

The authors would like to express their sincere gratitude to Direktorat Riset, Teknologi, dan Pengabdian kepada Masyarakat (DRTPM) for Grant Contract Number 002/PFR/LPPM UAD/VI/2023 for the financial support. This research would not have been possible without this grant's generous funding and resources. We also thank our colleagues and research collaborators for their valuable insights and contributions during this study.

## References

- [1] A. Dennis, B. H. Wixom, and R. M. Roth, *Systems Analysis and Design*, 7th ed., no. Mi. Wiley, 2019.
- [2] M. Choetkiertikul, H. K. Dam, T. Tran, A. Ghose, and J. Grundy, "Predicting Delivery Capability in Iterative Software Development," *IEEE Trans. Softw. Eng.*, vol. 44, no. 6, pp. 551–573, Jun. 2018, doi: 10.1109/TSE.2017.2693989.
- [3] G. Karner, "Resource Estimation for Objectory Projects." 1993.
- [4] Y. Xie, J. Guo, and A. Shen, "Use Case Points Method of Software Size Measurement Based on Fuzzy Inference," in *Proceedings of the 4th International Conference on Computer Engineering and Networks*, vol. 355, W. E. Wong, Ed. Cham: Springer International Publishing, 2015, pp. 11–18.
- [5] F. Wang, X. Yang, X. Zhu, and L. Chen, "Extended Use Case Points Method for Software Cost Estimation," in *2009 International Conference on Computational Intelligence and Software Engineering*, Dec. 2009, pp. 1–5, doi: 10.1109/CISE.2009.5364706.
- [6] A. B. Nassif, L. F. Capretz, and D. Ho, "Enhancing Use Case Points Estimation Method Using Soft Computing Techniques," *J. Glob. Res. Comput. Sci.*, vol. 1, no. 4, pp. 12–21, Dec. 2010, [Online]. Available: <http://arxiv.org/abs/1612.01078>.
- [7] L. Brezočnik, I. Fister, and V. Podgorelec, "Solving Agile Software Development Problems with Swarm Intelligence Algorithms," in *Lecture Notes in Networks and Systems*, vol. 76, E. Karabegovi, Ed. Sarajevo, Bosnia and Herzegovina: Springer, Cham, 2020, pp. 298–309.
- [8] F. Ferrucci, C. Gravino, R. Oliveto, and F. Sarro, "Genetic Programming for Effort Estimation: An Analysis of the Impact of Different Fitness Functions," *2nd Int. Symp. Search Based Softw. Eng.*, no. 25, pp. 89–98, 2010, doi: 10.1109/SSBSE.2010.20.
- [9] J. Murillo-Morera, C. Quesada-López, C. Castro-Herrera, and M. Jenkins, *A genetic*

- algorithm based framework for software effort prediction*, vol. 5, no. 1. Journal of Software Engineering Research and Development, 2017.
- [10] D. Wu, J. Li, and C. Bao, “Case-based reasoning with optimized weight derived by particle swarm optimization for software effort estimation,” *Soft Comput.*, vol. 22, no. 16, pp. 5299–5310, Aug. 2018, doi: 10.1007/s00500-017-2985-9.
  - [11] T. R. Benala and R. Mall, “DABE: Differential evolution in analogy-based software development effort estimation,” *Swarm Evol. Comput.*, vol. 38, no. May 2016, pp. 158–172, 2018, doi: 10.1016/j.swevo.2017.07.009.
  - [12] M. Dashti, T. J. Gandomani, D. H. Adeg, H. Zulzalil, and A. B. M. Sultan, “LEMABE: a novel framework to improve analogy-based software cost estimation using learnable evolution model,” *PeerJ Comput. Sci.*, vol. 7, no. January, 2021, doi: 10.7717/PEERJ-CS.800.
  - [13] Z. Shahpar, V. K. Bardsiri, and A. K. Bardsiri, “An evolutionary ensemble analogy-based software effort estimation,” *Softw. Pract. Exp.*, vol. 52, no. 4, pp. 929–946, Apr. 2022, doi: 10.1002/spe.3040.
  - [14] Z. Shahpar, V. Khatibi, and A. Khatibi Bardsiri, “Hybrid PSO-SA Approach for Feature Weighting in Analogy-Based Software Project Effort Estimation,” *J. AI Data Min.*, vol. 9, no. 3, pp. 329–340, 2021, doi: 10.22044/jadm.2021.10119.2152.
  - [15] N. Ghatasheh, H. Faris, I. Aljarah, and R. M. H. Al-Sayyed, “Optimizing Software Effort Estimation Models Using Firefly Algorithm,” *J. Softw. Eng. Appl.*, vol. 08, no. 03, pp. 133–142, 2015, doi: 10.4236/jsea.2015.83014.
  - [16] V. Resmi, S. Vijayalakshmi, and R. S. Chandrabose, “An effective software project effort estimation system using optimal firefly algorithm,” *Cluster Comput.*, vol. 22, no. S5, pp. 11329–11338, Sep. 2019, doi: 10.1007/s10586-017-1388-0.
  - [17] K. Langsari and R. Sarno, “Optimizing effort and time parameters of cocomo ii estimation using fuzzy multi-objective PSO,” *Int. Conf. Electr. Eng. Comput. Sci. Informatics*, vol. 4, no. September, pp. 466–471, 2017, doi: 10.11591/eecs.4.1047.
  - [18] N. A. Zakaria, A. R. Ismail, N. Z. Abidin, N. H. M. Khalid, and A. Y. Ali, “Optimized COCOMO parameters using hybrid particle swarm optimization,” *Int. J. Adv. Intell. Informatics*, vol. 7, no. 2, p. 177, Apr. 2021, doi: 10.26555/ijain.v7i2.583.
  - [19] M. D. Alanis-Tamez, C. López-Martín, and Y. Villuendas-Rey, “Particle Swarm

- Optimization for Predicting the Development Effort of Software Projects,” *Mathematics*, vol. 8, no. 10, p. 1819, Oct. 2020, doi: 10.3390/math8101819.
- [20] S. Chhabra and H. Singh, “Optimizing Design of Fuzzy Model for Software Cost Estimation Using Particle Swarm Optimization Algorithm,” *Int. J. Comput. Intell. Appl.*, vol. 19, no. 01, p. 2050005, Mar. 2020, doi: 10.1142/S1469026820500054.
- [21] M. Khazaiepoor, A. Khatibi Bardsiri, and F. Keynia, “A Hybrid Approach for Software Development Effort Estimation using Neural networks, Genetic Algorithm, Multiple Linear Regression and Imperialist Competitive Algorithm,” *Int. J. Nonlinear Anal. Appl.*, vol. 11, no. 1, pp. 2008–6822, 2020, doi: 10.22075/ijnaa.2020.4259.
- [22] K. E. Rao and G. A. Rao, “Ensemble learning with recursive feature elimination integrated software effort estimation: a novel approach,” *Evol. Intell.*, vol. 14, no. 1, pp. 151–162, Mar. 2021, doi: 10.1007/s12065-020-00360-5.
- [23] A. Ardiansyah, R. Ferdiana, and A. E. Permanasari, “Optimizing complexity weight parameter of use case points estimation using particle swarm optimization,” *Int. J. Adv. Intell. Informatics*, vol. 8, no. 2, p. 165, Jul. 2022, doi: 10.26555/ijain.v8i2.811.
- [24] A. Ardiansyah, R. Ferdiana, and A. E. Permanasari, “MUCPSO: A Modified Chaotic Particle Swarm Optimization with Uniform Initialization for Optimizing Software Effort Estimation,” *Appl. Sci.*, vol. 12, no. 3, p. 1081, Jan. 2022, doi: 10.3390/app12031081.
- [25] M. R. Braz and S. R. Vergilio, “Using fuzzy theory for effort estimation of object-oriented software,” in *16th IEEE International Conference on Tools with Artificial Intelligence*, 2004, no. Ictai, pp. 196–201, doi: 10.1109/ICTAI.2004.119.
- [26] G. Robiolo and R. Orosco, “Employing use cases to early estimate effort with simpler metrics,” *Innov. Syst. Softw. Eng.*, vol. 4, no. 1, pp. 31–43, Apr. 2008, doi: 10.1007/s11334-007-0043-y.
- [27] P. Mohagheghi, B. Anda, and R. Conradi, “Effort estimation of use cases for incremental large-scale software development,” in *Proceedings. 27th International Conference on Software Engineering, 2005. ICSE 2005.*, 2005, pp. 303–311, doi: 10.1109/ICSE.2005.1553573.
- [28] B. Anda, H. Dreiem, D. I. K. Sjoberg, and M. Jorgensen, “Estimating Software Development Effort based on Use Cases – Experiences from Industry,” in *The Unified Modeling Language. Modeling Languages, Concepts, and Tools*, C. G. Kobryn, Ed.

Springer Berlin Heidelberg, 2001.

- [29] B. Anda, E. Angelvik, and K. Ribu, “Improving Estimation Practices by Applying Use Case Models,” *Prod. Focus. Softw. Process Improv.*, vol. 2559, no. 1325, pp. 383–397, 2002, doi: 10.1007/3-540-36209-6\_32.
- [30] M. Ochodek, J. Nawrocki, and K. Kwarciak, “Simplifying effort estimation based on Use Case Points,” *Inf. Softw. Technol.*, vol. 53, no. 3, pp. 200–213, Mar. 2011, doi: 10.1016/j.infsof.2010.10.005.
- [31] M. Ochodek, B. Alchimowicz, J. Jurkiewicz, and J. Nawrocki, “Improving the reliability of transaction identification in use cases,” *Inf. Softw. Technol.*, vol. 53, no. 8, pp. 885–897, 2011, doi: 10.1016/j.infsof.2011.02.004.
- [32] H. L. T. K. Nhung, V. Van Hai, R. Silhavy, Z. Prokopova, and P. Silhavy, “Parametric Software Effort Estimation Based on Optimizing Correction Factors and Multiple Linear Regression,” *IEEE Access*, vol. 10, pp. 2963–2986, 2022, doi: 10.1109/ACCESS.2021.3139183.
- [33] A. B. Nassif, D. Ho, and L. F. Capretz, “Towards an early software estimation using log-linear regression and a multilayer perceptron model,” *J. Syst. Softw.*, vol. 86, no. 1, pp. 144–160, Jan. 2013, doi: 10.1016/j.jss.2012.07.050.
- [34] A. B. Nassif, L. F. Capretz, D. Ho, and M. Azzeh, “A treeboost model for software effort estimation based on use case points,” *Proc. - 2012 11th Int. Conf. Mach. Learn. Appl. ICMLA 2012*, vol. 2, no. June, pp. 314–319, 2012, doi: 10.1109/ICMLA.2012.155.
- [35] K. Qi, A. Hira, E. Venson, and B. W. Boehm, “Calibrating use case points using bayesian analysis,” in *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement - ESEM '18*, 2018, pp. 1–10, doi: 10.1145/3239235.3239236.
- [36] R. Silhavy and P. Silhavy, “Using Actors and Use Cases for Software Size Estimation,” *Electronics*, vol. 10, no. 5, pp. 1–20, 2021, doi: 10.3390/electronics10050592.
- [37] H. Le Thi Kim Nhung, H. T. Hoc, and V. Van Hai, “An Evaluation of Technical and Environmental Complexity Factors for Improving Use Case Points Estimation,” in *Advances in Intelligent Systems and Computing*, vol. 1294, 2020, pp. 757–768.
- [38] K. Qi and B. W. Boehm, “Detailed use case points (DUCPs),” in *Proceedings of the 10th International Workshop on Modelling in Software Engineering - MiSE '18*, 2018, pp. 17–

- 24, doi: 10.1145/3193954.3193955.
- [39] D. D. Galorath and M. W. Evans, *Software Sizing, Estimation, and Risk Management*. Boca Raton: Auerbach Publications, 2006.
  - [40] K. Periyasamy and A. Ghode, “Cost Estimation Using Extended Use Case Point (e-UCP) Model,” in *2009 International Conference on Computational Intelligence and Software Engineering*, Dec. 2009, pp. 1–5, doi: 10.1109/CISE.2009.5364515.
  - [41] A. Minkiewicz, “Use Case Sizing.” PRICE Research, 2004.
  - [42] A. B. Nassif, “Software Size and Effort Estimation from Use Case Diagrams Using Regression and Soft Computing Models,” The University of Western Ontario, 2012.
  - [43] A. B. Nassif, L. F. Capretz, and D. Ho, “Calibrating use case points,” in *Companion Proceedings of the 36th International Conference on Software Engineering - ICSE Companion 2014*, 2014, pp. 612–613, doi: 10.1145/2591062.2591141.
  - [44] H. T. Hoc, V. Van Hai, and H. Le Thi Kim Nhung, “AdamOptimizer for the Optimisation of Use Case Points Estimation,” in *Advances in Intelligent Systems and Computing*, vol. 1294, 2020, pp. 747–756.
  - [45] M. Hariyanto and R. S. Wahono, “Estimasi Proyek Pengembangan Perangkat Lunak Dengan Fuzzy Use Case Points,” *J. Softw. Eng.*, vol. 1, no. 1, pp. 54–63, 2015.
  - [46] S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey Wolf Optimizer,” *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014, doi: 10.1016/j.advengsoft.2013.12.007.
  - [47] X. S. Yang, “Firefly algorithms for multimodal optimization,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5792 LNCS, pp. 169–178, 2009, doi: 10.1007/978-3-642-04944-6\_14.
  - [48] Xin-She Yang, *Nature Inspired Metaheuristic Algorithms*, 2nd ed. Luniver Press, 2010.
  - [49] J. H. Holland, *Adaptation in Natural and Artificial Systems*. London: MIT Press, 1992.
  - [50] R. Silhavy, P. Silhavy, and Z. Prokopova, “Analysis and selection of a regression model for the Use Case Points method using a stepwise approach,” *J. Syst. Softw.*, vol. 125, pp. 1–14, Mar. 2017, doi: 10.1016/j.jss.2016.11.029.
  - [51] E. Kocaguneli and T. Menzies, “Software effort models should be assessed via leave-one-out validation,” *J. Syst. Softw.*, vol. 86, no. 7, pp. 1879–1890, 2013, doi: 10.1016/j.jss.2013.02.053.
  - [52] W. B. Langdon, J. Dolado, F. Sarro, and M. Harman, “Exact Mean Absolute Error of

- Baseline Predictor, MARP0,” *Inf. Softw. Technol.*, vol. 73, pp. 16–18, May 2016, doi: 10.1016/j.infsof.2016.01.003.
- [53] M. Shepperd and S. MacDonell, “Evaluating prediction systems in software project estimation,” *Inf. Softw. Technol.*, vol. 54, no. 8, pp. 820–827, Aug. 2012, doi: 10.1016/j.infsof.2011.12.008.
- [54] M. H. Nadimi-Shahraki, S. Taghian, and S. Mirjalili, “An improved grey wolf optimizer for solving engineering problems,” *Expert Syst. Appl.*, vol. 166, no. August 2020, p. 113917, 2021, doi: 10.1016/j.eswa.2020.113917.
- [55] H. Peng, W. Zhu, C. Deng, and Z. Wu, “Enhancing firefly algorithm with courtship learning,” *Inf. Sci. (Ny)*, vol. 543, pp. 18–42, 2020, doi: 10.1016/j.ins.2020.05.111.
- [56] L. Abualigah, M. A. Elaziz, P. Sumari, Z. W. Geem, and A. H. Gandomi, “Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer,” *Expert Syst. Appl.*, vol. 191, no. November, p. 116158, Apr. 2022, doi: 10.1016/j.eswa.2021.116158.
- [57] M. Azzeh and A. B. Nassif, “A hybrid model for estimating software project effort from Use Case Points,” *Appl. Soft Comput. J.*, vol. 49, pp. 981–989, 2016, doi: 10.1016/j.asoc.2016.05.008.
- [58] R. Silhavy, P. Silhavy, and Z. Prokopova, “Evaluating subset selection methods for use case points estimation,” *Inf. Softw. Technol.*, vol. 97, no. June 2017, pp. 1–9, 2018, doi: 10.1016/j.infsof.2017.12.009.
- [59] J. Demsar, “Statistical Comparisons of Classifiers over Multiple Data Sets,” *J. Mach. Learn. Res.*, vol. 7, pp. 1–30, 2006.
- [60] I. Boussaïd, J. Lepagnot, and P. Siarry, “A survey on optimization metaheuristics,” *Inf. Sci. (Ny)*, vol. 237, pp. 82–117, Jul. 2013, doi: 10.1016/j.ins.2013.02.041.
- [61] A. H. Halim, I. Ismail, and S. Das, *Performance assessment of the metaheuristic optimization algorithms: an exhaustive review*, vol. 54, no. 3. Springer Netherlands, 2021.
- [62] J. O. Agushaka, A. E. Ezugwu, L. Abualigah, S. K. Alharbi, and H. A. E. W. Khalifa, “Efficient Initialization Methods for Population-Based Metaheuristic Algorithms: A Comparative Study,” *Archives of Computational Methods in Engineering*, vol. 30, no. 3, pp. 1727–1787, 2023, doi: 10.1007/s11831-022-09850-4.
- [63] M. A. Tawhid and A. M. Ibrahim, “Improved salp swarm algorithm combined with chaos,” *Math. Comput. Simul.*, vol. 202, pp. 113–148, 2022, doi:

10.1016/j.matcom.2022.05.029.