Annales Univ. Sci. Budapest., Sect. Comp. 41 (2013) 307–322

MONOTONIC MATRICES AND CLIQUE SEARCH IN GRAPHS

Sándor Szabó (Pécs, Hungary)

Dedicated to Professors Zoltán Daróczy and Imre Kátai on their 75th anniversary

Communicated by Bui Minh Phong

(Received December 03, 2012; accepted April 10, 2013)

Abstract. Monotonic matrices are combinatorial objects defined in connection with certain questions in coding theory in [10]. It is an open problem to determine the maximum number of filled cells in an n by n monotonic matrix. In this note we determine this maximum value for $7 \leq n \leq 9$. The arguments are based partly on theoretical considerations and partly on computer aided exhaustive searches.

1. Introduction

Monotonic matrices are combinatorial objects. Let n be a positive integer and consider an n by n square-shaped array that consists of n rows, n columns and n^2 cells. A cell is either empty or contains an element of the set $\{1, \dots, n\}$. The fact that the cell at the intersection of the (a_1)-th row and the (a_2)-th column is filled with the entry a_3 will be recorded by the triplet (a_1, a_2, a_3).

We call a partially filled array a monotonic matrix if the next three conditions hold for any two distinct filled cells (a_1, a_2, a_3), (b_1, b_2, b_3) of the array.

Key words and phrases: Tripod, semi-cross, monotonic matrix, integer code, k-clique problem, maximum clique problem, set packing problem, clique search algorithms, practical algorithms for NP-complete problems.
2010 Mathematics Subject Classification: Primary 94B60; Secondary 90C27, 05D99.

(i) (Row condition.) If $a_2 = b_2$, then

$$a_1 \neq b_1,\ a_3 \neq b_3,$$
$$a_1 > b_1 \Longleftrightarrow a_3 > b_3.$$

(ii) (Column condition.) If $a_1 = b_1$, then

$$a_2 \neq b_2,\ a_3 \neq b_3,$$
$$a_2 > b_2 \Longleftrightarrow a_3 > b_3.$$

(iii) (Positive slope condition.) If $a_3 = b_3$, then

$$a_1 \neq b_1,\ a_2 \neq b_2,$$
$$a_1 > b_1 \Longleftrightarrow a_2 > b_2.$$

The intuitive meaning of the row condition is the following. The equation $a_2 = b_2$ says that the two cells are in the same row. They are not in the same column as $a_1 \neq b_1$. They do not hold the same number since $a_3 \neq b_3$. Finally, $a_1 > b_1 \Longleftrightarrow a_3 > b_3$ informs us that the cell on the right contains the larger number.

Similarly, the intuitive content of the column condition is the following. Since $a_1 = b_1$, the two cells are in the same column. As $a_2 \neq b_2$, the cells are not in the same row. Then $a_3 \neq b_3$ means that the cells are not filled with the same number. The equivalence $a_2 > b_2 \Longleftrightarrow a_3 > b_3$ asserts that the upper cell contains the larger number.

Finally, the intuitive meaning of the positive slope condition is the next. By $a_3 = b_3$, two cells are filled the same number. By $a_1 \neq b_1$, $a_2 \neq b_2$, the cells are not in the same row and they are not in the same column. The condition $a_1 > b_1 \Longleftrightarrow a_2 > b_2$ means that the straight line that passes through the centers of the cells has a positive slope.

8			2			4	7	8
7			1	7	8			
6	7	8						
5		2		4				6
4		1				3	6	
3	4				6			
2	2				3			5
1	1			3			5	
	1	2	3	4	5	6	7	8

Table 1. Hickerson's example

Table 1 exhibits an 8 by 8 monotonic matrix which has 23 filled-in cells. The example is due to D. R. Hickerson.

A monotonic matrix is called maximal if no more filled cells can be added to it and produce a larger monotonic matrix. A monotonic matrix is called a maximum monotonic matrix if there is no monotonic matrix with the same order containing more filled-in cells. The reader can verify that the 3 by 3 monotonic matrices given by Table 2 are maximal and maximum monotonic matrices respectively.

		3
	2	
1		

2		3
		1
1	3	

Table 2. A maximal and a maximum monotonic matrix

Let $f(n)$ be the number of the filled-in cells in an n by n maximum monotonic matrix.

The problem below is due to S. K. Stein.

Problem 1.1. *Find upper and lower bounds for $f(n)$. In particular determine $f(n)$.*

In order to find out more about the connection between coding theory and Stein's problem and for further interesting results the reader should consult [3], [5], [12], [13].

Lower bounds of $f(n)$ are presented in [13] for $n \leq 16$ and upper bounds of $f(n)$ are listed in [14] for $n \leq 10$. The exact values of $f(n)$ are known for $n \leq 9$. These values are summarized in Table 3. They were computed by K. Joy for $n \leq 5$. (See [11].) For $n = 6$ and for $7 \leq n \leq 9$ they were established by A. Tiskin [13] and the present paper respectively.

n	1	2	3	4	5	6	7	8	9
$f(n)$	1	2	5	8	11	14	19	23	28

Table 3. Values of $f(n)$ for $n \leq 9$

Both K. Joy and A. Tiskin used custom made programs. The extensive list of very good quality lower bounds compiled in [13] speaks eloquently for the strength of this approach.

Besides the efficiency there are other important factors to take into consideration like reliability and ease of setting up a computation. Determining $f(n)$ can be reduced to various standard combinatorial optimization problems. For instance it can be reduced to the maximum set packing problem or the maximum independent set or maximum clique problems. R. Pratt reformulated the maximum independent set equivalent of the problem in terms of a zero-one linear program in the case $n = 6$. Then he used the SAS/OR MILP zero-one linear program solver. (See [9].)

This approach is worth mentioning even if it just repeats an already successfully completed computation. A zero-one linear program solver can be used to solve various problems not only for finding maximum monotonic matrices. This means that the solver can be tested on a host of problems while for a custom made program the set of test problems is limited. A popular zero-one linear program solver is used by a large number of users in different computing environments. For this reason widely used programs tend to be more reliable.

There is a number of clique search algorithms (see [1]) and also there are well tested implementations (see [2], [6], [8], [7]). For a collection of benchmark problems see [4]. The author thinks that finding maximum monotonic matrices is a good benchmark for testing clique search algorithms. The description of the problem is simple and easy to communicate. Some cases are solved and some remain open. The search spaces are large enough to make the computations demanding. The results contribute to mathematical knowledge.

It is tempting to try the strengths of the clique solvers on computing $f(n)$. In this short note first we describe two combinatorial ideas that help to reduce the size of the search. The reduced problems then were processed by a clique search program. It was not the purpose of this note to make a detailed comparison of the performance of the various clique search algorithms and implementations. The main point is that the clique approach with the available clique solvers can settle Problem 1.1 for $n \leq 9$.

2. The clique reformulation

Let Γ be a finite graph without loops and double edges. A subgraph Δ of Γ is called a clique if any two distinct nodes of Δ are connected by an edge. A clique Δ in Γ is called a k-clique if it has k nodes. It is called a maximal clique if Γ does not have any nodes outside Δ that are connected with each node of Δ. In other words Δ is a maximal clique in Γ if Δ cannot be extended to a larger clique by adding a further node of Γ to it. A k-clique Δ in Γ is called a maximum clique if Γ does not have any $(k + 1)$-clique.

The problems below are known as the maximum clique problem and the k-clique problem respectively.

Problem 2.1. *Find a maximum clique in a given graph.*

Problem 2.2. *Given a positive integer k, decide if a given graph has a k-clique.*

We define a graph Γ. The nodes of Γ are the elements of the 3-dimensional Hamming space over the alphabet $\{1, \ldots, n\}$

$$H_n = \{(a_1, a_2, a_3) : \ 1 \leq a_1, a_2, a_3 \leq n\}.$$

Two distinct nodes $a, b \in H_n$ are connected by an edge in Γ if the row, column, positive slope conditions hold for a, b.

Note that if M is an n by n monotonic matrix, then the triplets corresponding to the filled cells in M form the nodes of a clique Δ in Γ. Conversely, if Δ is a clique in Γ, then the nodes of Δ form the filled cells of an n by n monotonic matrix M. Maximum monotonic matrices correspond to maximum cliques. As an illustration, Table 4 contains the adjacency matrix of Γ in the special case $n = 3$. The rows and the columns of the adjacency matrix are labeled by the elements of H_3. A bullet in the cell at the intersection of row a and column b indicates the triplets a, b are connected by an edge in Γ.

If M is a monotonic matrix, then $\varphi(M)$ denotes number of filled-in cells in M. The following lemma is folklore. For example it was used (without proof) in [13] page 3. We state it for the sake of easier reference and completeness.

Lemma 2.1. *Let M be an n by n monotonic matrix. There is an n by n monotonic matrix M' such that $\varphi(M) \leq \varphi(M')$ and the cell in M' at the position $(1, 1)$ contains 1 and the cell in M' at the position (n, n) contains n.*

Proof. Let M be an n by n monotonic matrix. The case $n = 1$ is trivial and so we may assume that $n \geq 2$. We prove the first case, the proof of the second being similar. If the $(1, 1)$ cell in M is filled with 1, then there is nothing to prove. For the remaining part of the proof we may assume that the $(1, 1)$ cell in M is not filled with 1.

Case 1. The $(1, 1)$ cell in M is empty. Let us look at the (1)-st row and the (1)-st column of M. If none of them contains a 1, then they contain elements from the set $\{2, \ldots, n\}$. We now can place a 1 in the position $(1, 1)$ to get a new monotonic matrix M'. Next suppose that the (1)-st row contains a 1. Then the (1)-st column cannot contain a 1. We can move this 1 to the position $(1, 1)$ to get a monotonic matrix M'. Finally suppose that the (1)-st column contains a 1. Now we can move this 1 to the position $(1, 1)$.

Case 2. The $(1, 1)$ cell in M is not empty. Suppose that the position $(1, 1)$ contains k, where $k \geq 2$. We simply can replace k by 1 to get a new monotonic matrix M'. ■

	111	112	113	121	122	123	131	132	133	211	212	213	221	222	223	231	232	233	311	312	313	321	322	323	331	332	333
111					•	•		•	•		•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•
112						•			•			•	•	•	•	•	•	•			•	•	•	•	•	•	•
113													•	•	•	•	•	•				•	•	•	•	•	•
121								•	•		•	•		•	•	•	•	•		•	•		•	•	•	•	•
122	•								•	•		•			•	•	•	•	•		•			•	•	•	•
123	•	•								•	•					•	•	•	•	•					•	•	•
131											•	•		•	•		•	•		•	•		•	•		•	•
132	•			•						•		•	•		•			•	•		•	•		•			•
133	•	•		•	•					•	•		•	•					•	•		•	•				
211					•	•		•	•					•	•		•	•		•	•	•	•	•	•	•	•
212	•			•		•	•		•						•			•			•	•	•	•	•	•	•
213	•	•		•	•		•	•														•	•	•	•	•	•
221	•	•	•					•	•								•	•		•	•		•	•	•	•	•
222	•	•	•	•			•		•	•								•	•		•			•	•	•	•
223	•	•	•	•	•		•	•		•	•								•	•					•	•	•
231	•	•	•	•	•	•														•	•		•	•		•	•
232	•	•	•	•	•	•	•			•			•						•		•	•		•			•
233	•	•	•	•	•	•	•	•		•	•		•	•					•	•		•	•				
311					•	•		•	•					•	•		•	•					•	•		•	•
312	•			•		•	•		•	•			•		•	•		•						•			•
313	•	•		•	•		•	•		•	•		•	•		•	•										
321	•	•	•					•	•	•	•	•					•	•								•	•
322	•	•	•	•			•		•	•	•	•	•			•		•	•								•
323	•	•	•	•	•		•	•		•	•	•	•	•		•	•		•	•							
331	•	•	•	•	•	•				•	•	•	•	•	•												
332	•	•	•	•	•	•	•			•	•	•	•	•	•	•			•			•					
333	•	•	•	•	•	•	•	•		•	•	•	•	•	•	•	•		•	•		•	•				

Table 4. The adjacency matrix of Γ in the case $n = 3$

×							
×							
×							
×							
×							
×							
×							
×	×	×	1	×	×	×	×

×							
×							
×							
×							
×							
×							
×							
k	×	×	×	×	×	×	×

Table 5. Two cases in Lemma 2.1

3. Symmetries

Let $a = (a_1, a_2, a_3)$, $b = (b_1, b_2, b_3)$ be triplets of an n by n monotonic matrix. Set

$$a' = (a'_1, a'_2, a'_3) = (a_2, a_1, a_3), \tag{3.1}$$

$$b' = (b'_1, b'_2, b'_3) = (b_2, b_1, b_3). \tag{3.2}$$

We constructed a' from a by swapping the first and the second components. Similarly, we constructed b' from b by swapping the first and the second components.

Lemma 3.1. *The row, column, positive slope conditions hold for a', b'.*

Proof. The row condition for the triplets a', b' is the following.

(i′) (Row condition.) If $a'_2 = b'_2$, then

$a'_1 \neq b'_1$, $a'_3 \neq b'_3$,
$a'_1 > b'_1 \Longleftrightarrow a'_3 > b'_3$.

Using (3.1), (3.2) we can replace a'_i, b'_i in (i′) in terms of a_i, b_i and we get the column condition (ii) for a, b. The column and positive slope conditions for a'_i, b'_i can be checked in an analogous way. ∎

Let again $a = (a_1, a_2, a_3)$, $b = (b_1, b_2, b_3)$ be triplets of an n by n monotonic matrix. Set

$$a'' = (a''_1, a''_2, a''_3) = (n + 1 - a_1, n + 1 - a_2, n + 1 - a_3), \tag{3.3}$$

$$b'' = (b''_1, b''_2, b''_3) = (n + 1 - b_1, n + 1 - b_2, n + 1 - b_3). \tag{3.4}$$

Lemma 3.2. *The row, column, positive slope conditions hold for a'', b''.*

Proof. Let us write the row condition for the triplets a'', b''. The result is the following.

(i″) (Row condition.) If $a''_2 = b''_2$, then

$a''_1 \neq b''_1$, $a''_3 \neq b''_3$,
$a''_1 > b''_1 \Longleftrightarrow a''_3 > b''_3$.

Then using (3.3), (3.4) we can express a_i'', b_i'' in (i$''$) in terms of a_i, b_i. We will end up with the row condition (i) for a, b. The column and positive slope conditions for a_i'', b_i'' can be checked in a similar way. ■

Let the map $\Phi_{i,j} : H_n \to H_n$ be defined by $\Phi_{i,j}(a) = a'$, where a' is constructed from a by swapping the (i)-th and the (j)-th components. Lemma 3.1 shows that $\Phi_{i,j}$ is an automorphism of the graph Γ.

Let the map $\Phi : H_n \to H_n$ be defined by $\Phi(a) = a''$, where

$$a'' = (n + 1 - a_1, n + 1 - a_2, n + 1 - a_3).$$

Lemma 3.2 gives that Φ is an automorphism of the graph Γ.

The automorphisms $\Phi_{i,j}$, Φ generate a subgroup G of the group of automorphisms of Γ. We define a binary relation $\sim$ on the vertex set H_n of Γ. Namely, $a \sim b$ if there is a $g \in G$ such that $b = g(a)$. The relation $\sim$ is an equivalence relation. Therefore the nodes of Γ are partitioned into equivalence classes $C_1, \ldots, C_s$. We order the equivalence classes by decreasing size. The largest equivalence class comes first and the smallest equivalence class comes last. In other words we assume that $|C_1| \geq \cdots \geq |C_s|$.

We come now to the point where the equivalence classes are utilized in the clique search. Let a be a vertex of the graph Γ. If b is vertex of Γ such that $\{a, b\}$ is an edge of Γ, then b is adjacent to a or in other words b is a neighbor of a. The set of all the neighbors of a is denoted by $N(a)$. As Γ does not contain loops, a is not an element of $N(a)$.

Let Δ be a maximum clique in Γ. Choose a vertex a of Γ. Note that either $\Delta \setminus \{a\}$ is a maximum clique of the subgraph of Γ spanned by $N(a)$ or Δ is a maximum clique of the subgraph of Γ spanned by $H_n \setminus \{a\}$. Therefore we end up with two smaller instances of the maximum clique problem. The usual way to proceed is to iterate this procedure and build a search tree to locate a maximum clique in Γ. This approach does not rely on equivalence classes.

Next suppose that the vertex set of Γ is partitioned into equivalence classes $C_1, \ldots, C_s$. Further assume that $a_1, \ldots, a_s$ is a fixed complete set of representatives of the equivalence classes such that $a_1 \in C_1, \ldots, a_s \in C_s$. Again let Δ be a maximum clique in Γ. The sets $C_1, \ldots, C_s$ form a partition of the vertex set of Γ. Consequently each node of Δ belongs to some C_i. If C_1 does not contain any nodes of Δ, then Δ is a maximum clique of the subgraph of Γ spanned by $H_n \setminus C_1$. If C_1 contains a node of Δ, then applying the maps $\Phi_{i,j}$, Φ we may assume that a_1 is a node of Δ. In this case $\Delta \setminus \{a_1\}$ is a maximum clique of the subgraph of Γ spanned by $N(a_1)$ As before we end up with two smaller instances of the maximum clique problem. However, typically $|C_1| > 1$ and so the subgraph of Γ spanned by $H_n \setminus C_1$ is typically smaller than the subgraph of Γ spanned by $H_n \setminus \{a\}$. This observation shows how the use of equivalence classes reduces the size of the search tree.

This result is crucial and so we spell it out as a lemma and offer a more formal justification. Consider the subgraphs Γ_i of Γ spanned by

$$N(a_i) \setminus (C_1 \cup \cdots \cup C_{i-1}), \qquad 1 \le i \le s.$$

Lemma 3.3. *If Γ contains a k-clique, then Γ_i contains a $(k-1)$-clique for some i, $1 \le i \le s$.*

Proof. Let Δ be a k-clique in Γ. Let $V = H_n$ be the set of vertices of Γ and let U be the set of vertices of Δ. The equivalence classes $C_1, \dots, C_s$ form a partition of V. Thus either $C_1 \cap U = \emptyset$ or $C_1 \cap U \neq \emptyset$.

If $C_1 \cap U \neq \emptyset$, then there is a $u \in C_1 \cap U$. As $a_1, u \in C_1$, there is a $g \in G$ such that $g(u) = a_1$. Let Δ' be the subgraph of Γ spanned by $g(U) = \{g(u) : u \in U\}$. Since Δ is a k-clique in Γ, Δ' is a k-clique in Γ. Let Δ_1 be the subgraph of Γ spanned by $g(U) \setminus \{a_1\}$. As Δ' is a k-clique in Γ, it follows that Δ_1 is a $(k-1)$-clique in the subgraph Γ_1 spanned by $N(a_1)$.

In this case there is a $(k-1)$-clique in Γ_1, as required. Therefore for the remaining part of the proof we may assume that $C_1 \cap U = \emptyset$. Now Δ is a k-clique in the subgraph $\Gamma^{(1)}$ spanned by $V_1 = V \setminus C_1$. The equivalence classes $C_2, \dots, C_s$ form a partition of V_1 and consequently either $C_2 \cap U = \emptyset$ or $C_2 \cap U \neq \emptyset$.

If $C_2 \cap U \neq \emptyset$, then there is a $u \in C_2 \cap U$. The earlier argument gives that there is a $(k-1)$-clique in the subgraph Γ_2 spanned by $N(a_2) \setminus C_1$, as required. For the remaining part of the proof we may assume that $C_2 \cap U = \emptyset$. Now Δ is a k-clique in the subgraph $\Gamma^{(2)}$ spanned by $V_2 = V \setminus (C_1 \cup C_2)$. Continuing in this way finally we get that Γ_i contains a $(k-1)$-clique for some i, $1 \le i \le s$. ■

We used two methods to compute the equivalence classes $C_1, \dots, C_s$ of the equivalence relation $\sim$. The first can be called the graph decomposition method. Let us construct a graph Ω. The nodes of Ω are the elements of H_n. We connect the node (a_1, a_2, a_3) with the nodes (a_2, a_1, a_3) and (a_3, a_2, a_1). Further we connect the node (a_1, a_2, a_3) with the node $(n+1-a_1, n+1-a_2, n+1-a_3)$. The graph Ω can be decomposed into connected components. The nodes of the connected components form the equivalence classes of the relation $\sim$. As an example we exhibited the adjacency matrix of Ω in the case $n = 3$ in Table 6. The graphical representation of this graph (without a loop on the node $(2, 2, 2)$) can be seen in Figure 1. The connected components can be identified readily and so one can read off the equivalence classes.

The second method of constructing the equivalence classes of the relation $\sim$ can be called the valuation method. Consider an element $a \in H_n$, $a = (a_1, a_2, a_3)$. Let us arrange the three coordinates of a in increasing order to

	111	112	113	121	122	123	131	132	133	211	212	213	221	222	223	231	232	233	311	312	313	321	322	323	331	332	333
111																											•
112										•																•	
113																			•						•		
121										•														•			
122											•		•										•				
123												•										•					
131																			•		•						
132																•				•							
133																			•		•				•		
211		•		•														•									
212					•												•										
213						•										•				•							
221					•										•												
222														•													
223													•										•				
231								•				•										•					
232											•												•				
233										•														•		•	
311			•				•		•																		
312								•				•															
313							•		•																		
321						•										•											
322					•										•		•										
323				•														•									
331			•						•																		
332		•																•									
333	•																										

Table 6. The adjacency matrix of Ω in the case $n = 3$

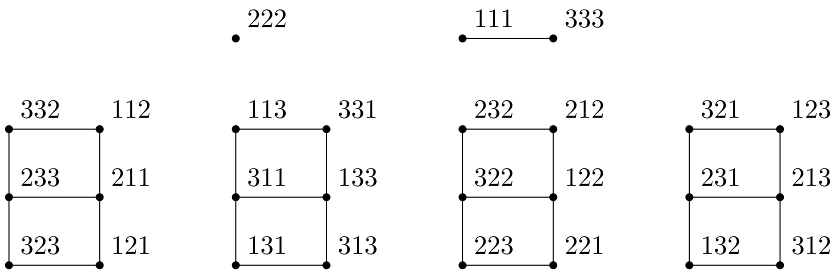

Figure 1. The graphical representation of Ω in the case $n = 3$

get the triplets

$$a' = (a'_1, a'_2, a'_3),$$
$$a'' = (a''_1, a''_2, a''_3) = (n+1-a'_3, n+1-a'_2, n+1-a'_1).$$

This means that

$$\{a_1, a_2, a_3\} = \{a'_1, a'_2, a'_3\}, \quad a'_1 \le a'_2 \le a'_3.$$

To the triplet a let us assign the minimum of the values

$$\mu(a') = (a'_1 - 1)n^2 + (a'_2 - 1)n + (a'_3 - 1),$$
$$\mu(a'') = (a''_1 - 1)n^2 + (a''_2 - 1)n + (a''_3 - 1)$$

and denote this number by $\nu(a)$. Further let a^* be either a' or a'' depending on which has the smaller ν value.

Lemma 3.4. *$a \sim b$ if and only if $\nu(a) = \nu(b)$.*

Proof. Let $a = (a_1, a_2, a_3)$, $b = (b_1, b_2, b_3)$ and suppose that $a \sim b$. Now one of

$$a' = b', \quad a' = b'', \quad a'' = b', \quad a'' = b''$$

holds and consequently $\nu(a) = \nu(b)$, as required.

Next assume that $\nu(a) = \nu(b)$. Now one of

$$\mu(a') = \mu(b'), \quad \mu(a') = \mu(b''), \quad \mu(a'') = \mu(b'), \quad \mu(a'') = \mu(b'')$$

must hold. From this it follows that one of

$$a' = b', \quad a' = b'', \quad a'' = b', \quad a'' = b''$$

holds and so $a \sim b$, as required. ■

Table 7 illustrates the equivalence class computations in the case $n = 3$. From Table 7 one can read off that the triplets

$$(1,1,2), \quad (1,2,1), \quad (2,1,1), \quad (2,3,3), \quad (3,2,3), \quad (3,3,2)$$

form an equivalence class. The number of the elements of an equivalence class can be as large as 12 and can be as small as 1. When $n = 7$ the equivalence class of the triplet $(1,2,3)$ contains each permutation of the elements 1, 2, 3. In addition the triplet $(1,2,3)$ is equivalent to $(8-1, 8-2, 8-3) = (7,6,5)$. The equivalence class of $(1,2,3)$ contains each permutation of the elements 5, 6, 7. On the other hand the equivalence class of the triplet $(4,4,4)$ contains only one element.

a	a^*	ν	a	a^*	ν	a	a^*	ν
111	111	0	211	112	1	311	113	2
112	112	1	212	122	4	312	123	5
113	113	2	213	123	5	313	113	2
121	112	1	221	122	4	321	123	5
122	122	4	222	222	13	322	122	4
123	123	5	223	122	4	323	112	1
131	113	2	231	123	5	331	113	2
132	123	5	232	122	4	332	112	1
133	113	2	233	112	1	333	111	0

Table 7. The equivalence classes in the case $n = 3$

A group of order 12, the direct product of the non-commutative group of order 6 and the group of order 2, is acting on the elements on the Hamming space H_n. The equivalence classes are the orbits of this action. Therefore the size of each equivalence class is a divisor of 12. The triplets $(1, 1, 1)$, (n, n, n) always form one complete equivalence class of the nodes H_n of the graph Γ for each n.

4. Semi-crosses

Let U be a fixed finite ground set and let $A_1, \ldots, A_r$ be fixed subsets of U. A family of subsets $B_1, \ldots, B_k \in \{A_1, \ldots, A_r\}$ is called a packing set of U if $B_i \cap B_j = \emptyset$ for each i, j, $1 \leq i, j \leq k$, $i \neq j$. A packing set $\{B_1, \ldots, B_k\}$ of U is called a k-packing set. The packing set $\{B_1, \ldots, B_k\}$ is called a maximal packing set if it is not part of any larger packing set of U. The packing set $\{B_1, \ldots, B_k\}$ is called a maximum packing set of U if U does not have any $(k + 1)$-packing set. The next problem is known as the maximum packing set problem.

Problem 4.1. *Given a finite set U and a family of subsets $A_1, \ldots, A_r$ of U. Find a maximum packing set of U.*

We will show that by introducing certain 3-dimensional star bodies, the so called semi-crosses, we can reformulate our maximum monotonic matrix search as a suitable maximum packing set problem.

Let e_1, e_2, e_3 be orthogonal unit vectors in 3-dimensional Euclidean space. The union of the 3-dimensional unit cubes parallel to the coordinate unit vectors e_1, e_2, e_3 whose centers are

$$je_i, \quad 1 \le i \le 3, \ \ 0 \le j \le n-1$$

is called the standard 3-dimensional semi-cross with arm length $n-1$. An $(n-1,3)$-semi-cross is any translate of that semi-cross by an integer vector. It consists of a corner cube and three arms composed of $n-1$ cubes.

One can consider the Hamming space H_n embedded into Z^3 and note that the Hamming sphere centered at $(1,1,1)$ with radius 1 coincides with the $(n-1,3)$-semi-cross in Z^3 centered at $(1,1,1)$. Let $a \in H_n$ and let S_a be the $(n-1,3)$-semi-cross centered at a. Set $A_a = S_a \cap H_n$.

Lemma 4.1. *The distinct triplets $a, b \in H_n$ are filled cells of an n by n monotonic matrix if and only if $A_a \cap A_b = \emptyset$.*

Proof. Let $a = (a_1, a_2, a_3)$, $b = (b_1, b_2, b_3)$. Suppose that the distinct triplets a, b are filled cells in an n by n monotonic matrix. In order to prove the claim of the lemma assume on the contrary that $A_a \cap A_b \neq \emptyset$. It follows that $S_a \cap S_b \neq \emptyset$ and so one of $a_1 = b_1$, $a_2 = b_2$, $a_3 = b_3$ must hold. If $a_1 = b_1$, then either $a_2 = b_2$ or $a_3 = b_3$. This violates column condition (ii). If $a_2 = b_2$, then either $a_1 = b_1$ or $a_3 = b_3$. Consequently the row condition (i) is violated. If $a_3 = b_3$, then either $a_1 = b_1$ or $a_2 = b_2$ and so the positive slope condition (iii) is violated.

Next suppose that $A_a \cap A_b = \emptyset$. It follows that $S_a \cap S_b = \emptyset$. Let d be hamming distance of a and b in H_n. If $d = 0$, then we get the contradiction that $a = b$. If $d = 1$, then we get the contradiction that $S_a \cap S_b \neq \emptyset$. If $d = 2$, then one of $a_1 = b_1$, $a_2 = b_2$, $a_3 = b_3$ must hold. Now conditions (ii), (i), (iii) hold respectively. If $d = 3$, then conditions (i), (ii), (iii) hold vacuously. This completes the proof. ■

Using Lemma 4.1 we can reformulate our maximum monotonic matrix problem as a maximum packing set problem. Let U be the Hamming space H_n. The given subsets A_i of U let be identified with the sets $A_a = S_a \cap H_n$ for each $a \in H_n$. The maximum packing sets of U naturally correspond to maximum n by n monotonic matrices. As an illustration the subsets $A_a = S_a \cap H_n$, $a \in H_n$ are presented in Table 8 in the special case $n = 3$.

We would like to point out that the maximum packing set problem can be reduced to a maximum clique problem. We do not claim that this reduction is necessarily advantageous. We define a graph Γ. The nodes of Γ are the elements of the set U, that is, the members of the family $\{A_1, \ldots, A_r\}$. Two distinct nodes A_i and A_j are connected by an edge if $A_i \cap A_j = \emptyset$. Cliques in Γ naturally correspond to packing sets of U.

	111	112	113	121	122	123	131	132	133	211	212	213	221	222	223	231	232	233	311	312	313	321	322	323	331	332	333
111	•	•	•	•			•			•									•								
112		•	•		•			•			•									•							
113			•			•			•			•									•						
121				•	•	•	•						•									•					
122					•	•		•						•									•				
123						•			•						•									•			
131							•	•	•							•									•		
132								•	•								•									•	
133									•									•									•
211										•	•	•	•			•			•								
212											•	•		•			•			•							
213												•			•			•			•						
221													•	•	•	•						•					
222														•	•		•						•				
223															•			•						•			
231																•	•	•							•		
232																	•	•								•	
233																		•									•
311																			•	•	•	•			•		
312																				•	•		•			•	
313																					•			•			•
321																						•	•	•	•		
322																							•	•		•	
323																								•			•
331																									•	•	•
332																										•	•
333																											•

Table 8. The family of subsets $A_a = S_a \cap H_n$ in the case $n = 3$

5. The computations

One cannot be cautious enough when carrying out a computer aided proof. First, because this type of proof is not universally greeted with enthusiasm. Second, because there are many doors open for mistakes. Being mindful of these difficulties we exercised the utmost care.

We constructed the adjacency matrix of the graph Γ based on the definition given in Section 1. These are n^3 by n^3 size matrices. For small values of n such as $n = 2, 3, 4$ we inspected the matrix. In order to double check the matrix construction by means of an independent computation we constructed the adjacency matrix of the graph Γ using the packing reformulation described in Section 4 and compared the corresponding matrices.

The equivalence classes we defined in Section 3 were computed and inspected for small values of n. Since again there were independent ways to arrive at these equivalence classes we could check the results against each other. Let $g(n)$ be the number of equivalence classes of the n by n monotonic matrices. Table 9 displays some values of $g(n)$. The most simple minded way to utilize the equivalence classes is the following.

n	3	4	5	6	7	8	9	10
$g(n)$	6	10	19	28	44	60	85	110

Table 9. Values of $g(n)$ for $3 \leq n \leq 10$.

The triplets $x = (1,1,1)$ and $y = (n,n,n)$ form one complete equivalence class C_0. Let $C_1, \ldots, C_s$ be the remaining equivalence classes such that $|C_1| \geq \cdots \geq |C_s|$. Let $a_1, \ldots, a_s$ be a complete set of representatives of the equivalence classes $C_1, \ldots, C_s$ such that $a_1 \in C_1, \ldots, a_s \in C_s$. Let the subgraphs Γ_i of Γ be spanned by

$$[N(a_i) \cap N(x) \cap N(y)] \setminus (C_0 \cup \cdots \cup C_{i-1}), \qquad 1 \leq i \leq s.$$

When we want to verify that for a given k the graph Γ contains a k-clique, by Lemmas 2.1 and 3.3, we may focus our attention to check if the graph Γ_i contains a $(k-3)$-clique for some i, $1 \leq i \leq s$.

For example when we want to check if there is a 9 by 9 monotonic matrix M with 29 filled-in entries, that is, for which $\varphi(M) = 29$ holds we check if Γ_i has a 26-clique for some i, $1 \leq i \leq 84$. This means that we carry out 84 clique searches independently of each other.

Studying Table 7 the reader will notice that if the triplets $(1,1,1)$ and $(3,3,3)$ are filled cells of a 3 by 3 monotonic matrix then neither the triplet $(1,1,2)$ nor the triplet $(1,1,3)$ can be filled cells in this monotonic matrix. This means that the whole equivalence class of $(1,1,2)$ can be removed in the course of the clique search. Similarly, the whole equivalence class of $(1,1,3)$ can be removed. So we may restrict our attention on fewer cases than Table 9 suggests at first glance.

Two different clique search algorithms starting on the same graph generally do not construct the same search tree. Therefore comparing the search trees built by different algorithms cannot be exploited to double check computations. However, we can check our computation by running the clique search algorithm on the same graph but on different machines and checking if in a given level the two search trees have the same number of nodes.

References

[1] **Bomze, I.M, M. Budinich, P.M. Pardalos and M. Pelillo,** *The Maximum Clique Problem, Handbook of Combinatorial Optimization* Vol. 4, Kluwer Academic Publisher, 1999.

[2] **Carraghan R. and P.M. Pardalos,** An exact algorithm for the maximum clique problem, *Operation Research Letters*, **9** (1990), 375–382.

[3] **Hamaker W. and S.K. Stein,** Combinatorial packing of R^3 by certain error spheres, *IEEE Trans. Inf. Theory* IT-30 (1984), 364–368.

[4] **Hasselberg, J., P.M. Pardalos and G. Vairaktarakis,** Test case generators and computational results for the maximum clique problem, *Journal of Global Optimization*, **3** (1993), 463–482. http://www.springerlink.com/content/p2m65n57u657605n

[5] **Hickerson, D.R. and S. Stein,** Abelian groups and packing by semicrosses, *Pacific. J. Math.*, **122** (1986), 95–109.

[6] **Kumlander, D.,** *Some Practical Algorithms to Solve the Maximal Clique problem*, PhD. Thesis, Tallin University of Technology, 2005.

[7] **Morgan, C.,** *A Combinatorial Search with Dancing Links*, PhD. Thesis, Univ. of Warwick, 1999–2000.

[8] **Östergård, P.R.J.,** A fast algorithm for the maximum clique problem, *Discrete Applied Mathematics* **120** (2002) 197–207.

[9] **Pratt, R.,** Personal communication, 28 06 2011.

[10] **Stein, S.K.,** Packing of R^n by certain error spheres, *IEEE Trans. Inf. Theory* IT-30 (1984), 356–363.

[11] **Stein, S.K. and S. Szabó,** *Algebra and Tiling, Homomorphisms in the Service of Geometry*, Math. Assoc. of Amer., 1994.

[12] **Stein, S.K.,** Packing tripods, *Mathematical Intelligencer* **17** (1995), 37–39.

[13] **Tiskin, A.,** Packing tripods: narrowing the density gap, *Discrete Math.* **307** (2007), 1973–1981.

[14] **Weisstein, E.W.,** *Monotonic Matrix*, MathWorld a Wolfram Web Resource, http://mathworld.wolfram.com/MonotonicMatrix.html

S. Szabó
Institute of Mathematics and Informatics
University of Pécs
Ifjúság u. 6
7624 Pécs, HUNGARY
sszabo7@hotmail.com