REFINING THE IN-PARAMETER-ORDER STRATEGY FOR CONSTRUCTING COVERING ARRAYS

MICHAEL FORBES, JIM LAWRENCE, YU LEI, RAGHU N. KACKER, AND D. RICHARD KUHN

ABSTRACT. Covering arrays are structures for well-representing extremely large input spaces and are used to efficiently implement blackbox testing for software and hardware. This paper proposes refinements over the In-Parameter-Order strategy (for arbitrary t). When constructing homogeneous-alphabet covering arrays, these refinements reduce runtime in nearly all cases by a factor of more than 5 and in some cases by factors as large as 280. This trend is increasing with the number of columns in the covering array. Moreover, the resulting covering arrays are about 5% smaller. Consequently, this new algorithm has constructed many covering arrays that are the smallest in the literature. A heuristic variant of the algorithm sometimes produces comparably sized covering arrays while running significantly faster.

REFERENCES

1. R. C. Bryce and C. J. Colbourn, *The density algorithm for pairwise interaction testing*, Software Testing, Verification, and Reliability, To Appear.
2. Charlie J. Colbourn, *Covering array tables*, Accessed on July 2, 2007.
3. Donald E. Knuth, *The art of computer programming*, vol. 4, ch. 7.2.1.3, Addison-Wesley, 2005.
4. D. R. Kuhn and M. J. Reilly, *An investigation of the applicability of design of experiments to software testing*, Proceedings of the 27th NASA/IEEE Software Engineering Workshop, NASA Goddard Space Flight Center, December 2002.
5. Y. Lei and K. C. Tai, *In-parameter-order: a test generation strategy for pairwise testing*, Proceedings of the Third IEEE International Conferance on High-Assurance Systems Engineering Symposium, 1998, pp. 254–261.
6. Yu Lei, Raghu Kacker, D. Richard Kuhn, Vadim Okun, and James Lawrence, *IPOG: A general strategy for t-way software testing*, ECBS '07: Proceedings of the 14th Annual IEEE International Conference and Workshops on the Engineering of Computer-Based Systems (Washington, DC, USA), IEEE Computer Society, 2007, pp. 549–556.
7. K. C. Tai and Y. Lei, *A test generation strategy for pairwise testing*, IEEE Transactions on Software Engineering **28** (2002), no. 1, 109–111.