REVUE FRANÇAISE D'AUTOMATIQUE, D'INFORMATIQUE ET DE RECHERCHE OPÉRATIONNELLE. RECHERCHE OPÉRATIONNELLE

SHIGERU YAMADA
HIROYUKI NARIHISA
HIROSHI OHTERA

Nonhomogeneous software error detection rate model : data analyses and applications

Revue française d'automatique, d'informatique et de recherche opérationnelle. Recherche opérationnelle, tome 20, nº 1 (1986), p. 51-60.

<http://www.numdam.org/item?id=RO_1986__20_1_51_0>

R.A.I.R.O. Recherche opérationnelle/Operations Research
(vol. 20, n° 1, février 1986, p. 51 à 60)

NONHOMOGENEOUS SOFTWARE ERROR DETECTION RATE MODEL: DATA ANALYSES AND APPLICATIONS (*)

by SHIGERU YAMADA (1) HIROYUKI NARIHISA (1) and HIROSHI OHTERA (1)

Abstract. — *A software reliability growth model called a nonhomogeneous error detection rate model is reviewed and applied to an actual data set of software failure occurrence time. In particular optimal software release policies with both software cost and software reliability requirements,* i. e. *cost-reliability optimal software release policies, are discussed for the model. Using the numerical results of the data analyses, the cost-reliability optimal software release policy is illustrated.*

Keywords: Software reliability, nonhomogeneous error detection rate model, data analysis, cost-reliability optimal, optimum software release time.

Résumé. — *Un modèle de croissance de la fiabilité des logiciels, appelé modèle de taux de détection des erreurs, est exposé et appliqué à des données réelles concernant les instants d'occurrence des pannes de logiciel. En particulier, des politiques optimales de mise à la disposition du public, incluant à la fois les exigences de coût et de fiabilité, sont discutées pour ce modèle. La politique optimale coût-fiabilité de mise à disposition est illustrée en utilisant les résultats numériques de l'analyse des données.*

1. INTRODUCTION

In recent years break-downs of a computer system due to software failures as well as hardware ones have remarkably increased in number. For this reason software reliability engineers and researchers have developed a number of software reliability models to describe software failure occurrence phenomena caused by software errors in a computer program. One of the useful models applied during software testing phase is a software reliability growth model (Ramamoorthy and Bastani [4]). In this area the model can estimate and predict software reliability indices in terms of the number of software errors detected during software testing phase. A simple and important software reliability growth model was proposed by Goel and Okumoto [1], based on a nonhomogeneous Poisson process (NHPP). This model stands on the assumption that the error detection rate per error, which is one of the software reliability indices characterizing software reliability growth, is a constant throughout the software testing period.

(*) Received in August 1984.

(1) Graduate School of Systems Science, Okayama University of Science, Ridai-cho 1-1, Okayama 700, Japan.

R.A.I.R.O. Recherche opérationnelle/Operations Research, 0399-0559/86/01 51 10/$ 3.00

However, the assumption of a constant error detection rate per error is not valid since the chance of detecting an error on a given test run is not constant. In fact the errors detected early in the testing are different from those detected later on. Yamada *et al.* [7] proposed a software reliability growth model based on an NHPP for such an error detection process, assuming that there are two types of errors: some are easily detected and the others more difficultly. This model is called a nonhomogeneous error detection rate model. That is, the error detection rate per error depends on the elapsed time of the software testing.

In this paper the nonhomogeneous error detection rate model proposed by Yamada *et al.* [7] is reviewed and applied to an actual data set of software failure occurrence time. Besides, cost-reliability optimal software release policies, i. e. optimal software release policies with both cost and reliability requirements, are discussed for the nonhomogeneous error detection rate model. The optimum software release times are determined both by minimizing on an expected total software cost and satisfying a software reliability requirement. Using the results of the software data analyses, the optimal software release policy is illustrated.

2. NONHOMOGENEOUS ERROR DETECTION RATE MODEL

Consider an implemented software system which is tested and is subject to software failures at random times caused by software errors present in the system. An error detection is assumed to mean a failure occurrence synonymously.

2.1. Model

In general the chance of detecting an error on a given test run is not constant throughout the testing period since the errors detected early in the testing are different from those detected later on from the viewpoint of detectability. Yamada *et al.* [7] proposed a nonhomogeneous error detection rate model, i. e. a software reliability growth model with two types of errors: Type 1 (Type 2) errors are easy (difficult) to be detected. The model stands on the following assumptions:

1. A failure is caused by an error.

2. Each time a failure occurs the error which caused it can be immediately removed.

3. A correction of detected errors does not introduce any new errors.

4. The initial error content proportion of Type i ($i=1, 2$) error is $p_i>0$ where $p_1+p_2=1$.

Let $\{N(t), t \geqq 0\}$ denote a counting process representing the cumulative number of errors detected up to testing time $t \geqq 0$. Then, the software reliability growth model for such an error detection process can be described by an NHPP as:

$$\Pr\{N(t)=n\}=\frac{\{m_p(t)\}^n}{n!}\exp[-m_p(t)] \qquad (n=0, 1, 2, \ldots), \tag{1}$$

$$m_p(t)=\sum_{i=1}^{2} m_i(t), \tag{2}$$

$$m_i(t)=p_i a(1-e^{-b_i t}), \tag{3}$$

$$a>0, 0<b_2<b_1<1, \tag{4}$$

where:

$a=$ the expected initial error content in a software system;

$b_i=$ the error detection rate per Type i error (per unit time) ($i=1$, 2).

The expected value of $N(t)$, $m_p(t)$, called a mean value function of the NHPP means the cumulative number of errors detected up to time $t \geqq 0$. The intensity function of the NHPP is given by:

$$\lambda_p(t) \equiv \frac{d}{dt} m_p(t)=a \sum_{i=1}^{2} p_i b_i e^{-b_i t}, \tag{5}$$

which means the instantaneous error detection rate per unit time.

It is of great use to investigate the software reliability growth aspect in terms of an error detection rate per error at an arbitrary testing time point. The error detection rate per error (per unit time) at time $t \geqq 0$ is given by:

$$d_p(t) \equiv \lambda_p(t)/(a-m_p(t))=\sum_{i=1}^{2}\left(\frac{p_i e^{-b_i t}}{p_1 e^{-b_1 t}+p_2 e^{-b_2 t}}\right) . b_i. \tag{6}$$

It is shown that $d_p(t)$ is a monotone decreasing function with $d_p(0)=\sum_{i=1}^{2} p_i b_i$ and $d_p(\infty)=b_2$. This implies that most of remaining errors in the late phase of testing are Type 2 errors which are difficult to be detected.

2.2. Software reliability measures

Let X_k denote a random variable representing the time-interval between $(k-1)$st and kth failures ($k=1, 2, \ldots$). Then, $S_k \equiv \sum_{i=1}^{k} X_i$ is a random variable representing the kth failure occurrence time. The joint probability density function of $\{S_1, S_2, \ldots, S_n\}$ is given by:

$$f_{S_1, S_2, \ldots, S_n}(s_1, s_2, \ldots, s_n) = \exp[-m_p(s_n)] \prod_{i=1}^{n} \lambda_p(s_i), \tag{7}$$

where $0 \leqq s_1 \leqq s_2 \leqq \ldots \leqq s_n < \infty$. The probability density function $f_{S_k}(t)$ of S_k can be obtained by the marginal density of S_k from (7):

$$f_{S_k}(t) = \frac{\lambda_p(t)\{m_p(t)\}^{k-1}}{\Gamma(k)} \exp[-m_p(t)] \qquad (k=1, 2, \ldots, n). \tag{8}$$

It should be noted that the cumulative distribution function of S_k is improper. Consequently, there does not exist the mean time-interval of failures.

Then, we consider the following quantitative measures which are useful for software reliability assessment. The expected number of errors remaining in a software system at testing time t is given by:

$$r_p(t) = a \sum_{i=1}^{2} p_i e^{-b_i t}. \tag{9}$$

The conditional probability that a software failure does not occur in $(s, s+x]$, given that the last occurrence time of a failure is s, is given by:

$$R_p(x \mid s) = \exp\left[-a \sum_{i=1}^{2} p_i \{e^{-b_i s} - e^{-b_i (s+x)}\}\right]. \tag{10}$$

The reliability function of (10) is called software reliability of the nonhomogeneous error detection rate model.

2.3. Estimation of parameters

Suppose that the data on n failure occurrence times s_k $(k=1, 2, \ldots, n)$ $(0 \leqq s_1 \leqq s_2 \leqq \ldots \leqq s_n)$ are observed during the testing phase. Then, the likelihood function for the unknown parameters a and b_i $(i=1, 2)$ in the NHPP model with $m_p(t)$ is given by (7). Taking the natural logarithm of the likelihood function, the maximum likelihood estimates $\hat{a}$ and $\hat{b}_i$ $(i=1, 2)$ can be obtained by solving the following likelihood equations under condition that $0 < b_2 < b_1$:

$$\frac{n}{a} = \sum_{i=1}^{2} p_i (1 - e^{-b_i s_n}), \tag{11}$$

$$a s_n e^{-b_j s_n} = \sum_{k=1}^{n} \frac{(e^{-b_j s_k} - b_j s_k e^{-b_j s_k})}{\sum_{i=1}^{2} p_i b_i e^{-b_i s_k}} \qquad (j=1, 2). \tag{12}$$

3. COST-RELIABILITY OPTIMAL SOFTWARE RELEASE POLICIES

Consider a software release problem for the nonhomogeneous error detection rate model defined by (1)-(4). This problem is often discussed to determine an appropriate time when the developer should stop software testing and deliver a software system to the user (e. g. Koch and Kubat [2] and Okumoto and Goel [3]).

The following parameters are introduced:

$c_{1\,i}$ = cost of fixing a Type i error during testing ($i=1, 2$);

$c_{2\,i}$ = cost of fixing a Type i error during operation:

$$(c_{2\,i} > c_{1\,i}) \qquad (i=1, 2),$$

c_3 = cost of testing per unit time;

T_{LC} = software life-cycle length;

T = software release time, i. e. total testing time.

Then, since $m_p(T)$ represents the expected total number of detected errors during $(0, T]$ the expected total software cost during testing and operation phases is given by:

$$C_p(T) = \sum_{i=1}^{2} c_{1\,i}\, m_i(T) + \sum_{i=1}^{2} c_{2\,i} \{ m_i(T_{LC}) - m_i(T) \} + c_3\, T. \tag{13}$$

It is of great interest to determine an optimum software release time satisfying both cost and reliability requirements (Yamada and Osaki [6]). Let us discuss optimal software release policies which minimize expected total software cost $C_p(T)$ of (13) subject to the condition that software reliability achieved by software testing, $R_p(x \mid T)$ of (10), is not less than some prespecified value, say R_0. Then, the optimal software release problem for given operation time $x \geqq 0$ can be formulated as follows:

$$\begin{aligned} &\text{Minimize} \quad C_p(T), \\ &\text{subject to} \quad R_p(x \mid T) \geqq R_0 \text{ and } T \geqq 0, \end{aligned} \tag{14}$$

where $0 < R_0 < 1$. Putting $d\, C_p(T)/dT = 0$ yields:

$$Q(T) \equiv \sum_{i=1}^{2} (c_{2\,i} - c_{1\,i})\, \lambda_i(T) = c_3. \tag{15}$$

From the monotonicity of $Q(T)$ and $R_p(x \mid T)$ in testing time $T \geqq 0$, it is easily shown that if $\sum_{i=1}^{2} (c_{2\,i} - c_{1\,i})\, p_i\, b_i > c_3/a$, there exists a unique solution T_0 to

(15), and if $R_p(x\,|\,0)<R_0$, there exists a unique solution T_1 to:

$$R_p(x\,|\,T)=R_0. \tag{16}$$

It is reasonable to assume that $T_{LC}>\max\{T_0, T_1\}$ since T_{LC} is the software life-cycle length. Then, we have the following theorem for the nonhomogeneous error detection rate model, which gives an optimum software release time $T=T^*$ satisfying (14), where $\lambda_i(t)\equiv dm_i(t)/dt$ $(i=1, 2)$.

THEOREM: *Suppose that* $c_{2\,i}>c_{1\,i}>0$ $(i=1,2)$, $c_3>0$, $x\geqq 0$ *and* $0<R_0<1$.

(1) *If* $\sum_{i=1}^{2}(c_{2\,i}-c_{1\,i})p_i b_i>c_3/a$ *and* $R_p(x\,|\,0)<R_0$, *then there exist a positive and unique* T_0 *and* T_1, *satisfying* (15) *and* (16), *respectively, and the optimum software release time is:*

$$T^*=\max\{T_0, T_1\}.$$

(2) *If* $\sum_{i=1}^{2}(c_{2\,i}-c_{1\,i})p_i b_i>c_3/a$ *and* $R_p(x\,|\,0)\geqq R_0$, *then* $T^*=T_0$.

(3) *If* $\sum_{i=1}^{2}(c_{2\,i}-c_{1\,i})p_i b_i\leqq c_3/a$ *and* $R_p(x\,|\,0)<R_0$, *then* $T^*=T_1$.

(4) *If* $\sum_{i=1}^{2}(c_{2\,i}-c_{1\,i})p_i b_i\leqq c_3/a$ *and* $R_p(x\,|\,0)\geqq R_0$, *then* $T^*=0$.

4. DATA ANALYSES AND APPLICATIONS

Let us apply the nonhomogeneous error detection rate model defined by (1)-(4) to an actual data set of software failure occurrence time observed during testing phase. The data set available in the form s_k $(k=1,2,\ldots,26)$ (days), which was cited by Goel and Okumoto [1], is analyzed here. When the numerical results of the data analyses are given, the cost-reliability optimal software release policy is illustrated.

Based on a Newton-Raphson method, solving (11) and (12) numerically yields

$$\begin{aligned}\hat{a}=36.16,\ \hat{b}_1&=0.59672\times 10^{-2},\\ \hat{b}_2&=0.96432\times 10^{-3},\end{aligned} \tag{17}$$

i. e.

$$\hat{m}_p(t)=36.16.[(0.9).(1-e^{-0.59672\times 10^{-2}t}) +(0.1).(1-e^{-0.96432\times 10^{-3}t})], \tag{18}$$

where the prespecified content proportions are assumed to be $p_1 = 0.9$ and $p_2 = 0.1$. The estimated mean value function $\hat{m}_p(t)$ is plotted in figure 1 along with the actual data and the 90% confidence bounds. By a Kolmogorov-Smirnov goodness-of-fit test (*see* Yamada and Osaki [5]) it was shown that the nonhomogeneous error detection rate model with (18) adequately fits the data set at a 5% level of significance. As a software reliability growth index, the estimated error detection rate per error of (6) is plotted in figure 2.

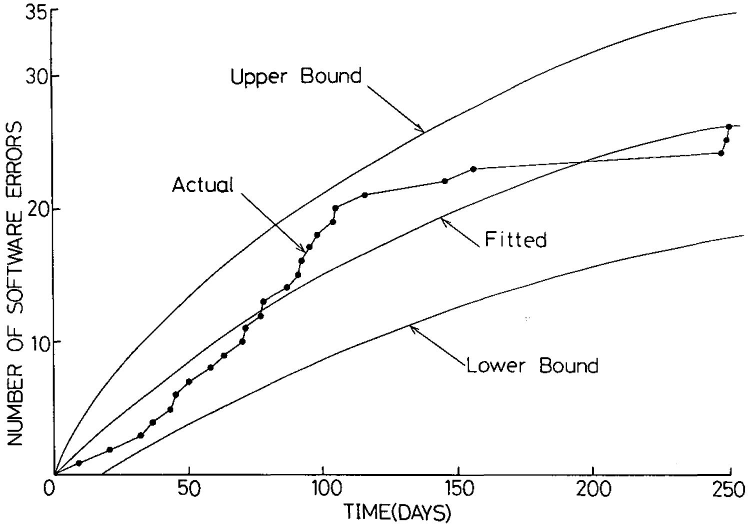

Figure 1. — Estimated mean value function $\hat{m}_p(t)$ and the 90% confidence bounds.

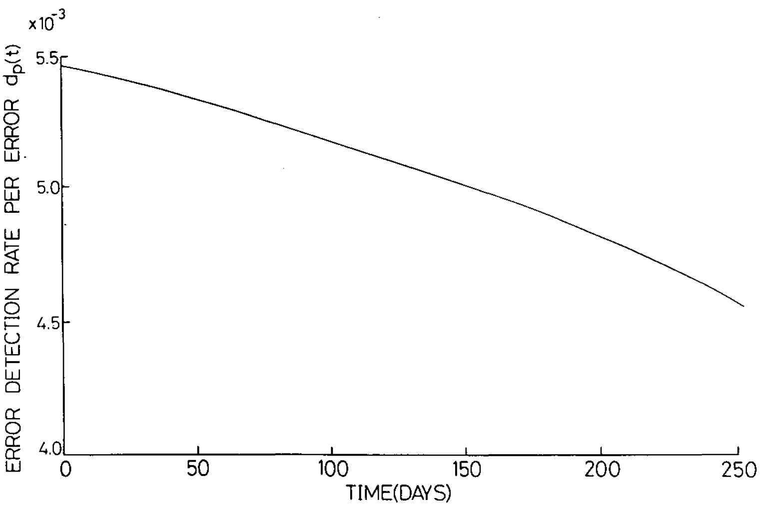

Figure 2. — Estimated error detection rate per error $\hat{d}_p(t)$.

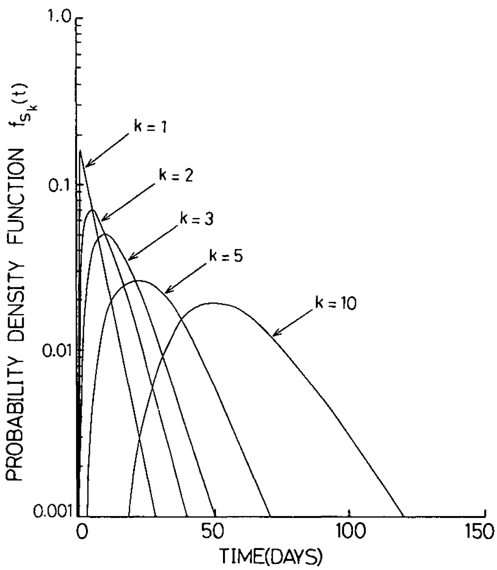

Figure 3. — Estimated probability density functions $\hat{f}_{S_k}(t)$ ($k=1-3$, $k=5$, and $k=10$).

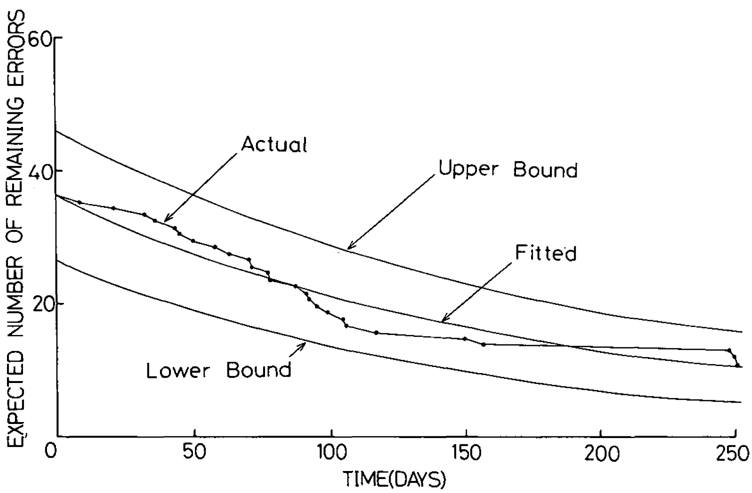

Figure 4. — Estimated expected number of remaining errors $\hat{r}_p(t)$.

From (8), using the estimated model parameters of (17), the estimated probability density functions of S_k ($k=1-3$, $k=5$, and $k=10$) are plotted in figure 3. As mentioned above, the estimated cumulative distribution functions of S_k ($k=1-3$, $k=5$, and $k=10$) become improper.

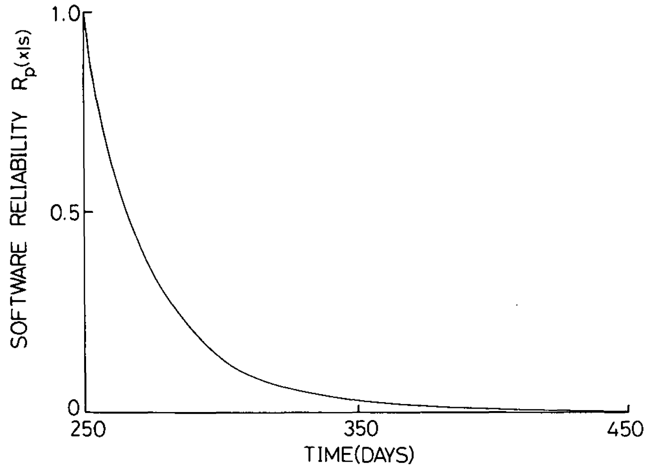

Figure 5. — Estimated software reliability $\hat{R}_p(x \mid s)$ $(s = 250)$.

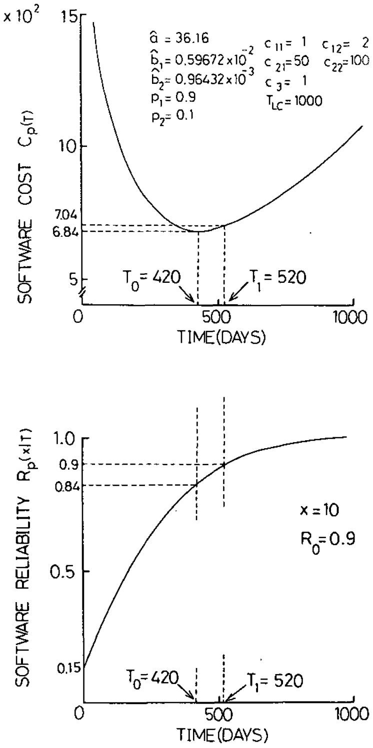

Figure 6. — An illustration of a cost-reliability optimal software release policy.

From (9) and (10) the maximum likelihood estimates of $r_p(t)$ and $R_p(x \mid s)$ can be obtained. The estimated $\hat{r}_p(t)$ and $\hat{R}_p(x \mid s)$ [$s=250$ (days)] are shown in figure 4 and figure 5, respectively.

Now, using the numerical results above, a cost-reliability optimal software release problem formulated by (14) is illustrated. Figure 6 shows the relationship between the estimated $C_p(T)$ of (13) for $c_{11}=1$, $c_{12}=2$, $c_{21}=50$, $c_{22}=100$, $c_3=1$, and $T_{LC}=1{,}000$, and the predicted $R_p(x \mid T)$ of (10) for $x=10$ and $R_0=0.9$. The testing time minimizing the estimated $C_p(T)$ is $T_0=420$ and the testing time satisfying the predicted $R_p(x \mid T)=0.9$ ($=R_0$) is $T_1=520$. From the theorem on the cost-reliability optimal software release policies, the optimum software release time is given by

$$\begin{aligned} T^* &= \max\{T_0, T_1\}, \\ &= \max\{420, 520\}; \\ &= 520, \end{aligned} \tag{19}$$

since:

$$\sum_{i=1}^{2} (c_{2\,i}-c_{1\,i}) p_i b_i > c_3/a \qquad \text{and} \qquad R_p(x \mid 0) < R_0.$$

ACKNOWLEDGMENTS

The authors would like to thank Professor Shunji Osaki of Hiroshima University for the encouragement throughout this study.

REFERENCES

1. A. L. Goel and K. Okumoto, *Time-Dependent Error-Detection Rate Model for Software Reliability and Other Performance Measures*, I.E.E.E. Trans. Reliab., Vol. R-28, 1979, p. 206-211.
2. H. S. Koch and P. Kubat, *Optimal Release Time of Computer Software*, I.E.E.E. Trans. Software Eng., Vol. SE-9, 1983, p. 323-327.
3. K. Okumoto and A. L. Goel, *Optimum Release Time for Software Systems Based on Reliability and Cost Criteria*, J. Syst. Software, Vol. 1, 1980, p. 315-318.
4. C. V. Ramamoorthy and F. B. Bastani, *Software Reliability-Status and Perspectives*, I.E.E.E. Trans. Software Eng., Vol. SE-8, 1982, p. 354-371.
5. S. Yamada and S. Osaki, *Reliability Growth Models for Hardware and Software Systems Based on Nonhomogeneous Poisson Processes: A Survey*, Microelectron. Reliab., Vol. 23, 1983, p. 91-112.
6. S. Yamada and S. Osaki, *On a Cost-Reliability Optimal Software Release Problem*, Abstracts presented in the Autumn Meeting of Operations Res. Soc. Japan, 1984.
7. S. Yamada, S. Osaki and H. Narihisa, *A Software Reliability Growth Model with Two Types of Errors*, RAIRO-Operations Research, Vol. 19, 1984, pp. 87-104.