THE ASTRINGENCY OF THE GP ALGORITHM FOR FORECASTING SOFTWARE FAILURE DATA SERIES

Yong-qiang Zhang* and Hua-shan Chen

**The Information and Electricity-Engineering Institute, Hebei University of Engineering, Handan 056038, China*

Email: YQZHANG@hebeu.edu.cn

ABSTRACT

The forecasting of software failure data series by Genetic Programming (GP) can be realized without any assumptions before modeling. This discovery has transformed traditional statistical modeling methods as well as improved consistency for model applicability. The individuals' different characteristics during the evolution of generations, which are randomly changeable, are treated as Markov random processes. This paper also proposes that a GP algorithm with "optimal individuals reserved strategy" is the best solution to this problem, and therefore the adaptive individuals finally will be evolved. This will allow practical applications in software reliability modeling analysis and forecasting for failure behaviors. Moreover it can verify the feasibility and availability of the GP algorithm, which is applied to software failure data series forecasting on a theoretical basis. The results show that the GP algorithm is the best solution for software failure behaviors in a variety of disciplines.

Keywords: GP, Forecasting for failure data, Optimal individuals reserved strategy, Markov random processes, Astringency

1 INTRODUCTION

Because of the increasingly broad application and importance of software, the quality that people request in software is becoming higher and higher. The appraisal and prediction of software's reliability, as a significant characteristic for weighing software qualities, has been an emphasis that people focus on and study actively. Models for predicting software reliability are the kernels of this research.

During the process of software reliability research, we have already created many different reliability models. When applying them for reliability prediction, we always face many problems, such as which model should we select or if the prediction result is credible. As the capabilities for models are difficult to identify, operators seldom being familiar with every model, tend to select the models they need blindly. Meanwhile, there are numerous inconsistencies in software reliability prediction. For example, different models will get different prediction results for the same software system. The prediction quality of the same model for predicting different data may make a large difference. The same model for different phases of the same software system may yield very different predicting qualities. When using one or several models to predict, they will all have low prediction qualities. A discussion of these problems can be found in Cong, Lu, and Bai (2000) and Wang and Jin (2002).

In order to solve the problems above, this paper makes use of the GP algorithm for forecasting software failure data. It then analyzes and verifies that the GP algorithm with an “optimal individual reserved strategy” can be converged, which indicates it is possible to get the best solution for software failure. This approach can create specific programming aimed at generating specific software failure data and then obtain an approximate solution or the best one. This approach removes subjective assumptions of statistical methods. It can improve the consistency of model application and the analysis of software reliability models, resulting in better forecasting for software failure behaviors.

2 SUMMARY OF THE GP ALGORITHM

Genetic Programming is a technique based on biological evolution, which is developed from the Genetic Algorithm (GA). Here, the depth of the tree is defined as not more than N and is a given positive integer. T_i are the individual trees, and each tree’s root nodes $r_k \in F$; their leaf nodes are defined as $l_j \in T$, so $S = \{T_i \mid r_k \in F, l_j \in T\}$, in which F is the function set and T is the terminal set. We also have to define the fitness function $f : S \to R$, which can search for the best individuals T_i^* in the space S : $f(T_i^*) = f^* = \max\{f(T_i) : T_r \quad T_r \in S\}$.

The approach of creating trees randomly by a growth method, whose depth cannot exceed the maximum depth D, is defined as follows: The root nodes are selected from the function sets in order to generate non-ordinary individuals, while the other nodes are selected from $T \cup F$ if the depth of nodes to be selected is less than D, and if the depth equals D, they are selected from T (Lin, Li, & Kou, 2000). We also adopt the “best-reserved strategy” to make the best individuals reserved in the next generation, so that these individuals do not attend the genetic operations, which are shown in detail in section 4.1

.

3 FORECASTING FOR SOFTWARE’S NEXT FAILURE TIME BY THE GP ALGORITHM

As we know, traditional software reliability models are all based on different statistical assumptions that constrain both the modes of software failure behaviors for specific models and the suitable range of each model. The randomness of software reliability models’ selection has hindered the application of software reliability models. This paper reports results from adopting the GP algorithm to model failure data series. Examples of accumulative time series (or the next failure time series) of DACS failure data set SYS1 and SYS2, from Musa in 1979, are used to model and test the feasibility and availability of GP. The parameters for GP programming are given in Table 1.

Table1. Parameters for GP programming

Parameters	span solution	Parameters	span solution
Function set(F)	+、-、×、/、log、sin、exp、cos、tan、sqrt	Selection methods	tournament and so on
Terminal set(T)	argument x, constants	Terminal condition	100 generations
Population size	30	Number of generation	100
The probability of crossover	0.90	Maximum of the tree depth after crossover	7

The probability of mutation	0.05	Maximum of the tree depth after mutation	7
The way to generate initial population	total and grow	Maximum of the initial tree depth	5

Under the programming environment of MATLAB6.5, we get a better fitness model structure after evolving for 50 generations which is shown as follows:

$$\text{SYS1: } f(x) = 3.823 \times x \times \ln\left(\left(x + \ln(x)\right) \times e^{x}\right) \qquad \text{Eq.(1)}$$

$$\text{SYS2: } f(x) = 13.1874 \times x \times \ln\left(x \times e^{x}\right) \qquad \text{Eq.(2)}$$

where x stands for the numbers for software failures and $f(x)$ are the results of the failure data model. Figures 1 and 2 give the transformation curve (model simulation output) for models and the true data series.

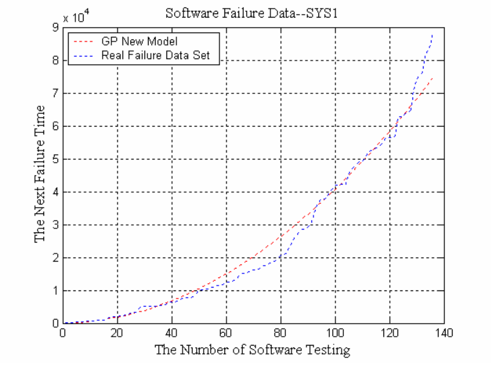

Figure1. Simulation result for SYS1

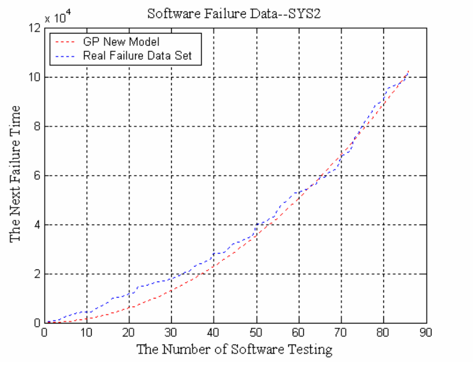

Figure 2. Simulation result for SYS2

From the figures above, we can conclude that the GP model has better applicability (accuracy) in comparison with other models, as well as a better goodness of fit. Furthermore, SYS3 provided by the Musa record set has also been created by automatic programming as well as the test example (from one to sixteen data series) provided by the Armed Forces Engineering Institute and the error statistic data provided by the Naval Tactical Data System's (NTDS) development and testing procedure (from one to thirty data series) of the U.S. Navy's Fleet Computer Programming Center. These models can get optimal results matching with the corresponding dataset that better fit and forecast. Our tests show that GP can create automatic programming of the specified error data without any assumptions being added during modeling. The feasibility and efficiency of using the GP algorithm for the evolutionary modeling of the Software Failure Data Series are to some extent further indicated.

4 ASTRINGENCY ANALYSIS OF GP

4.1 Affect of optimal reserve strategy on astringency

To illustrate the necessity of the Optimal Reserve Strategy, first the structure is analyzed. If the colonies in each generation created during GP algorithm are treated as one state, a random process taking the place of the entire evolutionary process can be considered and analyzed as a Markov chain for astringency. Further the concept of a "realized history" is above all possible.

Consider the case of a strategy $\pi = \{\pi_0, \pi_1, \cdots\} \in \prod$, for any one history $h_t = \{i_0, a_0, i_1, a_1 \cdots\} \in H_t$. If this condition, $P_\pi\{h_t \mid i_0\} > 0$, is satisfied, that is to say, the probability of event h_t is positive under the measure of strategy π, then h_t is called the realized history of strategy π. In other words, on the condition of strategy π, a state i_0 is diverted to state i_1 after the experienced behavior a_0. The procedure keeps going until state i_t is reached at the instant of t if the experienced behavior a_1 is adopted. If the probability of the whole event inducted by strategy π is not zero, the history is called "realized history" under the strategy π.

The optimal activity set is given as follows. For each state $i \in S$, the following equation is defined as the practicable optimal activities of the state i.

$$A^*(i) \equiv \underset{a \in A(i)}{\arg} \max \left\{ r(i, a) + \beta \sum_{j \in S} p(j \mid i, a) V_\beta{}^*(j) \right\} \qquad \text{Eq. (3)}$$

Theorem of Optimal Strategy:

The necessary and sufficient condition of strategy $\pi = \{\pi_0, \pi_1, \cdots\} \in \prod$ as an optimal strategy is: for $\forall t \geq 0$, if history $h_t = \{i_0, a_0, i_1, a_1 \cdots\} \in H_t$ is a realized history of strategy π, then the equation $\pi_t(a \mid h_t) = 0$ can be satisfied on condition that $a \in A(i_t) - A^*(i_t)$ (Liu, 2004).The proof of this theorem can be found in Dong and Liu (1986). The significance lies in the following: a strategy is optimal only when each decision rule has to make use of the optimal behaviors of every realized history.

From the theorem above, GP reserves the optimal individual of every generation for the next generation, which can be expressed as: $A(t+1) = \max\{A(t), A_{best}\}$, where A_{best} is the best individual of the $t+1$ generation and $A(t)$ is the optimal one of the t generation. Therefore the GP joined with optimal strategy is still a homogeneous Markov chain. In other words, the probability of going from any state to a state that includes the optimal solution is greater than zero, but it is zero on the contrary. Therefore GP with optimal strategy has the ability of a non-holonomic ergodic process and always can be convergent to the optimal solution with a probability of 1.

4.2 Analysis of GP constringency by Markov Chain

The concept of homogeneous time is given first. As we know, the visualized significance of the transition probability $p_{ij}(m, n)$ is the probability of transferring a state from i to j, considering time from m to n. As there are $n - m$ time units, or $n - m$ steps from m to n, $p_{ij}(m, n)$ is defined as the transition probability for $n - m$ steps.

If $p_{ij}(m, n + m)$ has no relation with m and $m, n \geq 0$, in other words, no matter when state i in the system starts, the probability is identical when transferred to state j after n steps. And now the Markov chain $\{X_n, n \geq 0\}$ is time homogeneous, which can be expressed as $p_{ij}(m, n+m) = p_{ij}(n)$. A one-step transition probability is presented by p_{ij}. For a Markov chain that is time homogeneous $\{X_n, n \geq 0\}$, the equation $p_{ij}(n - m) = P(X_n = j \mid X_m = i)$ is satisfied. Especially when $m = 0$, the equation $p_{ij}(n) = P(X_n = j \mid X_0 = i)$ can be satisfied (Zhao & Zhu, 1993).

As standard GA, a Markov chain of the standard GP algorithm is time homogeneous, which can be expressed as:

$$P_{ij}(m, m+1) = P(X(m+1) = j \mid X(m) = i) = P_{ij},$$

where $i, j \in I$ are states while m is the initial time. The initial distribution of population can be random because the initial distribution has no effect on the limited behavior of a Markov chain.

Astringent Theorem of Markov:

The probability of GP converging to optima is less than 1. This can be shown as follows: The probable states of a colony can be divided into two types, one is I_0 , which includes optimal individuals and the other is I_n , which does not include optimal individuals. The result $I = I_0 \bigcup I_n \quad I_0 \bigcap I_n = \Phi$ can be satisfied. The stable probability of P_1 transferred to state I_0 is less than 1, which can be proved by contradiction. Assume that the probability for which GP astringent evolves to optima is 1, that is to say, the probability of evolving to state I_n is zero, and $\lim_{t \to \infty} P\{P_t \in I_n\} = 0$ can be satisfied. During the evolution process of GP, the state of the colony transferred to $j \in I$ from $i \in I$ by duplication, crossover, and mutation is described by the transition probability of genetic operators s_{ij}, c_{ij}, m_{ij} , respectively. Then the random matrices can be structured separately as $S = \{s_{ij}\}, C = \{c_{ij}\}, M = \{m_{ij}\}$, and the transferred matrix of the colony states GP is $R = SCM = \{r_{ij}\}$.

As matrices S, C, M are all random matrices, and $m_{ij} = P_m^{H(i,j)}(1 - P_m)^{1-H(i,j)} > 0$ (H_{ij} is the Hamming distance between i and j), the inequality $r_{ij} > 0$ can be easily proven. In other words, the matrix R is positive definite. At time t , the probability of a colony under state j is $P_j(t) = \sum_{i \in I} P_i(0) \cdot r_{ij}^t$, $(t = 0,1,2,\cdots)$.

According to the property of a homogeneous Markov chain, the stable probability distribution is independent of the initial probability distribution, or $P_j(\infty) = P_i(\infty) r_{ij} > 0$, while $j \in I$. That is to say, j is a state of I_n so $\lim_{t \to \infty} P\{P_t \in I_n\} > 0$, which is inconsistent with the assumption above and the theorem comes into existence.

Therefore, the fact that standard GP can be astringent on the condition that the optimal reserve is adapted has been proved. Otherwise, astringency is not certain.

4.3 Summary

Assume that the evolutional colony of the k generation is $X_k = \{x_1^k, x_2^k, \cdots, x_n^k\}$ during the implementation of GP, where *n* is the size of colony, and $f_k = \max\{f(x_i^k) \quad i = 1,2,\cdots,n\}$ is the maximum fitness of the individuals in current generation. When GP is used in practice, the optimal strategy should be used, which selects optimal individuals in the initial generation and reserves them in the colony as individuals for $n+1$, which means they never participate in the evolutional operation from initial generation to the next.

Similarly, for the inherited generation k , the optimal individual selected out from generation k can be compared with optimal individuals from the previous generation. Then the better one can be added into the colony as the optimal individual of the current generation, which can be defined as $\mathrm{x}_{\mathrm{n+1}}^{\mathrm{k}}$ for the $n+1$ individual. However, it never participates in any evolutional operation. Then we will get the equality $f_k = f(x_{n+1}^k)$ when the optimal reserved strategy is used.

Without the optimal reserved strategy, consider the nontrivial case if f is not a constant value. Assume that $m = \min\left\{f^* - f\left(T_i\right): T_i \in S \setminus T^*\right\}$.The result is that m exists and m >0 when S is finite. It can be obtained that $\mathrm{P}\{\mathrm{D_k} < \varepsilon\} = \mathrm{P}\{\mathrm{D_k} > 0\} = \mathrm{P}$ (No individual in the generation k, which belongs to X^*) on condition that $0 < \varepsilon < m$. During the process of variation from generation $(\mathrm{k} - 1)$ to k, the probability for every individual of the mutated individual not in the optimal set satisfies the equation $\mathrm{P_{not}} \geq \mathrm{D} > 0$, where $\mathrm{P_{not}}$ is the probability and D is a fixed constant. So the probability that all individuals in generation k do not belong to X^* is greater than or equal to $(\mathrm{P_m} \cdot \mathrm{D})^{\mathrm{n}}$, that is $P\{D_k < \varepsilon\} \geq (P_m \cdot D)^n$. Therefore we can obtain the inequality $P\{D_k \leq \varepsilon\} \leq 1 - (P_m \cdot D)^n < 1$. As D is a constant independent of k and $\forall k \geq 1$ is always correct, GP is not convergent at this time. In all, the astringency of GP is impossibly promised if the mutation operator in any form is not applied.

Therefore, no matter how large the population, it is finite. Sampling errors of the genetic operation are inevitable, which may make certain elements in F and T needed by individuals in X^* disappear from the colony after a number of stages. Even these elements may never have a chance to participate in the colony without mutation. Therefore they have no possibility of getting an optimal solution. In a word, GP is not likely to be astringent unless certain mutation forms or the optimal reserved strategy is utilized (Jia, Kang, & Chen, 2003).

5 CONCLUSION

Forecasting of software failure data by Genetic Programming has removed some of the subjective assumptions of statistical models and adds consistency in the application of the models. This makes sense in a practical application for the analysis of software reliability models and forecasting software failure behaviors. This paper treats different states during individuals' evolution in GP as Markov random processes and shows that it will converge to the best solution if the "best-individuals reserved strategy" is used, which can consequently evolve to better individuals. It is proved that the GP algorithm is able to obtain a better solution and may probably be feasible and available for practical applications. However, the influence of generation size as well as the setup for genetic operations on the constringent speed (or the speed that best solutions achieve) may be reduced, and the time for computing may be too long, which involves the efficiency of problem solving. All of these issues should be studied further. In other words, the elements related to the constringent speed should be improved to be fitter for modeling and forecasting time series problems accordingly.

6 ACKNOWLEDGEMENTS

Project 60573088 supported by National Natural Science Foundation of China (NSFC);

Project 603407 supported by National Natural Science Foundation of Hebei Province.

7 REFERENCES

Bai, Y., Cong M., & Lu, M. (2000) Research and realization of software reliability predicting approach. Transaction of *Beijing Aviation and Spaceflight University*.

Chen, Y., Jia, J., & Kang, L. (2003) The astringency analysis and improvement of evolvement algorithm. *Computer Engineering and Applications 19*, pp.91-92.

Dong, Z. & Liu, K. (1986) Structure of best-strategy by semi-Markov decision processes with unbounded reward models. *Mathematical Research and Comment 6(3),* pp.125-134.

Dong, Z. & Liu, K. (1985) Structure of best-strategy by reward models. *Chinese Science* A (11), pp.975-985.

Jin, M. & Wang H. (2002) *Research on the software reliability models based on data mining technology*, PhD thesis, Beijing Aviation and Spaceflight University. Beijing, China.

Musa, J. (1980) Software Reliability Data. *DACS*, RADC, New York.

Kou, J., Lin, D., & Li, M. (2000) A theorem about convergence of GP. *Transaction of Xiamen University (The Natural Science Edition 1).*

Liu, K. (2004) *Applied Markov Decision Processes*. Tsing Hua University Press.

Zhao, D. & Zhu, Y. (1993) *Applied Random Processes* [M]. Mechanical and Industrial Press.