5. CONCLUSIONS AND FUTURE SCOPE

In this article nature-inspired, Modified ACO optimization algorithm is used to reduce the test time of SoCs. The efficiency of the Modified ACO algorithm is realized by comparing it with the ACO algorithm. Experiments with d695 and p22810 SoC comparisons have shown that the modified ACO algorithm is better than the ACO algorithm. In the case of d695, the Modified ACO algorithm achieves a reduction in test time of 47% when compared to the ACO algorithm, wherein the p22810 case, the Modified ACO algorithm achieves a reduction in test time of 10% when compared to ACO algorithm. The results of the experiment clearly indicate that the modified ACO algorithm is better suited to reducing the test time. In future recent algorithms like Grey Wolf Optimization Algorithm, Whale Optimization Algorithm, Dragonfly Algorithm and Artificial Fish Swarm Algorithm can be used to minimize the test time further.

REFERENCES

[1] Pouget, J., Larsson, E., Peng, Z.B. (2005). Multiple-constraint driven system-on-chip test time optimization. Journal of Electronic Testing, 21(6): 599-611. https://doi.org/10.1007/s10836-005-2911-4

[2] Kang, W.J., Hwang, S.Y. (2014). A test wrapper design to reduce test time for multi-core SoC. The Journal of Korean Institute of Communications and Information Sciences, 39(1): 1-7. https://doi.org/10.7840/kics.2014.39B.1.1

[3] Marrouche, W., Farah, R., Harmanani, H.M. (2018). A strength pareto evolutionary algorithm for optimizing system-on-chip test schedules. International Journal of Computational Intelligence and Applications, 17(02): 1850010. https://doi.org/10.1142/S1469026818500104

[4] Chakrabarty, K. (2000). Test scheduling for core-based systems using mixed-integer linear programming. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 19(10): 1163-1174. https://doi.org/10.1109/43.875306

[5] Iyengar, V., Chakrabarty, K. (2002). System-on-a-chip test scheduling with precedence relationships, preemption, and power constraints. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 21(9): 1088-1094. https://doi.org/10.1109/TCAD.2002.801102

[6] Chakrabarty, K., Marinissen, E.J. (2003). Test access mechanism optimization, test scheduling, and tester data volume reduction for system-on-chip. IEEE Transactions on Computers, 52(12): 1619-1632. https://doi.org/10.1109/TC.2003.1252857

[7] Wang, Z., Chakrabarty, K., Wang, S. (2009). Integrated LFSR reseeding, test-access optimization, and test scheduling for core-based system-on-chip. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 28(8): 1251-1264. https://doi.org/10.1109/TCAD.2009.2021731

[8] Zou, W., Reddy, S.M., Pomeranz, I., Huang, Y. (2003). SOC test scheduling using simulated annealing. In Proceedings IEEE VLSI Test Symposium, 325-330. https://doi.org/10.1109/VTEST.2003.1197670

[9] Im, J.B., Chun, S., Kim, G., An, J.H., Kang, S. (2004). RAIN (Random INsertion) scheduling algorithm for SOC Test. IEEE Asian Test Symposium, 242-247. https://doi.org/10.1109/ATS.2004.71

[10] Chakrabarty, K., Iyengar, V., Krasniewski, M.D. (2005). Test planning for modular testing of hierarchical SOCs. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 24(3): 435-448. https://doi.org/10.1109/TCAD.2004.842816

[11] Goel, S., Marinissen, E.J., Sehgal, A., Chakrabarty, K. (2008). Testing of SoCs with hierarchical cores: common fallacies, test access optimization, and test scheduling. IEEE Transactions on Computers, 58(3): 409-423. https://doi.org/10.1109/TC.2008.169

[12] Yang, Q., Chen, W.N., Yu, Z., Gu, T., Li, Y., Zhang, H., Zhang, J. (2016). Adaptive multimodal continuous ant colony optimization. IEEE Transactions on Evolutionary Computation, 21(2): 191-205. https://doi.org/10.1109/TEVC.2016.2591064

[13] Dorigo, M., Blum, C. (2005). Ant colony optimization theory: A survey. Theoretical Computer Science, 344(2-3): 243-278. https://doi.org/10.1016/j.tcs.2005.05.020

[14] Wang, Z.Y., Xing, H.L., Li, T.R., Yang, Y., Qu, R., Pan, Y. (2015). A modified ant colony optimization algorithm for network coding resource minimization. IEEE Transactions on Evolutionary Computation, 20(3): 325-342. https://doi.org/10.1109/TEVC.2015.2457437

[15] Liu, J.H., Yang, J.G., Liu, H.P., Tian, X.J., Gao, M. (2017). An improved ant colony algorithm for robot path planning. Soft Computing, 21(19): 5829-5839. https://doi.org/10.1007/s00500-016-2161-7

[16] Dorigo, M., Stützle, T. (2019). Ant colony optimization: overview and recent advances. In Handbook of Metaheuristics, Springer, Cham, 311-351. https://doi.org/10.1007/978-3-319-91086-4_10