Neural, Parallel, and Scientific Computations, **27**, No. 3&4 (2019), 185-200ISSN: 1061-5369

COMMENTS ON A NEW SOFTWARE RELIABILITY MODEL

GEORGI SPASOV[1], OLGA RAHNEVA[2], AND ANGEL GOLEV[3]

[1,3]Faculty of Mathematics and Informatics
University of Plovdiv Paisii Hilendarski
24, Tzar Asen Str., 4000 Plovdiv, BULGARIA

[3]Faculty of Economy and Social Sciences
University of Plovdiv Paisii Hilendarski
24, Tzar Asen Str., 4000 Plovdiv, BULGARIA

ABSTRACT: In [1] Diwakar and Aggarwal considered a new three–parameter expected mean number of faults - function $m(t)$ by:

$$m(t) = a\left(1 - e^{-b\frac{t^{k+1}}{k+1}}\right) = aF(t),$$

where $F(t)$ is the cdf of Weibull distribution with shape parameter k. Then, the four–parameter Weibull model for open source software under the effect of imperfect debugging will be [1]:

$$M(t) = \frac{a}{1-\alpha}\left(1 - e^{-b(1-\alpha)\frac{t^{k+1}}{k+1}}\right),$$

where α is the constant rate at which new faults are introduced.

Also of interest to the specialists is the task of approximating the Heaviside function

$$h_{t_0}(t) = \begin{cases} 0, & \text{if} \quad t < t_0, \\ [0,1], & \text{if} \quad t = t_0, \\ 1, & \text{if} \quad t > t_0 \end{cases}$$

where t_0 is the median, i.e. $M(t_0) = \frac{1}{2}$ with the new function in the Hausdorff sense.

We give some examples with real datasets.

Numerical examples, illustrating our results are presented using programming environment *CAS Mathematica*.

AMS Subject Classification: 41A46

Key Words: three–parameter expected mean number of faults, four–parameter Diwakar and Aggarwal's software reliability model, Heaviside step–function $h_{t_0}(t)$, Hausdorff distance

Received: August 12, 2019 ; **Accepted:** December 18, 2019 ;
Published: December 31, 2019. **doi:** 10.12732/npsc.v27i3&4.5
Dynamic Publishers, Inc., Acad. Publishers, Ltd. https://acadsol.eu/npsc

1. INTRODUCTION AND PRELIMINARIES

Some software reliability models and studies on their "intrinsic properties", can be found in [3]–[42].

In this note we study the Hausdorff approximation of the Heaviside function $h_{t_0}(t)$ by function $M(t)$, defined by Diwakar and Aggarwal.

The model have been tested with real-world data.

Definition 1. *Diwakar and Aggarwal [1] developed the following new function:*

$$M(t) = \frac{a}{1-\alpha}\left(1 - e^{-b(1-\alpha)\frac{t^{k+1}}{k+1}}\right), \tag{1}$$

where α is the constant rate at which new faults are introduced.

Definition 2. *The shifted Heaviside step function is defined by*

$$h_{t_0}(t) = \begin{cases} 0, & \text{if} \quad t < t_0, \\ [0,1], & \text{if} \quad t = t_0, \\ 1, & \text{if} \quad t > t_0 \end{cases} \tag{2}$$

Definition 3. *[2] The Hausdorff distance (the H–distance) $\rho(f,g)$ between two interval functions f, g on $\Omega \subseteq \mathbb{R}$, is the distance between their completed graphs $F(f)$ and $F(g)$ considered as closed subsets of $\Omega \times \mathbb{R}$. More precisely,*

$$\rho(f,g) = \max\{ \sup_{A\in F(f)} \inf_{B\in F(g)} ||A-B||, \sup_{B\in F(g)} \inf_{A\in F(f)} ||A-B||\},$$

wherein $||.||$ is any norm in $\mathbb{R}^2$, e. g. the maximum norm $||(t,x)|| = \max\{|t|,|x|\}$; hence the distance between the points $A = (t_A, x_A)$, $B = (t_B, x_B)$ in $\mathbb{R}^2$ is $||A-B|| = max(|t_A - t_B|, |x_A - x_B|)$.

2. MAIN RESULTS

2.1. A NOTE ON THE NEW DIWAKAR AND AGGARWAL'S SOFTWARE RELIABILITY GROWTH MODEL (1)

The investigation of the characteristic "supersaturation" of the model (1) to the horizontal asymptote is important.

Let t_0 is the value for which $M(t_0) = \frac{1}{2}$.

The one–sided Hausdorff distance d between the function $h_{t_0}(t)$ and the $M(t)$ satisfies the relation

$$M(t_0 + d) = 1 - d. \tag{3}$$

For given α, a, b, $k0$ and t_0, the nonlinear equation $M(t_0 + d) - 1 + d = 0$ has unique positive root – d.

The model (1) for $a = 0.5$, $b = 5.1$, $\alpha = 0.5$, $k = 2$ and $t_0 = 0.934262$ is visualized on Fig. 1.

From the nonlinear equation (3) we have: $d = 0.246646$.

The model (1) for $a = 0.9$, $b = 40$, $\alpha = 0.1$, $k = 1.4$ and $t_0 = 0.277741$ is visualized on Fig. 2.

From the nonlinear equation (3) we have: $d = 0.146776$.

The model (1) for $a = 0.9999$, $b = 76$, $\alpha = 0.0001$, $k = 1.2$ and $t_0 = 0.1692$ is visualized on Fig. 3.

From the nonlinear equation (3) we have: $d = 0.114746$.

Some computational examples are presented in Table 1.

From the above examples, it can be seen that the "supersaturation" by the $M(t)$ is faster.

Obviously, this "advantage" can actually be used to approximate some specific data.

In the next Section, we will support what is said by analyzing real datasets from the spread of computer viruses.

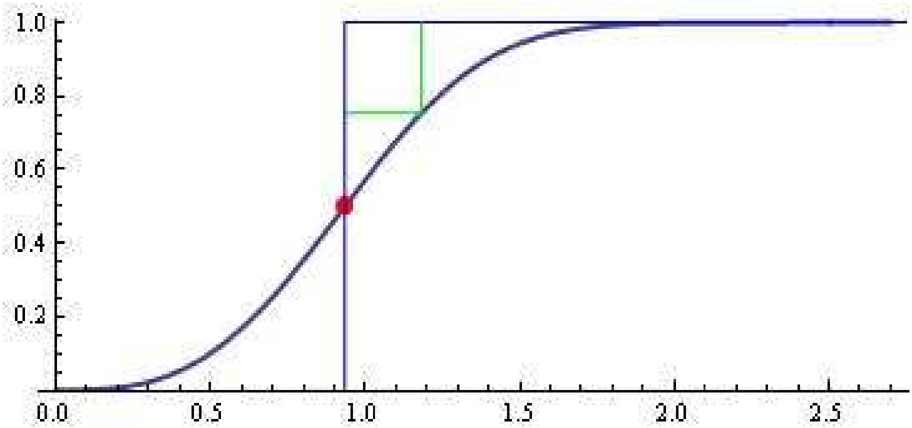

Figure 1: The model (1) for $a = 0.5$, $b = 5.1$, $\alpha = 0.5$, $k = 2$ and $t_0 = 0.934262$; H–distance $d = 0.246646$.

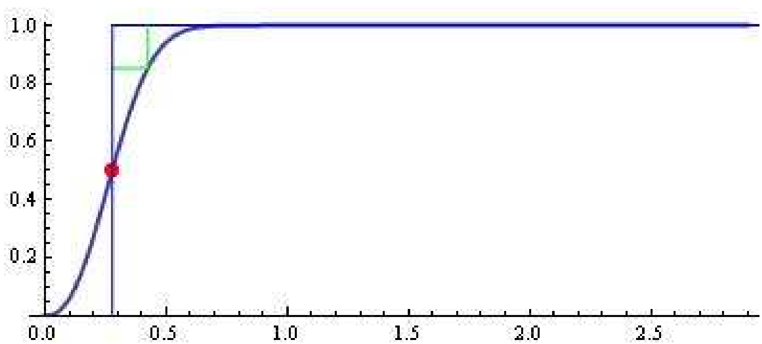

Figure 2: The model (1) for $a = 0.9$, $b = 40$, $\alpha = 0.1$, $k = 1.4$ and $t_0 = 0.277741$; H–distance $d = 0.146776$.

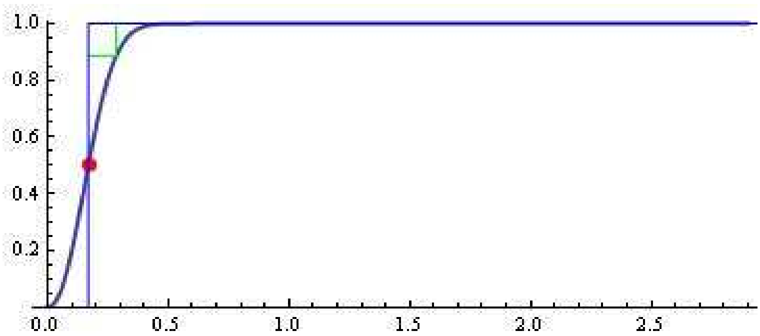

Figure 3: The model (1) for $a = 0.9999$, $b = 76$, $\alpha = 0.0001$, $k = 1.2$ and $t_0 = 0.1692$; H–distance $d = 0.114746$.

a	b	α	k	t_0	$H - distance$
0.5	5.1	0.5	2	0.934262	0.246646
0.95	35	0.05	1.5	0.306754	0.15129
0.9	40	0.1	1.4	0.277741	0.146776
0.9999	76	0.0001	1.2	0.1692	0.114746
0.96	55	0.04	1.3	0.218321	0.13049
0.99	100	0.01	1.5	0.156727	0.0976525

Table 1: The Hausdorff distance d computed by nonlinear equation (3)

2.2. APPLICATIONS

1. We analyze the following "actual data to estimate the number of software residual faults" [56]–[57] (see, Fig. 4).

After that using the model $M(t)$ for $\alpha = 0.1$, $a = 4667.4$, $b = 0.0293167$ and $k = 0.306508$ we obtain the fitted model (see, Fig. 5).

2. Analysis of Witty worm infection behavior [58] (see, Fig. 6–7.).

Here we will give an application of the model $M(t)$ when provide analysis of this real "data" [58], see Fig. 8.

Week	Cumulative number of software faults	Week	Cumulative number of software faults
1	248	31	4351
2	262	32	4401
3	372	33	4439
4	526	34	4488
5	742	35	4548
6	958	36	4596
7	1215	37	4629
8	1471	38	4680
9	1738	39	4713
10	1936	40	4749
11	1971	41	4783
12	2147	42	4817
13	2258	43	4849
14	2418	44	4877
15	2567	45	4901
16	2688	46	4928
17	2809	47	4950
18	2925	48	4970
19	3026	49	4998
20	3205	50	5024
21	3348	51	5060
22	3476	52	5085
23	3573	53	5088
24	3719	54	5090
25	3750	55	5110
26	3952	56	5129
27	4048	57	5139
28	4137	58	5167
29	4251	59	5186
30	4301		

Figure 4: the "actual data to estimate the number of software residual faults" [56]–[57].

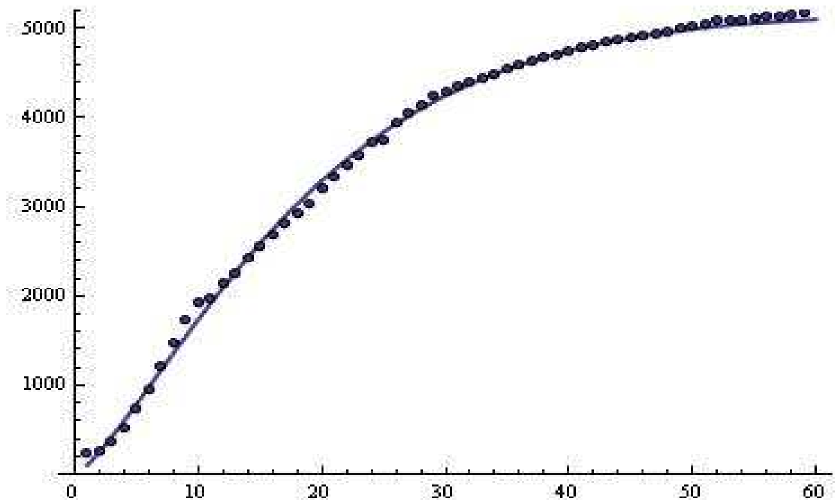

Figure 5: The fitted model $M(t)$.

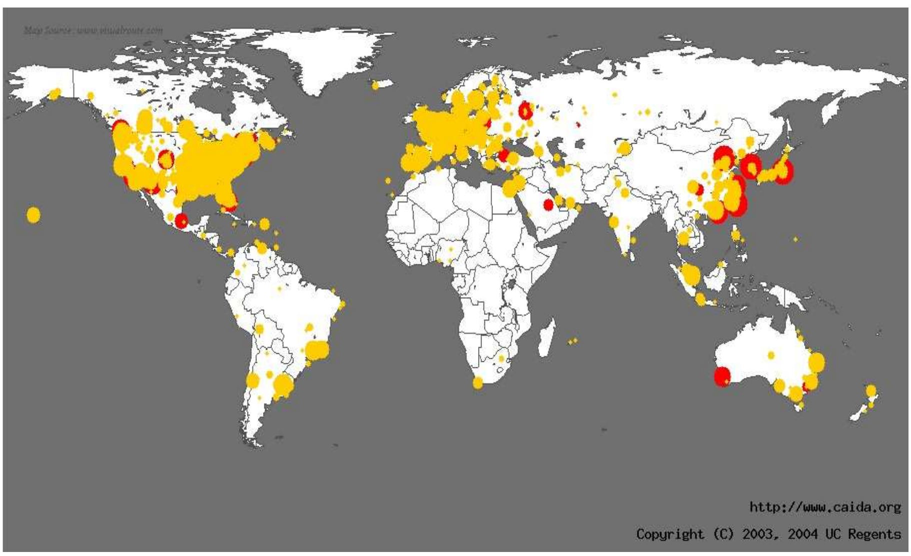

Figure 6: Global epidemic of Witty worm for entire world

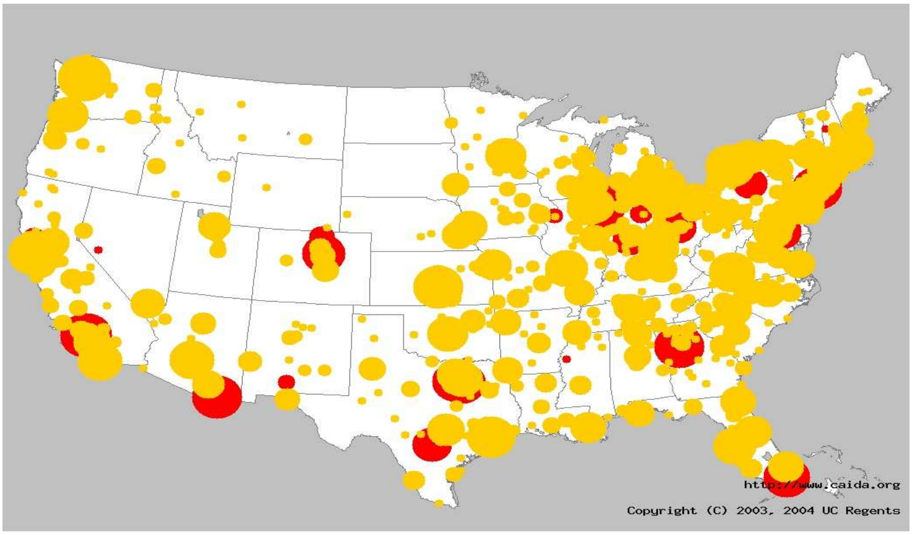

Figure 7: Global epidemic of Witty worm for USA

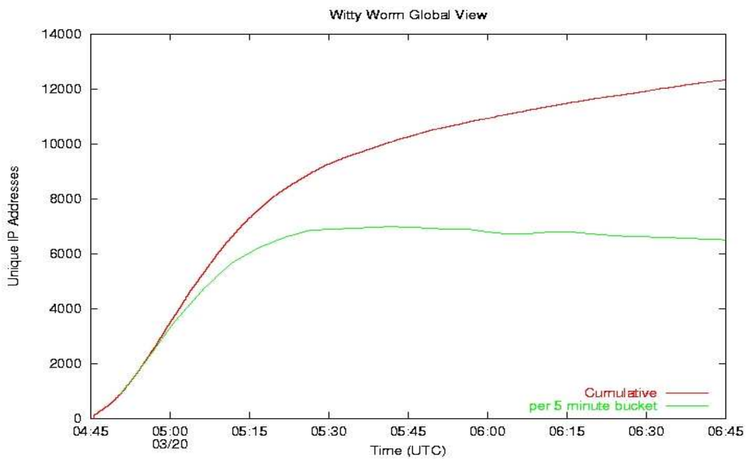

Figure 8: Epidemic data for entire world of Witty worm [58]

$data\_Witty\_World =$

$\{\{0.1, 150\}, \{5, 869\}, \{10, 2141\}, \{15, 3637\}, \{20, 5312\},$
$\{26, 6602\}, \{31, 7562\}, \{36, 8340\}, \{41, 8941\}, \{46, 9389\}, \{51, 9734\},$
$\{56, 10060\}, \{61, 10349\}, \{66, 10586\}, \{71, 10800\}, \{76, 11169\},$
$\{86, 11362\}, \{91, 11532\}, \{96, 11684\}, \{101, 11823\}, \{106, 11972\},$
$\{111, 12118\}, \{116, 12256\}, \{121, 12372\}\}$

$data\_Witty\_USA =$

$\{\{0.1, 150\}, \{5, 576\}, \{10, 1236\}, \{15, 1963\}, \{20, 2973\},$
$\{26, 3488\}, \{31, 3953\}, \{36, 4343\}, \{41, 4630\}, \{46, 4825\}, \{51, 4986\},$
$\{56, 5153\}, \{61, 5280\}, \{66, 5380\}, \{71, 5468\}, \{76, 5590\}, \{86, 5706\},$
$\{91, 5769\}, \{96, 5831\}, \{101, 5877\}, \{106, 5939\}, \{111, 5989\},$
$\{116, 6033\}, \{121, 6063\}\}.$

For entire World spreading parameters are (see Fig. 9)

$$\alpha = 0.05; a = 11753.4; b = 0.02112; k = 0.138046.$$

For USA spreading parameters are (see Fig.10)

$$\alpha = 0.1; a = 5456.7; b = 0.0257034; k = 0.13253.$$

Remark. In many cases it is appropriate to use the following new modified model:

$$M^*(t) = \frac{a}{1-\alpha}\left(1 - e^{-b(1-\alpha)(ct^k + c_1 t)}\right). \tag{4}$$

3. TROPICO R Failure Data

TROPICO-R software is divided into several implementation modules.

We will analyze the following real data "Cumulative number of failures (CNF) per periods of 10 days (time unit)" [59] (see Fig. 11).

The model for values of the parameters $a = 414.9$, $\alpha = 0.1$, $c = 0.01$, $c_1 = 0.13$, $b = 0.163274$, $k = 1.64671$ is shown in Fig. 12.

We will explicitly note that the study of " saturation " by the model in Hausdorff sense can be made in the way described in this article, and we will omit it here.

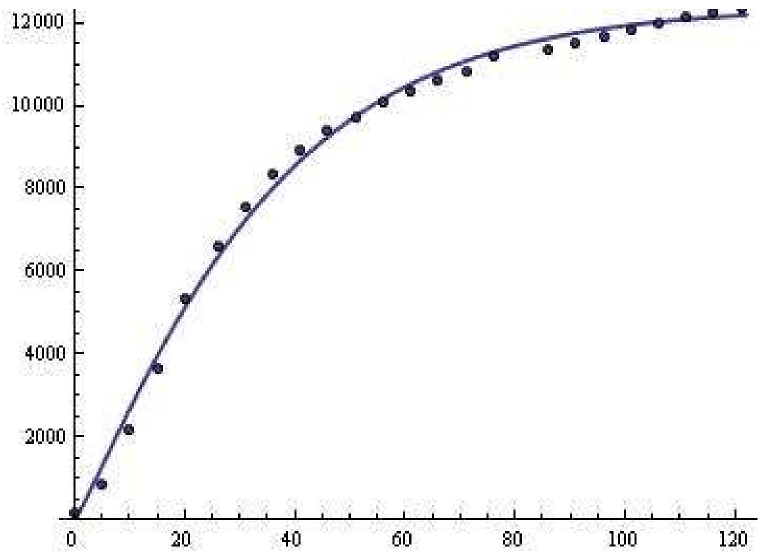

Figure 9: The fitted model $M(t)$

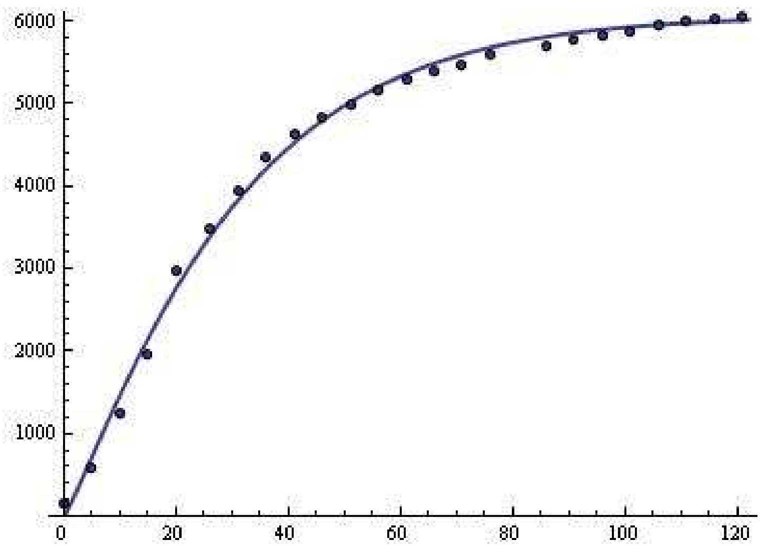

Figure 10: The fitted model $M(t)$

Validation		Field trial		Operation	
Time unit	CNF	Time unit	CNF	Time unit	CNF
1	7	31	301	43	356
2	8	32	302	44	367
3	36	33	310	45	373
4	45	34	317	46	373
5	60	35	319	47	378
6	74	36	323	48	381
7	82	37	324	49	383
8	98	38	338	50	384
9	106	39	342	51	384
10	115	40	345	52	387
11	120	41	350	53	387
12	134	42	352	54	387
13	139			55	388
14	142			56	393
15	145			57	398
16	153			58	400
17	157			59	407
18	174			60	413
19	183			61	414
20	196			62	417
21	200			63	419
22	214			64	420
23	223			65	429
24	246			66	440
25	257			67	443
26	277			68	448
27	283			69	454
28	286			70	456
29	292			71	456
30	297			72	456
				73	457
				74	458
				75	459
				76	459
				77	459
				78	460
				79	460
				80	460
				81	461

Figure 11: Cumulative number of failures (CNF) per periods of 10 days (time unit) [59]

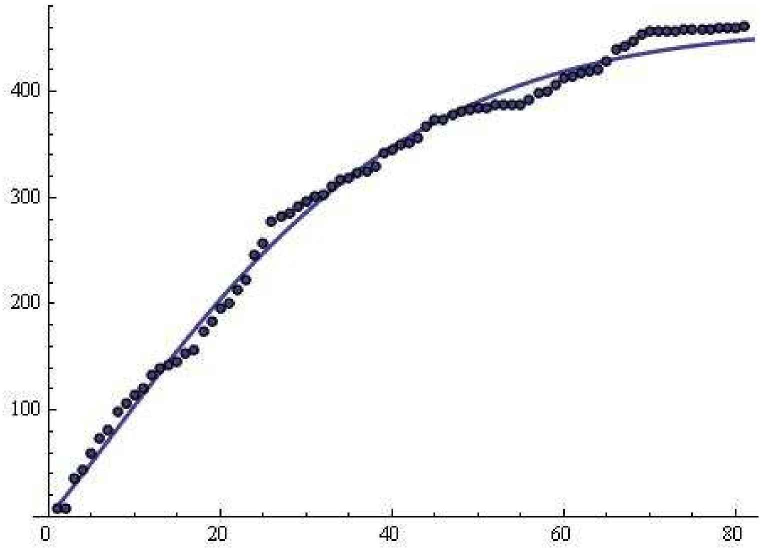

Figure 12: The fitted model $M^*(t)$

3. CONCLUDING REMARKS

The analysis we conducted in this article on the new Diwakar and Aggarwal's model shows its advantages and reliability compared to other similar models.

For other approximation and modelling results, see [46]–[55].

We hope that the results will be useful for specialists in this scientific area.

Acknowledgments
This paper is supported by the Project FP19-FMI-002 "Innovative ICT for Digital Research Area in Mathematics, Informatics and Pedagogy of Education" of the Scientific Fund of the University of Plovdiv Paisii Hilendarski, Bulgaria.

REFERENCES

[1] Diwakar, A. G. Aggarwal, Multi release reliability growth modeling for open source software under imperfect debugging, In: P. Kapur et al. (eds.) *System Performance and Management Analytics*, Springer Nature Singapore Pte Ltd., 2019.

[2] B. Sendov, *Hausdorff Approximations*, Kluwer, Boston (1990).

[3] H. Pham, A distribution function and its applications in software reliability, *Int. J. of Performability Engineering,* **15** (2019), 8 pp.

[4] K. Song, I. Chang, H. Pham, NHPP software reliability model with inflection factor of the fault detection rate considering the uncertainty of software operating environments and predictive analysis, *Symmetry*, **11** (2019), 521.

[5] Q. Li, H. Pham, A generalized software reliability growth model with consideration of the uncertainty of operating environment, *IEEE Access*, **XX** (2017).

[6] J. D. Musa, A. Ianino, K. Okumoto, Software Reliability: Measurement, Prediction, Applications, McGraw–Hill (1987).

[7] S. Yamada, Software Reliability Modeling: Fundamentals and Applications, Springer (2014).

[8] S. Yamada, Y. Tamura, *OSS Reliability Measurement and Assessment*, In: Springer Series in Reliability Engineering (H. Pham, Ed.), Springer International Publishing Switzerland (2016).

[9] H. Pham, *System Software Reliability*, In: Springer Series in Reliability Engineering, Springer–Verlag London Limited (2006).

[10] N. Pavlov, A. Iliev, A. Rahnev, N. Kyurkchiev, *Some software reliability models: Approximation and modeling aspects*, LAP LAMBERT Academic Publishing (2018), ISBN: 978-613-9-82805-0.

[11] N. Pavlov, A. Iliev, A. Rahnev, N. Kyurkchiev, *Nontrivial Models in Debugging Theory (Part 2)*, LAP LAMBERT Academic Publishing (2018), ISBN: 978-613-9-87794-2.

[12] K. Ohishi, H. Okamura, T. Dohi, Gompertz software reliability model: Estimation algorithm and empirical validation, *J. of Systems and Software*, **82**, No. 3 (2009), 535–543.

[13] D. Satoh, S. Yamada, Discrete equations and software reliability growth models, in: *Proc. 12th Int. Symp. on Software Reliab. and Eng.*, (2001), 176–184.

[14] S. Yamada, A stochastic software reliability growth model with Gompertz curve, *Trans. IPSJ*, **33** (1992), 964–969. (in Japanese)

[15] E. P. Virene, Reliability growth and its upper limit, in: Proc. 1968, *Annual Symp. on Reliab.*, (1968), 265–270.

[16] S. Rafi, S. Akthar, Software Reliability Growth Model with Gompertz TEF and Optimal Release Time Determination by Improving the Test Efficiency, *Int. J. of Comput. Applications*, **7**, No. 11 (2010), 34–43.

[17] S. Yamada, M. Ohba, S. Osaki, S–shaped reliability growth modeling for software error detection, *IEEE Trans. Reliab.*, **R–32** (1983), 475–478.

[18] S. Yamada, S. Osaki, Software reliability growth modeling: Models and Applications, *IEEE Transaction on Software Engineering*, **SE–11**, (1985), 1431–1437.

[19] N. Pavlov, G. Spasov, A. Rahnev, N. Kyurkchiev, Some deterministic reliability growth curves for software error detection: Approximation and modeling aspects, *International Journal of Pure and Applied Mathematics*, **118**, No. 3 (2018), 599–611.

[20] N. Pavlov, A. Golev, A. Rahnev, N. Kyurkchiev, A note on the Yamada–exponential software reliability model, *International Journal of Pure and Applied Mathematics*, **118**, No. 4 (2018), 871–882.

[21] N. Pavlov, A. Iliev, A. Rahnev, N. Kyurkchiev, A Note on The "Mean Value" Software Reliability Model, *International Journal of Pure and Applied Mathematics*, **118**, No. 4 (2018), 949–956.

[22] N. Pavlov, A. Iliev, A. Rahnev, N. Kyurkchiev, Analysis of the Chen's and Pham's Software Reliability Models, *Cybernetics and Information Technologies*, **18**, No. 3 (2018), 37–47.

[23] N. Pavlov, A. Golev, A. Rahnev, N. Kyurkchiev, A note on the generalized inverted exponential software reliability model, *International Journal of Advanced Research in Computer and Communication Engineering*, **7**, No. 3 (2018), 484–487.

[24] A. L. Goel, Software reliability models: Assumptions, limitations and applicability, *IEEE Trans. Software Eng.*, **SE–11** (1985), 1411–1423.

[25] N. Pavlov, A. Iliev, A. Rahnev, N. Kyurkchiev, Transmuted inverse exponential software reliability model, *Int. J. of Latest Research in Engineering and Technology*, **4**, No. 5 (2018), 1–6.

[26] A. Pandey, N. Goyal, *Early Software Reliability Prediction. A Fuzzy Logic Approach*, In: Studies in Fuzziness and Soft Computing (J. Kacprzyk, Ed.), **303**, Springer, London (2013).

[27] N. Pavlov, G. Spasov, A. Rahnev, N. Kyurkchiev, A new class of Gompertz–type software reliability models, *International Electronic Journal of Pure and Applied Mathematics*, **12**, No. 1 (2018), 43–57.

[28] O. Rahneva, H. Kiskinov, A. Malinova, G. Spasov, A Note on the Lee-Chang-Pham-Song Software Reliability Model, *Neural, Parallel, and Scientific Computations*, **26**, No. 3 (2018), 297–310.

[29] A. Wood, Predicting software reliability, *IEEE Computer*, **11** (1996), 69–77.

[30] H. Pham, L. Nordmann, X. Zhang, A General Imperfect-Software-Debugging Model with S-Shaped Fault-Detection Rate, *IEEE Trans. Rel.*, **48**, No. 2 (1999), 169–175.

[31] P. K. Kapur, H. Pham, A. Gupta, P. C. Jha, *Software Reliability Assessment with OR Applications*, In: Springer Series in Reliability Engineering, Springer-Verlag, London (2011).

[32] P. Karup, R. Garg, S. Kumar, *Contributions to Hardware and Software Reliability*, World Scientific, London (1999).

[33] M. Lyu (Ed. in Chief), *Handbook of Software Reliability Engineering*, IEEE Computer Society Press, The McGraw-Hill Companies, Los Alamitos (1996).

[34] M. Ohba, Software reliability analysis models, *IBM J. Research and Development*, **21** (1984).

[35] D.R. Jeske, X. Zhang, Some successful approaches to software reliability modeling in industry, *J. Syst. Softw.*, **74** (2005), 85–99.

[36] K. Song, H. Pham, A Software Reliability Model with a Weibull Fault Detection Rate Function Subject to Operating Environments, *Appl. Sci.*, **7** (2017), 983, doi:10.3390/app7100983, 16 pp.

[37] N. Pavlov, G. Spasov, A. Iliev, A. Rahnev, N. Kyurkchiev, A note on the Song–Chang–Pham's Software Reliability Model. Some Applications. I., *Neural, Parallel, and Scientific Computations*, **27**, No. 2 (2019), 115–129.

[38] N. Pavlov, G. Spasov, M. Stieger, A. Golev, A Note on the Extended Song-Chang-Pham's Software Reliability Model. II, *International Journal of Differential Equations and Applications,* **18**, No. 1 (2019), 87–98.

[39] J. D. Musa, *Software Reliability Data*, DACS, RADC, New York (1980).

[40] N. Pavlov, G. Spasov, A. Iliev, A. Rahnev, N. Kyurkchiev, A note on the Song–Chang–Pham's Software Reliability Model. Some Applications. I., *Neural, Parallel, and Scientific Computations*, **27**, No. 2 (2019), 115–129.

[41] Q. Li, H. Pham, NHPP Software Reliability Model Considering the Uncertainty of Operating Debugging and Testing Coverage, *Applied Mathematical Modelling*, **41**, No. 11 (2017), 68–85.

[42] H. Pham, A Generalized Fault-Detection Software Reliability Model Subject to Random Operating Environments, *Journal of Computer Science*, **3**, No. 3 (2016), 145–150.

[43] M. Conti, A. Gangwal, S. Ruj, On the economic significance of ransomware campaigns: A Bitcoin transactions perspective, *Computers & Security*, **79** (2018), 162–189.

[44] S. Sarat, A. Terzis, HiNRG Technical Report: 01-10-2007 Measuring the Storm Worm Network, (2007).

[45] L. Rogers-Bennett, D. W. Rogers, S. A. Schultz, Modeling growth and mortality of red abalone Haliotis Rufescens in Northern California, *J. of Shellfish Research*, **26**, No. 3 (2007), 719-727.

[46] N. Kyurkchiev, A. Iliev, A. Rahnev, *Some Families of Sigmoid Functions: Applications to Growth Theory*, LAP LAMBERT Academic Publishing (2019), ISBN: 978-613-9-45608-6.

[47] N. Kyurkchiev, S. Markov, On the Hausdorff distance between the Heaviside step function and Verhulst logistic function, *J. Math. Chem.*, **54** (2016), 109–119.

[48] N. Kyurkchiev, On a Sigmoidal Growth Function Generated by Reaction Networks. Some Extensions and Applications, *Communications in Applied Analysis*, **23**, No. 3 (2019), 383–400.

[49] S. Markov, N. Kyurkchiev, A. Iliev, A. Rahnev, On the approximation of the generalized cut functions of degree p+1 by smooth hyper-log-logistic function, *Dynamic Systems and Applications*, **27**, No. 4 (2018), 715–728.

[50] S. Markov, A. Iliev, A. Rahnev, N. Kyurkchiev, A note on the Log-logistic and transmuted Log-logistic models. Some applications, *Dynamic Systems and Applications*, **27**, No. 3 (2018), 593–607.

[51] A. Iliev, N. Kyurkchiev, A. Rahnev, T. Terzieva, *Some models in the theory of computer viruses propagation*, LAP LAMBERT Academic Publishing (2019), ISBN: 978-620-0-00826-8.

[52] S. Markov, A. Iliev, A. Rahnev, N. Kyurkchiev, A note on the n-stage growth model. Overview, *Biomath Communications*, **5**, No. 2 (2018), 79–100.

[53] N. Kyurkchiev, S. Markov, *Sigmoid functions: Some Approximation and Modelling Aspects*, LAP LAMBERT Academic Publishing, Saarbrucken (2015), ISBN 978-3-659-76045-7.

[54] N. Kyurkchiev, A. Iliev, *Extension of Gompertz-type Equation in Modern Science: 240 Anniversary of the birth of B. Gompertz*, LAP LAMBERT Academic Publishing (2018), ISBN: 978-613-9-90569-0.

[55] N. Kyurkchiev, A. Iliev, S. Markov, *Some techniques for recurrence generating of activation functions*, LAP LAMBERT Academic Publishing (2017), ISBN: 978-3-330-33143-3.

[56] T. Mitsuhashi, *A method of software quality evaluation*, JUSE Press, Tokyo (1981). (in Japanese)

[57] D. Satoh, A discrete Gompertz equation and a software reliability growth model, *IEICE Trans. Inf. and Syst.*, **E83–D**, No. 7 (2000), 1508–1513.

[58] C. Shannon, D. Moore, The Spread of the Witty Worm, *IEEE Security & Privacy*, **July/August**, (2004), 46–50.

[59] Kanoun, K., de Bastos Martini, M. R., & De Souza, J. M., A method for software reliability analysis and prediction application to the TROPICO-R switching system, *IEEE Transactions on Software Engineering*, **17** No. 4 (1991), 334-344.