UNIVERZA NA PRIMORSKEM
FAKULTETA ZA MATEMATIKO, NARAVOSLOVJE IN
INFORMACIJSKE TEHNOLOGIJE

Master's thesis
(Magistrsko delo)

Reconstructing perfect phylogenies via branchings in acyclic digraphs: a new lower bound and efficiently solvable cases

(Rekonstrukcija popolnih filogenij prek vejitev v acikličnih digrafih: nova spodnja meja in učinkovito rešljivi primeri)

Ime in priimek: Narmina Baghirova
Študijski program: Matematične znanosti, 2. stopnja
Mentor: prof. dr. Martin Milanič

Koper, julij 2020

Ključna dokumentacijska informacija

Ime in PRIIMEK: Narmina Baghirova

Naslov magistrskega dela: Rekonstrukcija popolnih filogenij prek vejitev v acikličnih digrafih: nova spodnja meja in učinkovito rešljivi primeri

Kraj: Koper

Leto: 2020

Število listov: 75 Število slik: 28 Število tabel: 3

Število referenc: 29

Mentor: Prof. dr. Martin Milanič

UDK: 519.17(043.2)

Ključne besede: Popolna filogenija, problem najmanjšega brezkonfliktnega razcepa vrstic, problem vejitve najmanjšega nepokritja, vejitev, acikličen digraf, particija delno urejene množice na verige, Dilworthov izrek

Math. Subj. Class. (2020): 05C90, 05C20, 06A07, 05C85, 92D10

Izvleček:
Glavno področje magistrskega dela je kombinatorična optimizacija, glavni cilj pa je podati pregled znanih in doseči nekaj novih rezultatov o dveh ekvivalentnih NP-težkih optimizacijskih problemih, problemu najmanjšega brezkonfliktnega razcepa vrstic (MCRS) in problemu vejitve najmanjšega nepokritja (MUB). V magistrskem delu širše razpravljamo o motivaciji za problem MCRS (in posledično za MUB), ki sega na področje genomike raka. Podamo formalne definicije in pregled znanih rezultatov o računski zahtevnosti in aproksimacijskih algoritmih za problema, ki so jih uvedli Hujdurović idr. (ACM Trans. Algorithms, 2018). Poleg tega predstavimo naslednje nove rezultate: uvedemo polinomsko izračunljivo spodnjo mejo za optimalno vrednost problema, identificiramo dva zadostna pogoja za polinomsko rešljivost problema in izvedemo podrobno študijo problemov MCRS in MUB na več posebnih družinah vhodnih podatkov.

Key document information

Name and SURNAME: Narmina BAGHIROVA

Title of the thesis: Reconstructing perfect phylogenies via branchings in acyclic digraphs: a new lower bound and efficiently solvable cases

Place: Koper

Year: 2020

Number of pages: 75 Number of figures: 28 Number of tables: 3

Number of references: 29

Mentor: Prof. Martin Milanič, PhD

UDC: 519.17(043.2)

Keywords: Perfect phylogeny, minimum conflict-free row split problem, minimum uncovering branching problem, branching, acyclic digraph, chain partition, Dilworth's theorem

Math. Subj. Class. (2020): 05C90, 05C20, 06A07, 05C85, 92D10

Abstract: The main area of the thesis is combinatorial optimization and the main objective is to present developments on two equivalent NP-hard optimization problems, the Minimum Conflict-free Row Split (MCRS) and the Minimum Uncovering Branching (MUB) problems. In the master thesis we broadly discuss the motivation of the MCRS (and consequently MUB) problem, which lies in the field of cancer genomics. We give formal definitions and an overview of the known computational complexity results and approximation algorithms introduced by Hujdurović et al. (ACM Trans. Algorithms, 2018). Furthermore, we obtain the following new results: we introduce a polynomially computable lower bound for the optimal value of the problem, identify two sufficient conditions for polynomial-time solvability, and perform a detailed study of the MCRS and MUB problems on several specific families of instances.

List of contents

List of Tables

List of Figures

List of Abbreviations

i.e.	that is
w.l.o.g.	Without loss of generality

Acknowledgement

I would like to thank a lot my mentor Prof. Martin Milanič for his time, patience, help, advices and comments regarding my work. I would also like to thank committee members, more explicitly Assist. Prof. Nino Bašić and Assoc. Prof. Ademir Hujdurović for carefully reading my thesis and proposing suggestions. I would like to thank UP FAMNIT for helping its students, giving us opportunities and caring about us. And finally, I would also like to express gratitude to my boyfriend, mom and friends for supporting me.

1 Introduction

1.1 Motivation

The combinatorial optimization problems investigated in this master thesis are motivated by cancer genomics, a rather new research field that takes advantage of progress of modern technology to study cancer, which is one of the leading causes of human deaths all over the world (see [6]). The clonal theory of cancer states that all cells in a tumor are developed from a single initial cell and cancer arises after a sufficient number of mutations in a tumor, which is a result of a complex evolutionary process. During the tumor evolution, each cell passes its mutations to its descendants as it divides, while the daughter cells accumulate new mutations over time. It is crucial to understand what mutations lead to an uncontrollable growth of population of abnormal cells (see, e.g., [26]) for developments in better diagnosis and more targeted therapies (see [25]).

One could examine tumor samples in the most accurate way by sampling and sequencing every single cell contained in the sample, which is impossible with current biotechnological methods. Today, single-cell analysis has made remarkable progress with possibility to output high-resolution interpretations from individual cells within the population (see, e.g., [4]), however, the population of cells may contain various cancer cells, which makes understanding of the history of tumor evolution a challenging process.

A possible solution for overcoming the challenge of reconstructing the evolutionary history of mutations is to make use of a computational approach. One of the possibilities is the following. There exists a widespread belief that all organisms are derived from a common ancestor and that new species arise by splitting one species into two, rather that mixing two species into one. Therefore a history of evolution is ideally displayed by a rooted (directed) tree (see [15]). The view of evolutionary history as a tree must be modified in each case and there are various types of evolutionary trees representing different types of phenomena. An important approach in the field of phylogenetics is given by the so-called character-based models, and one of the most basic character-based model is the so-called *perfect phylogeny* (which we formally define in Section 3). Some of its generalizations have been recently applied in

some important areas of bioinformatics, such as analysis of data from protein domains, protein networks, genetic markets, etc. (see [5]).

Going back to our discussion regarding the reconstruction of the evolutionary history of a tumor, we may make use of the perfect phylogeny evolutionary model, since the tumor progression is assumed to satisfy the following two common properties of a phylogenetic evolution:

1. All mutations in the parent cells are passed to the descendants.

2. A mutation does not occur twice at the same particular site, which is a so-called 'infinite sites assumption'.

The problems studied in this master thesis have their origins in the work of Hajirasouliha and Raphael [17], who proposed the following computational approach for reconstructing the history of somatic mutations. We are given a collection of samples of the corresponding tumor and the task is to reconstruct the history of the evolutionary process of the tumor so that the resulting model corresponds to a perfect phylogeny. We assume that each cell is in one of the following states: 0 = normal; 1 = mutated. Further, given those samples and information regarding the occurrence of the mutations, we construct a binary matrix M, where rows and columns correspond to the samples and mutations, respectively. Not surprisingly, the (i, j)-th entry of matrix M equals 1 if the mutation j occurs in sample i, otherwise the entry equals 0. Given a binary matrix M, the *perfect phylogeny problem* asks whether the matrix corresponds to a perfect phylogeny evolutionary model. The answer to the question is a known result (see [10, 15]), stating that the rows of a binary matrix M corresponds to a perfect phylogeny if and only if the matrix is conflict-free. For more details see Section 3.

1.2 Related work

Under ideal conditions, each mutation would be identified without errors, and the samples would not contain reads[1] from several leaves of the perfect phylogeny. In practice, however, each tumor sample is a mixture of reads from several tumor types, and thus the corresponding binary matrix M does typically not correspond to a perfect phylogeny, that is, it is not conflict-free. To tackle the problem, Hajirasouliha and Raphael proposed in [17] the so-called *Minimum Split Row* optimization problem, aimed at explaining each of the rows of the binary conflict matrix M with a set of rows,

[1]In the field of bioinformatics, in DNA sequencing, the term *read* refers to a sequence obtained at the end of the sequencing process and represents a part of the sequence corresponding to the entire genome.

in an overall simplest possible way, so that the resulting binary matrix M' is conflict-free and thus corresponds to a perfect phylogeny. In this master thesis we investigate a variant of the problem, since the formulation has evolved during subsequent works by Hujdurović et al. first in [19] and then in [18]. We investigate the so-called Minimum Conflict-free Row Split problem introduced in [19], which informally says the following: split each row of a given binary matrix M into bitwise OR of a set of rows so that the resulting matrix corresponds to a perfect phylogeny and has the minimum possible number of rows among all matrices with this property.

In [17], Hajirasouliha and Raphael showed that the problem is NP-complete. In addition, they introduced several concepts used in future research papers in this area, including a polynomially computable lower bound expressed in terms of chromatic numbers of certain derived graphs, called *conflict graphs*, corresponding to the binary matrix M and a row r of M (see Section 7 for the definition). Following the work of Hajirasouliha and Raphael, Hujdurović et al. introduced in [19] the Minimum Conflict-free Row Split (MCRS) problem, which is equivalent to the Minimum Split Row problem. Furthermore, Hujdurović et al. showed in [19] that some results and proofs proposed in [17] are incorrect. For instance, despite the fact that the NP-completeness claim turns out to be correct, the NP-completeness proof introduced in [17] is incorrect, due to the wrong assumption stating that every graph is a row-conflict graph (for more details see [17] and [19]). In the same paper, Hujdurović et al. gave a different NP-completeness proof by using a reduction from 3-edge-colorability of cubic graphs, which is known to be NP-complete (see [22]). Moreover, Hujdurović et al. in [19] introduced the following two results: a polynomial-time algorithm for a particular subset of instances and an efficient (not necessarily optimal) heuristic algorithm for the Minimum Conflict-free Row Split problem.

Subsequently, Hujdurović et al. in [18], firstly, showed that the MCRS problem may be equivalently formulated in terms of branchings in the so-called *containment digraph* D_M (for the precise definitions, see Section 5), which is a directed acyclic graph (DAG) derived from the given binary matrix M. Secondly, the NP-completeness result is strengthened to an APX-hardness result, proved by an L-reduction from the vertex cover problem in cubic graphs, which is known to be APX-hard (see [3]). Two approximation algorithms for the MCRS problem are introduced, with approximation ratios described in terms of the height and the width of the binary matrix M – where these quantities are defined in terms of parameters of the corresponding containment digraph D_M (for more details and formal definitions see Section 6). Moreover, the proof of a new min-max result is presented, which improves the heuristic algorithm for the Minimum Conflict-free Row Split problem introduced in [19] and is used to introduce a polynomial-time algorithm to solve the so-called Minimum Uncovering Linear

Branching (MULB) problem (see Section 6 for the definition of a linear branching and Section 9 for the definition of the problem). The min-max result is a strengthening of Dilworth's Theorem, a classical result in the theory of partially ordered sets, which states that in any finite partially ordered set, the minimum number of chains in a partition of the set equals the maximum size of an antichain [7].

In [20], Husić et al. introduced the Minimum Perfect Unmixed Phylogenies (MIPUP) method for finding the tumor evolution. This method relies on a relation between perfect phylogenies and branchings in directed acyclic graphs established in [18]. The MIPUP method is based on an Integer Linear Programming (ILP) formulation of the problem. The method was tested against four well known tools for discovering history of tumor evolution, more specifically, against CITUP (see [23]), LICHeE (see [27]), AncesTree (see [9]) and Treeomic (see [29]). The MIPUP method was shown to be the most accurate, where the accuracy was measured by the number of the original ancestor-descendant relations from original tree that were kept also in the reconstructed tree, as done also in [27] and [9]. In some cases (see [20] for more details) MIPUP reconstructs more than 92% of all relations. The method is implemented in Java and uses the CPLEX ILP solver. Apart from the optimal number of rows in a conflict-free row split M' of a binary matrix M, MIPUP also outputs the perfect phylogeny corersponding to it. The Java implementation, which is freely available at `https://github.com/zhero9/MIPUP`, was used in this master thesis for finding an optimal solution for various examples to get a better understanding of the new results introduced in the thesis. We would also like to remark that a phylogenomic approach based on phylogenetic trees returned by MIPUP is being used to study the evolution of the SARS-CoV-2 virus (see [28]), which started in 2019 in China (see [11]) and according to World Health Organization (WHO) is a part of worldwide pandemic.

1.3 Results and structure of the thesis

In the master thesis we provide the preliminary theory, consisting of the definitions necessary for understanding all the notions mentioned in this work and specify the assumptions made throughout the thesis. We formally define all the concepts mentioned in the motivation and give an overview of related results. Moreover, we summarize known results using a unified notation and give detailed proofs of particular results. We also provide some new results about the MCRS and MUB problems.

More specifically, we first give some preliminary definitions in Section 2. In Section 3 we define the concept of a perfect phylogeny and give an overview of the Minimum Conflict-free Row Split (MCRS) problem along with some concrete examples. In Section 4 we give an overview of linear-time algorithms introduced by Dan Gusfield in [16]

for testing whether a binary matrix M corresponds to a perfect phylogeny evolutionary model and if this is the case, constructs one. Then we move on to Section 5, where we present the MUB problem, a problem equivalent to the MCRS problem and defined in terms of branchings in the so-called containment digraph D_M corresponding to the input binary matrix M of the MCRS problem. In Section 6 we recall known computational complexity results and review known approximation algorithms introduced in [18]. We discuss the proof ideas and outputs of the algorithms. In Section 7, we give an overview of known polynomially computable lower bounds and justify the polynomial time complexity. Further, in Section 8, we present a min-max result from [18], which is a generalization of Dilworth's theorem. The result is essential for the proof of polynomial time complexity of an upper bound introduced in Section 9.

Finally, in Section 10 we bring to light the following results. An important result introduced in the thesis is a theorem stating that the MCRS problem is polynomial-time solvable for instances of width 2. Another result shown in this thesis is that the MCRS problem is polynomially solvable for instances such that the width of the binary matrix M equals the number of maximal elements in the corresponding poset P_M (for details see Section 10). Then, we introduce a new polynomially computable lower bound in terms of the maximum weight of an antichain and analyze the quality of this bound on two specific families of instances generalizing two families of instances introduced by Hujdurović et al. in [18]. Finally, we define a new family of instances for analyzing an open problem introduced in Section 7. A positive answer to the open problem would imply the existence of a constant factor approximation algorithm for the MCRS problem (formally defined in Section 3). We conclude the thesis by stating few open questions in this area, which could be interesting for future research.

2 Preliminaries

In this section we specify the assumptions made throughout this thesis and give an overview of the definitions crucial for understanding known as well as new results discussed in the thesis.
From now on, when we talk about a binary matrix M, we assume that there are no duplicated columns and there are no rows having all zero entries.

2.1 Graph related definitions

A *graph* G is an ordered pair (V, E), where V is a finite set and $E \subseteq \{\{x, y\} \mid (x, y) \in V \times V \wedge x \neq y\}$; elements of V are *vertices* of G and elements of E are unordered pairs of vertices called *edges*. We say that two vertices in a graph are adjacent if there is an edge between them.

A *directed graph* (or *digraph*) D is an ordered pair (V, A), where V is a finite set and $A \subseteq V \times V \setminus \{(x, x) \mid x \in V\}$; elements of V are *vertices* of D and elements of A are ordered pairs of vertices called *arcs* (or *(directed) edges*). A digraph $D = (V, A)$ is *transitive* if any three vertices v_1, v_2, v_3 such that edges $(v_1, v_2), (v_2, v_3) \in A$ imply $(v_1, v_3) \in A$. We say that two vertices in a digraph are adjacent if there is an arc between them, independently of the direction of the arc. A *$(v_1, v_k$-$)$path* in a directed graph $D = (V, A)$ is a sequence of $k \geq 1$ distinct vertices $v_1, \ldots, v_k$ such that $(v_i, v_{i+1}) \in A$ for all $i \in \{1, \ldots, k-1\}$. Given two vertices u and v in a digraph D, we say that v is *reachable* from u if D contains a u, v-path; otherwise, we say that v is *unreachable* from u. In a directed graph $D = (V, A)$ we distinguish the outgoing arcs from a vertex from the incoming arcs. *Outgoing arcs* from a vertex v are arcs having vertex v as its first coordinate, that is, arcs in A of the form (v, v') for some vertex $v' \in V$. *Incoming arcs* to a vertex v are arcs having vertex v as its second coordinate, that is, arcs in A of the form (v', v) for some vertex $v' \in V$.

See Fig. 1 for an example of a graphical representation of a graph G and a directed graph D.

We say that two graphs $G = (V_1, E_1)$ and $H = (V_2, E_2)$ are isomorphic if there exists a bijective function $f : V_1 \to V_2$ such that

$$uv \in E_1 \text{ if and only if } f(u)f(v) \in E_2 \text{ for all } u, v \in V_1.$$

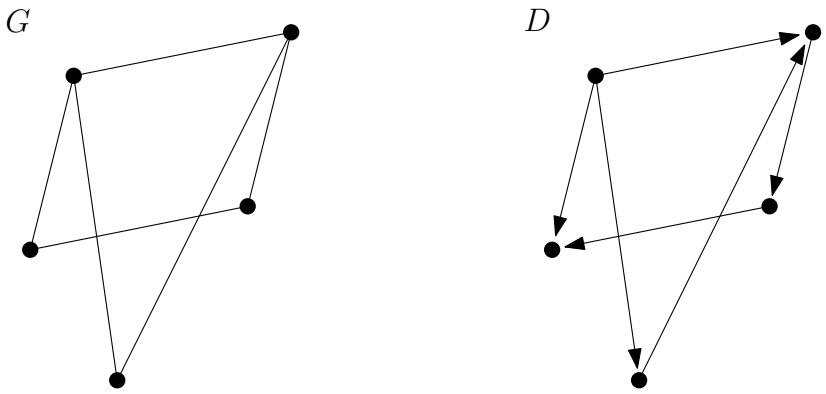

Figure 1: An example of a graph G and a directed graph D.

Let $G = (V, E)$ be a graph. By $\chi(G)$ we denote the *chromatic number* of G, defined as the smallest number of colors needed to color the vertices of the graph so that no two adjacent vertices share the same color. See Fig. 2 for an example.

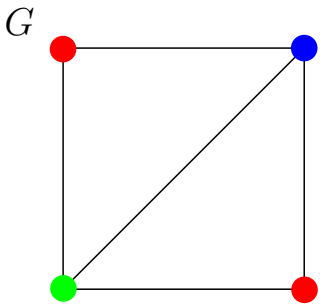

Figure 2: An example of a graph G with $\chi(G) = 3$.

The *complement* of a graph $G = (V, E)$ is the graph $\overline{G}$ on the same vertex set V such that two distinct vertices are adjacent in $\overline{G}$ if and only if they are not adjacent in G. See Fig. 3 for an example.

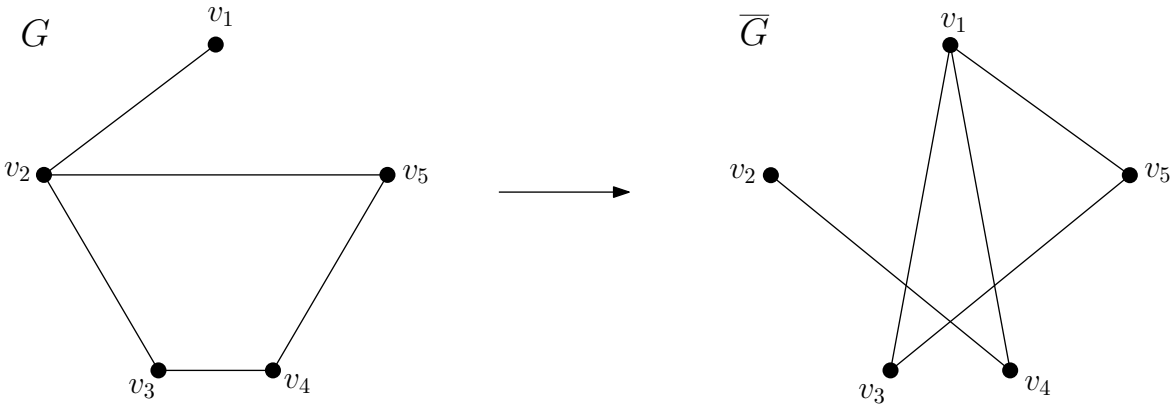

Figure 3: A graph G and its complement.

A graph is *acyclic* if it contains no cycles. A *tree* is a connected acyclic graph. A

clique in a graph G is a set of pairwise adjacent vertices. A graph G is a *perfect graph* if the chromatic number of every induced subgraph equals the size of the largest clique of that subgraph.

We say that a graph G is *cubic* if every vertex of G is incident with exactly three edges, i.e., if for every vertex $v \in V(G)$ we denote $E(v) = \{e \in E(G) \mid v \in e\}$, then the graph G is cubic if and only if $|E(v)| = 3$ for all $v \in V(G)$.

A *vertex cover* of a graph G is a subset $C \subseteq V(G)$ such that for all $e = \{v_1, v_2\} \in E(G)$ either $v_1 \in C$ or $v_2 \in C$. See Fig. 4 for an example.

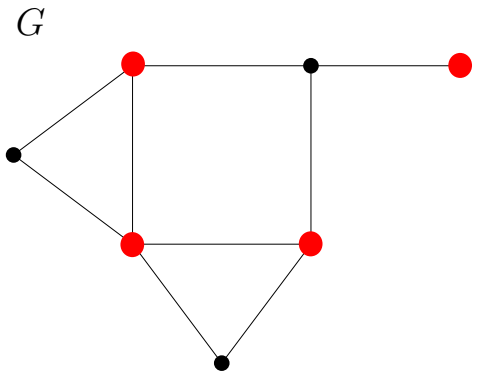

Figure 4: A graph G and one of its vertex covers (shown by vertices in red).

2.2 Other definitions

An *optimization problem* Π is a 4-tuple (D, S, f, opt) where:

- D is a set of *input instances*.
- For every $x \in D$, set $S(x)$ is the set of all *feasible solutions* for instance x.
- $f : \cup_{x \in D} S(x) \to \mathbb{R}$ is a function that assigns a value to each solution.
- $\text{opt} \in \{\min, \max\}$.

A *minimization problem* Π is an optimization problem (D, S, f, opt) such that $\text{opt} = \min$. Given $x \in D$, the task of a minimization problem Π is to find a solution $sol^x_{opt} \in S(x)$ such that for all feasible solutions $sol \in S(x)$ the following holds

$$f(sol^x_{opt}) \leq f(sol).$$

We denote the value $f(sol^x_{opt})$ by $opt_\Pi(x)$.

An *approximation algorithm* A for an optimization problem Π is a polynomial-time algorithm such that given an input instance $x \in D$, the algorithm outputs some

$sol \in S(x)$. We denote by $A(x)$ the value $f(sol)$. Furthermore, we say that A has *approximation ratio* α if for every instance $x \in D$, $A(x)$ is within a (multiplicative) factor of α of $opt_\Pi(x)$. (Here $\alpha \geq 1$ is a real number the value of which could depend on x.) Equivalently, we say that A is an *α-approximation algorithm*.

To understand the definition of the main optimization problem investigated in this thesis it is crucial to understand what the bitwise OR operation does. Hence, let us give a formal definition. *Bitwise OR* is a binary operation that takes two binary sequences of equal length as input and performs the logical inclusive OR operation on each pair of corresponding bits. In other words, the resulting sequence has entry 0 if both corresponding bits are 0, and 1 otherwise. See Table 1 for an example.

Table 1: An example of bitwise OR operation

	10100
OR	01101
=	11101

The bitwise OR operation is commutative and associative, and can thus be naturally extended to any finite number of input binary sequences of the same length.

In computational complexity theory, we refer to NP as the class of decision problems such that if the answer is 'yes', that can be verified by a polynomial-time algorithm if the input instance is equipped with a suitable "certificate". As mentioned in Section 6 and shown in [18], the MUB (and consequently MCRS) problem (see Sections 3 and 5 for formal definitions) belongs to the class of APX-hard problems. Hence, it is crucial to understand what this means. APX is the class of all NP optimization problems that allow approximation algorithms with a constant approximation ratio. Furthermore, an optimization problem is *APX-hard* if there exists some $\epsilon > 0$ such that it is not possible to approximate the problem in polynomial time to within a factor of $(1 + \epsilon)$ unless P = NP.

To demonstrate APX-hardness of an optimization problem, a commonly used tool is the so-called L-reduction scheme. Let us introduce this concept formally.

Let Π and Π' be two optimization problems. Recall that for an instance x of Π, we denote by $opt_\Pi(x)$ the optimal value of Π given x, and similarly for Π'. Problem Π is said to be *L-reducible* to Π' if there exists a polynomial-time transformation f mapping instances of Π to instances of Π' and constants $a, b \in \mathbb{R}_+$, where $\mathbb{R}_+$ is the set of positive real numbers, such that for every instance x of Π, the following holds:

- for every instance x of Π we have $opt_{\Pi'}(f(x)) \leq a \cdot opt_\Pi(x)$
- for every feasible solution y' of $f(x)$, we can compute in polynomial time a feasible

solution y for x such that

$$|opt_{\Pi}(x) - c_1| \leq b \cdot |opt_{\Pi'}(f(x)) - c_2|,$$

where c_1 and c_2 denote the objective function values of y and y', respectively.

To show APX-hardness of the MUB problem Hujdurović et al. used an L-reduction from the vertex cover problem in cubic graphs.

In the idea of the proof for Lemma 5.6 explained in Section 5 we used the following notion. Let $f : X \to Y$ be a function. We say that f is *one-to-one* (or injective) if for all $a, b \in X$, $f(a) = f(b)$ implies that $a = b$. In other words, f maps distinct elements of the domain to distinct elements of the co-domain.

In addition, at the end of Section 5 we refer to the vertices of containment digraph as to the poset P_M. Let us define what we mean by a poset in our case.

A *strict partially ordered set* is an ordered pair $P = (X, <)$, where X is a set called the *ground set* of P and $<$ is a binary relation of X, that is irreflexive, transitive, and asymmetric. More specifically, for all $a, b, c \in X$:

- $a < a$ does not hold,
- if $a < b$ and $b < c$ then $a < c$,
- if $a < b$ then $b < a$ does not hold.

Note that the condition that relation $<$ is irreflexive is a consequence of the assumption that the relation is asymmetric. In this thesis we will refer to a strict partially ordered set as a *poset*.

3 The Minimum Conflict-free Row Split (MCRS) problem

In this section we introduce the perfect phylogeny evolutionary model and give a formal definition of the MCRS problem.

A *rooted tree* is a tree with a distinguished vertex, called the *root* of the tree.

Definition 3.1. A *perfect phylogeny* is a rooted tree representing the evolutionary history of a set of m objects such that:

- The m objects bijectively label the leaves of the tree, representing different samples.
- There are n binary characters, each labeling exactly one edge of the tree, and representing mutations occurred during the evolution of the tumor.

See Fig. 5 for an example.

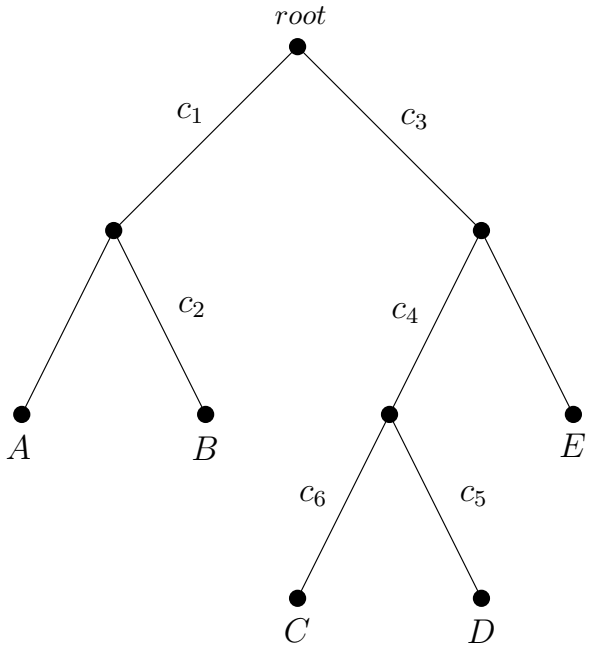

Figure 5: An example of a perfect phylogeny. The figure is adapted from [20].

An $m \times n$ *matrix* is a rectangular array of m rows and n columns in the following form

$$M = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix}$$

The elements of a matrix can be numbers, symbols, or mathematical expressions. We denote by $\{0,1\}^{m\times n}$ the set of all $m \times n$ binary matrices, where a matrix is said to be *binary* if all its elements belong to the set $\{0,1\}$. A perfect phylogeny naturally corresponds to an $m \times n$ binary matrix having objects as rows and characters as its columns. See Fig. 6 for an example.

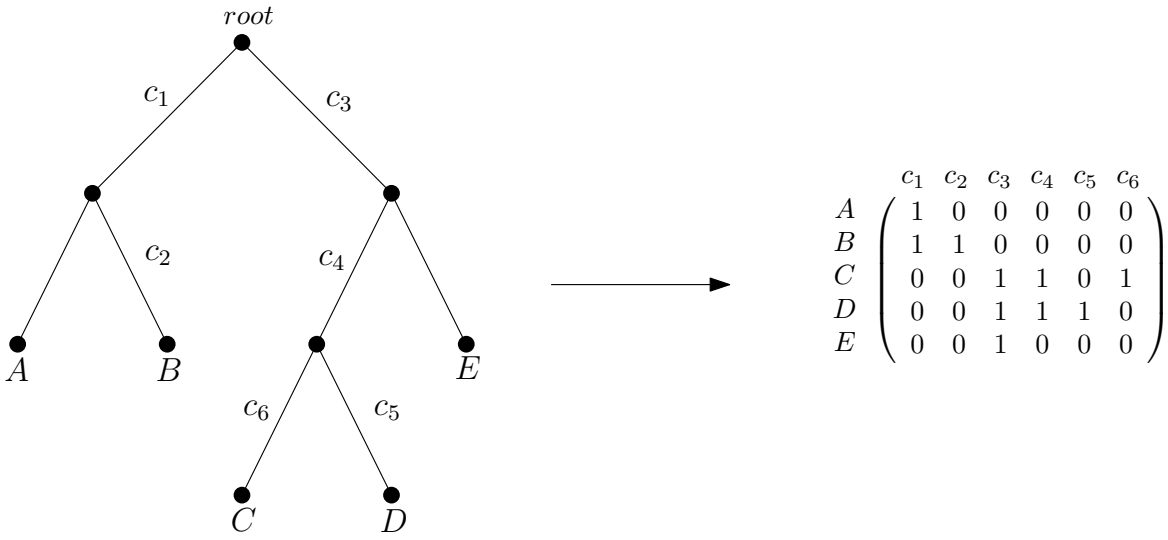

Figure 6: A perfect phylogeny and the corresponding binary matrix M.

However, as mentioned in the motivation, we are interested in the opposite direction. Given a binary matrix M, whose columns and rows represent mutations and samples, respectively, we would like to reconstruct a perfect phylogeny to understand the history of somatic mutations in the corresponding tumor. Firstly, we need to classify matrices which correspond to the evolutionary model of our interest. To this end, the notion of a *conflict-free* binary matrix M will be relevant.

Definition 3.2. Two columns i and j of a binary matrix M are said to be *in conflict* if there exist three rows r, r', r'' of M such that there exists a 3×2 submatrix of M of the following form:

$$M[(r, r', r''), (i, j)] = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$$

A binary matrix M is *conflict-free* if no two of its columns are in conflict.

Otherwise, we say that a binary matrix M is a *conflict* matrix.

It is a known result (see [10], [15]), that the rows of a binary matrix M correspond to a perfect phylogeny if and only if M is conflict-free. Furthermore, if this is the case, a corresponding perfect phylogeny can be computed from M in linear time (see [16] and Section 4).

As mentioned in the introduction, each tumor sample is typically a mixture of reads from several tumor types, and thus the corresponding matrix M is likely to be a conflict matrix. Since we are interested in representing the evolutionary history of somatic mutations by means of the perfect phylogeny model, we would like to explain the conflict binary matrix M containing the measurements with some conflict-free binary matrix M'. To tackle the problem, Hajirasouliha and Raphael (see [17]) introduced so-called *Minimum-Split-Row* optimization problem to minimally decompose the subset of samples (rows), so that they form perfect phylogeny. Minimality is a common criterion used when modeling real-life problems, and it is one of the simplest assumptions one can make. In the master thesis we will refer to the *Minimum Conflict-free Row Split* problem introduced by Hujdurović et al. in [19], following the work of Hajirasouliha and Raphael. To formally state the problem, we will need the following definition.

Definition 3.3. Let $M \in \{0,1\}^{m\times n}$. Let $r_1, \ldots, r_m$ denote the rows of M. We say that a binary matrix $M' \in \{0,1\}^{m'\times n}$ is a *row split* of M if there exists a partition of the set of rows of M' into m sets $R_1, \ldots, R_m$ such that for all $i \in \{1, \ldots, m\}$, r_i is the bitwise OR of the binary vectors in R_i. See Fig. 7 for an example.

$$M = \begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \\ r_4 \end{array} \begin{array}{c} \begin{array}{ccccc} c_1 & c_2 & c_3 & c_4 & c_5 \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 \end{pmatrix} \end{array} \longrightarrow M' = \begin{array}{c} \\ R_1 \\ R_2 \\ R_3 \\ R_4 \end{array} \begin{array}{c} \begin{array}{ccccc} c_1 & c_2 & c_3 & c_4 & c_5 \end{array} \\ \left(\begin{array}{ccccc} 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ \hline 1 & 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 0 & 1 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \end{array}\right) \end{array}$$

Figure 7: An example of a conflict binary matrix M and a conflict-free row split M' of M. The figure is adapted from [18].

Let us denote the minimum number of rows in a conflict-free row split M' of M by $\gamma(M)$. We introduce the following problem:

MINIMUM CONFLICT-FREE ROW SPLIT (MCRS):

Input: A binary matrix M.
Task: Compute $\gamma(M)$.

We remark that a trivial solution to the *Minimum Conflict-free Row Split* problem can be obtained by splitting each row r of M into as many rows of the $n \times n$ identity matrix I_n as the number of ones in r. However in order to obtain a meaningful, in terms of the motivation, conflict-free row split M', we are interested in solving the MCRS problem.

4 A linear-time algorithm for reconstructing a perfect phylogeny

Dan Gusfield introduced in [16] an $\mathcal{O}(mn)$ time algorithm for checking whether an $m \times n$ binary matrix M is conflict-free and an algorithm for reconstructing a perfect phylogeny from a conflict-free binary matrix M. In this section we give an overview of these two algorithms.

4.1 A linear algorithm that tests whether a binary matrix corresponds to a perfect phylogeny

Before we give an overview of the algorithm we give several definitions. A *binary number* is number expressed in the binary numeral system, that is, a number that uses symbols 0 and 1. We refer to each digit in a number as a *bit*. The *most significant bit* is the bit position with the greatest value. Let us more explicitly define the notion of a value. Every non-negative integer can be uniquely represented as a binary number and vice-versa.

Example 4.1. Let us consider as an example a binary number from Fig. 8.

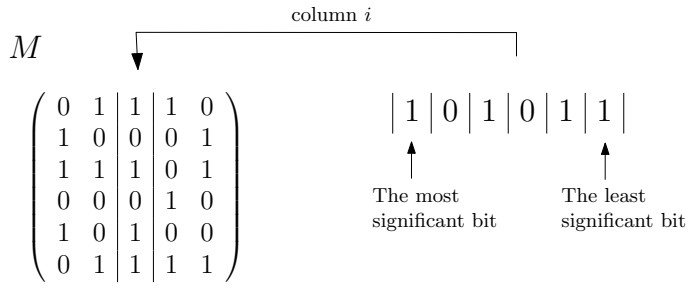

Figure 8: A binary matrix M and graphical representation of its column c_i viewed as a binary number having the most significant bit in row 1.

$$101011_2 = 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 32 + 8 + 2 + 1 = 43_{10}$$

Hence, the leftmost bit has the greatest value, equal to 32.

Before we move on to the algorithm, recall that in the thesis we assume that there are no duplicated columns in a binary matrix M.

Table 2: Algorithm 1

Algorithm: Testing whether a binary matrix corresponds to perfect phylogeny

Input: A binary matrix $M \in \{0,1\}^{m\times n}$
Output: 'Yes' if M is conflict-free / 'No' if M is a conflict matrix

1. Columns of a binary matrix M may be viewed as a binary numbers with the most significant bit in the first row and the least significant bit in the last row. See Fig. 8 for a graphical representation. Firstly, we sort the numbers into decreasing order, that is, we place the largest number in column 1 and the smallest in column n. We denote the sorted matrix by $\tilde{M}$. See the figure below for an example of a binary matrix M and sorted binary matrix $\tilde{M}$.

M $\qquad\qquad$ $\tilde{M}$

$$\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{pmatrix} \longrightarrow \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

2. Let $S = \{(i,j) \mid \tilde{M}(i,j) = 1,\ i \in \{1,\dots,m\},\ j \in \{1,\dots,n\}\}$. For each $(i,j) \in S$ we define $L(i,j)$ to be the largest $k \in \{1,\dots,n\}$ such that $k < j$ and $(i,k) \in S$. If there does not exist such a k, set $L(i,j) = 0$. In addition we define, for all $j \in \{1,\dots,n\}$, $L(j)$ to be the largest $L(i,j)$ with $(i,j) \in S$.

3. If $L(i,j) = L(j)$ for all $(i,j) \in S$, then the answer is 'Yes'. Otherwise 'No'.

Remark 4.2. The columns can be sorted in time $\mathcal{O}(nm)$ by using the so-called *radix sort* sorting algorithm. For the details see [2].

Let M be a binary matrix. We will denote a multiset of its columns and rows by C_M and R_M, respectively. Given a column $c_j \in C_M$, the *support* of c_j is the set defined as follows: $\{r_i \in R_M : M_{i,j} = 1\}$. The set is denoted by $\mathrm{supp}_M(c_j)$. The conflict-freeness property of a binary matrix M can be defined equivalently in the following way. A binary matrix M is conflict-free if and only if for every pair of columns c_i and c_j, either $\mathrm{supp}_M(c_j) \cap \mathrm{supp}_M(c_i) = \emptyset$ or one contains the other. Otherwise, we say that c_i and c_j are in conflict. The correctness of the above algorithm is a consequence of the following results.

Lemma 4.3. *A binary matrix M is conflict-free if and only if the corresponding sorted binary matrix $\tilde{M}$ is conflict-free.*

The above lemma follows from the fact that any permutation of the columns of a binary matrix M maps any two columns that are in conflict in M to a pair of columns that are in conflict in the permuted matrix, and similarly for pairs of columns that are not in conflict.

Theorem 4.4. *Matrix $\tilde{M} \in \{0,1\}^{m\times n}$ is conflict-free if and only if $L(i,j) = L(j)$ for all $(i,j) \in S$.*

Proof. Firstly, assume that $L(i,j) = L(j)$ for all $(i,j) \in S$. For every column index j, if $L(j) \neq 0$ then $\text{supp}_M(c_j) \subset \text{supp}_M(c_{L(j)})$ since $\tilde{M}$ is a sorted binary matrix. In addition, for every column $j \in C_M$ and for every column k strictly between $L(j)$ and j, we have that $\text{supp}_M(c_k) \cap \text{supp}_M(c_j) = \emptyset$, since we assumed that $L(i,j) = L(j)$ for all $(i,j) \in S$. For an arbitrary column index j let $L(j) = k > 0$ and $L(k) = k'$. If $k' > 0$, then $\text{supp}_M(c_j) \subset \text{supp}_M(c_{k'})$ since $\tilde{M}$ is a sorted binary matrix and $\text{supp}_M(c_j) \cap \text{supp}_M(c_p) = \emptyset$ for any p such that $k' < p < k$, since $L(i,j) = L(j)$ for all $(i,j) \in S$. And finally, if $k' = 0$, then we have that $\text{supp}_M(c_j) \cap \text{supp}_M(c_p) = \emptyset$ for every p from 1 to $k-1$, again since $L(i,j) = L(j)$ for all $(i,j) \in S$. A similar argument holds for all columns c_j and c_p such that $p < j$. Since j was arbitrary, we conclude that no pair of columns of M are in conflict.

Assume that matrix $\tilde{M}$ is conflict-free and suppose for a contradiction that there exists rows $r, r' \in \{1,\ldots,m\}$ and a column c_j with $j \in \{1,\ldots,n\}$ such that $(r,j) \in S$, $(r',j) \in S$, $L(j) = L(r,j) = k$, and $L(r',j) = k' < k$. Denote by b_k and b_j the numbers representing columns c_k and c_j, respectively. Since $L(r,j) = k = L(j)$ and $L(r',j) = k'$, we know that $\tilde{M}(r,k) = \tilde{M}(r,j) = 1$, $\tilde{M}(r',j) = 1$ and $\tilde{M}(r',k) = 0$. Since $b_k > b_j$, there exist a row $r'' \in \{1,\ldots,m\}$ such that $\tilde{M}(r'',k) = 1$ and $\tilde{M}(r'',j) = 0$. Hence,

$$M[(r,r'',r'),(k,j)] = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix},$$

a contradiction. □

4.2 A linear algorithm for reconstructing a perfect phylogeny

While the algorithm given in Section 4.1 decides that M corresponds to a perfect phylogeny, the linear-time algorithm presented next constructs one.

Before we move on to the algorithm, let us intoroduce a relevant notion. Let T be a rooted tree. A *leaf* is a vertex $v \neq r \in V(T)$ of degree 1, where r is the root of T. A *leaf edge* is an edge of T incident with a leaf.

Table 3: Algorithm 2

Algorithm: Constructing a perfect phylogeny T from a conflict-free, sorted binary matrix $\tilde{M}$

Input: A binary matrix $\tilde{M} \in \{0,1\}^{m \times n}$ obtained in step 1 of Algorithm 1 in Table 2.
Output: A perfect phylogeny T representing M.

1. Firstly, for every $c_j \in C_{\tilde{M}}$ create a node n_j. Then, create a root r. For all $j \in \{1, \ldots, n\}$ such that $L(j) = 0$ create an edge (r, n_j) and label the edge with character c_j. For all $j \in \{1, \ldots, n\}$ such that $L(j) > 0$ create an edge $(n_{L(j)}, n_j)$ and label the edge with character c_j.

2. For every $r_i \in R_{\tilde{M}}$ let m_i be the largest index such that $(r_i, m_i) \in S$, defined in Algorithm 1 in Table 2. Let $e = (n_i, n_j)$ be the edge labeled by m_i.

   (a) If e is a leaf edge, we label the leaf incident to e by r_i.

   (b) Otherwise, we create an edge connecting vertex n_j with a new leaf and label the leaf by r_i.

See Figure 9 for an example of a sorted binary matrix $\tilde{M}$ and the corresponding perfect phylogeny obtained using Algorithm 2.

Theorem 4.5 (Gusfield)**.** *Algorithm 2 correctly builds a perfect phylogeny T for M.*

For the proof of the above theorem see [16].

$$\tilde{M} = \begin{array}{c} \\ r_1 \\ r_2 \\ r_3 \\ r_4 \\ r_5 \end{array} \begin{array}{c} \begin{array}{ccccc} c_1 & c_2 & c_3 & c_4 & c_5 \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

$$S = \{(1,1),(1,2),(1,3),(2,1),(2,4),(2,5),(3,1),(3,2),(4,1),(4,4),(5,1)\}$$

$$L(1) = 0;\ L(2) = 1;\ L(3) = 2;\ L(4) = 1;\ L(5) = 4$$

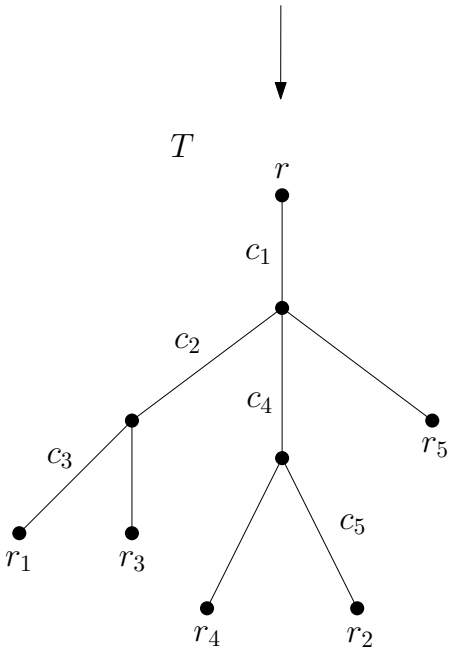

Figure 9: A sorted binary matrix $\tilde{M}$ and the corresponding perfect phylogeny obtained using Algorithm 2.

5 The Minimum Uncovering Branching (MUB) problem

Hujdurović et al. showed in [18] that the MCRS problem can be restated as an optimization problem in terms of branchings in directed acyclic graphs, more specifically in terms of the so-called containment digraph D_M corresponding to the binary matrix M, which we define in this section. This equivalence led to a strengthening of the computational complexity results from [19], development of approximation algorithms, and an improvement of a heuristic for the MCRS problem from [19]. First, we provide the necessary definitions.

Recall that given a column $c_j \in C_M$, the *support* of c_j is the set defined as follows: $\text{supp}_M(c_j) = \{r_i \in R_M : M_{i,j} = 1\}$.

Definition 5.1. Let $M \in \{0,1\}^{m\times n}$ be a binary matrix. The *containment digraph* $D_M = (V, A)$ corresponding to M has vertex set $V = \{\text{supp}_M(c) \mid c \in C_M\}$ and an arc set $A = \{(v_i, v_j) \mid v_i, v_j \in V \text{ and } v_i \subset v_j\}$.

See Fig. 10 for an example.

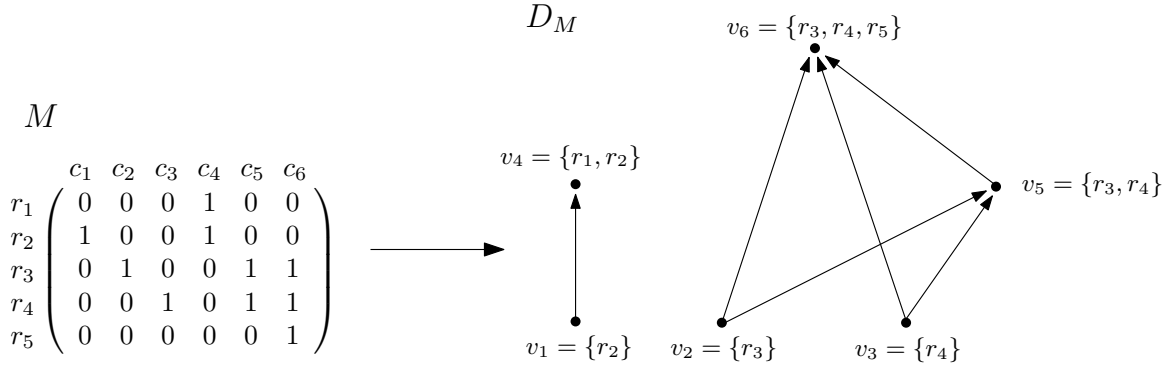

Figure 10: An example of a binary matrix M and the corresponding containment digraph D_M.

Definition 5.2. Let $D_M = (V, A)$ be the containment digraph of a binary matrix M. A *branching* of D_M is a subset B of A containing at most one outgoing arc from each vertex.

Let $M \in \{0,1\}^{m\times n}$ and let $D_M = (V, A)$ be the containment digraph corresponding to M. Let B be a branching of D_M. For a vertex $v \in V$, we define $N_B^-(v) := \{v' \in V \mid (v', v) \in B\}$. Let $r \in R_M$ and let $v \in V(D_M)$, such that $r \in v$. We say that r is *covered* in v with respect to branching B if $r \in \cup N_B^-(v)$. Otherwise, we say that r is *uncovered* in v with respect to B. A *B-uncovered pair* is a pair (r, v) such that $r \in R_M$, $v \in V(D_M)$, and r is uncovered in V with respect to B. Let us denote by $U(B)$ the set of all B-uncovered pairs and for $r \in R_M$ we denote by $U_B(r)$ the set of all B-uncovered pairs with the first coordinate equal to r. See Example 5.3 for a better understanding of the notions of branching and uncovered pairs.

Example 5.3. The six uncovered pairs with respect to the branching B shown on Fig. 11 are the following: $U(B) = \{(r_2, v_1), (r_3, v_2), (r_4, v_3), (r_1, v_4), (r_3, v_5), (r_5, v_6)\}$.

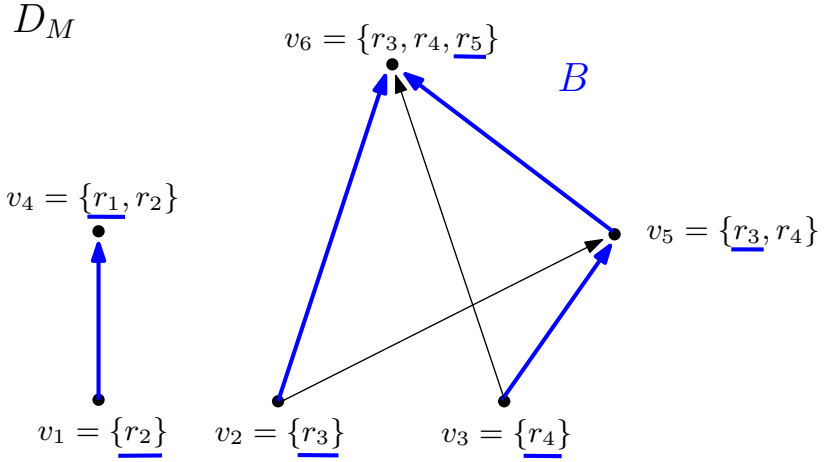

Figure 11: An example of a branching B (shown in blue) and six uncovered pairs (that are underlined) with respect to the branching.

Let us denote the minimum number of uncovered pairs over all branchings B of D_M by $\beta(M)$. Then, the Minimum Uncovering Branching problem is the following:

Minimum Uncovering Branching (MUB):

Input: A binary matrix M.
Task: Compute $\beta(M)$.

Next, we describe a process of reconstructing a conflict-free binary matrix M' of M given an optimal branching B with $|U(B)|$ number of uncovered pairs. Let $M \in \{0,1\}^{m\times n}$ be a binary matrix. Let $R_M = \{r_1, \ldots, r_m\}$ and $C_M = \{c_1, \ldots, c_n\}$. Let D_M

be the corresponding containment digraph and B be an arbitrary branching of D_M. Let $U(B) = \{u_1, \ldots, u_k\}$. We define the *B-split* of M, denoted by M^B, as the following matrix. The rows and columns of the matrix M^B are indexed by $\{u_1, \ldots, u_k\}$ and $\{c'_1, \ldots, c'_n\}$, respectively. Let $v_j = \mathrm{supp}_M(c_j)$ for all $j \in \{1, \ldots, n\}$ and $V = V(D_M)$. For $v \in V$, denote by $B^+(v)$ the set of all vertices in V reachable by a path from v in (V, B). For all $(r, v) \in U(B)$ and all $j \in \{1, \ldots, n\}$, set

$$M^B_{(r,v),j} = \begin{cases} 1, & \text{if } v_j \in B^+(v); \\ 0, & \text{otherwise.} \end{cases}$$

See Fig. 12 for an example.

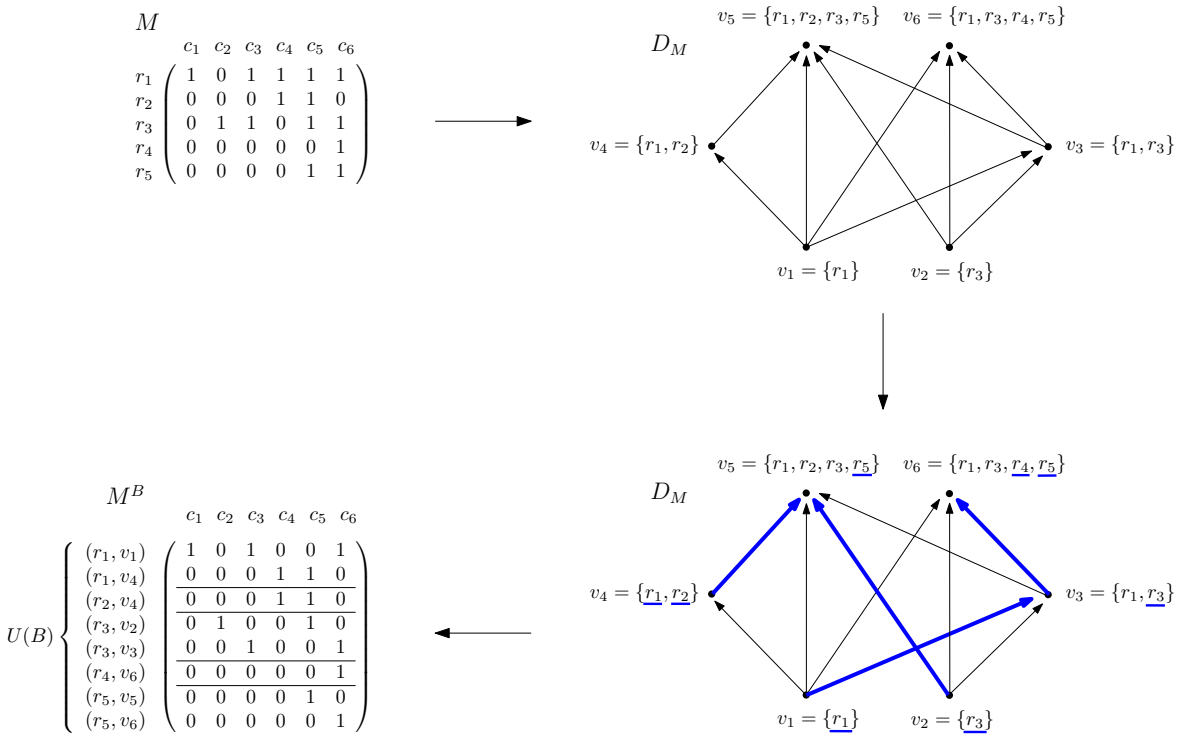

Figure 12: An example showing a binary matrix M, its containment digraph D_M, a branching B (shown in blue), the underlined elements in vertices of D_M, corresponding to the B-uncovered elements, and the row split M^B of M obtained from branching B.

The following lemma from [18] shows that the B-split of M is indeed a conflict-free row split of M.

Lemma 5.4. *Let M be a binary matrix and D_M its containment digraph. Let B be an arbitrary branching of D_M and M^B the B-split of M. Then M^B is a conflict-free row split of M with $|U(B)|$ rows, obtained by splitting each row r_i of M into rows of M^B indexed by the elements of $U_B(r_i)$.*

Proof. The proof idea of Lemma 5.4 is the following. The number of rows in M^B is, clearly, $|U(B)|$ by the definition of a B-split. We have to show that the row r is the

bitwise OR of the rows of M^B indexed by $U_B(r)$. Since M is a binary matrix, there are two possible values in each entry of row r. Firstly assume that $M_{r,j} = 1$ for some j. By the definition of the containment digraph, we have $r \in v_j$. Now we have to show that there exist a vertex $v \in V(D_M)$ such that $(r, v) \in U_B(r)$ and $M^B_{(r,v),j} = 1$. If r is B-uncovered in v_j we are done as in this case $(r, v_j) \in U_B(r)$ and $M^B_{(r,v_j),j} = 1$ since $v_j \in B^+(v_j)$. Otherwise, there exist some $v_k \in N^-_B(v_j)$ such that $r \in v_k$, and if r is covered in v_k we repeat the argument with a "covering" in-neighbor of v_k. Since $|C_M|$ is finite and (V, B) is acyclic, the process will stop after finitely many steps. Hence we can assume w.l.o.g. that $(r, v_k) \in U_B(r)$ and $M^B_{(r,v_k),j} = 1$ since $v_j \in B^+(v_k)$. If $M_{r,j} = 0$ for some j, then clearly $M^B_{(r,v),j} = 0$ for all $v \in V(D_M)$ such that $(r, v) \in U_B(r)$. Hence M^B is a row split of M.

Finally, we have to show that M^B is conflict-free and we do this by contradiction. Assume that M^B is a conflict matrix. Then there exist row indices $(r_i, v_{i'}), (r_j, v_{j'})$ and $(r_k, v_{k'})$ and column indices $p, q \in V(D_M)$ such that $M^B_{(r_i,v_{i'}),p} = M^B_{(r_i,v_{i'}),q} = M^B_{(r_j,v_{j'}),p} = M^B_{(r_k,v_{k'}),q} = 1$ and $M^B_{(r_j,v_{j'}),q} = M^B_{(r_k,v_{k'}),p} = 0$. Since $M^B_{(r_i,v_{i'}),p} = M^B_{(r_i,v_{i'}),q} = 1$, we have that $v_p, v_q \in B^+(v_{i'})$. However, since B is a branching and in every branching there is at most one outgoing edge from each vertex, we must have either $v_q \in B^+(v_p)$ or $v_p \in B^+(v_q)$. Assume w.l.o.g. that $v_q \in B^+(v_p)$.

Since $M^B_{(r_j,v_{j'}),p} = 1$, it follows that $v_p \in B^+(v_{j'})$, which implies that $v_q \in B^+(v_{j'})$. Since $r_j \in v_{j'}$ and $v_q \in B^+(v_{j'})$ it follows that $r_j \in v_q$ and $M^B_{(r_j,v_{j'}),q} = 1$, which is a contradiction. Hence, M^B is conflict free. □

The following theorem, again from [18], captures the result showing that the MCRS and MUB problems are equivalent to each other, with equal optimal values. In the theorem we denote by ω any real number such that there exists an $\mathcal{O}(n^\omega)$ algorithms for multiplying any two $n \times n$ binary matrices (e.g., $\omega = 2.3728639$, see [13]).

Theorem 5.5 (Hujdurović et al., 2018)**.** *For every binary matrix $M \in \{0,1\}^{m\times n}$ the following holds:*

1. *Any branching B of D_M can be transformed in time $\mathcal{O}(n^2m)$ to a conflict-free row split M' of M with exactly $|U(B)|$ rows.*

2. *Any conflict-free row split $M' \in \{0,1\}^{m'\times n}$ of M can be transformed in time $\mathcal{O}(m'n^2 + n^\omega)$ to a branching B of D_M such that $|U(B)|$ is at most the number of rows in M'.*

In particular, $\gamma(M) = \beta(M)$.

Theorem 5.5 is proved in [18] in two steps. The proof of the first part of the theorem relies on Lemma 5.4 and the second part on the following lemma.

Lemma 5.6. *There exists an algorithm that takes as input a binary matrix M and a conflict-free row split M' of M and computes in time $\mathcal{O}(m'n^2 + n^\omega)$ a branching B of D_M such that M^B can be obtained from M' by removing some rows.*

Proof. The proof idea of Lemma 5.6 is the following. Assume that we are given a binary matrix M and a conflict-free row split M' of M. Let us introduce some notation. Denote the columns and the rows of M by $c_1, \ldots, c_n$ and $r_1, \ldots, r_m$, respectively. Let R_i be the set of rows splitting r_i, and denote the set of columns of M' by $c'_1, \ldots, c'_n$, where column c'_i of M' corresponds to column c_i of M. Denote by D_M the containment digraph corresponding to M and by $D_{M'}$ the containment digraph corresponding to M'. For $i \in \{1, \ldots, n\}$, let $v_i = \mathrm{supp}_M(c_i)$ and $v'_i = \mathrm{supp}_{M'}(c'_i)$. We say that an arc (v'_i, v'_j) of $D_{M'}$ is *elementary* if it is not a consequence of transitivity, that is, there exists no $k \in \{1, \ldots, n\}$ such that both (v'_i, v'_k) and (v'_k, v'_j) are arcs in D'_M.

Define a subset B of the arc set of D_M as follows: $(v_i, v_j) \in B$ if and only if $v'_i \neq \emptyset$ and (v'_i, v'_j) is an elementary arc of $D_{M'}$. We claim that B is a branching of D_M. Initially, it has to be shown that the branching is well defined, that is, that an arc $(v'_i, v'_j) \in A(D_{M'})$ implies the existence of an arc $(v_i, v_j) \in A(D_M)$ for all $i, j \in \{1, \ldots, n\}$. Assume, by contradiction, that there exist an arc such that $(v'_i, v'_j) \in A(D_{M'})$ and $(v_i, v_j) \notin A(D_M)$. By definition of the containment digraph, this implies that $v'_i \subset v'_j$ and $v_i \not\subset v_j$. Let $r_k \in v_i \setminus v_j$. Then $M'_{r',i} = 1$ for some $r' \in R_k$. This implies that $M'_{r',j} = 1$, since by assumption $v'_i \subset v'_j$, a contradiction, since $r_k \notin v_j$. Further, we have to show that B is a branching, that is, the out-degree of each vertex is at most one with respect to the branching. Assume for a contradiction that $(v_i, v_j) \in B$ and $(v_i, v_k) \in B$ for some vertex v_i and $j \neq k$. Then $(v'_i, v'_j), (v'_i, v'_k)$ are elementary arcs in $D_{M'}$, which implies that $v'_i \subset v'_k$ and $v'_i \subset v'_j$. Since M' is conflict-free, we have either $v'_j \cap v'_k = \emptyset$, $v'_j \subset v'_k$, or $v'_k \subset v'_j$. Since $v'_i \subset v'_k \cap v'_j$ and $v'_i \neq \emptyset$, either $v'_j \subset v'_k$ or $v'_k \subset v'_j$. By the definition of elementary arcs, we obtain that $v'_j = v'_k$. However, since $v_j \neq v_k$, we can assume w.l.o.g. that exist some $r_p \in v_j \setminus v_k$. Then there exist some $r' \in R_p$ such that $r' \in v'_j$, which implies that $r' \in R_p \cap v'_j$. Since $r_p \notin v_k$, we have that $R_p \cap v'_k = \emptyset$. Since, $v'_k = v'_j$, we infer that $r' \in R_p \cap v'_j = R_p \cap v'_k$, a contradiction.

Next, it has to be shown that M^B can be obtained from M' by removing some rows, or equivalently there exists a one-to-one mapping that maps each row r of M^B to an identical row r' of M', where we say that two rows are identical if the corresponding binary row vectors are the same. We define the one-to-one mapping as follows. We map a row of a B-split M^B indexed by $(r_i, v_k) \in U_B(r_i) \subseteq U(B)$ to the row indexed by (r_i, v_k) of M'. It can be easily verified that the mapping is one-to-one (see [18] for more details). However, it has to be shown that such a mapping is well defined, that is, there exists an identical row in M', more specifically there exists a row $r' \in R(M')$ such that r' equals to the row of M^B indexed by (r_i, v_k). We show this as follows.

Let (r_i, v_k) be a row of M^B. Then by the definition of an uncovered pair, $r_i \in v_k$. By the definition of a row split M', there exist a row $r' \in R_i$ such that $M'_{r',k} = 1$. Assume that $M'_{r',j} = 1$. Since $M'_{r',k} = 1$ and M' is conflict-free, it follows that either $v'_j \subset v'_k$ or $v'_k \subset v'_j$. If $v'_k \subset v'_j$, then we are done. If $v'_j \subset v'_k$, then there exist a path P' in $D_{M'}$ consisting only of elementary arcs of $D_{M'}$, hence path P' corresponds to a nontrivial (nontrivial, because v'_j is non-empty) v_j, v_k-path P in B. Hence $v_k \in B^+(v_j)$, a contradiction. Hence, $M'_{r',j} = 1$ if and only if $v_j \in B^+(v_k)$.

It only remains to estimate the time complexity. The containment digraph $D_{M'}$ of M' can be computed in time $\mathcal{O}(m'n^2)$, then the set of elementary arcs in time $\mathcal{O}(n^\omega)$ using the algorithm of Aho et al. introduced in [1]. And finally, branching B can be computed in time $\mathcal{O}(n^2)$. □

Theorem 5.5 uses Lemmas 5.4 and 5.6 to prove the equivalence. It suffices to show that the time complexity of computing the B-split of M equals $\mathcal{O}(n^2m)$. For more details and a formal proof of Theorem 5.5 and Lemmas 5.4 and 5.6 see [18].

Note that the equivalence improves on the time complexity of a direct brute-force approach that follows immediately from the definitions. Let us consider the time complexity of the approach based on generating all possible row-splits of M. Let us consider a row $r \in R_M$. Assume it has exactly k entries equal to 1. Then there are at least as many splits of row r as there are partitions of a k-element set, which equals the Bell number B_k, which is bounded by 2^k from below. Hence, in general, for a matrix M with m rows and n columns, having at least k_i ones in each row, we get that the total number of row splits is at least $2^{\sum_i k_i}$.

On the other hand, the complexity of the approach that considers all the possible branchings of the containment digraph D_M and choosing the one with least number of uncovered pairs is of the order, that can be expressed as a function of n only. More specifically, the time complexity of the approach is of the order $\mathcal{O}(n^n)$.

We would like to conclude this section by noting that the vertices of the containment digraph D_M correspond to the family of support sets of the columns of M, and arcs correspond to the relation of proper inclusion of those sets. Hence, essentially, we may refer to the containment digraph as a strict partially ordered set $P_M = (V(D_M), \subset)$. We will refer to the poset P_M later on, in Section 10.

6 Complexity and approximation of the MUB problem

In this section we give an overview of complexity results and known approximation algorithms introduced by Hujdurović et al. in [18]. More specifically, as mentioned in Section 1, the MUB problem is known to be APX-hard and there exist approximation algorithms with approximation ratios bounded by the width and the height of the input matrix, respectively.

6.1 Computational complexity

To introduce the theorem that captures the result that MUB problem is APX-hard, we initially introduce the notion of the height of the containment digraph D_M.

Definition 6.1. Let M be a binary matrix and D_M its containment digraph. A *chain* in D_M is the vertex set of a path in D. The *height* of M is the maximum cardinality of a chain in D_M. Notation: $h(M)$.

The following theorem shows that the problem cannot be approximated arbitrarily well already for restricted input instances.

Theorem 6.2 (Hujdurović et al. 2018)**.** *The MUB problem (and consequently the MCRS problem) is APX-hard, even for instances of height 2.*

The above theorem is proved by showing that the vertex cover problem in cubic graphs, which is know to be APX-hard, as mentioned in Section 1, is L-reducible to the MUB problem. The construction introduced in Theorem 10.12 on p. 50 serves as the main ingredient in the construction of an L-reduction. And consequently, since the MUB and MCRS problems are equivalent to each other, the MCRS problem is APX-hard as well. For a formal definition of L-reduction see Section 2 and for a detailed proof of Theorem 6.2 see [18].

The result of the Theorem 6.2 gives rise to the question whether the MUB (and consequently the MCRS) problem admits a constant factor approximation. Note that Hujdurović et al. in [18] consider a variant of the MCRS problem, an optimization

problem called the Minimum Distinct Conflict-free Row Split (MDCRS) problem. The MDCRS problem was initially considered by Hajirasouliha an Raphael in [17] and it minimizes the number of *distinct* rows in a conflict-free row split M' of M. Let us denote by $\eta(M)$ the minimum number of distinct rows in a conflict-free row split M' of M.

Let us formally define another problem on branchings, the Minimum Irreducible Branching (MIB) problem. Let M be a binary matrix and D_M the corresponding containment digraph. For a branching B of D_M, we say that a vertex $v \in V(D_M)$ is *B-irreducible* if there exists some element $r \in v$ that is uncovered in v with respect to B. Denote the set of all B-irreducible vertices by $I(B)$. As an example, on Fig. 12, we have $I(B) = V(D_M)$, however on Fig. 13, we have $I(B) = V(D_M) \setminus \{v_5\}$. Let us denote the minimum number of irreducible vertices over all branchings B of D_M by $\zeta(M)$.

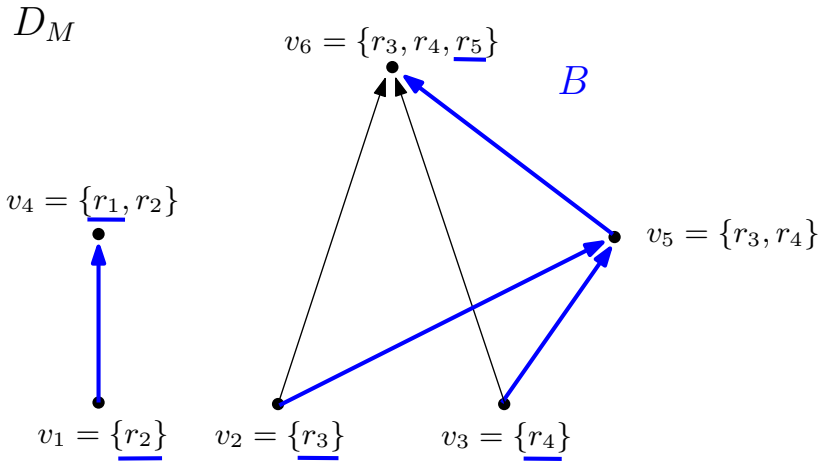

Figure 13: An example of a containment digraph D_M, a branching B (shown in blue) and one B-irreducible vertex v_5.

Then the corresponding optimization problem is the following:

Minimum Irreducible Branching (MIB):

Input: A binary matrix M.
Task: Compute $\zeta(M)$.

The following theorem shows that the MDCRS problem is equivalent to the MIB problem, moreover, the optimal values are equal.

Theorem 6.3 (Hujdurović et al. 2018)**.** *For every binary matrix $M \in \{0,1\}^{m\times n}$ the following holds:*

1. *Any branching B of D_M can be transformed in polynomial time to a conflict-free row split M' of M with exactly $|I(B)|$ distinct rows.*
2. *Any conflict-free row split $M' \in \{0,1\}^{m'\times n}$ of M can be transformed in polynomial time to a branching B of D_M such that $|I(B)|$ is at most the number of distinct rows in M'.*

In particular, $\eta(M) = \zeta(M)$.

It was shown by Hujdurović et al. in [18] that the MIB (and consequently the MCDRS) problem is APX-hard even on instances of height 2 by an L-reduction from vertex cover problem in cubic graphs. The construction from Theorem 10.13 on p. 52 serves as a main ingredient in a construction of an L-reduction.

Going back to our discussion regarding the existence of a constant factor approximation for the MUB problem, the corresponding problem has been solved for the MIB problem. It was shown by Hujdurović et al. in [18] that there exist a simple 2-factor approximation algorithm for the MIB problem. On the other hand, while it remains an open question whether the MCRS (and consequently the MUB) problem admits a constant factor approximation for general input instances, it is shown that the MUB problem admits a constant factor approximation on instances of bounded height and width. We present these results next.

6.2 Known approximation algorithms

In this subsection we give an overview of the known approximation algorithms for the MUB problem, give proof ideas and specify what branchings are output by the algorithms. Let us initially define several notions.

Definition 6.4. An *antichain* in D_M is a set of pairwise non-adjacent vertices. The *width* of D_M, which is a transitive DAG, is the maximum number of vertices in an antichain. Notation: $wdt(M)$.

A *linear branching* of M is a subset of A with at most one outgoing and incoming arc from each vertex. See Fig. 14 for an example.

We say that a linear branching B *consists of k paths* if B is the union of edge sets of k paths not sharing any vertices.

Theorem 6.5 (Hujdurović et al. [18])**.** *Any algorithm that, given a binary matrix M, computes a linear branching B of D_M consisting of $wdt(M)$ paths and returns*

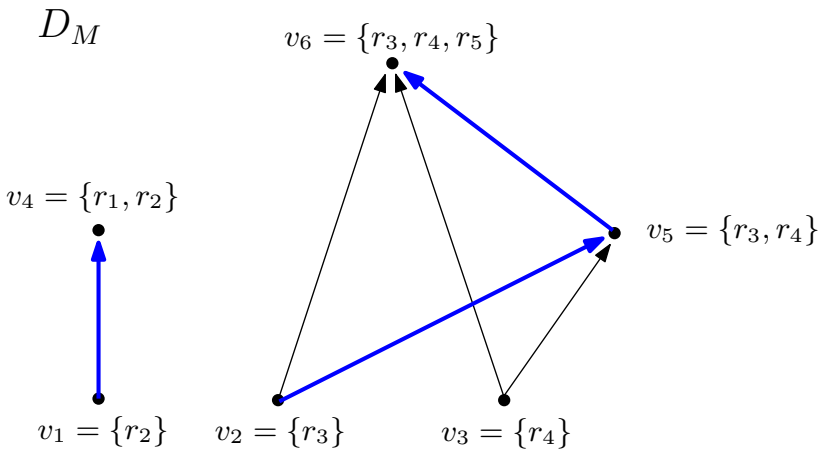

Figure 14: A containment digraph D_M and a linear branching (shown in blue).

the corresponding B-split of M is a $wdt(M)$-approximation algorithm for the MCRS problem.

The corresponding approximation algorithm for the MUB problem outputs any branching arising from an optimal chain partition of D_M (see Section 8 for a formal definiton of a chain partition), which is polynomially computable using Dilworth's Theorem [7] (see Theorem 8.1 in Section 8).

The idea of the proof of Theorem 6.5 is the following. Let M be a binary matrix and D_M its containment digraph and let $w = wdt(M)$. Then by Theorem 8.1 we can compute a chain partition of D_M consisting of w chains in polynomial time. Fix a branching corresponding to such a chain partition. We consider an arbitrary row $r \in R_M$ and the main idea of the proof is that each chain of the partition contains at most one vertex v such that the pair (r, v) is uncovered with respect to the branching. Hence, in total r appears uncovered at most w times in the branching. From this fact, the following inequality can be derived:

$$|U(B)| \leq w|R_M| \leq w\beta(M) = w\gamma(M)\,,$$

where the second inequality follows from the assumption that no row of M has all zero entries.

As the next result shows, an $h(M)$-approximation can be obtained by the B-split corresponding to any branching in D_M. Before we move on to formally stating the theorem, let us define the height of a branching.

Let B be a branching. The *height* of B is the maximum number of vertices in a path contained in (V, B).

Theorem 6.6 (Hujdurović et al. [18])**.** *Let M be a binary matrix and let B be an arbitrary branching of D_M. Then, the number of rows in the B-split of M is at most $h(M) \cdot \gamma(M)$.*

The idea of the proof of Theorem 6.6, expressed again in terms of the MUB problem, is the following. Let M be an binary matrix and let $h = h(M)$. Then, the height of an optimal branching is at most h. Fix an optimal branching B_{opt}. Let (r, v) be an uncovered pair in $U(B_{opt})$. Then at most h pairs of the form (r, v') can be reached in B_{opt} from (r, v) (since it is impossible to reach more vertices than the maximum number of vertices in a directed path). Hence the inequality $|U(B)| \leq h\beta(M)$ holds for any branching B.

Note that the output of the algorithm can be any branching, since there are no restriction in Theorem 6.6. Hence, $B = \emptyset$ would work as well.

For more details regarding the analysis of the two approximation algorithms mentioned above see [18].

7 A polynomially computable lower bound on $\beta(M)$

In this section we give an overview of a known polynomially computable lower bound and a detailed proof of certain results from previous research papers related to the problem investigated in the master thesis.

Hajirasouliha and Raphael proved in [17] the following lower bound.

Definition 7.1. Let M be an arbitrary binary matrix and r be a row of the matrix. The *conflict graph* $G_{M,r}$ is a graph corresponding to matrix M and row r with the following vertex set. We associate a vertex in $G_{M,r}$ with each entry 1 in r, and two vertices in $G_{M,r}$ are connected by an edge if and only if the corresponding columns in $G_{M,r}$ are in conflict.

Lemma 7.2. *For every binary matrix M, we have $\beta(M) \geq \sum_r \chi(G_{M,r})$.*

Before we move on to the proof of the above Lemma, let us introduce several notions.

Definition 7.3. Let M be a binary matrix and r a row of M. Let D_M be the corresponding containment digraph. The *principal subgraph of D_M corresponding to r* is denoted by $D_{M,r}$ and defined as the subgraph of D_M induced by the set of vertices $v \in V(D_M)$ such that $r \in v$. A *principal subgraph* of D_M is any subgraph of the form $D_{M,r}$. See Fig. 15 for an example.

As the next lemma shows, the maximum size of an antichain in the principal subgraph $D_{M,r}$ equals to the chromatic number of the conflict graph $G_{M,r}$. Recall that an *antichain* in $D_{M,r}$ is a set of pairwise non-adjacent vertices.

Lemma 7.4. *The maximum size of an antichain in the principal subgraph $D_{M,r}$ equals the chromatic number of the conflict graph $G_{M,r}$.*

Proof. Let M be a binary matrix and $r \in R_M$ an arbitrary row. Let $D_{M,r} = (V_1, E_1)$ and $G_{M,r} = (V_2, E_2)$. Then $V_1 = V_2$, since the vertices of $D_{M,r}$ and $G_{M,r}$ are the vertices v of D_M such that $r \in v$. Let $G'_{M,r}$ be the undirected underlying graph of $D_{M,r}$ obtained by replacing each directed edge with the corresponding undirected edge. Then, an antichain of $D_{M,r}$ is an independent set in $G'_{M,r}$ and vice versa. By the

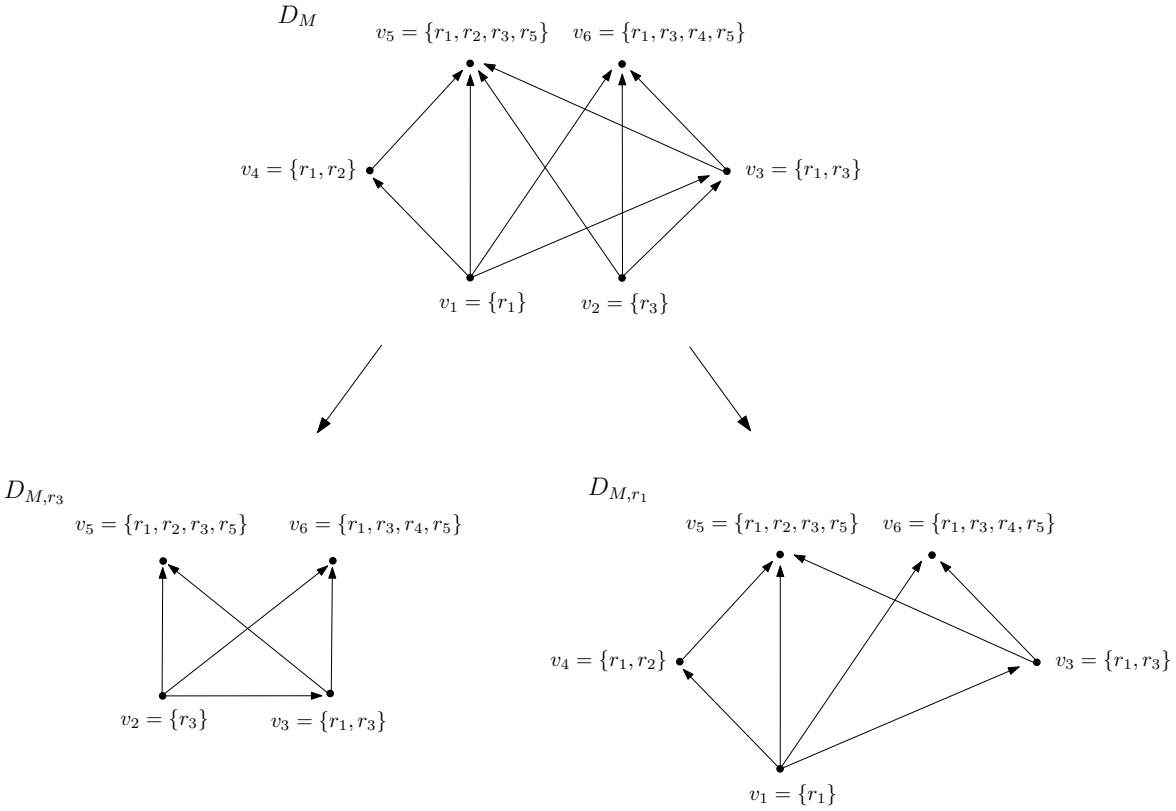

Figure 15: A containment digraph D_M and two of its principal subgraphs, D_{M,r_3} and D_{M,r_1}.

definition of the conflict graph and the principal subgraph, we have that $\overline{G'_{M,r}} = G_{M,r}$. Then the maximum size of an independent set in $G'_{M,r}$ becomes the maximum size of a clique in $G_{M,r}$. Since $G_{M,r}$ is a perfect graph (see [19]) the maximum size of a clique equals its chromatic number. □

Hence Lemma 7.2 has the following equivalent formulation, for which we provide a direct proof.

Lemma 7.5. *For every binary matrix M, we have $\beta(M) \geq \sum_r wdt(D_{M,r})$.*

Proof. Fix an optimal branching B_{opt} of D_M. Let us consider $|U(B_{opt})|$. Let $r \in R_M = \{1, \ldots, m\}$ and $D_{M,r}$ be a principal subgraph. Let $wdt(D_{M,r}) = w$. Then, the maximum size of an antichain in $D_{M,r}$ equals w. Denote such an antichain as follows: $N = \{v_1, \ldots, v_w\}$. For every two distinct vertices, say v_i and v_j, in antichain N, we have that $v_i \not\subset v_j$ and $v_j \not\subset v_i$. For each $i \in \{1, \ldots, w\}$, we will define a vertex $v'_i \in V(D_{M,r})$ such that (r, v'_i) is uncovered with respect to B_{opt}. If (r, v_i) is uncovered with respect to B_{opt}, then we set $v'_i = v_i$. Otherwise, there exists an edge $(v', v_i) \in B_{opt}$ such that $r \in v'$. If $(r, v') \notin U(B_{opt})$ we repeat the argument with v' replaced with a "covering" in-neighbor. The procedure will stop after finitely many steps since $D_{M,r}$

is finite. It remains to show that the mapping $i \mapsto v'_i$ is one-to-one. If this is not the case, say $v'_i = v'_j$ for some $i \neq j$ with $i, j \in \{1, \ldots, w\}$, then using the fact that vertices v_i and v_j are unreachable from each other in $D_{M,r}$, we infer that the union of the two paths from v'_i to v_i and from $v'_j = v'_i$ to v_j consisting only of edges of B_{opt} contains a vertex with two outgoing edges in B_{opt}, a contradiction. Therefore, there are at least w uncovered pairs with first coordinate equal to r. Since this holds for all $r \in R_M$, we obtain $\beta(M) = |U(B_{opt})| \geq \sum_r wdt(D_{M,r})$, as claimed. □

Given a binary matrix M, we denote $W(M) = \sum_r wdt(D_{M,r})$, where $D_{M,r}$ is the principal subgraph of the containment digraph D_M corresponding to r. Lemma 7.5 states that $\beta(M) \geq W(M)$ for every binary matrix M. Please note that this lower bound is polynomially computable by the approach of Fulkerson [12]. Hence, it would be interesting to consider the following question.

Open problem 1. *Does there exist a constant c such that $\beta(M) \leq c \cdot W(M)$ for all binary matrices M?*

An affirmative answer to the Open Problem 1 would lead to a constant factor approximation algorithm for the MUB (and consequently to the MCRS) problem. In Section 10.4 we will show that if the answer to the question is affirmative, then any constant c with the above property has to satisfy the inequality $c \geq 7/6$.

8 A weighted generalization of Dilworth's Theorem

In this section we give an overview of a known result introduced by Hujdurović et al. in [18], which implies the existence of a polynomial-time algorithm for computing an upper bound for the optimal value to the MUB problem discussed in Section 9.

The corresponding result introduced in [18] is a min-max relation, which is a generalization of Dilworth's Theorem applicable to an arbitrary DAG D. However, we will consider only the application of the theorem to the problem investigated in the master thesis, that is, we will consider the containment digraphs D_M instead of an arbitrary DAGs.

In order to state Dilworth's Theorem for general (not necessarily transitive) digraphs, the concepts of chains and antichains have to be redefined in this more general context as follows. A *chain* is a set of vertices that are pairwise reachable from each other. Similarly, an *antichain* is a set of vertices that are pairwise unreachable from each other. A *chain partition* in a digraph D is a family of vertex disjoint chains $P = \{C_1, \ldots, C_p\}$ such that every vertex of D is contained in exactly one chain of P.

Dilworth's Theorem states that the maximum size of an antichain equals to the minimum number of chains in a chain partition of D, where D is any DAG (see [7]).

By applying the approach of Fulkerson [12], a minimum chain partition of D can be computed by solving a maximum matching problem in a derived bipartite graph having $2n$ vertices, which in turn can be done in time $\tilde{\mathcal{O}}(n^\omega)$ using the algorithm of Ibarra and Moran [21], where $n = |V(D)|$ and ω is any real number such that there exists an $\mathcal{O}(n^\omega)$ algorithm for multiplying two $n \times n$ binary matrices. These results are summarized in the following theorem.

Theorem 8.1 (Dilworth's Theorem)**.** *Every DAG D admits a chain partition of size $wdt(D)$. Such a chain partition can be computed in time $\tilde{\mathcal{O}}(|V(D_M)|^\omega)$.*

See Fig. 16 for an example of an antichain and chain partition of size $wdt(D_M)$.

Let us move on to defining an optimization problem that will be used for deriving a min-max theorem. Let us denote the set of non-negative integers by $\mathbb{Z}_+$. Let $D_M = (V, A)$ be the containment digraph corresponding to a binary matrix M and let $f : V \to \mathbb{Z}_+$ be a *weight function* on the vertices of D_M. We say that the weight function

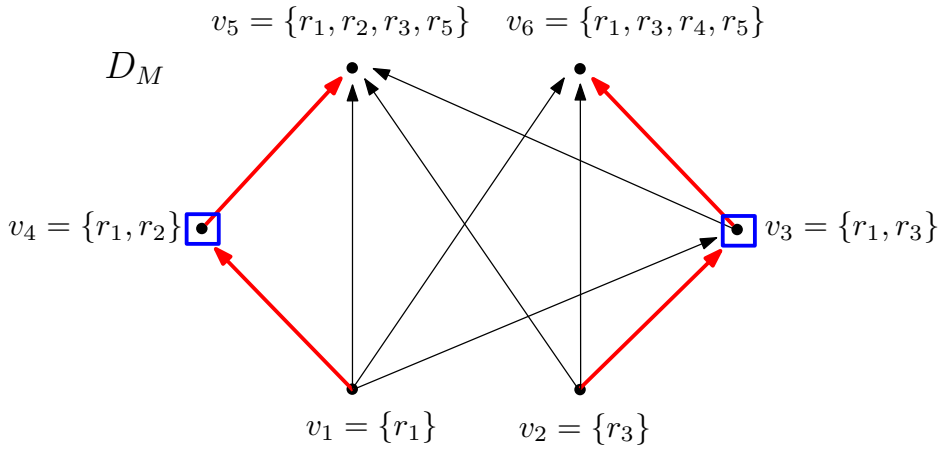

Figure 16: A containment digraph D_M, a maximum antichain (shown in blue squares) and a minimum chain partition (shown by edges in red).

is *monotone* if $f_u \leq f_v$ for every two vertices $u, v \in V$ such that $(u, v) \in A$. Given a chain C in D_M, define the *price of chain* C as follows: $\Pi(C) = \max_{v \in C} f_v$. Given a chain partition P of M, we define the *price* of P as follows: $\Pi(P) = \sum_{i=1}^{p} \Pi(C_i)$.

Remark 8.2. Please note the following. Let C be any chain and N be any antichain of the containment graph D_M. Then $|C \cap N| \leq 1$, since if at least two vertices of C (which is a path in D_M) contained in N, that would contradict the definition of an antichain.

Let us introduce the following optimization problem:

Minimum Price Chain Partition

Input: A binary matrix M, its containment digraph $D_M = (V, A)$ and a monotone weight function $f : V \to \mathbb{Z}_+$ of D_M.
Task: Compute a chain partition P of D_M with minimum price.

A *tower of antichains* of D_M is a sequence of antichains $T = \{N_1, \ldots, N_{wdt(D_M)}\}$ with $|N_i| = i$ for all i. Let us define the *value of an antichain* as $val(N) = \min_{v \in N} f_v$ and the *value of a tower of antichains* as: $val(T) = \sum_{i=1}^{wdt(D_M)} val(N_i)$.

Before we move on to the examples, let us introduce one more definition. Let $D_M = (V, A)$ be a containment digraph. Denote by A^t the set of arcs of D_M that follow from transitivity. The *transitive reduction* of D_M is the directed graph $tr(D_M) = (V, A \setminus A^t)$.

Example 8.3. Let us consider the following binary matrix M and the transitive re-

duction $tr(D_M)$ of its containment digraph.

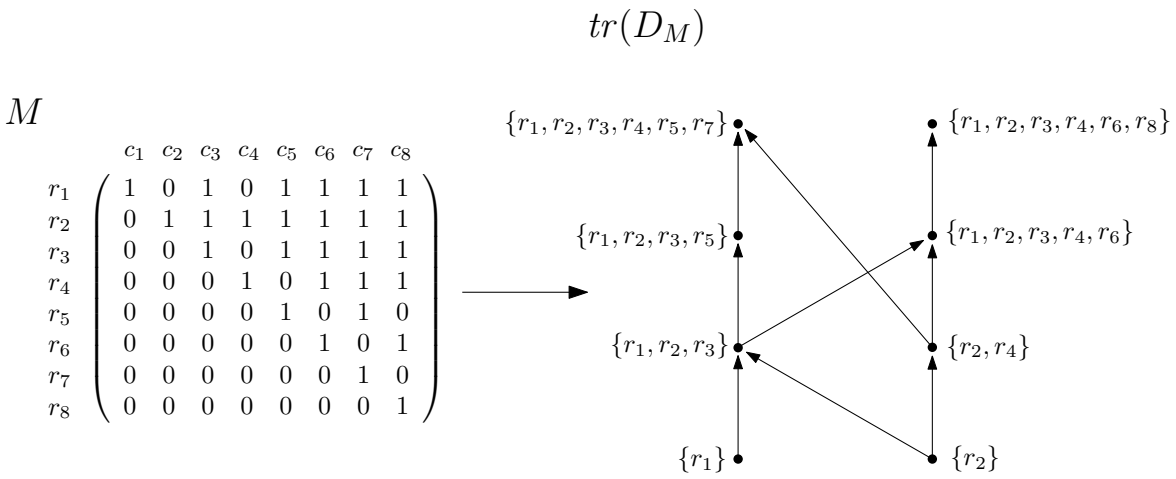

Figure 17: An example of a binary matrix M and the transitive reduction $tr(D_M)$ of its containment digraph.

Define a monotone weight function $f : V(D_M) \to \mathbb{Z}_+$ as follows, $f_v = |v|$ for all $v \in V(D_M)$. Let us partition the vertices of D_M in the following two chains, $P = \{C_1, C_2\}$:

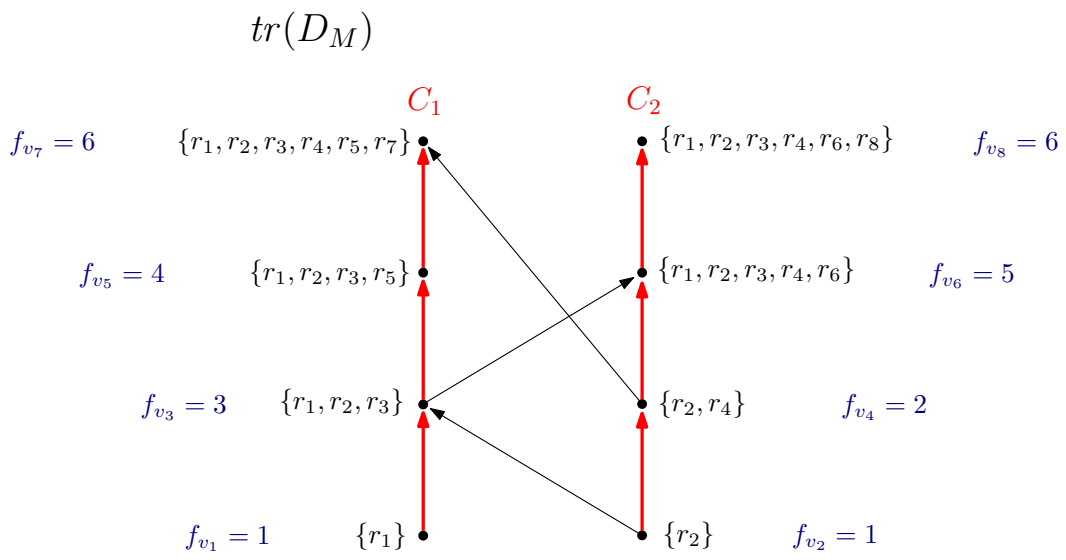

Figure 18: The transitive reduction $tr(D_M)$ of containment digraph D_M partitioned into two chains $\{C_1, C_2\}$

Then $\Pi(C_1) = \max_{v \in C_1} f_v = |v_7| = 6$ and $\Pi(C_2) = \max_{v \in C_2} f_v = |v_8| = 6$, thus $\Pi(P) = \Pi(C_1) + \Pi(C_2) = 12$. Further, let us consider the tower of antichains $T = \{N_1, N_2\}$, graphically represented on Fig. 19.

Then $val(N_1) = \min_{v \in N_1} f_v = |v_2| = 1$ and $val(N_2) = \min_{v \in N_2} f_v = 4$, thus $val(T) = val(N_1) + val(N_2) = 4 + 1 = 5$.

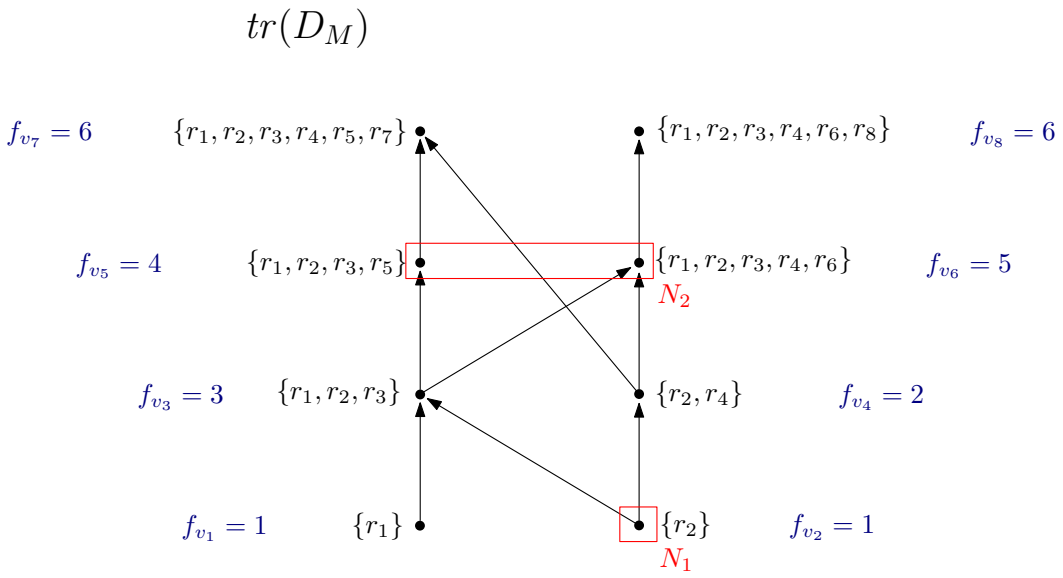

Figure 19: The transitive reduction $tr(D_M)$ of containment digraph D_M and two antichains $\{N_1, N_2\}$

Let us now move on to the main results of the section. First, we introduce the following lemma.

Lemma 8.4 (Hujdurović et al. [18]). *Let M be a binary matrix and D_M its containment digraph. Let $P = \{C_1, \ldots, C_p\}$ be a chain partition of D_M, let $T = \{N_1, \ldots, N_{wdt(D_M)}\}$ be a tower of antichains in D_M. Then, $\Pi(P) \geq val(T)$, for any $f : V \to \mathbb{Z}_+$.*

Proof idea. The key facts of the proof of Lemma 8.4 are the following. Firstly for every chain C and antichain N we have that $|C \cap N| \leq 1$. Secondly, if C is a chain, N an antichain, and $C \cap N = \{z\}$ for some $z \in V(D_M)$, then the following inequality holds:

$$\Pi(C) = \max_{v \in C} f_v \geq f_z \geq \min_{v \in N} f_v = val(N).$$

Furthermore, let $P = \{C_1, \ldots, C_p\}$ be an arbitrary chain partition of D_M and let $T = \{N_1, \ldots, N_{wdt(D_M)}\}$ be a tower of antichains in D_M. Then $|P| \geq wdt(D_M)$. We rename chains in the chain partition P as follows. Let $C'_1, \ldots, C'_p$ be the elements of P such that for all $i = 1, \ldots, wdt(D_M)$, chain C'_i intersects antichain N_i, that is, $|C'_i \cap N_i| = 1$ for all i. Then the following holds:

$$\Pi(P) = \sum_{i=1}^{p} \Pi(C_i) = \sum_{i=1}^{p} \Pi(C'_i) \geq \sum_{i=1}^{wdt(D_M)} \Pi(C'_i) \geq \sum_{i=1}^{wdt(D_M)} val(N_i) = val(T).$$

□

For a monotone function $f : V \to \mathbb{Z}_+$, the following generalization of Dilworth's Theorem holds.

Theorem 8.5 (Hujdurović et al. [18])**.** *Let M be a binary matrix and D_M its containment digraph. Let f be a monotone weight function on the vertices of D_M. Then D_M admits a chain partition $P = \{C_1, \dots C_{wdt(D_M)}\}$ and a tower of antichains $T = \{N_1, \dots N_{wdt(D_M)}\}$ such that $\Pi(P) = val(T)$. Such a pair (P, T) can be computed in time $\tilde{\mathcal{O}}(|V(D_M)|^{\omega+1})$.*

Proof idea. The proof is by induction on number of vertices of D_M. Let $n = |D_M|$. For $n = 1$ the statement, clearly, holds. For the induction step we consider a vertex $v \in V(D_M)$ such that v does not have any incoming arcs and consider a containment digraph D' defined as $D' = D_M - v$ (that is, D' is the subgraph of D_M induced by $V(D_M) \setminus \{v\}$). Note, that the digraph D' is the containment digraph of some binary matrix M'. We assume that the theorem statement holds for D', that is, there exist a chain partition $P = \{C_1, \dots, C_{wdt(D')}\}$ and a tower of antichains $T = \{N_1, \dots, N_{wdt(D')}\}$ of D' with $\Pi(P') = val(T')$.

Let us consider D_M. There are two possible cases. Firstly, assume that $wdt(D_M) > wdt(D')$. Consider the chain partition P of D_M obtained from P' by adding a chain consisting of vertex v, and a tower T of antichains obtained from T' by adding an antichain $N_{wdt(D_M)}$ such that $|N_{wdt(D_M)}| = wdt(D_M)$. Hence $\Pi(P) = \Pi(P') + f_v = val(T') + f_v = val(T)$. Secondly, we consider the case when $wdt(D_M) = wdt(D')$. We pick an antichain $T = \{t_1, \dots, t_{|T|}\}$ of D' such that $|T| = wdt(D_M)$. We define $\hat{T}$ to be the set of vertices of D_M such that there exists a path from the vertices of $\hat{T}$ to some vertex from T. Let us consider the directed acyclic graph $D_M - T$, which is the subgraph of D_M induced by $V(D_M) \setminus T$. Since $wdt(D_M - T) = |T| = wdt(D')$, by the induction hypothesis there exist a chain partition $P^T = \{C_1^T, \dots, C_{wdt(D')}^T\}$ with $\Pi(P^T) \leq \Pi(P') = val(T')$. Moreover, the subgraph $D_M[T \cup \hat{T}]$ has width equal to $|T|$, which by Dilworth's Theorem implies the existence of a chain partition $P^{\hat{T}} = \{C_1^{\hat{T}}, \dots, C_{wdt(D')}^{\hat{T}}\}$ covering all its vertices. Further we construct the chain partition of D_M as follows. We rename, if necessary, the elements of the chains P^T and $P^{\hat{T}}$ so that $C_i^T \cap T = \{t_i\}$ and $C_i^{\hat{T}} \cap T = \{t_i\}$ for every $i \in \{1, \dots |T|\}$. We define the chain as follows: $\tilde{C}_i = C_i^{\hat{T}} \cup C_i^T$. Let $\tilde{P} = \{\tilde{C}_1, \dots \tilde{C}_{wdt(D_M)}\}$. We get $\Pi(\tilde{P}) \leq val(T')$, and combining this with Lemma 8.4 we get that the chain partition $\tilde{P}$ is what we want. The proof is algorithmic and an optimal chain partition P and a tower of antichains T can be computed in the stated time. □

Remark 8.6. To see that Theorem 8.5 is a generalization of Dilworth's Theorem, we simply let the monotone weight function f be constantly equal to one. Then, the price of any chain equals its cardinality and the value of any tower of antichains equals to the width of the containment digraph D_M.

Lemma 8.4 and Theorem 8.5 imply the following result.

Corollary 8.7. *The Minimum Price Chain Partition problem can be optimally solved in time* $\tilde{\mathcal{O}}(|V(D_M)|^{\omega+1})$. *More precisely, a minimum price chain partition* P *of* D_M *can be found in time* $\tilde{\mathcal{O}}(|V(D_M)|^{\omega+1})$ *with the additional property that* $|P| = wdt(D_M)$.

For more details regarding the results introduced in this section see [18].

9 A polynomially computable upper bound on $\beta(M)$

Hujdurović et al. in [18] showed that Corollary 8.7 leads to a polynomially computable upper bound on an optimal value of uncovered pairs over all branchings B of D_M. The main idea is to restrict ourselves to the subfamily of linear branchings instead of the family of all branchings. Let M be a binary matrix and $D_M = (V, A)$ be its corresponding containment digraph. A *linear branching* of M is a subset of A with at most one outgoing and incoming arc from each vertex. Clearly, each linear branching corresponds to a disjoint union of directed paths in D_M, and vice versa. We also remark that such branchings correspond bijectively to the chain partitions of D_M.

Recall that we denote the set of uncovered pairs with respect to a branching B by $U(B)$. Given a binary matrix M, we denote by $\beta_\ell(M)$ the minimum number of elements in $U(B)$ over all linear branchings B of D_M. The upper bound for $\beta(M)$ is based on the Minimum Uncovering Linear Branching problem, an optimization problem defined formally as follows:

Minimum Uncovering Linear Branching (MULB):

Input: A binary matrix M.
Task: Compute $\beta_\ell(M)$.

As shown by the next theorem, an optimal linear branching can be computed in polynomial time.

Theorem 9.1 (Hujdurović et al. 2018 [18])**.** *The Minimum Uncovering Linear Branching problem is solvable in time $\mathcal{O}(|V(D_M)|^{\omega+1})$, where ω is any real number such that there exists an $\mathcal{O}(n^\omega)$ algorithm for multiplying any two $n \times n$ binary matrices.*

Proof idea. Let M be a binary matrix and $D_M = (V, A)$ its containment digraph. Let us define a monotone weight function $f : V \to \mathbb{Z}_+$ with $f(v) = |v|$. It is easy to see that f is indeed a monotone weight function, since by the definition of D_M, whenever $(u, v) \in A$ we have that $u \subset v$. It is also easy to see that for a linear branching B_ℓ and a

chain partition P corresponding to it, we have that $\Pi(P) = |U(B_\ell)|$. We noted above that the linear branchings correspond bijectively to the chain partitions, hence the MULB problem is a special case of the Minimum Price Chain Partition problem, with f defined as above. Thus, the claimed time complexity follows from Corollary 8.7. □

Lemma 9.2. *Let M be a binary matrix. Then*

$$\beta(M) \leq \beta_\ell(M).$$

We omit the proof of Lemma 9.2, since the above inequality follows immediately from the definitions.

Remark 9.3. Concerning the motivation, please note that the output of the MULB problem corresponds to the simplest possible reconstruction of the mutational history, since in this case we restrict ourselves to the space of rooted trees, where the root of the tree is the only node (vertex) allowed to have more than one non-leaf child (or more simply, only one outgoing edge which does not lead to a leaf).

10 Main results

Having in mind all the theory and known results from previous research on the Minimum Conflict-Free Row Split and Minimum Uncovering Branching problems, let us now move on to the main results of the thesis. Section 10 will be divided into four subsections. Firstly, we introduce a polynomially computable lower bound in terms of the maximum weight of an antichain in the corresponding containment digraph. Based on that, we identify some efficiently solvable cases of the MUB problem in Section 10.2.

Recall that in Section 7 we introduced Open Problem 1, asking whether there exist a constant c such that $\beta(M) \leq c \cdot W(M)$. In Section 10.3 we give an affirmative answer to this problem for specific families of instances introduced in [18]. Finally, in Section 10.4 we introduce a new type of construction of specific families for further analysis of Open Problem 1.

10.1 A lower bound on $W(M)$

We first introduce several notations and definitions. Then we introduce a polynomially computable lower bound on $W(M)$ (and hence also on $\beta(M)$) and justify the time complexity of computing the quantity. The lower bound introduced in this subsection is further used in Section 10.2 for identifying the efficiently solvable cases.

Let M be a binary matrix and D_M the corresponding containment digraph. Let us define the *weight* of a vertex $v \in V(D_M)$ simply as the cardinality of the vertex. (Recall that each vertex of D_M is a subset of the set of rows of M.) Notation: $w(v) = |v|$. Recall also that an antichain in D_M is a set of vertices inducing no arcs. We define the *weight* of an antichain X in D_M as follows: $w(X) = \sum_{v \in X} w(v)$. We denote $\alpha_w(M) := \max\{w(X) \mid X \text{ is an antichain in } D_M\}$.

Lemma 10.1. *For every binary matrix M, we have $\alpha_w(M) \leq W(M)$.*

Proof. Let M be an arbitrary binary matrix and D_M its containment digraph. Let $X = \{v_1, \ldots, v_q\}$ be an antichain of maximum weight in D_M. Recall that we assumed in Section 2 that a binary matrix M does not contain a row of all zeros and that all the columns are pairwise distinct. This implies that all vertices v_i for $i \in \{1, \ldots, q\}$ are non-empty subsets of the set of rows of M. By renaming the rows if necessary, we may thus assume that there exists a positive integer p such that $v_1 \cup \ldots \cup v_q = \{r_1, \ldots, r_p\}$.

Let us consider an arbitrary row r_i for $i \in \{1, \ldots, p\}$. Assume that r_i appears k_i times in an antichain X. Let D_{M,r_i} be the principal subgraph corresponding to r_i. Let $w_i = wdt(D_{M,r_i})$ for $i \in \{1, \ldots, p\}$. Then $k_i \leq w_i$ for all i, since the set $X \cap V(D_{M,r_i})$ is an antichain of D_{M,r_i} of cardinality k_i.

Hence, it follows that

$$\alpha_w(M) = \sum_{j=1}^{q} |v_j| = \sum_{i=1}^{p} k_i \leq \sum_{i=1}^{p} w_i = \sum_{i=1}^{p} wdt(D_{M,r_i}) \leq \sum_{r} wdt(D_{M,r}) = W(M).$$

□

To justify the polynomial time complexity for computing the lower bound introduced above, let us firstly introduce a definition. A *comparability graph* is a graph that admits a *transitive orientation*, that is, an assignment of directions to the edges of the graph such that the resulting directed graph is transitive.

The lower bound defined in terms of the maximum weight of an antichain in our case is indeed polynomially computable, since the problem of computing $\alpha_w(M)$ is a special case of the maximum weight independent set problem in the class of comparability graphs. The polynomial-time solvability of this latter problem follows from two facts: 1) that the class of comparability graphs is a subclass of the class of perfect graphs, and 2) that the maximum weight independent set problem is polynomially solvable in the class of perfect graphs, as shown by Grötschel et al. (see [14]).

The fact that every comparability graph is a perfect graph can be seen as follows. Suppose that G is a comparability graph. Let D be a transitive orientation of G. Then, a set $X \subseteq V(G)$ is an independent set in G if and only if X is an antichain in D. Furthermore, X is a clique in G if and only if X is a chain in D. Thus to show that G is perfect it suffices to show that the maximum size of a chain in D equals the minimum number of antichains partitioning $V(D)$. This equality was shown by Mirsky [24].

As the next example shows, the inequality from 10.1 can be strict.

Example 10.2. Let M be a binary matrix corresponding to the containment digraph D_M given on Fig. 20.

We have that $wdt(D_{M,r_2}) = wdt(D_{M,r_3}) = wdt(D_{M,r_4}) = 1$ and $wdt(D_{M,r_1}) = 2$, hence $W(M) = 5$. There are four non-empty antichains with the following weights: $w(\{v_1\}) = 2$, $w(\{v_2\}) = 2$, $w(\{v_1, v_2\}) = 4$ and $w(\{v_3\}) = 4$. Hence $\alpha_w(M) = 4$.

Since the inequality from Lemma 10.1 can be strict, it would be interesting to answer the following question.

Open problem 2. *Is there a function f such that for every binary matrix M, we have $W(M) \leq f(\alpha_w(M))$?*

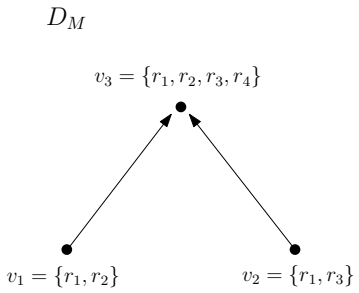

Figure 20: A containment digraph D_M, introduced as an example attaining strict inequality in Lemma 10.1.

Lemmas 7.5, 9.2, and 10.1 yield the following chain of inequalities valid for every binary matrix M.

Corollary 10.3. *For every binary matrix M, we have*

$$\alpha_w(M) \leq W(M) \leq \beta(M) \leq \beta_\ell(M).$$

A similar question as in Open Problem 2 could also be asked for other pairs of quantities involved in the above chain of inequalities. For example:

Open problem 3. *Is there a function g such that for every binary matrix M, we have $\beta(M) \leq g(W(M))$?*

Open problem 4. *Is there a function h such that for every binary matrix M, we have $\beta_\ell(M) \leq h(\beta(M))$?*

In Section 10.2 the chain of inequalities from Corollary 10.3 will be used to prove optimality of algorithms for computing $\beta(M)$ for particular inputs.

10.2 New efficiently solvable cases

We move on to identifying some polynomially solvable cases. For understanding the results introduced in this subsection we define several notions.

Let M be a binary matrix and D_M its containment digraph. We say that two vertices $v_1, v_2 \in V(D_M)$ are *comparable* if either $v_1 \subset v_2$ or $v_2 \subset v_1$, otherwise we say that v_1 and v_2 are *incomparable*. Furthermore, for the poset $P_M = (V(D_M), \subset)$ corresponding to the containment digraph D_M, an element $m \in P_M$ is *maximal* if there is no element of P_M properly containing m. (Formally, there is no $s \in P_M$ such that $m \subset s$.)

Since the family of maximal elements of P_M forms an antichain, the value of $wdt(M)$ is bounded from below by the number of maximal elements of P_M. As the next result

shows, the case when this inequality is satisfied with equality has interesting consequences for our problems of interest.

Theorem 10.4. *Let M be a binary matrix such that the corresponding poset P_M has exactly n maximal elements, where $n = wdt(M)$. Then $\alpha_w(M) = \beta_\ell(M)$.*

Proof. Let D_M be the containment digraph corresponding to M. Since $wdt(M) = n$, by Dilworth's Theorem there exists a chain partition of D_M into n chains. Let $P = \{C_1, \ldots, C_n\}$ be such a chain partition. Label the vertices of C_i by $\{v_1^i, \ldots, v_{k_i}^i\}$, where k_i is the number of vertices contained in the chain and $v_1^i \subset \ldots \subset v_{k_i}^i$, as in Fig. 21.

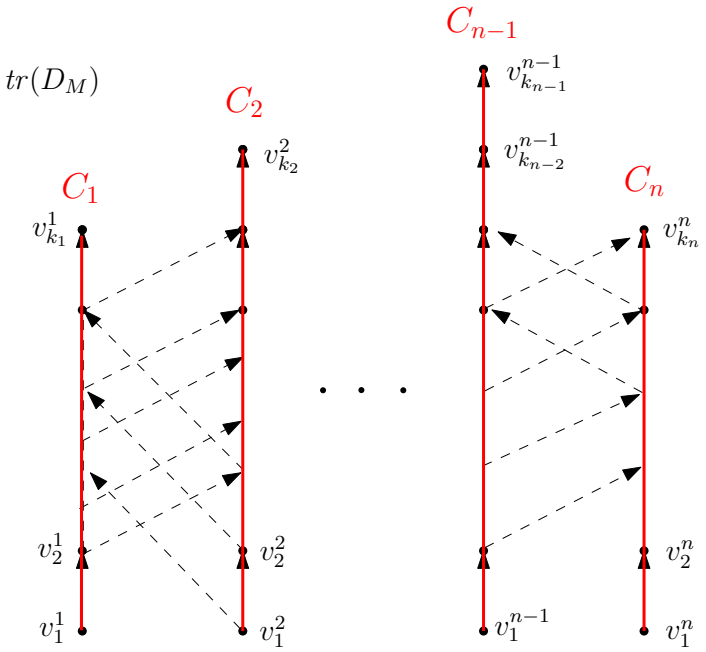

Figure 21: A transitive reduction $tr(D_M)$ of a containment digraph D_M, corresponding to a binary matrix M with $wdt(M) = n$, partitioned into n chains.

Since, by assumption, there are n maximal elements in P_M, and no two maximal elements can belong to a common chain, the maximal elements of P_M are precisely the vertices $v_{k_1}^1, \ldots, v_{k_n}^n$. In particular, this implies that these vertices are pairwise incomparable; in fact, it is easy to see that they form an antichain of maximal weight. Denote the weights of vertices $v_1^i, \ldots, v_{k_i}^i$ in C_i by $w_1^i, \ldots, w_{k_i}^i$, respectively. Then, $\alpha_w(M) = w_{k_1}^1 + \ldots + w_{k_n}^n$.

Let us now consider the number of uncovered pairs in the linear branching B of D_M corresponding to P. We will consider the number of uncovered pairs in each chain separately. For each $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, k_i\}$, there are exactly $|v_j^i \setminus v_{j-1}^i| = w_j^i - w_{j-1}^i$ uncovered pairs with second coordinate equal to vertex $v_j^i \in C_i$ (where $w_0^i = 0$). Hence, the total number of uncovered pairs having second coordinate from

chain C_i equals $\sum_{j=1}^{k_i}(w^i_j - w^i_{j-1}) = w^i_{k_i}$. This holds for every chain in P, and thus, since P is a chain partition, we infer that $|U(B_\ell)| = w^1_{k_1} + \ldots w^n_{k_n} = \alpha_w(M)$.

The claimed equality holds due to the following chain of inequalities:

$$\beta_\ell(M) \leq |U(B_\ell)| = w^1_{k_1} + \ldots w^n_{k_n} = \alpha_w(M) \leq \beta_\ell(M)\,,$$

where the first inequality holds by definition of $\beta_\ell(M)$ and the last one by Corollary 10.3. □

Theorem 10.4 shows that, whenever the width of a binary matrix M equals the number of maximal elements in the corresponding poset P_M, all the four quantities involved in the chain of inequalities $\alpha_w(M) \leq W(M) \leq \beta(M) \leq \beta_\ell(M)$ given by Corollary 10.3 are equal. In particular, since all the quantities in the above chain of inequalities except $\beta(M)$ are known to be polynomial-time computable on general input instances, this implies the existence of a polynomial-time algorithm for computing $\beta(M)$ for binary matrices M for which the antichain of maximal elements in the corresponding poset is a maximum antichain. Furthermore, it follows from the proof of Theorem 10.4 that for such matrices, the number of uncovered pairs over all linear branchings corresponding to optimal chain partitions is constantly equal to $\alpha_w(M)$.

Note that if M is a binary matrix of width 1, then M is conflict-free, the condition from Theorem 10.4 is satisfied, and the quantities $\alpha_w(M) = W(M) = \beta(M) = \beta_\ell(M)$ are all equal to the number of rows of M. The next result shows that the case of width 2 is also well understood.

Theorem 10.5. *For every binary matrix M with $wdt(M) = 2$, we have $\beta(M) = W(M)$.*

Proof. The proof is by induction on $\Gamma(M) = |V(D_M)|$, where D_M is the corresponding containment digraph. *Base case:* $\Gamma(M) = 2$. In this case, P_M consists of two incomparable vertices u and v. Then

$$W(M) = \sum_r wdt(D_{M,r}) = 2|u \cap v| + (|u| - |u \cap v|) + (|v| - |u \cap v|) = |u| + |v| = \beta(M)\,.$$

Induction hypothesis: Assume that the theorem holds for all binary matrices M' such that $wdt(M') = 2$ and $\Gamma(M') = n$ for some $n \geq 2$.

Induction step: Let M be a binary matrix such that $wdt(M) = 2$ and $\Gamma(M) = n+1$. If there are two maximal elements in P_M, we are done by applying Theorem 10.4. Assume that there is one maximal element in P_M. As for the inductive step, consider the column $c \in C_M$ corresponding to the maximal element $m \in P_M$. Let M' be the binary matrix obtained from M by first deleting from it column c and then removing

any rows containing only zeros. Then $D_{M'} = D_M - m$ and hence $wdt(M') = 2$. By the induction hypothesis, we obtain $\beta(M') = W(M')$. Since $W(M) \leq \beta(M)$ holds for all M by Corollary 7.5, it is sufficient to show that $\beta(M) \leq W(M)$. Since we did not make any assumption about the number of maximal elements in $P_{M'}$, this poset may have either one or two maximal elements. Let us examine the two cases individually.

Assume first that there are two maximal elements in $P_{M'}$, say m_1, m_2, and let B' be an optimal branching of M'. Let B be a branching of D_M obtained from B' by including the two edges (m_1, m), (m_2, m). See Fig. 22.

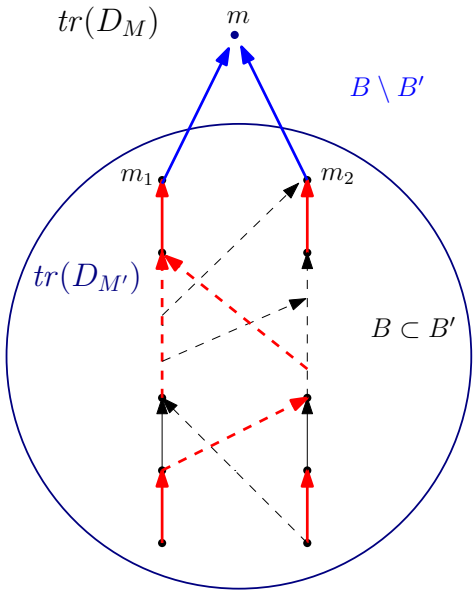

Figure 22: A transitive reduction $tr(D'_M)$ of a containment digraph $D_{M'}$ corresponding to the binary matrix M' having two maximal elements $m_1, m_2 \in P_{M'}$, an optimal branching B' of $D_{M'}$ (shown in red), a transitive reduction $tr(D_M)$ of a containment digraph D_M corresponding to a binary matrix M obtained by adding a maximal element m to $P_{M'}$ and a branching B of D_M obtained from the branching B' by adding the edges in blue.

Let $b = |U(B)|$ and let $s = |m \setminus (m_1 \cup m_2)|$. Then, $b = \beta(M') + s$. Further, let us consider the value of $W(M) = \sum_r wdt(D_{M,r})$. For all $r \in m$ such that $r \in m_1 \cup m_2$ we have that $wdt(D_{M',r}) = wdt(D_{M,r})$ for all $r \in D_{M'}$, since m is a maximal element and it is comparable with all $x \in V(D_M) \setminus \{m\}$. For all $r \in m$ such that $r \notin m_1 \cup m_2$, we have $wdt(D_{M,r}) = 1$. Hence $\sum_r wdt(D_{M,r}) = \sum_r wdt(D_{M',r}) + s$. It follows that

$$\begin{aligned} \beta(M) &\leq |U(B)| = |U(B')| + s = \beta(M') + s = W(M') + s \\ &= \sum_r wdt(D_{M',r}) + s = \sum_r wdt(D_{M,r}) = W(M)\,. \end{aligned}$$

Together with the inequality $W(M) \leq \beta(M)$, which holds for all binary matrices (see Corollary 10.3), we obtain the desired equality $\beta(M) = W(M)$.

Finally, assume that there is only one maximal element in M', say m'. Let B' be an optimal branching of M'. Let B be a branching of D_M obtained from B' by including the edge (m', m). See Fig. 23.

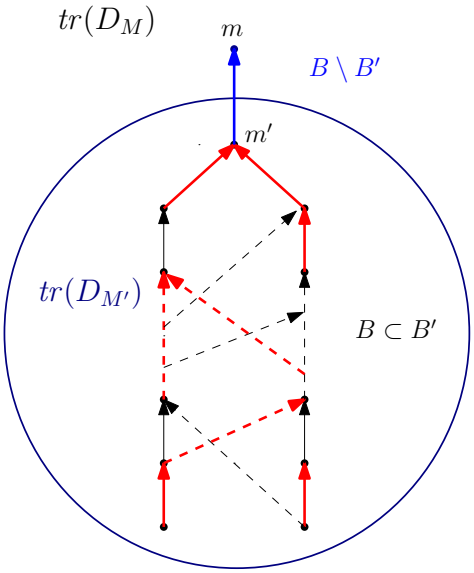

Figure 23: A transitive reduction $tr(D_M)$ of a containment digraph $D_{M'}$ corresponding to the binary matrix M' having only one maximal element $m' \in P_{M'}$, an optimal branching B' of $D_{M'}$ (shown in red), a transitive reduction $tr(D_M)$ of a containment D_M corresponding to a binary matrix M obtained by adding a maximal element m to $P_{M'}$ and a branching B of D_M obtained from B' by adding the edge shown in blue.

Let b denote the number of uncovered pairs with respect to branching B and let $s = |m \setminus m'|$. Then $b = \beta(M') + s$ and $W(M) = \sum_r wdt(D_{M,r}) = \sum_r wdt(D_{M',r}) + s = W(M') + s$. To conclude, we obtain

$$\beta(M) \leq b = \beta(M') + s = W(M') + s = W(M) \leq \beta(M)\,.$$

Hence, $\beta(M) = W(M)$. □

Theorem 10.5 and the fact that $W(M)$ is polynomially computable (see Section 7) imply the following.

Corollary 10.6. *Let $M \in \{0,1\}^{m \times n}$ such that $wdt(M) = 2$. Then $\beta(M)$ is polynomial-time computable.*

Remark 10.7. In fact, the proof of Theorem 10.5 is constructive and leads to a polynomial-time algorithm for computing an optimal branching for a given matrix of width 2.

Note that the result of Theorem 10.5 is a sense best possible. There exist binary matrices M of width 3 such that $W(M) < \beta(M)$ (see Example 10.14).

10.3 Improving bounds for specific families of instances

In this subsection we give an affirmative answer to Open Problem 1 for two specific families of instances introduced by Hujdurović et al. in [18]. Let us, firstly, introduce the following definitions.

Definition 10.8. A *hypergraph* is a pair $H = (V, E)$, where $V = V(H)$ is a set and $E = E(H)$ is a subset of the power set $\mathcal{P}(V)$. The elements of $V(H)$ are the *vertices* of H and elements of $E(H)$ are the *hyperedges* of H.

Definition 10.9. A hypergraph is *Sperner* if no hyperedge contains another one.

We only consider hypergraphs H in which every vertex is contained in a hyperedge.

Definition 10.10. The *column hypergraph* H_M of a binary matrix M is the hypergraph having the rows of M as vertices and the support sets of the columns of M as hyperedges. Formally, H_M has vertex set $V(H_M) = R_M$ and hyperedge set $E(H_M) = \{\text{supp}_M(c) \mid c \in C_M\}$.

Remark 10.11. Note that the set of hyperedges of the column hypergraph of M equals the vertex set of the containment digraph D_M.

We now generalize a construction from [18], used to prove APX-hardness of the MUB (and consequently MCRS) problem, from graphs to Sperner hypergraphs. Let H be a Sperner hypergraph. Let us now construct the following hypergraph. Let x, y be two new vertices not in $V(H)$. Let H' be the hypergraph with vertex set $V(H') = V(H) \cup \{x, y\}$ and hyperedge set

$$E(H') = \{V(H) \cup \{x\}\} \cup \{e \cup \{x\} \mid e \in E(H)\} \cup \{e \cup \{y\} \mid e \in E(H)\} \cup \{e \cup \{x, y\} \mid e \in E(H)\}.$$

The hypergraph H' is the containment hypergraph of a binary matrix derived from hypergraph H. See Fig. 24 for an example construction, representing the containment digraph D_M of the binary matrix derived from the complete graph K_3. A special case of the construction above was introduced by Hujdurović et al. in [18] in the proof of Theorem 6.2. In fact, the construction was initially introduced as a main tool for establishing an L-reduction from the vertex cover problem in cubic graphs to the MUB problem, hence this type of construction was performed for cubic graphs G (which would correspond to the hypergraphs with vertex set $E(G)$ and having all sets of edges incident with a fixed vertex of G as hyperedges). See Section 2 for the definition of cubic graphs.

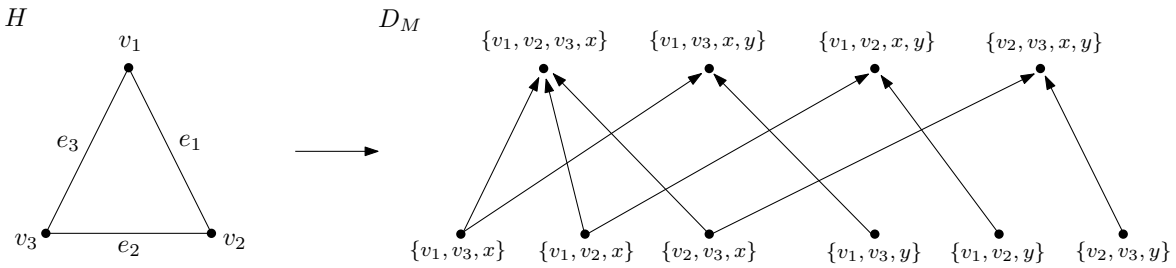

Figure 24: An example construction of the hypergraph H' from Theorem 10.12: the column hypergraph of a binary matrix M derived from the complete graph K_3.

Theorem 10.12. *Let H be a Sperner hypergraph, let H' be the hypergraph with vertex set $V(H') = V(H) \cup \{x, y\}$ and hyperedge set*

$$E(H') = \{V(H) \cup \{x\}\} \cup \{e \cup \{x\} \mid e \in E(H)\} \cup \{e \cup \{y\} \mid e \in E(H)\} \\ \cup \{e \cup \{x, y\} \mid e \in E(H)\},$$

and let M be a binary matrix such that its column hypergraph is H'. Then $\beta(M) \leq 2\,W(M)$.

Proof. Let H be a Sperner hypergraph and let $n = |V(H)|$ and let $m = |E(H)|$. Denote the vertices and hyperedges of H by $v_1, \ldots, v_n$ and $e_1, \ldots, e_m$, respectively. For simplicity let us divide the hyperedges of H' into 4 types:

1. $V(H) \cup \{x\} = \{v_1, \ldots, v_n, x\}$,
2. $\{e \cup \{x\} \mid e \in E(H)\} = \{e_1 \cup \{x\}, \ldots, e_m \cup \{x\}\}$,
3. $\{e \cup \{y\} \mid e \in E(H)\} = \{e_1 \cup \{y\}, \ldots, e_m \cup \{y\}\}$,
4. $\{e \cup \{x, y\} \mid e \in E(H)\} = \{e_1 \cup \{x, y\}, \ldots, e_m \cup \{x, y\}\}$.

As mentioned earlier in this section, the set of hyperedges of H' is equal to the vertex set of the containment digraph D_M. Hence we can refer to the hyperedges of the hypergraph H' as the vertices of containment digraph D_M. Vertex $V(H) \cup \{x\}$ has m incoming edges of the form $(e_i \cup \{x\}, V(H) \cup \{x\})$, for all $i \in \{1, \ldots, m\}$. For all $i \in \{1, \ldots, m\}$, vertex $e_i \cup \{x, y\}$ has two incoming edges, namely $(e_i \cup \{x\}, e_i \cup \{x, y\})$ and $(e_i \cup \{y\}, e_i \cup \{x, y\})$.

Firstly, let us analyze the sum $\sum_r wdt(D_{M,r})$. Let us consider an antichain N formed by all vertices of type 2 and 3. Each v_i for $i \in \{1, \ldots, n\}$ will appear in vertices of type 2 as many times as its *degree* $d_H(v_i)$, defined as the number of hyperedges of H containing v_i. Similar argument applies for vertices of type 3. Hence, in total each v_i will appear $2d_H(v_i)$ times in an antichain N. Let us consider the width of the principal

subgraph D_{M,v_i} corresponding to v_i. From the argument above, we conclude that $wdt(D_{M,v_i}) \geq 2d_H(v_i)$ for all $i \in \{1,\ldots,n\}$. Further, let us consider $D_{M,x}$ and $D_{M,y}$. Character x appears in all vertices of type 2 or 4, and in vertex $V(H) \cup \{x\}$. Hence, it is easy to see that an antichain Z in $D_{M,x}$ formed by the vertices of type 4 and vertex $V(H) \cup \{x\}$ form an antichain with $|Z| = m+1$, which implies $wdt(D_{M,x}) \geq m+1$. Similarly, $wdt(D_{M,y}) \geq m$, since each vertex of the form $e_i \cup \{x,y\}$ will have one incoming edge of the form $e_i \cup \{y\}$ for $i \in \{1,\ldots,m\}$ and we get two antichains of size m. Hence, $\sum_r wdt(D_{M,r}) \geq \sum_i 2d_H(v_i) + 2m + 1$.

Next, let us examine the number of uncovered pairs with respect to the empty branching. We claim that $|U(\emptyset)| \leq 2\left(\sum_i 2d_H(v_i) + 2m + 1\right)$. Let us count the number of times a character v_i, for $i \in \{1,\ldots,n\}$, appears as the first coordinate of an uncovered pair with respect to the empty branching, or, equivalently, the number of times v_i appears as an element of a hyperedge of H'. Character v_i will appear in the vertices of D_M of type 2, 3, 4 as many times as the degree of v_i in H, as well as in vertex $V(H) \cup \{x\}$. Secondly, character x appears in all the vertices of type 2 or 4, and in addition in vertex $V(H) \cup \{x\}$. Finally, character y will appear in all vertices of type 3 or 4. Hence in total we have

$$|U(\emptyset)| = \sum_{i=1}^{n}(3d_H(v_i)+1) + 4m + 1\,.$$

Hence to show that $|U(\emptyset)| \leq 2\left(2\sum_{i=1}^{n} d_H(v_i) + 2m + 1\right) \leq 2 \cdot W(M)$ it suffices to show the following inequality:

$$2\left(2\sum_{i=1}^{n} d_H(v_i) + 2m + 1\right) - \left(\sum_{i=1}^{n}(3d_H(v_i)+1) + 4m + 1\right) \geq 0\,.$$

To justify this inequality, note that

$$4\sum_{i=1}^{n} d_H(v_i) + 4m + 2 - 3\sum_{i=1}^{n} d_H(v_i) - n - 4m - 1 = \sum_{i=1}^{n} d_H(v_i) - n + 1 \geq 0$$

since by the assumption $d_H(v_i) \geq 1$ for all $i \in \{1,\ldots,n\}$. We conclude that $\beta(M) \leq |U(\emptyset)| \leq 2W(M)$. □

Now we consider another construction that generalizes a different construction from [18], used to prove APX-hardness of the MIB (and consequently MCDRS) problem, from graphs to Sperner hypergraphs. Let H be a Sperner hypergraph. Let us now construct a hypergraph, $H' = (V(H), \{\{v\} \mid v \in V(H)\} \cup E(H))$. Again, H' can be represented with a binary matrix M. See Fig. 25 for an example construction, representing the containment digraph D_M of the binary matrix derived from the complete graph K_3. The construction was initially introduced as a main tool for establishing

an L-reduction from the vertex cover in cubic graphs to the MIB problem, hence this type of construction was performed for cubic graphs G. (Again, the corresponding hypergraphs would have vertex set $E(G)$ and all sets of edges of G incident with a fixed vertex as hyperedges.)

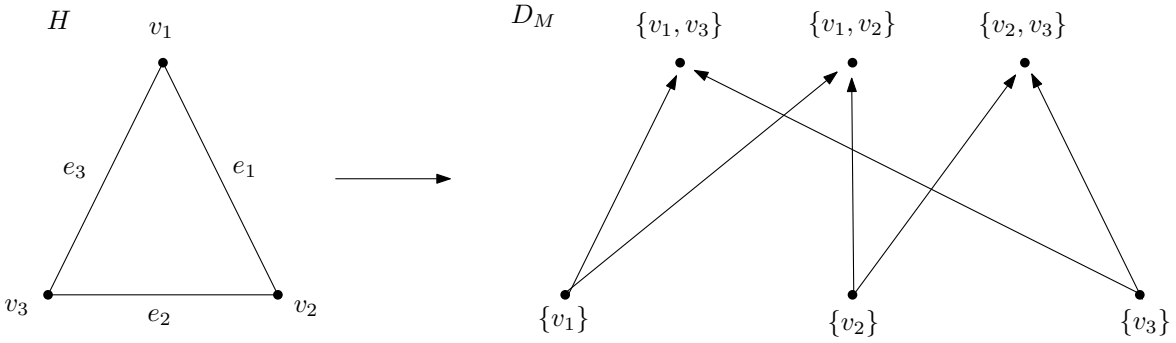

Figure 25: An example construction of the hypergraph H' from Theorem 10.13: the column hypergraph of a binary matrix M derived from the complete graph K_3.

Theorem 10.13. *Let H be a Sperner hypergraph, let H' be the hypergraph with vertex set $V(H') = V(H)$ and hyperedge set*

$$E(H') = \{\{v\} \mid v \in V(H)\} \cup E(H)\,,$$

and let M be a binary matrix such that its column hypergraph is H'. Then $\beta(M) \leq 2\,W(M)$.

Proof. Let H be a Sperner hypergraph and let $n = |V(H)|$ and $m = |E(H)|$. Denote the vertices and hyperedges of H by $v_1, \ldots, v_n$ and $e_1, \ldots, e_m$, respectively. For simplicity let us divide the hyperedges of H' into two types:

1. $\{\{v\} : v \in V(H)\}\,,$
2. $\{e \mid e \in E(H)\} = \{e_1, \ldots, e_m\}\,.$

Again, we can refer to the hyperedges of H' as the vertices of containment digraph D_M. Let us analyze the edges of the containment digraph D_M. There are directed edges from vertices $\{v_i\}$ to vertices of the form e_j that contain v_i for all $i \in \{1, \ldots, n\}$ and $j \in \{1, \ldots, m\}$, and there are no other edges.

Let us consider the value of $\sum_r wdt(D_{M,r})$. Fix $i \in \{1, \ldots, \}$ and let $k = d_H(v_i)$ be the number of hyperedges of H containing v_i. Then, D_{M,v_i} contains k vertices of type 2 and vertex $\{v_i\}$. There are k directed edges of the form $(\{v_i\}, e_j)$ for all $j \in \{1, \ldots, m\}$ such that hyperedge e_j contain v_i. Hence the maximum size of an antichain in D_{M,v_i} is at least k. It follows that $\sum_r wdt(D_{M,r}) \geq \sum_{i=1}^{n} d_H(v_i)$. Let us count the number of

times a character v_i for $i \in \{1, \ldots, n\}$, appears as the first coordinate of an uncovered pair with respect to the empty branching, or equivalently, the number of times v_i appears as an element of a hyperedge of H'. For each $i \in \{1, \ldots, n\}$, character v_i appears $d_H(v_i)+1$ many times in the vertices of D_M. Hence, $|U(\emptyset)| = \sum_{i=1}^{n}(d_H(v_i)+1)$.

To show that $|U(\emptyset)| \leq 2 \cdot (\sum_{i=1}^{n} d_H(v_i)) \leq 2 \cdot W(M)$ it sufficies to show that

$$2\left(\sum_{i=1}^{n} d_H(v_i)\right) - \sum_{i=1}^{n}(d_H(v_i)+1) = \left(\sum_{i=1}^{n} d_H(v_i)\right) - n \geq 0\,.$$

Since, $d_H(v_i) \geq 1$ for all i, the inequality above holds. We conclude that $\beta(M) \leq |U(\emptyset)| \leq 2W(M)$, as claimed. □

10.4 Further improvements

In this subsection we present some improvements in the direction of Open Question 1 introduced in Section 7, asking whether there exists a constant c such that $\beta(M) \leq cW(M)$ for all binary matrices M. More specifically, we show that if such a constant c exists, then $c \geq \frac{7}{6}$. In symbols, we show that $\sup_M \left\{\frac{\beta(M)}{W(M)}\right\} \geq \frac{7}{6}$.

Let us introduce the following construction of particular containment digraphs. For positive integers n and k with $n \geq 2$, we define a two-parametric family of containment digraphs $MD_{n,k} = (V, A)$ in the following way. Let $A_1, \ldots, A_n$ be pairwise disjoint sets such that $|A_1| = \ldots = |A_n| = k$. Let $Z = A_1 \cup \ldots \cup A_n$ and let $x_1, \ldots, x_{n-1}$ be pairwise distinct elements such that $x_i \notin Z$ for $i = 1, \ldots, n-1$. The vertex set of $MD_{n,k}$ is

$$V = \{Z \cup \{x_i\} \mid i \in \{1, \ldots, n-1\}\} \cup \{Z \setminus A_i \mid i \in \{1, \ldots, n\}\}\,,$$

and there is an arc $(u, v) \in A$ if and only if $u \subset v$. Furthermore, let $M_{n,k}$ denote any binary matrix such that its containment digraph is $MD_{n,k}$.

Example 10.14. Let $M = M_{3,k}$ be a binary matrix such that its containment digraph is $MD_{3,k}$, graphically represented on Fig. 26.

Let us compute $W(M)$. For all $i \in \{1, 2, 3\}$ and all $r \in A_i$, we have $wdt(D_{M,r}) = 2$, and for $j \in \{1, 2\}$, we have $wdt(D_{M,x_j}) = 1$. Hence $W(M) = 2 \cdot 3k + 2 = 6k + 2$.

Next, let us compute $\beta(M)$, that is, the minimal number of elements in $U(B)$ over all branchings B of $MD_{3,k}$. By symmetry, it suffices to analyze two edge-maximal branchings B_1 and B_2, depicted in Fig. 27 and 28, respectively.

We have $|U(B_1)| = 9k + 2$ and $|U(B_2)| = 7k + 2$. We conclude that

$$\beta(M) = \min\{|U(B_1)|, |U(B_2)|\} = \min\{9k + 2, 7k + 2\} = 7k + 2\,.$$

Since $W(M) = 6k + 2$ and

$$\lim_{k \to \infty} \frac{7k + 2}{6k + 2} = \frac{7}{6}\,,$$

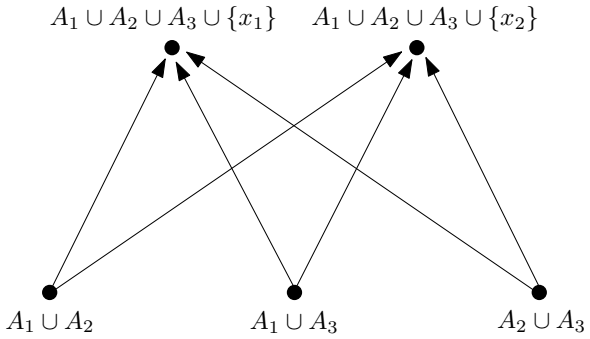

Figure 26: A graphical representation of containment digraph $MD_{3,k}$.

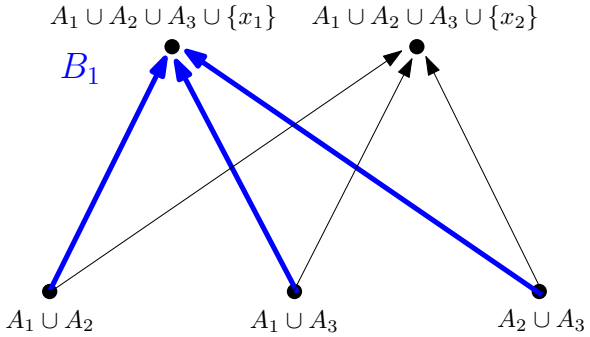

Figure 27: A graphical representation of $MD_{3,k}$ and branching B_1.

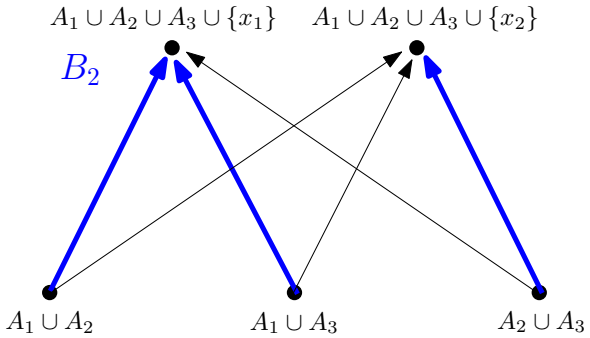

Figure 28: A graphical representation of $MD_{3,k}$ and branching B_2.

the ratio of $\beta(M)$ vs. $W(M)$ can be as close to $7/6$ as desired.

The above example shows that $\sup_M \left\{ \frac{\beta(M)}{W(M)} \right\} \geq \frac{7}{6}$.

Let us now consider the general case. Let $N = \{Z \setminus A_i \mid i \in \{1, \ldots, n\}\}$. Then N is an antichain of $MD_{n,k}$ of maximum size. Note that $|N| = n$ and hence $wdt(MD_{n,k}) = n$. By Dilworth's Theorem there exist a chain partition P of $MD_{n,k}$ consisting of n chains. Since $|V(MD_{n,k})| = 2n-1$, every chain partition $P = \{C_1, \ldots, C_n\}$ of size n consists of $n-1$ pairwise disjoint chains corresponding to edges of $MD_{n,k}$ and a chain consisting of a single vertex v_i of the form $Z \setminus A_i$ for some $i \in \{1, \ldots, n\}$. Let us say that a branching of $MD_{n,k}$ is *canonical* if it consists of the edges of such a chain partition P and an edge of the form $(v_i, Z \cup \{x_j\})$ for some $j \in \{1, \ldots, n-1\}$. Note that branching

B_2, depicted in Fig. 28, is a canonical branching in $MD_{3,k}$, while branching B_1 depicted in Fig. 27, is not.

Lemma 10.15. *Every optimal branching B of $MD_{n,k}$ is a canonical branching and satisfies $|U(B)| = k(n^2-2)+n-1$. Consequently, $\beta(M_{n,k}) = k(n^2-2)+n-1$.*

Before we move to the proof of the lemma, let us introduce the following notation. We denote by $d_B^-(v)$ the in-degree of a vertex v with respect to a branching B.

Proof. Firstly, for simplicity, let us divide the vertices of $MD_{n,k}$ into two types:

1. $\{Z \cup \{x_i\} \mid i \in \{1,\dots,n-1\}\}$,
2. $\{Z \setminus A_i \mid i \in \{1,\dots,n\}\}$.

For $i \in \{1,\dots,n-1\}$ let $v_i = Z \cup \{x_i\}$ and for $i \in \{1,\dots,n\}$ let $v_i' = Z \setminus A_i$.

Note that pairs (x_i, v_i) for $i \in \{1,\dots,n-1\}$ are uncovered with respect to any branching B of $MD_{n,k}$. Furthermore, all the pairs with second coordinate equal to a vertex v_i' of type 2 are uncovered, since $d^-_{MD_{n,k}}(v_i') = 0$ for all i. Hence, there are n vertices of type 2 each giving rise to $k(n-1)$ uncovered pairs and $|U(B^{opt})| \geq k(n^2-n)+(n-1)$.

Suppose that B is a canonical branching consisting of the edges of a chain partition $P = \{C_1,\dots,C_n\}$ and an edge of the form $(v_i, Z \cup \{x_j\})$ for some $i \in \{1,\dots,n\}$ and $j \in \{1,\dots,n-1\}$. Then all pairs with second coordinate equal to $Z \cup \{x_j\}$ are covered with respect to B, while for all $j' \neq j$, there exists some $i' \in \{1,\dots,n\}$ such that each pair of the form $(x, Z \cup \{x_{j'}\})$ with $x \in A_{i'}$ is uncovered with respect to B. There are no other uncovered pairs with respect to B, we obtain $|U(B)| = k(n^2-n)+(n-1)+(n-2)k = k(n^2-2)+n-1$.

Suppose for a contradiction that there exists an optimal branching B^{opt} that is not canonical. Since B^{opt} is optimal, we have $|U(B^{opt})| \leq k(n^2-2)+n-1$. Let us analyze this optimal branching. Firstly, since B^{opt} is a branching, there is at most one outgoing arc from each vertex. We can assume w.l.o.g. that B^{opt} is a maximal branching with respect to its edge set, that is, that $|E(B^{opt})| = n$.

If $d^-_{B^{opt}}(v_i) = 0$ for some i, then there are $kn+1$ uncovered pairs in v_i with respect to branching B^{opt} and $|U(B^{opt})| \geq kn^2+n-1 > k(n^2-2)+n-1$. Hence, we conclude that $d^-_{B^{opt}}(v_i) \geq 1$ for all i.

Since, as mentioned at the beginning of the proof, $|E(B^{opt})| = n$, we have that $d^-_{B^{opt}}(v_1)+\ldots+d^-_{B^{opt}}(v_{n-1}) = n$ with $d^-_{B^{opt}}(v_i) \geq 1$ for all $i \in \{1,\dots n-1\}$. This implies that there exists some $j \in \{1,\dots n-1\}$ such that $d^-_{B^{opt}}(v_i) = 1$ for $i \in \{1,\dots,n-1\}\setminus\{j\}$ and $d^-_{B^{opt}}(v_j) = 2$. Hence, B^{opt} is a canonical branching, a contradiction. □

Theorem 10.16. *We have that*

$$\sup\left\{ \frac{\beta(M_{n,k})}{W(M_{n,k})} \;\middle|\; n \geq 2, k \geq 0 \right\} = \frac{7}{6}.$$

Proof. By Lemma 10.15, $\beta(M_{n,k}) = k(n^2-2)+n-1$ for any positive integers n and k with $n \geq 2$. Let us now consider the value of $W(MD_{n,k})$. Firstly, note that $wdt(D_{M_{n,k},x_i}) = 1$ for all $i \in \{1,\ldots,n-1\}$. Furthermore, since the sets $A_1,\ldots,A_n$ are pairwise disjoint and of size k, we can write $Z = \{r_1,\ldots,r_p\}$ where $p = nk$. For each such element r_j, we have $wdt(D_{M_{n,k},r_j}) = n-1$. Hence, $W(M_{n,k}) = nk(n-1)+n-1 = (nk+1)(n-1)$.

Firstly, note that the fraction can be equivalently written as follows.

$$\begin{aligned}
\frac{\beta(M_{n,k})}{W(M_{n,k})} &= \frac{k(n^2-2)+n-1}{nk(n-1)+n-1} = \frac{n^2k-2k+n-1}{nk(n-1)+n-1} = \frac{n^2k-nk+nk-k-k+n-1}{nk(n-1)+n-1} \\
&= \frac{nk(n-1)+k(n-1)-k+n-1}{nk(n-1)+n-1} = \frac{nk+k+1-\frac{k}{n-1}}{nk+1} = \frac{n+1+\frac{1}{k}+\frac{1}{n-1}}{n+\frac{1}{k}} \\
&= 1+\frac{1-\frac{1}{n-1}}{n+\frac{1}{k}}\,.
\end{aligned}$$

Clearly, for every fixed $n \geq 2$ the above ratio is a strictly increasing function of k and

$$\lim_{k\to\infty}\left(1+\frac{1-\frac{1}{n-1}}{n+\frac{1}{k}}\right) = 1+\frac{1-\frac{1}{n-1}}{n}\,,$$

as $\lim_{k\to\infty}\frac{1}{k} = 0$. We now express the above ratio equivalently as follows:

$$1+\frac{1-\frac{1}{n-1}}{n} = 1+\frac{n-2}{n-1}\cdot\frac{1}{n} = 1+\frac{n-2}{n(n-1)} = 1+\frac{2(n-1)-n}{n(n-1)} = 1+\frac{2}{n}-\frac{1}{n-1}\,.$$

Let us analyze the function $f:[2,\infty)\to\mathbb{R}$ defined by the rule $f(x) = 1+\frac{2}{x}-\frac{1}{x-1}$ for all $x \in [2,\infty)$. Note that

$$f'(x) = -\frac{x^2-4x+2}{(x-1)^2x^2} = 0$$

at $x^* = 2+\sqrt{2} \approx 3.414$. It is easy to check that f is a strictly increasing function on the interval $[2,x^*)$ and then strictly decreasing limiting to 1 as $x \to \infty$. Hence, function f has a unique maximum attained at x^*, and if we consider the values $f(n)$ for integers $n \geq 2$, then maximum can be attained at either $n = 3$ or $n = 4$. Let us consider the two cases.

1. For $n = 3$ we obtain the following value:

$$f(3) = 1+\frac{2}{3}-\frac{1}{2} = \frac{7}{6}\,.$$

2. For $n = 4$ we obtain the following value:

$$f(4) = 1 + \frac{2}{4} - \frac{1}{3} = \frac{7}{6}\,.$$

Hence, we conclude that $\sup_{n,k} \left\{ \frac{\beta(M_{n,k})}{W(M_{n,k})} \right\} = \frac{7}{6}$. □

Corollary 10.17. *If there exists a real number c such that $\beta(M) \leq cW(M)$ for all binary matrices M, then $c \geq 7/6$.*

11 Conclusion

Evolutionary processes allow us to understand and study different phenomena in field of Molecular Biology [8]. It is crucial to understand the evolution of the molecular structure of the cancer cells in a tumor, in order to discover what mutations lead to out-of-control increase of the anomalous cells. As mentioned in Section 1, there has been a remarkable progress in single-cell analysis, however the input may still consist of different cancer cells, which gives rise to the challenge of understanding the cause of out-of-control growth of those cells. A common way of representing an evolutionary history is by means of a phylogeny [5]. It is known that tumor mutations satisfy the so-called infinite sites assumption (see [15]), which makes us refer to the perfect phylogeny evolutionary model. In the master thesis we relied on an approach introduced by Hajirasouliha and Raphael [17], who were among the first to propose the use of the perfect phylogeny evolutionary model to study the evolutionary history of tumor mutations.

We started the thesis by mentioning the motivation for the MUB (and consequently MCRS) problem, then we gave an overview of related recent research in the area, and provided the preliminary theory necessary for understanding the key known and new results. In Section 3, we gave an overview of the Minimum Conflict-free Row Split (MCRS) problem introduced by Hujdurović et al. in [18], following the work of Hajirasouliha and Raphael. Further, we introduced an equivalent problem, the so-called Minimum Uncovering Branching (MUB) problem formulated in terms of branchings in directed acyclic graphs, introduced in the same paper. We illustrated the key concepts with several concrete examples. Further, we presented the known results regarding the computational complexity of the MUB (and consequently MCRS) problem and two known approximation algorithms with approximation ratios expressed in terms of the width, resp. the height of the corresponding containment digraph. We summarized the main proof ideas from paper [18].

We reviewed a polynomially computable lower bound from [17] presented in terms of chromatic numbers of derived conflict graphs. We expressed the lower bound in an equivalent way, more specifically, in terms of widths of the principal subgraphs of the containment digraph and gave a detailed proof of the result. In addition, in Section 10 we introduced a new polynomially computable lower bound in terms of maximum

weight of an antichain and justified the polynomial time complexity of computing the bound. We also gave an overview of a min-max result that is a generalization of Dilworth's Theorem, introduced by Hujdurović et al. in [18]. The result implies the existence of a polynomial-time algorithm for computing an upper bound on the optimal value of the MUB problem, which relies on a variant of the problem, restricted to looking for an optimal solution only among the so-called linear branchings.

In the master thesis we introduced a couple of new results. Firstly, we identified two new polynomially solvable cases of the MUB problem. Secondly, as mentioned before, we introduced a new polynomially computable lower bound and investigated the open problem asking whether there exist a constant c such that $\beta \leq c \cdot \sum_r wdt(D_{M,r})$ for all binary matrices M on some specific families of instances.

To conclude, this area of research is rather new and many open questions remain. For instance, the following three:

- Does there exist a constant c such that $\beta \leq c \cdot \sum_r wdt(D_{M,r})$ for all binary matrices M? That would imply the existence of a c-approximation algorithm for the minimum conflict-free row-split problem.

- More generally, does there exist a constant factor approximation algorithm for the MUB problem?

- What is the complexity of the MUB problem when restricted to instances of width at most 3? Or, more generally, for instances of bounded width?

12 Povzetek dela v slovenskem jeziku

Optimizacijski problem, preučevan v magistrskem delu, je motiviran z genomiko raka. Predstavlja zelo pomembno področje raziskav, saj je rak eden vodilnih vzrokov smrti po vsem svetu (glej [6]). Klonalna teorija raka pravi, da rak nastane po zadostnem številu mutacij v tumorju, kar pomeni, da je ključnega pomena razumeti, katere mutacije sprožijo nekontrolirano rast abnormalnih celic. Čeprav se sodobne klinične tehnologije soočajo z izjemnim napredkom, je nemogoče natančno preučiti zgodovine mutacij tumorjev. Ena izmed možnosti je uporaba računskih modelov. Predpostavimo, da imamo m vzorcev tumorja in seznam n mutacij, do katerih je prišlo v vsakem od vzorcev. Hajirasouliha in Raphael v [17] predlagata uporabo naslednje metode. Uporabimo binarno matriko M velikosti $m \times n$, ki ima vzorce tumorja v vrsticah in mutacije v stolpcih. Element $M(i, j)$ ima vrednost 1, če v vzorcu i pride do mutacije j in 0 sicer. Matriko M želimo predstaviti z uporabo evolucijskega modela *popolne filogenije*, ki je eden najpogosteje uporabljenih znakovnih modelov za prikaz evolucijske zgodovine. V [10] in [15] je bilo dokazano, da binarna matrika M ustreza popolni filogeniji, če in samo če je M brezkonfliktna. Pravimo, da je matrika brezkonfliktna, če ne premore podmatrike velikosti 3×2 na poljubnih treh vrsticah r, r', r'' M in dveh stolpcih i, j, naslednje oblike

$$M[(r, r', r''), (i, j)] = \begin{pmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}.$$

V praksi je vsak vzorec tumorja mešanica odčitkov različnih tumorjev, zato binarna matrika M ni konfliktna. Za reševanje tega problema Hujdurović idr. v članku [19] sledijo delu Hajirasoulihe in Raphaela ter uvedejo problem najmanjšega brezkonfliktnega razcepa vrstic (Minimum Conflict-free Row Split (MCRS)). Dana je binarna konfliktna matrika M. Vsako vrstico matrike M želimo zapisati kot rezultat logične operacije ALI na množici vrstic, tako da zamenjava vsake vrstice z argumenti operacije ALI vodi do brezkonfliktne matrike, ki ima najmanjše možno število vrstic.

Na kratko orišimo potek magistrskega dela. V poglavju 1 razložimo motivacijo, razpravljamo o sorodnem gradivu na tem področju in podamo oris strukture prispevka ter navedemo glavne rezultate, predstavljene v magistrskem delu. Nato preidemo na

poglavje 2, kjer povzamemo osnovne definicije, potrebne za razumevanje glavnih rezultatov magistrskega dela. V poglavju 3 formalno definiramo popolno filogenijo in problem najmanjšega brezkonfliktnega razcepa vrstic (MCRS). Koncepte ilustriramo na številnih primerih, podanih v grafični obliki. V poglavju 4 predstavimo dva algoritma, ki tečeta v linearnem času, ki ju je uvedel Dan Gusfield v [16]. Prvi algoritem pove, ali binarna matrika M ustreza popolni filogeniji, drugi pa tako popolno flogenijo konstruira, če je matrika brezkonfliktna. Nato predstavimo problem, ki je enakovreden problemu MCRS. Problem se imenuje problem vejitve najmanjšega nepokritja (Minimum Uncovering Branching (MUB)) in je opisan v jeziku vejitev v usmerjenih acikličnih grafih. V tem poglavju predstavimo ideje dokazov glavnih rezultatov v zvezi s tem problemom. Nato preidemo na rezultate računske zahtevnosti problema MCRS. Natančneje, problema MUB in posledično MCRS sta APX-težka. V istem poglavju naredimo pregled znanih aproksimacijskih algoritmov. Aproksimacijska faktorja algoritmov sta izražena prek dveh količin, imenovani širina in višina. To sta invarianti digrafa vsebovanosti D_M, ki ustreza matriki M (podrobnosti in formalne definicije so navedene v poglavju 5). Poleg tega za vse rezultate iz poglavja 6 predstavimo ideje dokazov.

Predstavimo tudi znane, polinomsko izračunljive spodnje in zgornje meje za optimalno vrednost problemov MCRS in MUB. V poglavju 7 podamo definicijo znane spodnje meje, enakovredno definicijo in podroben dokaz. Nato v poglavju 8 uvedemo znan min-max rezultat, ki je posplošitev Dilworthovega izreka (glej [7]). Ta rezultat implicira polinomsko časovno zahtevnost izračuna zgornje meje, ki je predstavljena v poglavju 9. Nato v poglavju 10 predstavimo glavne rezultate magistrskega dela. Najprej uvedemo novo polinomsko izračunljivo spodnjo mejo, nato pa dokažemo izrek, ki pravi, da je problem MUB (in posledično MCRS) polinomsko rešljiv za vhodne podatke, podane z matriko širine 2. Poleg tega predstavimo dokaz izreka, ki pravi, da je problem MUB (in posledično MCRS) polinomsko rešljiv v primerih, kjer je širina enaka številu maksimalnih elementov v pripadajoči delno urejeni množici P_M binarne matrike M. Nadaljujemo z analizo kvalitete spodnje meje za določene posploštive posebnih družin primerov, ki so jih uvedli Hujdurović idr. v [18]. Na koncu poglavja predstavimo novo konstrukcijo vhodnih podatkov, na kateri podrobno preučimo odprto vprašanje o razmerju med optimalno vrednostjo problemov MCRS in MUB in polinomsko izračunljivo spodnjo mejo, podano v [17]. S poglavjem 11 zaključimo magistrsko delo in omenimo nekaj odprtih vprašanj s tega področja, ki bi lahko bila zanimiva za prihodnje raziskave.

13 References

[1] A. V. Aho, M. R. Garey, and J. D. Ullman, The transitive reduction of a directed graph. *SIAM J. Comput.* Vol. 1(2) (1972) 131–137. *(Cited on page 25.)*

[2] A. Aho and J. E. Hopcroft, *The Design and Analysis of Computer Algorithms.* Addison-Wesley Longman Publishing Co., Inc., 1974. *(Cited on page 16.)*

[3] P. Alimonti, and V. Kann, Some APX-Completeness Results for Cubic Graphs. *Theoretical Computer Science* 237 (2000) 123–134. *(Cited on page 3.)*

[4] D. Barh, V. Azevedo, Single-Cell Omics. In: S. Dwivedi, P. Purohit, R. Misra, M. Lingeswaran, J. Ram Vishnoi, P. Pareek, P. Sharma, S. Misra, *Application of Single-Cell Omics in Breast Cancer*, Academic Press, 2019, 69–103. *(Cited on page 1.)*

[5] P. Bonizzoni, A. P. Carrieri G. D. Vedova, R. Dondi, T. M. Przytycka, Discrete and Topological Models in Molecular Biology. In: N. Jonoska, M. Saito, *When and How the Perfect Phylogeny Model Explains Evolution*, Springer Berlin Heidelberg, 2014, 67–83. *(Cited on pages 2 and 58.)*

[6] F. Bray, J. Ferlay, I. Soerjomataram, R. L. Siegel, L. A. Torre, and A. Jemal, Global Cancer Statistics 2018: GLOBOCAN Estimates of Incidence and Mortality Worldwide for 36 Cancers in 185 Countries. *CA: A Cancer Journal for Clinicians* 68 (2018) 394–424. *(Cited on pages 1 and 60.)*

[7] R. P. Dilworth, A decomposition theorem for partially ordered sets. *Annals of Mathematics. Second Series* 51 (1950) 161–166. *(Cited on pages 4, 29, 34, and 61.)*

[8] T. Dobzhansky, Nothing in Biology Makes Sense except in the Light of Evolution. *The American Biology Teacher* 35 (1973) 125–129. *(Cited on page 58.)*

[9] M. El-Kebir, G. Satas and L. Oesper, and B. J. Raphael, Inferring the Mutational History of a Tumor Using Multi-state Perfect Phylogeny Mixtures. *Cell Systems* 3 (2016) 43–53. *(Cited on page 4.)*

[10] G. F. Estabrook, C. S. Johnson, and F. R. McMorris, An idealized concept of the true cladistic character. *Mathematical Biosciences* 23 (1975) 263–272. *(Cited on pages 2, 13, and 60.)*

[11] W. Fan, Z. Su, B. Yu, Y. Chen, W. Wang, Z. Song, Y. Hu, Z. Tao, J. Tian, Y. Pei, M. Yuan, F. Dai, Y. Liu, Q. Wang, J. Zheng, L. Xu, E. C. Holmes, and Y. Zhang, A new coronavirus associated with human respiratory disease in China. *Nature* 579 (2020) 265–269. *(Cited on page 4.)*

[12] D. R. Fulkerson, Note on Dilworth's decomposition theorem for partially ordered sets. *Proceedings of the American Mathematical Society* 7 (1956) 701–702. *(Cited on pages 33 and 34.)*

[13] F. L. Gall, Powers of Tensors and Fast Matrix Multiplication. In *Proceedings of the 39th International Symposium on Symbolic and Algebraic Computation*, 2014, 296–303. *(Cited on page 23.)*

[14] M. Grötschel, L. Lovász, and A. Schrijver, *Geometric Algorithms and Combinatorial Optimization*. Springer-Verlag, 1988. *(Cited on page 43.)*

[15] D. Gusfield, *Algorithms on Strings, Trees and Sequences: Computer Science and Computational Biology*, Cambridge Univ. Press, 1997. *(Cited on pages 1, 2, 13, 58, and 60.)*

[16] D. Gusfield, Efficient algorithms for inferring evolutionary trees. *Networks* 21 (1991) 19–28. *(Cited on pages 4, 13, 15, 18, and 61.)*

[17] I. Hajirasouliha and B. J. Raphael, Reconstructing mutational history in multiply sampled tumors using perfect phylogeny mixtures. *Proceedings of the 14th International Workshop on Algorithms in Bioinformatics (WABI'14)* Lecture Notes in Comput. Sci. 8701. Springer, Heidelberg (2018) 354–367. *(Cited on pages 2, 3, 13, 27, 31, 58, 60, and 61.)*

[18] A. Hujdurović, E. Husić, M. Milanič, R. Rizzi, and A. I. Tomescu, Perfect Phylogenies via Branchings in Acyclic Digraphs and a Generalization of Dilworth's Theorem. *ACM Transactions on Algorithms* 14 (2018) 1–26. *(Cited on pages VII, 3, 4, 5, 9, 13, 20, 22, 23, 24, 25, 26, 28, 29, 30, 34, 37, 38, 39, 40, 42, 49, 51, 58, 59, and 61.)*

[19] A. Hujdurović, U. Kačar, M. Milanič, B. Ries, and A. I. Tomescu, Complexity and Algorithms for Finding a Perfect Phylogeny from Mixed Tumor Samples. *IEEE/ACM Transactions on Computational Biology and Bioinformatics* 15 (2018) 96–108. *(Cited on pages 3, 13, 20, 32, and 60.)*

[20] E. Husić, X. Li, A. Hujdurović, M. Mehine, R. Rizzi, V. Mäkinen, M. Milanič, and A. I. Tomescu, MIPUP: minimum perfect unmixed phylogenies for multi-sampled tumors via branchings and ILP. *Bioinformatics* 35 (2018) 769–777. *(Cited on pages VII, 4, and 11.)*

[21] O. H. Ibarra, and S. Moran, Deterministic and probabilistic algorithms for maximum bipartite matching via fast matrix multiplication. *Information Processing Letters* 13 (1981) 12–15. *(Cited on page 34.)*

[22] M. Kochol, Complexity of 3-edge-coloring in the class of cubic graphs with a polyhedral embedding in an orientable surface. *Discrete Applied Mathematics* 158 (2010) 1856–1860. *(Cited on page 3.)*

[23] S. Malikic, A. W. McPherson, N Donmez, and C. S. Sahinalp, Clonality inference in multiple tumor samples using phylogeny. *Bioinformatics* 31 (2015) 1349–1356. *(Cited on page 4.)*

[24] L. Mirsky, A dual of Dilworth's decomposition theorem. *Amer. Math. Monthly* Vol. 78 (1971) 876–877. *(Cited on page 43.)*

[25] D. E. Newburger, D. Kashef-Haghighi, Z. Weng, R. Salari, R. T. Sweeney, A. L. Brunner, S. X. Zhu, X. Guo, S. Varma, M. L. Troxell, R. B. West, S. Batzoglou, and A. Sidow, Genome evolution during progression to breast cancer. *PCR Methods and Applications* 23 (2013) 1097–1108. *(Cited on page 1.)*

[26] P. C. Nowell, The Clonal Evolution of Tumor Cell Populations. *Science* 194 (1976) 46–52. *(Cited on page 1.)*

[27] V. Popic, R. Salari, I. Hajirasouliha, D. Kashef-Haghighi, R. West, and S. Batzoglou, Fast and Scalable Inference of Multi-Sample Cancer Lineages. *Genome biology* 16 (2014) 91. *(Cited on page 4.)*

[28] D. Ramazzotti, F. Angaroni, D. Maspero, C. Gambacorti-Passerini, M. Antoniotti, A. Graudenzi, and R. Piazza, Characterization of intra-host SARS-CoV-2 variants improves phylogenomic reconstruction and may reveal functionally convergent mutations. bioRxiv 2020.04.22.044404. *(Cited on page 4.)*

[29] J. Reiter, A. Makohon-Moore, J. Gerold, I. Bozic, K. Chatterjee, C. Iacobuzio-Donahue, B. Vogelstein, and M. Nowak, Reconstructing metastatic seeding patterns of human cancers. *Nature Communications* 8 (2017) 14114. *(Cited on page 4.)*