Title	A Gibbs-sampler approach to estimate the number of faults in a system using capture-recapture sampling
Author(s)	Hayakawa, Y; Yip, PSF
Citation	Ieee Transactions On Reliability, 2000, v. 49 n. 4, p. 342-350
Issued Date	2000
URL	http://hdl.handle.net/10722/43503
Rights	Creative Commons: Attribution 3.0 Hong Kong License

A Gibbs-Sampler Approach to Estimate the Number of Faults in a System Using Capture-Recapture Sampling

Yu Hayakawa and Paul S. F. Yip

***Abstract*—A new recapture debugging model is suggested to estimate the number of faults in a system, ν, and the failure intensity of each fault, ϕ. The Gibbs sampler and the Metropolis algorithm are used in this inference procedure. A numerical illustration suggests a notable improvement on the estimation of ν and ϕ compared with that of a removal debugging model.**

***Index Terms*—Capture-recapture sampling, credible interval, Gibbs sampler, metropolis algorithm.**

I. Introduction

A. Acronyms[1]

CODA	Convergence Diagnosis and Output Analysis software for Gibbs sampling output
HPP	homogeneous Poisson process
H-Y	Hayakawa-Yip model (developed in this paper)
J-M	Jelinski-Moranda model
L-V	Littlewood-Verrall model
MLE	maximum likelihood estimate
PCPO	prequential conditional predictive ordinate
TTT	total time on test

B. Notation

ϕ	failure intensity for each fault
ν	number of faults in a system
$\{N_i(t);$ $t \geq 0\}$	counting process of failure occurrences caused by fault i
$Y_i(t)$	indicator function of the event that fault i is undetected or its counter is not removed on $[0,t]$
$\mathcal{F}_t$	σ-algebra generated by $\{N_i(s);\ s \in [0,t],\ i = 1,\cdots,\nu\}$
$S_{i,j}$	encounter-time $\#j$ caused by fault i
X_i	$S_{i,1}$
$\mathcal{N}$	$\{N(t);\ t \geq 0\}$ of encounter times caused by all faults in a system: $N(t) = \sum_{i=1}^{\nu} N_i(t)$
$\mathrm{TTT}_i(t)$	TTT with respect to fault i
$\pi(\cdot\|*)$	conditional distribution, given $*$
α, β	[shape, scale] parameter
$G(\alpha,\beta)$	Gamma distribution with parameters α, β
$\lambda(t)$	intensity process for $N(t)$
$\mathcal{M}(t)$	zero-mean martingale
$\mathrm{poi}(\omega)$	Poisson distribution with mean ω
$\delta_i(t)$	indicator function: at least 1 failure caused by fault i occurs in $[0,t]$
ζ	remaining number of faults in a system
q	quantile to be estimated
r	desired accuracy of an estimate of q
s	Pr{an estimate of q within $\pm r$ is obtained}
k	thinning interval of the Gibbs chain
M	number of initial iterations to be discarded
N	number of further iterations required to estimate q to accuracy r with probability s (Not $\mathcal{N}$)
$N_{\min}$	minimum number of iterations needed to estimate q, when the samples in the Gibbs chain are s-independent
I	$N/N_{\min}$
T_i	$X_i - X_{i-1}$: interfailure time $\#i$
λ_i	failure rate of T_i under the L-V model
$\psi(i)$	scale parameter of the prior for λ_i
c_i	PCPO for T_i

In the early 1970s software reliability models began to appear in the literature, *e.g.*, [1, J-M], [2], and various models for software reliability have been suggested since then. For a comprehensive study of software reliability models, see [3]–[5].

J-M is one of the most well-known and earliest models in which the interfailure times are distributed {\it s}-independently as exponential with failure rate proportional to the remaining number of faults in the system, and the faults are removed permanently upon detection without introducing new faults. This model can be interpreted as an exponential order statistic model with a constant failure intensity [6].

Reference [7, Nayak] extends the J-M model [1]: a recapture debugging model; the assumptions are the same as those for the J-M model except that the number of times the "faults" are accessed after their removal is recorded by using of counters inserted into the location of detected faults. Therefore, the faults in a system are accessed according to s-independent HPP with the same failure intensity ϕ. The motivation for the recapture debugging model is to obtain extra information on ϕ and ν, so that they can be estimated better. The MLE and moment estimators of ν and ϕ are based on the recapture sampling information along with the first failure detection times of individual faults.

Manuscript received: May 30, 1998; revised: March 24, 1999, January 18, 2000, February 10, 2000.

This work is supported by the Research Grant Council of the Hong Kong Special Administrative Government.

Y. Hayakawa is with the School of Mathematical and Computing Sciences, Victoria University of Wellington, Wellington, New Zealand (e-mail: yu.hayakawa@vuw.ac.nz).

P. S. F. Yip is with the Department of Statistics and Actuarial Science, The University of Hong Kong, Hong Kong (e-mail: sfpyip@hku.hk).

Publisher Item Identifier S 0018-9529(00)11780-4.

[1]The singular and plural of an acronym are always spelled the same.

Following the Nayak [7] suggestion, this paper constructs a recapture debugging model which includes the Nayak model as a special case. The distinguishing properties of the H-Y model from the Nayak model are:

1. The removal of counters is allowed.
2. ν and ϕ are treated as random quantities.

The H-Y model can be considered as an extension of any Bayes version of the J-M model, *e.g.*, a model studied in Kuo & Yang [8]. H-Y is studied via the Gibbs sampler and is compared with a removal debugging model: a Bayes version of the J-M model. The L-V model is also used to validate H-Y.

Section II describes the H-Y assumptions in detail.

Section III explains briefly the Gibbs sampler and the Metropolis algorithm which are instrumental for our inference on the remaining number of faults.

Section IV provides a numerical illustration of H-Y: first the assumption of exponentiality of failure times is validated via the modified Anderson-Darling statistics; then another data set is simulated to compare recapture and removal debugging models; finally the posterior analysis on the parameters of interest is presented.

Section V compares the removal and recapture versions of the J-M model and a version of the L-V model based on the criterion of the prequential conditional predictive ordinate.

Section VI presents the results.

II. Model Assumptions

Consider a system which contains an unknown number of faults, ν; $\{N_i(t), t \geq 0\}$ is the counting process for fault i, $i = 1, \cdots, \nu$.

A. Assumptions

1. When a fault is detected for the first time, it is immediately removed without introducing new faults, and a counter is inserted so that the number of occurrences of failures caused by the fault can be counted as if it were still there.
2. The CPU-clock is stopped each time a new fault is detected. After this fault is removed, the CPU-clock is restarted when the testing resumes.
3. Each fault has the same ϕ.
4. Given ν and ϕ, the $N_i(t)$ are s-independent HPP with failure intensity ϕ.

In practice, the assumptions of perfect repair, immediate correction, and identical fault-detection rate are unrealistic. These would likely lead to an optimistic estimate for the remaining number of faults in the system [5, pp. 32–33, 54–56]. However, [9] shows that some software reliability models can be derived from Bayes versions of the J-M model, and many models can be considered as generalizations of the J-M model. Hence, as in [7], the J-M model is used as a starting point to present the approach to estimation and prediction in software reliability. This approach can be extended to other models.

To account for removals of the counters, see the following stochastic processes: for $i = 1, \cdots, \nu$,

$$Y_i(t) = \begin{cases} 1 & \text{if fault } i \text{ is undetected or its counter is not removed in } [0, t), \\ 0 & \text{otherwise.} \end{cases}$$

In the Nayak model, $Y_i(t) = 1$ for any t. From the definition of $S_{i,j}$,

$$S_{i,1} < S_{i,2} < \cdots < S_{i,N_i(t)}, \qquad i = 1, \cdots, \nu.$$

Then the likelihood function is

$$\begin{aligned} L(\phi, \nu | \mathcal{F}_t) = {} & \frac{\nu!}{\left(\nu - \sum_{i=1}^{\nu} \delta_i(t)\right)!} \\ & \cdot \prod_{i=1}^{\nu} \phi^{N_i(t)} \cdot \exp[-\phi\{S_{i,N_i(t)} \\ & + Y_i(t) \cdot (t - S_{i,N_i(t)})\} \\ \delta_i(t) \equiv {} & \begin{cases} 1 & \text{if } S_{i,1} \leq t, \\ 0 & \text{otherwise.} \end{cases} \end{aligned} \tag{1}$$

$\mathrm{TTT}_i(t) = S_{i,N_i(t)} + Y_i(t) \cdot (t - S_{i,N_i(t)})$ is the total time on test with respect to fault i. If fault i is detected and its counter is removed before t, say at time s $(s < t)$, then $\mathrm{TTT}_i(t) = s$. Otherwise $\mathrm{TTT}_i(t) = t$ if fault i is undetected on $[0, t]$ or its counter is never removed on $[0, t]$.

Now consider the counting process, $\mathcal{N} = \{N(t), t \geq 0\}$, of all faults in the system:

$$N(t) = \sum_{i=1}^{\nu} N_i(t), \qquad t \geq 0.$$

Assume that, given ν, ϕ has the gamma distribution, *a priori:*

$$\pi(\phi | \mathcal{F}_0) \sim G(\alpha, \beta).$$

By Bayes theorem,

$$\begin{aligned} \pi(\phi | \mathcal{F}_t) & \propto \pi(\phi | \mathcal{F}_0) \cdot L(\phi, \nu | \mathcal{F}_t) \\ & \propto \phi^{\alpha - 1} \cdot \exp(-\beta \cdot \phi) \\ & \quad \cdot \prod_{i=1}^{\nu} \phi^{N_i(t)} \cdot \exp[-\phi \cdot \mathrm{TTT}_i(t)] \\ & = \phi^{\alpha + \sum_{i=1}^{\nu} N_i(t) - 1} \\ & \quad \cdot \exp\left(-\phi \cdot \left[\beta + \sum_{i=1}^{\nu} \mathrm{TTT}_i(t)\right]\right). \end{aligned}$$

Therefore,

$$\pi(\phi | \mathcal{F}_t) \sim G\left(\alpha + \sum_{i=1}^{\nu} N_i(t), \beta + \sum_{i=1}^{\nu} \mathrm{TTT}_i(t)\right).$$

$$\lambda(t)\, dt = \mathrm{E}\{dN(t) | \mathcal{F}_t\},$$

$dN(t)$ is the increment of $\mathcal{N}$ over the interval $[t, t + dt)$, [10]. Given ϕ and ν, then $N(t)$ is the sum of s-independent HPP; hence the failure intensity for $N(t)$ is:

$$\left\{\sum_{i=1}^{\nu} Y_i(t)\right\} \cdot \phi.$$

The posterior mean of ϕ given $\mathcal{F}_t$ is:

$$\frac{\alpha+\sum_{i=1}^{\nu} N_i(t)}{\beta+\sum_{i=1}^{\nu} \mathrm{TTT}_i(t)}.$$

Therefore,

$$\lambda(t)=\sum_{i=1}^{\nu} Y_i(t)\cdot\frac{\alpha+\sum_{i=1}^{\nu} N_i(t)}{\beta+\sum_{i=1}^{\nu} \mathrm{TTT}_i(t)}.$$

$$\mathcal{M}(t)=N(t)-\int_0^t \lambda(s)\,ds. \tag{2}$$

The intensity processes of the J-M model and the Littlewood model [11] are in [10, pp. 133–134], which are relevant to (2). The intensity processes of some other models are in [12]–[16].

III. Inference Procedures

Inference on the 'remaining number of faults' and the 'failure intensity per fault' is made from a Bayes perspective. Assign a distribution to ν, and assume a hierarchical model for ϕ. Estimates of ϕ and ν are obtained using the Gibbs sampler [17] along with the Metropolis algorithm [18]; these are Markov chain Monte Carlo methods. As explained in [19], the Gibbs sampler is a special case of the Metropolis algorithm which was generalized in [20]. The Gibbs sampler enables one to generate random samples from the target density, in our setting posterior densities of the parameters, by iteratively simulating values from the full conditionals. It was popularized in [17] and has had a great impact on some of the statistical literature. References on both practical and theoretical issues of the Gibbs sampler are abundant, *e.g.*, [21]–[23]. Both algorithms are now briefly described.

A. Gibbs Sampler

We want to generate samples from the posterior density of $(\theta_1,\cdots,\theta_d)$ given information $\mathcal{F}_t$. Begin with initial values $(\theta_1^{(0)},\cdots,\theta_d^{(0)})$. Then sample iteratively as follows: For $i=0,1,\cdots,k-1$,

sample $\theta_1^{(i+1)}$ from $f(\theta_1,|\theta_2^{(i)},\cdots,\theta_d^{(i)},\mathcal{F}_t)$;
sample $\theta_2^{(i+1)}$ from $f(\theta_2,|\theta_1^{(i+1)},\theta_3^{(i)},\cdots,\theta_d^{(i)},\mathcal{F}_t)$;
$\ddots$
sample $\theta_d^{(i+1)}$ from $f(\theta_d,|\theta_1^{(i+1)},\cdots,\theta_{d-1}^{(i+1)},\mathcal{F}_t)$,
$f(\theta_j|\theta_1^{(i+1)},\cdots,\theta_{j-1}^{(i+1)},\theta_{j+1}^{(i)},\cdots,\theta_d^{(i)},\mathcal{F}_t)$
$\equiv$ posterior full conditional densities of θ_j,
$j=1,\cdots,d$.

Under certain conditions [22], as $k\to\infty$, $(\theta_1^{(k)},\cdots,\theta_d^{(k)})$ converges to $(\theta_1,\cdots,\theta_d)$ in distribution, whose density is $f(\theta_1,\cdots,\theta_d|\mathcal{F}_t)$. Likewise, for k large enough, $\theta_j^{(k)}$ is effectively a sample from the posterior marginal of θ_j, $f(\theta_j|\mathcal{F}_t)$, $j=1,\cdots,d$.

B. Metropolis Algorithm

Begin with initial values

$$\theta^{(0)}=(\theta_1^{(0)},\cdots,\theta_d^{(0)}).$$

Then the sampling proceeds as: for $i=0,1,\cdots,k-1$, generate y from the candidate generating density $q(\theta^{(i)},y)$; accept y as the value of $\theta^{(i+1)}$ with probability

$$\min\left[1,\frac{f(y|\mathcal{F}_t)}{f(\theta^{(i)}|\mathcal{F}_t)}\right],$$

otherwise the value of the chain remains unchanged; repeat this procedure to create a chain $\{\theta^{(1)},\cdots,\theta^{(k)}\}$.

For k large enough, $\theta^{(k)}=(\theta_1^{(k)},\cdots,\theta_d^{(k)})$ is effectively a sample from the target density $f(\theta|\mathcal{F}_t)$, the joint posterior density of $\theta=(\theta_1,\cdots,\theta_d)$.

Although one can never conclude with certainty when these 2 procedures should be terminated and the simulated values used for analyses of unobservables, it is advisable to perform convergence diagnostics on the simulated values. For a numerical illustration of this inference procedure in Section IV, the diagnostic method in [24] is chosen and is implemented by the CODA software (Convergence Diagnosis and Output Analysis Software for Gibbs sampling output) [25], [26] which is a set of S-Plus routines for 'convergence diagnostics and output analysis' for the Markov chain Monte Carlo methods. Reference [27] compares various convergence diagnostic methods.

For the example in Section IV, assume the following priors for the parameters.

$$\pi(\phi|\alpha,\beta)\sim G(\alpha,\beta) \tag{3}$$
$$\pi(\alpha|\lambda,\mu)\sim G(\lambda,\mu) \tag{4}$$
$$\pi(\beta|\gamma,\delta)\sim G(\gamma,\delta) \tag{5}$$
$$\pi(\nu|\omega)\sim \mathrm{poi}(\omega) \tag{6}$$

The full posterior conditionals necessary to implement the Gibbs sampler are:

$$\pi(\phi|\nu,\alpha,\beta,\mathcal{F}_{t-})\sim G\left(\alpha+\sum_{i=1}^{\nu}N_i(t),\beta+\sum_{i=1}^{\nu}\mathrm{TTT}_i(t)\right)$$
$$\pi(\beta|\nu,\phi,\alpha,\mathcal{F}_{t-})\sim G(\gamma+\alpha,\delta+\phi)$$
$$\pi(\zeta|\phi,\alpha,\beta,\mathcal{F}_{t-})\sim \mathrm{poi}[\omega\cdot\exp(-\phi\cdot t)]$$
$$\zeta\equiv\nu-\sum_{i=1}^{\nu}\delta_i(t), \tag{7}$$

viz, the remaining number of faults in the system. As for α, its full posterior conditional can be obtained up to a proportionality constant:

$$\frac{\pi(\alpha|\mathcal{F}_0)\beta^{\alpha}\cdot\phi^{\alpha-1}}{\Gamma(\alpha)}.$$

Following the studies [8], [28], use the Metropolis algorithm to generate samples from the full posterior conditional distribution of α.

TABLE I
FAILURE DATA FOR AN INFORMATION SYSTEM AND SIMULATED ENCOUNTER TIMES

Fault #	1st	Simulated Subsequent Failure Times					
1	880	64653	329685	520016			
2	4310	305937	428364	432134	576243		
3	7170	563910					
4	18930	186946	195476	473206			
5	23680	259496	469180				
6	23920	126072	252204	371939			
7	26220	251385					
8	34790	353576					
9	39410	53878	147409	515884			
10	40470	371824	466719				
11	44290	83996	327296	352035	395324	494037	
12	59090	61435	222288	546577			
13	60860	75630	576386				
14	85130						
15	89930	205224	292321	294935	342811	536573	553312
16	90400	228283	334152	360218	368811	377529	547048
17	90440	511836	511967				
18	100610	367520	429213				
19	101730	162480	534444				
20	102710	194399	294708	295030	360344	511025	
21	127010	555065					
22	128760						
23	133210	167108	370739				
24	138070	307101	451668				
25	138710	232743					
26	142700	215589					
27	169540	299094	428902	520533			
28	171810	404887					
29	172010	288750					
30	211190						
31	226100	378185	446070	449665			
32	240770	266322	459440				
33	257080	374384					
34	295490	364952					
35	296610						
36	327170	374032	430077				
37	333380						
38	333500	480020					
39	353710						
40	380110	433074					
41	417910	422153	479514	511308			
42	492130						
43	576570						

IV. A NUMERICAL EXAMPLE

To illustrate the proposed inference procedure, the data-set from [10, p. 43] (originally reported in [29]) is used. It has 43 occurrence times (in seconds CPU time) of first failures caused by 43 distinct faults of an information system for registering aircraft movements; these numbers are in column 2 of Table I.

Step 1. Anderson-Darling statistics are computed to check if failure times in this data-set are identically exponentially distributed.

Step 2. The data-set is augmented by simulating subsequent failure times of each detected fault by treating failure-time #43 as a stopping time.

Step 3. CODA is explained and the Gibbs sampler is used to conduct analyzes on ν and ϕ regarding both recapture and removal debugging models.

A modified version of Anderson-Darling statistics is used to test the exponentiality of failure times [30], [31, pp. 114–115, 141–145]. The Table I data-set is assumed to be right (type 2) censored. A test procedure [31] is given next.

$X_1, \cdots, X_\nu$ are failure times, and
$X_{(1)}, \cdots, X_{(\nu)}$ are the corresponding order statistics.

Observation is stopped at $X_{(r)}$.

1. Compute

$$\hat{\beta} = \frac{\left[\sum_{i=1}^{r} X_{(i)}\right] + (\nu - r) \cdot X_{(r)}}{r}.$$

2. Calculate

$$Z_{(i)} = 1 - \exp\left(-\frac{X_{(i)}}{\hat{\beta}}\right), \qquad i = 1, \cdots, r.$$

TABLE II
MODIFIED ANDERSON-DARLING STATISTICS

ν	${}_2A^2_{r,n}$	ν	${}_2A^2_{r,n}$
43	0.2706479	52	0.4914793
44	0.1759536	53	0.5621464
45	0.1408457	54	0.6315637
46	0.1455952	55	0.6990203
47	0.1768338	56	0.7640453
48	0.2253955	57	0.8263412
49	0.2849368	58	0.8857355
50	0.3510377	59	0.9421463
51	0.4206065	60	0.9955566

3. Calculate a modified version of Anderson-Darling statistics:

$$
\begin{aligned}
{}_2A^2_{r,n} = &-\frac{1}{n}\cdot\sum_{i=1}^{r}(2i-1)\cdot[\log(Z_{(i)})-\log(\overline{Z}_{(i)})] \\
&-2\sum_{i=1}^{r}\log(\overline{Z}_{(i)})-\frac{1}{n} \\
&\cdot[(r-n)^2\cdot\log(\overline{Z}_{(r)})-r^2\cdot\log(Z_{(r)})+n^2\cdot Z_{(r)}]; \\
\overline{Z}_{(\cdot)} &\equiv 1-Z_{(\cdot)}.
\end{aligned}
$$

Since ν is unknown, compute ${}_2A^2_{r,n}$ for various values of $\nu \geq 43$; the results are in Table II.

Refer the ${}_2A^2_{r,n}$ values in Table II to the percentage points in '[31, Table 4.16, pp. 143–144];' the hypothesis of exponentiality of $X_1,\cdots,X_\nu$, *ie*, $S_{1,1},\cdots,S_{\nu,1}$ cannot be rejected at the 5% s-significance level for $\nu = 43,\cdots,58$, nor at the 2.5% s-significance level for ν = 59, 60, based on interpolated percentage points, which does not reject the assumption.

From column 2 of Table I, compute [10, pp. 418–419] the MLE for ϕ of individual faults: $\hat{\phi} = 5.40\cdot 10^{-6}$.

Now generate interfailure times for each fault from the exponential distribution with failure rate $\hat{\phi}$. Column 3 of Table I contains the simulated failure encounter times between the first encounter time of individual faults and the stopping time, 576 570 seconds in CPU time.

If the removal of counters was allowed, the $\mathrm{TTT}_i(t)$ and/or $N_i(t)$ would be changed. For example, consider fault 15; suppose that its counter is removed at 400 000 seconds in CPU time. Then,

$$\mathrm{TTT}_{15}(576\,570) = 400000, \quad N_{15}(576\,570) = 5$$

instead of 576 570 and 7, respectively. The only change in implementation of the Gibbs sampler in this case would be through the full posterior conditional of ϕ, because it depends on $\mathrm{TTT}_i(t)$s and $N_i(t)$s (7).

For both recapture and removal debugging models, two sets of priors, (A) and (B), are used; see (3)–(6) for notation. To study the effect of prior means for ν, denoted by ω, the only difference between the (A) and (B) priors is the value of ω.

If there exist past data of similar software, then the parameters of the prior (referred to as hyperparameters) can be determined, based on such data. See [32, chapter 3] for various methods on elicitation of prior information. As an *ad hoc* approach, one could choose values for the hyperparameters using estimates of the parameters obtained by classical methods such as MLE.

TABLE III
HYPERPARAMETER VALUES

selection	λ	μ	γ	δ	ω
(A)	5	100	100	0.01	58
(B)	5	100	100	0.01	73

TABLE IV
AN OUTPUT OF CODA—RAFTERY & LEWIS METHOD [24]

RAFTERY AND LEWIS CONVERGENCE DIAGNOSTIC:
Iterations used = 1:8000
Thinning interval = 1, Sample size per chain = 8000
Quantile = 0.025
Accuracy = ± 0.0125, Probability = 0.95
Chain: chain_1

Variable	k	M	N	$N_{\min}$	I
α	1	2	608	600	1.01
β	1	2	590	600	0.983
ϕ	1	4	749	600	1.25
ζ	1	3	1078	600	1.8

The method in [24] is implemented by CODA to detect convergence to the stationary distribution, and is used to determine the number of iterations to achieve a prespecified accuracy of an estimate, the spacing between iterations, and the number of burn-in iterations.

Next is an analysis based on [24] regarding a Gibbs chain, a sequence of samples simulated by the Gibbs sampler, of the recapture model with prior choice (B).

This method requires a quantile, q, to be estimated, a desired accuracy, r, of the estimate of q, and a probability, s, of obtaining an estimate of q within $\pm r$. N represents the number of further iterations required to estimate q to a prespecified accuracy r with probability s. $N_{\min}$ is the minimum number of iterations needed to estimate q, assuming that the samples in the Gibbs chain are s-independent. I measures the increase in the number of iterations needed for estimation of q due to dependence in the Gibbs chain. An $I \gg 1$ often indicates a high level of correlation in the Gibbs chain. Reference [24] shows that $I > 5$ often indicates difficulties and suggests changes in the implementation such as starting values and reparameterization.

For example, from the results associated with ϕ in Table IV, discard the initial 4 iterations and require 749 further iterations with the thinning interval of 1 to estimate 0.025 quantile of ϕ to within ± 0.0125 with 0.95 probability. The chain of ϕ values converges to the stationary distribution very quickly. I is close to 1, hence the iterates of the chain do not show a high level of correlation.

For convergence diagnostics of all the Gibbs chains simulated to analyze this numerical example, choose $q = 0.025$, $r = 0.0125$, $s = 0.95$. In most cases, the number of burn-in iterations are less 5, and 2000 iterations are more than sufficient to achieve the accuracy in the previous paragraph.

Table V summarizes the results from CODA's output analysis of the Gibbs chains, which contain estimates of the:

- posterior mean
- posterior standard deviation

TABLE V
ESTIMATES OF POSTERIOR MEANS, STANDARD DEVIATIONS AND PERCENTILES

var	data	ω	post. mean	SD	2.5%	97.5%
ζ	original	58	5.72	4.19	0	16
ζ	original	73	15.15	8.60	2	34
ζ	generated	58	4.27	2.39	1	10
ζ	generated	73	5.78	2.87	1	12
ϕ	original	58	$4.33{\cdot}10^{-6}$	$1.09{\cdot}10^{-6}$	$2.41{\cdot}10^{-6}$	$6.57{\cdot}10^{-6}$
ϕ	original	73	$3.00{\cdot}10^{-6}$	$1.05{\cdot}10^{-6}$	$1.47{\cdot}10^{-6}$	$5.50{\cdot}10^{-6}$
ϕ	generated	58	$4.61{\cdot}10^{-6}$	$4.70{\cdot}10^{-7}$	$3.73{\cdot}10^{-6}$	$5.57{\cdot}10^{-6}$
ϕ	generated	73	$4.47{\cdot}10^{-6}$	$4.77{\cdot}10^{-7}$	$3.59{\cdot}10^{-6}$	$5.45{\cdot}10^{-6}$

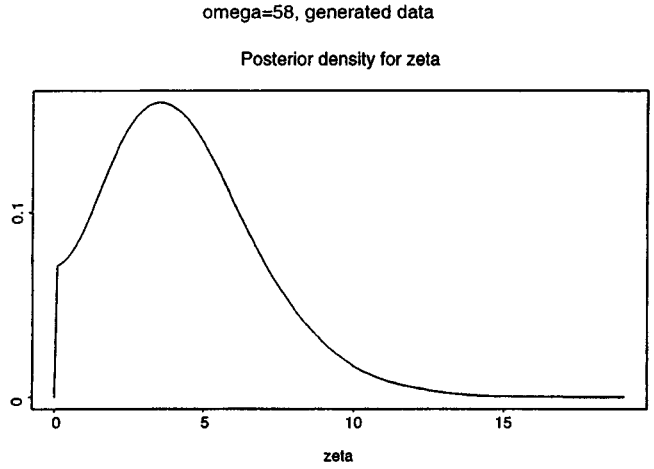

Fig. 1. Posterior (A)—Generated.

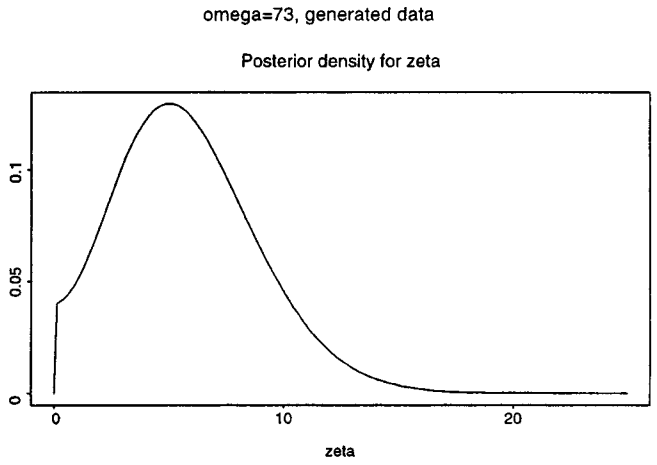

Fig. 2. Posterior (B)—Generated.

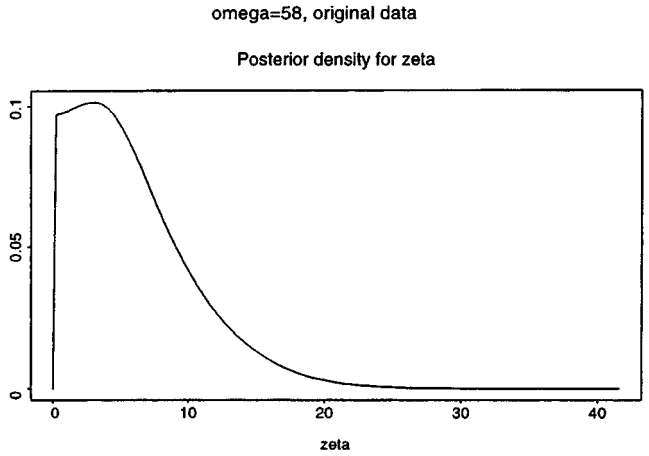

Fig. 3. Posterior (A)—Original.

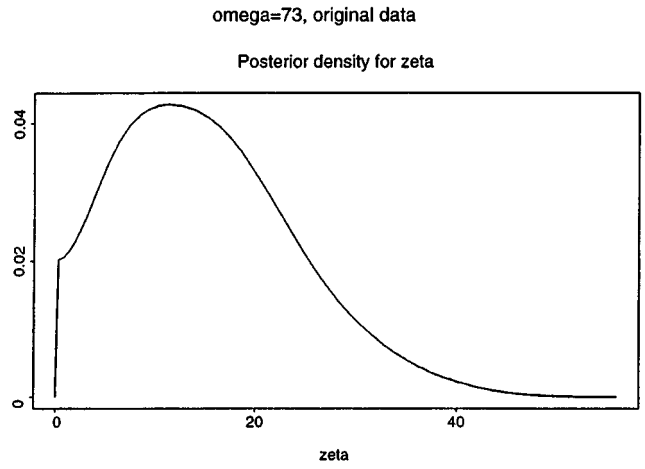

Fig. 4. Posterior (B)—Original.

- 2.5 percentile
- 97.5 percentile

of ζ and ϕ. The estimates of 95% credible intervals for ζ and ϕ can be obtained using the last 2 columns of Table II. In addition, the graphs of estimates of the posterior densities for ζ produced by CODA are in Figs. 1–4.

Under the quadratic loss function, the posterior mean is the Bayes point estimate; thus compare the posterior means of ζ and ϕ in regard to the choice of model and prior distribution. With respect to the recapture debugging model (denoted by "Generated" in Table V and Figs. 1–2), the point estimates of ζ are approximately 4.27 and 5.78 for the prior choices of (A) and (B), respectively.

For the removal model (denoted by 'Original' in Table V and Figs. 3–4), the prior distributions of ν corresponding to (A) and (B) have more bearing on the posterior estimates of ζ than those for the recapture debugging model.

The posterior standard deviations for ζ of the recapture debugging model are noticeably reduced compared with those of the removal debugging model. The posterior estimates of ϕ for the recapture debugging model are $4.61 \cdot 10^{-6}$ and $4.47 \cdot 10^{-6}$ corresponding to the prior choices (A) and (B), and those for the removal debugging model are $4.33 \cdot 10^{-6}$ and $3.00 \cdot 10^{-6}$,

respectively. Similar to the posterior estimates of ζ, the prior distributions of ϕ for the removal model have more effect on the posterior estimates of ϕ than those for the recapture model. The uncertainty about ϕ with respect to the recapture debugging model for both (A) and (B) is greatly reduced as opposed to those with respect to the removal debugging model. Consequently, the credible intervals of ϕ for the removal debugging model are wider than those for the recapture debugging model. The removal debugging model tends to give lower point estimates of ϕ than the recapture debugging model.

From these observations we conclude that the effect of ignoring subsequent encounter times of failures has a great impact on the estimation of ν and ϕ. With the recapture model, the estimates are likely to be more robust to prior selection of hyperparameters (parameters of the prior) values and are more accurate in the sense that the posterior distributions of the parameters have smaller standard-deviations compared to those of the removal model. Hence, one should use the information obtained from re-encounter times in practice in pursuit of more robust and accurate estimates of the parameters of interest.

V. Model Comparison

In order to validate the approach in this paper, the removal and recapture versions of the J-M model and a version of the L-V model are compared, based on the criterion of PCPO (defined in this section).

The L-V model [2] assumes that:

1. The interfailure times, T_i, $i = 1, \cdots, n$, are s-independent and exponentially distributed with failure rates λ_i.
2. The λ_i are s-independent; each λ_i is gamma distributed with shape parameter α and scale parameter

$$\psi(i) = \beta_0 + \beta_i \cdot i + \cdots + \beta_k \cdot i^k,$$

for some $k \geq 0$. Hence, the unconditional marginal distribution of T_i is Pareto with density

$$f(t_i|\alpha, \psi(i)) = \frac{\alpha \cdot [\psi(i)]^{\alpha}}{[t_i + \psi(i)]^{\alpha+1}}. \tag{8}$$

Unlike the J-M model, the L-V model allows imperfect debugging. Because $\psi(i)$ is increasing in i, the E$[T_i]$ increases in i.

Reference [8] uses the PCPO for model selection. This paper follows that approach for comparison of the H-Y model two other models. The PCPO for observation i is [8, p. 74] of the H-Y model with

$$c_i = p(t_i|t_1, \cdots, t_{i-1}), \qquad i \geq 2$$
$$c_1 = p(t_1).$$

For the removal and recapture versions of the J-M model, the c_i are approximated using the Gibbs chains by

$$\hat{c}_i = \hat{p}(t_i|t_1, \cdots, t_{i-1}) = \frac{1}{N} \cdot \sum_{j=1}^{N} \zeta^{(j)} \cdot \phi^{(j)} \cdot \exp[-\zeta^{(j)} \cdot \phi^{(j)}]$$

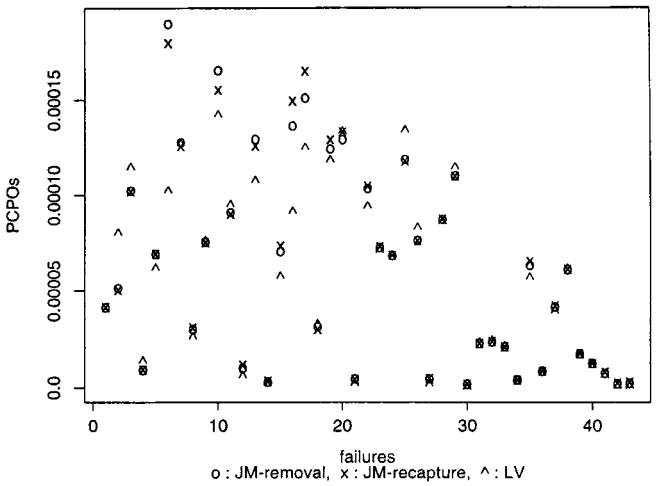

Fig. 5. PCPO of J-M Removal and Recapture Models and L-V Model.

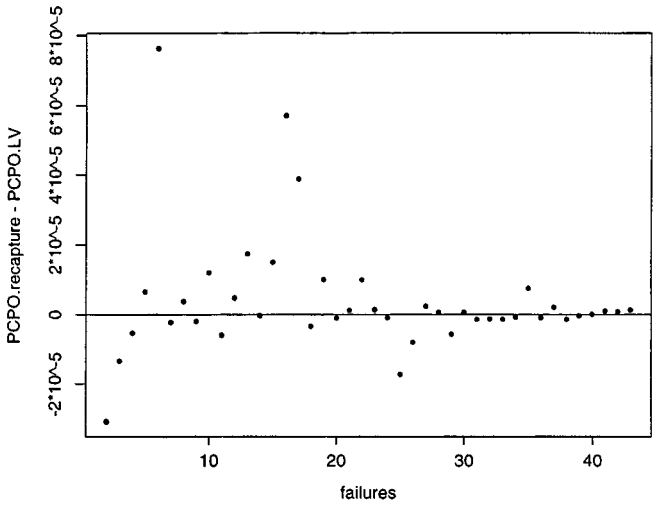

Fig. 6. (PCPO of J-M Recapture Model)—(PCPO of L-V Model).

$\zeta^{(j)}$ and $\phi^{(j)}$ denote the samples $\#j$ of the Gibbs chains of ζ and ϕ, and assume $\{t_1, \cdots, t_{i-1}\}$ is given after a convergence diagnostic is conducted.

For the L-V model, the linear function for $\psi(i)$ is used: $\psi(i) = \beta_0 + \beta_1 \cdot i$. To approximate c_i, first estimate α, β_0, β_1 using a least squares method. The estimates, $\hat{\alpha}$, $\hat{\beta}_0$, $\hat{\beta}_1$ can be obtained by minimizing

$$\sum_{j=1}^{i-1} \left(t_j - \frac{\psi(j)}{\alpha - 1} \right)^2;$$

$\psi(j)/(\alpha - 1)$ is the s-expected value of T_j [3, pp. 107–108]. c_i is approximated by substituting $\hat{\alpha}, \hat{\beta}_0, \hat{\beta}_1$ for α, β_0, β_1 respectively in (8).

Fig. 5 plots the PCPO of the J-M removal and recapture models with prior option of (A) and the L-V model. The PCPO of all models are larger for about the first 30 interfailure times, then they decrease for the remainder of the data points.

The difference between the PCPO of the J-M recapture model and the L-V model is plotted in Fig. 6. Fig. 7 plots the difference between those of the J-M recapture model and the removal model. In Fig. 6, the number of points above and below zero are

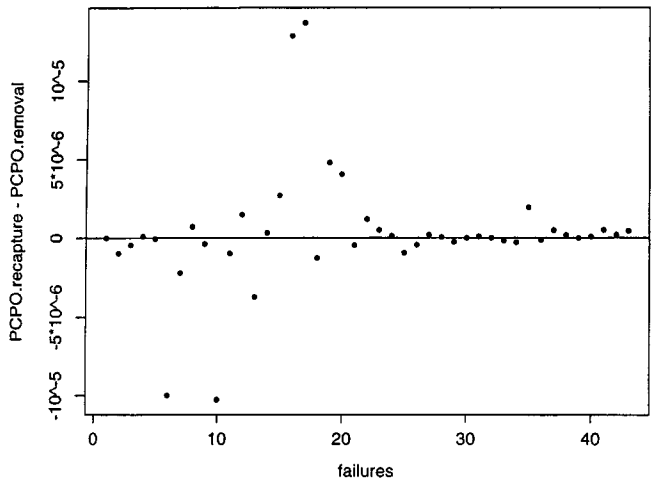

Fig. 7. (PCPO of J-M Recapture Model)—(PCPO of J-M Removal Model).

TABLE VI
MODEL COMPARISON

	J-M Recapture	J-M Removal	L-V
$\prod_{i=2}^{43} c_i$	$6.226014 \cdot 10^{-189}$	$3.453923 \cdot 10^{-189}$	$1.421187 \cdot 10^{-190}$
$\prod_{i=1}^{43} c_i$	$2.569416 \cdot 10^{-193}$	$1.425401 \cdot 10^{-193}$	Not Applicable

about the same, but the average magnitude of the 'points above 0' appears to be slightly bigger than that of the 'points below 0.' This suggests that the J-M recapture model is preferred for the data set. According to Fig. 7, the J-M removal and recapture models are analogous.

As a model selection criterion [8, p. 74], this paper maximizes the joint density of interfailure times of distinct faults,

$$p(t_1, \cdots, t_{43}) = \prod_{i=1}^{43} c_i.$$

Since there is no c_1 for the L-V model, compute $\prod_{i=2}^{43} c_i$ for all 3 models as well. These values are summarized in Table VI. Based on $\prod_{i=2}^{43} c_i$, the J-M recapture model is the best. It again is slightly more desirable than the J-M removal model by comparing the values of $\prod_{i=1}^{43} c_i$.

VI. RESULTS

The Nayak [7] recapture debugging model is generalized by allowing the removal of counters and treating the ν and ϕ as random quantities. The Gibbs sampler and the Metropolis algorithm are used to analyze a data-set through both recapture and removal debugging models. Two prior means for ν (denoted by ω) are used to study the sensitivity of estimates. The results in Section IV suggest that the removal model is likely to give lower estimates of ϕ than the recapture debugging model. Inferential results under the recapture debugging model are much less sensitive to the prior specification of ω than those under the removal model. Posterior standard deviations for both ζ and ϕ are greatly reduced in comparison with the removal model's counterparts. Hence, the recapture debugging models offers more robust estimates for ν and ϕ with less posterior standard-deviation than those based on the removal model.

ACKNOWLEDGMENT

We are pleased to thank G. Houston for his help coding the algorithms and getting outputs from CODA, and C. Lloyd, D. Harte, and P. Smith for their helpful, stimulating discussions. We are very grateful to the Associate Editor and the referees for their comments and suggestions. This study was done while Y. Hayakawa was visiting P. Yip at the Department of Statistics and Actuarial Science, University of Hong Kong. Y. Hayakawa greatly appreciates the hospitality shown by the Department.

REFERENCES

[1] Z. Jelinski and P. B. Moranda, "Software reliability research," in *Statistical Computer Performance Evaluation*, W. Freiberger, Ed: Academic Press, 1972, pp. 465–484.
[2] B. Littlewood and J. L. Verrall, "A Bayesian reliability growth model for computer software," *Applied Statistics*, vol. 22, pp. 332–346, 1973.
[3] *Handbook of Software Reliability Engineering*, McGraw-Hill, 1996.
[4] J. D. Musa, A. Iannino, and K. Okumoto, *Software Reliability—Measurement, Prediction, Application*: McGraw-Hill, 1990.
[5] M. Xie, *Software Reliability Modeling*: World Scientific, 1991.
[6] D. R. Miller, "Exponential order statistic models of software reliability growth," *IEEE Trans. Software Engineering*, vol. SE-12, pp. 12–24, 1986.
[7] T. K. Nayak, "Estimating population size by recapture sampling," *Biometrika*, vol. 75, pp. 113–120, 1988.
[8] L. Kuo and T. Yang, "Bayesian computation of software reliability," *J. Computational and Graphical Statistics*, vol. 4, pp. 65–82, 1995.
[9] N. Langberg and N. D. Singpurwalla, "A unification of some software reliability models," *SIAM J. Scientific and Statistical Computing*, vol. 6, pp. 781–790, 1985.
[10] P. K. Anderson, O. Borgan, R. D. Gill, and N. Keiding, *Statistical Models Based on Counting Processes*: Springer, 1992.
[11] B. Littlewood, "Theories of software reliability: How good are they and how can they be improved?," *IEEE Trans. Software Engineering*, vol. 6, pp. 489–500, 1980.
[12] G. Koch and P. J. C. Spreij, "Software reliability as an application of martingale & filtering theory," *IEEE Trans. Reliability*, vol. R-32, no. 4, pp. 342–345, Oct. 1983.
[13] M. C. J. van Pul, *Statistical Analysis of Software Reliability Models*: CWI Tracts, 1993.
[14] P. S. F. Yip, "Estimating the number of errors in a system using a martingale approach," *IEEE Trans. Reliability*, vol. 44, no. 2, pp. 322–326, June 1995.
[15] P. S. F. Yip, D. Y. T. Fong, and K. Wilson, "Estimating population size by recapturing sampling via estimating function," *Stochastic Models*, vol. 9, pp. 179–193, 1993.
[16] P. S. F. Yip, L. Xi, D. Y. T. Fong, and Y. Hayakawa, "Sensitivity analysis and estimating number of faults in removal debugging," *IEEE Trans. Reliability*, vol. 48, no. 3, pp. 300–305, Sept. 1999.
[17] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution, and the Bayesian restoration of images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. PAMI-6, pp. 721–741, 1984.
[18] N. Metropolis, A. W. Rosenbluth, and M. N. Rosenbluth, *et al.*, "Equations of state calculations by fast computing machines," *J. Chemical Physics*, vol. 21, pp. 1087–1092, 1953.
[19] S. Chib and E. Greenberg, "Understanding the Metropolis-Hastings algorithm," *The American Statistician*, vol. 49, pp. 327–335, 1995.
[20] W. K. Hastings, "Monte Carlo sampling methods using Markov chain and their applications," *Biometrika*, vol. 57, pp. 97–109, 1970.
[21] G. Casella and E. I. George, "Explaining the Gibbs sampler," *The American Statistician*, vol. 46, pp. 167–174, 1992.
[22] L. Tierney, "Markov chains for exploring posterior distributions'," *The Annals of Statistics*, vol. 22, pp. 1701–1762, 1994.
[23] M. A. Tanner, *Tools for Statistical Inference: Methods for the Exploration of Posterior Distributions and Likelihood Functions*, 3rd ed: Springer-Verlag, 1996.

[24] A. L. Raftery and S. Lewis, "How many iterations in the Gibbs sampler," in *Bayesian Statistics 4*: Oxford Univ. Press, 1992, pp. 763–774.
[25] N. G. Best, M. K. Cowles, and S. K. Vines, CODA Manual, ver 0.30, MRC Biostatistics Unit, 1995.
[26] ——, CODA Manual ver 0.40, MRC Biostatistics Unit, 1997.
[27] M. K. Cowles and B. P. Carlin, "Markov chain Monte Carlo convergence diagnostics: A comparative review," *J. American Statistical Assoc.*, vol. 91, pp. 883–904, 1996.
[28] L. Kuo and T. Yang, "Bayesian computation for nonhomogeneous Poisson processes in software reliability," *J. American Statistical Assoc.*, vol. 91, pp. 763–773, 1996.
[29] G. Moek, "Comparison of software reliability models for simulated and real failure data," *Int. J. Model Simulation*, vol. 4, pp. 29–41, 1984.
[30] A. N. Pettitt and M. A. Stephens, "Modified Cramér-von mises statistics for censored data," *Biometrika*, vol. 63, pp. 291–298, 1976.
[31] R. B. D'Agostino and M. A. Stephens, *Goodness-of-Fit Techniques*: Marcel Dekker, 1986.
[32] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*: Springer-Verlag, 1985.

Yu Hayakawa is a Senior Lecturer in the School of Mathematical and Computing Sciences at Victoria University of Wellington, New Zealand. She received a PhD (1992) in Operations Research from the University of California at Berkeley. Her research interests include reliability theory and Bayes statistics.

Paul Yip is a Senior Lecturer in the Department of Statistics and Actuarial Science of the University of Hong Kong. His research interests are in reliability studies and statistical modeling of biological and ecological systems.