PRES LUNAM
Ecole Doctorale STIM
Sciences et Technologies de l'Information et Mathématiques

Spécialité : Informatique
Laboratoire : LERIA
Equipe : MOA

A STUDY OF MEMETIC ALGORITHM FOR THE MINIMUM SUM COLORING PROBLEM

Yan JIN
Mél : jin@info.univ-angers.fr

Abstract: Given an undirected graph G, the minimum sum coloring problem (MSCP) is to find a legal assignment of colors (represented by natural numbers) to each vertex of G such that the total sum of the colors assigned to the vertices is minimized. This paper presents a memetic algorithm for MSCP based on a tabu search procedure with two neighborhoods and a multi-parent crossover operator. Experiments on a set of 42 DIMACS and COLOR02 well-known benchmark instances show that the proposed algorithm achieves highly competitive results in comparison with five state-of-the-art algorithms. In particular, the proposed algorithm can improve the best known results for five instances.

Keywords: *Sum coloring, memetic algorithm, heuristics, combinatorial optimization.*

1 Introduction

Let $G = (V,E)$ be an undirected graph with vertex set V and edge set $E \subset V \times V$. A proper k-coloring c of G is a partition of V into k independent sets $c_1,\dots,c_k$ (color sets). The objective of the minimum sum coloring problem (MSCP) is to find a proper k-coloring c which minimizes $\sum_{i=1}^{k} i|c_i|$ where $|c_i|$ is the cardinality of the color set c_i, and the k color sets are sorted in non-ascending order of $|c_i|$, i.e., $|c_1| \geq |c_2| \geq \cdots \geq |c_k|$. This minimum, denoted as $\sum(G)$, is called the chromatic sum of G.

This paper is dedicated to the NP-hard MSCP[2], whose applications include VLSI design, scheduling, and resource allocation [4]. Considering the theoretical intractability of MSCP, a number of solution algorithms have been proposed to solve this NP-hard problem, including the EXSCOL algorithm[6], the MDS5 algorithm[1], the DBG algorithm[7], the BLS algorithm[8] and the MRLF algorithm[9]. To our knowledge, EXSCOL, BLS and MDS5 are the state-of-the-art algorithms in the literature.

2 A memetic algorithm for MSCP

2.1 Main scheme

The proposed memetic algorithm (denoted by MASC) is a hybrid algorithm combining genetic search and local search, which is summarized in Algorithm 1. MASC is composed of three main components: a dedicated crossover operator(MGPX) (Section 2.2), a local optimization procedure based on tabu search (DNTS) (Section 2.3) and a population updating mechanism(Section 2.4).

From a population of randomly generated solutions which are improved by DNTS, MASC applies at each generation the crossover operator to multi-parent to generate an offspring solution. This new solution, improved by the DNTS procedure, is inserted into the population if it is better than the worst solution of the population (discarded otherwise). The DNTS procedure is based on a token-ring application of two different neighborhoods (one-vertex-move and chain-vertices-swap) to explore the search space and a perturbation strategy to escape the local optima. The crossover operator is based on the idea of transmitting large color sets from the parents to the offspring solution.

2.2 Crossover operators

The crossover operator is an important component in a population-based algorithm[5]. It is used to generate one or more new offspring individuals to discover new promising search areas.

The MASC uses a multi-parent crossover operator, called MGPX, which generates only one offspring solution o from $p \in \{2,3,4\}$ parents. It builds the color classes of the o offspring one by one, transmitting as more vertices as possible from the parents at each step (for quality purpose). Once a parent has been selected for transmitting an

Algorithm 1 An overview of the MASC memetic algorithm for MSCP

input: A graph G
output: The minimum sum coloring c_* and $f(c_*)$ found
Population_Initialization(P, p) /* Population P has p solutions */
$f_* \leftarrow \min_{c \in P} f(c)$ /* f_* records the best objective value found so far */
for $i \leftarrow 1$ **to** *MaxGeneration* **do**
  $P' \leftarrow$ Selection(P) /* Select 2 or more parents at random for crossover */
  $o \leftarrow$ Crossover(P') /* Crossover to get an offspring solution (Sect. 2.2) */
  $o \leftarrow$ DNTS(o) /* improve o with the DNTS procedure /* Sect. 2.3) */
  if $f(o) < f_*$ **then**
    $f_* \leftarrow f(o); c_* \leftarrow o$
  end if
  Population_Updating(P, o) /* Sect. 2.4 */
end for
return f_*, c_*

entire color class to o, the parent is not considered for a few steps with purpose of varying the origins of the color classes of o. This strategy avoids transmitting always from the same parent and introduces some kind of diversity in o.

2.3 Tabu search procedure

Local optimization[3] is another important element within a memetic algorithm. In our case, its role is to improve as far as possible the quality (i.e., the sum of colors) of a given solution returned by the MGPX crossover operator.

The DNTS uses two different and complementary neighborhoods one-vertex-move N_1 and chain-vertices-swap N_2 which are applied in a token-ring way to find good local optima (intensification). More precisely, we start our search with one neighborhood and when the search ends with its best local optimum, we switch to the other neighborhood to continue the search while using the last local optimum as the starting point. When this second search terminates, we switch again to the first neighborhood and so on. For the search with each neighborhood N_i ($i = 1, 2$), DNTS stops its exploitation and and switch to the other neighborhood if the best solution is not updated for p_1 (repsectively p_2) iterations.

This neighborhood-based intensification phase stops when the best local optimum is not updated for p_3 consecutive iterations. At this point, we trigger a perturbation phase to escape from the local optimum (diversification). The DNTS algorithm stops when a maximum of p_4 iterations is met.

2.4 Population updating

The management of the population usually controls and balances two important factors in population-based heuristics: Quality and diversity.

In MASC, the coloring sum function f and the Hamming distance H are combined in a s "score" function used to decide whether a new proper k-coloring (namely the o improved offspring) replaces an individual in the population P or not: $s(c_i) = f(c_i) + e^{0.08|V|/H_{i,P}}\ \forall c_i \in P$.

Assume that o has been inserted into P, all $s(c_i)$ has been computed, and the worse configuration c_w has been identified (i.e., $s(c_w)$ is maximum), the replacement strategy obeys to the following rules:

Case 1 (c_w is not o): Remove c_w from P;

Case 2 (c_w is o): Remove the second worse individual from P with probability 0.2, and discard o otherwise.

3 Experimental results

Table 1 presents the best known results in the literature (sum^*) and the computational results of five effective algorithms and our MASC algorithm for 17 DIMACS and 25 COLOR02 instances. The best and average results of MASC are obtained by 30 independent runs with a maximum of 10 000 iterations of DNTS and 50 generations of

Table 1: Comparative results of MASC with five state-of-the-art algorithms

Instance	*sum**	EXSCOL[6]		BLS[8]		MDS5[1]	MRLF[9]	DBG[7]	MASC	
		sum	*Avg*	*sum*	*Avg*	*sum*	*sum*	*sum*	*sum*	*Avg*
myciel3	21	21	21.0	21	21.0	21	21	21	**21**	21.0
myciel4	45	45	45.0	45	45.0	45	45	45	**45**	45.0
myciel5	93	93	93.0	93	93.0	93	93	93	**93**	93.0
myciel6	189	189	189.0	189	196.6	189	189	189	**189**	189.0
myciel7	381	381	381.0	381	393.8	381	381	381	**381**	381.0
anna	276	283	283.2	276	276.0	276	277	-	**276**	276.0
david	237	237	238.1	237	237.0	237	241	-	**237**	237.0
huck	243	243	243.8	243	243.0	243	244	243	**243**	243.0
jean	217	217	217.3	217	217.0	217	217	-	**217**	217.0
queen5.5	75	75	75.0	75	75.0	75	75	-	**75**	75.0
queen6.6	138	150	150.0	138	138.0	138	138	138	**138**	138.0
queen7.7	196	196	196.0	196	196.0	196	196	-	**196**	196.0
queen8.8	291	291	291.0	291	291.0	291	303	-	**291**	291.0
games120	443	443	447.9	443	443.0	443	446	446	**443**	443.0
miles250	325	328	333.0	327	328.8	325	334	343	**325**	325.0
miles500	709	709	714.5	710	713.3	712	715	755	***705***	705.0
fpsol2.i.1	3 403	-	-	-	-	3 403	-	3 405	**3 403**	3 403.0
mug88-1	178	-	-	-	-	178	-	190	**178**	178.0
mug88-25	178	-	-	-	-	178	-	187	**178**	178.0
mug100-1	202	-	-	-	-	202	-	211	**202**	202.0
mug100-25	202	-	-	-	-	202	-	214	**202**	202.0
2-Inser_3	62	-	-	-	-	62	-	62	**62**	62.0
3-Inser_3	92	-	-	-	-	92	-	92	**92**	92.0
zeroin.i.2	1 004	-	-	-	-	1 004	-	1 013	**1 004**	1 004.0
zeroin.i.3	998	-	-	-	-	998	-	1007	**998**	998.0
DSJC125.1	326	326	326.7	326	326.9	326	352	-	**326**	326.6
DSJC125.5	1 012	1 017	1 019.7	1 012	1 012.9	1 015	1 141	-	**1 012**	1 020.0
DSJC125.9	2 503	2 512	2 512.0	2 503	2 503.0	2 511	2 653	-	**2 503**	2 508.0
DSJC250.1	973	985	985.0	973	982.5	977	1 068	-	974	990.5
DSJC250.5	3 219	3 246	3 253.9	3 219	3 248.5	3 281	3 658	-	3 230	3 253.7
DSJC250.9	8 286	8 286	8 288.8	8 290	8 316.0	8 412	8 942	-	***8 280***	8 322.7
flat300_20_0	3 150	3 150	3 150.0	-	-	-	-	-	**3 150**	3 150.0
flat300_26_0	3 966	3 966	3 966.0	-	-	-	-	-	**3 966**	3 966.0
flat300_28_0	4 282	4 282	4 286.1	-	-	-	-	-	***4 238***	4 313.4
le450_15a	2 632	2 632	2 641.9	-	-	-	-	-	2 706	2 742.6
le450_15b	2 642	2 642	2 643.4	-	-	-	-	-	2 724	2 756.2
le450_15c	3 866	3 866	3 868.9	-	-	-	-	-	***3 491***	3 491.0
le450_15d	3 921	3 921	3 928.5	-	-	-	-	-	***3 506***	3 511.8
le450_25a	3 153	3 153	3 159.4	-	-	-	-	-	3 166	3 176.8
le450_25b	3 366	3 366	3 371.9	-	-	-	-	-	**3 366**	3 375.1
le450_25c	4 515	4 515	4 525.4	-	-	-	-	-	4 700	4 773.3
le450_25d	4 544	4 544	4 550.0	-	-	-	-	-	4 722	4 805.7

the memetic algorithm (mean computation times range from less than 1 seconds to about 5 hours). From Table 1, one observes that MASC improves the best known results for five instances (miles500, DSJC250.9, flat300_28_0, le450_15c, and le450_15d) and matches the best ones for 30 instances (see non-slanted entries in bold typeface). Notice that some reference algorithms give only results for a subset of the instances, so we compare the performances between MASC and these reference algorithms one by one. We observe that except on five Leighton graphs, the results of MASC are better than or equal to those of EXSCOL for 28 instances. Compared to BLS which is the best performing algorithm for the tested instances, MASC gets results of better or equal quality for 20 instances except for two instances where MASC reaches worse results. Besides, MASC gets no worse results than the MDS5, MRLF and DBG algorithms. Hence, MASC is quite competitive on these graphs.

4 Analysis the influence of the multi-parent crossover operator

For our memetic algorithm, it is relevant to evaluate the effectiveness of its crossover operator. To verify this, we carry out experiments on the 16 selected graphs and run both MASC (using the MGPX crossover) and DNTS for 30 times (with the same parameter p_1, p_2 and p_3 settings). The DNTS starts with a single solution which is generated for MASC. DNTS stops when a maximum number of 5×10^5 iterations in order to make sure that MASC and DNTS are given the same search effort. The results are given in Table 2.

From Table 2, one notices that DNTS equals and improves respectively 5 and 3 best known results while MASC equals and improves respectively 5 and 11 best known results. Furthermore, the last column *tt* (t-test) indicates whether the observed difference between MASC and DNTS is statistically significant when a 95% confidence t-test is performed in terms of the best result obtained (Σ_*). The t-test indicates that MASC is statistically better than DNTS for 12 out of 16 cases except for the instances where DNTS can achieve the best known results (Σ). These comparative results provide clear evidences that the MGPX crossover operator plays an important role in the MASC algorithm.

Table 2: Comparative results of MASC and DNTS

Graph		MASC		DNTS		tt
Name	Σ	Σ_*	Avg.	Σ_*	Avg.	
anna	276	*276*	276.0	*276*	276.0	N
queen6.6	138	*138*	138.0	*138*	138.0	N
miles250	325	*325*	325.0	*325*	325.0	N
miles500	$\leq$ 709	**705**	705.0	**705**	705.6	Y
DSJC125.1	326	*326*	326.6	*326*	328.6	Y
DSJC125.5	1 012	**1 012**	1 020.0	1 016	1 029.8	Y
DSJC125.9	2 503	**2 503**	2 508.0	2 506	2 530.1	Y
DSJC250.1	973	**974**	990.5	981	997.7	Y
DSJC250.5	3 219	**3 230**	3 253.7	3 234	3 301.7	Y
DSJC250.9	$\leq$ 8 286	**8 280**	8 322.7	8 321	8 381.9	Y
flat300_26_0	3 966	*3 966*	3 966.0	*3 966*	3 966.0	N
flat300_28_0	$\leq$ 4 282	**4 238**	4 313.4	4 303	4 406.3	Y
le450_15c	$\leq$ 3 866	**3 491**	3 491.0	**3 491**	3 492.1	Y
le450_15d	$\leq$ 3 921	**3 506**	3 511.8	**3 506**	3 515.0	Y
le450_25c	4 515	**4 700**	4 773.3	4 749	4 803.9	Y
le450_25d	4 544	**4 722**	4 805.7	4 784	4 835.3	Y

5 Conclusion

The proposed MASC algorithm for the MSCP employs an effective DNTS procedure with one-vertex-move and chain-vertices-swap neighborhood combination and a multi-parent crossover operator to balance the intensification and diversification. The experiments on two sets of 42 benchmark instances demonstrate that MASC can improve 5 previous best solutions and frequently match the best known results in the literature for most tested cases. Furthermore, we investigated an important component of the proposed memetic algorithm. The experiment demonstrates the relevance of the multi-parent crossover operator for the overall performance of MASC.

References

[1] A. Helmar, M. Chiarandini. A local search heuristic for chromatic sum. MIC 2011: The IX Metaheuristics International Conference 2011; 161-170.

[2] E. Kubicka, A. Schwenk. An introduction to chromatic sums, in: Proceedings of the 17th ACM Annual Computer Science Conference, ACM Press, New York (NY) USA, 1989, pp. 39–45.

[3] F. Glover, M. Laguna. Tabu Search, Kluwer Academic Publishers, Dordrecht, The Nederlands, 1997.

[4] M. Malafiejski. Sum coloring of graphs, in: M. Kubale (ed.), Graph Colorings, vol. 352 of Contemporary mathematics, American Mathematical Society, New Providence (Rhode Island) USA, 2004, pp. 55–65.

[5] P. Moscato, C. Cotta. A gentle introduction to memetic algorithms, in: F. Glover, G. Kochenberger (eds.), Handbook of Metaheuristics, vol. 57 of International Series in Operations Research and Management Science, chap. 5, Kluwer Academic Publishers, Dordrecht, The Nederlands, 2003, pp. 105–144.

[6] Q. Wu, J.K. Hao. An effective heuristic algorithm for sum coloring of graphs. Computers & Operations Research 2012; 39(7):1593-1600.

[7] S.M. Douiri, S. Elbernoussi. New algorithm for the sum coloring problem. International Journal of Contemporary Mathematical Sciences 2011; 6(10): 453-463.

[8] U. Benlic, J.K. Hao. A study of breakout local search for the minimum sum coloring problem. L. T. Bui et al. (Eds.) SEAL 2012, Lecture Notes in Computer Science 7673: 128-137.

[9] Y. Li, C. Lucet, A. Moukrim, K. Sghiouer. Greedy algorithms for the minimum sum coloring problem, in: Logistique et Transports Conference, 2009.