Random (DSJ) graphs which are created from David Johnson and R250_5 graph are difficult to solve benchmark graphs [15]. Flat graphs are created from Culberson. [26]. First parameter of the Flat graphs represents the number of the vertices and the second parameter represents the chromatic number.

Table 12 shows the results for Random and Flat graphs. According to the Table 12, RLF and DSATUR algorithms generally find out the best/χ(G) results. For the R250_5 graph, DSATUR algorithm's computation time is further than RLF's. But DSATUR finds out quite better result than RLF for R250_5 graph. On the other hand, RLF finds out quite better results for the other graphs and RLF is faster than the other algorithms.

fpsol2*, inithx*, zeroin* and mulsol* graphs are computer register allocation problem graphs which are generated from Gary Lewandowski [24]. These graphs are real-world problem's graphs. The computer registers and the operations are defined as vertices. If a register and an operation have a relationship, there is an edge generated between them.

Table 13 shows the results for computer register allocation graphs. All algorithms which are used for this study reach the χ(G) results for computer register allocation graphs. If the algorithms compare to each other about their computation times, the best algorithm for register allocation graphs is WP algorithm and also FF algorithm reaches the χ(G) results a quite short times. The slowest algorithm for these graphs is DSATUR algorithm.

**Table 12**.The results and computation times for Random and Flat graphs

| **Graf** | **V** | **E** | **Den.** | **Eniyi/ χ(G)** | **RLF** | | **DSATUR** | | **WP** | | **LDO** | | **IDO** | | **FF** | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | **R** | **T** | **R** | **T** | **R** | **T** | **R** | **T** | **R** | **T** | **R** | **T** |
| DSJC125_1 | 125 | 736 | 0,09 | 5 | **6** | **0,0135** | **6** | 0,0846 | 7 | 0,0005 | 7 | 0,0032 | 7 | 0,0320 | 8 | 0,0024 |
| DSJC125_5 | 125 | 3891 | 0,50 | 17 | **21** | **0,0468** | 22 | 0,6111 | 23 | 0,0011 | 23 | 0,0079 | 25 | 0,2966 | 26 | 0,0074 |
| DSJC125_9 | 125 | 6961 | 0,89 | 44 | **49** | **0,1811** | 51 | 1,4215 | 53 | 0,0019 | 53 | 0,0162 | 54 | 0,7575 | 56 | 0,0154 |
| DSJC250_1 | 250 | 3218 | 0,10 | 8 | **10** | **0,0665** | **10** | 0,4791 | 11 | 0,0011 | 11 | 0,0142 | 12 | 0,2086 | 13 | 0,0097 |
| DSJC250_5 | 250 | 15668 | 0,50 | 28 | **35** | **0,4661** | 37 | 4,9399 | 41 | 0,0025 | 41 | 0,0371 | 40 | 3,0145 | 43 | 0,0394 |
| DSJR500_1 | 500 | 3555 | 0,03 | 12 | **12** | **0,2863** | 13 | 0,5829 | 13 | 0,0023 | 13 | 0,0237 | 13 | 0,2609 | 15 | 0,0199 |
| R250_5 | 250 | 14849 | 0,48 | 65 | 71 | 0,7803 | **68** | **4,6108** | 70 | 0,0034 | 70 | 0,0439 | 69 | 2,7790 | 79 | 0,0478 |
| flat300_20 | 300 | 21375 | 0,48 | 20 | **38** | **0,7199** | 42 | 8,5125 | 44 | 0,0032 | 44 | 0,0554 | 45 | 5,3253 | 47 | 0,0609 |
| flat300_26 | 300 | 21633 | 0,48 | 26 | **39** | **0,8435** | 41 | 8,5990 | 45 | 0,0030 | 45 | 0,0566 | 48 | 5,5501 | 45 | 0,0610 |
| flat300_28 | 300 | 21695 | 0,48 | 28 | **38** | **0,8624** | 42 | 8,6611 | 45 | 0,0030 | 45 | 0,0564 | 48 | 5,5324 | 46 | 0,0613 |

R: Result of the algorithm, T: Computation time (in second)

**Table 13.** The results and computation times for Register Allocation graphs

| **Graph** | **V** | **E** | **Den.** | **Best/χ(G)** | **RLF** | | **DSATUR** | | **WP** | | **LDO** | | **IDO** | | **FF** | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | **R** | **T** | **R** | **T** | **R** | **T** | **R** | **T** | **R** | **T** | **R** | **T** |
| fpsol2_i1 | 496 | 11654 | 0,09 | 65 | 65 | 0,9869 | 65 | 3,1791 | 65 | **0,0044** | 65 | 0,0646 | 65 | 1,8096 | 65 | 0,0552 |
| fpsol2_i2 | 451 | 8691 | 0,09 | 30 | 30 | 0,5217 | 30 | 1,9960 | 30 | **0,0024** | 30 | 0,0442 | 30 | 1,1139 | 30 | 0,0409 |
| fpsol2_i3 | 425 | 8688 | 0,10 | 30 | 30 | 0,5184 | 30 | 1,9752 | 30 | **0,0022** | 30 | 0,0427 | 30 | 1,0739 | 30 | 0,0407 |
| mulsol_i1 | 197 | 3925 | 0,20 | 49 | 49 | 0,1299 | 49 | 0,6347 | 49 | **0,0021** | 49 | 0,0153 | 49 | 0,2924 | 49 | 0,0137 |
| mulsol_i2 | 188 | 3885 | 0,22 | 31 | 31 | 0,1171 | 31 | 0,6423 | 31 | **0,0015** | 31 | 0,0145 | 31 | 0,2899 | 31 | 0,0133 |
| mulsol_i3 | 184 | 3916 | 0,23 | 31 | 31 | 0,1164 | 31 | 0,6189 | 31 | **0,0015** | 31 | 0,0143 | 31 | 0,2805 | 31 | 0,0134 |
| mulsol_i4 | 185 | 3946 | 0,23 | 31 | 31 | 0,1243 | 31 | 0,6328 | 31 | **0,0015** | 31 | 0,0145 | 31 | 0,2994 | 31 | 0,0130 |
| mulsol_i5 | 186 | 3973 | 0,23 | 31 | 31 | 0,1253 | 31 | 0,6286 | 31 | **0,0015** | 31 | 0,0145 | 31 | 0,2900 | 31 | 0,0128 |
| inithx_i1 | 864 | 18707 | 0,05 | 54 | 54 | 2,7427 | 54 | 6,7614 | 54 | **0,0066** | 54 | 0,1337 | 54 | 4,2802 | 54 | 0,1266 |
| inithx_i2 | 645 | 13979 | 0,07 | 31 | 31 | 1,4014 | 31 | 4,2319 | 31 | **0,0037** | 31 | 0,0839 | 31 | 2,5214 | 31 | 0,0800 |
| inithx_i3 | 621 | 13969 | 0,07 | 31 | 31 | 1,3034 | 31 | 4,1724 | 31 | **0,0035** | 31 | 0,0819 | 31 | 2,5577 | 31 | 0,0780 |
| zeroin_i1 | 211 | 4100 | 0,18 | 49 | 49 | 0,1427 | 49 | 0,6636 | 49 | **0,0020** | 49 | 0,0157 | 49 | 0,3188 | 49 | 0,0139 |
| zeroin_i2 | 211 | 3541 | 0,16 | 30 | 30 | 0,1062 | 30 | 0,5390 | 30 | **0,0014** | 30 | 0,0136 | 30 | 0,2504 | 30 | 0,0124 |
| zeroin_i3 | 206 | 3540 | 0,17 | 30 | 30 | 0,1150 | 30 | 0,5439 | 30 | **0,0014** | 30 | 0,0134 | 30 | 0,2530 | 30 | 0,0123 |

R: Result of the algorithm, T: Computation time (in second)

In the Leighton graphs, each graph consists of 450 vertices. First parameter of the Leighton graphs represents the number of the vertices and the second parameter represents the chromatic number [20]. Table 14 shows the results for Leighton graphs. Experimental results show that RLF algorithm finds out quite better results for Leighton graphs. Just for le450_25b graph, WP algorithm finds out the χ(G) result the better computation time. The other algorithms are generally deficient.

**Table 14.** The results and computation times for Leighton graphs

| **Graph** | **V** | **E** | **Den.** | **Eniyi/χ(G)** | **RLF** | | **DSATUR** | | **WP** | | **LDO** | | **IDO** | | **FF** | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | **R** | **T** | **R** | **T** | **R** | **T** | **R** | **T** | **R** | **T** | **R** | **T** |
| le450_15b | 450 | 8169 | 0,08 | 15 | 17 | 0,3071 | **16** | 1,7589 | 18 | 0,0025 | 18 | 0,0348 | 18 | 0,9585 | 22 | 0,0337 |
| le450_25a | 450 | 8260 | 0,08 | 25 | **25** | **0,3502** | **25** | 1,7952 | 26 | 0,0029 | 26 | 0,0367 | 25 | 1,0172 | 28 | 0,0355 |
| le450_25b | 450 | 8263 | 0,08 | 25 | 25 | 0,3583 | 25 | 1,9924 | **25** | **0,0028** | 25 | 0,0371 | 25 | 1,0341 | 27 | 0,0355 |
| le450_25c | 450 | 17343 | 0,17 | 25 | **28** | **0,7839** | 29 | 5,9978 | 29 | 0,0034 | 29 | 0,0626 | 31 | 3,6658 | 37 | 0,0674 |
| le450_5c | 450 | 9803 | 0,10 | 5 | **5** | **0,2226** | 10 | 2,4336 | 12 | 0,0020 | 12 | 0,0352 | 12 | 1,3233 | 17 | 0,0375 |
| le450_5d | 450 | 9757 | 0,10 | 5 | **6** | **0,2315** | 12 | 2,4073 | 14 | 0,0025 | 14 | 0,0362 | 13 | 1,2504 | 18 | 0,0382 |

R: Result of the algorithm, T: Computation time (in second)

## 5. Conclusion

Experimental results show that while RLF and DSATUR algorithms are sufficient for the GCP, FF algorithm is generally deficient. WP algorithm finds out the best solution in the shortest time on Register Allocation, CAR, Mycielski, Stanford Miles, Book and Game graphs. On the other hand, RLF algorithm is quite better than the other algorithms on Leighton, Flat, Random (DSJC) and Stanford Queen graphs. As shown in the study, firstly it should be decided that the problems which we want solve with graph coloring algorithms is similar to what benchmark graphs. After that, the optimum graph coloring algorithms must be applied to the problem for finds out the the best solution. Thus, it can be avoided to waste of times and it can be reached the best results a quite short time.

## Acknowledge

This study was supported by "Scientific Research Projects of Selcuk University". This paper has been presented as an oral presentation at the International Conference on Advanced Technology&Sciences (ICAT'16) held in KONYA (Turkey), September 01-03, 2016 and selected for the International Journal of Intelligent Systems and Applications in Engineering (IJISAE).

## References

[1] D. B. West, Introduction to Graph Theory, *Prentice Hall, U. S. A., 588 pp,* 2001.

[2] J. L. Gross And J. Yellen, Graph Theory and Its Applications, CRC Press, Mathematics, 600 pages, 1998.

[3] R. Fritsch, and G. Fritsch, The Four-Color Theorem: History, Topological Foundations and Idea of Proof, Newyork: Springer, pages 260, 1998.

[4] F. Ge, Z. Wei, Y. Tian, Z. Huang, Chaotic Ant Swarm for Graph Coloring , Intelligent Computing and Intelligent Systems (ICIS), IEEE International Conference, 512-516 p., 2010.

[5] D. J. Welsh, and M. B. Powell, An upper bound for the chromatic number of a graph and its application to timetabling problems, The Computer Journal 10 (1): 85–86, 1967.

[6] I. M. Díaz and P. Zabala, A Generalization of the Graph Coloring Problem, Departamento de Computacion, Universidad de Buenes Aires, 1999.

[7] B. H. Gwee, M. H. Limand and J. S. Ho, Solving fourcolouring map problem using genetic algorithm. In Proceedings of First New Zealand International Two-Stream Conference on Artificial Neural Networks and Expert Systems, 332-333, 1993.

[8] N. Chmait, and K. Challita, Using Simulated Annealing and Ant-Colony Optimization Algorithms to Solve the Scheduling Problem, Computer Science and Information Technology 1(3), 208-224, 2013.

[9] K. Dowsland, J. Thompson, Ant colony optimization for the examination scheduling problem, J. Oper. Res. Soc. 56, 426–438, 2005.

[10] G.J. Chaitin, M.A. Auslander, A.K. Chandra, J. Cocke, , M.E. Hopkins and P.W. Mark-stein, Register allocation via coloring, Comput. Lang. 6 47–57, 1981.

[11] F.C. Chow, J.L. Hennessy, The priority based coloring approach to register allocation, ACM Trans. Program. Lang. Syst, 501–536, 1990.

[12] S. Ono, R. Miyamoto, S. Nakayama, and K. Mizuno, "Difficulty estimation of number place puzzle and its problem generation support." ICCAS-SICE, 2009. IEEE, 2009.

[13] W. K. Hale, Frequency assignment: *Theory and applications. Proceedings of the IEEE 12: 1497-1514*, 1980.

[14] M.R. Garey and D.S. Johnson, Computers and intractability, in: A Guide to the Theoryof NP-Completeness, W.H. Freeman & Co., New York, NY, U.S.A., 1979.

[15] S. Mahmoudi and S. Lotfi, Modified cuckoo optimization algorithm (MCOA) to solve graph coloring problem, *Applied soft computing Volume 33, 48–64*, 2015.

[16] A. Hertz and D. de Werra, Using tabu search techniques for graph coloring, *Computing 39, 345–351*, 1987.

[17] M. Chams, A. Hertz and D. de Werra, Some experiments with simulated annealing for coloring graphs, *European Journal of Operational Research, 32(2):260-266*, 1987.

[18] S. Ahn, S. Lee, T. Chung, Modified Ant Colony System for Coloring Graphs, *Information, Communications and Signal Processing, 2003 and Fourth Pacific Rim Conference on Multimedia. Proceedings of the Joint Conference of the Fourth International Conference on, IEEE, 1849 – 1853,* 2003.

[19] H. Al-Omari and K.E. Sabri, New Graph Coloring Algorithms, *American Journal of Mathematics and Statistics 2 (4): 739-741*, 2006.

[20] F.T. Leighton, A graph coloring algorithm for large scheduling problems, *J. Res.Nat. Bur. Stand. 84 489–506*, 1979.

[21] D. Brélaz, New methods to color the vertices of a graph, *Commun. ACM 22 251–256*, 1979.

[22] DIMACS graph coloring instances. (2016) instances homepage on CMU. [online]. Available: http:mat.gsia.cmu.edu/COLOR/instances.html.

[23] I. C. R. Ruiz, "Gravitational Swarm for Graph Coloring", PHD thesis, The University of the Basque Country Donostia - San Sebastian, 2012.

[24] D. S. Johnson and M. A. Trick, Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge, 1993, vol. 26 of DIMACS Series in Discrete Mathematics and Theoretical Computer Science, American Mathematical Society. 1996.

[25] M. Caramia and P. Dell'Olmo. A lower bound on the chromatic number of mycielski graphs. Discrete Mathematics, 235(1-3):79_86, 2001.

[26] B. Yılmaz, A novel meta-heuristic for graph coloring problem: simulated annealing with backtracking, Yeditepe University Graduate School Of Natural Scıences, Istanbul, 2011.