International Journal of Computer and Electrical Engineering, Vol. 2, No. 2, April, 2010

1793-8163

A Mathematical Realization of software analysis

P.Chakrabarti, *Member, IACSIT* and R.Supakar

Abstract **– In this paper we have pointed at several approaches in regarding the various aspects of software engineering. We have given stress on reliability, functionality, upgradation in terms of mathematical realization. In this paper we have also tried to find out optimum throughput using the concept of pattern matching and decision making . Another approach is given in the light of statistical based gain estimation theory.**

Index terms **- software engineering, reliability, functionality, upgradation , decision making , throughput, pattern matching, estimation theory**

I. Introduction

Software engineering comprises interrelated and recurring entities, which are essential for software development. Software is developed efficiently and effectively with the help of well defined activities or the desired output. A process can also be defined as a collection of procedures to develop a software product according to certain goals or standards. Reliability is the manner in which the process is designed so that errors in the process are avoided or trapped before they result in errors in the product. We can realize the various processes based on error estimation. Functionality refers to the degree of performance of the software against its intended purpose. There is need of artificial intelligence based decision making for controlling error in an efficient manner maintaining gain in time complexity.

II. Statistical Based Estimation Theory

Let software reliability is measured in terms of parameter x. We assume that number of observations be n.The observations are independent of density function is denoted by $f(p_1 , x)$ $f(p_2 , x)$ ………..$f(p_n , x)$. where x is the estimate of likelihood towards reliability optimum value. For maximum achievement solution of n in terms of p from the following equation.

$1/f(p_1 , x)\, d\, f(p_1 , x)/dx + \ldots\ldots + 1/f(p_n , x)\, d\, f(p_n , x)/dx = 0$

Manuscript received July 31 . 2009.

Dr.P.Chakrabarti is with Computer Science and Engineering department , Sir Padampat Singhania University, Udaipur , Rajasthan , India (email- prasun9999@rediffmail.com)

R.Supakar is with Computer Science and Engineering department , Bengal Institute of Technology and Management , Bolpur-731236 , West Bengal, India (email- rsupakar@rgmail..com)

III. Reliability

A. Concept

Let the software dependent process be P1,P2……….Pn. The applications by involving the process be A1,A2,…………..An show the reliability of condition assuming that some of the process are optimum among the set of all the set of all the processes. We assume the following applications are dependent on the respective processes. We further assume that out of the process P1, P2 and P5. P1 and P2 jointly can perform A1.Similarly for the subsequent application P3, P4 can jointly perform A2.For A3, P6, P7 can jointly perform A3. We further assume that the priority of joint operation of P1, P2 is greater than that of P5, hence the process allocation for operation of A1, will deal the processes P1, P3 rather than P5. The efficiency and throughput have been taken into consideration in the particular case. It is also seen from the above table that probability of co existence criteria of (P1, P5) = 1 .

Applications	Dependant Processes
A_1	P_1 , P_2 , P_5
A_2	P_3 , P_4
A_3	P_6 , P_7

B. Estimation in series form

Collorary

Let, in a given set there are n_1 and n_2 reability factors , and G_1,G_2 are their respective geometric means,then the geometric mean(G) of the combined set of reability measure is given by $G=(G_1^{n1} \cdot G_2^{n2})^{(1/(n_1+n_2))}$

Proof

Let, $r_{11},r_{12},\ldots\ldots r_{1n1}$ be the values of reability factor of the first software ,while $r_{21},r_{22},\ldots\ldots,r_{2n2}$ be those of second one. Then,

$G_1=(r_{11}.r_{12}. \ldots\ldots .r_{1n1})$ or

$r_{11}.r_{12} . \ldots\ldots . r_{1n1} =G_1^{n1}$

In the similar way, $r_{21}.r_{22} . \ldots\ldots . r_{2n2} =G_2^{n2}$

Hence the product of (n1+n2) values in the combined set is equal to $G_1^{n1} \cdot G_2^{n2}$. It's therefore,evident that the geometric mean (G) of the combined set is given by, $G=(G_1^{n1} . G_2^{n2})^{(1/n_1+n_2)}$

C. Estimation over time in exponential form

If reability factor (r) exchanges over time (t) exponentially, then the value of the variable at the mid-point of an interval (t_1,t_2) i.e at $(t_1+t_2)/2$ is the geometric mean of it's values at t_1 and t_2.

Suppose $rt=ab^{t}_{2}$

Also,the value of r at $(t_1+t_2)/2$ is,

$$ab^{(t_1+t_2)/2} = \{a^2b(t_1+t_2)\}^{1/2}$$
$$= \{(ab^{t}_{1})(ab^{t}_{2})\}^{1/2}$$
$$= (r_{t1}.r_{t2})$$

D. Estimation at specific instant of time

Suppose the reliability distribution f(r; ɵ) of a random variable r involves a parameter ɵ whose value is unknown, and we want to make a guess about (or to estimate) the value of ɵ on the basis of a random sample$(r_1,r_2,r_3,\ldots,r_n)$ of size n drawn on from the various softwares. For this,we use a suitably chosen statistic $t=g(r_1,r_2,\ldots,r_n)$, i.e. a function of the sample observations.It's a random variable whose value is completely determined by the sample values.Now, for a given sample $(r_1,r_2,\ldots,r_n)$,the value of 't' is $g(r_1,r_2,\ldots,r_n)$;it's simply a number and it's taken as the guess for the value of ɵ.The statistic 't' is called an *estimator* of ɵ,while the value of 't' is obtained from a given sample is called an *estimate* of ɵ.Naturally, for 't' to be a satisfactory estimator of ɵ, the difference |t - ɵ | should be as small as possible.E(t) = ɵ, whatever the true value of ɵ may bei.e. $Var(t)\leq var(t^1)$,whatever the true value of ɵ may be,where t^1 is any other estimator This reveals minimum deviation towards achieving optimum reliability.

IV. Functionality

A. Based on time

From the functionality point of view, the main focus is on proper scheduling of processes towards achieving a particular task.In the particular scenario, we have to find out the gain in time complexity of the processes using single/iterative mode, parallel mode and pipeline mode.Using single / iterative mode time required will be T+T+T = 3T where, T= time required by each application.In parallel mode it will beT as such application takes identical time as per our assumption. In pipeline mode the time complexity will be based on (T+N-1) where T is the time taken for a particular task , N is the number of applications. Hence in this the particular case T+3-1 = T+2

Hence in this case the time required for performing the applications in non parallel mode is identical to that performed in pipeline mode when T=1 is unit of time.We have to calculate the time complexity based on the nature of application and accordingly apply the mode of processing.The nature of application basically means the dependency criteria which governs whether the parallel processing is effective or not. For example if there are three processes A1, A2 and A3, where the processing of A2 depends on the completion of A1, then A2 is dependent, while A3 isan independent process. In this case let there be two time periods T1 and T2 and during T1, A1 and A3 will perform and for T2 , A2 will perform i.e. here the parallel processing of A1, A2 is not possible , while that of A3 is possible as it is independent. It can be explained as

Time Taken	Applications
T_1	A_1 , A_3
T_2	A_2

We can find out the relation and accordingly adjust the timing sequence of application.

Applications	Dependency Criteria		
	A_1	A_2	A_3
A_1	X	0	0
A_2	1	X	0
A_3	0	0	X

From the above table, we can predict that the dependency criteria is governed based on the element A1 of the above matrix. Hence the application A1 and A3 can be avoided in parallel form in T1. In the next timing sequence T2, A2 is performed. Hence if T1= T2 = T then total time = 2T.

B. Based on embedded system based software

ASIC form: In case of application oriented base programming, the architecture is basically involving high space complexity. Thereby the cost is more. However, the speed in case of ASIC is much faster, since there is no necessity of dynamic programming in this particular case.

For reduced space the FPGA is used but there is a limitation as regards the time complexity since the bit streams downloaded consumes some time in this particular case. There is another growing model which is a hybrid one and it is known as SOC (system on chip). In this particular model,there is dual core programming, where a particular partition is dedicated for ASIC and another one for FPGA. Hence there is a control signal which will regulate the functional activity of these two partitions. If it is 0/1 then activated partitions is ASIC/FPGA (as per our assumption) and if C=X, then power is in standby mode.

For embedded system based software program RTOS (Real Time O.S.) is applicable. In this particular case, a timing deadline is specified. The following two conditions may occur:

If the task is not completed within the timing deadline and then also the system doesn't crash, the particular case is known as Soft Real Time O.S.

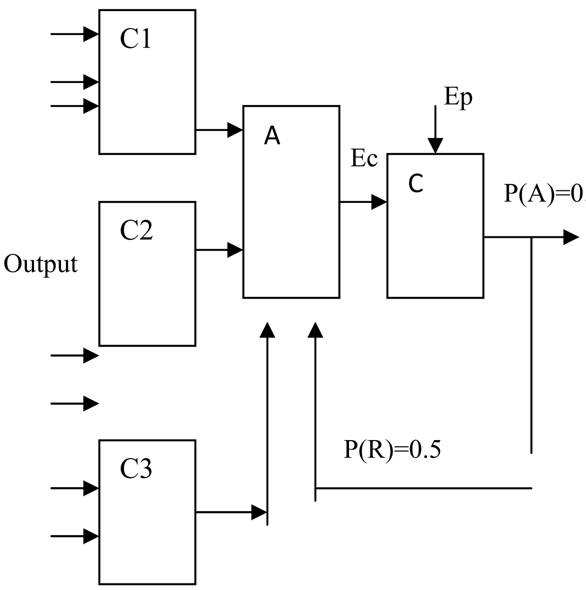
If the deadline is missed and the system completely fails, then it is known as Hard Real Time O.S.

The application of this is in the control of motion of a projectile (Rocket/Missile) where each unit is completed using NANO programming environment.

V. Decision Making

Based on the feasibility study, the decision of progress of a software model developing is made. If the cost, time and the demand of the customer are in an optimum manner, then the progress of the model is started with accumulation of software requirements analysis

We assume that the parameters required for the initial stage are P1,P2……Pn. The functions involved in this particular system are F1,F2…….Fn.Let the computed or predicted efficiency is Ec , while the optimum pre known efficiency is Ep. So, as I Ep – Ec I >> 1, it implies that the deviation of the computed value from the biased values (optimum value is termed as bias value). Hence the mathematical model for this particular case is as follows:



P(A)=probability of acceptance

P(R) = probability of reject

C1,C2,C3 are computing machines

A is the machine or delivering computed accumulated output

C is the comparator

Here the Ep and Ec should be nearly equal, i.e. the difference between the Ep and Ec should be minimum so that it will be accepted otherwise the output will be rejected, so there are only two conditions, either accepted or rejected. So the probability of acceptance and rejection is 0.5 i.e. ½.

VI. CONCLUSION

In this paper we have realized certain prediction based on reliability factor of software. In terms of functionality we have depicted certain condition based on nature of application . Finally we have pointed out artificial intelligence based decision making for estimation of output.

REFERENCES

[1] P.Chakrabarti et.al. "A novel approach towards realizing time variant key in cryptography" published in **International Journal** of Computer Science and Network Security, Korea , May 08

[2] P.Chakrabarti et.al. **"**Automatic Variable Key (AVK) in the light of various attacks , protocols , DES and AES " published in HIT Transaction on ECCN , an **internatonal journal** , Vol2,No.7, 2007

[3] P.Chakrabarti et.al. "An Intelligent Scheme towards information retrieval" accepted for publication in **international journal** AJIT(Asian Journal of Information Technology), ISSN: 1682-3915, Article ID: 706-AJIT

[4] P.Chakrabarti et.al. "Approach towards realizing resource mining and secured information transfer" published in **international journal** of IJCSNS, Korea , July08

[5] P.Chakrabarti et.al . "A statistical approach towards key prediction from hacker's point of view in cryptosystem" , IEEE conference ICCEE'08 , Thailand , Dec08