

Evaluation of Metaheuristics for Robust Graph Coloring Problem

Zbigniew Kokosiński and Łukasz Ochał

Cracow University of Technology,

Faculty of Electrical and Computer Eng.,

Dept. of Automatic Control and Information Technology,

Warszawska 24, 31-155 Cracow, Poland

Email: {zk@pk.edu.pl; lukasz.ochal@gmail.com}

Abstract—In this paper a new formulation of the robust graph coloring problem (RGCP) is proposed. In opposition to classical GCP defined for the given graph $G(V,E)$ not only elements of E but also \bar{E} can be subject of color conflicts in edge vertices. Conflicts in \bar{E} are assigned penalties $0 < P(e) < 1$. In addition to satisfying constraints related to the number of colors and/or a threshold of the acceptable sum of penalties for color conflicts in graph complementary edges (rigidity level), a new bound called the relative robustness threshold (RRT) is proposed. Then two metaheuristics – SA, TS and their parallel analogues PSA and PTS – for that version of RGCP are presented and experimentally compared. For comparison we use DIMACS graph coloring instances in which a selected percentage of graph edges E is randomly moved to \bar{E} . Since graph densities and chromatic numbers of DIMACS GCP instances are known in advance, the RGCP instances generated on their basis are more suitable for testing algorithms than totally random instances used so far. The results of the conducted experiments are presented and discussed.

I. INTRODUCTION

THE classical graph k -colorability problem belongs to the class of NP-hard combinatorial problems [9]. This decision problem is defined for an undirected graph $G=(V,E)$ and positive integer $k \leq |V|$: is there an assignment of available k colors to graph vertices, providing that adjacent vertices receive different colors? In optimization version of the basic problem called GCP, a conflict-free coloring with minimum number of colors k is searched.

Many particular colorings of graph vertices and/or edges represent solutions of variety of practical problems that can be modeled by graphs with specific constraints put on the elements of the sets V and E . With additional assumptions many variants of the coloring problem can be defined such as equitable coloring, sum coloring, contrast coloring, harmonious coloring, circular coloring, consecutive coloring, list coloring, total coloring etc. [14], [18].

One of the most interesting variants of GCP is the robust graph coloring problem (RGCP) [24]. It models a class of vertex coloring problems in which adjacency relation between graph vertices is not “stable”. In certain circumstances nonadjacent vertices u and v can become adjacent and the

edge (u,v) is assigned a penalty $0 < P(u,v) < 1$ when there is a color conflict: $c(u)=c(v)$. If all penalties $P(u,v)$ are known it is possible to define requirements for solution feasibility. A conflict-free coloring of all vertices in E is required, while a number of penalized color conflicts in the set of complementary edges \bar{E} , not exceeding certain threshold (f.i. rigidity level) can be tolerated. Given a number of colors k , the coloring with a lower rigidity level is more robust. In general, feasibility conditions can be expressed in terms of the maximum number of colors used and an upper bound on a cost function.

Similarly as GCP also RGCP is known to be NP-hard [26]. Therefore, application of approximate algorithms and metaheuristics for solving this problem is reasonable [1], [10], [11]. In literature no r -approximation algorithms for RGCP were reported so far. Research conducted in this area contains a number of algorithms and metaheuristics for RGCP [2], [19], [20], [24], [25] a new formulation of specific robust coloring problems [4] and a combination of system robustness and fuzziness [8], [12]. Research results were gathered and summarized in [24]. Other recent papers on system robustness are [7] and [23].

In research papers [19], [20] there appears a problem in experimental verification of the investigated metaheuristic methods. How to measure quality of the solution, when nothing is known about chromatic properties of the given graph? How to validate the assumed penalty threshold for a feasible solution? In our approach RGCP is redefined in order to allow the system designer to use a new cost function – the relative robustness of the solution, which can be expressed by a percentage – the relative robustness $RR=100\%$ means a conflict-free vertex coloring in both edges E and complementary edges \bar{E} with $P(e) > 0$. This view is very natural and meets common expectations of system designers. For experimental verification two metaheuristics – Tabu Search (TS), Simulated Annealing (SA) and their parallel versions – PTS and PTA – are used. In standard and parallel versions they were applied earlier in similar research [3], [6], [16], [17], [19], [20], [21], [22]. As input graphs we used DIMACS graph coloring instances [27], [28], [29] in

which a constant percentage of graph edges E , denoted by E' is assigned penalties $0 < P(e) < 1$. Since graph densities and chromatic numbers of DIMACS GCP instances are known in advance [15], the RGCP instances generated on their basis are more suitable for testing algorithms than totally random instances used so far. The presented results justify both theoretical assumptions and application of parallel metaheuristics for solving RGCP problem.

In the next section RGCP problem is defined together with its new formulation. TS/PTS and SA/PSA algorithms and their parameters are presented in sections III. Then, in section IV, computer experiments are described and their results discussed. In conclusion some general suggestions related to the obtained results and future research in this area are derived.

II. ROBUST GRAPH COLORING PROBLEM

GCP is defined for an undirected graph $G(V, E)$ as an assignment of available colors $\{1, \dots, k\}$ to graph vertices providing that adjacent vertices receive different colors and the number of colors k is minimal. The resulting coloring c is called conflict-free and k is called graph chromatic number $\chi(G)$.

A. RGCP – a simple formulation

RGCP is defined for undirected weighted graph $G(V, E)$ with function $w(e) = p_{uv} \in [0, 1]$, as an assignment of available colors $\{1, \dots, k\}$ to graph vertices, providing that

$$\forall (u, v) \in E : (p_{uv} = 1) \Rightarrow c(u) \neq c(v) \quad (1)$$

and rigidity level for \bar{E} is minimum

$$RL(c) = \sum_{((u,v) \in \bar{E}) \wedge (c(u)=c(v)) \wedge (p(u,v) > 0)} p_{uv} \quad (2)$$

In some cases weight (penalty) p_{uv} may be considered as a probability of edge existence; in the classical vertex coloring $p_{uv} \in \{0, 1\}$.

For most practical problems it suffices, that

$$RL(c) \leq T \quad (3)$$

where :

T – is an assumed threshold.

The question is what value of T is reasonable for the modeled problem? How it reflects system robustness? What level of T should be guaranteed for the given system? In order to answer such questions an alternative formulation of RGCP problem is proposed. The alternative formulation of RGCP does not change the nature of the problem, but allows the system designer to apply the relative robustness level instead of the absolute robustness level which is not known in advance.

B. RGCP – an alternative formulation

Let us characterize system robustness more precisely. The robustness threshold is given by the following formula

$$RT(c) = \frac{RRT(c)}{100\%} \sum_{(e \in \bar{E}) \wedge (p((u,v)) > 0)} p_{uv}, \quad (4)$$

where:

$RRT(c)$ – is a relative robustness threshold set up by the designer and expressed in [%].

Thus, our optimization goal is to find a coloring c satisfying equation (1) and inequalities (5) and (6):

$$\sum_{((u,v) \in \bar{E}) \wedge (c(u) \neq c(v)) \wedge (p(u,v) > 0)} p_{uv} \geq RT(c) \quad (5)$$

$$\frac{\sum_{((u,v) \in \bar{E}) \wedge (c(u) \neq c(v)) \wedge (p(u,v) > 0)} p_{uv}}{\sum_{((u,v) \in \bar{E}) \wedge (p(u,v) > 0)} p_{uv}} \cdot 100\% \geq RRT(c) \quad (6)$$

C. Generation of RGCP instances

Parametrized RGCP instances can be generated by a random modification of GCP instances: a percentage of $E' \in E$ is moved to \bar{E} with weights $0 < p(u, v) < 1$. In Fig. 1 three out of seven edges are selected at random and assigned new values $p(u, v)$.

D. Cost function for RGCP

In problem formulation the priority is given to conflict-free coloring of edges with $P(u, v) = 1$. Otherwise, the solution is not feasible. Feasible solutions have the cost function:

$$cf(c) = \sum_{(u,v) \in E \cup \bar{E}} q_{uv} \quad (7)$$

where:

$$q_{uv} = \begin{cases} 1, & \text{if } (c(u) = c(v) \text{ and } p_{uv} = 1) \\ p_{uv} & \text{if } (c(u) = c(v) \text{ and } p_{uv} < 1) \\ 0 & \text{if } (c(u) \neq c(v)) \end{cases} \quad (8)$$

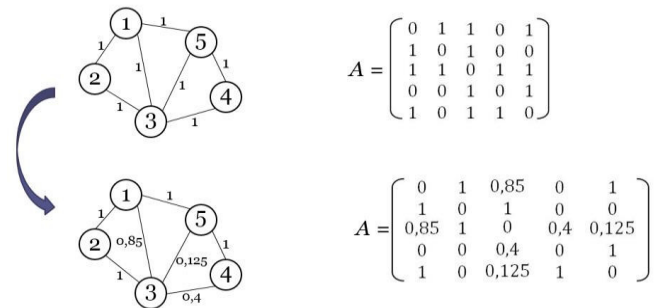


Fig. 1 Generation of RGCP graph instances from GCP instances (42,86% of E moved to \bar{E} ; \bar{E} percentage increased from 30 to 60%).

III. PTS AND PSA METAHEURISTICS

The applications of basic metaheuristics for RGCP was reported in [19]. The first parallel metaheuristic for RGCP – Parallel Evolutionary Algorithm – was presented in [5]. In the present paper we deal with two other popular parallel metaheuristics PTS and PSA. The details of implementation are skipped here for the sake of brevity. In order to determine their parameters at first we investigate algorithms TS and SA.

A. Parameters for Tabu Search Algorithm

Tabu Search metaheuristic presented in [19] is adapted for parallelization. PTS algorithm includes three TS processes that periodically exchange information when 1/3 and 2/3 of the required RR is obtained. There are at least two key parameters of TS/PTS algorithms that have to be set [6]: tMAX and MaxTenure. This parameters were found experimentally. The results of conducted experiments are shown in Table I and Table II.

The values of parameters recommended for TS and PTS algorithms are as follows: tMAX=10 and MaxTenure=15. As a selection criterion majority of optimum solutions with respect to relative robustness RR was used.

B. Parameters for Simulated Annealing Algorithm

A Simulated Annealing metaheuristic for RGCP presented in [19] is adapted for parallelization. There are three important parameters of SA and PSA algorithms that have to be set [21]: MinIteration, ControlFactor (speed of convergence) and Tmax. These parameters were also found experimentally. The results of conducted experiments are shown in Tables III, IV, and V. We can assume Tmin=0,25.

PSA algorithm includes also three SA processes that periodically exchange information when 1/3 and 2/3 of the required RR is obtained. All processes resume computations with new best solution. The values of parameters recommended for PSA are the following: MinIteration=5, ControlFactor=0,9 and Tmax=10.

Table I Efficiency of TSA with tMax (MaxTenure=10)

Graphs	tMax : 5			tMax : 10			tMax : 15		
	c.f.	time [s]	RR [%]	c.f.	time [s]	RR [%]	c.f.	time [s]	RR [%]
queen5.5_40 $\chi(g)=5$ dens.=54%	6,3	0,4	91,0	4,9	0,3	93,0	6,2	0,5	91,2
games120_4 $\chi(g)=9$ dens.=9%	0	260	100	0	247	100	0	253	100
myciel7_40 $\chi(g)=8$ dens.=13%	1,8	795	99,8	0	766	100	0	745	100

Table II Efficiency of TSA with MaxTenure (tMAX=10)

Graphs	MaxTenure : 5			MaxTenure : 10			MaxTenure : 15		
	c.f.	time [s]	RR [%]	c.f.	time [s]	RR [%]	c.f.	time [s]	RR [%]
queen5.5_40 $\chi(g)=5$ dens.=54%	5,2	0,5	92,6	6,5	0,4	90,7	3,8	0,3	94,6
games120_4 $\chi(g)=9$ dens.=9%	0	252	100	0	260	100	0	250	100
myciel7_40 $\chi(g)=8$ dens.=13%	0,2	879	100	0,7	776	100	0,4	833	100

Table III Efficiency of SA algorithm with MinIteration (Tmin=0,25; Tmax=10; ControlFactor=0,9)

Graphs	MinIteration : 5			MinIteration : 10			MinIteration : 15		
	c.f.	time [s]	RR [%]	c.f.	time [s]	RR [%]	c.f.	time [s]	RR [%]
queen5.5_40 $\chi(g)=5$ dens.=54%	7,7	0,3	89,1	6,5	0,6	90,8	7,9	0,9	88,8
games120_4 $\chi(g)=9$ dens.=9%	3,7	29,9	98,6	15,9	14,2	94,0	9,3	18,3	96,5
myciel7_40 $\chi(g)=8$ dens.=13%	30	121	97,2	11k	58,8	-	31k	45,7	-

Table IV Efficiency of SA algorithm with ControlFactor (Tmin=0,25; Tmax=10; MinIteration=5)

Graphs	ControlFactor : 0,85			ControlFactor : 0,9			ControlFactor : 0,95		
	c.f.	time [s]	RR [%]	c.f.	time [s]	RR [%]	c.f.	time [s]	RR [%]
queen5.5_40 $\chi(g)=5$ dens.=54%	14	0,2	79,3	6,4	0,3	90,9	7,9	0,5	88,9
games120_4 $\chi(g)=9$ dens.=9%	104	18,3	96,1	4,1	28,7	98,5	6,4	57,3	97,6
myciel7_40 $\chi(g)=8$ dens.=13%	2k	75,2	-	13,8	119	98,8	5,4	240	99,5

Table V Efficiency of SA algorithm with Tmax (Tmin=0,25; MinIteration=5; ControlFactor=0,9)

Graphs	Tmax : 5			Tmax : 10			Tmax : 15		
	c.f.	time [s]	RR [%]	c.f.	time [s]	RR [%]	c.f.	time [s]	RR [%]
queen5.5_40 $\chi(g)=5$ dens.=54%	10	0,2	85,8	9,5	0,3	86,6	12,0	0,3	83,1
games120_4 $\chi(g)=9$ dens.=9%	4,0	24,2	98,5	5,4	29,8	98,0	6,9	30,7	97,4
myciel7_40 $\chi(g)=8$ dens.=13%	19	94,1	98,3	14,5	122	98,7	18,1	137	98,3

Similarly, as a selection criterion for a given parameter the majority of optimum solutions with respect to relative robustness RR was used.

SA/PSA algorithm has SA2/PSA2 version with automatic computation of Tmax (the initial temperature), cf. [21].

The computed parameters of TS and SA metaheuristics were used for parallel versions of both iterative methods.

IV. EXPERIMENTAL RESULTS

The application has been written in C++, while GUI in C#, accordingly. Microsoft Visual Studio 2008 v.9.0 was used. All computer experiments we performed on a machine with Intel Core 2 Duo, CPU P8400, 2,27 GHz, 4,00 GB of RAM memory.

A. Goals of optimization and algorithms

The main purpose of optimization was obtaining the best available robustness with minimum number of colors used. It is possible to:

- compute maximal RR for the given number of colors (algorithms: TS, SA, SA2)
- compute the above in parallel (algorithms: PTS, PSA2)
- find a robust coloring with minimal number of colors for the given RR (algorithms TS, SA^C, SA2^C)
- compute the above in parallel (algorithms PTS^C, PSA2^C)

The program solves RGCP problem providing value of cost function, relative robustness RR and the number of colors. There is a pool of algorithm's variants to choose from, including sequential and parallel versions.

In next subsections a number of experiments performed with the help of that program is reported.

B. TS versus SA

The first experiment was devoted to efficiency comparison of sequential versions of the two basic metaheuristics. For comparison 9 DIMACS graphs were selected the number of colors was set up to $k = \chi(G)$. The results are shown in Fig. 2. For most combinations of test graphs and the size of the set E' the TS outperforms SA in terms of relative robustness RR of the modeled system. Typically, TS was able to achieve 100% RR and never less than 95%. SA issued a bit worse results: for only three graphs maximum RR=100% was obtained. In majority of cases RR was within the range 91–99%. In a single case when SA algorithm failed to achieve a conflict-free coloring for a graph with density 46%, the value k was incremented. Basically, more dense graph are more difficult to color. SA is simpler than TS, much faster for bigger graphs and its power relies on randomization in a higher degree than TS which is more precise in searching for a good solution, checking all color combinations for all vertices in each iteration. Regardless of E' size both algorithms delivered solutions with similar values of cost function and RR. However, when E' size is bigger the number of iterations required to obtain a conflict-free coloring decreases in both methods and the speed of TS decreases. The graph density is more essential than the graph size.

C. TS^C versus SA^C and SA2^C

Three subsequent experiments were based on eight graphs instances with the percentage of E' equal 60%. The number of colors was computed that allows to achieve the given level of system relative reliability RR on the levels 70%, 85%, and 95% respectively.

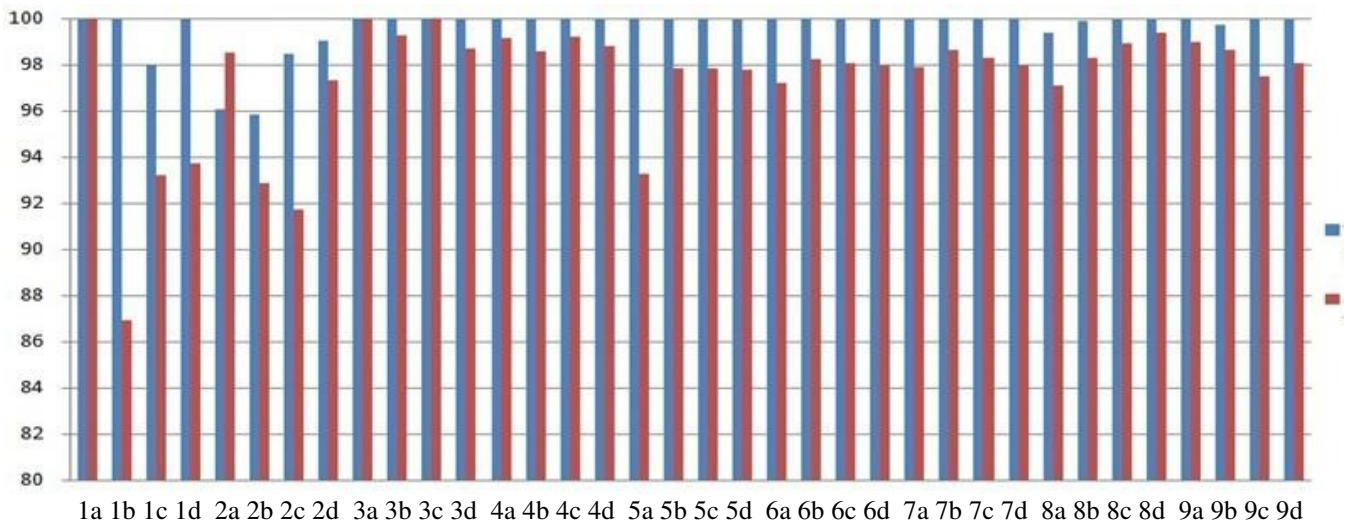


Fig. 2 Relative robustness RR [TS-blue, SA-red]. Graphs: 1-queen5.5, 2-queen6.6, 3-myciel, 4-huck, 5-david, 6-games120, 7-anna, 8-mulsol.i.4, 9-myciel7. E' : a=10%, b=20%, c=40%, d=60%. Number of colors $k = \chi(G)$.

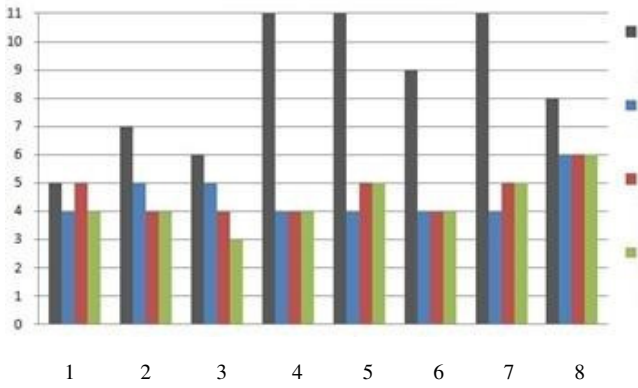


Fig. 3 Number of colors required for $RR=70\%$, $[\chi(G), TS^C, SA^C, SA2^C]$. Basic graphs: 1-queen5.5, 2-queen6.6, 3-myciel, 4-huck, 5-david, 6-games120, 7-anna, 8-myciel7; $E' = 60\%$.

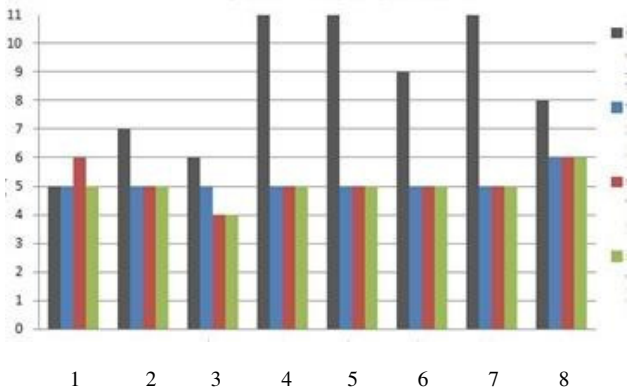


Fig. 4 Number of colors required for $RR=85\%$, $[\chi(G), TS^C, SA^C, SA2^C]$. Graphs: 1-queen5.5, 2-queen6.6, 3-myciel, 4-huck, 5-david, 6-games120, 7-anna, 8-myciel7; $E' = 60\%$.

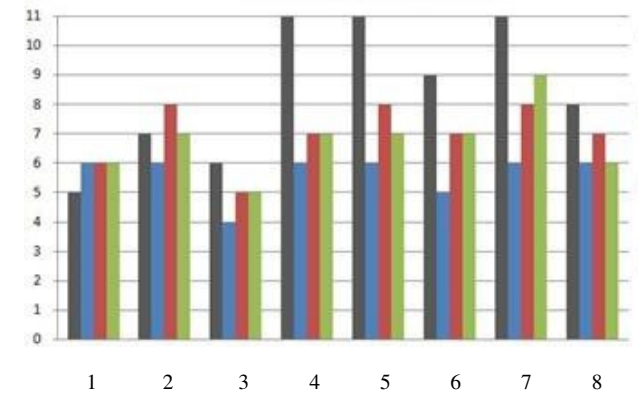


Fig. 5 Number of colors required for $RR=95\%$, $[\chi(G), TS^C, SA^C, SA2^C]$. Graphs: 1-queen5.5, 2-queen6.6, 3-myciel, 4-huck, 5-david, 6-games120, 7-anna, 8-myciel7; $E' = 60\%$.

In Fig. 3-5 the order of bars characterizing experiments for the given input graph is as follows: $\chi(G)$, TS^C , SA^C and $SA2^C$. The results depicted in Fig. 3 present the number of colors used by the corresponding methods for the set of all 8 graphs with $RR=70\%$. The average number of colors used is as follows : $TS^C=4,5$, $SA^C=4,625$ and $SA2^C=4,375$, with the average sum of $\chi(G)$ equal 8,5.

Similarly, the results depicted in Fig. 4 can be characterized in short by average number of colors used by

the corresponding methods for the set of all 8 graphs $RR=85\%$: $TS^C=5,125$, $SA^C=5,125$ and $SA2^C=5,0$ with the same sum of $\chi(G)$.

Finally, the general results depicted in Fig. 5 can be summarized by average number of colors used by the corresponding methods for the set of all 8 graphs with $RR=95\%$: $TS^C=5,625$, $SA^C=7,0$ and $SA2^C=6,75$ with respect to the sum of $\chi(G)$ as above.

D. PTS^C versus $PSA2^C$

The experiment reported in subsection C was then repeated for parallel metaheuristics PTS^C and $PSA2^C$ (with an automatic computing of initial temperature T_{max}).

Three subsequent experiments were based on eight graphs instances with the percentage of E' equal 60%. The number of colors was computed that allows to achieve the given level of system relative reliability RR on the levels 70%, 85%, and 95% respectively

Results of the research concerning minimization of colors in a conflict free robust graph coloring with fixed RR level can be summarized by the average number of colors used by the corresponding sequential and parallel methods for the set of all eight graphs from subsection C: $PTS^C=5,5$, $TS^C=5,625$, $PSA2^C=6,625$ and $SA2^C=6,75$ when the average $\chi(G)$ is 8,5. As expected , the results obtained by parallel metaheuristics are slightly improved in comparison to classical metaheuristics.

In addition total computation time of sequential and parallel versions of both metaheuristics was compared for the set of all eight graphs from subsection C. Average processing time of PTS^C is 666,8 [s] while TS^C 693,2 [s]. The average processing time of $PSA2^C$ is 674,8 [s] while $SA2^C$ 435,9 [s]. Solutions generated by PTS^C are often repeatable while $PSA2^C$ results are less stable and with similar quality as those from $SA2^C$.

V. CONCLUSIONS

In this paper new formulation of RGCP problem is given that seems to be more appropriate for designers of robust systems. Relative robustness is a versatile measure for characterization of any robust system modeled by a graph. For experimental verification two popular parallel metaheuristics TS/PTS and SA/PSA were used.

We proposed a new method of test instance generation by random modification of a given percentage E' of graph edges E . DIMACS hard-to-color graph instances were used for modification. The results confirm that the proposed approach and the used tools can be efficiently used for practical applications.

An interesting goal of the future research is to apply to RGCP – and verify experimentally – more metaheuristics like Parallel Evolutionary Algorithm (PEA), Parallel

Immune Algorithm (PIA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO) and others [10], [11]. For particular applications the robustness measures can be modified to reflect specific properties of the given system.

REFERENCES

- [1] E. Alba (Ed.), *Parallel metaheuristic - a new class of algorithms*, John Wiley & Sons, 2005. DOI: 10.1002/0471739383
- [2] C. Archetti, N. Bianchessi and A. Hertz, "A branch-and-price algorithm for the robust graph coloring problem", *Les Cahiers du Gerad*, G-2011-75, Montreal, 2011.
- [3] H. Bouziri, M. Jouini, "A tabu search approach for the sum coloring problem", *Electronic Notes in Discrete Mathematics*, Vol. 36, pp. 915-922, 2010 DOI:10.1016/j.endm.2010.05.116
- [4] R. L. Bracho, J. R. Rodriguez, F. J. Martinez : "Algorithms for Robust Graph Coloring on Paths", in *Proc. 2nd International Conference on Electrical and Electronics Engineering*, Mexico, pp. 9-12, IEEE 2005. DOI: 10.1109/ICEEE.2005.1529561
- [5] G. Chrzyszcz, *Parallel evolutionary algorithm for robust scheduling in power systems*, M.Sc. thesis, Cracow University of Technology, (in Polish) 2009.
- [6] J. Dąbrowski, "Parallelization techniques for tabu search", in *Proc. 8th Int. Conference on Applied Parallel Computing: State of the Art in Scientific Computing*. – 2007.
- [7] S. Deleplanque, J.-P. Derutin, A. Quilliot, "Anticipation in the Dial-a-Ride Problem: an introduction to the robustness", *Proc. of the 2013 Federated Conference on Computer Science and Information Systems*, FedCSIS'2013, Kraków, Poland, pp. 299-305, 2013.
- [8] A. Dey, R. Pradhan, A. Pal, T Pal, "The Fuzzy Robust Graph Coloring Problem", in S. C., Biswal B. N., Udgata S. K., Mandal J.K. (Eds.) in *Proc. of the 3rd International Conference on Frontiers of Intelligent Computing: Theory and Applications (FICTA) 2014*, Vol. 1, Advances in Intelligent Systems and Computing Proceedings, Vol. 327, pp. 805-813, Springer 2015. DOI: 10.1007/978-3-319-11933-5_91
- [9] R. Garey, D. S. Johnson, *Computers and intractability. A guide to the theory of NP-completeness*, Freeman, 1979.
- [10] M. Gendreau, J. Y. Potvin (Eds.), *Handbook of metaheuristics*, International Series in Operations Research & Management Science, Springer US, 2010. DOI: 10.1007/978-1-4419-1665-5
- [11] F. Glover, G. A. Kochenberger (Eds.), *Handbook of metaheuristics*, Kluwer 2003.
- [12] B. Gładysz, "Fuzzy robust courses scheduling problem", *Fuzzy Optimization and Decision Making*, Vol. 6, pp. 155-161, 2007. DOI: 10.1007/s10700-007-9303-0
- [13] G. Hutchinson, "Partitioning algorithms for finite sets", *Comm. ACM*, No. 6, pp. 613-614, 1963.
- [14] T. R. Jensen, B. Toft, *Graph coloring problems*, Wiley Interscience, 1995.
- [15] D. S. Johnson, M. A. Trick: Cliques, coloring and satisfiability: Second DIMACS Implementation Challenge, *DIMACS Series in Discr. Math. and Theor. Comp. Sc.* Vol. 26, 1996.
- [16] Z. Kokosiński, M. Kołodziej, K. Kwarciany, "Parallel genetic algorithm for graph coloring problem", in *Proc. of the International Conference on Computational Science, ICCS'2004*, LNCS, Vol. 3036, pp. 215-222, 2008. DOI: 10.1007/978-3-540-24685-5_27
- [17] Z. Kokosiński, "Parallel metaheuristics in graph coloring", *Bulletin of the National University "Lviv Politechnic"*, Series: Computer sciences and information technologies, No. 744 , pp. 209-214, 2012.
- [18] M. Kubale (Ed.): *Graph colorings*, American Mathematical Society, 2004. DOI: 10.1090/conm/352
- [19] A. Lim, F. Wang, "Metaheuristic for robust graph coloring problem", in *Proc. 16th IEEE Int. Conference on Tools with Artificial Intelligence*, ICTAI. – 2004.
- [20] A. Lim, F. Wang, "Robust graph coloring for uncertain supply chain management", in *Proc. 38th Annual Hawaii Int. Conf. on System Science, HICSS 2005*, p. 81b, IEEE 2005. DOI : 10.1109/HICSS.2005.526
- [21] S. Łukasik, Z. Kokosiński, G. Świętoń, "Parallel simulated annealing algorithm for graph coloring problem", in *Proceedings of the Int. Conference Parallel Processing and Applied Mathematics, PPAM'2007*, LNCS, Vol. 4967, pp. 229-238, 2008. DOI: 10.1007/978-3-540-68111-3_25
- [22] A. Pahlavani, K. Eshghi, "A hybrid algorithm of simulated annealing and tabu search for graph colouring problem", *International Journal of Operational Research*, Vol.11, No.2, pp. 136-159, 2011. DOI: 10.1504/IJOR.2011.040694
- [23] D. Ruta, "Robust Method of Sparse Feature Selection for Multi-Label Classification with Naive Bayes", *Proc. of the 2014 Federated Conference on Computer Science and Information Systems*, FedCSIS'2014, Warsaw, Poland, pp. 375-380, 2014. DOI: 10.15439/2014F502
- [24] F. Wang, Z. Xu, "Metaheuristics for robust graph coloring", *Journal of Heuristics*, Vol. 19, No.4, pp.529-548, 2013. DOI: 10.1007/s10732-011-9180-4
- [25] M. Xu, Y. Wang, and A. Wei: "Robust graph coloring based on the matrix semi-tensor product with application to examination timetabling", *Control Theory and Technology*, Vol. 12, No. 2, pp. 187-197, 2014. DOI: 10.1007/s11768-014-0153-7
- [26] J. Yáñez J., J. Ramirez, "The robust coloring problem", *European Journal of Operational Research*, Vol.148, No.3, pp. 546-558, 2003. DOI: 10.1016/S0377-2217(02)00362-4
- [27] COLOR web site. Available at : <http://mat.gsia.cmu.edu/COLOR/instances.html>.
- [28] DIMACS ftp site. Available at : <ftp://dimacs.rutgers.edu/pub/challenge/graph/benchmarks/>.