Journal of Quality Measurement and Analysis *JQMA* 21(3) 2025, 21-34
e-ISSN: 2600-8602 *http://www.ukm.my/jqma*
https://doi.org/10.17576/jqma.2103.2025.02

INPUT-OUTPUT BASED RELATION COMBINATORIAL TESTING USING WHALE OPTIMIZATION ALGORITHM FOR GENERATING NEAR OPTIMUM NUMBER OF TEST SUITE

(*Pengujian Kombinatorial Perhubungan Berasaskan Input-Output Menggunakan Algoritma Pengoptimuman Paus untuk Menjana Nombor Hampir Optimum Suite Ujian*)

ANJILA J SUALI*, NURAMINAH RAMLI, ROZMIE RAZIF OTHMAN, HASNEEZA LIZA ZAKARIA, ISZAIDY ISMAIL, NOR SHAHIDA MOHD JAMAIL, RIMULJO HENDRADI & NUROL HUSNA CHE ROSE

ABSTRACT

Software testing is necessary to verify that the employed system is confident enough to be used in a specific task according to the required demand. Testing every possible interaction parameter to identify and resolve defects or minimize the number of faults might fall under exhaustive testing. Given the impracticality of testing every possible interaction due to time, budget, and resource constraints, combinatorial testing, which is t-way testing, is adopted to cover parameter interactions efficiently. This research focuses on the Input-Output Based Relations (IOR) testing strategy, which optimizes the test suite size by selecting critical parameters and employing "don't care" values for non-essential inputs. Combinatorial testing offers an alternative to overcome the problem. This study proposes a combinatorial testing method utilizing the Whale Optimization Algorithm (WOA). The study compares the performance of WOA with various existing strategies, such as Greedy, Density, TVG, Union, ParaOrder, ReqOrder, ITTDG, AURA, Java Algorithm (CTJ), TTSGA, and AFA. Experimental results indicate that WOA outperforms or matches the effectiveness of these strategies in generating smaller test suites, particularly for higher interaction strengths. The findings demonstrate WOA's competitive advantage in optimizing combinatorial test suites.

Keywords: Whale Optimization Algorithm, t-way testing, optimization Input-Output Relationship (IOR), artificial intelligence

ABSTRAK

Ujian perisian adalah penting untuk mengesahkan bahawa sistem yang digunakan cukup yakin untuk digunakan dalam tugasan tertentu mengikut permintaan yang diperlukan. Menguji setiap kebarangkalian interaksi parameter untuk mengenal pasti dan menyelesaikan kekurangan atau meminimumkan bilangan kerosakan juga dikenali sebagai ujian menyeluruh. Memandangkan ujian menyeluruh setiap interaksi tidak praktikal disebabkan oleh kekangan masa, belanjawan dan sumber terhad, ujian kombinatoial juga dikenali sebagai ujian *t-way* digunakan untuk mencakupi interaksi parameter dengan efektif. Penyelidikan ini memfokuskan pada strategi ujian Perhubungan Berasaskan Input-Output (IOR), yang mengoptimumkan saiz suite ujian dengan memilih parameter kritikal dan menggunakan nilai "tidak peduli" untuk input yang tidak penting. Ujian kombinatorial menawarkan alternatif untuk mengatasi masalah ujian menyeluruh. Kajian ini mencadangkan kaedah ujian gabungan menggunakan Algoritma Pengoptimuman Paus (WOA). Kajian ini membandingkan prestasi WOA dengan pelbagai strategi sedia ada, seperti Greedy, Density, TVG, Union, ParaOrder, ReqOrder, ITTDG, AURA, Java Algoritma (CTJ), TTSGA, dan AFA. Keputusan eksperimen menunjukkan bahawa WOA mengatasi atau sepadan dengan keberkesanan strategi ini dalam menjana suite ujian yang lebih kecil, terutamanya untuk kekuatan interaksi yang lebih tinggi. Penemuan menunjukkan kelebihan daya saing WOA dalam mengoptimumkan suite ujian gabungan.

Kata kunci: Algoritma Pengoptimuman Paus (WOA); ujian *t-way*; Perhubungan Berasaskan Input-Output (IOR); kecerdasan buatan (AI)

References

Ahmed B.S., Abdulsamad T.S. & Potrus M.Y. 2015. Achievement of minimized combinatorial test suite for configuration-aware software functional testing using the Cuckoo Search algorithm. *Information and Software Technology* **66**: 13–29.

Ali A.S.M., Othman R.R., Yacob Y.M. & Abdelmula H.S.A.B. 2021. An efficient combinatorial input output-based using adaptive firefly algorithm with elitism relations testing. *Advances in Science, Technology and Engineering Systems Journal* **6**(4): 223–232.

Amiriebrahimabadi M. & Mansouri N. 2024. A comprehensive survey of feature selection techniques based on whale optimization algorithm. *Multimedia Tools and Applications* **83**(16): 47775-47846.

Arshem J. 2010. Test Vector Generator (TVG). https://sourceforge.net/projects/tvg/ (30 March 2025).

Bryce R.C. & Colbourn C.J. 2007. The density algorithm for pairwise interaction testing. *Software Testing, Verification and Reliability* **17**(3): 159-182.

Chen X., Gu Q., Li A. & Chen D. 2009. Variable strength interaction testing with an ant colony system approach. *2009 16th Asia-Pacific Software Engineering Conference*, pp. 160–167.

Cohen M.B., Colbourn C. J. & Ling A.C.H. 2003. Augmenting simulated annealing to build interaction test suites. *14th International Symposium on Software Reliability Engineering,* pp. 394–405.

Dorigo M., Birattari M. & Stutzle T. 2006. Ant colony optimization. *IEEE Computational Intelligence Magazine* **1**(14): 28–39.

Esfandyari S. & Rafe V. 2021. GALP: A hybrid artificial intelligence algorithm for generating covering array. *Soft Computing* **25**(11): 7673–7689.

Jahanbin S. & Zamani B. 2018. Test model generation using equivalence partitioning. *2018 8th International Conference on Computer and Knowledge Engineering, ICCKE 2018*, pp. 98–103.

Lei Y., Kacker R., Khun D.R., Okun V. & Lawrence J. 2008. IPOG/IPOG-D: efficient test generation for multi-way combinatorial testing. *Softw. Test. Verif. Reliab* **18**(3): 125–148.

Leveson N. 2020. Are you sure your software will not kill anyone?. *Communications of the ACM* **63**(2): 25–28.

Mirjalili S. 2016. SCA: A Sine Cosine Algorithm for solving optimization problems. *Knowledge-Based Systems* **96**: 120–133.

Mirjalili S. & Lewis A. 2016. The whale optimization algorithm. *Advances in Engineering Software* **95**: 51–67.

Muazu A.A., Hashim A.S. & Sarlan A. 2022. Application and adjustment of "don't care" values in t-way testing techniques for generating an optimal test suite. *Journal of Advances in Information Technology* **13**(4): 347–357.

Nadimi-Shahraki M.H., Zamani H., Asghari Varzaneh Z. & Mirjalili S. 2023. A systematic review of the whale optimization algorithm: Theoretical foundation, improvements, and hybridizations. *Archives of Computational Methods in Engineering* **30**(7): 4113-4159.

Norris K.S. 2023. *Whales, Dolphins, and Porpoises*. Berkeley and Los Angeles, CA: Univ of California Press.

Ong H.Y. & Zamli K.Z. 2011. Development of interaction test suite generation strategy with input-output mapping supports. *Scientific Research and Essays* **6**(16): 3418–3430.

Othman R.R., Khamis N. & Zamli K.Z. 2014. Variable strength t-way test suite generator with constraints support. *Malaysian Journal of Computer Science* **27**(3): 204–217.

Othman R.R. & Zamli K.Z. 2011. ITTDG: Integrated T-way test data generation strategy for interaction testing. *Scientific Research and Essays* **6**(17): 3638–3648.

Othman R.R., Zamli K.Z. & Nugroho L.E. 2012. General variable strength t-way strategy supporting flexible interactions. *Maejo International Journal of Science and Technology* **6**(3): 415–429.

Ramli N., Othman R.R., Abdul Khalib Z.I. & Jusoh M. 2017. A review on recent T-way combinatorial testing strategy. *MATEC Web of Conferences* **140**: 01016.

Ramli N., Othman R.R. & Ali M.S.A.R. 2016. Optimizing combinatorial input-output based relations testing using Ant Colony algorithm. *2016 3rd International Conference on Electronic Design (ICED)*, pp. 586–590.

Schroeder P.J. & Korel B. 2000. Black-box test reduction using input-output analysis. *ACM SIGSOFT Software Engineering Notes* **25**(5): 173–177.

Shiba T., Tsuchiya T. & Kikuno T. 2004. Using artificial life techniques to generate test cases for combinatorial testing. *Proceedings of the 28th Annual International Computer Software and Applications Conference, COMPSAC 2004*, pp. 72–77.

Tatale S. & Prakash V.C. 2022. Combinatorial test case generation using q-value based particle swarm optimization.

Revue d'Intelligence Artificielle **36**(2): 319–326.
Younis M.I., Alsewari A.R.A., Khang N.Y. & Zamli K.Z. 2020. CTJ: Input-output based relation combinatorial testing strategy using jaya algorithm. *Baghdad Science Journal* **17**(3): 1002–1009.
Zamli K.Z., Othman R.R., Younis M.I. & Mohamed Zabil M.H. 2011. Practical adoptions of T-way strategies for interaction testing. *Software Engineering and Computer Systems: Second International Conference, ICSECS 2011, Kuantan, Pahang, Malaysia, June 27-29, 2011, Proceedings, Part III 2*, pp. 1–14.
Zhang J., Zhang Z. & Ma F. 2014. Introduction to combinatorial testing. *Automatic Generation of Combinatorial Test Data*: 1-16. Berlin, Heidelberg: Springer.
Ziyuan W., Changhai N. & Baowen X. 2007. Generating combinatorial test suite for interaction relationship. *SOQUA'07: Fourth International Workshop on Software Quality Assurance - In Conjunction with the 6th ESEC/FSE Joint Meeting*, pp. 55–61

Faculty of Electronic Engineering Technology (FTKEN)
Universiti Malaysia Perlis,
02600 Arau, Perlis, MALAYSIA
E-mail: anjilajsuali91@gmail.com[*]*, husnarose@unimap.edu.my*

Advanced Computing, Centre of Excellence (CoE),
Universiti Malaysia Perlis
02600 Arau, Perlis, MALAYSIA
E-mail: nuraminah@unimap.edu.my, rozmie@unimap.edu.my, hasneeza@unimap.edu.my, iszaidy@unimap.edu.my

Faculty of Computer and Information Sciences,
Prince Sultan University, Riyadh
P.O.Box No. 66833 Rafha Street,
Riyadh, 11586, SAUDI ARABIA
E-mail: Njamail@psu.edu.sa

Faculty of Sciences and Technology,
Universitas Airlangga,
60115, Surabaya Jawa Timur, INDONESIA
E-mail: rimuljohendradi@fst.unair.ac.id

Received: 18 December 2024
Accepted: 2 May 2025

[*]Corresponding author