A Bayesian Inference Tool for NHPP-Based Software Reliability Assessment

Takumi Hirata, Hiroyuki Okamura, and Tadashi Dohi*

Department of Information Engineering, Graduate School of Engineering,
Hiroshima University, 1–4–1 Kagamiyama, Higashi-Hiroshima 739-8527, Japan
dohi@rel.hiroshima-u.ac.jp
http://www.rel.hiroshima-u.ac.jp/

Abstract. In this paper, we concern a sampling method for Markov chain Monte Carlo (MCMC) in estimating software reliability, and propose a unified MCMC algorithm based on the Metropolis-Hasting method regardless of model on data structures. The resulting MCMC algorithm is implemented as a Java-based tool. Using the Java-based Bayesian inference tool, we illustrate how to assess the software reliability in actual software development processes.

1 Introduction

Software reliability is one of the most significant attributes of software quality. During the last four decades, many software reliability models (SRMs) have been proposed. In particular, SRMs based on non-homogeneous Poisson processes (NHPPs) have gained much popularity for describing stochastic behavior of the number of failures experienced over time.

In general, the reliability assessment consists of three phases: (i) data collection, (ii) model fitting and evaluation, and (iii) model application. At the first phase, we observe and collect software metrics of a target software development project to estimate the software reliability. Software metrics can be classified to design and testing metrics. Typical examples of design metrics are lines of code, the number of branches, and code complexity. On the other hand, testing metrics indicate testing effort, the number of software reviews and the number of experienced software failures. When we wish to estimate quantitative software reliability, it requires software failure data, i.e., time-series data for the number of failures. At the model fitting and evaluation phase, the observed failure data are applied to statistical estimation of model parameters and model selection. At the third phase, a selected model can be employed to estimate several reliability measures such as software reliability function and the number of residual software faults. This paper mainly focuses on the second and third phases in the reliability assessment using NHPP-based SRMs, that is, our main issues are

* This research was supported by the Ministry of Education, Science, Sports and Culture, Grant-in-Aid for Young Scientists (B), Grant No. 21700060 (2009-2011) and Scientific Research (C), Grant No. 21510167 (2009-2012).

Y.-h. Lee et al. (Eds.): FGIT 2009, LNCS 5899, pp. 225–236, 2009.

statistical parameter estimation of NHPP-based SRMs, model selection criteria and computation of quantitative reliability measures.

Point estimation for parameters of NHPP-based SRMs models has been used in many of the past literature. Specifically, maximum likelihood (ML) estimation is commonly applied to compute point estimates for NHPP-based SRMs. Statistical estimates generally involve estimation errors. Although estimation errors of ML estimates can be reduced as observations available to estimation increase, we cannot obtain a sufficient number of observations to reduce the errors completely, because the number of software failures observed during software testing is limited. In other words, since software reliability estimation essentially involve the risk of large estimation errors, point estimation often fails to compute accurate software reliability measures.

One approach to estimate the software reliability is based on Bayesian statistics [21,10,16,17]. It requires prior knowledge about the model parameters in terms of a joint probability distribution. The prior information is then updated by using the likelihood of the observed data, generating a posterior distribution. Interval estimation, so-called credible intervals in the Bayesian context, can be performed based on quantile of this posterior distribution. However, posterior distributions which play a central role in Bayesian estimation are usually difficult to compute, except for a few cases. They cannot be derived as closed-form solutions. Hence approximate or numerical methods to compute posterior distributions are needed when we implement the Bayesian estimation.

Meinhold and Singpurwalla [10] presented an explicit form of the posterior distribution for Jelinski-Moranda SRM [6]. Yin and Trivedi [21] exhibited the direct numerical integral to Goel-Okumoto SRM [4] and delayed S-shaped SRM [18]. Recently, Okamura et al. [15,14] proposed variational Bayesian approaches for Goel-Okumoto SRM and delayed S-shaped SRM. Since the direct numerical integration and the variational approximation can be applied only in specific NHPP-based SRMs, they are not general methods for arbitrary NHPP-based SRMs.

The most popular and versatile approach would be Markov chain Monte Carlo (MCMC), which uses samples via pseudo-random variates instead of the analytical distribution. The MCMC has the advantage of applicability over many kinds of SRMs, but needs concrete sampling algorithms to evaluate posterior distributions. The concrete forms of MCMC should follow ad-hoc manners, namely, they have different forms with different types of SRMs. Kuo and Yang [7,8] proposed the MCMC approach to compute posterior distributions, where the resulting MCMC can be regarded as a versatile method by sampling from posterior distributions and can be effectively applied to NHPP-based SRMs. In fact, Kuo and Yang [7,8] gave the Gibbs algorithms incorporating Metropolis-Hasting (MH) methods to 8 kinds of NHPP-based SRMs.

This paper presents a unified MCMC sampling scheme based on the MH method, which is different from Kuo and Yang's Gibbs sampling [7,8]. The most remarkable difference from the existing work is to handle group data as well as time point data. Kuo and Yang [7,8] focused on only software failure time