

# Interval Estimation for Software Reliability Assessment based on MCMC Method

Shinji Inoue<sup>a,\*</sup> and Shigeru Yamada<sup>b</sup>

<sup>a</sup>Kansai University, 2-1-1, Ryozenji-cho, Takatsuki-shi, Osaka, 569-1095, Japan

<sup>b</sup>Tottori University, 4-101, Minami, Koyama-cho, Tottori-shi, Tottori, 680-8552, Japan

---

## Abstract

Interval estimation in assessment in software systems must be useful because it is hard to obtain a sufficient amount of software reliability data for conducting point estimation and the software reliability data is essentially incomplete. We discuss flexible software reliability measurement considering the uncertainty of model parameters in a reliability model. Concretely, applying a discrete-time domain model, we analyze the Bayesian inference of the parameters in the model by using the Markov chain Monte Carlo method. Furthermore, numerical illustrations of the approach for our method are also shown in this paper.

**Keywords:** software reliability assessment; software reliability model; integrable difference equation; Bayesian interval estimation; MCMC method

(Submitted on October 6, 2017; Revised on March 12, 2018; accepted on April 4, 2018)

© 2019 Totem Publisher, Inc. All rights reserved.

---

## 1. Introduction

The reliability of software system is treated as so-called "must-be" quality in software characteristics. Therefore, we need to ensure the software reliability requirement. It plays an especially important role in measuring the reliability attainment quantitatively to check the final quality of software. A software reliability growth model [1-2] enables us to assess software reliability and to obtain some useful information on software project management and decision making in software projects. In reliability assessment, point estimation is widely used to obtain information on the quality/reliability of a software system. Considering a practical situation, the interval estimation may be better in the situation that the data used for software reliability assessment is not sufficient. The interval estimation can estimate parameters with uncertainty. However, we need to analytically derive their probability distribution. In the case that the software reliability growth process observed follows a nonhomogeneous Poisson process (NHPP), which is widely applied in software reliability measurement, it is known that it is very hard to derive them.

To overcome the problems above, Kaneishi and Dohi [3] discussed nonparametric and parametric bootstrapping approaches for conducting reliability assessment with the uncertainty of software reliability parameters. Inoue and Yamada [4] proposed a nonparametric bootstrapping approach for interval estimation of the optimal software shipping time based on a discrete time (discretized) model. Nowadays, the Bayesian approach has been used to obtain the uncertainty of model parameters and some other software reliability assessment measures [5]. This paper focuses on the Bayesian approach as one of the solution mentioned above. In recent years, the Markov chain Monte Carlo (MCMC) approach [6] has been often used in the Bayesian estimation approach to obtain the Bayesian inference of parameter of interest. This paper discusses the Bayesian and simulation-based approach for software reliability assessment based on a discretized NHPP model [7]. It has been checked that the discretized NHPP model has high fitting and prediction performance for the actual data in our previous works [7]. Then, we analyze software reliability with the uncertainty of some measures related to software reliability through a certain interval estimation method in the Bayesian approach. Finally, we demonstrate our approach by using actual software fault-count data.

\* Corresponding author.

E-mail address: [ino@kansai-u.ac.jp](mailto:ino@kansai-u.ac.jp)

## 2. Discretized Software Reliability Growth Model

Now, we introduce a counting process  $\{N_i, i = 0, 1, 2, \dots\}$ , which denotes the total number of faults eliminated up to the  $i^{\text{th}}$  testing-period. A discrete-time NHPP [7] is given as

$$Pr\{N_i = x | N_0 = 0\} = \frac{\{\Lambda_i\}^x}{x!} \exp[-\Lambda_i], \quad i, x = 0, 1, 2, \dots \quad (1)$$

Where  $\Lambda_i$  is the expectation of  $N_i$ . Regarding a discretized exponential model,  $\Lambda_i (\equiv H_i)$  is characterized as

$$H_{i+1} - H_i = \delta b(a - H_i) \quad (2)$$

from the original differential equation [8]. In Equation (2),  $a$  is the expected number of faults remaining at the test beginning, and  $b$  is the fault detection rate per one fault. The exact solution  $H_i$  of Equation (2) is derived as

$$\Lambda_i \equiv H_i = a[1 - (1 - \delta b)^i], \quad a > 0, b > 0 \quad (3)$$

Where  $\delta$  represents the constant time-interval.

In the point estimation, the parameters in Equation (3) can be estimated by using the regression analysis. Supposing that fault counting data  $\mathcal{D} \equiv (i, y_i)$ , ( $i = 1, 2, \dots, n$ ) has been observed, we can obtain a regression equation from Equation (2) as

$$c_i = \alpha + \beta d_i \quad (4)$$

In the data,  $y_i$  represents the observed cumulative number of faults detected during  $(0, i]$ , where  $c_i = H_{i+1} - H_i \equiv y_{i+1} - y_i$ ,  $d_i = H_i \equiv y_i$ ,  $\alpha = \delta ab$ , and  $\beta = -\delta b$ , respectively. Concretely, we obtain the model parameters as  $\hat{a} = -\hat{\alpha}/\hat{\beta}$  and  $\hat{\delta b} = -\hat{\beta}$ , respectively.

After the parameter estimations, some measurements for reliability assessment derived from the property of Equation (1) play important roles in quantitative reliability assessment. As one of the measures, the discrete residual fault content,  $M_i$ , at the arbitrary testing-period can be derived as

$$M_i \equiv E[N_\infty - N_i] = a - \Lambda_i = a(1 - \delta b)^i \quad (5)$$

if the expectation of  $N_i$  obeys Equation (3). The software reliability function is defined as the failure free operation during time-interval  $(i, i + h]$ , ( $h = 1, 2, \dots$ ). Then, we have

$$\begin{aligned} R(i, h) &\equiv Pr\{N_{i+h} - N_i = 0 | N_i = x\} \\ &= \exp[-\{\Lambda_{i+h} - \Lambda_i\}] \\ &= \exp[-H_h(1 - \delta b)^i] \end{aligned} \quad (6)$$

## 3. Uncertainty of Parameters

The method of maximum-likelihood assuming  $c_i \sim N(\alpha + \beta d_i, \sigma^2)$  is essentially the same as the estimation procedure discussed in Section 2. The likelihood function for  $\mathcal{D}$  is derived as (see Appendix)

$$p(\mathcal{D} | \alpha, \beta, \sigma^2) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(c_i - \alpha - \beta d_i)^2}{2\sigma^2}\right] \propto \exp\left[-\frac{n(\alpha - \hat{\alpha})^2 + \sum_{i=1}^n (\beta - \hat{\beta})^2 d_i^2}{2\sigma^2}\right] \quad (7)$$

Where  $\hat{\alpha}$  and  $\hat{\beta}$  are the parameters estimated through the procedure discussed in Section 2. From the Bayes' theorem, we have  $p(\alpha | \beta, \sigma^2, \mathcal{D}) \propto p(\mathcal{D} | \alpha, \beta, \sigma^2) p(\alpha)$  when  $\mathcal{D}$ ,  $\beta$ , and  $\sigma^2$  are given. This means the prior of  $\alpha$ ,  $p(\alpha)$ , is updated by the likelihood function for  $\mathcal{D}$ ,  $p(\mathcal{D} | \alpha, \beta, \sigma^2)$ . Assuming  $\alpha \sim N(\mu_\alpha, \tau_\alpha^2)$ , we can derive the full conditional posterior for  $\alpha$  as

(see Appendix)

$$\alpha|\beta, \sigma^2, \mathcal{D} \sim N\left(\frac{n\hat{\alpha}\tau_\alpha^2 + \sigma^2\mu_\alpha}{\tau_\alpha^2 n + \sigma^2}, \frac{\sigma^2\tau_\alpha^2}{\tau_\alpha^2 n + \sigma^2}\right) \quad (8)$$

The full conditional posterior of  $\beta$  given  $\alpha, \sigma^2$ , and  $\mathcal{D}$  is derived as  $p(\beta|\alpha, \sigma^2, \mathcal{D}) \propto p(\mathcal{D}|\alpha, \beta, \sigma^2)p(\beta)$ . Then,

$$\beta|\alpha, \sigma^2, \mathcal{D} \sim N\left(\frac{\tau_\beta^2 \hat{\beta} \sum_{i=1}^n d_i^2 + \sigma^2\mu_\beta}{\tau_\beta^2 \sum_{i=1}^n d_i^2 + \sigma^2}, \frac{\sigma^2\tau_\beta^2}{\tau_\beta^2 \sum_{i=1}^n d_i^2 + \sigma^2}\right) \quad (9)$$

Where the prior of  $\beta$  is assumed as  $\beta \sim N(\mu_\beta, \tau_\beta^2)$ . Regarding the posterior of  $\sigma^2$ , an inverse gamma distribution can be applied to the prior as the conjugate distribution of the variance for data following the normal distribution. The inverse gamma distribution is given by

$$IG\left(\frac{r_0}{2}, \frac{s_0}{2}\right) = \frac{\left(\frac{s_0}{2}\right)^{\frac{r_0}{2}}}{\Gamma\left(\frac{r_0}{2}\right)} (\sigma^2)^{-\frac{r_0}{2}+1} \exp\left[-\frac{s_0}{2\sigma^2}\right] \quad (10)$$

Where  $r_0/2 > 0$  and  $s_0/2 > 0$ . The full conditional posterior of  $\sigma^2$  given  $\alpha, \beta$ , and  $\mathcal{D}$  follows

$$p(\sigma^2|\alpha, \beta, \mathcal{D}) \propto p(\mathcal{D}|\alpha, \beta, \sigma^2)p(\sigma^2) \quad (11)$$

Then, the posterior is derived as

$$\sigma^2|\alpha, \beta, \mathcal{D} \sim IG\left(\frac{n+r_0}{2}, \frac{\sum_{i=1}^n (y_i - \alpha - \beta d_i)^2 + s_0}{2}\right) \quad (12)$$

because the likelihood function in terms of  $\sigma^2$  is

$$p(\mathcal{D}|\alpha, \beta, \sigma^2) \propto (\sigma^2)^{-\frac{n}{2}} \exp\left[-\frac{\sum_{i=1}^n (y_i - \alpha - \beta d_i)^2}{2\sigma^2}\right] \quad (13)$$

We apply the following Gibbs sampling method based on the derived full conditional posterior distribution for each parameter. First, we obtain  $\hat{\alpha}$  and  $\hat{\beta}$  from  $\mathcal{D}$  by the regression analysis. Then, we substitute  $\hat{\alpha}$ ,  $\hat{\beta}$ , and  $\sigma^2 = 1$  into  $\alpha^{(1)}$ ,  $\beta^{(1)}$ , and  $\sigma^{2(1)}$ , respectively. From the full conditional posterior distributions in Equations (8), (9), and (12), we generate  $\alpha^{(r)}$ ,  $\beta^{(r)}$ , and  $\sigma^{2(r)}$ , respectively. The parameter estimates of interest,  $a^{(r)}$  and  $\delta b^{(r)}$ , can be obtained by following  $-\alpha^{(r)}/\beta^{(r)}$  and  $-\beta^{(r)}$ , respectively. Repeating the procedure above as  $r \leftarrow r+1$ , we consequently obtain the samples and the simulation-based posterior distribution of each parameter.

#### 4. Numerical Example

We use the actual data as  $(n, y_n)$ ,  $(n = 1, 2, \dots, 25; y_{25} = 136)$  [4]. We generated  $r = 10,000$  samples for each parameter and reliability assessment measure by the simulation steps. The first 1,000 samples were discarded to avoid the burn-in samples.

For example, Figure 1 shows the MCMC samples and the posterior distributions of the parameters,  $a$  and  $\delta b$  in Equation (3), respectively. Figure 2 also show the posteriors for  $M_{25}$  in Equation (5) and  $R(25,1)$  in Equation (6), respectively. From these posterior distributions, the interval estimation can be obtained by following the notion of the credible interval. The  $100(1 - \gamma)\%$  credible interval, denoted by  $C$ , satisfies

$$\int_C p(\theta|\mathcal{D})d\theta = 1 - \gamma \quad (14)$$

Where  $p(\theta|\mathfrak{D})$  is the posterior. The  $100(1 - \gamma)\%$  HPD interval,  $C_{HPD}$ , is estimated as  $C_{HPD} = \{\theta \in \boldsymbol{\theta} \mid p(\theta|\mathfrak{D}) \geq k(\gamma)\}$ , where  $\boldsymbol{\theta}$  is the set of values of the parameter and  $k(\gamma)$  is the largest value satisfying

$$\int_{C_{HPD}} p(\theta|\mathfrak{D})d\theta = 1 - \gamma \quad (15)$$

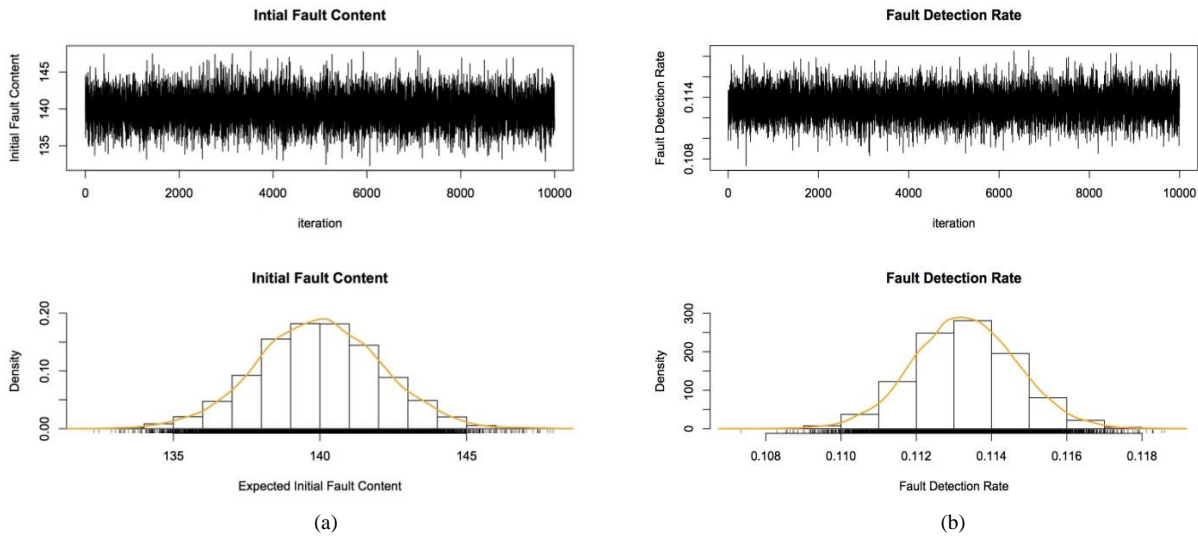


Figure 1. The MCMC samples and posterior distributions for (a) the parameter  $a$ , and (b) the parameter  $\delta b$

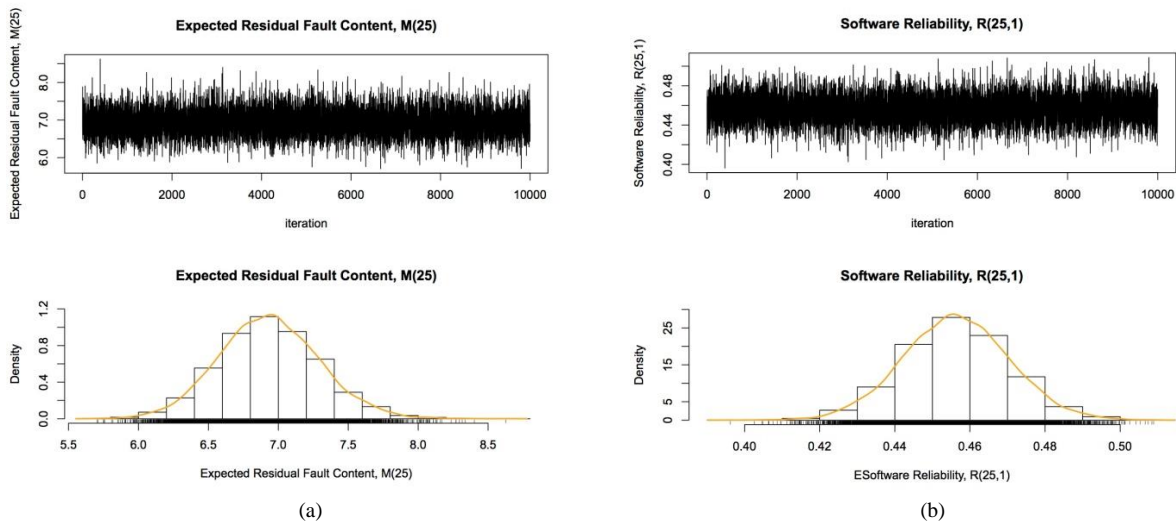


Figure 2. The MCMC samples and posterior distributions for (a)  $M_{25}$  and (b)  $R(25,1)$

Table 1 shows the results of interval estimations based on the 95% HPD interval for each parameter and assessment measure.

Table 1. 95% HPD intervals ( $\gamma = 0.05$ )

	HPD Interval	
	Lower	Upper
The parameter $a$	135.76	144.06
The parameter $\delta b$	0.1106	0.1159
$M_{25}$	6.244	7.642
$R(25,1)$	0.429	0.485

## 5. Conclusions

MCMC-based interval estimation for software reliability measurement was discussed in this paper. In addition, the derivation process for the posterior distribution was mentioned concretely. We demonstrated our approach for interval estimation by using actual software fault-count data. We still need to discuss how to set hyper-parameters in the prior distributions. In our further studies, we will apply our method to the interval estimation of optimal software release time in the near future.

## References

1. P. K. Kapur, H. Pham, and A. Gupta, "Software Reliability Assessment with OR Applications," Springer, London, 2011
2. J. D. Musa, D. Iannio, and K. Okumoto, "Software Reliability: Measurement, Prediction, Application," McGraw-Hill, New York, 1987
3. S. Osaki, "Applied Stochastic System Modeling," Springer-Verlag, Berlin, 1992
4. T. Kaneishi and T. Dohi, "Parametric Bootstrapping for Assessing Software Reliability Measures," in *Proceedings of the 17th IEEE Pacific Rim International Symposium on Dependable Computing (PRDC)*, pp. 1-9, Vancouver, California, British Columbia, Canada, December 2010
5. S. Inoue and S. Yamada, "Nonparametric Bootstrap Interval Estimation of Software Reliability and its Application to an Optimal Release Problem based on a Discretized Model," *International Journal of Reliability, Quality and Safety Engineering*, Vol. 20, No. 5, pp. 1350019 (14 pages), 2013
6. H. Okamura, M. Grottko, T. Dohi, and K. S. Trivedi, "Variational Bayesian Approach for Interval Estimation of NHPP-based Software Reliability Models," in *Proceedings of the International Conference on Dependable Systems and Networks 2007*, pp. 698-707, Edinburgh, UK, 2007
7. S. P. Brooks, "Markov Chain Monte-Carlo Method and its Application," *Journal of the Royal Statistical Society, Series D*, Vol. 47, No. 1, pp. 69-100, 1998
8. S. Inoue and S. Yamada, "Discrete Software Reliability Assessment with Discretized NHPP Models," *Computers and Mathematics with Applications: An International Journal*, Vol. 51, No. 2, pp. 161-170, 2006
9. S. Inoue and S. Yamada, "Generalized Discrete Software Reliability Modeling with Effect of Program Size," *IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans*, Vol. 37, No. 2, pp. 170-179, 2007
10. A. L. Goel and K. Okumoto, "Time-Dependent Error-Detection Rate Model for Software Reliability and other Performance Measures," *IEEE Transactions on Reliability*, Vol. R-28, No. 3, pp. 206-211, 1979

## Appendix

The derivation process in Equation (7):

$$p(\mathcal{D}|\alpha, \beta, \sigma^2) = \prod_{i=1}^n \frac{1}{\sqrt{2\pi\sigma^2}} \exp \left[ -\frac{(c_i - \alpha - \beta d_i)^2}{2\sigma^2} \right] = (2\pi\sigma^2)^{-\frac{n}{2}} \exp \left[ -\frac{(c_i - \alpha - \beta d_i)^2}{2\sigma^2} \right] \propto \exp \left[ -\frac{(c_i - \alpha - \beta d_i)^2}{2\sigma^2} \right] \quad (A1)$$

We define

$$S \equiv \sum_{i=1}^n (c_i - \alpha - \beta d_i)^2 \quad (A2)$$

in Equation (A1), and transform  $d_i \leftarrow (d_i - \frac{1}{n} \sum_{i=1}^n d_i)$ . Then,

$$\begin{aligned} S &= \sum_{i=1}^n \{ \hat{\alpha} - \alpha + (\hat{\beta} - \beta) d_i + (c_i - \hat{\alpha} - \hat{\beta} d_i) \}^2 \\ &= \sum_{i=1}^n (\hat{\alpha} - \alpha)^2 + \sum_{i=1}^n (\hat{\beta} - \beta)^2 d_i^2 + \sum_{i=1}^n (c_i - \hat{\alpha} - \hat{\beta} d_i)^2 + 2(\hat{\alpha} - \alpha)(\hat{\beta} - \beta) \sum_{i=1}^n d_i \\ &\quad + 2(\hat{\beta} - \beta) \sum_{i=1}^n d_i (c_i - \hat{\alpha} - \hat{\beta} d_i) + 2(\hat{\alpha} - \alpha) \sum_{i=1}^n (c_i - \hat{\alpha} - \hat{\beta} d_i) \\ &= n(\hat{\alpha} - \alpha)^2 + \sum_{i=1}^n (\hat{\beta} - \beta)^2 d_i^2 + \sum_{i=1}^n (c_i - \hat{\alpha} - \hat{\beta} d_i)^2 \end{aligned} \quad (A3)$$

Where

$$\begin{aligned}\sum_{i=1}^n d_i (c_i - \hat{\alpha} - \hat{\beta} d_i) &= 0 \\ \sum_{i=1}^n (c_i - \hat{\alpha} - \hat{\beta} d_i) &= 0\end{aligned}\quad (\text{A4})$$

respectively. From Equations (A1) and (A3),

$$p(\mathcal{D}|\alpha, \beta, \sigma^2) \propto \exp\left[-\frac{(c_i - \alpha - \beta d_i)^2}{2\sigma^2}\right] = \exp\left[-\frac{n(\alpha - \hat{\alpha})^2 + \sum_{i=1}^n (\beta - \hat{\beta})^2 d_i^2}{2\sigma^2}\right] \quad (\text{A5})$$

Regarding the derivation process of Eq. (8), the posterior is derived as

$$\begin{aligned}p(\alpha|\beta, \sigma^2, \mathcal{D}) &\propto p(\mathcal{D}|\alpha, \beta, \sigma^2)p(\alpha) \\ &\propto \exp\left[-\frac{n(\alpha - \hat{\alpha})^2}{2\sigma^2} - \frac{(\alpha - \mu_\alpha)^2}{2\tau_\alpha^2}\right] \\ &= \exp\left[-\frac{n\sigma^{-2}(\alpha - \hat{\alpha})^2 + \tau_\alpha^{-2}(\alpha - \mu_\alpha)^2}{2}\right]\end{aligned}\quad (\text{A6})$$

The numerator of Equation (A6) can be rewritten as

$$\begin{aligned}n\sigma^{-2}(\alpha - \hat{\alpha})^2 + \tau_\alpha^{-2}(\alpha - \mu_\alpha)^2 &= \alpha^2(n\sigma^{-2} + \tau_\alpha^{-2}) - 2(n\sigma^{-2}\hat{\alpha} + \tau_\alpha^{-2}\mu_\alpha)\alpha + n\sigma^{-2}\hat{\alpha}^2 + \tau_\alpha^{-2}\mu_\alpha^2 \\ &= (n\sigma^{-2} + \tau_\alpha^{-2})\left(\alpha - \frac{n\sigma^{-2}\hat{\alpha} + \tau_\alpha^{-2}\mu_\alpha}{n\sigma^{-2} + \tau_\alpha^{-2}}\right)^2 + \text{const}\end{aligned}\quad (\text{A7})$$

Then, substituting Equation (A7) into Equation (A6) yields

$$p(\alpha|\beta, \sigma^2, \mathcal{D}) \propto \exp\left[-\frac{n\sigma^{-2} + \tau_\alpha^{-2}}{2}\left(\alpha - \frac{n\sigma^{-2}\hat{\alpha} + \tau_\alpha^{-2}\mu_\alpha}{n\sigma^{-2} + \tau_\alpha^{-2}}\right)^2\right] \quad (\text{A8})$$

Equation (A8) means that the kernel of the full conditional posterior of the parameter  $\alpha$  is

$$N\left(\frac{n\hat{\alpha}\tau_\alpha^2 + \sigma^2\mu_\alpha}{\tau_\alpha^2 n + \sigma^2}, \frac{\sigma^2\tau_\alpha^2}{\tau_\alpha^2 n + \sigma^2}\right)$$

Then, we obtain Equation (8). Needless to say, we can obtain the full conditional posteriors of the other parameters by following the essentially same procedures above.