UNIVERZA NA PRIMORSKEM
FAKULTETA ZA MATEMATIKO, NARAVOSLOVJE IN INFORMACIJSKE TEHNOLOGIJE

Zaključna naloga
(Final project paper)

Aproksimacijski vidiki vektorske dominacije in vektorske povezanosti v grafih

(Approximability Aspects of Vector Domination and Vector Connectivity in Graphs)

Ime in priimek: Mirza Krbezlija
Študijski program: Matematika
Mentor: izr. prof. dr. Martin Milanič

Koper, september 2018

Ključna dokumentacijska informacija

Ime in PRIIMEK: Mirza KRBEZLIJA

Naslov zaključne naloge: Aproksimacijski vidiki vektorske dominacije in vektorske povezanosti v grafih

Kraj: Koper

Leto: 2018

Število listov: 46 Število slik: 1 Število tabel: 2

Število referenc: 32

Mentor: izr. prof. dr. Martin Milanič

Ključne besede: graf, vektorska dominacija, vektorska povezanost, problem submodularnega pokritja, aproksimacijski algoritem, požrešna metoda, celoštevilsko linearno programiranje

Math. Subj. Class. (2010): 05696, 05C40, 05B35, 05C58, 68W25, 90C10

Izvleček:
V zaključni nalogi obravnavamo problema vektorske dominacije in vektorske povezanosti v grafih. Problem vektorske dominacije je najti najmanjšo tako podmnožico S množice vozlišč danega grafa, da ima vsako vozlišce, ki ni v množici S, vsaj določeno število sosedov v množici S. Problem vektorske povezanosti je podobne narave, pri čemer namesto določenega števila sosedov želimo, da je vsako vozlišče, ki ni v množici, z množico povezano prek določenega števila disjunktnih poti. Znano je, da sta oba problema NP-težka na splošnih grafih. Po drugi strani, pa, kot so ugotovili Cicalese idr. leta 2013 in Boros idr. leta 2014, lahko s preprostima požrešnima algoritmoma problema aproksimiramo v polinomskem času na n-vozliščnih grafih s faktorjem $\ln n + 2$. To sledi iz splošnejšega rezultata, ki ga je Wolsey leta 1982 izpeljal za tako imenovani problem submodularnega pokritja. V zaključni nalogi podamo enotno predstavitev teh rezultatov, pokažemo povezavo z Mengerjevim izrekom in matroidi, problema vektorske dominacije in vektorske povezanosti modeliramo s celoštevilskima linearnima programoma in z uporabo CPLEX-a empirično ovrednotimo kvaliteto požrešnih metod na naključno generiranih vhodnih podatkih.

Key words documentation

Name and SURNAME: Mirza KRBEZLIJA

Title of final project paper: Approximability Aspects of Vector Domination and Vector Connectivity in Graphs

Place: Koper

Year: 2018

Number of pages: 46 Number of figures: 1 Number of tables: 2

Number of references: 32

Mentor: Assoc. Prof. Martin Milanič, PhD

Keywords: graph, vector domination, vector connectivity, submodular set covering problem, approximation algorithm, greedy heuristic, integer linear programming

Math. Subj. Class. (2010): 05696, 05C40, 05B35, 05C58, 68W25, 90C10

Abstract:
In this final project paper we consider the vector domination and vector connectivity problems in graphs. The vector domination problem is to find a smallest subset S of the vertex set of a given graph such that every vertex not in S has at least a certain number of neighbors in S. The vector connectivity problem is a problem of similar nature, however, instead of a certain number of neighbors we want that every vertex not in the set is connected to the set by a certain number of disjoint paths. Both problems are known to be NP-hard on general graphs. On the other hand, as observed by Cicalese et al. in 2013 and by Boros et al. in 2014, natural greedy algorithms can be used to approximate both problems in polynomial time on n-vertex graphs to within a factor of $\ln n + 2$. This follows from a more general result due to Wolsey from 1982 on the so-called submodular set covering problem. In the final project paper we give a unified presentation of these results, show a connection with Menger's theorem and matroids, model the two problems with integer linear programs, and use the CPLEX solver to evaluate the performance of the two greedy methods on randomly generated instances.

Acknowledgement

I would like to express my sincere gratitude to my mentor and professor Assoc. Prof. Martin Milanič for his guidance throughout the planning and working of this final project paper. I thank him for his patience and time that he spared.
I thank the Faculty of Mathematics, Natural Sciences and Information Technologies for giving me many opportunities to learn and study many things.
Finally, I would like to thank my parents and my brother for supporting me in every possible way throughout my education.

Contents

List of Tables

List of Figures

List of Abbreviations

i.e.	that is
e.g.	for example
s.t	subject to
et al.	and others

1 Introduction

Domination theory in graphs is a well-developed and still developing area of graph theory, motivated by a variety of real-life settings, including facility location problems, problems involving finding sets of representatives, monitoring communication or electrical networks, and land surveying. The main results and applications of domination in graphs are discussed by Haynes et al. in the two books [17, 18]. We will consider a quite general variant of domination, called Vector Domination, and a related problem, called Vector Connectivity.

The Vector Domination problem is the following: given a graph $G = (V, E)$ and a vector k indexed by vertices of G, that is, $k = (k_v : v \in V)$ such that for each $v \in V$, k_v is an integer between 0 and the degree of v, find a smallest set $S \subseteq V$ such that every vertex v that is not in S has at least k_v neighbors in S. The Vector Domination problem was introduced by Harant et al. [16] as a generalization of the classical Dominating Set and Vertex Cover problems in graphs.

The Vector Connectivity problem is the following: given a graph $G = (V, E)$ and a vector k indexed by vertices of G, that is, $k = (k_v : v \in V)$, such that, k_v is an integer between 0 and the degree of vertex v of G, find a smallest set $S \subseteq V$ such that every vertex v that is not in S has at least k_v vertex-disjoint paths to S, where, in this case, "vertex-disjoint" means that the paths have only one vertex in common, namely v. If the paths are required to be of length one, the problem becomes the vector domination problem. Therefore, the Vector Connectivity probelm is a relaxation of the Vector Domination problem, where the condition of the number of neighbors is replaced by the condition of the number of disjoint paths to the set S.

In Section 3, we will present Wolsey's algorithm for the submodular set covering problem [32] and its analysis. In Sections 4 and 5 we will show, following Cicalese et al. [8] and Boros et al. [5], respectively, that Vector Domination and Vector Connectivity problems are special cases of the submodular set covering problem, which leads to logarithmic approximation of the two problems. Analogous results for some related problems will also be presented, as well as the known inapproximability results for the considered problems.

We will model the Vector Domination and Vector Connectivity problems with integer linear programs, in Section 6 and analyze, in Section 7, with help of computer, the quality of the approximation algorithm based on Wolsey's theorem on randomly generated data. The optimal solution values will be computed exactly using CPLEX, via our integer linear programs. These experiments and results are presented in Section 7. We conclude the paper in Section 8 with a short summary and discussion.

2 Preliminaries

Before we start with the main topics, let us first overview some basic terminology, notations, and properties of graphs, hypergraphs, matroids, and submodular functions. Unless stated otherwise, we consider finite, undirected, and simple graphs without isolated vertices. A graph G will be denoted by $G = (V, E)$ where V is the vertex set and E the edge set of G. The degree of a vertex v in G will be denoted by $d(v)$ (or $d_G(v)$ if the graph is not clear from the context), the maximum degree of a vertex in G will be denoted by $\Delta(G)$. The vertex and the edge sets of G will be denoted by $V(G)$ and $E(G)$, respectively. By $N(v)$ (or $N_G(v)$ if the graph is not clear from the context) we will denote the set of neighbors of vertex v of $G = (V, E)$, that is the (open) neighborhood of v, and by $N[v] := N(v) \cup \{v\}$ (or $N_G[v]$), the closed neighborhood of v. For graph theoretic notions not defined here, we refere to West [31].

A *vertex cover* in a graph G is a set $S \subseteq V(G)$ such that each edge of G is incident to at least one vertex in S.

A *path* in G is a finite sequence $v_1, \{v_1, v_2\}, v_2, \ldots, \{v_{m-1}, v_m\}, v_m$ of distinct vertices v_i $(1 \leq i \leq m)$ and distinct edges (v_{j-1}, v_j) $(2 \leq j \leq m)$. We will sometimes identify a path in a graph G with the corresponding subgraph in G.

We say that a graph is *planar* if it can be drawn in the plane so that its edges intersect only at their endpoints. Given a graph G, its *line graph* $L(G)$ is a graph such that the vertices of $L(G)$ are in a bijective correspondence of G, and two disjoint vertices of $L(G)$ are adjacent if and only if their corresponding edges share a common endpoint in G. A graph G is said to be *bipartite* if its vertex set can be partitioned into two disjoint independent sets.

The *girth* of a graph G is the length of a shortest cycle contained in G. If we have an acyclic graph, that is, a graph that contains no cycle, then the girth is defined to be ∞.

A *directed graph* or *digraph* is an ordered pair $D = (V, A)$ where V is the set of vertices and A is the set of ordered pairs of vertices called directed edges. A *(directed) path* in a digraph D is a finite sequence $v_1, (v_1, v_2), v_2, \ldots, (v_{m-1}, v_m), v_m$ of pairwise distinct vertices v_i $(1 \leq i \leq m)$ and pairwise distinct directed edges (v_{j-1}, v_j) $(2 \leq j \leq m)$. The vertices v_1 and v_m are called the *initial* and the *terminal* vertex of the path, respectively, and all the other vertices in the path are called *internal*. A set of paths

in D is *pairwise vertex-disjoint* if no two paths have a vertex in common.
Let $D = (V, A)$ be a digraph and let $s, t \in V$. A function $f : A \to \mathbb{R}$ is called a *flow* from s to t, or an *s-t flow*, if:

(*i*) $f(a) \geq 0$ for all $a \in A$,

(*ii*) $\sum_{w:(w,v)\in A} f((w, v)) = \sum_{w:(v,w)\in A} f((v, w))$ for each $v \in V \setminus \{s, t\}$.

The *value* of an *s-t* flow f is, by definition $value(f) := \sum_{w:(w,s)\in A} f((w, s)) - \sum_{w:(s,w)\in A} f((s, w))$.
Let $c : A \to \mathbb{R}_+$ be a capacity function. We say that a flow f is *under c* if $f(a) \leq c(a)$ for each $a \in A$. We define the *maximum s-t flow*, or just *maximum flow*, to be an *s-t* flow under c, of maximum value. For more maximum flow related concepts, see, e.g., [28].
For the running time analysis of algorithms on graphs or digraphs in Sections 4 and 5 we will assume, for simplicity, that each graph or digraph is represented with its adjacency matrix.
For a set X we denote by $\mathcal{P}(X)$ the *power set* of X, that is, the set of all subsets of a X.
For a finite set U let $f : \mathcal{P}(U) \to \mathbb{R}$ be a function that assigns to each subset $S \subseteq U$ a real value $f(S)$. We say that f is *submodular* if for all $X, Y \in \mathcal{P}(U)$ the following inequality holds: $f(X) + f(Y) \geq f(X \cup Y) + f(X \cap Y)$ (see, e.g., [28]).
The following characterization of submodular functions can be found in e.g., Schrijver [28, Theorem 44.1].

Proposition 2.1. *A set function $f : \mathcal{P}(U) \to \mathbb{R}$ is submodular if and only if*

$$f(S \cup \{s\}) + f(S \cup \{t\}) \geq f(S) + f(S \cup \{s, t\}) \tag{2.1}$$

for each $S \subseteq U$ and distinct $s, t \in U \setminus S$.

We say that a set function $f : \mathcal{P}(U) \to \mathbb{R}$ is *non-decreasing* if $f(A) \leq f(B)$ whenever $A \subseteq B \subseteq U$. Non-decreasing submodular functions are characterized as follows.

Proposition 2.2 (Nemhauser and Wolsey [23]). *Let U be a finite set and let $f : \mathcal{P}(U) \to \mathbb{R}$ be a set function. Then, the following statements are equivalent:*

(a) f is submodular and non-decreasing,

(b) $f(S \cup \{j\}) - f(S) \geq f(T \cup \{j\}) - f(T) \geq 0$ for all $S \subseteq T \subseteq U$ and $j \in U \setminus T$,

(c) $f(T) \leq f(S) + \sum_{j\in T\setminus S}(f(S \cup \{j\}) - f(S))$ for all $S, T \subseteq U$.

Proof. We will first prove the equivalence between (a) and (b). Suppose that f is submodular and non-decreasing. Let $S \subseteq T \subseteq U$, and let us take an arbitrary $j \in U \backslash T$. Since f is submodular, we have $f(S \cup \{j\}) + f(T) \geq f((S \cup \{j\}) \cup T) + f((S \cup \{j\}) \cap T)$ which is equivalent to $f(S \cup \{j\}) - f(S) \geq f(T \cup \{j\}) - f(T)$. Since f is non-decreasing and $T \subseteq T \cup \{j\}$ we get that $f(T \cup \{j\}) - f(T) \geq 0$.
Now, suppose that (b) holds. Let $X, Y \subseteq U$ be arbitrary and let $X \backslash Y = \{x_1, x_2, \ldots, x_r\}$. If $X \subseteq Y$, then $f(X \cup Y) + f(Y \cap Y) = f(X) + f(Y)$ thus, f is submoduler. Suppose that $r \geq 1$. Then by our condition we get the following inequalities:

$f(Y \cup \{x_1\}) - f(Y) \leq f((X \cap Y) \cup \{x_1\}) - f(X \cap Y)$,

$f(Y \cup \{x_1, x_2\}) - f(Y \cup \{x_1\}) \leq f((X \cap Y) \cup \{x_1, x_2\}) - f((X \cap Y) \cup \{x_1\})$,

$f(Y \cup \{x_1, x_2, x_3\}) - f(Y \cup \{x_1, x_2\}) \leq f((X \cap Y) \cup \{x_1, x_2, x_3\}) - f((X \cap Y) \cup \{x_1, x_2\})$,

$$\vdots$$

$f(Y \cup \{x_1, \ldots, x_r\}) - f(Y \cup \{x_1, \ldots, x_{r-1}\}) \leq f((X \cap Y) \cup \{x_1, \ldots, x_r\}) - f((X \cap Y)$ $\cup \{x_1, \ldots, x_{r-1}\})$. If we sum them up, we get:

$$f(Y \cup \{x_1, \ldots, x_r\}) - f(Y) \leq f(X \cap Y \cup \{x_1, \ldots, x_r\}) - f(X \cap Y),$$

and since $Y \cup \{x_1, \ldots, x_r\} = X \cup Y$ and $X \cap Y \cup \{x_1, \ldots, x_r\} = X$, we obtain the inequality $f(X \cup Y) - f(Y) \leq f(X) - f(X \cap Y)$. If we take $X \subseteq Y \subseteq U$ and $Y \setminus X = \{y_1, \ldots, y_r\}$ (with $r \geq 1$) we get from $f(S \cup \{j\} - f(S) \geq 0$ that $f(X) \leq f(X \cup \{y_1\} \leq f(X \cup \{y_1, y_2\}) \leq \ldots \leq f(X \cup \{y_1, \ldots, y_r\}) = f(Y)$ which means that f is non-decreasing.
Now we will prove that (a) is equivalent to (c). Assume that f is a submodular, non-decreasing function and take arbitrary $S, T \subseteq U$. If $T \subseteq S$, then since f is non-decreasing, we have $f(T) \leq f(S) = f(S) + \sum_{j \in T \setminus S}(f(S \cup \{j\}) - f(S))$ and (c) holds. Suppose now that $T \setminus S = \{j_1, \ldots, j_r\}$ with $r \geq 1$. A repeated application of the fact that f is submodular yields:

$$\begin{aligned}
\sum_{j \in T \setminus S} f(S \cup \{j\}) = f(S \cup \{j_1\}) + f(S \cup \{j_2\}) + f(S \cup \{j_3\}) + \ldots + f(S \cup \{j_r\}) &\geq \\
f(S) + f(S \cup \{j_1, j_2\}) + f(S \cup \{j_3\}) + \ldots + f(S \cup \{j_r\}) &\geq \\
2f(S) + f(S \cup \{j_1, j_2, j_3\}) + \ldots + f(S \cup \{j_r\}) &\geq \\
\ldots \geq f(S \cup \{j_1, \ldots, j_r\}) + (r-1)f(S).
\end{aligned}$$

Thus we obtain:

$$f(S) + \sum_{j \in T \setminus S}(f(S \cup \{j\}) - f(S)) = \sum_{j \in T \setminus S} f(S \cup \{j\}) - (r-1)f(S) \geq$$

$$\geq f(S \cup \{j_1, \ldots, j_r\}) = f(T).$$

To see that (c) implies (a) suppose that $f(T) \leq f(S) + \sum_{j\in T\setminus S}(f(S\cup\{j\}) - f(S))$ holds for all $S, T \subseteq U$. Let $S \subseteq U$. Let $s, t \in U \setminus S$. If we take $T := S \cup \{s, t\}$, then

$$f(T) = f(S\cup\{s,t\}) \leq f(S) + (f(S\cup\{s\}) - f(S)) + (f(S\cup\{t\}) - f(S))$$

which is equivalent to:

$$f(S\cup\{s,t\}) + f(S) \leq f(S\cup\{s\}) + f(S\cup\{t\})$$

and since the last inequality holds for arbitrary S, by Proposition 2.1, we have that f is submodular.

Finally, let $A \subseteq B$, where $B \setminus A = \{b_1, \ldots, b_r\}$ with $r \geq 1$. If we take that $T = A$ and $S = A \cup \{b_1\}$, in condition (c), then we have $f(A) \leq f(A \cup \{b_1\}) + 0$. Furthermore, if we take $T = A \cup \{b_1\}$ and $S = A \cup \{b_1, b_2\}$, we get $f(A \cup \{b_1\}) \leq f(A \cup \{b_1, b_2\})$ and if we iterate that r times we obtain the following sequence of inequalities: $f(A) \leq f(A \cup \{b_1\}) \leq f(A \cup \{b_1, b_2\}) \leq \ldots \leq f(A \cup \{b_1, \ldots, b_r\}) = f(B)$, which implies that f is non-decreasing. Thus, all the three statements are equivalent. □

We show next that the family of submodular functions defined over the same ground set is closed under finite sums.

Proposition 2.3. *Let U be a finite ground set. Suppose that $g_1, g_2, \ldots, g_n : \mathcal{P}(U) \to \mathbb{R}$ are submodular. Then the function $f : \mathcal{P}(U) \to \mathbb{R}$, defined as $f(X) = g_1(X) + \ldots + g_n(X)$ for all $X \subseteq U$, is also submodular.*

Proof. Let $X, Y \subseteq U$. Since $g_1, \ldots, g_n$ are submodular, we have that $g_i(X) + g_i(Y) \geq g_i(X \cap Y) + g_i(X \cup Y)$ for all $i = 1, \ldots, n$. Summing up over all i, we get

$$\sum_{i=1}^{n}(g_i(X) + g_i(Y)) \geq \sum_{i=1}^{n}(g_i(X\cap Y) + g_i(X\cup Y)),$$

which is equivalent to

$$\sum_{i=1}^{n} g_i(X) + \sum_{i=1}^{n} g_i(Y) \geq \sum_{i=1}^{n} g_i(X\cap Y) + \sum_{i=1}^{n} g_i(X\cup Y),$$

and that is further equivalent to

$$f(X) + f(Y) \geq f(X\cap Y) + f(X\cup Y).$$ □

A *hypergraph* is a pair $H = (U, \mathcal{F})$ where U is a finite set of vertices, also called the ground set, and $\mathcal{F}$ is a set of subsets of U. Hypergraphs are combinatiorial objects generelizing graphs. Many results on hypergraphs can be found in [3].

A kind of hypergraphs that are of particular importance for combinatorial computations are matroids, see, e.g., Welsh [30]. A *matroid* is a hypergraph $M = (U, \mathcal{F})$ such that $\mathcal{F}$

is nonempty and its elements, called *independent sets*, satisfy the following properties: (a) if Y is an independent set and $X \subseteq Y$, then X is also an independent set; and (b) *the exchange property*: for every two independent sets A and B such that $|A| < |B|$, there exists an element of B whose addition to A results in a larger independent set. A subset of the ground set U that is not independent is called *dependent*.
A *maximal independent set* is an independent set in a matroid that becomes dependent after adding to it any element of the ground set U, not contained in the set. A maximal independent set is called also called a *basis* of the matroid. A *maximum independent set* is an independent set of maximum size.
The following well-known property of matroids follows easily from the exchange property.

Proposition 2.4. *Let $M = (U, \mathcal{F})$ be a matroid and let $J \in \mathcal{F}$ be a maximal independent set. Then J is also a maximum independent set.*

Proof. Suppose that $J \in \mathcal{F}$ is a maximal independent set and that J is not a maximum independent set. Let $I \in \mathcal{F}$ be a maximum independent set. Since J is not maximum, we have that $|J| < |I|$. By the exchange property of matroids, there exists $i \in I$ such that $J \cup \{i\}$ is independent, which is a contradiction to the assumption that J is maximal. Hence, J is also a maximum independent set. □

Proposition 2.4 can also be stated as follows: All bases of a matroid have the same size. A special case of this statement is the fact that all bases of a finite-dimensional vector space are of the same size.
We say that $I \in \mathcal{F}$ is a *spanning set* if $B \subseteq I$ for some basis B of the matroid.
Given a matroid $M = (U, \mathcal{F})$, the *rank function* of M is the function $r_M : \mathcal{P}(U) \to \mathbb{Z}_+$ that assigns to every subset S of U the maximum size of an independent set contained in S.
The following property of rank functions of matroids is well known.

Lemma 2.5. *For every matroid $M = (U, \mathcal{F})$, its rank function r_M is submodular.*

Proof. Consider any two sets $X, Y \subseteq U$. Let J be a maximal independent subset of $X \cap Y$; thus, $|J| = r_M(X \cap Y)$. Let J_X be any maximal independent set contained in X such that $J \subseteq J_X$. Then, by Proposition 2.4 we have that J_X is maximum, hence, $|J_X| = r_M(X)$. Furthermore, by the maximality of J within $X \cap Y$, we know that

$$J_X \setminus Y = J_X \setminus J. \tag{2.2}$$

Now extend J_X to a maximal independent set J_{XY} of $X \cup Y$. Thus, $|J_{XY}| = r_M(X \cup Y)$. In order to prove that

$$r_M(X) + r_M(Y) \geq r_M(X \cap Y) + r_M(X \cup Y)$$

or equivalently

$$|J_X| + r_M(Y) \geq |J| + |J_{XY}|,$$

we need to show that $r_M(Y) \geq |J|+|J_{XY}|-|J_X|$. Observe that $J_{XY} \cap Y$ is independent (by property (a) of definition of matroids) and a subset of Y, and thus $r_M(Y) \geq |J_{XY} \cap Y|$. Observe now that

$$J_{XY} \cap Y = J_{XY} \setminus (J_X \setminus Y) = J_{XY} \setminus (J_X \setminus J),$$

the first equality following from the fact that $J_{XY} \setminus (J_X \setminus Y) = (J_{XY} \cap Y) \cup (J_{XY} \setminus J_X)$ and $J_{XY} \setminus J_X \subseteq Y$, and the second equality by (2.2). Therefore,

$$r_M(Y) \geq |J_{XY} \cap Y| = |J_{XY} \setminus (J_X \setminus J) = |J_{XY}| - |J_X| + |J|,$$

proving the lemma. □

A *gammoid* is a hypergraph $\Gamma = (U, \mathcal{F})$ derived from a triple (D, S, T) where $D = (V, A)$ is a digraph and $S, T \subseteq V$, $S \cap T = \emptyset$, such that $U = S$ and a subset S' of S forms a hyperedge if and only if there exist $|S'|$ vertex-disjoint directed paths in D connecting S' to a subset of T.

Lemma 2.6 (Perfect [26], Pym [27])**.** *Let $\Gamma = (U, \mathcal{F})$ be a gammoid, derived from a triple (D, S, T). Then Γ is a matroid.*

We give a proof of a weaker version of Lemma 2.6, following Perfect [26].

Lemma 2.7 (The case of Theorem 5.1 in [26] for finite graphs)**.** *Let $\Gamma = (U, \mathcal{F})$ be a gammoid, derived from a triple (D, S, T), such that no path from S to T has an internal vertex in S. Then Γ is a matroid.*

The proof is based on the following lemma.

Lemma 2.8. *Let $\Gamma = (U, \mathcal{F})$, derived from the triple (D, S, T), be a gammoid. Let $S' \subseteq S$ and let k be an integer such that $|S'| \leq k \leq |S|$. Let D' be the graph with vertex set $V(D) \cup F$ and edge set $A(D) \cup \{(u, v) : u \in S \setminus S',\ v \in F\}$, where F is a set that satisfies $V(D) \cap F = \emptyset$ and $|F| = |S| - k$. Then, if there exist $|S|$ pairwise vertex-disjoint paths from S to $T \cup F$ in D', then there exist k pairwise vertex-disjoint paths from S to T in D whose initial vertices contain the set S'.*

Proof. Assume all the notation from the lemma and suppose also that there exist $|S|$ pairwise vertex-disjoint paths from S to $T \cup F$ in D'.
Let $t = k - |S'|$. Since the number of pairwise vertex-disjoint paths from S to $T \cup F$ in D' is $|S|$, and $|F| = |S| - k$, and there is no path leading from F, we have that there are at least $|S| - |F|$ pairwise vertex-disjoint paths from S to T in D. Observe that

there is no path from S' to F.[1] Therefore, there are $|S'|$ pairwise vertex-disjoint paths from S' to T. We have that $|F| = |S| - k = |S| - (|S'| + t)$. Hence, there are at least t paths going from $S \setminus S'$ to T. Thus, there are $|S'| + t = k$ pairwise vertex-disjoint paths from S to T in D. □

Now, we can prove Lemma 2.7.

Proof of Lemma 2.7. To prove that Γ is a matroid, we need to verify that $\mathcal{F}$ is nonempty and its elements satisfy properties (a) and (b) from the definition of the matroid.
Obviously, $\mathcal{F}$ is nonempty since $\emptyset \in \mathcal{F}$.
To prove property (b), we will prove the following equivalent statement: For every two independent sets A and B such that $|B| = |A| + 1$, there exists an element b in B such that $A \cup \{b\}$ is independent. Let $A \subseteq S$ be such that $A \in \mathcal{F}$, and let $|A| = m$. Let $B \subseteq S$ be such that $B \in \mathcal{F}$ and $|B| = m+1$. Let $C = V(D) \setminus (A \cup B)$, F be a set of new vertices such that $F \cap V = \emptyset$ and $|F| = |A \cup B| - (m+1)$. Let D' be a new graph with vertex set $V(D') = V(D) \cup F$ and edge set $A(D') = A(D) \cup \{(u, v) : u \in B \setminus A,\ v \in F\}$. Let $V' = \widetilde{K} \cup \widetilde{C} \cup \widetilde{F}$, where $\widetilde{K} \subseteq A \cup B$, $\widetilde{C} \subseteq C$ and $\widetilde{F} \subseteq F$, be a set of vertices separating $T \cup F$ from $A \cup B$ in D'. Since there are m pairwise vertex-disjoint paths from A to T in D and no path from A to T has an internal vertex in $B \setminus A$ we have:

(i) $|\widetilde{K} \cap A| + |\widetilde{C}| \geq m$

and,

since there are $m + 1$ vertex-disjoint paths from B to T in D, we have:

(ii) $|\widetilde{K}| + |\widetilde{C}| \geq m + 1$.

Note that, either $\widetilde{F} = F$ or $\widetilde{K} \supseteq B \setminus A$.
If $F = \widetilde{F}$ then from (ii) we get $|V'| = |\widetilde{K}| + |\widetilde{C}| + |A \cup B| - (m + 1) \geq |A \cup B|$.
If $\widetilde{K} \supseteq B \setminus A$ then from (i) we get $|V'| = |A \cup B| - m + |\widetilde{K} \cap A| + |\widetilde{C}| + |\widetilde{F}| \geq |A \cup B|$.
Both cases yield that $|V'| \geq |A \cup B|$ and hence, by Menger's theorem [22], there exist $|A \cup B|$ pairwise vertex-disjoint paths from $A \cup B$ to $T \cup F$ in D'. Therefore, by Lemma 2.8, there exists set $K \subseteq S$ such that $K \in \mathcal{F}$, $|K| = m+1$ and $A \subseteq K \subseteq A \cup B$. This proves property (b). □

We will denote the natural logarithm (that is, logarithm with base e) as ln. For an $m \times n$ matrix A and a set $\mathbb{F} \in \{\mathbb{R}, \mathbb{Q}, \mathbb{Z}\}$, we write $A \in \mathbb{F}^{m \times n}_{\geq 0}$ when $a_{ij} \in \mathbb{F}$ and $a_{ij} \geq 0$ for all $i = 1, \ldots, n$ and $j = 1, \ldots, m$. And for a vector b, we write $b \in \mathbb{F}^{m}_{>0}$ when $b_j \in \mathbb{F}$ and $b_j > 0$ for all $j = 1, \ldots, m$.

[1]Suppose that there are l such paths. Since F has only edges in $S \setminus S'$, we need to have at least l vertices in $S \setminus S'$ to be part of the paths from S' to F. Then the number of pairwise vertex-disjoint paths from S to $T \cup F$ is at most $|S| - l$. Therefore, $l = 0$ and there is no path from S to F.

For a finite set U and a set $S \subseteq U$, we define the *characteristic vector* of S, denoted by $x^S = (x_j : \ j \in U)$, by the rule

$$x_j^S = \begin{cases} 1, & \text{if } j \in S; \\ 0, & \text{otherwise.} \end{cases}$$

We define an *optimization problem* to be a problem of finding the "best" solution from all feasible solutions. More formally, an optimization problem is a 4-tuple $\Pi = (\mathcal{I}, \mathcal{F}, f, opt)$, where $\mathcal{I}$ is a set of *instances*, $\mathcal{F} = (X_I : I \in \mathcal{I})$ is a collection of (usually implicitly given) *feasible solutions* for the problem Π on each instance I, $f : \bigsqcup_{I \in \mathcal{I}} X_I \to \mathbb{R}$ is the *objective function* and $opt \in \{\max, \min\}$.
Let us consider an minimization problem Π. For any instance I of Π we denote the optimal value of the objective function for I as $OPT_\Pi(I)$. If we have a polynomial-time algorithm A which returns some feasible solution X for Π, we denote by $A_\Pi(I)$ the *objective value* $f(X)$ returned by A given an instance I of Π. Then, assuming that Π is a minimization problem, the ratio

$$\rho_\Pi(I) = \frac{A_\Pi(I)}{OPT_\Pi(I)}$$

denotes the *approximation ratio* of algorithm A on the instance I.
Suppose that for each $\epsilon > 0$, we have an $(1+\epsilon)$-approximation algorithm for a minimization problem Π, whose running time is polynomial for a fixed ϵ. Such a family of algorithms is called a *polynomial time approximation scheme (PTAS)* for Π.
For any term mentioned regarding linear or integer programming that is not defined in this paper, see, e.g., [15] and for approximation algorithms, see, e.g. [2], [25], or [29].
For a positive integer d, we define $H(d) = \sum_{i=1}^{d} \frac{1}{i}$. The following well-known upper bound on the values of H will be useful for an analysis of the approximation ratios of the greedy algorithms for special cases of the submodular set covering problem.

Lemma 2.9. *Let n be a positive integer. Then $H(n) \leq 1 + \ln n$.*

Proof. Notice first that

$$\int_1^n \frac{1}{t} dt = \ln n - \ln 1 = \ln n.$$

Consequently, we have that

$$\sum_{k=2}^{n} \int_{k-1}^{k} \frac{1}{t} dt \leq \ln n.$$

Because $\frac{1}{t}$ is decreasing, $\frac{1}{a} \geq \frac{1}{t} \geq \frac{1}{b}$ for any $a, b \in (0, \infty)$ and $t \in [a, b]$, we therefore have:

$$\sum_{k=2}^{n} \int_{k-1}^{k} \frac{1}{k} dt \leq \ln n,$$

which is equivalent to

$$\sum_{k=2}^{n} \frac{1}{k} \leq \ln n$$

which implies $H(n) \leq 1 + \ln n$. $\square$

Given real numbers $a_1, \ldots, a_n$ we denote by $\operatorname{avg}_i a_i$ their average, given by the expression $\frac{1}{n} \sum_{i=1}^{n} a_i$, and by $\operatorname{stdev}_i a_i$ their standard deviation given by the expression

$$\sqrt{\frac{\sum_{j=1}^{n}(a_j - \operatorname{avg}_i a_i)^2}{n}}.$$

3 Greedy Algorithm for the Submodular Set Covering Problem

This section follows Wolsey [32]. The results presented in this section will be used in Sections 4 and 5 to derive the approximation results. All the results in this section are from [32], unless stated otherwise.

3.1 The submodular set covering problem

Before defining the submodular set covering problem, we will first look at the *integer covering problem*

$$\min\left\{\sum_{j=1}^{n} c_j x_j :\ Ax \geq b,\ x_j \in \{0,1\},\ j = 1,\dots,n\right\} \tag{C}$$

where $A \in \mathbb{R}_{\geq 0}^{m\times n}$ (or $\mathbb{Q}_{\geq 0}^{m\times n}$), $b \in \mathbb{R}_{>0}^{m}$ (or $\mathbb{Q}_{>0}^{m}$) and $c_j > 0$ for all $i = 1,\dots,n$ and $j = 1,\dots,m$. The behaviour of the greedy heuristic for the integer covering problem has been studied by Dobson [11]. The optimality of the greedy algorithm for finding a minimum weight basis in a matroid is also a classic result by now and well known, see [12].

We can define the *submodular set covering problem*, which is a generalization of both problems above, as follows:

$$Z = \min_{S\subseteq U}\left\{\sum_{j\in S} c_j :\ f(S) = f(U)\right\} \tag{Q}$$

where $f : \mathcal{P}(U) \to \mathbb{R}$ is a non-decreasing, submodular set function on a finite non-empty set U. To see that the integer covering problem (C) is a special case of (Q), it suffices to take the function $f : \mathcal{P}(U) \to \mathbb{R}$ defined as $f(S) := \sum_{i=1}^{m} \min\{\sum_{j\in S} a_{ij}, b_i\}$ for all $S \subseteq U$, while we obtain the minimum weight spanning set of a matroid by taking f to be the rank function of the matroid. Indeed, an optimal solution S satisfies $f(S) = f(U)$, which implies that S is a spanning set of the matroid [13], and by construction, it will be of minimum weight.

Observation 1. *Function $f : \mathcal{P}(U) \to \mathbb{R}$, defined as $f(S) = \sum_{i=1}^{m} \min\{\sum_{j\in S} a_{ij}, b_i\}$ for all $S \subseteq U$, is submodular.*

Proof. Let $X, Y \subseteq U$ and let $f : \mathcal{P}(U) \to \mathbb{R}$ be defined as above. Consider the following inequality for arbitrary $i \in \{1, \ldots, m\}$:

$$\min\left\{\sum_{j\in X} a_{ij}, b_i\right\} + \min\left\{\sum_{j\in Y} a_{ij}, b_i\right\} \geq \min\left\{\sum_{j\in X\cup Y} a_{ij}, b_i\right\} + \min\left\{\sum_{j\in X\cap Y} a_{ij}, b_i\right\} \tag{3.1}$$

To prove that (3.1) is true we consider the following cases.

1: If $b_i \leq \sum_{j\in X} a_{ij}$ then $b_i \leq \sum_{j\in X\cup Y} a_{ij}$ (since $a_{ij} \geq 0$ for all i, j) and hence (3.1) becomes equivalent to $\min\{\sum_{j\in Y} a_{ij}, b_i\} \geq \min\{\sum_{j\in X\cap Y} a_{ij}, b_i\}$ which implies, since $\sum_{j\in Y} a_{ij} \geq \sum_{j\in X\cap Y} a_{ij}$ and the function $g_i : \mathbb{R} \to \mathbb{R}$, defined by $g_i(x) = \min\{x, b_i\}$, is non-decreasing, that (3.1) is true.

2: If $b_i \leq \sum_{j\in Y} a_{ij}$, then, by symmetry with case 1, we get that (3.1) is true.

3: If $b_i > \sum_{j\in X} a_{ij}$ and $b_i > \sum_{j\in Y} a_{ij}$ then we have that either $b_i \geq \sum_{j\in X\cup Y} a_{ij}$ or $b_i < \sum_{j\in X\cup Y} a_{ij}$. If $b_j \geq \sum_{j\in X\cup Y} a_{ij}$, then (3.1) is equivalent to $\sum_{j\in X} a_{ij} + \sum_{j\in Y} a_{ij} \geq \sum_{j\in X\cup Y} a_{ij} + \sum_{j\in X\cap Y} a_{ij}$ which is obviously true (in fact, it holds with equality). If $b_i < \sum_{j\in X\cup Y} a_{ij}$, then (3.1) is equivalent to $\sum_{j\in X} a_{ij} + \sum_{j\in Y} a_{ij} \geq b_i + \sum_{j\in X\cap Y} a_{ij}$ which is equivalent to $\sum_{j\in X} a_{ij} + \sum_{j\in Y\setminus X} a_{ij} + \sum_{j\in X\cap Y} a_{ij} \geq b_i + \sum_{j\in X\cap Y} a_{ij}$. Since the last equality is true, we have that (3.1) is true as well.

Since in all cases, we have that

$$\min\left\{\sum_{j\in X} a_{ij}, b_i\right\} + \min\left\{\sum_{j\in Y} a_{ij}, b_i\right\} \geq \min\left\{\sum_{j\in X\cup Y} a_{ij}, b_i\right\} + \min\left\{\sum_{j\in X\cap Y} a_{ij}, b_i\right\}$$

is true for arbitrary $i \in \{1 \ldots, m\}$, hence we get that

$$\begin{aligned}\sum_{i=1}^{m}\min\left\{\sum_{j\in X} a_{ij}, b_i\right\} + \sum_{i=1}^{m}\min\left\{\sum_{j\in Y} a_{ij}, b_i\right\} \;\; &\geq \;\; \sum_{i=1}^{m}\min\left\{\sum_{j\in X\cup Y} a_{ij}, b_i\right\} \\ &\quad + \sum_{i=1}^{m}\min\left\{\sum_{j\in X\cap Y} a_{ij}, b_i\right\}\end{aligned}$$

is true as well. Therefore, f, defined as above, is submodular. □

In this section, we will show that if a greedy heuristic is applied to problem (Q), the value Z^G of a greedy heuristic solution always satisfies $Z^G \leq (1+\ln\gamma)Z$ where γ is one of several possible problem parameters. In the special case when f is integer-valued, the analysis gives $Z^G/Z \leq H(\max_j f(\{j\}) - f(\emptyset))$. This leads to an error factor of $H(\max_j \sum_{i=1}^{m} a_{ij})$ for problem (C) with integer data, which is a result of Dobson [11], generalizing earlier results of Johnson [19], Lovász [21], and Chvátal [7] for the set covering problem. The set covering problem takes as input a hypergraph $H = (U, \mathcal{F})$

and the task is to find a minimum size subfamily $\mathcal{F}' \subseteq \mathcal{F}$ such that every element of the ground set appears in at least one of the sets in $\mathcal{F}'$.
If f is the rank function of a nontrivial matroid, then $\max_j(f(\{j\}) - f(\emptyset)) = 1$, $H(1) = 1$; consequently, the greedy is optimal.

3.2 Problem reformulation and the greedy heuristic

We will present a reformulation of (Q) as an integer linear program. To do this, we will first see two properties of submodular functions.
Let $\varrho_j(S) := f(S \cup \{j\}) - f(S)$.
Consider the following integer linear program:

$$Z_I = \min \sum_{j \in U} c_j x_j \tag{Q^I}$$

$$\begin{aligned} s.t. \quad & \sum_{j \in U} \varrho_j(S) x_j \geq f(U) - f(S) \quad \textit{for all } S \subseteq U \\ & x_j \in \{0,1\}, \ \textit{for all } j \in U. \end{aligned}$$

Proposition 3.1. *A set $T \subseteq U$ is feasible for (Q) if and only if its characteristic vector x^T is feasible for (Q^I).*

Proof. Suppose that T is feasible for (Q). Let $S \subseteq U$. If $j \in S$, then $\varrho_j(S) = f(S \cup \{j\}) - f(S) = f(S) - f(S) = 0$. Thus, for the characteristic vector x^T, we have that $\sum_{j \in U} \varrho_j(S) x_j^T = \sum_{j \in T \setminus S} \varrho_j(S)$. By Proposition 2.2 we have that $\sum_{j \in T \setminus S} \varrho_j(S) \geq f(T) - f(S)$. Since T is feasible in (Q), we have that $f(T) = f(U)$. Thus, $\sum_{j \in U} \varrho_j(S) x_j^T \geq f(U) - f(S)$ for all $S \subseteq U$, which means that x^T is feasible in (Q^I).
On the other hand, if x^T is feasible in (Q^I) and we consider the constraint indexed by T, we have $x_j^T = 0$ for all $j \notin T$ and $f(T \cup \{j\}) = f(T)$ for $j \in T$, therefore, $0 = \sum_{j \in U} \varrho_j(T) x_j^T \geq f(U) - f(T)$ and hence $f(T) = f(U)$ (note that $f(T) > f(U)$ is impossible since f is non-decreasing and $T \subseteq U$), which means that T is feasible in (Q).

□

Consider the following greedy algorithm for (Q):

Greedy heuristic for (Q):

Initialization: Set $t = 1$, $S^0 = \emptyset$. Stop if $f(\emptyset) = f(U)$.
Iteration:

1. Let $\theta^t = \min_{j \in U \setminus S^{t-1}} \left\{ \frac{c_j}{\varrho_j(S^{t-1})} \right\}$.
2. Let $j_t = \arg\min_{j \in U \setminus S^{t-1}} \left\{ \frac{c_j}{\varrho_j(S^{t-1})} \right\}$.
3. Let $\varrho_t = \varrho_{j_t}(S^{t-1})$.
4. Set $S^t = S^{t-1} \cup \{j_t\}$ and $\sigma_t = f(S^t) - f(S^{t-1})$.

If $f(S^t) = f(U)$ then set $\tau = t$ and stop. Otherwise, set $t = t + 1$ and repeat the iteration.

Table 1: Greedy heuristic for (Q)

Then, S^τ is a *greedy heuristic* solution with value $Z^G = \sum_{j \in S^\tau} c_j$. The following theorem holds.

Theorem 3.2. *If the greedy algorithm is applied to (Q), then*

(i) $\frac{Z^G}{Z} \leq 1 + \ln \left(\max_{j,r} \left\{ \frac{\varrho_j(S^0)}{\varrho_j(S^r)} : \ \varrho_j(S^r) > 0 \right\} \right)$,

(ii) $\frac{Z^G}{Z} \leq 1 + \ln \frac{\theta^\tau}{\theta^1}$,

(iii) $\frac{Z^G}{Z} \leq 1 + \ln \left(\frac{f(U) - f(\emptyset)}{f(U) - f(S^{\tau-1})} \right)$, *if f is integer valued,*

(iv) $\frac{Z^G}{Z} \leq H(\max_j f(\{j\}) - f(\emptyset))$, *where $H(d) = \sum_{i=1}^d \frac{1}{i}$ for a positive integer d.*

In order to prove Theorem 3.2, we will need the following preliminary result.

Proposition 3.3. *Let $0 < u_1 \leq u_2 \leq \ldots \leq u_n$, and $x_1 \geq x_2 \geq \ldots \geq x_n > 0$. Let $S = \sum_{i=1}^{n-1} u_i(x_i - x_{i+1}) + u_n x_n = u_1 x_1 + \sum_{i=1}^{n-1}(u_{i+1} - u_i)x_{i+1}$. Then*

$$S \leq (\max_i u_i x_i) \left[1 + \ln \min \left(\frac{x_1}{x_n}, \frac{u_n}{u_1} \right) \right].$$

Moreover,
if $\{x_i\}_{i=1}^n$ are integers, then $S \leq (\max_i u_i x_i) H(x_1)$, and
if $\{u_i\}_{i=1}^n$ are integers, then $S \leq (\max_i u_i x_i) H(u_n)$.

Proof. Since $u_i, \ x_i > 0$ for all $i = 1, \ldots, n$ we have that

$$u_i x_i \leq \max_j u_j x_j \iff u_i \leq (\max_j \ u_j x_j) \frac{1}{x_i} \ ,$$

thus we get that

$$\begin{aligned} S = \sum_{i=1}^{n-1} u_i(x_i - x_{i+1}) + u_n x_n &\leq (\max_j u_j x_j)(x_i - x_{i+1}) + (\max_j u_j x_j) \\ &= (\max_j u_j x_j)\left[\sum_{i=1}^{n-1}\left(1 - \frac{x_{i+1}}{x_i}\right) + 1\right]. \end{aligned} \tag{3.2}$$

Using the fact that $1 - \frac{1}{x} \leq \ln x$, for all $x \geq 1$ and $\frac{x_i}{x_{i+1}} \geq 1$, we get that

$$\begin{aligned} (\max_j u_j x_j)\left[1 + \ln \frac{x_1}{x_n}\right] &= (\max_j u_j x_j)\left[1 + \sum_{i=1}^{n-1}\left(\ln \frac{x_i}{x_{i+1}}\right)\right] \\ &\geq (\max_j u_j x_j)\left[1 + \sum_{i=1}^{n-1}\left(1 - \frac{x_{i+1}}{x_i}\right)\right] \geq S, \end{aligned}$$

which we get from equation (3.2). If $\{x_i\}_{i=1}^n$ are integers, then $1 - \frac{x_{i+1}}{x_i} \leq \frac{1}{x_{i+1}+1} + \frac{1}{x_{i+1}+2} + \ldots + \frac{1}{x_i}$, since $x_i \geq x_{i+1} \geq 1$ for all $i \in \{1, \ldots, n-1\}$ [1], we get that

$$\begin{aligned} S &\leq (\max_j u_j x_j)\left[\sum_{i=1}^{n-1}\left(1 - \frac{x_{i+1}}{x_i}\right) + 1\right] \\ &\leq (\max_j u_j x_j)\left[\sum_{i=1}^{n-1}\left(\frac{1}{x_{i+1}+1} + \frac{1}{x_{i+1}+2} + \ldots + \frac{1}{x_i}\right) + 1\right] \\ &= (\max_j u j_x j)\left[\frac{1}{x_n+1} + \ldots + \frac{1}{x_{n-1}} + \frac{1}{x_{n-1}+1} + \ldots + \frac{1}{x_{n-2}} + \ldots + \frac{1}{x_1} + 1\right] \\ &= (\max_j u_j x_j) H(x_1). \end{aligned} \tag{3.3}$$

Analogously, we will get that $S \leq (\max_j u_j x_j)[1 + \ln \frac{u_n}{u_1}]$ and $S \leq (\max_j u_j x_j) H(u_n)$, using the fact that $x_i \leq (\max_j u_j x_j)\frac{1}{u_j}$ which concludes the proof. □

Let $k_1 = \max_{j,r}\left\{\frac{\varrho_j(S^0)}{\varrho_j(S^r)} :\ \varrho_j(S^r) > 0\right\}$; $\quad k_2 = \frac{\theta^\tau}{\theta^1}$; $\quad k_3 = \frac{f(U) - f(\emptyset)}{f(U) - f(S^{\tau-1})}$.

Proof of Theorem 3.2. Consider the following linear programming relaxation of (Q^I):

$$Z^L = \min \sum_{j \in U} c_j x_j \tag{Q^L}$$

$$s.t. \quad \sum_{j \in U} \varrho_j(S^t) x_j \geq f(U) - f(S^t) \text{ for all } t \in \{0, \ldots, \tau - 1\}$$

$$x_j \geq 0 \text{ for all } j \in U.$$

[1]Let $x_i - x_{i+1} = k$, now since $x_1 \geq x_2 \geq \ldots \geq x_n$ we get $\frac{x_i}{x_{i+1}+1} + \frac{x_i}{x_{i+1}+2} + \ldots + \frac{x_i}{x_{i+1}+k-1} + 1 \geq k = x_i - x_{i+1}$ (since $\frac{x_i}{x_{i+1}+j} \geq 1$ for all $j = 1, \ldots, k-1$) and that is equivalent to $\frac{1}{x_{i+1}+1} + \frac{1}{x_{i+1}+2} + \ldots + \frac{1}{x_i} \geq 1 - \frac{x_{i+1}}{x_i}$.

We would like to find a lower bound on Z and to do so we will look for appropriate dual feasible solutions for (Q^L).
The dual of (Q^L) is the following:

$$Z_D^L = \max \sum_{t=0}^{\tau-1} (f(U) - f(S^t))y_t \tag{Q^{LD}}$$

$$s.t. \quad \sum_{t=0}^{\tau-1} \varrho_j(S^t)y_t \le c_j \textit{ for all } j \in U$$

$$y_t \ge 0 \textit{ for all } t = 0, \dots, \tau - 1 \ .$$

(i) and (ii). Let $\theta^* = (\theta^1, \theta^2 - \theta^1, \dots, \theta^\tau - \theta^{\tau-1})$. Since for an arbitrary j, there exists $r \le \tau$ such that $\varrho_j(S^{r-1}) > 0$ and $\varrho_j(S^r) = 0$, we can apply Proposition 3.3 with $0 < \theta^1 \le \dots \le \theta^r$ and $\varrho_j(S^0) \ge \dots \ge \varrho_j(S^{r-1}) > 0$ [2], to get that

$$\theta^1 \varrho_j(S^0) + (\theta^2 - \theta^1)\varrho_j(S^1) + \dots + (\theta^r - \theta^{r-1})\varrho_j(S^{r-1}) \le$$
$$\le \left\{ \max_{t=1,\dots,r} \theta^t \varrho_j(S^{t-1}) \right\} \left[1 + \ln \min_{j,r} \left\{ \frac{\varrho_j(S^0)}{\varrho_j(S^{r-1})}, \frac{\theta^r}{\theta^1} \right\} \right] \le c_j \left[1 + \ln(\min\{k_1, k_2\})\right],$$

while we get that $\theta^t \varrho_j(S^{t-1}) \le c_j$ as a consequence of the greedy heuristic.
Hence, $(1 + \ln \min\{k_1, k_2\})^{-1}\theta^*$ is dual feasible for (Q^L) and therefore

$$(1 + \ln \min\{k_1, k_2\})^{-1}[\theta^1(f(U) - f(S^0)) +$$
$$+ (\theta^2 - \theta^1)(f(U) - f(S^2)) + \dots + (\theta^\tau - \theta^{\tau-1})(f(U) - f(S^{\tau-1}))] \le Z^L \le Z.$$

On the other hand,

$$Z^G = \sum_{t=1}^{\tau} \theta^t (f(S^t) - f(S^{t-1})) = \theta^1(f(U) - f(S^0)) + \sum_{t=2}^{\tau} (\theta^t - \theta^{t-1})(f(U) - f(S^{t-1})),$$

thus, $Z^G \le Z \cdot (1 + \ln \min\{k_1, k_2\})$.
(iii) Let us define $u^t \in \mathbb{R}^\tau$ by $u_i^t = \theta^t$ if $i = t$ and $u_i^t = 0$ otherwise for all $i \in \{1, \dots, \tau\}$. Then,

$$u^t(\varrho_j(S^0), \dots, \varrho_j(S^{\tau-1})) = \theta^t \varrho_j(S^{t-1}) \le c_j,$$

and hence u^t is dual feasible for $t = 1, \dots, \tau$. It follows that

$$\max_{t=1,\dots,\tau} u^t(f(U) - f(S^0), \dots, f(U) - f(S^{\tau-1})) = \max_{t=1,\dots,\tau} \theta^t(f(U) - f(S^{t-1})) \le Z^L \le Z.$$

Applying Proposition 3.3 with $0 < \theta^1 \le \dots \le \theta^\tau$, and $f(U) - f(S^0) \ge f(U) - f(S^1) \ge \dots \ge f(U) - f(S^{\tau-1})$ gives

$$Z^G = \sum_{t=1}^{\tau-1} \theta^t(f(S^t) - f(S^{t-1})) + \theta^\tau(f(U) - f(S^{\tau-1})) \le$$
$$\le \max_t \{\theta^t(f(U) - f(S^{t-1}))\} \left[1 + \ln \frac{f(U) - f(S^0)}{f(U) - f(S^{\tau-1})} \right] \le Z \cdot (1 + \ln k_3).$$

[2] $0 < \theta^1 \le \dots \le \theta^r$ holds, since $\min_{j \in U \setminus S^{k-2}} \frac{c_j}{\varrho_j(S^{k-2})} \le \min_{j \in U \setminus S^{k-1}} \frac{c_j}{\varrho_j(S^{k-1})}$ for all $k = 2, \dots, \tau$ and $\varrho_j(S^0) \ge \dots \ge \varrho_j(S^{r-1}) > 0$ holds because of the submodularity of f.

(iv) If f is integer-valued, $\varrho_j(S^t)$ is integer-valued as well for all j and t, and from Proposition 3.3, we obtain

$$\theta^1\varrho_j(S^0) + \ldots + (\theta^r - \theta^{r-1})\varrho_j(S^{r-1}) \leq c_j H(\max_j \varrho_j(S^0)).$$

The rest of the proof is same as that of *(i)* and *(ii)* above. □

Corollary 3.4. *Given $p \geq 1$ matroids, for the problem of finding a minimum weight set that is a spanning set in each of p matroids, there exists a greedy heuristic for which $Z^G/Z \leq H(p)$.*

Proof. For $i = 1, \ldots, p$, let r_i be the rank function of matroid i. Take $f = \sum_{i=1}^{p} r_i$. Since f is a sum of submodular functions r_i, hence submodular (by Proposition 2.3), we can apply the greedy heuristic to the resulting problem (Q). As $f(S) = f(U)$ only if $r_i(S) = r_i(U)$ for all i, the result follows from Theorem 3.2. □

4 Approximating Vector Domination and Related Problems

This section follows Cicalese et al. [8] and all the definitions and theorems are from [8] unless stated otherwise.

4.1 Problem Definitions

Let us now formally introduce dominating sets and total dominating sets to define the Vector Domination problem as well as the related total vector domination and multiple domination problems.

Definition 4.1. A *dominating set* in a graph $G = (V, E)$ is a subset S of the vertex set V such that every vertex not in the set S has a neighbor in it. Equivalently, for every $v \in V$, we have $S \cap N[v] \neq \emptyset$.

Definition 4.2. A *total dominating set* in $G = (V, E)$ is a subset $S \subseteq V$ such that every vertex of the graph has a neighbor in S, that is, for every $v \in V$ there exists a vertex $u \in S$ such that $uv \in E$. Equivalently, for every $v \in V$, we have $S \cap N(v) \neq \emptyset$.

Remark 4.3. Only graphs without isolated vertices have total dominating sets.

The corresponding minimization problems can be formally defined as follows.

DOMINATION	
Input:	A graph $G = (V, E)$.
Task:	Find a dominating set of minimum size.

TOTAL DOMINATION	
Input:	A graph $G = (V, E)$.
Task:	Find a total dominating set of minimum size.

Definition 4.4. A *vector dominating set* in a graph $G = (V, E)$ with the requirement vector $\mathbf{k} = (k_v : v \in V)$ with $k_v \in \{0, 1, \ldots, d_G(v)\}$ for all $v \in V$, is a subset S of the vertex set V of G such that every vertex not in S has at least k_v neighbors in S.

Definition 4.5. A *total vector dominating set* in a graph $G = (V, E)$ with the requirement vector $\mathbf{k} = (k_v : v \in V)$ with $k_v \in \{0, 1, \ldots, d_G(v)\}$ for all $v \in V$, is a subset S of the vertex set V of G such that every vertex v of $V(G)$ has at least k_v neighbors in S.

We will use the same definition as in the introduction for the *Vector Domination* problem :

VECTOR DOMINATION	
Input:	A graph $G = (V, E)$ and a vector $\mathbf{k} = (k_v : v \in V)$ with $k_v \in \{0, 1, \ldots, d_G(v)\}$ for all $v \in V$.
Task:	Find a vector dominating set of minimum size, that is, a set $S \subseteq V$ minimizing $\|S\|$ such that $\|S \cap N(v)\| \geq k_v$ for all $v \in V \setminus S$.

The *total vector domination* problem is defined analogously:

TOTAL VECTOR DOMINATION	
Input:	A graph $G = (V, E)$ and a vector $\mathbf{k} = (k_v : v \in V)$ with $k_v \in \{0, 1, \ldots, d_G(v)\}$ for all $v \in V$.
Task:	Find a total vector dominating set of minimum size, that is, a set $S \subseteq V$ minimizing $\|S\|$ and such that $\|S \cap N(v)\| \geq k_v$ for all $v \in V$.

If in the definition of the total vector domination we replace open neighborhoods with closed ones, we get the so called *multiple domination problem*:

MULTIPLE DOMINATION	
Input:	A graph $G = (V, E)$ and a vector $\mathbf{k} = (k_v : v \in V)$ with $k_v \in \{0, 1, \ldots, d_G(v)\}$ for all $v \in V$.
Task:	Find a set $S \subseteq V$ minimizing $\|S\|$ and such that $\|S \cap N[v]\| \geq k_v$ for all $v \in V$.

We will also look at the following special cases of Vector Domination, total vector domination, and multiple domination:

- For $0 < \alpha \leq 1$, an *α-dominating set* in G is a subset $S \subseteq V$ such that every vertex not in the set has at least an α-fraction of its neighbors in the set, that is, for all $v \in V \setminus S$ it holds that $|N(v) \cap S| \geq \alpha |N(v)|$.

- For $0 < \alpha \leq 1$, a *total α-dominating set* in G is a subset $S \subseteq V$ such that every vertex has at least an α-fraction of its neighbors in the set, that is, for all $v \in V$ it holds that $|N(v) \cap S| \geq \alpha |N(v)|$.

- For $0 < \alpha \leq 1$, an *α-rate dominating set* in G is a subset $S \subseteq V$ such that every vertex has at least an α-fraction of the members of its closed neighborhood in the set, that is, for all $v \in V$, it holds that $|N[v] \cap S| \geq \alpha|N[v]|$.

Notice that for every $\alpha > 0$, every α-dominating set is a dominating set, every total α-dominating set is a total dominating set, every vertex cover is an α-dominating set and every 1-dominating set is a vertex cover.
In the next subsections we will see some approximability and inapproximability results for the related optimization problems.

4.2 Approximability results

In this section, we will show that Vector Domination and total vector domination can be approximated in polynomial time by a factor of $\ln(2\Delta(G)) + 1$ and $\ln(\Delta(G)) + 1$, respectively.
The results will be based on results for the submodular set covering problem[1]. Let us consider the following generalization of the set covering problem:

SET MULTICOVERING	
Instance:	A set system $\mathcal{C} = (U, \mathcal{F})$, where U is a finite ground set and $\mathcal{F}$ is a collection of subsets of U; a non-negative integer requirement $req(u)$ for every element u of the ground set.
Taks:	Find a minimum size subcollection $\mathcal{F}' \subseteq \mathcal{F}$ such that every element u of the ground set appears in at least $req(u)$ sets in $\mathcal{F}'$.

Every instance $(G, \mathbf{k})$ of the total vector domination problem can be described as an instance of the *Set Multicovering* problem in the following way: Take $U = V(G)$ and define $\mathcal{F}$ to be the collection of all (open) neighborhoods of G. Set $req(u) = k_u$ for all $u \in U$. It is clear that a subset $S \subseteq V(G)$ is a total vector dominating set for $(G, \mathbf{k})$ if and only if the collection $(N(v) : v \in S)$ is a feasible solution to the instance $(U, \mathcal{F}, req)$ of the *Set Multicovering* problem. Suppose that $U = \{u_1, \ldots, u_n\}$ and $\mathcal{F} = \{U_1, \ldots, U_m\}$. Then the *Set Multicovering* problem can be written as an integer linear program in the following way:

$$\min \sum_{j=1}^{m} x_j$$

[1]We will consider the unweighted submodular set covering problem, from Section 3, which is a special case of the submodular set covering problem in which all weights are unit: $c_j = 1$ for all j.

$$s.t \sum_{j:u_i \in U_j} x_j \geq req(u_i) \textit{ for all } u_i \in U$$

$$x_j \in \{0,1\} \textit{ for all j=1,\ldots, m.}$$

The *Set Multicovering* problem is a special case of the integer covering problem (C). Namely, if we take $c_j = 1$ for all $j = 1, \ldots, m$, $b_i = req(u_i)$ and define a matrix A with entries

$$a_{ij} = \begin{cases} 1 & \text{if } u_i \in U_j \\ 0 & \text{otherwise,} \end{cases}$$

we get the formulation of *Set Multicovering* problem as an integer covering problem. Since this is a special case of (Q), we obtain the following theorem and corollary, by part (*iv*) of Theorem 3.2 (or more directly from Dobson [11]).

Theorem 4.6. *Total vector domination can be approximated in* $\mathcal{O}(|V(G)|^4)$ *time by a factor of* $\ln(\Delta(G)) + 1$.

Proof. As already showed above, total vector domination can be transformed to an instance of Set Multicovering which, in turn, is a special case of the Submodular Set Covering problem. This was shown in Section 3.1 by observing that the function $f : \mathcal{P}(V(G)) \to \mathbb{R}$, where $(G, \mathbf{k})$ is the input for total vector domination, given by $f(S) = \sum_{i \in V(G)} \min\{\sum_{j \in S} a_{ij}, k_j\}$ for all $S \subseteq V(G)$, where $a_{ij} = 1$ if and only if $i \in N_G(j)$, is submodular. Therefore, by part (*iv*) of Theorem 3.2, the greedy algorithm approximates total vector domination by a factor of $H(\max \sum_{i \in V(G)} a_{ij}) = H(\Delta(G)) \leq \ln(\Delta(G)) + 1$. Moreover, note that the function f as defined as above can be easily evaluated in $\mathcal{O}(|V(G)|^2)$ time and consequently the greedy algorithm runs in $\mathcal{O}(|V(G)|^4)$ time. □

Corollary 4.7. *For each* $\alpha \in (0, 1]$, *total* α*-domination can be approximated in* $\mathcal{O}(|V(G)|^4)$ *time by a factor of* $\ln(\Delta(G)) + 1$.

Proof. Immediately from Theorem 4.6, using the fact that if G is an input instance for total α-domination, then $(G, \mathbf{k})$ is an equivalent input instance for total vector domination provided $k_v = \lceil \alpha \cdot d_G(v) \rceil$ for all $v \in V(G)$. Since total vector domination was approximated in $\mathcal{O}(|V(G)|^4)$, the total α-domination is approximated in $\mathcal{O}(|V(G)|^4)$ time as well. □

A similar reduction as for the total vector domination, except that the family of open neighborhoods is replaced by the family of closed neighborhoods, shows that the Set Multicovering problem is also a generalization of the multiple domination problem. We therefore obtain the following results.

Theorem 4.8. *The multiple domination problem can be approximated in* $\mathcal{O}(|V(G)|^4)$ *time by a factor of* $\ln(\Delta(G)+1)+1$.

Corollary 4.9. *For each* $\alpha \in (0,1]$, α*-rate domination can be approximated in* $\mathcal{O}(|V(G)|^4)$ *time by a factor of* $\ln(\Delta(G)+1)+1$.

It is not clear whether Vector Domination can be expressed as a special case of Set Multicovering. Nevertheless, Theorem 3.2 applies.

Theorem 4.10. *Vector Domination can be approximated in* $\mathcal{O}(|V(G)|^4)$ *time by a factor of* $\ln(2\Delta(G))+1$.

Proof. For a graph $G=(V,E)$ and a vector $\mathbf{k}=(k_v : v \in V)$ such that $k_v \in \{0,1,\ldots,d(v)\}$ for all $v \in V$, we define a function $f : \mathcal{P}(V) \to \mathbb{N}$, as follows:

$$\begin{aligned} f(S) &= \sum_{v \in V} \tau_v(S), \quad \textit{where } S \subseteq V, \textit{ and} \\ \tau_v(S) &= \begin{cases} \min\{|S \cap N(v)|, k_v\} & \textit{if } v \notin S; \\ k_v & \textit{if } v \in S. \end{cases} \end{aligned} \tag{4.1}$$

The following properties of f can be verified:

(*i*) f is integer valued. Since k_v is integer valued, $\tau_v(S)$ is integer valued as well, hence, f is also integer valued.

(*ii*) $f(\emptyset)=0$.

$$f(\emptyset) = \sum_{v \in V} \tau_v(\emptyset) = \sum_{v \in V} \min\{|\emptyset \cap N(v)|, k_v\} = \sum_{v \in V} \min\{0, k_v\} = 0.$$

(*iii*) f is non-decreasing. Let $S \subseteq T \subseteq V$. Then

$$\begin{aligned} f(T) = \sum_{v \in V} \tau_v(T) &= \sum_{v \in T} k_v + \sum_{v \in V \setminus T} \min\{|T \cap N(v)|, k_v\} = \\ &= \sum_{v \in S} k_v + \sum_{v \in T \setminus S} k_v + \sum_{v \in V \setminus T} \min\{|T \cap N(v)|, k_v\} \geq \\ &\geq \sum_{v \in S} k_v + \sum_{v \in T \setminus S} \min\{|S \cap N(v)|, k_v\} + \sum_{v \in V \setminus T} \min\{|S \cap N(v)|, k_v\} = \\ &= \sum_{v \in S} k_v + \sum_{v \in V \setminus S} \min\{|S \cap N(v)|, k_v\} = \sum_{v \in V} \tau_v(S) = f(S). \end{aligned}$$

(*iv*) A set $S \subseteq V$ satisfies $f(S)=f(V)$ if and only if S is a vector dominating set.

Suppose first that $S \subseteq V$ satisfies $f(S) = f(V)$. We have that $f(S) = f(V)$ is equivalent, by (4.1), to $\sum_{v\in V} \tau_v(S) = \sum_{v\in V} \tau_v(V)$, which is, again by (4.1), equivalent to

$$\sum_{v\in V\setminus S} \min\{|S\cap N(v)|, k_v\} + \sum_{v\in S} k_v = \sum_{v\in V} k_v,$$

and that can be further reduced to

$$\sum_{v\in V\setminus S} \min\{|S\cap N(v)|, k_v\} = \sum_{v\in V\setminus S} k_v. \tag{4.2}$$

If there exists a $w \in V \setminus S$ such that $k_w > |S \cap N(v)|$, then from (4.2), we can deduce that

$$\sum_{v\in (V\setminus S)\setminus\{w\}} \min\{|S\cap N(v)|, k_v\} > \sum_{v\in (V\setminus S)\setminus\{w\}} k_v.$$

Since $k_v \geq \min\{|S \cap N(v)|, k_v\}$ for all $v \in (V \setminus S) \setminus \{w\}$, it follows that $\sum_{v\in(V\setminus S)\setminus\{w\}} k_v \geq \sum_{v\in(V\setminus S)\setminus\{w\}} \min\{|S \cap N(v)|, k_v\} > \sum_{v\in(V\setminus S)\setminus\{w\}} k_v$, which is not possible. Thus, we have that $|S\cap N(v)| \geq k_v$ for all $v \in V \setminus S$, hence, S is a vector dominating set.

Now, suppose that S is a vector dominating set. Then for all $v \in V \setminus S$, $|S \cap N(v)| \geq k_v$, which implies that $\min\{|S\cap N(v)|, k_v\} = k_v$ and hence

$$f(S) = \sum_{v\in V} \tau_v(S) = \sum_{v\in V} k_v = f(V).$$

(v) f is submodular.

The proof of (v) is given below.

Lemma 4.11. *The function $f : \mathcal{P}(V) \to \mathbb{N}$, given by (4.1), is submodular.*

Proof. To show that f is submodular, by Proposition 2.3, it suffices to show that all the functions $\tau_v(\cdot)$ are submodular, that is, by Proposition 2.2, we need that for all $S \subseteq T \subseteq V$ and for all $w \in V \setminus T$,

$$\tau_v(T \cup \{w\}) - \tau_v(T) \leq \tau_v(S \cup \{w\}) - \tau_v(S). \tag{4.3}$$

Observe that τ_v is non-decreasing.
Suppose first that $\tau_v(T) = k_v$. Then $\tau_v(T \cup \{w\}) = k_v$ and the left-hand side of inequality (4.3) is equal to 0. Hence inequality (4.1) holds since τ_v is non-decreasing.
From now on, we assume that $\tau_v(T) < k_v$, which implies $\tau_v(T) = |T \cap N_G(v)|$. Since τ_v is non-decreasing, $\tau_v(S) < k_v$, and hence $\tau_v(S) = |S\cap N_G(v)|$. Inequality (4.3) simplifies to

$$\tau_v(T) - \tau_v(S) = |(T \setminus S) \cap N_G(v)| \geq \tau_v(T \cup \{w\}) - \tau_v(S \cup \{w\}). \tag{4.4}$$

We may assume that $\tau_v(T \cup \{w\}) > \tau_v(S \cup \{w\})$, since otherwise the right-hand side of (4.4) equals 0, and inequality (4.4) holds.
Therefore, $\tau_v(S \cup \{w\}) < k_v$ implying $\tau_v(S \cup \{w\}) = |(S \cup \{w\}) \cap N_G(v)|$. If also $\tau_v(T \cup \{w\}) < k_v$ then $\tau_v(T \cup \{w\}) = |(T \cup \{w\}) \cap N_G(v)|$ and equality holds in (4.4). So we may assume that $\tau_v(T \cup \{w\}) = k_v$. Note that v does not belong to $T \cup \{w\}$ since otherwise either $\tau_v(T)$ or $\tau_v(S \cup \{w\})$ would equal to k_v. Suppose that inequality (4.4) fails. Then

$$|(T \setminus S) \cap N_G(v)| < k_v - |(S \cup \{w\}) \cap N_G(v)|,$$

which implies

$$|(T \cup \{w\}) \cap N_G(v)| < k_v.$$

However, together with the fact that $v \notin T \cup \{w\}$, this contradicts the assumption that $\tau_v(T \cup \{w\}) = k_v$. □

Back to proof of Theorem 4.10, by (iv) we have that an optimal solution to the vector dominating set is provided by a minimum size set S such that $f(S) = f(V)$. In other words, we have recast Vector Domination as a particular case of the *Minimum Submodular Set Covering* problem from Section 3.
Let $\mathbb{A}$ denote the natural greedy strategy which starts with $S = \emptyset$ and iteratively adds to S the element $v \in V \setminus S$ such that $f(S \cup \{v\}) - f(S)$ is maximum, until $f(S) = f(V)$ is achieved. By Theorem 3.2 (iv), it follows that algorithm $\mathbb{A}$ is a $(\ln(\max_{y \in V} f(\{y\})) + 1)$-approximation algorithm for Vector Domination. For every $y \in V$, we have $f(\{y\}) = \sum_{v \in V \setminus \{y\}} \tau_v(\{y\}) + \tau_y(\{y\}) \leq d(y) + k_y \leq 2d(y)$. Hence $\max_{y \in V} f(\{y\}) \leq 2\Delta(G)$, yielding the desired result. Since the value of $f(S)$ can be computed in $\mathcal{O}(|V(G)|^2)$ time, the greedy algorithm finds the set S in $\mathcal{O}(|V(G)|^4)$ time. □

Since α-domination is a special case of the Vector Domination problem, Theorem 4.10 implies the following result:

Corollary 4.12. *For each $\alpha \in (0, 1]$, α-domination can be approximated in $\mathcal{O}(|V(G)|^4)$ time by a factor of* $\ln(2\Delta(G)) + 1$.

4.3 Inapproximability results

We now present the best known inapproximability results for Vector Domination and related problems under the assumption that $P \neq NP$. Together with approximation ratios of greedy algorithms presented in Section 4.3, these results give a complete

answer to the question of how well these problems can be approximated in polynomial time, up to a constant multiplicative factor.
The following inapproximability result about domination and total domination is from Bonomo et al. [4, Theorem 6.5] (improving an analogous result under the stronger hypothesis that $NP \not\subseteq DTIME(n^{\mathcal{O}(\ln \ln n)})$ due to Chlebík and Chlebíkova [6]).

Theorem 4.13. *For every* $\epsilon > 0$ *there is no polynomial-time algorithm approximating domination or total domination on* n*-vertex graphs within a factor of* $(1-\epsilon)\ln n$*, unless* $P = NP$.

Since Vector Domination and multiple domination generalize domination and total vector domination generalizes total domination, Theorem 4.13 implies the following result.

Corollary 4.14. *For every* $\epsilon > 0$*, there is no polynomial-time algorithm approximating Vector Domination, total vector domination, or multiple domination within a factor of* $(1-\epsilon)\ln n$ *on* n*-vertex graphs, unless* $P = NP$.

The following inapproximability results for variants of domination are a consequence of [8, Theorem 12].

Theorem 4.15. *There exist a number* $c > 0.0755$ *such that for every* $\alpha \in (0, 1)$ *there is no polynomial-time algorithm approximating* α*-domination, total* α*-domination, or* α*-rate domination within a factor of* $c \cdot \ln n$ *on* n*-vertex graphs, unless* $P = NP$.

The above theorem was derived using a similar inapproximability result for the set covering problem due to Alon et al. [1]. This result was further improved by Dinur and Steurer [10]. In turn, this immediately leads to the following improvement of Theorem 4.15.

Theorem 4.16. *Unless P=NP, the following holds:*

- *For every* $\alpha \in (0, 1)$ *and every* $\epsilon > 0$*, there is no polynomial-time algorithm approximating* α*-domination within a factor of* $(\frac{1}{2} - \epsilon)\ln n$ *on* n*-vertex graphs.*
- *For every* $\alpha \in (0, 1)$ *and every* $\epsilon > 0$*, there is no polynomial-time algorithm approximating total* α*-domination or* α*-rate domination within a factor of* $(\frac{1}{3} - \epsilon)\ln n$ *on* n*-vertex graphs.*

The result of Dinur and Steurer was also instrumental for the derivation of Theorem 4.16. Further related results can be found in [4, Section 6.1]

5 Approximating Vector Connectivity

In this section, we will state the formal definition of Vector Connectivity, following Boros et al. who introduced the problem in [5], and discuss the greedy approximation of the problem, which we will use in Section 6.

5.1 Problem definition

Definition 5.1. Given a graph $G = (V, E)$, a set $S \subseteq V$, and a vertex $v \in V \setminus S$, a v, S-*fan of order* k is a collection of k paths $P_1, \ldots, P_k$ such that

(1) every P_i is a path connecting v to a vertex of S,

(2) the paths are pairwise vertex-disjoint except at v, i.e., $V(P_i) \cap V(P_j) = \{v\}$ holds for all $1 \leq i < j \leq k$.

Let $A \subseteq V$, and let $v \in V$, then we denote by $\sigma(v, A)$ the maximum order of a v, A-fan in G.

Definition 5.2. Given an integer-valued vector $\mathbf{k} = (k_v : v \in V)$, a *Vector Connectivity set* for $(G, \mathbf{k})$ is a set $S \subseteq V$ such that for every $v \in V \setminus S$, there exists a v, S-fan of order k_v. We say that k_v is the *requirement* of vertex v.

Definition 5.3. Given an undirected simple graph $G = (V, E)$ and a vector $\mathbf{k}$ indexed by vertices of G, such that $\mathbf{k} = (k_v : v \in V)$ and k_v is between 0 and the degree of vertex v of G, $d(v)$, the Vector Connectivity problem is to find a Vector Connectivity set for $(G, \mathbf{k})$ of minimum size.

Formally, the problem is stated as follows:

Vector Connectivity	
Input:	A graph $G = (V, E)$ and a vector $\mathbf{k} = (k_v : v \in V) \in \mathbb{Z}_+^n$ with $k_v \in \{0, 1, \ldots, d_G(v)\}$ for all $v \in V$.
Task:	Find a minimum size Vector Connectivity set for $(G, \mathbf{k})$.

Definition 5.4. For every $v \in V$ and every set $S \subseteq V \backslash \{v\}$, we say that v is *k-connected* to S if there is a v, S-fan of order k in G.

Hence, given an instance $(G, \mathbf{k})$ of *Vector Connectivity*, a set $S \subseteq V$ is a Vector Connectivity set for $(G, \mathbf{k})$ if and only if every $v \in V \setminus S$ is k_v-connected to S.

5.2 A polynomial-time approximation algorithm

By showing that *Vector Connectivity* can be recast as a particular case of the *Minimum Submodular Set Covering* problem, which we saw in Section 3, we will conclude that *Vector Connectivity* can be approximated in polynomial time by a factor of $\ln n + 2$ on n-vertex graphs.
Recall that an instance of the *Minimum Submodular Set Covering* problem consists of a finite set U and an integer-valued, non-decreasing submodular function $f \ : \mathcal{P}(U) \to \mathbb{Z}_+$, and the objective will be to find a set $S \subseteq U$ of minimum cardinality such that $f(S) = f(U)$. (We can ignore the weights if we take them to be $c_j = 1$ for all $j \in U$.)
For any instance $(G = (V, E), \mathbf{k})$ of *Vector Connectivity*, we will define a function $f : \ \mathcal{P}(V) \to \mathbb{Z}_+$ as follows:

$$\begin{aligned} f(X) &= \sum_{v \in V} f_v(X) \textit{ where } X \subseteq V, \textit{ and} \qquad (5.1) \\ f_v(X) &= \begin{cases} \min\{\sigma(v, X), k_v\} & \textit{if } v \notin X \\ k_v & \textit{if } v \in X. \end{cases} \end{aligned}$$

Proposition 5.5 (Boros et al. [5])**.** *A set $S \subseteq V$ satisfies $f(S) = f(V)$ if and only if S is a Vector Connectivity set for $(G, \boldsymbol{k})$.*

Proof. Suppose first that $S \subseteq V$ satisfies $f(S) = f(V)$. Then we have that $f(S) = f(V)$ is equivalent, by (5.1), to $\sum_{v \in V} f_v(S) = \sum_{v \in V} f_v(V)$, which is, again by (5.1), equivalent to

$$\sum_{v \in V \backslash S} \min\{\sigma(v, S), k_v\} + \sum_{v \in S} k_v = \sum_{v \in V} k_v,$$

and that can be further reduced to

$$\sum_{v \in V \backslash S} \min\{\sigma(v, S), k_v\} = \sum_{v \in V \backslash S} k_v. \qquad (5.2)$$

If there exists a $w \in V \setminus S$ such that $k_w > \sigma(w, S)$, then from (5.2) we can deduce that

$$\sum_{v \in (V \backslash S) \backslash \{w\}} \min\{\sigma(v, S), k_v\} > \sum_{v \in (V \backslash S) \backslash \{w\}} k_v.$$

Since $k_v \geq \min\{\sigma(v,S), k_v\}$ for all $v \in (V \setminus S) \setminus \{w\}$, it follows that $\sum_{v\in(V\setminus S)\setminus\{w\}} k_v \geq \sum_{v\in(V\setminus S)\setminus\{w\}} \min\{\sigma(v,S), k_v\} > \sum_{v\in(V\setminus S)\setminus\{w\}} k_v$, which is not possible. Thus, we have that $\sigma(v,S) \geq k_v$ for all $v \in V \setminus S$, hence, there is a v, S-fan of order k_v for all $v \in V \setminus S$ which implies that S is a Vector Connectivity set for $(G, \mathbf{k})$.
Now, suppose that S is a Vector Connectivity set for $(G, \mathbf{k})$. Then for all $v \in V \setminus S$, there exists a v, S-fan of order k_v. Thus, $\sigma(v,S) \geq k_v$ for all $v \in V \setminus S$, which implies:

$$f(S) = \sum_{v\in V} f_v(S) = \sum_{v\in V} k_v = f(V). \qquad \square$$

Consequently, Lemma 5.6 below implies that *Vector Connectivity* is a special case of *Minimum Submodular Set Cover.*

Lemma 5.6 (Boros et al. [5])**.** *Let* $(G = (V,E), \boldsymbol{k})$ *be an instance of* Vector Connectivity. *Then the function* $f : \mathcal{P}(V) \to \mathbb{Z}_+$*, given by (5.1), satisfies the following properties:*

(*i*) $f(\emptyset) = 0$.

(*ii*) f *is integer valued.*

(*iii*) f *is non-decreasing.*

(*iv*) f *is submodular.*

Proof. It is easy to verify that properties (*i*), (*ii*), and (*iii*) hold: Since $f_v(X) \in \mathbb{Z}$ for all $v \in V$ and for all $X \subseteq V$, we have that f is integer valued. Suppose that $X \subseteq Y \subseteq V$. Then $f(Y) = \sum_{v\in V} f_v(Y) = \sum_{v\in Y} k_v + \sum_{v\in V\setminus Y} \min\{\sigma(v,Y), k_v\} = \sum_{v\in X} k_v + \sum_{v\in Y\setminus X} k_v + \sum_{v\in V\setminus Y} \min\{\sigma(v,Y), k_v\} \geq \sum_{v\in V} f_v(X) = f(X)$, hence, f is non-decreasing.
In order to show that f is submodular, by Proposition 2.3, it suffices to show that the functions $f_v(\cdot)$ are submodular for all $v \in V$. Let $v \in V$ be an arbitrary vertex, and let $C := \max\{\Delta(G), \max_{v\in V(G)}\{k_v\}\} + 1$. We define a function $g_v : \mathcal{P}(V) \to \mathbb{Z}_+$ as follows:

$$g_v(X) = \begin{cases} \sigma(v,X) & if v \notin X \\ C & if v \in X. \end{cases}$$

It is easy to verify that g_v is non-decreasing. Let $S \subseteq T \subseteq V$. Then, if $v \in S$, we get that $g_v(T) = C = g_v(S)$. If $v \in T \setminus S$, then, since the maximal size of a v, S-fan is $\Delta(G)$, we get that $g_v(T) = C \geq \sigma(v,S) = g_v(S)$, and lastly, if $v \in V \setminus T$, we have that $b_v(T) = \sigma(v,T) \geq \sigma(v,S) = g_v(S)$ which proves that g_v is non-decreasing for all $v \in V$.

Moreover, we have $f_v(X) = \min\{g_v(X), k_v\}$ for every $X \subseteq V$. Therefore, in order to prove the submodularity of f_v, it suffices to prove that g_v is submodular (see, e.g., [28]), which is equivalent to (see Proposition 2.2) proving that for all $X \subseteq Y \subseteq V$ and for all $w \in V \setminus Y$,

$$g_v(Y \cup \{w\}) - g_v(Y) \leq g_v(X \cup \{w\}) - g_v(X). \tag{5.3}$$

If $v \in Y$, then $g_v(Y) = g_v(Y \cup \{w\}) = C$ and (5.3) holds, since g_v is non-decreasing. Similarly, if $w = v$, then $g_v(Y \cup \{w\}) = g_v(X \cup \{w\}) = C$, and inequality (5.3) holds, since g_v is non-decreasing.

Now, suppose that $w \notin Y \cup \{v\}$. Since $X \subseteq Y$, we also have that $w \notin X \cup \{v\}$. In this case, inequality (5.3) simplifies to

$$\sigma(v, Y \cup \{w\}) - \sigma(v, Y) \leq \sigma(v, X \cup \{w\}) - \sigma(v, X). \tag{5.4}$$

In order to show that inequality (5.4) holds, it suffices to prove that the function $h_v : \mathcal{P}(V \setminus \{v\}) \to \mathbb{Z}_+$, defined by $h_v(W) = \sigma(v, W)$ for all $W \subseteq V \setminus \{v\}$, is submodular. Consider the gammoid Γ derived from the triple $(D, V \setminus \{v\}, N_G(v))$ where D is a digraph obtained from G by replacing each edge with a pair of oppositely directed arcs. By Lemma 2.6, Γ is a matroid. It follows directly from the definition that function h_v is equal to the rank function r_Γ of Γ. Therefore, by Lemma 2.5, the function h_v is submodular, which completes the proof of Lemma 5.6. □

Theorem 5.7 (Boros et al. [5])**.** Vector Connectivity *can be approximated in* $\mathcal{O}(n^6)$ *time within a factor of* $\ln n + 2$ *on* n*-vertex graphs.*

Proof. The proof follows [5] and adds the complexity analysis. Let $(G = (V, E), \mathbf{k})$ be an instance of *Vector Connectivity* with $|V| = n$ and $|E| = m$. From the definition of the function f, given by (5.1), it follows that a set $S \subseteq V$ satisfies $f(S) = f(V)$ if and only if S is a Vector Connectivity set for $(G, \mathbf{k})$. Hence, an optimal solution to the *Vector Connectivity* problem is provided by a minimum size subset $S \subseteq V$ such that $f(S) = f(V)$, i.e., by an optimal solution for the *Minimum Submodular Set Cover*. An approximation to such a set S can be found in the following way.

Let $\mathbb{A}$ denote the natural greedy strategy that starts with $S = \emptyset$ and iteratively adds to S the element $v \in V \setminus S$ such that $f(S \cup \{v\}) - f(S)$ is maximum, until $f(S) = f(V)$ is achieved. The maximum order of a v, S-fan can be computed in polynomial time using a standard reduction to the well-known *Maximum Flow* problem, see, e.g., [28] (see Example 5.8). The reduction is done in $\mathcal{O}(m + n)$ time and a maximum flow can be computed in $\mathcal{O}(mn)$ time (see Orlin [24]), thus, the maximum order of a v, S-fan can be calculated in $\mathcal{O}(mn)$ time and the function f can be calculated in $\mathcal{O}(mn^2)$ time. Therefore, the greedy strategy, can be implemented in $\mathcal{O}(mn^4) = \mathcal{O}(n^6)$ time. Moreover, by Theorem 3.2 (iv), since f satisfies the four properties listed in Lemma 5.6, algorithm $\mathbb{A}$ is an $H(\max_{y\in V} f(\{y\}) - f(\emptyset))$-approximation algorithm for *Minimum Submodular Set Cover*, and consequently for *Vector Connectivity.* For every $y \in V$, we have

$$f(\{y\}) = \sum_{v\in V\setminus\{y\}} f_v(\{y\}) + f_y(\{y\}) \le n - 1 + k_y \le n + \Delta(G).$$

Since $f(\emptyset) = 0$, this implies $\tau \le n + \Delta(G)$. Hence, algorithm $\mathbb{A}$ is an $H(n + \Delta(G))$-approximation algorithm for *Vector Connectivity.* Since $H(n) \le \ln n + 1$ for $n \ge 1$ (see Lemma 2.9), we can further bound the approximation ratio ρ of $\mathbb{A}$ from above as follows:

$$\rho \le H(n + \Delta(G)) \le \ln(n + \Delta(G)) + 1 \le \ln(2n) + 1 \le \ln n + \ln 2 + 1 \le \ln n + 2,$$

yielding the desired result. □

Example 5.8. Given a graph $G = (V, E)$ and a vertex $v \in V$ and a set S, we construct a digraph $D = (V_1, A)$ where $V_1 = V(G) \cup \{t\}$ for a new vertex t and $A = \{(x, y) : \{x, y\} \in E(G), y \ne v\} \cup \{(s, t) : s \in S\}$. Then for every $w \in V_1 \setminus \{v, t\}$ we add a vertex w', add an arc from w to w' and take each outgoing arc from w, say (w, z), and replace it with the arc (w', z) to make a new graph D'. Then, the maximum order of a v, S-fan in G equals the maximum value of the flow in the network obtained from D' by assigning unit capacity to each arc and taking v as the source and t as the sink. We can see an example of the transformation in Figure 5.2. Let $n = |V(G)|$ and $m = |E(G)|$. The transformation from G to D takes $\mathcal{O}(m + n)$ time, while the transformation from D to D' takes $\mathcal{O}(m + n)$ time as well. Hence, the whole transformation is done in $\mathcal{O}(m + n) + \mathcal{O}(m + n) = \mathcal{O}(m + n)$ time.

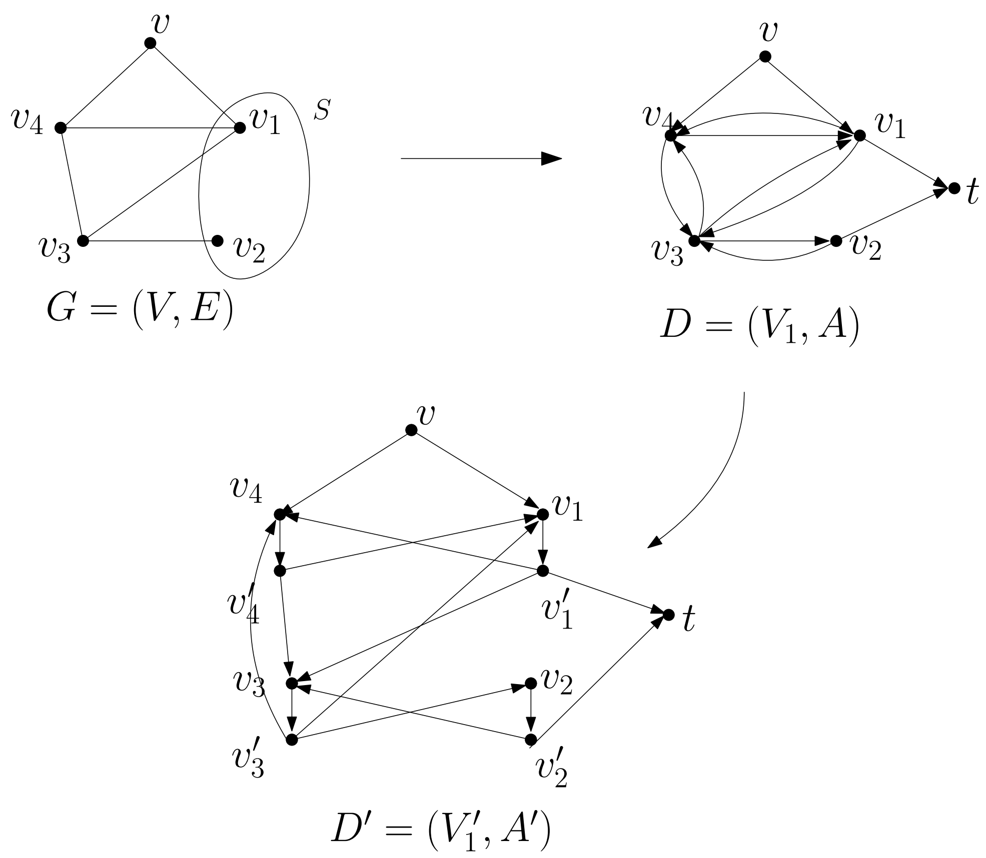

Figure 1: Reduction of a graph G to compute the maximum order of a v, S-fan

5.3 An inapproximability result

We now present the best known inapproximability result for vector connectivity, due to Cicalese et al. [9]. Unlike Vector Domination and related problems, it is not known whether there exists a positive number c such that the existence of a polynomial-time approximation algorithm for vector connectivity within a factor of $c \ln n$ on n-vertex graphs would imply $P = NP$. In fact, it is not even known whether any constant factor approximation for vector connectivity is possible in polynomial time, unless $P = NP$. However, it was shown by Cicalese et al. [9], that the problem cannot be approximated to an arbitrary precision, unless $P = NP$. This was established using the notion of APX-hardness. Since the definition of APX-hardness is somewhat technical and not necessary for our purposes we refer the interested reader, e.g., to [2].

Theorem 5.9 (Cicalese et al. [9])**.** Vector Connectivity *is APX-hard. In particular,* Vector Connectivity *admits no $PTAS$, unless $P = NP$.*

Using the fact that the maximum order of v, S-fan can be computed in polynomial

time using a reduction as in Example 5.8 shows that we can check in polynomial-time whether a given set $S \subseteq V(G)$ is a vector connectivity set fot $(G, \boldsymbol{k})$. Therefore, the decision version of Vector Connectivity is in NP. Moreover, Theorem 5.9 implies that Vector Connectivity is NP-hard.
In fact the problem is NP-complete in rather restricted graphs as the following theorem states.

Theorem 5.10 (Cicalese et al. [9])**.** *The decision version of the* Vector Connectivity *problem restricted to instances with maximum requirement* 4 *is NP-complete, even for:*

- *2-connected planar bipartite graphs of maximum degree 5 and girth at least k (for every fixed k),*
- *2-connected planar line graphs of maximum degree 5.*

6 Integer Programming Formulations

6.1 Vector domination problem

We can represent the Vector Domination problem as an integer linear problem in the following way. For a given graph $G = (V, E)$ and vertex requirement vector $\mathbf{k} = (k_v : v \in V)$, let us denote by x_v the decision variable expressing whether vertex v is in the dominating set or not. Then we can make the following integer linear program:

$$\min \sum_{v \in V} x_v \tag{D^I}$$

$$\begin{aligned} s.t. \quad & \sum_{u \in N(v)} x_u + k_v x_v \geq k_v \text{ for all } v \in V \\ & x_v \in \{0, 1\} \text{ for all } v \in V. \end{aligned}$$

Proposition 6.1. *Given an input $(G = (V, E), \boldsymbol{k})$ for the Vector Domination problem and a subset $S \subseteq V$, set S is vector dominating set for $(G, \boldsymbol{k})$ if and only if its characteristic vector x^S is feasible for (D^I). In particular, there is a bijective correspondence between minimum vector dominating set for $(G, \boldsymbol{k})$ and optimal solutions of (D^I).*

Proof. Suppose first that S is a vector dominating set for $(G, \mathbf{k})$. If $v \in S$ then $x_v^S = 1$ and we have $\sum_{u \in N(v)} x_u^S + k_v x_v^S \geq k_v x_v^S = k_v$. If $v \in V \setminus S$, then $x_v^S = 0$, however, since v has at least k_v neighbors in S, we have $\sum_{u \in N(v)} x_u^S \geq k_v$. Hence, S is a feasible solution for (D^I).

Suppose now that for some $S \subseteq V$, its characteristic vector x^S is feasible for (D^I). Let $v \in V \setminus S$. From the constraints, it follows that $\sum_{u \in N(v)} x_u^S + k_v x_v^S \geq k_v$ and, since $x_v^S = 0$, that is equivalent to $\sum_{u \in N(v)} x_u^S \geq k_v$. Hence, for every $v \in S \setminus V$, we have that v has at least k_v neighbors in S, which implies that S is a vector dominating set for $(G, \mathbf{k})$.

The above shows the existence of a bijective correspondence between vector dominating sets and feasible solutions for (D^I). Moreover, since the size of a vector dominating set S for $(G, \mathbf{k})$ is equal to the objective function value of (D^I) at its characteristic vector x^S, the second part of the proposition follows. $\square$

6.2 Vector connectivity problem

Let $G = (V, E)$ be a simple undirected graph and let $X \subseteq V$. Given a vertex requirements vector $\mathbf{k} = (k_v \in \mathbb{Z} : v \in V)$, if we take $\emptyset \neq X \subseteq V$, then we define $k(X) = \max_{x \in X} k_x$ and $N_G(X) = (\cup_{v \in X} N_G(v)) \setminus X$.
The proof of [9, Proposition 1] implies the following:

Proposition 6.2. *For every graph $G = (V, E)$, vertex requirement vector $\boldsymbol{k}$ and a set $S \subseteq V$, the following conditions are equivalent:*

(i) S is a vector connectivity set for $(G, \boldsymbol{k})$.

(ii) For every non-empty set $X \subseteq V$ such that $k(X) > |N_G(X)|$, we have $S \cap X \neq \emptyset$.

Proof. The proof follows [9]. Let S be a vector connectivity set for $(G, \boldsymbol{k})$. Suppose for a contradiction that there is a non-empty set $X \subseteq V$ such that $k(X) > |N_G(X)|$ and $S \cap X = \emptyset$. Let $C = N_G(X)$, and let $x \in X$ be a vertex such that $k_x > |C|$. Since $S \cap X = \emptyset$, we have $x \notin S$. Moreover, the definition of C implies that in the graph $G - C$, there is no path from x to S. Therefore, by Menger's Theorem [22], the maximum number of disjoint x, S-paths is at most $|C|$, which contradicts the fact that x is k_x-connected to S and $k_x > |C|$.
Now, suppose for a contradiction that $S \subseteq V$ is not a vector connectivity set for $(G, \boldsymbol{k})$, and suppose that for every non-empty set $X \subseteq V$ such that $|N_G(X)| < k(X)$, we have $S \cap X \neq \emptyset$. Since S is not a vector connectivity set for $(G, \boldsymbol{k})$, there exists a vertex $x \in V \setminus S$ such that x is not k_x-connected to S. By Menger's Theorem [22], there exists a set $C \subseteq V \setminus \{x\}$ such that $|C| < k_x$ and every path connecting x to S contains a vertex of C. Let X be the component of $G - C$ containing x. Then, $N_G(X)$ is contained in C, implying $|N_G(X)| \leq |C| < k_x \leq k(X)$. Hence, by the assumption on S, we have $S \cap X \neq \emptyset$. But this means that there exists a path connecting x to S avoiding C, contrary to the choice of C. $\square$

This proposition allows us to express the vector connectivity problem as an integer linear problem (ILP) in the following way:

$$\min \sum_{v \in V} x_v$$

$$s.t. \quad \sum_{u \in X} x_u \geq 1 \textit{ for all } \emptyset \neq X \subseteq V \textit{ such that } k(X) > |N_G(X)|$$

$$x_v \in \{0, 1\} \textit{ for all } v \in V.$$

Unfortunately, the number of constraints in the above ILP can be exponential in the size of the input.[1] On the other hand, the fact that the decision version of vector connectivity is in NP, along with the fact that integer programming is NP-hard (see, e.g., [14]), imply that there exists a polynomially sized integer programming formulation of vector connectivity. We are, however, not aware of any explicit such formulation.

[1]To construct an explicit family of inputs $(G, \mathbf{k})$ such that the number of inclusion-minimal subsets $X \subseteq V(G)$ with $k(X) > |N_G(X)|$ grows exponentially with $|V(G)|$, let us consider complete graphs with uniform requirements, that is, n-vertex complete graphs where every vertex requirement is k, with $1 \leq k \leq n-1$. Then subset X of the vertex set such that $|X| \geq n-k+1$ satisfies the condition $k(X) > |N_G(X)|$, and the number of constraints grows exponentially with $|V(G)|$, since the number of sets X such that $|X| = n-k+1$ is $\binom{n}{n-k+1} = \binom{n}{k-1}$, which is exponential in n when k is close to $\frac{n}{2}$.

7 Numerical Results

In this section we present our experimental evaluation, testing the quality of solutions for Vector Domination and Vector Connectivity problems obtained by the two greedy algorithms in Sections 4 and 5. This was done by comparing them to the optimal solutions computed by the use of integer linear programming formulations presented in Section 6. For the comparison we used randomly generated graphs on 12 vertices with each edge present with probability p, independently of other edges, for the range of edge probabilities $p \in \{0.25,\ 0.5,\ 0.75\}$. For each choice of p we generated five 12-vertex graphs and for each obtained graph G we generated five requirement vectors at random, by selecting for each vertex $v \in V(G)$, its requirement k_v uniformly at random from the set $\{0, 1, \dots, d_G(v)\}$, independently of other vertices. Hence, for each choice of the edge probability p, we generated a total of 25 input instances for each of the two problems. The corresponding sets of instances will be denoted by $\mathcal{I}_{\Pi}^{p}$, where Π denotes either Vector Domination or Vector Connectivity and with A_{Π} we denote the corresponding greedy algorithm. Correspondingly, for an instance I of Π we denote by $A_{\Pi}(I)$ the size of the solution obtained by the greedy algorithm and by $OPT_{\Pi}(I)$ the corresponding optimal solution value. For each instance $I = (G, \mathbf{k})$ we computed the ratio $\frac{A_{\Pi}(I)}{OPT_{\Pi}(I)}$ and the difference $A_{\Pi}(I) - OPT_{\Pi}(I)$, as well as the upper bound $UB_{\Pi}(I)$ on the approximation ratio of the corresponding greedy algorithm given by the theoretical analysis in Section 3;

- If Π is the Vector Domination problem, then

$$UB_{\Pi}(I) = H(\max_{j \in V(G)} (k_j + d_G(j)))$$

(see Theorem 3.2 and the proof of Theorem 4.10).

- If Π is the Vector Connectivity problem, then

$$UB_{\Pi}(I) = H(\max_{j \in V(G)} (k_j + |\{z \in N_G(j) : k_z > 0\}|))$$

(see Theorem 3.2 and the proof of Theorem 5.7).

The resulting program was implemented in Java and, with the help of the CPLEX solver, we solved the two problems on the random instances to optimality. For the

same instances we also implemented the greedy algorithms described in earlier sections and computed the corresponding values.
The results of the experiment are summarized in the Table 7.

	Π	Vector Domination			Vector Connectivity		
		$p = 0.25$	$p = 0.5$	$p = 0.75$	$p = 0.25$	$p = 0.5$	$p = 0.75$
1.	$\max_{I \in \mathcal{I}_\Pi^p} UB_\Pi(I)$	5.7549	6.8775	7.5761	4.7887	6.2613	6.1861
2.	$\text{avg}_{I \in \mathcal{I}_\Pi^p} UB_\Pi(I)$	4.4729	6.0599	6.8096	4.0371	5.2111	4.7288
3.	$\text{stdev}_{I \in \mathcal{I}_\Pi^p} UB_\Pi(I)$	0.4574	0.3567	0.3432	0.3659	0.4229	0.3494
4.	$\max_{I \in \mathcal{I}_\Pi^p} \frac{A_\Pi(I)}{OPT_\Pi(I)}$	1.2000	1.1428	1.2000	1.0000	1.0000	1.0000
5.	$\text{avg}_{I \in \mathcal{I}_\Pi^p} \frac{A_\Pi(I)}{OPT_\Pi(I)}$	1.0533	1.0221	1.0337	1.0000	1.0000	1.0000
6.	$\text{stdev}_{I \in \mathcal{I}_\Pi^p} \frac{A_\Pi(I)}{OPT_\Pi(I)}$	0.0859	0.0508	0.0679	0.0000	0.0000	0.0000
7.	$\max_{I \in \mathcal{I}_\Pi^p}(A_\Pi(I) - OPT_\Pi(I))$	1.0000	1.0000	1.0000	0.0000	0.0000	0.0000
8.	$\text{avg}_{I \in \mathcal{I}_\Pi^p}(A_\Pi(I) - OPT_\Pi(I))$	0.2800	0.1600	0.2000	0.0000	0.0000	0.0000
9.	$\text{stdev}_{I \in \mathcal{I}_\Pi^p}(A_\Pi(I) - OPT_\Pi(I))$	0.4490	0.3667	0.4000	0.0000	0.0000	0.0000

Table 2: Computational results about the greedy algorithms.

The reason for the small size of randomly generated instances is due to the fact that the integer programming formulation of Vector Connectivity given in Section 6 has an exponential number of constraints. This limited the size of the integer programs that we could solve optimally using the freely available version of the CPLEX solver.
The above experiments show that on the given set of instances, the greedy algorithm computes solutions that are never more than 20% away from the optimum (see line 5.) which is much better than the theoretically predicted upper bounds on the approximation ratios of the two algorithms (see lines 1–3). The known inapproximability result for Vector Domination (see Section 4.3) implies that the theoretically predicted upper bound will be asymptotically tight in the worst case. However, it is possible that on random instances the greedy algorithm performs much better on average.
Furthermore, the results seem to suggest that we can be even more optimistic about Vector Connectivity. First of all, on the given set of instances the greedy algorithm always solved the problem to optimality. Second, as pointed out at the end of Section 5, the worst case polynomial-time approximability status for Vector Connectivity is not known, leaving open the existence of a constant factor approximation algorithm for the problem. In particular, it is not known whether the greedy algorithm approximates Vector Connectivity within a constant factor. Of course, due to the small size of the generated random instances, one should be careful before claiming any educated guesses about the actual worst case (in)approximability status of Vector Connectivity.

In order to perform a more extensive experimental analysis of the average performance of the greedy algorithm for Vector Connectivity, an explicit polynomially sized integer programming formulation of the problem would probably be helpful.

8 Conclusion

Our main topics were the problems of Vector Domination and vector connectivity. In order to state the relevant results, we needed to give definitions of submodular functions and define the submodular set covering problem, as well as its special case, the integer covering problem. We stated the results for the submodular set covering problem, which included a greedy algorithm that would approximate the integer covering problem in polynomial time within a factor of $\ln n + 2$, where n is the number of variables.
To get a polynomial-time approximation of Vector Domination, we defined a suitable submodular function and used the greedy algorithm stated for the submodular set covering problem. This resulted in an approximation of Vector Domination within a factor of $\ln(2\Delta(G)) + 1$, where $\Delta(G)$ is the maximum degree of a vertex in G.
We also stated some results for problems related to Vector Domination, such as total vector domination, α-domination, α-rate domination, and multiple domination. We established that total vector domination and total α-domination can be approximated in polynomial time by a factor of $\ln(\Delta(G)) + 1$ by showing that every instance of total vector domination can be described as an instance of set multicovering problem, which is a special case of the integer covering problem. In a similar way we showed that multiple domination and α-rate domination can be approximated by a factor of $\ln(\Delta(G) + 1) + 1$.
We presented the best known inapproximability results for Vector Domination and the related problems under the assumption that $P \neq NP$. The result for Vector Domination states that there is no polynomial-time algorithm approximating Vector Domination within a factor of $(1 - \epsilon)\ln n$ on n-vertex graphs, unless $P = NP$.
For vector connectivity we also defined a suitable function, the submodularity of which was established by means of gammoids and Menger's theorem. We again used the greedy algorithm stated for the submodular set covering to see that the problem can be approximated, in polynomial time, within a factor of $\ln n + 2$ on n-vertex graphs. We also stated the best known inapproximability results for vector connectivity. Unlike for Vector Domination and related problems, the gap between upper and lower bounds regarding approximabilty of vector connectivity is large. It is only known that the problem cannot be approximated in polynomial time to an arbitrary precision unless $P = NP$; even the existence of a constant-factor approximation algorithm is an

open problem. For further reading on approximation and parameterized algorithms for vector connectivity the interested reader is referred to [20].

9 Povzetek naloge v slovenskem jeziku

V zaključni nalogi smo obravnavali problema vektorske dominacije in vektorske povezanosti v grafih kot posebna primera problema submodularnega pokritja in analizirali požrešno metodo za omenjena problema glede na kvaliteto aproksimacije, tako s teoretičnega vidika (po Wolseyu) kot tudi eksperimentalno, na slučajno generiranih vhodnih podatkih.
Za dan graf $G = (V, E)$ in tak vektor $\mathbf{k} = (k_v : v \in V)$, indeksiran z vozlišči grafa G, da je za vsak $v \in V$ ustrezna komponenta k_v celo število med 0 in stopnjo vozlišča v, problem vektorske dominacije zahteva, da najdemo najmanjšo tako množico $S \subseteq V$, da ima vsako vozlišče v, ki ni v množici S, vsaj k_v sosedov v množici S. Za vektorsko povezanost pa je cilj najti najmanjšo tako mnžico $S \subseteq V$, da ima vsako vozlišče v, ki ni v množici S, vsaj k_v disjunktnih poti do množice S. Če so dolžine poti enake ena, problem postane problem vektorske dominacije. Vektorska povezanost je torej relaksacija problema vektorske dominacije, kjer je pogoj o številu sosedov v množici S nadomeščen s pogojem o številu (poljubno dolgih) disjunktnih poti do množice S.
Preden v zaključni nalogi navedemo rezultate za omenjena problema, definiramo pojem submodularne funkcije in problem submodularnega pokritja. Uporabimo rezultat Wolseyja iz leta 1982, ki pravi, da obstaja požrešni algoritem, ki aproksimira problem submodularnega pokritja s faktorjem $(1+\ln\gamma)$, kjer je γ eden od več možnih parametrov problema. V posebnem primeru, ko je funkcija celoštevilska, analiza daje zgornjo mejo za faktor aproksimacije vrednosti $H(\max_j f(\{j\}) - f(\emptyset))$, kjer je f dana submodularna funkcija in $H(d) = \sum_{i=1}^{d} \frac{1}{i}$. To privede do faktorja aproksimacije $H(\max \sum_{i=1}^{m} a_{ij})$ za problem celoštevilskega pokritja, kar je rezultat Dobsona, ki posplošuje prejšnje rezultate Johnsona, Lovásza in Chvátala za problem celoštevilskega pokritja.
Z definicijo ustrezne submodularne funkcije problem vektorske dominacije obravnavamo kot poseben primer problema submodularnega pokritja. Od tod z uporabo požrešnega aproksimacijskega algoritma za problem submodularnega pokritja izpeljemo, da je problem vektorske dominacije mogoče aproksimirati v polinomskemu času s faktorjem aproksimacije $\ln(2\Delta(G)) + 1$, kjer $\Delta(G)$ označuje največjo stopnjo vozlišča v grafu G. Verjetno je, da je do členov nižjega reda natančno ta rezultat najboljši možen. Ob-

stoj algoritma, ki bi problem vektorske dominacije v polinomskem času aproksimiral s faktorjem aproksimacije $(1-\epsilon)\ln n$ na grafih z n vozlišči, bi namreč impliciral enakost $P = NP$.

V zaključni nalogi obravnavamo tudi nekatere sorodne probleme, kot so totalna dominacija, večkratna dominacija, α-dominacija, totalna α-dominacija in α-kratna dominacija. Nekatere od teh problemov prevedemo na problem večkratnega pokritja, ki je poseben primer problema celoštevilskega pokritja. Vsi omenjeni problemi pa so poseben primer problema submodularnega pokritja. Prevedbe omogočajo aproksimacijo navedenih problemov v polinomskem času s faktorjem aproksimacije $\ln n + 2$ za n-vozliščne grafe.

Podobno kot za problem vektorske dominacije tudi problem vektorske povezanosti z uvedbo ustrezne submodularne funkcije preoblikujemo v poseben primer problema submodularnega pokritja, kar vodi do aproksimacije s faktorjem $\ln n + 2$ za n-vozliščne grafe v polinomskem času. Za razliko od problema vektorske dominacije in z njim povezanih problemov, pa ni znano, ali obstaja kakšno tako pozitivno število c, da bi obstoj algoritma polinomske časovne zahtevnosti, ki bi vektorsko povezanost na grafih z n vozlišči aproksimiral s faktorjem $c\ln n$, impliciral enakost $P = NP$. Tudi obstoj aproksimacijskega algoritma za problem vektorske povezanosti s konstantnim faktorjem aproksimacije je odprt problem.

Problema vektorske dominacije in vektorske povezanosti nadalje modeliramo s celoštevilskima linearnima programoma in s pomočjo računalnika analiziramo kvaliteto aproksimacijskega algoritma, ki temelji na Wolseyevem izreku, na slučajno generiranih podatkih. Optimalne vrednosti so izračunane z uporabo CPLEX-a in izpeljanih celoštevilskih linearnih programov.

10 Bibliography

[1] N. Alon, D. Moshkovitz, and S. Safra, *Algorithmic Construction of Sets for k-restrictions.* ACM Transactions on Algorithms 2, 2006 (153–177). *(Cited on page 26.)*

[2] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi, *Complexity and Approximation: Combinatorial Otimization Problems and Their Approximability Properties.* Springer Science & Business Media, 2003. *(Cited on pages 10 and 32.)*

[3] C. Berge, *Hypergraphs: Combinatorics of Finite Sets*, North Holland, (1989). *(Cited on page 6.)*

[4] F. Bonomo,B. Brešar,L.N. Grippo, M. Milanič, and M.D. Safe, Domination parameters with number 2: Interrelations and algorithmic consequences. *Discrete Applied Mathematics* 235 (2018) 23–50. *(Cited on page 26.)*

[5] E. Boros, P. Heggernes, P. van 't Hof, and M. Milanič, Vector Connectivity in graphs. *Networks 63* (2014) 1–9. *(Cited on pages 1, 27, 28, 29, and 30.)*

[6] M. Chlebík and J. Chlebíkova, Approximation hardness of dominating set problems in bounded degree graphs. *Information and Computation* 206 (2008) 1264–1275. *(Cited on page 26.)*

[7] V. Chvátal, A greedy heuristic for the set-covering problem. *Mathematics of operations research* 4 (1979) 233–235. *(Cited on page 13.)*

[8] F. Cicalese, M. Milanič, and U. Vaccaro, On the approximability and exact algorithms for vector domination and related problems in graphs. *Discrete Applied Mathematics 161* (2013) 750–767. *(Cited on pages 1, 19, and 26.)*

[9] F Cicalese, M Milanič, and R. Rizzi, On the complexity of the Vector Connectivity problem. *Theoretical Computer Science 591* (2015) 60–71. *(Cited on pages 32, 33, and 35.)*

[10] I. Dinur and D. Steurer, *Analytical Approach to Parallel Repetition.* ACM, New York, 2014. *(Cited on page 26.)*

[11] G. DOBSON, Worst-case analysis of greedy heuristics for integer programming with nonnegative data. *Mathematics of Operations Research, 7 (4)* (1982) 515–531. *(Cited on pages 12, 13, and 22.)*

[12] J. EDMONDS, Matroids and the greedy algorithm. *Combinatorica* 15 (1995) 215–245. *(Cited on page 12.)*

[13] T. FUJITO, Approximation algorithms for submodular set cover with applications. *IEICE Transactions on Information and Systems* 83 (2000) 480–487. *(Cited on page 12.)*

[14] M. GAREY and D.S. JOHNSON, *Computers and Intractability: A Guide to NP-completeness*. WH Freeman and Company, San Francisco, 1979. *(Cited on page 36.)*

[15] B. GÄRTNER and J. MATOUSEK, *Understanding and Using Linear Programming.* Springer Science & Business Media, 2007. *(Cited on page 10.)*

[16] J. HARANT ,A. PRUCHNEWSKI, and M. VOIGT, On dominating sets and independent sets of graphs. *Combinatorics, Probability and Computing 11* (1993) 1–10. *(Cited on page 1.)*

[17] T.W. HAYNES, S. HEDETNIEMI, and P. SLATER, *Fundamentals of Domination in Graphs, Marcel Dekker*. Inc. New York, 1998. *(Cited on page 1.)*

[18] T.W. HAYNES, T. W. HEDETNIEMI, and P. SLATER, *Domination in Graphs (Advanced Topics) Marcel Dekker Publications.* (1998). *(Cited on page 1.)*

[19] D.S. JOHNSON, Approximation algorithms for combinatorial problems. *Journal of Computer and System Sciences 9 (3)* (1974) 256–278. *(Cited on page 13.)*

[20] S. KRATSCH and M. SORGE, On Kernelization and Approximation for the Vector Connectivity Problem. *Algorithmica* 79 (2017) 96–138. *(Cited on page 41.)*

[21] L. LOVÁSZ, On the ratio of optimal integral and fractional covers. *Discrete Mathematics 13 (4)* (1975) 383–390. *(Cited on page 13.)*

[22] K. MENGER, Zur allgemeinen Kurventheorie. *Fundamenta Mathematicae* 10 (1927) 96–115. *(Cited on pages 9 and 35.)*

[23] G.L. NEMHAUSER and L.A. WOLSEY, An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming* 14 (1978) 256–294. *(Cited on page 4.)*

[24] J.B. Orlin, *Max Flows in O (nm) Time, or Better*, Proceedings of the Forty-Fifth Annual ACM Symposium on Theory of Computing, 2013. *(Cited on page 30.)*

[25] C.H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Prentice-Hall, Inc. 1982. *(Cited on page 10.)*

[26] H. Perfect, Applications of Menger's graph theorem. *Journal of Mathematical Analysis and Applications* 22 (1968) 96–111. *(Cited on page 8.)*

[27] J.S. Pym, A proof of the linkage theorem. *Journal of Mathematical Analysis and Applications* 27 (1969) 636–638. *(Cited on page 8.)*

[28] A. Schrijver, *Combinatorial Optimization: Polyhedra and Efficiency*, Springer Science & Business Media, Volume 24, 2003. *(Cited on pages 4 and 30.)*

[29] V.V. Vazirani, *Approximation Algorithms*, Springer Science & Business Media, 2013. *(Cited on page 10.)*

[30] D.J.A Welsh, *Matroid Theory*, Academic Press, 1976. *(Cited on page 6.)*

[31] D.B. West, *Introduction to Graph Theory*, Prentice Hall Upper, Saddle River, Volume 2, 2001. *(Cited on page 3.)*

[32] L.A. Wolsey, An analysis of the greedy algorithm for the submodular set covering problem. Combinatorica 2 (1982) 385–393. *(Cited on pages 1 and 12.)*