the small proportions). Since the faults are fixed in our simulation, we do not expect the asymptotic normality works for all n, but (10) still provides a good guideline.

5. Comparison with existing methods in software testing

It is difficult to compare the new method with the dynamic model methods discussed in Section 1. The key features in pre-release testing, such as cumulative failures under testing and debugging, do not exist in operational testing. However, we may compare the gain of using (6) with a special case in pre-release testing, i.e., when tests reveal no error. In operational testing, if the faults were removed after the cause for

Table 1 Fault Distribution from One of Adams' Data Sets

Number of faults	Failure rate (q_i)
1	0.01
7	0.0032
16	0.001
13	0.00032
48	0.0001
91	0.000032
82	0.00001
75	0.0000032

Table 2 Faults Partitioning According to an Operational Profile

Operational profile strata			
1	2	3	4
A/2			
A/4		A/4	
A/4	A/4		A/4
	A/4	A/4	
			A/4

Table 3 Definitions and Estimates for the Symbols in Table 4

Notation	Definition	Estimate from simulation
BIAS	$E[\hat{\theta}] - E[\theta]$	$(\sum \hat{\theta}_i - \sum \theta_i)/s^*$
RMSE	$\{E(\hat{\theta} - \theta)^2\}^{1/2}$	$\{\sum(\hat{\theta}_i - \theta_i)^2/s\}^{1/2}$
ERMSE	$\lvert\hat{e}\rvert$ in Formula (9)	sample mean of $\lvert\hat{e}\rvert$
Z	$(\hat{\theta} - \theta)/\sqrt{\hat{e}^2}$	histogram of $(\hat{\theta} - \theta)/\sqrt{\hat{e}^2}$

*The summation is always from $i = 1$ to s, where s = simulation size=3,000. Subscript i is the trial index indicating the value at the ith trial.

Table 4 Performance of using (6) as the θ estimator. Symbols are defined in Table 3.

n	10^4	10^5	10^6
E(θ]	0.0068314	0.000999053	0.000018480
BIAS	-0.0000076	-0.000000934	-0.000000094
RMSE	0.0010305	0.000150136	0.000007899
ERMSE	0.0009610	0.000130626	0.000006158
$E\hat{[}Z]$	-0.0459360	-0.0303118	-0.058330356
$V\hat{[}Z]$	1.1797402	1.3348065	1.733800168
$S\hat{[}Z]$	-0.5662937	-0.3583334	-1.463598791
$K\hat{[}Z]$	4.4661881	5.2901182	11.069177476

Note: Because of a few small numbers, large decimals are used for the table uniformity. Reliability of these numbers can be extended only to two significant digits.

each complaint has been investigated, then there would exist a time T_0 and from this time on, all the additional reports would produce no new faults. During the simulation, we let the failures be reported sequentially and consider from the time T_0 to the time of revision as testing without failure. Miller et al. (1987) provide a survey of methods that can be used to estimate the reliability in the no failure situation. Though there are many reasonable methods, there is no dominant one. The method they advocate is to find the Bayesian posterior distribution of θ with a beta prior distribution. Suppose t tests have been done without failure. Then the Bayesian point estimate of θ is

$$\tilde{\theta} = \frac{a}{t + a + b}, \tag{13}$$

where a and b are the two parameters in the beta prior distribution. When the operational profile is partitioned into K strata, (13) becomes

$$\tilde{\theta} = \sum_{k=1}^{K} \alpha_i \frac{a_i}{t_i + a_i + b_i}$$

where a_i, b_i and t_i are the parameters for the beta prior distribution, and the number of tests without failure in stratum i. Without prior knowledge, the commonly used beta parameters are $a = b = a_i = b_i = 1$. Note that in this case,

$$\tilde{\theta} = \sum_{k=1}^{K} \alpha_i \frac{1}{t_i + 2} \approx \sum_{k=1}^{K} \alpha_i \frac{1}{t_i} \tag{14}$$

How much gain do we have if (6) is used instead of (14)? In other words, how is (8) compared to

$$\epsilon^2 \equiv E(\tilde{\theta} - \theta)^2? \tag{15}$$

The results parallel to Table 4 are given in Table 5, where the symbols are defined as in Table 3 except for replacing $\hat{\theta}$ by $\tilde{\theta}$ and e^2 by ϵ^2. Note E[θ] are the same for both tables because it represent the same true failure rate after all the observed faults are removed. It is obvious that (14) is worse than (8) by a large magnitude. The large bias and mean square error make (14) inefficient in reliability estimation.

Table 5 Performance of using (14) as an estimator for θ. To be compared with Table 4.

n	10^4	10^5	10^6
BIAS	0.029	0.011	0.00114
RMSE	0.042	0.019	0.00337
ERMSE	0.021	0.008	0.00092

6. Concluding remarks

1. Random sample assumption is reasonable if we can partition Ω fine enough. Note that the size of K does not affect our reliability estimation from the derivation in the previous two sections. Two sampling schemes are possible. One is active sampling: The software developer will call some randomly selected users regularly and ask them the intensity of their usage and the problems they have encountered. The only assumption needed here is that the users will cooperate. The other scheme is passive sampling: The software developer will ask the usage information only when a user is seeking help or reporting problems. Since only troubles are reported, passive sampling will give a conservative (lower) reliability estimate if the formulae for the active sampling is used. The theory developed in Section 2 would be best applicable under the active sampling

scheme.

2. A related problem to the estimation of the unrevealed proportion is the estimation of the number of (unrevealed) species. It can be intuitively and theoretically shown that the latter is a very difficult problem (Esty 1983, Bunge and Fitzpatrick 1993), because there can be a large number of extremely rare species that has almost no chance to appear in the sample. Hence, there is no way to ascertain this number with certainty, unless there are further assumptions on the population structure. Some of the assumptions are (1) a lower proportion bound (Stroke, 2003) or (2) beta type of prior (Hass et al., 2006). Nevertheless, the estimate of the unrevealed proportion provides useful information for the number of undiscovered species (Chao and Lee 1992). Software engineers have also worked on find unknown faults in a different dirction. They try to estimate the properties of a module that make it more error-prone (see Ostrand, et al. 2005, and Zhou and Leung, 2006). The statistical theory behind those rule remains to be worked out.

3. A major criticism of software testing is the problem of determining the operational profile, i.e., the profile used by the tester may not be that of the future users. Using the estimate in (6) for θ in (4), one can estimate the reliability for any operational profile change in the future by changing the proportions $\{\alpha_k\}$. The derivations in Section 2 do not require n_k/n to be the same as the future α_k.

4. In many practical situations, new features are added when a software is revised. We may consider this as an operational profile extension from Ω_0 to $\Omega_0 \cup \Omega_1$, where Ω_1 is the input domain for the newly added features. The failure rate of the newly added portion cannot be estimated by the operational testing because it has not been used, but (4) and (6) can be easily adapted to this situation. Let θ be the true failure rate, α' be the usage proportion for the new domain Ω_1, and $\hat{\theta}_1$ be an estimate of θ_1 by any method used in conventional reliability estimation. Then an estimate of the new failure rate,

$$\theta = (1 - \alpha')\theta_0 + \alpha'\theta_1,$$

is to substitute θ_0 and θ_1 by their respective estimates $\hat{\theta}_0$ and $\hat{\theta}_1$. If α' is small, the overall reliability θ can still be estimated accurately.

Appendix A : Justification of (5) - (9)

Let

$$\begin{aligned}\theta_k &= \sum_{i=1}^{M} q_{ki} I[X_k(i) = 0, \forall k] \\ S_k &= \sum_{i=1}^{M} I[X_k(i) = 1 \text{ and } X_l(i) = 0, \forall l \neq k].\end{aligned}$$

The derivations of (5) and (7) are based on the following facts:

$$\begin{aligned}EI[X_k(i) = 0, \forall k] &= \prod_{l=1}^{K} (1 - q_{li})^{n_l} \\ EI[X_k(i) = 1 \text{ and } X_l(i) = 0, \forall l \neq k] &= \binom{n_k}{1} q_{ki} (1 - q_{ki})^{n_k - l} \prod_{l \neq k}^{K} (l - q_{li})^{n_l}\end{aligned}$$

To derive (8), note that

$$\begin{aligned}(\hat{\theta} - \theta)^2 &= \{\sum_{k=1}^{K} \alpha_k (\frac{S_k}{n_k} - \theta_k)\}^2 \\ &= \sum_{k=1}^{K} \alpha_k^2 (\frac{S_k}{n_k} - \theta_k)^2 + 2 \sum \sum_{k<k'} \alpha_k \alpha_{k'} (\frac{S_k}{n_k} - \theta_k)(\frac{S_{k'}}{n_{k'}} - \theta_{k'}) \\ &= \sum_{k=1}^{K} \alpha_k^2 (\frac{S_k^2}{n_k^2} - 2\frac{S_k \theta_k}{n_k} + \theta_k^2) \\ &+ 2 \sum \sum_{k<k'} \alpha_k \alpha_{k'} (\frac{S_k S_{k'}}{n_k n_{k'}} - \frac{\theta_k S_{k'}}{n_{k'}} - \frac{\theta_{k'} S_k}{n_k} + \theta_{k'} \theta_k).\end{aligned}$$

The expected values of all the terms terms can be worked out. For example,

$$E[S_k \theta_k] = \sum_{i=1}^{M} \sum_{j \neq i} \binom{n_k}{1} q_{ki} q_{kj} (1 - q_{ki} - q_{kj})^{n_k - 1} \prod_{l \neq k}^{K} (1 - q_{li} - q_{lj})^{n_l}.$$

After algebraic simplification and the small q approximations like $(1 - q_{ki})^{n_k - 1} \approx (1 - q_{ki})^{n_k}$, we have (8).

To justify (9), we need to find the expected values of the f's in it. It can be easily seen that the first two terms in (9) and the the first two terms in (8) have the same expected values after using the $q_{ij} \approx 0$. The expected values of the third and fourth

terms in (9) can be found by

$$\begin{aligned}
E[f_1^{(k)}]^2 &= E(\sum_{i=1}^{M}\sum_{j=1}^{M} I(X_k(i)=X_k(j)=1; X_l^{n_k}(i)=X_l^{n_k}(j)=0 \forall l\neq k) \\
&= \sum_{i=1}^{M}\binom{n_k}{1} q_{ki}(1-q_{ki})^{(n_k-1)}\prod_{l\neq k}^{K}(1-q_{li})^{n_l} \\
&+ \sum_{i=1}^{M}\sum_{j\neq i}\frac{n_k!}{1!1!(n_k-2)!} q_{ki}q_{kj}(1-q_{ki}-q_{kj})^{n_k-2}\prod_{l\neq k}^{K}(1-q_{li}-q_{lj})^{n_l}. \\
E[f_{11}^{(kk')}] &= \sum_{i=1}^{M} q_{ki}q_{k'j}(1-q_{ki})^{n_k-1}(1-q_{k'i})^{n_{k'}-1}\prod_{l\neq k,k'}(1-q_{li})^{n_l}. \\
&\approx \sum_{i=1}^{M} q_{ki}q_{k'i}\prod_{l=1}^{K}(1-q_{li})^{n_l}.
\end{aligned}$$

Appendix B: Derivation of asymptotic normality of $\hat{\theta}-\theta$

Our derivation follows the ideas of Holst (1979) and Esty (1983). We state only the steps of the proof. The detail can be obtained from any author of this paper.

Lemma 1. Let $(V, U_1, ..., U_k)$ be a $(k+1)$-dimensional random vector, with $U_1, ..., U_K$ integer-valued. Then the conditional characteristic function for V is

$$\begin{aligned}
&E(e^{ivV}|U_1=n_1, ..., U_k=n_k) \\
&= \frac{1}{(2\pi)^K P(U_1=n_1, ..., U_k=n_k)}\int_{-\pi}^{\pi}...\int_{-\pi}^{\pi} Ee^{iu_1(U_1-n_1)+...+iu_k(U_k-n_k)+ivV}du_1...du_K.
\end{aligned}$$

Let the notation $X_k(i), \theta, \hat{\theta}$ remain the same as those in Appendix A. It can also be shown in the following lemma.

Lemma 2. For non-negative integers $\{x_k(i)\}$ with $\sum_{i=0}^{M} x_k(i) = n_k$, $P(X_k(i) = x_k(i); i = 0, ..., M) = P(Y_k(i) = x_k(i); i = 0, ..., M | \sum_{i=0}^{M} Y_k(i) = n_k)$ where $\{Y_k(i)\}$ are independent random variables and $Y_k(i)$ is Poisson with mean $n_k q_{ki}$.

Some regularity conditions are necessary. 1) There should be a balance between the n_i. Asymptotic normality cannot be reached if some of the n_i's remaining small.

Thus, we require that n_i/n_j tends to a non-zero constant for all i and j. 2) We have to avoid degenerating cases. In other words, if all the species can be easily discovered, there will be no estimation error. Thus, we require (Esty 1983, Theorem 1),

$$\frac{Ef_1^{(k)}}{n_k} \to 0 < d_1^{(k)} < 1, \text{ and } \frac{Ef_2^{(k)}}{n_k} \to d_2^{(k)} \geq 0, \forall k = 1, ..., M.$$

Let $f_k(X_1(i), ..., X_K(i)) = q_{ki}$, if $X_k(i) = 0 \forall k$; $= -1/n_k$ if $X_k(i) = 1$ and $X_l(i) = 0, \forall l \neq k$; and $= 0$ otherwise. Then

$$\theta - \hat{\theta} = \sum_{k=1}^{K} \alpha_k \sum_{i=1}^{M} f_k(X_1(i), ..., X_K(i)).$$

Let $V = n_1^{1/2} \sum_{k=1}^{K} \alpha_k \sum_{i=1}^{M} f_k(X_1(i), ..., X_k(i))$ and $\rho_k = (n_1/n_2)^{1/2}$, then by the above two lemmas and Theorem 2 of Esty (1983), we have

$$\begin{aligned}
& Ee^{ivV} \\
\to\ & \frac{1}{(2\pi)^{k/2}} e^{-v^2/2} \{ \sum_{i=1}^{M} \alpha_1^2 \rho_1^2 n_1 q_{1i}^2 \prod_{l=1}^{K} e^{-n_k q_{ki}} + \sum_{i=1}^{M} \alpha_1^2 \rho_1^2 q_{1i}^2 \prod_{l=1}^{K} e^{-n_k q_{ki}} + ... \\
+\ & \{ \sum_{i=1}^{M} \alpha_K^2 \rho_K^2 n_K q_{Ki}^2 \prod_{l=1}^{K} e^{-n_k q_{ki}} + \sum_{i=1}^{M} \alpha_K^2 \rho_K^2 q_{Ki}^2 \prod_{l=1}^{K} e^{-n_k q_{ki}} \\
+\ & \sum_{i=1}^{M} \sum \sum_{1 \leq j \leq l \leq K} 2\alpha_j \alpha_l \rho_j \rho_l q_{ji} q_{li} \sqrt{n_j n_l} \prod_{l=1}^{K} e^{-n_k q_{ki}} \} \\
\cdot\ & \int_{-\infty}^{\infty} exp(-t_1^2/2 + v\alpha_1 \rho_1 t_1) \sum_{i=1}^{M} q_{1i} \prod_{l=1}^{K} e^{-n_k q_{ki}} dt_1 \cdots \\
& \int_{-\infty}^{\infty} exp(-t_K^2/2 + v\alpha_K \rho_K t_K) \sum_{i=1}^{M} q_{Ki} \prod_{l=1}^{K} e^{-n_k q_{ki}} dt_K
\end{aligned}$$

Hence $\sqrt{n_1}(\theta - \hat{\theta})$ has asymptotic variance

$$\begin{aligned}
& \alpha_1^2 \rho_1^2 [\sum_{i=1}^{M} n_1 q_{1i}^2 \prod_{k=1}^{K} e^{-n_k q_{ki}} + \sum_{i=1}^{M} q_{1i} \prod_{k=1}^{K} e^{-n_k q_{ki}} - (\sum_{i=1}^{M} q_{1i} \prod_{k=1}^{K} e^{-n_k q_{ki}})^2] + \cdots \\
=\ & \alpha_K^2 \rho_K^2 [\sum_{i=1}^{M} n_K q_{Ki}^2 \prod_{k=1}^{K} e^{-n_k q_{ki}} + \sum_{i=1}^{M} q_{Ki} \prod_{k=1}^{K} e^{-n_k q_{ki}} - (\sum_{i=1}^{M} q_{Ki} \prod_{k=1}^{K} e^{-n_k q_{ki}})^2] \\
+\ & \sum_{i=1}^{M} [\sum \sum_{1 \leq j \leq l \leq K} 2\alpha_j \alpha_l \rho_j \rho_l q_{ji} q_{li} \sqrt{n_j n_l} \prod_{k=1}^{K} e^{-n_k q_{ki}}].
\end{aligned}$$

The estimator of the variance of $\sqrt{n_1}(\theta - \hat{\theta})$ is

$$\sum_{k=1}^{K} \alpha_k^2 \rho_k^2 [\frac{f_1^{(k)}}{n_k} + \frac{2f_2^{(k)}}{n_k} - (\frac{f_1^{(k)}}{n_k})^2 + 2\sum\sum_{1\le j\le l\le K} \frac{n_1 \alpha_j \alpha_l}{n_j n_l} f_{11}^{(jl)} + O(1),$$

which is equivalent to the variance estimator of $\hat{\theta} - \theta$ in (9).

Appendix C: Comparison of simple and proportional sampling

1. To prove $e_s^2 \ge e_p^2$ note that

$$e_s^2 - e_p^2 = \sum_{i=1}^{M} q_i (q_i + \frac{1}{n}) \{(1-q_i)^n - \prod_{l=1}^{K} (1 - q_{li})_l^n\} \tag{16}$$

and because $\ln(1-x)$ is a convex function, by Jensen's inequality

$$\ln(1 - \sum_{l=1}^{K} \alpha_l q_{li}) \ge \sum_{l=1}^{K} \alpha_l \ln(1 - q_{li}),$$

or

$$n \ln(1 - q_i) \ge \sum_{l=1}^{K} \alpha_l \ln(1 - q_{li}), \tag{17}$$

Thus, (16)≥ 0, or $e_s^2 \ge e_p^2$.

2. To prove $E[\hat{\theta}_p] \le E[\hat{\theta}_s]$, note that

$$E[\hat{\theta}_s] = \sum_{i=1}^{M} q_i (1 - q_i)^n.$$

Again by (17),

$$E[\hat{\theta}_p] \le \sum_{i=1}^{M} \sum_{k=1}^{K} \alpha_k q_{ki} (1 - q_i)^n = E[\hat{\theta}_s].$$

Acknowledgement

The authors are thankful for some valuable inputs from Dr. T. Maiti.

References

Adams, E. N. (1984). Optimizing preventive service of software product. *IBM Journal of Research and Development*, 2-14.

Bunge, J. and Fitzpatrick, M. (1993). Estimating the Number of Species: A review. *Journal of the American Statistical association*, **88**, 384-373.

Chao, A., and Lee, S. M. (1992). Estimating the Number of Classes Via Sample Coverage. *Journal of the American Statistical association*, **87**, 210-217.

Chao, A., Ma, C. M., and Yang, M. C. K. (1993). Stopping Rules and Estimation for Recapture Debugging with Unequal Failure Rates. *Biometrika*, **80**, 193-201.

Dalal, S. R., and Mallows, C. L. (1988). When Should One Stop Testing software? *Journal of the American Statistical association*, **83**, 872-879.

Esty, W. W. (1983). A Normal Limit Law for a Nonparametric Estimator of the Coverage of a Random Sample. *The Annals of Statistics*, **11**, 905-912.

Frankl, P. G., Hamlet, R. G., Littlewood, B., and Strigini, L., (1998). Evaluating Testing Methods by Delivered Reliability. *IEEE Transaction on Software Engineering*, **24**, 586-601.

Goel, A. L., and Okumoto, K. (1984). Time-dependent Error-detection Rate Model for Software Reliability and Other Performance Resources. *IEEE Transactions on Reliability*, **R-33**, 176-183.

Good, I. J., (1953). The Population Frequencies of Species and the Estimation of Population Parameters. *Biometrika*, **40**, 237-264.

Goudie, I. B. J. (1990). A Likelihood-based Stopping Rule for Recapture Debugging. *Biometrika*, **77**, 203-206.

Haas, P. J., Liu, Y. and Stokes, L. (2006). An estimator of number of species from quadrat sampling. *Biometrics*, **62**, 135-141

Holst, L. (1979). A Unified Approach to Limit Theorems for Urn Models. *Journal of Applied Probability*, **16**, 154-162.

Jelinski, Z. and Moranda, P. B. (1972). Software Reliability Research. in *Statistical Computer Performance Evaluation*. W. Freiberger, Ed., New York, Academic Press, 465-484.

Littlewood, B. (1981). Stochastic Reliability Growth: A Model for Fault-removal in Computer Programs and Hardware Designs. *IEEE Transaction on Reliability*, **30**, 313-320.

Littlewood, B. and Verrall, J. L. (1973). A Bayesian Reliability Growth Model for Estimating Software Reliability. *Journal of the Royal Statistical Society (C)*, 332-346.

Miller, K. W., Morell, L. J., Noonan, R. E., Park, S. K., Nicol, D. M., Murrill, B. W., and Voas, J. M. (1992). Estimating the Probability of Failure When Testing Reveals No Failure. *IEEE Transactions on Software Engineering*, **18**, 33-42.

Mills, D. M., Dyer, M. and Linger, R. C. (1987). Cleanroom Software Engineering. *IEEE Transactions on Software Engineering*. **13**, 19-25.

Musa, J. D., lannino, A. and Okumoto, K. (1987). *Software Reliability Measurement Prediction Application*, New York: McGraw-Hill.

Myers, G. J. (1976). *Software Reliability Principles & Practices*. New York: John Wiley.

Ostrand, T.J. Weyuker, E.J. and Bell, R.M. (2005). Predicting the location and number of faults in large software systems. *IEEE Transactions on Software Engineering*, **31**, 340- 355.

Podgurski, A., Yang, C., and Masri, W. (1993). Partition Testing, Stratified Sampling, and Cluster Analysis. *Proceedings of ACM SIGSOFT Symposium on the Foundations of Software Engineering*, 169-181.

Singpurwalla, N. D., and Wilson, S. P. (1999). *Statistical Methods in Software Engineering*. New York: Springer.

Stokes, S. L. (2003). Using auxiliary information for improving estimation in the number of species problem. *Statistica Sinica*, **13**, 655-671

Yamada, S. Hishitani, J. and Osaki, S. (1993). Software-reliability Growth with a Weibull Test-effort: A Model & Application. *IEEE Transactions on Reliability*, **42**, 100-105.

Yang, M. C. K., and Chao, A. (1995). Reliability-estimation and Stopping-rules for Software Testing, Based on Repeated Appearances of Bugs. *IEEE Transactions on Reliability*, **44**, 315-321.

Zhou, Y. and Leung H. (2006). Empirical Analysis of Object-Oriented Design Metrics for Predicting High and Low Severity Faults. *IEEE Transactions on Software Engineering*, **32**, 771- 789.

[Received December 2006; accepted January 2007.]