Asia Pac. J. Math. 2021 **8**:1

EXACT COVER WITH COLOR CODES AND INTERSECTION GRAPHS

SÁNDOR SZABÓ

Department of Applied Mathematics, University of Pécs, Hungary

Corresponding author: sszabo7@hotmail.com

Received Oct. 5, 2020

Abstract. Enlarging the set of items with new type of color coded items in a generalized exact cover problem, that is, proposing the so-called exact cover problem with color codes greatly enhances the modeling power of the generalized exact cover problem. It was pointed out that the new type of exact cover problem with color codes can be reduced to back the generalized exact cover problem without color codes. The reduction is accomplished by introducing tactically chosen new secondary items. This paper will show that the number of newly introduced secondary items is related to the chromatic number of a suitably defined auxiliary graph. The arguments are constructive. Carrying out the construction of the auxiliary graph and coloring its nodes in a greedy manner gives a practical algorithm for the above reduction. Since the number of colors provided by the greedy coloring procedure is not equal to the chromatic number of the auxiliary graph the number of the introduced secondary items in the reduction exceeds the necessary optimal minimum.

2010 Mathematics Subject Classification. 05C15; 05B45.

Key words and phrases. exact cover; dancing links; chromatic number; k-clique.

1. Introduction

The exact cover (or set partitioning) problem is the following decision problem.

Problem 1.1. *Given a finite set U and a family of non-empty subsets $A_1, \ldots, A_m$ of U. Decide if there are pair-wise disjoint subsets $B_1, \ldots, B_k$ among $A_1, \ldots, A_m$ such that $U = B_1 \cup \cdots \cup B_k$.*

By the complexity theory of algorithms, Problem 1.1 is an NP-complete problem. D. E. Knuth [6] proposed an algorithm for solving Problem 1.1. The procedure is referred as

DOI: 10.28924/APJM/8-1

dancing links algorithm. D. E. Knuth went on and solved a large collection of puzzles with his dancing links algorithm. It became clear that the algorithm is capable of solving highly non-trivial size instances of the exact cover problem and so it has potential applications on various fields others than puzzles. After a slight and natural modification of the dancing links procedure can solve a more general problem the so-called generalized exact cover problem. This extension significantly enhances the modeling power of the method.

Problem 1.2. *Given two disjoint finite sets U_1, U_2 and a family of non-empty subsets $A_1, \ldots, A_m$ of $U = U_1 \cup U_2$. Decide if there are pair-wise disjoint subsets $B_1, \ldots, B_k$ among $A_1, \ldots, A_m$ such that*

$$(B_1 \cap U_1) \cup \cdots \cup (B_k \cap U_1) = U_1,$$

$$(B_1 \cap U_2) \cup \cdots \cup (B_k \cap U_2) \subseteq U_2.$$

The elements of U are referred as items and the subsets $A_1, \ldots, A_m$ are referred as options. The elements of U_1 are called primary items and the elements of U_2 are called a secondary items. In a typical situation it is assumed that each option contains at least one primary item. In other words it is assumed that

$$A_1 \cap U_1 \neq \emptyset, \ldots, A_m \cap U_1 \neq \emptyset$$

and consequently $U_1 \neq \emptyset$. The set U_2 may be empty. In this case the generalized exact cover problem reduces to the simple exact cover problem.

Motivated by applications [7] extends Problem 1.2. Namely, new type of secondary items are introduced. Each of these secondary items receives a color code from a finite list of colors. Two options containing the same secondary item with different color codes are defined to be in conflict in the sense that the two options prohibited to appear together in a qualifying exact cover. The original dancing links algorithm solves the color coded version without major restructuring. The possibility of assigning color codes to secondary items is an intuitive tool in recognizing situation where the exact cover methodology is applicable.

The solution of Exercise 99 in [7] points out that an exact cover instance with color codes can be reduced to generalized exact cover instance without color codes. The price we pay for this reduction is that the number of the secondary items may increase and the number of the items in the options may increase.

The main objective of this paper is to investigate how many new secondary items makes possible the above reduction. As it turns out the number of the necessary new items is equal to the chromatic number of a suitable constructed auxiliary graph. The construction of the auxiliary graph is computationally not overly hard. However, finding the chromatic number (as an NP-hard optimization problem) is computationally demanding. We will look for procedures to locate approximate values of the number of the secondary items. We are not on completely uncharted territory. We build on a result of P. Erdős, A. W. Goodman, and L. Pósa [2]

2. Intersection graphs

In this paper only finite simple graphs will appear. In other words, we consider graphs only with finitely many vertices and edges without double edges and loops.

A subgraph Δ is a k-clique of a finite simple graph G if two distinct vertices of Δ are always adjacent in G and Δ has k vertices.

To a family of subsets $A_1, \ldots, A_m$ of a finite set U we assign an intersection graph H. The vertices of H are the subsets $A_1, \ldots, A_m$. Two distinct nodes of H are adjacent in H whenever they are not disjoint. In notations the unordered pair $\{A_i, A_j\}$ with $i \neq j$ is an edge of H if and only if $A_i \cap A_j \neq \emptyset$.

We make an observation. Suppose that an element u of U is an element of k members of the family $A_1, \ldots, A_m$, say

$$u \in A_{\alpha(1)}, \ldots, u \in A_{\alpha(k)}.$$

Now $A_{\alpha(1)}, \ldots, A_{\alpha(k)}$ are vertices of a k-clique Δ in the intersection graph H. Each element of the set U gives rise to a clique in the intersection graph H.

The following result is due to E. Szpilrajn-Marczewski [11] and K. Čulik [1]. The proof contains a construction what we need later and this is why we include it.

Lemma 2.1. *For each finite simple graph G there is a finite set U and a family of subsets $A_1, \ldots, A_m$ of U such that their intersection graph H is isomorphic to G.*

Proof. Let $\Delta_1, \ldots, \Delta_s$ be cliques in G such that each edge of G is an edge of at least one of the cliques. We set $U = \{\Delta_1, \ldots, \Delta_s\}$ and we set

$$A_p = \{\Delta_i : \ p \text{ is a vertex of } \Delta_i\}$$

for each vertex p of G. Clearly, the sets of vertices of G and H have the same number of elements. We consider a map h from H to G defined by $h(A_p) = p$ and we consider a map g from G to H defined by $g(p) = A_p$.

It remains to prove that both h and g preserve the adjacency. Suppose that the unordered pair $\{A_p, A_q\}$ is an edge of H. This means that $A_p \cap A_q \neq \emptyset$. There is an element Δ_i in U with $\Delta_i \in (A_p \cap A_q)$. From $\Delta_i \in A_p$ it follows that $p \in \Delta_i$. From $\Delta_i \in A_q$ it follows that $q \in \Delta_i$. As Δ_i is a clique we get that the unordered pair $\{p, q\}$ is an edge of G.

Next assume that the unordered pair $\{p, q\}$ is an edge of G. There is a clique Δ_i among $\Delta_1, \ldots, \Delta_s$ such that $\{p, q\}$ is an edge of Δ_i. Now $\Delta_i \in A_p$, $\Delta_i \in A_q$ must hold. Thus $A_p \cap A_q \neq \emptyset$ and so the unordered pair $\{A_p, A_q\}$ is an edge of H.

In order to complete the proof note that the edges of G are 2-cliques and the isolated vertices are 1-cliques of G. Therefore all the isolated vertices and all the edges together can play the role of the cliques $\Delta_1, \ldots, \Delta_s$. □

Motivated by the result in Lemma 2.1 P. Erdős, A. W. Goodman, and L. Pósa [2] posed an optimization problem.

Problem 2.1. *Find a finite set U such that for each finite simple graph G with m vertices U has a family of subsets $A_1, \ldots, A_m$ whose intersection graph is isomorphic to G and $|U|$ is as small as possible.*

They proved that the minimum of $|U|$ is $\lfloor m^2/4 \rfloor$. When we reduce an exact cover problem with color codes to a generalized exact cover problem without color codes we face a similar minimization problem. But this time all information about the given graph G are available, that is, our knowledge is not restricted solely to the number of nodes of G.

3. The edge auxiliary graph

Let G be a finite simple graph. Using G we construct a new graph Γ. The set of vertices of Γ is equal to the set of edges of G. Two distinct vertices

$$w_1 = \{u_1, v_1\},\ w_2 = \{u_2, v_2\}$$

of Γ are adjacent in Γ if the set $X = \{u_1, v_1, u_2, v_2\}$ induces a clique in G. Since w_1, w_2 are distinct edges of G the set X has either 3 or 4 elements. Thus X may induce a k-clique in G with $k = 3$ or $k = 4$.

We are interested in the connection between the cliques in G and Γ.

Let Δ be a k-clique in G and let $x_1, \ldots, x_k$ be all the nodes of Δ. Let us form the unordered pairs

$$\{x_i, x_j\},\ 1 \leq i < j \leq k. \tag{3.1}$$

Lemma 3.1. *The unordered pairs (3.1) are the nodes of an s-clique Ω in Γ, where $s = k(k-1)/2$.*

Proof. Let $\{x_{i(1)}, x_{j(1)}\}$, $\{x_{i(2)}, x_{j(2)}\}$ be distinct unordered pairs from list (3.1). The set $X = \{x_{i(1)}, x_{j(1)}, x_{i(2)}, x_{j(2)}\}$ induces a clique in G and so the unordered pairs $\{x_{i(1)}, x_{j(1)}\}$, $\{x_{i(2)}, x_{j(2)}\}$ are adjacent in Γ. □

Let Ω be an s-clique in Γ and let $\{x_1, y_1\}, \ldots, \{x_s, y_s\}$ be all the nodes of Ω. Let

$$z_1, \ldots, z_r \tag{3.2}$$

be all the distinct elements among $x_1, y_1, \ldots, x_s, y_s$.

Lemma 3.2. *The elements (3.2) are the nodes of an r-clique Δ in G.*

Proof. We show that the unordered pair $\{z_i, z_j\}$ is an edge of G for each i, j, $1 \leq i < j \leq r$. As z_i is on the list (3.2) there is an $\{x_p, y_p\}$ such that either $z_i = x_p$ or $z_i = y_p$. We may assume that $z_i = x_p$ since this is only a matter of exchanging the roles of x_p and y_p. If $z_j = y_p$, then the unordered pair $\{x_p, y_p\} = \{z_i, z_j\}$ is an edge of G. For the remaining part of the proof we may assume that $z_j \neq y_p$.

As z_j is on the list (3.2) there is an $\{x_q, y_q\}$ such that either $z_j = x_q$ or $z_j = y_q$. Again we may assume that $z_j = x_q$. If $z_i = y_q$, then the unordered pair $\{x_q, y_q\} = \{z_i, z_j\}$ is an edge of G. For the remaining part of the proof we may assume that $z_i \neq y_q$. From $z_i = x_p$, $z_j = x_q$, $z_i \neq z_j$ it follows that $\{x_p, y_p\}$ and $\{x_q, y_q\}$ are distinct nodes of the clique Ω and so the set $X = \{x_p, y_p, x_q, y_q\}$ induces a clique in G. In particular, the unordered pair $\{x_p, x_q\} = \{z_i, z_j\}$ is an edge of G. □

For each finite simple graph G there is an integer s satisfying the following conditions.

(1) There are cliques $\Delta_1, \ldots, \Delta_s$ in G such that each edge of G is an edge of at least one of the cliques.

(2) There are no cliques $\Delta_1, \ldots, \Delta_{s-1}$ in G such that each edge of G is an edge of at least one of the cliques.

This well defined integer s can be called the edge covering clique number of G and can be denoted by $\rho_e(G)$. From the proof of Lemma 2.1 we know that $\rho_e(G)$ is the minimum cardinality of the set U when we want to represent G as an intersection graph of a family of subsets of U.

For each finite simple graph G there is an integer s satisfying the following conditions.

(1) There are cliques $\Delta_1, \ldots, \Delta_s$ in G such that each vertex of G is a vertex of at least one of the cliques.

TABLE 1. The adjacency matrix of the conflict graph G in Example 4.1.

	1	2	3	4	5	6	7	8	9
1	×	•	•		•		•	•	•
2	•	×	•	•	•		•		
3	•	•	×	•	•	•			
4		•	•	×	•	•			
5	•	•	•	•	×				
6			•	•		×	•		•
7	•	•				•	×	•	•
8	•						•	×	•
9	•					•	•	•	×

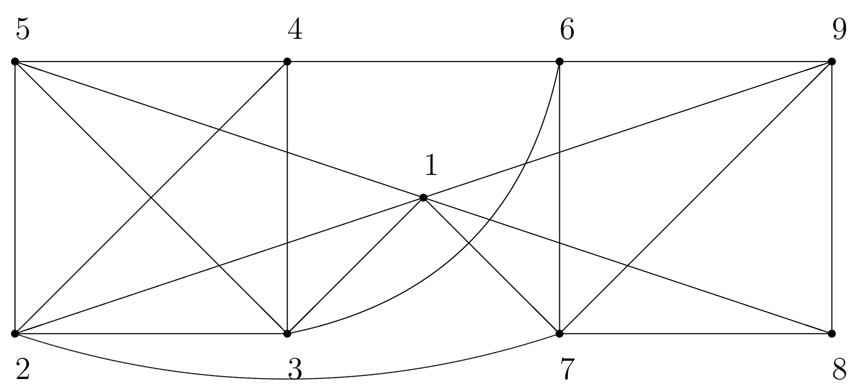

FIGURE 1. A graphical representation of the conflict graph G in Example 4.1.

(2) There are no cliques $\Delta_1, \ldots, \Delta_{s-1}$ in G such that each vertex of G is a vertex of at least one of the cliques.

This well defined integer s can be called the vertex covering clique number of G and can be denoted by $\rho_{\mathrm{v}}(G)$.

Lemma 3.3. *Using the notations introduced above the equation $\rho_{\mathrm{v}}(\Gamma) = \rho_{\mathrm{e}}(G)$ holds.*

TABLE 2. The adjacency matrix of the edge auxiliary graph Γ of G.

	1	1	1	1	1	1	2	2	2	2	3	3	3	4	4	6	6	7	7	8
	2	3	5	7	8	9	3	4	5	7	4	5	6	5	6	7	9	8	9	9
1,2	×	•	•	•			•		•	•		•								
1,3	•	×	•				•		•			•								
1,5	•	•	×				•		•			•								
1,7	•			×	•	•				•								•	•	•
1,8				•	×	•												•	•	•
1,9				•	•	×												•	•	•
2,3	•	•	•				×	•	•		•	•		•						
2,4							•	×	•		•	•		•						
2,5	•	•	•				•	•	×		•	•		•						
2,7	•			•						×										
3,4							•	•	•		×	•	•	•	•					
3,5	•	•	•				•	•	•		•	×		•						
3,6											•		×		•					
4,5							•	•	•		•	•		×						
4,6											•		•		×					
6,7																×	•		•	
6,9																•	×		•	
7,8				•	•	•												×	•	•
7,9				•	•	•										•	•	•	×	•
8,9				•	•	•												•	•	×

Proof. Set $s = \rho_{\mathrm{e}}(G)$. There are cliques $\Delta_1, \ldots, \Delta_s$ in G such that each edge of G is an edge of at least one of the cliques. By Lemma 3.1, there are cliques $\Omega_1, \ldots, \Omega_s$ in Γ such that each vertex of Γ is a vertex of at least one of the cliques. It follows that $\rho_{\mathrm{v}}(\Gamma) \le s = \rho_{\mathrm{e}}(G)$.

Set $s = \rho_{\mathrm{v}}(\Gamma)$. There are cliques $\Omega_1, \ldots, \Omega_s$ in Γ such that each vertex of Γ is a vertex of at least one of the cliques. By Lemma 3.2, there are cliques $\Delta_1, \ldots, \Delta_s$ in G such that each edge of G is an edge of at least one of the cliques. It follows that $\rho_{\mathrm{e}}(G) \le s = \rho_{\mathrm{v}}(\Gamma)$. □

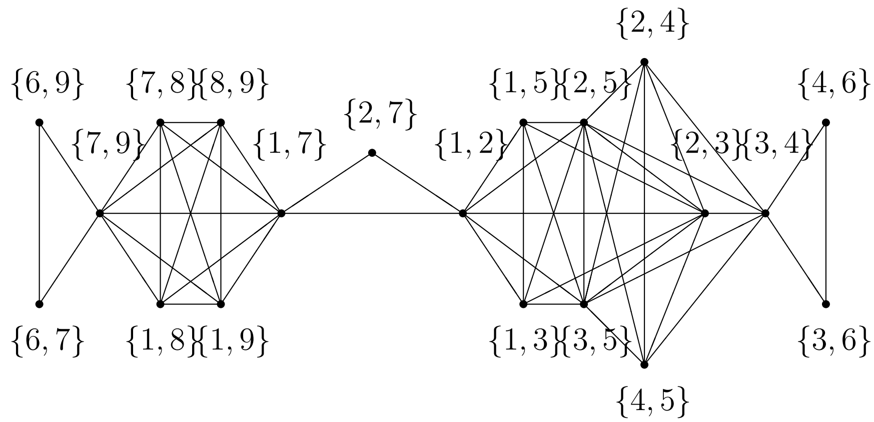

FIGURE 2. A possible geometric representation of the edge auxiliary graph Γ of the graph G in Example 4.1.

4. A TOY EXAMPLE

In this section we work out a small size toy example in details to illustrate the constructions and results we have seen so far.

Example 4.1. *Let us consider an instance of the generalized exact cover problem with colors codes. The primary items are* 1*,* 2*,* 3*,* 4*. The secondary items are* (5)*,* (6) *without color codes. The secondary items with color codes are* (7)*,* (8)*,* (9)*. The colors are* a*,* b*,* c*,* d*. The options are* $O_1, \ldots, O_9$*.*

Here is a list how the options $O_1, \ldots, O_9$ are composed from the primary items $1, 2, 3, 4$ and from the secondary items $(5), (6)$ without color codes and from the secondary items $(7), (8), (9)$ with color codes a, b, c, d.

$$
\begin{aligned}
O_1 &= \{1, (5), (9, a)\} \\
O_2 &= \{1, (8, a)\} \\
O_3 &= \{1, 2, (7, b)\} \\
O_4 &= \{2, (7, c), (8, b)\} \\
O_5 &= \{2, (8, c), (9, b)\} \\
O_6 &= \{4, (6), (7, a)\} \\
O_7 &= \{3, 4, (7, d), (8, d), (9, c)\} \\
O_8 &= \{3, (9, d)\} \\
O_9 &= \{3, (5), (6)\}
\end{aligned}
$$

Using this list of the options $O_1, \ldots, O_9$ we constructed a conflict graph G. The vertices of G are the options. Two options are connected by an edge if the options are in conflict. For example options O_1 and O_2 are adjacent in G because the primary item 1 appears in both options. Table 1 contains the adjacency matrix of the conflict graph G. In Table 1 we suppressed the letter O in the labels of rows and columns. We used only the subscripts of the options. A possible geometric representation of G can be seen in Figure 1.

From the conflict graph G we constructed the edge auxiliary graph Γ. Table 2 exhibits the adjacency matrix of Γ. Figure 2 is a geometric version of Γ. We have spotted 6 cliques $\Omega_1, \dots, \Omega_6$ that cover all nodes of Γ. We listed the nodes of these cliques below.

$$\begin{aligned}
\Omega_1 &: \{6,7\}, \{6,9\} \\
\Omega_2 &: \{1,7\}, \{1,8\}, \{1,9\}, \{7,8\}, \{7,9\}, \{8,9\} \\
\Omega_3 &: \{2,7\} \\
\Omega_4 &: \{1,2\}, \{1,3\}, \{1,5\}, \{2,3\}, \{2,5\}, \{3,5\} \\
\Omega_5 &: \{2,4\}, \{4,5\} \\
\Omega_6 &: \{3,4\}, \{3,6\}, \{4,6\}
\end{aligned}$$

To each clique Ω_i in Γ we constructed a clique Δ_i in G following the instructions of Lemma 3.2. The nodes of the cliques $\Delta_1, \dots, \Delta_6$ can be seen in the following list.

$$\begin{aligned}
\Delta_1 &: 6, 7, 9 \\
\Delta_2 &: 1, 7, 8, 9 \\
\Delta_3 &: 2, 7 \\
\Delta_4 &: 1, 2, 3, 5 \\
\Delta_5 &: 2, 4, 5 \\
\Delta_6 &: 3, 4, 6
\end{aligned}$$

As a next step we used the ideas in the proof of Lemma 2.1 to construct a family of subsets $A_1, \dots, A_9$ of the set $U = \{\Delta_1, \dots, \Delta_6\}$. The incidence matrix of the family is given in Table 3. The intersection graph H of this family is isomorphic to the conflict graph G. Adding the appropriate primary items to $A_1, \dots, A_9$ we get the new options $O'_1, \dots, O'_9$ below. (We do not intend to change the primary items in the options during the transformation.) The secondary

TABLE 3. The incidence matrix of the family $A_1, \ldots, A_9$.

	Δ_1	Δ_2	Δ_3	Δ_4	Δ_5	Δ_6
A_1		•		•		
A_2			•	•	•	
A_3				•		•
A_4					•	•
A_5				•	•	
A_6	•					•
A_7	•	•	•			
A_8		•				
A_9	•	•				

items in these new options are all without color codes.

$$\begin{aligned} O'_1 &= \{1, (2), (4)\} \\ O'_2 &= \{1, (3), (4), (5)\} \\ O'_3 &= \{1, 2, (4), (6)\} \\ O'_4 &= \{2, (5), (6)\} \\ O'_5 &= \{2, (4), (5)\} \\ O'_6 &= \{4, (1), (6)\} \\ O'_7 &= \{3, 4, (1), (2), (3)\} \\ O'_8 &= \{3, (2)\} \\ O'_9 &= \{3, (1), (2)\} \end{aligned}$$

Thus, the original generalized exact cover instance with color codes is reduced to a generalized exact cover instance without color codes.

5. NUMERICAL EXPERIMENTS

We say that the vertices of a finite simple graph are legally colored if each vertex receives exactly one color and adjacent vertices cannot have the same color. For each finite simple graph G there is an integer k such that the vertices of G can be legally colored using k colors and the vertices of G cannot be colored legally using $k-1$ colors. This number k is called the chromatic number of G and it is denoted by $\chi(G)$.

TABLE 4. Monoton matrices

n	$\|V\|$	$\|E\|$	s
3	27	189	69
4	64	1 296	233
5	125	5 500	563
6	216	17 550	1 151
7	343	46 305	2 074
8	512	106 624	3 400
9	729	221 616	5 232
10	1 000	425 250	7 736
11	1 331	765 325	10 971
12	1 728	1 306 800	15 067
13	2 197	2 135 484	20 238
14	2 744	3 362 086	26 536
15	3 375	5 126 625	34 046

TABLE 5. Deletion error correcting codes

n	$\|V\|$	$\|E\|$	s
3	8	9	9
4	16	57	29
5	32	305	75
6	64	1 473	153
7	128	6 657	294
8	256	28 801	518
9	512	121 089	844
10	1 024	499 713	1 278
11	2 048	2 037 761	1 904
12	4 096	8 247 297	2 735

TABLE 6. Johnson codes

n	$\|V\|$	$\|E\|$	s
6	15	45	15
7	35	385	87
8	70	1 855	175
9	126	6 615	241
10	210	19 425	342
11	330	49 665	464
12	495	114 345	613
13	715	242 385	794
14	1 001	480 480	1 000
15	1 365	900 900	1 226
16	1 820	1 611 610	1 476
17	2 380	2 769 130	1 757
18	3 060	4 594 590	2 069

For each finite simple graph G one can define a complement graph $\overline{G}$. The vertex sets of $\overline{G}$ and G are the same. Two vertices of $\overline{G}$ are adjacent exactly when the vertices are not adjacent in G. The set of vertices receiving the same color is called a color class of the coloring. A color class of a legal coloring cannot contain adjacent vertices. So the elements of a color class in $\overline{G}$ are vertices of a clique in G. This leads to the observation that the equation $\chi(\overline{G}) = \rho_{\mathrm{v}}(G)$ holds. The equations

$$\rho_{\mathrm{e}}(G) = \rho_{\mathrm{v}}(\Gamma) = \chi(\overline{\Gamma})$$

show that the number $\rho_{\mathrm{e}}(G)$, we are interested in, is equal to the chromatic number of $\overline{\Gamma}$. It is well-known from the complexity theory of computations that the optimization problem of computing the chromatic number belongs to the NP-hard complexity class. (See [3], [8].) So instead of computing $\chi(\overline{\Gamma})$ we try to establish upper bound for it.

From the proof of Lemma 2.1 we know that the sum of the number of edges and the number of isolated nodes of the conflict graph G is an upper bound for $\rho_{\mathrm{e}}(G)$. In the exact cover problem the isolated nodes of the associated conflict graph G can be ignored. Thus the number of edges of G, which is the number of vertices of $\overline{\Gamma}$, is an upper bound of $\rho_{\mathrm{e}}(G)$.

Sorting the nodes of $\overline{\Gamma}$ into color classes improves on this estimate even if the number of the color classes is not the smallest possible.

Exact cover problems involving thousands of items are routinely solved. Therefore the associated conflict graph G may have thousands of nodes and the edge auxiliary graph Γ may have millions of nodes. Using a simple greedy coloring procedure we legally colored the nodes of a few graphs to see whether it is practically feasible in this range.

We used three families of graphs for testing. They are related to coding theory. The graphs associated with monoton matrices are taken from [10]. The graphs connected to deletion error correcting codes are from [9]. The graphs related to Johnson codes are borrowed from [4]. In Table 4 the first column headed by n records the parameter of the graph, the second column headed by $|V|$ contains the number of vertices of the graph, the third column labeled by $|E|$ holds the number of edges. Using the graph we constructed the associated edge auxiliary graph. The greedy coloring procedure provided a legal coloring of the nodes of the edge auxiliary graph. The fourth column labeled by s records the number of colors. Tables 5 and 6 should be interpreted analogously.

The conclusion we draw from the numerical experiments is that the edge auxiliary graph can be constructed and its nodes can be legally colored in connection with such non-trivial size graphs that appear in practical exact cover applications.

Competing interests

The author(s) declare(s) that there is no conflict of interest regarding the publication of this paper.

References

[1] K. Čulik, Applications of graph theory to mathematical logic and linguistics. In: Proc. Sympos. "Theory of Graphs and its Applications", Smolenice, 1963, pp. 13–20. Publ. House Czechoslovak Acad. Sci., Prague (1964).

[2] P. Erdős, A. W. Goodman, L. Pósa, The representation of a graph by set intersections, Canadian Journal of Mathematics 18 (1966), 106–112.

[3] M. R. Garey, D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-completeness, Freeman, New York, 2003.

[4] J. Hasselberg, P. M. Pardalos, G. Vairaktarakis, Test case generators and computational results for the maximum clique problem, J. Glob. Optim. 3 (1993), 463–482.

[5] D. E. Knuth, Estimating the efficiency of back tracking programs, Math. Comp. 29 (1975), 121–131.

[6] D. E. Knuth, Dancing links, in *Millennial Perspectives in Computer Science*, J. Davies, B. Roscoe, and J. Woodcock, Eds., Palgrave Macmillan, Basingstoke, 2000, pp. 187–214.

[7] D. E. Knuth, The Art of Computer Programming, Vol. 4, Fascicle 5: Mathematical Preliminaries Redux; Introduction to Backtracking; Dancing Links 1st Ed. Addison-Wesley, 2020.

[8] C. H. Papadimitriou, Computational Complexity, Addison-Wesley Publishing Company, Inc., Reading, MA, 1994.

[9] N. J. A. Sloane, Challenge Problems: Independent Sets in Graphs, http://neilsloane.com/doc/graphs.html

[10] S. Szabó, Monoton matrices and finding cliques in a graph, Ann. Univ. Sci. Budapest., Sect. Comput. 41 (2013), 307–322.

[11] E. Szpilrajn-Marczewski, Sur deux propriétés des classes d'ensembles, Fund. Math. 33 (1945), 303–307.