International Journal of Research in Science And Technology http://www.ijrst.com/

(IJRST) 2011, Vol. No. 1, Issue No. III, Oct-Dec ISSN: 2249-0604

MATHEMATICS RELIABILITY GROWTH MODEL: PRIMARY- FAILURES GENERATE SECONDARY- FAULT UNDER IMPERFECT DEBUGGING

SUMAN

DR. ANUJ KUMAR

Professor in Mathematics Department

AIMT Greater Noida (U.P.)

Research Scholar, CMJ University, Shillong. Meghalaya

INTRODUCTION

Today, computer hardware and Mathematics permeates our modern society. Computers are embedded in wristwatches, telephones, home appliances, buildings, automobiles, and aircraft. Science and technology demand high-performance hardware and high-quality Mathematics for making improvements and breakthroughs. We can look at virtually any industry - automotive, avionics, oil, telecommunications, banking, semi-conductors, pharmaceuticals - all these industries are highly dependent on computers for their basic functioning. When the requirements for and dependencies on computers increase, the possibility of cries from computer failures also increase. It is always desirable to remove a substantial number of faults from the Mathematics. In fact the reliability of the Mathematics is directly proportional to the number of faults removed. Hence the problem of maximization of Mathematics reliability is identical to that of maximization of fault removal. At the same time testing resource are not unlimited, and they need to be judiciously used.

In focusing on error prevention for reliability, we need to identify and measure the quality attributes applicable at different life cycle phases. As discussed previously, we need to specifically focus on requirements, design, implementation, and test phases.

Mathematics development process is often called Mathematics Life Cycle, because it describes the life of a Mathematics product from its conception to its implementation. Every Mathematics development process model includes system requirements as input and a delivered product as output. Many life cycle models have been proposed, based on the tasks involved in developing and maintaining Mathematics, but they all consist of the following stages and faults can be introduced during any of these stages.

Material and method

This section presents a Mathematics reliability growth model which incorporates the possibility of introducing new fault into a Mathematics system due to the imperfect debugging of the original faults in the system. The original faults manifest themselves as primary failures and are assumed to be distributed as a non homogeneous poison process (NHPP). Imperfect debugging of each primary failure induces a secondary failure which is assumed to occur in a delayed sense from the occurrence time of the primary failure. The mean total number of failures, comprising the primary and secondary failures, is obtained. We also discuss a cost model and consider some optimal release policies based on the model.

Notations

$\boldsymbol{a}$: mean number of faults initial found in the Mathematics ($a > 0$).

$\boldsymbol{b}$: failure rate per fault ($b > 0$) .

$\boldsymbol{p_0}, \boldsymbol{q_0}$: Pr{successfully removing a [primary, secondary] fault}.

$\boldsymbol{N(t)}$:point process for the total number of failures in $[0, t)$.

$\boldsymbol{m(t)}$: $E[N(t)]$ mean number of failures in $[0,t)$.

$\boldsymbol{N^p(t), N^s(t)}$: Point process: total number of [primary, secondary] failures in $[0, t)$.

$\boldsymbol{m^p(t)}$: $E[N^p(t)]$mean number of primary failures in $[0,t)$.

$\boldsymbol{m^s(t)}$: $E[N^s(t)]$ mean number of secondary failures in $[0, t)$.

$\boldsymbol{N_1^p(t), N_2^p(t)}$:point process: total number of primary failures [unsuccessfully, successfully] removed in $[0, t)$.

$\boldsymbol{m_1^p(t)}$: $E[N_1^p(t)]$mean number of primary failures unsuccessfully removed in $[0, t)$

$\boldsymbol{m_2^p(t)}$: $E[N_2^p(t)]$ mean number of primary failures successfully removed in $[0, t)$.

$\boldsymbol{N_1^s(t), N_2^s(t)}$:point process: total number of secondary failures [unsuccessfully, successfully] removed in $[0, t)$.

$\boldsymbol{m_1^s(t)}$: $E[N_1^s(t)]$ mean number of secondary failures unsuccessfully removed in $[0,t)$.

$\boldsymbol{m_2^s(t)}$: $E[N_2^s(t)]$ mean number of secondary failures successfully removed in $[0,t)$

$\boldsymbol{N^r(t)}$: point process: total number of failures removed in $[0, t)$.

$\boldsymbol{m^r(t)}$: $E[N^r(t)]$ mean number of failures removed in $[0,t)$.

$\boldsymbol{L\{f\}}$: $\int_0^\infty e^{-st}.f(t)dt, s > 0$ Laplace transform of $f(t)$.

$\boldsymbol{g(t)}$: pdf of time delay from 'introduction of a secondary fault by an imperfectly debugged primary failure' to 'occurrence of the failure caused by this fault'.

$\boldsymbol{\phi}$: $d\phi / dt$, where ϕ is any function of t.

$\boldsymbol{C_1}$: cost-rate of testing .

$\boldsymbol{C_2, C_2'}$: cost of a perfect debugging of a primary failure during the [testing, operational] period; $C_2' > C_2$.

$\boldsymbol{C_3, C_3'}$: cost of an imperfect debugging of a primary failure during the [testing, operational] period; $C_3' > C_3$.

$\boldsymbol{C_4, C_5}$: cost of a [perfect, imperfect] debugging of a secondary failure during the testing & operational period.

$\boldsymbol{T}$: Mathematics-release time.

$\boldsymbol{T^\bullet}$: optimal Mathematics-release time.

$\boldsymbol{t_c}$: Mathematics life-cycle length.

Cost Model and Optimal Release Policies

After a Mathematics system is developed and tested, it is released to operational phase (sold to users) at time T. This section determines the optimal release time T^* by minimizing the total mean cost, subject to a specified reliability requirement. Mathematically, Minimize: $C\,(T)$ = [cost of testing] + [maintenance cost after testing],

$$= C_1 \cdot T + C_2 \cdot m_2^p(T) + C_3 \cdot m_1^p(T) + C_4 \cdot m_2^s(T) + C_5 \cdot m_1^s(T) + C_2'[m_2^p(t_c) - m_2^p(T)] +$$
$$C_3' \cdot [m_1^p(t_c) - m_1^p(T)] + C_4[m_2^s(t_c) - m_2^s(T)] + C_5 \cdot m_1^s(t_c) - m_1^s(T)]$$

$$= C_1.T - A.m^p(T) + (C_2' p_0 + C_3'.(1-p_0)).m_1^p(t_c) \quad + (C_4.q_o + C_5.(1-q_o)).m^s(t_c). \tag{3.58}$$

Subject to:

$$R(x|T) = e^{-[m(T+x)-m(T)]} \geq R_0, \tag{3.59}$$

Notations

A	$(C_2' - C_2)\cdot p_0 + (C_3' - C_3)\cdot(1-p_0) > 0$
$\boldsymbol{R(x\|T)}$	Pr{no failures in the interval [0, T +x)}.
$\boldsymbol{R_0}$	Specified reliability requirement, $0 < R_0 < 1$
x	Operational time, $x > 0$.

We are concerned here only with finding optimal release times for the following two most frequently occurring situations:

S1. $m^p(t)$ is concave & increasing for all $t > 0$, eg, [Goel-Okumoto,1979];

S2. $m^p(t)$ is S-shaped, eg, [Yamada et al.1983-1984].

To solve this problem, we first find the optimal T which minimizes $C(T)$ globally, viz, without considering (3.59). We then find the T which satisfies the reliability requirement. Finally, we combine these two results to obtain the optimal release policies.

Unconstrained Solution

Eq (3.42) implies $m^p(0+)$ *and* $(\dot{m}^p(0+))' = a \cdot b$. From (3.58),

$$\dot{C}(T) = C_1 - A\dot{m}_p(T), \tag{3.60}$$

$$\ddot{C}(T) = -A\ddot{m}^p(T) \tag{3.61}$$

Let T_1 be the optimal release time. We discuss the optimal release policies in S1 and S2 under the conditions: a) $C_1 \geq$ a.b.A, and b) *Otherwise.*

S1a. $C_1 \geq$ a.b.A : From (3.61) and A> 0, the $C(T)$ is convex for all T >0. The condition implies $\dot{C}(0) \geq 0$; thus $C(T)$ is an increasing function of T. Therefore, $T_1 = 0$; thus the Mathematics should be released immediately after it has been developed.

S1b. Otherwise: $\dot{C}(0) < 0$; thus $C(T)$ has a unique turning point at T_a>0, ie, $\dot{C}(T_a) = 0$;thus T_1=T_a.

S2a. $C_1 \geq$ a.bA : Given that $m^p(t)$ is S-shaped, there exists a unique point of inflection T_b where,

$$\ddot{m}^p(t) > 0 \;\; for\; all \;\; t < T_b.$$

$$\ddot{m}^p(t) < 0 \;\; for\; all \;\; t > T_{b,}$$

Therefore (3.61) implies $\ddot{C}(T) < 0$ for all $T < T_{b,}$ and $\ddot{C}(T) > 0$ for all $T > T_b$. If $\dot{C}(T_b) \geq 0$, then $C(T)$ is increasing for all $T \geq 0$; thus $T_1 = 0$. Otherwise $C(T)$ has a unique local minimum at $T_a > T_b$. If $C(T_a) < C(0)$, then T_1=T_a Otherwise T_1=0.

S2b. Otherwise: Necessarily $\dot{C}(T_b) < 0$, and $C(T)$ has a unique local minimum at $T_a > T_b$= where $C(T_a) < C(0)$.Therefore $T_1 = T_a$

Reliability Constraint

Differentiating (3.36) and using assumption 5, we obtain:

$$\ddot{m}^p(t) = b.(1-2p_0)\dot{m}^p(t) - b.q_d\dot{m}^s(t) < 0.$$

Thus,

$$\dot{R} = -R(x|T)\cdot(\dot{m}(T+x) - \dot{m}(T)] > 0 \;\; for\; T > 0.$$

Thus $R(x|T)$ is an increasing function of T. Hence for a specific operational time x>0 and reliability requirement R_0,

1. If $R(x|T) < R_0$, then a unique & finite T=T_r, exists with $R(x|T) = R_{0,}$ and (3.59) is satisfied for all $T > T_r$.

2. Otherwise: (3.59) is satisfied for all $T \geq 0$.

Conclusion

We have discussed two models based on NHPP with imperfect debugging and discussed optimal release policies based on cost-reliability criterion. Cost also includes the cost incurred on those failures which could not be fixed during the development and operational phases. Next we have discussed a model which allows for imperfect debugging and three different error types. This is done within the framework of NHPP. The three error types are categorized by the difficulty of removal and detection. Minor errors (Type 3) are easily detected and removed; major errors (Type 2) are more difficult to detect and remove; critical errors (Type 1) are very difficult to detect and remove. We also presented an SRGM which incorporates the possibility of introducing new faults (i.e. secondary faults) into a Mathematics system due to the imperfect debugging of the original faults (i.e. primary faults) in the system. These new faults are assumed to occur in a delayed sense. Further we discussed the cost model with multiple failures, imperfect debugging as well as random life cycle in which cost also includes the penalty cost.

The probability of perfect debugging can usually be increased with additional cost and, hence, it has a strong influence on total Mathematics development cost. A concept of testing level is introduced here. To achieve the lowest Mathematics cost, the management can use the proposed cost model to formulate the optimal testing level and release time problem by considering the effect of imperfect debugging. Our problem formulation and the proposed solution is useful in practice as the imperfect debugging probability can be managed by using test engineers with proper experience, by selecting testing strategy or even by including a suitable number of review staff.

References

1. ^ [a] [b] *NEWSCLUB Premium Goods*. Famicom Tsūshin. No.347. Pg.179. 11 August 1995.
2. ^ "Donkey Kong Jr. Math Release Information for NES". GameFAQs. http://www.gamefaqs.com/nes/587235-donkey-kong-jr-math/data. Retrieved 2010-10-15.
3. ^ [a] [b] Lucas M. Thomas (2007-09-06). "Donkey Kong Jr. Math Review - Wii Review at IGN". IGN. http://wii.ign.com/articles/818/818013p1.html. Retrieved 2010-10-15.
4. ^ "Donkey Kong Jr. Math (GameCube Download) - Game Boy Advance - GameSpy". GameSpy. http://gba.gamespy.com/gameboy-advance/donkey-kong-jr-math-n-port/.Retrieved 2010-10-15.
5. ^ "Donkey Kong Jr. Math Release Information for Wii". GameFAQs. http://www.gamefaqs.com/wii/938470-donkey-kong-jr-math/data. Retrieved 2010-10-15.
6. ^ [a] [b] Alan, Brett (2010-10-03). "Donkey Kong Jr. Math - Overview". Allgame. http://www.allgame.com/game.php?id=22152. Retrieved 2010-10-15.

7. ^ *a b c* David Epstein (2005-10-26). "News: Joystick Nation - Inside Higher Ed". Inside Higher Ed. http://www.insidehighered.com/news/2005/10/26/games?no_mobile_redirect=true. Retrieved 2010-10-15.
8. ^ *Young Nintendo Fans Want Fun, Not Lessons*. *Los Angeles Times*. 1990. p. 102. http://news.google.com/newspapers?id=5ekcAAAAIBAJ&sjid=9lgEAAAAIBAJ&pg=3557,1113164. Retrieved 2010-10-15.[*dead link*]
9. ^ Slaven, Andy (2004). Video Game Bible, 1985-2002. Trafford Publishing. p. 102. http://books.google.com/books?id=PnPRd6QwvbQC. Retrieved 2010-10-15.
10. ^ Bob Mackey. "The Most Explosively Awful Console Launch Titles from 1UP.com". 1UP.com. http://www.1up.com/do/feature?cId=3167419. Retrieved 2010-10-15.
11. ^ Shea, Cam (2007-04-23). "Virtual Console AU Buyer's Guide - Part 4 - Wii Feature at IGN". IGN. http://wii.ign.com/articles/782/782613p1.html. Retrieved 2010-10-15.
12. ^ Shea, Cam (2008-08-07). "The Worst Games on Virtual Console: Part 1 - Wii Feature at IGN". IGN. http://wii.ign.com/articles/897/897106p1.html. Retrieved 2010-10-15.
13. ^ Provo, Frank (2007-09-03). "Donkey Kong Jr. Math Review for Wii". GameSpot. http://www.gamespot.com/wii/puzzle/donkeykongjrmath/review.html. Retrieved 2010-10-15.[*dead link*]
14. ^ Dan Whitehead (2007-06-02). "Virtual Console Roundup Wii Review - Page 3". Eurogamer. http://www.eurogamer.net/articles/r_vcroundup010607_wii?page=3. Retrieved 2010-10-15.
15. ^ "Wrecking Crew (Famicom Mini 14) Review for from 1UP.com". 1UP.com. http://www.1up.com/do/reviewPage?cId=3128754. Retrieved 2010-10-15.
16. ^ Jeremy Parish (2007-11-19). "Retro Roundup: 1st Anniversary Edition - Sonic, Zonk, Wrecking Crew: News from 1UP.com". 1UP.com. http://www.1up.com/news/retro-roundup-1119. Retrieved 2010-10-15.
17. ^ Caron, Frank (2007-09-03). "Virtual Console Monday (9-03-07): a labor day foursome". Ars Technica. http://arstechnica.com/gaming/news/2007/09/virtual-console-monday-9-03-07-a-labor-day-foursome.ars. Retrieved 2010-10-15.
18. ^ Daniel Johnson (2009-08-25). "News - Analysis: The Fun Of Language Development with Bookworm Adventures". Gamasutra.

    http://www.gamasutra.com/view/news/24839/Analysis_The_Fun_Of_Language_Development_with_Bookworm_Adventures.php. Retrieved 2010-10-15.