The Search for Consecutive Ones Submatrices: Faster and More General

Michael Dom and Rolf Niedermeier

ABSTRACT. Finding for a given 0/1-matrix a maximum-size submatrix that fulfills the Consecutive Ones Property is generally an NP-hard problem. Based on previous work, we present improved approximation and fixed-parameter algorithms for obtaining such submatrices by a minimum number of column deletions. Moreover, we show how to extend these results to the non-symmetrical case where instead of column deletions we allow row deletions.

1 Introduction

The *Consecutive Ones Property (C1P)* of 0/1-matrices plays an important role in combinatorial optimization, including application fields such as bioinformatics [8] or railway optimization [7]. A 0/1-matrix has the C1P if there is a *permutation* of its columns, that is, a finite series of column swappings, that places the 1's consecutive in every row. The C1P being a desirable property that often leads to efficient algorithms, the natural problem arises what to do if a given matrix does not have the C1P. As a consequence, there has been recently increased interest in matrix modification problems that deal with the transformation of a given 0/1-matrix into a 0/1-matrix fulfilling the C1P [2, 4, 8]. The following three optimization problems show up naturally in this context:

- Find a minimum-cardinality set of *columns* to delete such that the resulting matrix has the C1P. This problem is referred to as Min-COS-C.

- Find a minimum-cardinality set of *rows* to delete such that the resulting matrix has the C1P. This problem is referred to as Min-COS-R.

- Find a minimum-cardinality set of *1-entries* in the matrix that shall be *flipped* (that is, replaced by 0-entries) such that the resulting matrix has the C1P. This problem is referred to as Min-CO-E.

These problems are NP-hard even in case of sparse matrices [4, 8]. In this work, we further contribute to the algorithmic results for these problems, extending and improving previous work which mainly focussed on the column deletion case.

To state the results, we need the concept of (x, y)-matrices [4, 8]. A 0/1-matrix is called an *(x, y)-matrix* if it has at most x 1's in a column and at most y 1's in a row. With $x = *$ and $y = *$, respectively, we indicate that there is no upper bound on the number of 1's in the columns and rows, respectively. Previous work [8, 2] focussed on the deletion of columns: Tan and Zhang [8] provided constant-factor polynomial-time approximation algorithms in case of (2,3)-matrices and (3,2)-matrices. Very recently, this was complemented by studying the case of $(*, \Delta)$-matrices for $\Delta \geq 2$ and the parameterized complexity of Min-COS-C. Herein, a parameter d is introduced, denoting the maximum number of column deletions allowed. The following results have been shown [2]:

1. Factor-$(\Delta + 2)$-approximation algorithms and fixed-parameter algorithms with respect to parameter d for Min-COS-C where the running time is only polynomial if Δ is a constant.

2. An $O(d^2)$-size problem kernel for the parameterized version of Min-COS-C restricted to $(*, 2)$-matrices.

3. Factor-6 approximation algorithms and fixed-parameter algorithms for Min-COS-C restricted to $(2, *)$-matrices.

In this work, we present two main technical results. First, we show that for Min-COS-C we can achieve the same performance as the previous approximation and fixed-parameter algorithms *without* having the degree of the polynomial in the running time depending on Δ. In particular, this means that where so far one had polynomial-time bounds only if $\Delta = O(1)$, now we obtain polynomial-time bounds for $\Delta = O(\log N / \log \log N)$ where N denotes the size of the matrix. Second, we show how to achieve analogous results for Min-COS-R (that is, row deletion) as are known for Min-COS-C (that is, column deletion). This applies to all three points of the above enumeration. Note that this remained open in previous work [2] because of the "non-symmetrical" behavior of the C1P when trying to achieve it by either column or row deletions. We summarize our main results for Min-COS-C and Min-COS-R in Table 1. Concerning Min-CO-E on $(*, \Delta)$-matrices, we only note that for the special case of $(*, 2)$-matrices constant-factor approximation and fixed-parameter tractability results can be achieved. Although we believe that for $(*, \Delta)$-matrices with $\Delta \geq 3$ it should be possible to prove similar approximability and fixed-parameter tractability results for

Approximation algorithms

MIN-COS-C	Factor	Running time	based on
$\Delta = 2$, $\Delta \geq 5$	$\Delta + 4$	1	Thm. 4
$\Delta = 2$, $\Delta \geq 4$	$\Delta + 2$	$\Delta^{\Delta+5}$	Thm. 5
$\Delta = 3$	6	1	Thm. 5

MIN-COS-R	Factor	Running time	based on
$\Delta \geq 2$	$\Delta + 1$	$(2\Delta)^{8\Delta^2}$	Thms. 5, 14

Fixed-parameter algorithms

MIN-COS-C	Running time	based on
$\Delta = 2$, $\Delta \geq 4$	$(\Delta + 2)^d \cdot (\Delta^{\Delta+5} + (3\Delta)^{\min\{d,\Delta\}})$	Thms. 5, 12
$\Delta = 3$	6^d	Thms. 5, 12

MIN-COS-R	Running time	based on
$\Delta \geq 2$	$(\Delta + 1)^d \cdot (\Delta^{\Delta+5} + (2\Delta)^{2\min\{d,4\Delta^2\}})$	Thms. 5, 14

Table 1. Algorithms for MIN-COS-C and MIN-COS-R on $(*, \Delta)$-matrices. We only emphasize the exponential parts of the running times, that is, the shown running times have to be multiplied with polynomials with respect to the input size.

MIN-CO-E as for MIN-COS-C and MIN-COS-R, this remains an open problem for future research.

Due to the lack of space, most proofs are omitted.

2 Preliminaries and Observations

By $\mathbb{N}$ we refer to the set of positive integers. Given a graph $G = (V, E)$ and a vertex $v \in V$, the closed neighborhood of v is denoted by $N[v]$. For $V' \subseteq V$, $G[V']$ denotes the subgraph of G induced by the vertices from V'. Concerning parameterized complexity, we call a problem of size n fixed-parameter tractable with respect to the parameter d if it can be solved in $f(d) \cdot n^{O(1)}$ time where f is a computable function only depending on d.

Basic Definitions. We study $m \times n$-matrices $M = (m_{i,j})$ with entries from $\{0, 1\}$, where m denotes the number of rows and n denotes the number of columns. Interpreting the rows and columns as two vertex sets, every 0/1-matrix $M = (m_{i,j})$ can be interpreted as the adjacency matrix of a bipartite graph G_M: For every 1-entry $m_{i,j}$ in M there is an edge in G_M connecting the vertices corresponding to the i-th row and the j-th column of M. We call G_M the *representing graph* of M. Denote with *line* a row or column of M. Then, two lines ℓ, ℓ' of M are *connected in* M if there is a path in G_M connecting the vertices corresponding to ℓ and ℓ'. A submatrix M' of M is called *connected* if each pair of lines belonging to M' is connected in M'. A maximal connected submatrix of M is called a *component* of M.

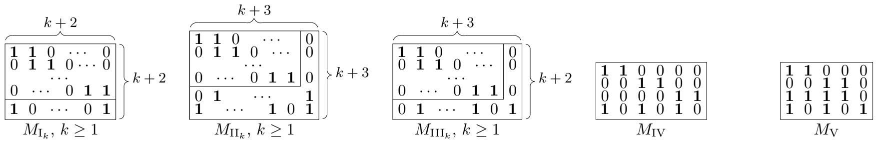

Figure 1. The set T of forbidden submatrices due to Tucker (Theorem 1).

Note that each submatrix M' of M corresponds to a vertex-induced subgraph of G_M and that each component of M corresponds to a connected component of G_M.

Fundamental Facts. When dealing with our matrix modification problems as defined in the introductory section, we can focus on the components of a given 0/1-matrix by treating them individually. The central combinatorial tool in our work as well as in previous work [2] is a characterization of matrices that have the C1P by a set of *forbidden submatrices* due to Tucker [9].

THEOREM 1 ([9, Theorem 9]). *A matrix M has the C1P iff it contains none of the matrices M_{I_k}, M_{II_k}, M_{III_k}, $k \geq 1$, M_{IV}, and M_{V} (Fig. 1) as a submatrix.*

We denote the set of submatrices given by Theorem 1 with T. Finally, an important concept we use on our way to generate C1P-matrices by a minimum number of modifications is the *Circular Ones Property (Circ1P)*. A matrix has the Circ1P if there exists a permutation of its columns such that in each row of the resulting matrix the 1's appear consecutively or the 0's appear consecutively (or both). In other words, if a matrix has the Circ1P, then there is a column permutation such that the 1's in each row appear consecutively when the matrix is wrapped around a vertical cylinder. We end with two straightforward observations. First, note that with respect to (2,2)-matrices, all problems (MIN-COS-C, MIN-COS-R, and MIN-CO-E) are polynomial-time solvable. Second, for $(*, 2)$-matrices the problems MIN-COS-R and MIN-CO-E are equivalent because deleting a row one-to-one corresponds to flipping a 1-entry since a row with only one 1-entry can be trivially omitted from further consideration.

Algorithmic Skeleton. In what follows, we briefly describe the basic algorithmic approach underlying all our algorithms and also those in previous work [2]. However, based on this algorithmic skeleton we will subsequently point out the essential new ideas needed for deriving our faster and more general algorithms. Theorem 1 can be exploited by iteratively searching and destroying in the given input matrix every submatrix that is isomorphic[1] to one of the forbidden submatrices from T: To derive a polynomial-time approximation algorithm for MIN-COS-C, all columns belonging to a forbidden submatrix are deleted, whereas to derive fixed-parameter tractability a search tree algorithm branches recursively into several subcases—deleting in each case one of the columns of the forbidden submatrix. It is important here that a $(*, \Delta)$-matrix cannot contain submatrices of types M_{II_k} and M_{III_k} for $k > \Delta - 2$ and $k > \Delta - 1$, respectively. Thus, the main difficulty is that every problem instance can contain submatrices of type M_{I_k} of unbounded size—the approximation factor or the number of cases to branch into would therefore not be bounded from above by Δ. To overcome this difficulty, the following two-phase approach has been proposed [2]:

1. Destroy only those forbidden submatrices that belong to a certain finite subset X from T, and whose sizes are upper-bounded, therefore. In the resulting matrix, every component has the Circ1P [2].

2. For each component of the resulting matrix solve the matrix modification problem individually.

THEOREM 2 ([2, Theorem 4]). *Let* $X := \{M_{\mathrm{I}_k} \mid 1 \leq k \leq \Delta - 1\} \cup \{M_{\mathrm{II}_k} \mid 1 \leq k \leq \Delta - 2\} \cup \{M_{\mathrm{III}_k} \mid 1 \leq k \leq \Delta - 1\} \cup \{M_{\mathrm{IV}}, M_{\mathrm{V}}\}$. *If a* $(*, \Delta)$*-matrix* M *contains none of the matrices in* X *as a submatrix, then each component of* M *has the Circ1P.*

Thus, if all submatrices from X are eliminated, then each component of the resulting matrix has the Circ1P. Hence, the fundamental challenge is to transform matrices with Circ1P into matrices with C1P by a minimum number of matrix modifications. So, among the main contributions of this work are faster algorithms for detecting the matrices from X *and* for finding a cost-minimal matrix transformation from Circ1P to C1P.

3 Fast Detection of Small Forbidden Submatrices

The algorithms based on the approach described in the previous section search in every step of phase 1 for a forbidden submatrix of the set T given by Tucker. For the approximation factors of the approximation algorithms as well as for the exponential factors of the running time of the fixed-parameter algorithms it is crucial that the number of columns (in the case of MIN-COS-C) or rows (in the case of MIN-COS-R) of the found submatrix is always bounded from above by a number depending on Δ.

[1] Two matrices M and M' are called *isomorphic* if M' is a permutation of the rows and columns of M.

In the previous algorithms [2] the task of finding such a submatrix was solved by an exhaustive search, resulting in a running time of $O(\Delta^2 \cdot mn^{\Delta+2})$. In this section, we will show how submatrices consisting of a small number of columns or rows can be found much faster, that is, in polynomial time with the degree of the polynomial *not* depending on Δ. Note that the known linear-time algorithms [1, 3, 5, 6] for deciding whether a given matrix has the C1P do not find a forbidden submatrix from T.

To quickly find forbidden submatrices, we use a characterization of matrices having the C1P via asteroidal triples due to Tucker [9]. Given a graph $G = (V, E)$, three vertices $u, v, w \in V$ are called an *asteroidal triple* if between any two of them there exists a path in G that does not contain a vertex from the closed neighborhood of the third vertex.

THEOREM 3 ([9, Theorem 6]). *A matrix M has the C1P iff its representing bipartite graph G_M does not contain an asteroidal triple whose three vertices correspond to columns of M.*

Using Theorem 3, a small forbidden submatrix from T in a given matrix M can be found as follows: For every vertex triple u, v, w in G_M corresponding to columns of M, determine the sum of the lengths of three shortest paths connecting u with v, u with w, and v with w, respectively, each time avoiding the closed neighborhood of the third vertex. If all three paths exist (and, hence, the sum is finite) the vertices u, v, w form an asteroidal triple in G_M. Select a triple u, v, w where the sum is minimum compared to all other triples, and return the rows and columns of M that correspond to the vertices of the three shortest paths computed for this triple. The returned submatrix must contain a submatrix from T (as given by Theorem 1), because the corresponding vertices in G_M induce a subgraph that contains an asteroidal triple, and, hence, the submatrix does not have the C1P. However, this procedure does not always return a submatrix of minimum size, because the sum of the lengths of the three paths computed for a triple u, v, w is not always the number of vertices in the union of the three paths—some vertices may be part of more than one path. More specifically, assume that the algorithm selects a triple u, v, w where the sum of the shortest paths is minimum and where the shortest paths are vertex-disjoint except for u, v, w. Then there can be another triple u', v', w' with the same sum but whose three shortest paths have several vertices in common. Selecting this triple would lead to a submatrix of smaller size. In what follows, we will analyze the size of the returned matrix and show that its number of columns and rows does never exceed the minimum number of columns and rows, respectively, of a submatrix from T plus a constant.

THEOREM 4. *Let M be a $(*, \Delta)$-matrix of size $m \times n$. If $\Delta = 3$ and the* *algorithm described above does not find a forbidden submatrix from T consisting of at most $\Delta+6$ columns (rows), or if $\Delta = 4$ and the algorithm does not find a forbidden submatrix from T consisting of at most $\Delta+5$ columns (rows), or if the algorithm does not find a forbidden submatrix from T consisting of at most $\Delta+4$ columns (rows), then M does not contain a forbidden submatrix from the set X specified in Theorem 2.*

Now we briefly turn our attention to the problem of finding the smallest forbidden submatrix with an "intelligent" exhaustive search, at the cost of an increased running time compared to the procedure presented above.

THEOREM 5. *Let M be a $(*, \Delta)$-matrix of size $m \times n$. A forbidden submatrix from T in M that has a minimum number of columns (rows) can be found in $O(\Delta^{\Delta+5} \cdot m^2n^2 + \Delta^3 \cdot m^2n^3)$ time.*

4 From Circ1P to C1P

In this section we consider the problems MIN-COS-C and MIN-COS-R restricted to input matrices that have the Circ1P, as they arise in the second phase of the algorithmic skeleton described in Sect. 2.

Circular Ones Orderings and Consecutive Ones Orderings. In what follows it is helpful to imagine the matrices as wrapped around a vertical cylinder. Hence, we will denote an order $c_1, \ldots, c_n$ of the columns of a matrix as a *circular order* if we aim to express that there is no first or last column, but that c_1 is the successor of c_n. Usually a matrix is said to have the strong C1P if—without column permutation—the ones appear consecutively in every row. Imagining a matrix as wrapped around a vertical cylinder leads to the following, "relaxed" definition of the strong C1P.

DEFINITION 6. An $m \times n$-matrix has the *strong Circ1P* if in every row the 1's appear consecutively or the 0's appear consecutively or both. A matrix $M = (m_{i,j})$ with columns $c_1, \ldots, c_n$ has the *strong C1P* if it has the strong Circ1P and, in addition, there exists a column index $j \in \{1, \ldots, n\}$ such that in every row r_i, $i \in \{1, \ldots, m\}$, that contains both 1's and 0's it holds that $m_{i,j} = 0$ or $m_{i,(j \bmod n)+1} = 0$ (or both). The column pair $(c_j, c_{(j \bmod n)+1})$ is called a *C1P cut*. A column permutation that yields the strong Circ1P (strong C1P) is called a *circular ones ordering* (*consecutive ones ordering*).

If a matrix M with columns $c_1, \ldots, c_n$ has the strong C1P and the column pair $(c_j, c_{(j \bmod n)+1})$ is a C1P cut of M, then the column permutation $c_{(j \bmod n)+1}, \ldots, c_n, c_1, \ldots, c_j$ places the 1's consecutive in every row of the resulting matrix.

A matrix that has the Circ1P can have several circular ones orderings. We will make use of the following relation between these orderings [5].

DEFINITION 7.

1. A subset C' of the columns of a matrix is called *uniform in row r* if all entries of row r in the columns of C' have the same value. Let M be a matrix and let C be the set of its columns. A *circular module* of M is a subset $C' \subseteq C$ such that in every row r the subset C' is uniform in r or $C \setminus C'$ is uniform in r.

2. Let M be a matrix and let $c_1, \ldots, c_n$ be the order of its columns. Given two column indices j_1, j_2, the operation $\mathsf{reverse}(c_{j_1}, c_{j_2})$ reverses the order of the columns $c_{j_1}, \ldots, c_{j_2}$ if $j_1 < j_2$, and it reverses the order of the columns $c_{j_1}, \ldots, c_n, c_1, \ldots, c_{j_2}$ in the circular order of M's columns if $j_1 > j_2$.

For example, applying the operation $\mathsf{reverse}(c_{n-1}, c_3)$ to a matrix with columns $c_1, \ldots, c_n$ leads to the column order $c_1, c_n, c_{n-1}, c_4, c_5, \ldots, c_{n-3}, c_{n-2}, c_3, c_2$.

THEOREM 8 ([5, Theorem 3.8]). *Let M be a matrix having the strong Circ1P. Then* every *circular ones ordering of M can be obtained by a sequence of* reverse *operations, each one applied to a circular module that is consecutive in the circular order of the columns of the current matrix.*

We are now ready to state a useful relation between the circular ones orderings and consecutive ones orderings of matrices having the C1P.

LEMMA 9. *Let M be a $(*, \Delta)$-matrix of size $m \times n$, $n \geq 2\Delta - 1$, that has the C1P. Then every circular ones ordering of M is also a consecutive ones ordering.*

Proof. Assume that M has the strong C1P. (If M does not have the strong C1P, then first permute its columns accordingly [1].) By definition, M has the strong Circ1P. We will prove that the matrix M' also has the strong C1P where M' is the matrix that results from reversing in M an arbitrary circular module that is consecutive in the circular order of M's columns. Due to Theorem 8 this suffices to prove the lemma.

Let C be the column set of M, and let $c_1, \ldots, c_n$ be its column order. Moreover, let $\tilde{C} \subseteq C$ be the circular module of M whose reversal leads to M'. Without loss of generality, assume that (c_n, c_1) is a C1P cut in M, that is, there is no row r_i in M with $m_{i,n} = 1$ and $m_{i,1} = 1$. If $\tilde{C}$ does not include c_1 and c_n, then (c_n, c_1) clearly is still a C1P cut after the column reversal. Moreover, M' has the strong Circ1P due to the definition of a circular module, which, together with the C1P cut (c_n, c_1), implies the strong C1P of M'. If $\tilde{C}$ includes both of c_1 and c_n, we can argue analogously, because then (c_1, c_n) is a C1P cut in M'.

Otherwise, assume that $\tilde{C}$ includes exactly one of c_1 and c_n, say c_1. Then $\tilde{C} = \{c_1, \ldots, c_h\}$ with $h < n$, and M' has the column ordering $c_h, \ldots, c_1, c_{h+1}, \ldots, c_n$. Assume for the sake of a contradiction that none of (c_n, c_h) and (c_1, c_{h+1}) is a C1P cut in M'. Then there must be two rows r_{i_1} and r_{i_2} such that, on the one hand, $m_{i_1,n} = 1$ and $m_{i_1,h} = 1$, and, on the other hand, $m_{i_2,1} = 1$ and $m_{i_2,h+1} = 1$. Because (c_n, c_1) is a C1P cut in M, we have $m_{i_1,1} = 0$, and $m_{i_2,n} = 0$. Therefore, $m_{i_1,j} = 1$ for every $j \in \{h+1, \ldots, n-1\}$ and $m_{i_2,j} = 1$ for every $j \in \{2, \ldots, h\}$—otherwise, there would be more than one block of 1's in the rows r_{i_1} and r_{i_2}. Because there are at most Δ 1's in each row, we have $|\{c_h, \ldots, c_n\}| \leq \Delta$ and, therefore, $h > n - \Delta \geq 2\Delta - 1 - \Delta = \Delta - 1$. For the same reason it holds that $|\{c_1, \ldots, c_{h+1}\}| \leq \Delta$ and, therefore, $h \leq \Delta - 1$, which is a contradiction to $h > \Delta - 1$. Hence, at least one of (c_n, c_h) and (c_1, c_{h+1}) must be a C1P cut in M'. ■

An Algorithm for Min-COS-C on Matrices with Circ1P. We first give an upper bound on the solution size for MIN-COS-C on matrices having the Circ1P and then characterize the structure of optimal solutions.

LEMMA 10. *Let M be a $(*, \Delta)$-matrix that has the Circ1P. Then* MIN-COS-C *on input M can be solved by deleting at most Δ columns.*

LEMMA 11. *Let M be a $(*, \Delta)$-matrix of size $m \times n$, $n \geq 3\Delta - 1$, that has the strong Circ1P, let $c_1, \ldots, c_n$ be its column order, let the set $\tilde{C}$ of columns be an optimal solution for* MIN-COS-C *on input M, and let M' be the matrix resulting from deleting[2] $\tilde{C}$ from M. Then*
1. M' has the strong C1P,
2. the columns from $\tilde{C}$ are consecutive in the circular order of M's columns, and
3. the column pair (c_α, c_β) is a C1P cut in M', where c_α and c_β are the two columns that are not contained in $\tilde{C}$ and $c_{(\alpha \bmod n)+1} \in \tilde{C}$ and $c_{\mathrm{pred}(\beta)} \in \tilde{C}$, where $\mathrm{pred}(\beta) := \beta - 1$ *for* $\beta > 1$ *and* $\mathrm{pred}(\beta) := n$ *for* $\beta = 1$.

By Lemma 11 the columns of an optimal solution $\tilde{C}$ are consecutive in *every* circular ones ordering of M. Hence, an optimal solution can easily be found:

THEOREM 12. MIN-COS-C*, restricted to $(*, \Delta)$-matrices of size $m \times n$ that have the Circ1P, can be solved in $O(\Delta \cdot mn)$ time if $n \geq 3\Delta - 1$, and in $O((3\Delta)^{\min\{d, \Delta\}} \cdot (\Delta \cdot m))$ time otherwise, where d is the number of allowed column deletions.*

[2]Note that when columns are deleted, the remaining columns retain the numbering scheme of the original matrix.

An Algorithm for Min-COS-R on Matrices with Circ1P. Compared to Min-COS-C, in the case of Min-COS-R we cannot upper-bound the size of an optimal solution in a similar way as we did in Lemma 10. However, Lemma 9 allows us to give a characterization of optimal solutions for Min-COS-R that is very similar to the one given in Lemma 11 for Min-COS-C.

LEMMA 13. *Let M be a $(*, \Delta)$-matrix of size $m \times n$, $n \geq 2\Delta - 1$, that has the strong Circ1P, let the set $\tilde{R}$ of rows be an optimal solution for* Min-COS-R *on input M, let M' be the matrix that results from deleting $\tilde{R}$ from M, and let $c_1, \ldots, c_n$ be the column order of M and M'. Then*
1. M' has the strong C1P, and
2. there is a C1P cut $(c_j, c_{(j \bmod n)+1})$ in M' such that

$$\tilde{R} = \{r_i \mid (1 \leq i \leq m) \wedge (m_{i,j} = 1) \wedge (m_{i,(j \bmod n)+1} = 1)\}.$$

THEOREM 14. Min-COS-R*, restricted to $(*, \Delta)$-matrices of size $m \times n$ that have the Circ1P, can be solved in $O(mn)$ time if $n \geq 2\Delta - 1$, and in $O((2\Delta)^{2\min\{d,4\Delta^2\}} \cdot (\Delta \cdot m))$ time otherwise, where d is the number of allowed row deletions.*

Proof. Let M be a $(*, \Delta)$-matrix of size $m \times n$ that has the Circ1P. If $n < 2\Delta - 1$, then first eliminate duplicate rows by assigning weights to the rows such that every row is weighted with the number of its occurrences. The row number of the resulting matrix is bounded from above by $(2\Delta)^2$: If one assumes that M has the strong Circ1P, then every row can be described uniquely by the index of the first and last column containing a 1 in this row. The task is now to find a row set of minimum weight whose deletion yields the C1P. An optimal solution for this problem can be found by trying all possibilities to delete at most $\min\{d, 4\Delta^2\}$ rows and checking in $O(\Delta \cdot m + n)$ time [1] whether the resulting matrix has the C1P.

If $n \geq 2\Delta - 1$, then assume that M has the strong Circ1P (a circular ones ordering of M can be found in $O(\Delta \cdot m + n)$ time [1]). Due to Lemma 13, an optimal solution can be found by counting for every column pair $(c_j, c_{(j \bmod n)+1})$ in $O(m)$ time the number of rows r_i with $m_{i,j} = 1$ and $m_{i,(j \bmod n)+1} = 1$—deleting these rows would result in a matrix with C1P cut $(c_j, c_{(j \bmod n)+1})$. ■

5 Algorithms for Min-COS-C and Min-COS-R

Based on the results from Sect. 3 and 4, in this section we obtain approximation and fixed-parameter algorithms for Min-COS-C and Min-COS-R on $(*, \Delta)$-matrices.

As sketched in the algorithmic skeleton of Section 2, our approximation algorithm for Min-COS-C (Min-COS-R) consists of two phases: First, it

	Approximation		FPT
MIN-COS-C	Fact.	Running time	Running time
$\Delta = 2$	4	$O(m^2n^3)$	$O(4^d \cdot m^2n^2)$
$\Delta = 2, 5, 6, \ldots$	$\Delta{+}4$	$O((m{+}n)^5)$	$O((\Delta{+}4)^d \cdot (3\Delta)^{\min\{d,\Delta\}} \cdot (m{+}n)^4)$
$\Delta = 3$	6	$O(m^2n^4)$	$O(6^d \cdot m^2n^3)$
$\Delta = 3, 4$	9	$O((m{+}n)^5)$	$O(9^d \cdot (m{+}n)^4)$
$\Delta \geq 4$	$\Delta{+}2$	$O(\Delta^{\Delta+5} \cdot m^2n^4)$	$O((\Delta{+}2)^d \cdot (\Delta^{\Delta+5} + (3\Delta)^{\min\{d,\Delta\}}) \cdot m^2n^3)$
MIN-COS-R	Fact.	Running time	Running time
$\Delta = 2$	3	$O(m^3n^2)$	$O(3^d \cdot m^2n^2)$
$\Delta = 2, 5, 6, \ldots$	$\Delta{+}4$	$O((2\Delta)^{8\Delta^2} \cdot (m{+}n)^5)$	$O((\Delta{+}4)^d \cdot (2\Delta)^{2\min\{d,4\Delta^2\}} \cdot (m{+}n)^4)$
$\Delta = 3, 4$	9	$O((m{+}n)^5)$	$O(9^d \cdot (m{+}n)^4)$
$\Delta \geq 3$	$\Delta{+}1$	$O((2\Delta)^{8\Delta^2} \cdot m^3n^3)$	$O((\Delta{+}1)^d \cdot (\Delta^{\Delta+5} + (2\Delta)^{2\min\{d,4\Delta^2\}}) \cdot m^2n^3)$

Table 2. Results for MIN-COS-C and MIN-COS-R on $(*, \Delta)$-matrices.

searches in every step for a matrix of the set X of forbidden submatrices given by Theorem 2 and then deletes all columns (rows) of the found submatrix. Because an optimal solution has to delete at least one column (row) of every forbidden submatrix from X, the approximation factor is bounded from above by the maximum number of columns (rows) of a submatrix found during this phase. Thereafter, due to Theorem 2, all components of the remaining matrix have the Circ1P. In case of MIN-COS-C, a solution of size at most Δ can be found for every component by permuting its columns such that a circular ones ordering is obtained and then deleting the first Δ columns—clearly, this yields a factor-Δ approximation for every component. The overall approximation factor is determined by the one achieved in the first phase of the algorithm. In case of MIN-COS-R, we do not have such a simple factor-Δ approximation for solving the problem on the components of the matrix resulting from the first phase. Hence, we use the approach of Theorem 14 for solving MIN-COS-R on every component exactly.

The fixed-parameter search tree algorithms look in every step for a forbidden submatrix of X and then branch on which column (row) belonging to the found submatrix shall be deleted. The solution for the resulting matrices without submatrices from X can be found without branching as shown in Theorem 12 (Theorem 14).

THEOREM 15. MIN-COS-C *and* MIN-COS-R*, restricted to* $(*, \Delta)$*-matrices can be approximated within the approximation factors and running times given in Table 2. Moreover,* MIN-COS-C *and* MIN-COS-R *can be solved exactly within the running times given in Table 2.*

6 Outlook

Our results mainly focus on Min-COS-C and Min-COS-R with no restriction on the number of 1's in the columns; similar studies would be desirable for the case that we have no restriction for the rows. Moreover, it should be investigated whether Δ can be eliminated from the exponents in the running times of the algorithms for Min-COS-C and Min-COS-R on $(*, \Delta)$-matrices. An important research direction is to consider the problem Min-CO-E (flipping of 1-entries). We conjecture that for $(*, \Delta)$-matrices the presented approximation and fixed-parameter tractability results should extend to Min-CO-E—however, we could not prove that.

BIBLIOGRAPHY

[1] K. S. Booth and G. S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. *Journal of Computer and System Sciences*, 13:335–379, 1976.

[2] M. Dom, J. Guo, and R. Niedermeier. Approximability and parameterized complexity of consecutive ones submatrix problems. In *Proceedings of the 4th Annual Conference on Theory and Applications of Models of Computation (TAMC 2007)*, volume 4484 of *LNCS*, pages 680–691. Springer, 2007.

[3] M. Habib, R. M. McConnell, C. Paul, and L. Viennot. Lex-BFS and partition refinement, with applications to transitive orientation, interval graph recognition and consecutive ones testing. *Theoretical Computer Science*, 234(1–2):59–84, 2000.

[4] M. T. Hajiaghayi and Y. Ganjali. A note on the consecutive ones submatrix problem. *Information Processing Letters*, 83(3):163–166, 2002.

[5] W.-L. Hsu and R. M. McConnell. PC trees and circular-ones arrangements. *Theoretical Computer Science*, 296(1):99–116, 2003.

[6] R. M. McConnell. A certifying algorithm for the consecutive-ones property. In *Proceedings of the 15th ACM-SIAM Symposium on Discrete Algorithms (SODA 2004)*, pages 768–777. SIAM, 2004.

[7] S. Mecke and D. Wagner. Solving geometric covering problems by data reduction. In *Proceedings of the 12th Annual European Symposium on Algorithms (ESA '04)*, volume 3221 of *LNCS*, pages 760–771. Springer, 2004.

[8] J. Tan and L. Zhang. The consecutive ones submatrix problem for sparse matrices. *Algorithmica*, 48(3):287–299, 2007.

[9] A. C. Tucker. A structure theorem for the consecutive 1's property. *Journal of Combinatorial Theory (B)*, 12:153–162, 1972.

Michael Dom

Institut für Informatik, Friedrich-Schiller-Universität Jena,
Ernst-Abbe-Platz 2, D-07743 Jena, Germany

dom@minet.uni-jena.de

Rolf Niedermeier

Institut für Informatik, Friedrich-Schiller-Universität Jena,
Ernst-Abbe-Platz 2, D-07743 Jena, Germany

niedermr@minet.uni-jena.de