9. REFERENCES

[1] A. T. Acree, Jr. *On Mutation*. PhD thesis, Atlanta, GA, USA, 1980. AAI8107280.

[2] A. T. Acree Jr. On mutation. Technical report, DTIC Document, 1980.

[3] P. Ammann and J. Offutt. *Introduction to software testing*. Cambridge University Press, 2008.

[4] J. H. Andrews, L. C. Briand, and Y. Labiche. Is mutation an appropriate tool for testing experiments? In *Software Engineering, 2005. ICSE 2005. Proceedings. 27th International Conference on*, pages 402–411. IEEE, 2005.

[5] Apache Software Foundation. Apache maven project. http://maven.apache.org.

[6] E. F. Barbosa, J. C. Maldonado, and A. M. R. Vincenzi. Toward the determination of sufficient mutant operators for c. *Software Testing, Verification and Reliability*, 11(2):113–136, 2001.

[7] T. A. Budd. *Mutation Analysis of Program Test Data*. PhD thesis, New Haven, CT, USA, 1980. AAI8025191.

[8] H. Coles. Pit mutation testing. http://pittest.org/.

[9] H. Coles. Pit mutation testing: Mutators. http://pitest.org/quickstart/mutators.

[10] R. A. DeMillo, D. S. Guindi, W. McCracken, A. Offutt, and K. King. An extended overview of the mothra software testing environment. In *Software Testing, Verification, and Analysis, 1988., Proceedings of the Second Workshop on*, pages 142–151. IEEE, 1988.

[11] L. Deng, J. Offutt, and N. Li. Empirical evaluation of the statement deletion mutation operator. In *IEEE 6th International Conference on Software Testing, Verification and Validation.*, Luxembourg, 2013.

[12] GitHub Inc. Software repository. http://www.github.com.

[13] R. Gopinath. Replication data for: A comparison of mutation approaches. http://dx.doi.org/10.7910/DVN/24936.

[14] R. Gopinath, C. Jensen, and A. Groce. Code coverage for suite evaluation by developers. In *36th International Conference on Software Engineering*, 2014.

[15] Y. Jia and M. Harman. An analysis and survey of the development of mutation testing. *Software Engineering, IEEE Transactions on*, 37(5):649–678, 2011.

[16] R. J. Lipton. Fault diagnosis of computer programs. Technical report, Carnegie Mellon Univ., 1971.

[17] A. Mathur. Performance, effectiveness, and reliability issues in software testing. In *Computer Software and Applications Conference, 1991. COMPSAC '91., Proceedings of the Fifteenth Annual International*, pages 604–605, 1991.

[18] E. S. Mresa and L. Bottaci. Efficiency of mutation operators and selective mutation strategies: An empirical study. *Software Testing Verification and Reliability*, 9(4):205–232, 1999.

[19] A. S. Namin and J. H. Andrews. Finding sufficient mutation operators via variable reduction. In *Proceedings of the 2nd Workshop on Mutation Analysis (MUTATION'06)*, page 5, 2006.

[20] A. Offutt, G. Rothermel, and C. Zapf. An experimental evaluation of selective mutation. In *Software Engineering, 1993. Proceedings., 15th International Conference on*, pages 100–107, 1993.

[21] A. J. Offutt, A. Lee, G. Rothermel, R. H. Untch, and C. Zapf. An experimental determination of sufficient mutant operators. *ACM Trans. Softw. Eng. Methodol.*, 5(2):99–118, Apr. 1996.

[22] A. J. Offutt, A. Lee, G. Rothermel, R. H. Untch, and C. Zapf. An experimental determination of sufficient mutant operators. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 5(2):99–118, 1996.

[23] A. J. Offutt, G. Rothermel, and C. Zapf. An experimental evaluation of selective mutation. In *Proceedings of the 15th international conference on Software Engineering*, pages 100–107. IEEE Computer Society Press, 1993.

[24] A. J. Offutt and R. H. Untch. Mutation 2000: Uniting the orthogonal. In *Mutation testing for the new century*, pages 34–44. Springer, 2001.

[25] C. Pacheco and M. D. Ernst. Randoop random test generation. http://code.google.com/p/randoop.

[26] D. Schuler and A. Zeller. Javalanche: Efficient mutation testing for java. In *ESEC/FSE '09: Proceedings of the 7th joint meeting of the European Software Engineering Conference and the ACM SIGSOFT International Symposium on Foundations of Software Engineering*, pages 297–298, Aug. 2009.

[27] A. Siami Namin, J. H. Andrews, and D. J. Murdoch. Sufficient mutation operators for measuring test effectiveness. In *Proceedings of the 30th international conference on Software engineering*, pages 351–360. ACM, 2008.

[28] R. H. Untch. On reduced neighborhood mutation analysis using a single mutagenic operator. In *Proceedings of the 47th Annual Southeast Regional Conference*, ACM-SE 47, pages 71:1–71:4, New York, NY, USA, 2009. ACM.

[29] W. Wong and A. P. Mathur. Reducing the cost of mutation testing: An empirical study. *Journal of Systems and Software*, 31(3):185 – 196, 1995.

[30] W. E. Wong. *On mutation and data flow*. PhD thesis, Citeseer, 1993.

[31] L. Zhang, M. Gligoric, D. Marinov, and S. Khurshid. Operator-based and random mutant selection: Better together. In *IEEE/ACM International Conference on Automated Software Engineering*. ACM, 2013.

[32] L. Zhang, S.-S. Hou, J.-J. Hu, T. Xie, and H. Mei. Is operator-based mutant selection superior to random mutant selection? In *Proceedings of the 32Nd ACM/IEEE International Conference on Software Engineering - Volume 1*, ICSE '10, pages 435–444, New York, NY, USA, 2010. ACM.

[33] L. Zhang, S.-S. Hou, J.-J. Hu, T. Xie, and H. Mei. Is operator-based mutant selection superior to random mutant selection? In *Proceedings of the 32nd ACM/IEEE International Conference on Software Engineering-Volume 1*, pages 435–444. ACM, 2010.