然后再在此基础上使用遗传算法搜索状态空间中的最优解,最后根据遗传算法的结果对程序实体排序以辅助开发人员查找和修改缺陷.本文的实验结果表明,GAMFal 方法在单缺陷程序中有与其他方法类似的定位效果,而在多缺陷程序中的定位效果要明显好于其他方法.同时,该方法能够有效处理开发中缺陷数量不确定和通用模块误判的情况,能够更好地满足实际开发中对缺陷定位的需求.

我们认为该方法仍有一些后续工作值得扩展,具体来说:(1) 我们将尝试将该方法应用到其他大规模程序中,分析本文结论是否具有一般性.(2) 在算法的第 1 阶段将进一步关注算法的执行效率,并尝试不同算法以期更高效、准确地搜索到最优种群.(3) 在算法的第 2 阶段将重点研究种群内个体排序对定位的影响,尝试采用不同方法研究定位问题.

References:

[1] Vessey I. Expertise in debugging computer programs: A process analysis. Int'l Journal of Man-Machine Studies, 1985,23(5): 459−494. [doi: 10.1016/S0020-7373(85)80054-7]

[2] Mayer W, Stumptner M. Evaluating models for model-based debugging. In: Proc. of the Int'l Conf. on Automated Software Engineering. L'Aquila: Springer-Verlag, 2008. 128−137. [doi: 10.1109/ASE.2008.23]

[3] Jones JA, Harrold MJ, Stasko J. Visualization of test information to assist fault localization. In: Proc. of the Int'l Conf. on Software Engineering. Orlando: ACM Press, 2002. 467−477. [doi: 10.1145/581339.581397]

[4] Jones JA, Bowring JF, Harrold MJ. Debugging in parallel. In: Proc. of the Int'l Symp. on Software Testing and Analysis. London: ACM Press, 2007. 16−26. [doi: 10.1145/1273463.1273468]

[5] Rui A, Zoeteweij P, Gemund AJC. On the accuracy of spectrum-based fault localization. In: Proc. of the Testing Academic and Industrial Conf. Practice and Research Techniques. Cumberland Lodge: Springer-Verlag, 2007. 89−98. [doi: 10.1109/TAIC.PART.2007.13]

[6] Chen MY, Kiciman E, Fratkin E, Fox A, Brewer E. Pin-Point: Problem determination in large, dynamic internet services. In: Proc. of the Int'l Conf. on Dependable Systems and Networks. Washington: IEEE Press, 2002. 595−604. [doi: 10.1109/DSN.2002.1029005]

[7] Ochiai A. Zoogeographic studies on the soleoid fishes found in Japan and its neighboring regions. Nihon-Suisan-Gakkai-Shi, 1957, 22(9):526−530. [doi: 10.2331/suisan.22.526]

[8] Naish L, Hua JL, Ramamohanarao K. A model for spectra-based software diagnosis. ACM Trans. on Software Engineering and Methodology, 2011,20(3):563−574. [doi: 10.1145/2000791.2000795]

[9] Abreu R, Zoeteweij P, Gemund AJCV. Spectrum-Based multiple fault localization. In: Proc. of the Int'l Conf. on Automated Software Engineering. Auckland: Springer-Verlag, 2009. 88−99. [doi: 10.1109/ASE.2009.25]

[10] Steimann F, Bertschler M. A simple coverage-based locator for multiple faults. In: Proc. of the Int'l Conf. on Software Testing Verification and Validation. Denver: IEEE Press, 2009. 366−375. [doi: 10.1109/ICST.2009.24]

[11] Moon S, Kim Y, Kim M, Yoo S. Ask the mutants: Mutating faulty programs for fault localization. In: Proc. of the Int'l Conf. on Software Testing, Verification and Validation. Abano Terme: IEEE Press, 2014. 153−162. [doi: 10.1109/ICST.2014.28]

[12] Chen X, Ju XL, Wen WZ, Gu Q. Review of dynamic fault localization approaches based on program spectrum. Ruan Jian Xue Bao/Journal of Software, 2015,26(2):390−412 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4708.htm [doi: 10.13328/j.cnki.jos.004708]

[13] Yu K, Lin MX. Advances in automatic fault localization techniques. Chinese Journal of Computers, 2011,34(8):1411−1423 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2011.01411]

[14] DiGiuseppe N, Jones JA. On the influence of multiple faults on coverage-based fault localization. In: Proc. of the Int'l Symp. on Software Testing and Analysis. Toronto: ACM Press, 2011. 210−220. [doi: 10.1145/2001420.2001446]

[15] Harman M, Jones BF. Search-Based software engineering. Information and Software Technology, 2001,43(14):833−839. [doi: 10.1016/S0950-5849(01)00189-6]

[16] Harman M, Mansouri SA, Zhang Y. Search-Based software engineering: Trends, techniques and applications. ACM Computing Surveys, 2012,45(1):1−61 [doi: 10.1145/2379776.2379787]

[17] Li Z, Gong DW, Nie CH, Jiang H. The progress and development tendency of the research on search-based software engineering. 2013-2014 Annual Report of Computer Science and Technology, Beijing: China Machine Press, 2014. 139−187 (in Chinese with English abstract).

[18] Buhler O, Wegener J. Evolutionary functional testing. Computers & Operations Research, 2008,35(10):3144−3160. [doi: 10.1016/j.cor.2007.01.015]

[19] Wegener J, Sthamer H, Jones BF, Eyres DE. Testing real-time systems using genetic algorithms. Software Quality Journal, 1997, 6(2):127−135. [doi: 10.1023/A:1018551716639]

[20] Briand LC, Feng J, Labiche Y. Using genetic algorithms and coupling measures to devise optimal integration test orders. In: Proc. of the Int'l Conf. on Software Engineering and Knowledge Engineering. 2002. 43−50. [doi: 10.1145/568760.568769]

[21] Briand LC, Labiche Y, Shousha M. Stress testing real-time systems with genetic algorithms. In: Proc. of the Genetic & Evolutionary Computation Conf. Washington: ACM Press, 2005. 1021−1028. [doi: 10.1145/1068009.1068183]

[22] Masud M, Nayak A, Zaman M, Bansal N. Strategy for mutation testing using genetic algorithms. In: Proc. of the Canadian Conf. on Electrical and Computer Engineering. Madrid: IEEE Press, 2005. 1049−1052. [doi: 10.1109/CCECE.2005.1557156]

[23] Ghazi SA, Ahmed MA. Pair-Wise test coverage using genetic algorithms. Evolution Computation, 2003,2:1420−1424. [doi: 10.1109/CEC.2003.1299837]

[24] Li Z, Harman M, Hierons RM. Search algorithms for regression test case prioritization. IEEE Trans. on Software Engineering, 2007, 33(4):225−237. [doi: 10.1109/TSE.2007.38]

[25] McMinn P. Search-Based software test data generation: A survey. Software Testing Verification & Reliability, 2004,14(2):105−156. [doi: 10.1002/stvr.294]

[26] Shaukat A, Briand LC, Hemmati H, Panesar-Walawege RK. A systematic review of the application and empirical investigation of search-based test case generation. IEEE Trans. on Software Engineering, 2011,36(6):742−762. [doi: 10.1109/TSE.2009.52]

[27] Qi YH, Mao XG, Lei Y, Dai ZY, Wang CS. The strength of random search on automated program repair. In: Proc. of the Int'l Conf. on Software Engineering. Hyderabad: ACM Press, 2014. 254−265. [doi: 10.1145/2568225.2568254]

[28] Qi YH, Mao XG, Lei Y, Wang CS. Using automated program repair for evaluating the effectiveness of fault localization techniques. In: Proc. of the Int'l Symp. on Software Testing and Analysis. Lugano: ACM Press, 2013. 191−201. [doi: 10.1145/2483760.2483785]

[29] Mao XG, Lei Y, Dai ZY, Qi YH, Wang CS. Slice-Based statistical fault localization. Journal of Systems & Software, 2014,89(2): 51−62. [doi: 10.1016/j.jss.2013.08.031]

[30] Lei Y, Mao XG, Dai ZY, Wang CS. Effective statistical fault localization using program slices. In: Proc. of the IEEE Computer Software & Applications Conf. Lzmir: IEEE Press, 2012,7204(4):1−10. [doi: 10.1109/COMPSAC.2012.9]

[31] Li MQ, Kou JS, Lin D. Genetic Algorithm Basic Theory and Application. Beijing: Science Press, 2002 (in Chinese).

[32] Holland JH. Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence. University of Michigan Press, 1975. [doi: 10.1086/418447]

[33] Liblit B, Naik M, Zheng AX, Aiken A, Jordan MI. Scalable statistical bug isolation. In: Proc. of the Conf. on Programming Language Design and Implementation. Chicago: ACM Press, 2005. 15−26. [doi: 10.1145/1065010.1065014]

[34] Liu C, Yan X, Fei L, Han JW, Midkiff SP. SOBER: Statistical model-based bug localization. In: Proc. of the European Software Engineering Conf. on Held Jointly with Int'l Symp. on Foundations of Software Engineering. Lisbon: ACM Press, 2005. 286−295. [doi: 10.1145/1081706.1081753]

[35] Li W, Zheng Z, Hao P, Gao YC, Rao PF, Gong C. Predicate execution-sequence based fault localization algorithm. Chinese Journal of Computers, 2013,36(12):2406−2419 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2013.02406]

[36] Hao P, Zheng Z, Zhang ZY, Gao YC, Gong C, Xue YZ. Self-Adaptive fault localization algorithm based on predicate executioninformation analysis. Chinese Journal of Computers, 2014,37(3):500−511 (in Chinese with English abstract).

[37] Wong WE, Debroy V. A survey on software fault localization [Ph.D. Thesis]. Department of Computer Science, UT Dallas, 2009.

[38] Yoo S. Evolving human competitive spectra-based fault localisation techniques. In: Search Based Software Engineering. Berlin, Heidelberg: Springer-Verlag, 2012. 244−258. [doi: 10.1007/978-3-642-33119-0_18]

[39] Xie XY, Kuo FC, Chen TY, Yoo S, Harman M. Provably optimal and human-competitive results in SBSE for spectrum based fault localization. In: Proc. of the Int'l Conf. on Search Based Software Engineering. Saint Petersburg: Springer-Verlag, 2013. 224−238. [doi: 10.1007/978-3-642-39742-4_17]

[40] Xuan JF, Monperrus M. Learning to combine multiple ranking metrics for fault localization. In: Proc. of the Int'l Conf. on Software Maintenance and Evolution. Victoria: IEEE Press, 2014. 191−200. [doi: 10.1109/ICSME.2014.41]

[41] Hao D, Zhang L, Pan Y, Mei H, Sun JS. On similarity-awareness in testing-based fault localization. Automated Software Engineering, 2008,15(2):207−249. [doi: 10.1007/s10515-008-0025-9]

[42] Hao D, Zhang L, Zhong H, Mei H, Sun JS. Eliminating harmful redundancy for testing-based fault localization using test suite reduction: An experimental study. In: Proc. of the Int'l Conf. on Software Maintenance and Evolution. Amsterdam: IEEE Press, 2005. 683−686. [doi: 10.1109/ICSM.2005.43]

[43] He T, Wang XM, Zhou XC, Li WJ, Zhang ZY, Cheung SC. A software fault localization technique based on program mutations. Chinese Journal of Computers, 2013,36(11):2236−2244 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2013.02236]

[44] Masri W. Fault localization based on information flow coverage. Software Testing, Verification and Reliability, 2010,20(2): 121−147. [doi: 10.1002/stvr.409]

[45] Zhang ZY, Jiang B, Chan WK, Tse TH. Debugging through evaluation sequences: A controlled experimental study. In: Proc. of the Int'l Computer Software and Applications Conf. Turku: IEEE Press, 2008. 128−135. [doi: 10.1109/COMPSAC.2008.207]

[46] Zhang ZY, Jiang B, Chan WK, Tse TH, Wang X. Fault localization through evaluation sequences. Journal of Systems and Software, 2010,83(2):174−187. [doi: 10.1016/j.jss.2009.09.041]

[47] Baah GK, Podgurski A, Harrold MJ. The probabilistic program dependence graph and its application to fault diagnosis. In: Proc of the Int'l Symp. on Software Testing and Analysis. Seattle: ACM Press, 2008. 189−200. [doi: 10.1145/1390630.1390654]

[48] Baah GK, Podgurski A, Harrold MJ. Causal inference for statistical fault localization. In: Proc. of the Int'l Symp. on Software Testing and Analysis. Trento: ACM Press, 2010. 73−84. [doi: 10.1145/1831708.1831717]

[49] Baah GK, Podgurski A, Harrold MJ. Mitigating the confounding effects of program dependences for effective fault localization. In: Proc. of the Joint Meeting of the European Software Engineering Conf. and the Symp. on the Foundations of Software Engineering. Szeged: ACM Press, 2011. 146−156. [doi: 10.1145/2025113.2025136]

[50] Agrawal H, Horgan JR. Dynamic program slicing. In: Proc. of the Conf. on Programming Language Design and Implementation. White Plains: ACM Press, 1990. 246−256. [doi: 10.1145/93542.93576]

[51] Agrawal H, Horgan JR, London S, Wong WE. Fault localization using execution slices and dataflow tests. In: Proc. of the Int'l Symp. on Software Reliability Engineering. Toulouse: IEEE Press, 1995. 143−151. [doi: 10.1109/ISSRE.1995.497652]

[52] Abreu R, Zoeteweij P, Gemund AJCV. Spectrum-Based multiple fault localization. In: Proc. of the Int'l Conf. on Automated Software Engineering. Auckland: Springer-Verlag, 2009. 88−99. [doi: 10.1109/ASE.2009.25]

[53] Wen WZ, Li BX, Sun XB, Qi SS. A technique of multiple fault localization based on conditional execution slicing spectrum. Journal of Computer Research and Development, 2013,50(5):1030−1043 (in Chinese with English abstract).

[54] Groce A. Error explanation with distance metrics. In: Proc. of the Tools and Algorithms for the Construction and Analysis of Systems. Vienna: Springer-Verlag, 2004. 108−122. [doi: 10.1007/s10009-005-0202-0]

[55] Li Z, Harman M, Hierons RM. Search algorithms for regression test case prioritization. IEEE Trans. on Software Engineering, 2007, 33(4):225−237. [doi: 10.1109/TSE.2007.38]

[56] Do H, Elbaum S, Rothermel G. Supporting controlled experimentation with testing techniques: An infrastructure and its potential impact. Empirical Software Engineering, 2005,10(4):405−435. [doi: 10.1007/s10664-005-3861-2]

[57] Wong WE, Wei T, Qi Y, Zhao L. A crosstab-based statistical method for effective fault localization. In: Proc. of the Int'l Conf. on Software Testing, Verification, and Validation. Lillehammer: IEEE Press, 2008. 42−51. [doi: 10.1109/ICST.2008.65]

[58] Friedman M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. Journal of the American Statistical Association, 1937,32(200):675−701. [doi: 10.2307/2279372]

[59] Wilcoxon F. Individual comparisons by ranking methods. Biometrics, 1945,1(6):80−83. [doi: 10.2307/3001968]

附中文参考文献:

[12] 陈翔,鞠小林,文万志,顾庆.基于程序频谱的动态缺陷定位方法研究.软件学报,2015,26(2):390−412. http://www.jos.org.cn/1000-9825/4708.htm [doi: 10.13328/j.cnki.jos.004708]

[13] 虞凯,林梦香.自动化软件错误定位技术研究进展.计算机学报,2011,34(8):1411−1422. [doi: 10.3724/SP.J.1016.2011.01411]

[17] 李征,巩敦卫,聂长海,江贺.基于搜索的软件工程研究进展与趋势.2013-2014 年度中国计算机科学技术年度报告.北京:机械工业出版社,2014.139−187.

[31] 李敏强,寇纪淞,林丹.遗传算法的基本理论与应用.北京:科学出版社,2002.

[35] 李伟,郑征,郝鹏,高乙超,饶培峰,宫成.基于谓词执行序列的软件缺陷定位算法.计算机学报,2013,36(12):2406−2419. [doi: 10.3724/SP.J.1016. 2013.02406]

[36] 郝鹏,郑征,张震宇,高乙超,宫成,薛云志.基于谓词执行信息分析的自适应缺陷定位算法.计算机学报,2014,37(3):500−511.

[43] 贺韬,王欣明,周晓聪,李文军,张震宇,张成志.一种基于程序变异的软件错误定位技术.计算机学报,2013,36(11):2236−2244. [doi: 10.3724/SP.J.1016.2013.02236]

[53] 文万志,李必信,孙小兵,齐珊珊.基于条件执行切片谱的多错误定位.计算机研究与发展,2013,50(5):1030−1043.

王赞(1979－),男,江苏泗洪人,博士,副教授,CCF 会员,主要研究领域为软件工程,软件测试,机器学习.

邹雨果(1989－),男,工程师,主要研究领域为软件开发,软件测试.

樊向宇(1992－),男,硕士生,CCF 学生会员,主要研究领域为软件测试,机器学习,软件质量.

陈翔(1980－),男,博士,副教授,CCF 会员,主要研究领域为软件缺陷预测,软件缺陷定位,回归测试,组合测试.