Density Functions subject to a Co-Matroid Constraint*

Venkatesan T. Chakaravarthy[1], Natwar Modani[1], Sivaramakrishnan R. Natarajan[2], Sambuddha Roy[1], and Yogish Sabharwal[1]

1 **IBM Research, New Delhi, India**
{vechakra,namodani,sambuddha,ysabharwal}@in.ibm.com
2 **IIT Chennai, India**
sivaramakrishnan.n.r@gmail.com

Abstract

In this paper we consider the problem of finding the *densest* subset subject to *co-matroid constraints*. We are given a *monotone supermodular* set function f defined over a universe U, and the density of a subset S is defined to be $f(S)/|S|$. This generalizes the concept of graph density. Co-matroid constraints are the following: given matroid $\mathcal{M}$ a set S is feasible, iff the complement of S is *independent* in the matroid. Under such constraints, the problem becomes NP-hard. The specific case of graph density has been considered in literature under specific co-matroid constraints, for example, the cardinality matroid and the partition matroid. We show a 2-approximation for finding the densest subset subject to co-matroid constraints. Thereby we improve the approximation guarantees for the result for partition matroids in the literature.

1998 ACM Subject Classification F.2.2 Nonnumerical Algorithms and Problems

Keywords and phrases Approximation Algorithms, Submodular Functions, Graph Density

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2012.236

1 Introduction

In this paper, we consider the problem of computing the densest subset with respect to a *monotone supermodular* function subject to *co-matroid* constraints. Given a universe U of n elements, a function $f : 2^U \to \mathbb{R}^+$ is *supermodular* iff

$$f(A) + f(B) \leqslant f(A \cup B) + f(A \cap B)$$

for all $A, B \subseteq U$. If the sign of the inequality is reversed for all A, B, then we call the function *submodular*. The function f is said to be monotone if $f(A) \leqslant f(B)$ whenever $A \subseteq B$; we assume $f(\emptyset) = 0$. We define a *density function* $d : 2^U \to \mathbb{R}^+$ as $d(S) \triangleq f(S)/|S|$. Consider the problem of maximizing the density function $d(S)$ given oracle access to the function f. We observe that the above problem can be solved in polynomial time (see Theorem 6).

The main problem considered in this paper is to maximize $d(S)$ subject to certain constraints that we call *co-matroid* constraints. In this scenario, we are given a *matroid* $\mathcal{M} = (U, \mathcal{I})$ where $\mathcal{I} \subseteq 2^U$ is the family of *independent* sets (we give the formal definition of a matroid in Section 2). A set S is considered feasible iff the complement of S is *independent*

* Work done by the third author while he was interning at IBM Research


© V. Chakaravarthy, N. Modani, S. R. Natarajan, S. Roy, Y. Sabharwal;
licensed under Creative Commons License NC-ND
32nd Int'l Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2012).
Editors: D. D'Souza, J. Radhakrishnan, and K. Telikepalli; pp. 236–248

Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

i.e. $\overline{S} \in \mathcal{I}$. The problem is to find the densest feasible subset S given oracle access to f and $\mathcal{M}$. We denote this problem as DEN-M.

We note that even special cases of the DEN-M problem are NP-hard [14]. The main result in this paper is the following:

▶ **Theorem 1.** *Given a monotone supermodular function f over a universe U, and a matroid $\mathcal{M}$ defined over the same universe, there is a 2-approximation algorithm for the* DEN-M *problem.*

Alternatively one could have considered the same problem under *matroid constraints* (instead of co-matroid constraints). We note that this problem is significantly harder, since the Densest Subgraph problem can be reduced to special cases of this problem (see [2, 14]). The Densest Subgraph problem is notoriously hard: the best factor approximation known to date is $O(n^{1/4+\epsilon})$ for any $\epsilon > 0$ [3].

Special cases of the DEN-M problem have been extensively studied in the context of graph density, and we discuss this next.

1.1 Comparison to Graph Density

Given an undirected graph $G = (V, E)$, the density $d(S)$ of a subgraph on vertex set S is defined as the quantity $\frac{|E(S)|}{|S|}$, where $E(S)$ is the set of edges in the subgraph induced by the vertex set S. The densest subgraph problem is to find the subgraph S of G that maximizes the density.

The concept of graph density is ubiquitous, more so in the context of social networks. In the context of social networks, the problem is to detect *communities*: collections of individuals who are relatively well connected as compared to other parts of the social network graph.

The results relating to graph density have been fruitfully applied to finding communities in the social network graph (or even web graphs, gene annotation graphs [15], problems related to the formation of most effective teams [9], etc.). Also, note that graph density appears naturally in the study of threshold phenomena in random graphs, see [1].

Motivated by applications in social networks, the graph density problem and its variants have been well studied. Goldberg [11] proved that the densest subgraph problem can be solved optimally in polynomial time: he showed this via a reduction to a series of max-flow computations. Later, others [7, 14] have given new proofs for the above result, motivated by considerations to extend the result to some generalizations and variants.

Andersen and Chellapilla [2] studied the following generalization of the above problem. Here, the input also includes an integer k, and the goal is to find the densest subgraph S subject to the constraint $|S| \geqslant k$. This corresponds to finding *sufficiently large* dense subgraphs in social networks. This problem is NP-hard [14]. Andersen and Chellapilla [2] gave a 2-approximation algorithm. Khuller and Saha [14] give two alternative algorithms: one of them is a greedy procedure, while the other is LP-based. Both the algorithms have 2-factor guarantees.

Gajewar and Sarma [9] consider a further generalization. The input also includes a partition of the vertex set into $U_1, U_2, \cdots, U_t$, and non-negative integers $r_1, r_2, \cdots, r_t$. The goal is to find the densest subgraph S subject to the constraint that for all $1 \leqslant i \leqslant t$, $|S \cap U_i| \geqslant r_i$. They gave a 3-approximation algorithm by extending the greedy procedure of Khuller and Saha [14].

We make the following observations: (i) The objective function $|E(S)|$ is monotone and supermodular. (ii) The constraint $|S| \geqslant k$ (considered by [2]) is a co-matroid constraint; this corresponds to the *cardinality matroid*. (iii) The constraint considered by Gajewar and

Sarma [9] is also a co-matroid constraint; this corresponds to the *partition matroid* (formal definitions are provided in Section 2). Consequently, our main result Theorem 1 improves upon the above results in three directions:

- *Objective function*: Our results apply to general monotone supermodular functions f instead of the specific set function $|E(S)|$ in graphs.
- *Constraints*: We allow co-matroid constraints corresponding to *arbitrary* matroids.
- *Approximation Factor:* For the problem considered by Gajewar and Sarma [9], we improve the approximation guarantee from 3 to 2. We match the best factor known for the at-least-k densest subgraph problem considered in [2, 14].

1.2 Other Results

Knapsack Covering Constraints:

We also consider the following variant of the `DEN-M` problem. In this variant, we will have a weight w_i (for $i = 1, \cdots, |U|$) for every element $i \in U$, and a number $k \in \mathbb{N}$. A set S of elements is *feasible* if and only if the following condition holds: $\sum_{i \in S} w_i \geqslant k$. We call this a *knapsack covering constraint*. We extend the proof of Theorem 1 to show the following:

▶ **Theorem 2.** *Suppose we are given a monotone supermodular function f over a universe U, weights w_i for every element $i \in U$, and a number $k \in \mathbb{N}$. Then there is a 3-approximation algorithm for maximizing the* density function $d(S)$ *subject to* knapsack covering constraints *corresponding to the weights w_i and the number k.*

Dependency Constraints:

Saha et. al[15] consider a variant of the graph density problem. In this version, we are given a specific collection of vertices $A \subseteq V$; a subset S of vertices is *feasible* iff $A \subseteq S$. We call this restriction the *subset* constraint. The objective is to find the densest subgraph among subsets satisfying a subset constraint. Saha et. al[15] prove that this problem is solvable in polynomial time by reducing this problem to a series of max-flow computations.

We study a generalization of the subset constraint problem. Here, we are given a monotone supermodular function f defined over universe U. Additionally, we are given a *directed graph* $D = (U, \vec{A})$ over the universe U. A feasible solution S has to satisfy the following property: if $a \in S$, then every vertex of the digraph D *reachable* from a also has to belong to S. Alternatively, $a \in S$ and $(a, b) \in \vec{A}$ implies that $b \in S$. We call the digraph D as the *dependency graph* and such constraints as *dependency* constraints. The goal is to find the densest subset S subject to the dependency constraints. We call this the `DENdep` problem. We note that the concept of dependency constraints generalizes that of the subset constraints: construct a digraph D by drawing directed arcs from every vertex in U to every vertex in A. The motivation for this problem comes from certain considerations in social networks, where we are to find the densest subgraph but with the restriction that in the solution subgraph all the members of a sub-community (say, a family) are present or absent simultaneously. In literature, such a solution S that satisfies the dependency constraints is also called a *closure* (see [18], Section 3.7.2). Thus our problem can be rephrased as that of finding the densest subset over all closures.

We note that dependency constraints are incomparable with co-matroid constraints. In fact dependency constraints are not even upward monotone: it is *not* true that if S is a feasible subset, *any* superset of S is feasible.

Our result is as follows:

▶ **Theorem 3.** *The* `DENdep` *problem is solvable in polynomial time.*

The salient features of the above result are as follows:

- While the result in [15] is specific to graph density, our result holds for density functions arising from arbitrary monotone supermodular functions.
- Our proof of this result is LP-based. The work of [15] is based on max-flow computations. We can extend our LP-based approach (via convex programs) to the case for density functions arising from arbitrary monotone supermodular f, while we are not aware as to how to extend the max-flow based computation.
- The proof technique, inspired by Iwata and Nagano [13] also extends to show "small support" results: thus, for instance, we can show that for the LP considered by [14] for the at-least-k-densest subgraph problem, every *non-zero* component of any basic feasible solution is one of *two* values.

Combination of Constraints:

We also explore the problem of finding the densest subset subject to a combination of the constraints considered. We are able to prove results for the problem of maximizing a density function subject to (a) *co-matroid* constraints and (b) *subset* constraints. Suppose we are given a monotone supermodular function f over a universe U, a matroid $\mathcal{M} = (U, \mathcal{I})$, and a subset of elements $A \subseteq U$. A subset S is called feasible iff (1) S satisfies the co-matroid constraints wrt $\mathcal{M}$ (i.e. $\overline{S} \in \mathcal{I}$) and (2) S satisfies the subset constraint wrt A (i.e. $A \subseteq S$). We show the following:

▶ **Theorem 4.** *There is a 2-approximation algorithm for the problem of maximizing the* density function $d(S)$ *corresponding to a monotone supermodular function f, subject to the co-matroid and subset constraints.*

1.3 Related Work

Recently, there has been a considerable interest in the problems of optimizing submodular functions under various types of constraints. The most common constraints that are considered are *matroid constraints*, *knapsack constraints* or combinations of the two varieties. Thus for instance, Calinescu et. al [5] considered the problem of maximizing a *monotone* submodular function subject to a matroid constraint. They provide an algorithm and show that it yields a $(1 - 1/e)$-approximation: this result is essentially optimal (also see the recent paper [8] for a combinatorial algorithm for the same). Goemans and Soto [10] consider the problem of minimizing a *symmetric* submodular function subject to arbitrary *matroid* constraints. They prove the surprising result that this problem can be solved in polynomial time. In fact, their result extends to the significantly more general case of *hereditary constraints*; the problem of extending our results to *arbitrary* hereditary functions is left open.

The density functions that we consider may be considered as "close" to the notion of supermodular functions. To the best of our knowledge, the general question of *maximizing* density functions subject to a (co-)matroid constraint has never been considered before.

1.4 Proof Techniques

We employ a greedy algorithm to prove Theorems 1 and 2. Khuller and Saha [14] and Gajewar and Sarma [9] had considered a natural greedy algorithm for the problem of maximizing graph density subject to co-matroid constraints corresponding to the cardinality matroid and partition matroid respectively. Our greedy algorithm can be viewed as a natural abstraction of the greedy algorithm to the generalized scenario of arbitrary monotone supermodular functions. However, our analysis is different from that in [14, 9]. In both of the earlier papers

[14, 9], a particular stopping condition is employed to define a set D_ℓ useful in the analysis. For instance, in Section 4.1 of [9] they define D_ℓ using the optimal set H^* directly. We choose a different *stopping condition* to define the set D_ℓ; it turns out that this choice is crucial for achieving a 2-factor guarantee. Another reason for our improvement is the following: a straightforward generalization of the arguments given in [9] (to the scenario of arbitrary monotone supermodular functions) would imply a version of Claim 4 with a factor of $d^*/4$ (instead of $d^*/2$ as provided in Claim 4).

We prove Theorem 3 using LP-based techniques. Our technique also provides another proof of the basic result that graph density is computable in polynomial time. The proof method is inspired by Iwata and Nagano [13].

1.5 Organization

We present the relevant definitions in Section 2. We proceed to give the proof of Theorem 1 in Section 3. The proof of Theorem 3 is presented in Section 4. For space considerations, we include the proofs of Theorems 2 and 4 in a fuller version of the paper available at [6].

2 Preliminaries

In this paper, we will use the following notation: given *disjoint* sets A and B we will use $A+B$ to serve as shorthand for $A \cup B$. Vice versa, when we write $A+B$ it will hold implicitly that the sets A and B are disjoint.

Monotone: A set function f is called *monotone* if $f(S) \leqslant f(T)$ whenever $S \subseteq T$.

Supermodular: A set function $f : 2^U \to \mathbb{R}^+$ over a universe U is called *supermodular* if the following holds for any two sets $A, B \subseteq U$:

$$f(A) + f(B) \leqslant f(A \cup B) + f(A \cap B)$$

If the inequality holds (for every A, B) with the sign reversed, then the function f is called *submodular*. In this paper, we will use the following equivalent definition of supermodularity: given disjoint sets A, B and C,

$$f(A + C) - f(A) \leqslant f(A + B + C) - f(A + B)$$

We can think of this as follows: the *marginal utility* of the set of elements C to the set A increases as the set becomes "larger" ($A + B$ instead of A).

It is well known (see [12, 16]) that supermodular functions can be *maximized* in polynomial time (whereas submodular functions can be minimized in polynomial time). Let us record this as:

▶ **Theorem 5.** *Any supermodular function $f : 2^U \to \mathbb{R}^+$ can be maximized in polynomial time.*

We also state the following folklore corollary:

▶ **Corollary 6.** *Given any supermodular function $f : 2^U \to \mathbb{R}^+$, we can find $\max_S \frac{f(S)}{|S|}$ in polynomial time.*

A proof of this Corollary is provided in the full version [6].

Density Function: Given a function f over U, the density of a set S is defined to be $d(S) = \frac{f(S)}{|S|}$.

Matroid: A matroid is a pair $\mathcal{M} = (U, \mathcal{I})$ where $\mathcal{I} \subseteq 2^U$, and

$i \leftarrow 1$
$H_i \leftarrow \arg\max_X \frac{f(X)}{|X|}$
$D_i \leftarrow H_i$
while D_i infeasible **do**
  $H_{i+1} \leftarrow \arg\max_{X: X \cap D_i = \emptyset} \frac{f(D_i+X)-f(D_i)}{|X|}$
  $D_{i+1} \leftarrow D_i + H_{i+1}$
  $i \leftarrow i+1$
end while
$L \leftarrow i$
for $i = 1 \rightarrow L$ **do**
  Add arbitrary vertices to D_i to make it minimal feasible
  Call the result D'_i
end for
Output the subset among the D'_i's with the highest density

Figure 1 Main Algorithm

1. (Hereditary Property) $\forall B \in \mathcal{I}, A \subset B \implies A \in \mathcal{I}$.
2. (Extension Property) $\forall A, B \in \mathcal{I} : |A| < |B| \implies \exists x \in B \setminus A : A + x \in \mathcal{I}$

Matroids are generalizations of vector spaces in linear algebra and are ubiquitous in combinatorial optimization because of their connection with greedy algorithms. Typically the sets in $\mathcal{I}$ are called *independent* sets, this being an abstraction of linear independence in linear algebra. The maximal independent sets in a matroid are called the *bases* (again preserving the terminology from linear algebra). An important fact for matroids is that all bases have equal cardinality – this is an outcome of the Extension Property of matroids.

Any matroid is equipped with a *rank function* $r : 2^U \rightarrow \mathbb{R}^+$. The rank of a subset S is defined to be the size of the *largest* independent set contained in the subset S. By the Extension Property, this is well-defined. See the excellent text by Schrijver [17] for details.

Two commonly encountered matroids are the (i) *Cardinality Matroid:* Given a universe U and $r \in \mathbb{N}$, the *cardinality matroid* is the matroid $\mathcal{M} = (U, \mathcal{I})$, where a set A is *independent* (i.e. belongs to $\mathcal{I}$) iff $|A| \leqslant r$. (ii) *Partition Matroid:* Given a universe U and a partition of U as $U_1, \cdots, U_r$ and non-negative integers $r_1, \cdots, r_t$, the *partition matroid* is $\mathcal{M} = (U, \mathcal{I})$, where a set A belongs to $\mathcal{I}$ iff $|A \cap U_i| \leqslant r_i$ for all $i = 1, 2, \cdots, t$.

Convex Programs: We will need the definition of a convex program, and that they can be solved to arbitrary precision in polynomial time, via the ellipsoid method(see [12]). We refer the reader to the excellent text [4].

3 Proof of Theorem 1

We first present the algorithm and then its analysis. To get started, we describe the intuition behind the algorithm.

Note that co-matroid constraints are *upward monotone*: if a set S is feasible for such constraints, then any *superset* of S is also feasible. Thus, it makes sense to find a *maximal* subset of U with the maximum density. In the following description of the algorithm, one may note that the sets $D_1, D_2, \cdots, D_i$ are an attempt to find the maximal subset with the largest density. Given this rough outline, the algorithm is presented in Figure 1.

We note that we can find the maximum $\max_{X: X \cap D_i = \emptyset} \frac{f(D_i+X)-f(D_i)}{|X|}$ in polynomial time. This is because the function $f(D_i + X)$ for a fixed D_i is supermodular (and we appeal to

Corollary 6).

Let H^* denote the optimal solution, i.e. the subset that maximizes the density $d(S)$ subject to the co-matroid constraints. Let d^* denote the optimal density, so that $f(H^*) = d^* \cdot |H^*|$.

We can make the following easy claim:

▶ Claim 1. The subset D_1 obeys the inequality $d(D_1) \geqslant d^*$.

This is because D_1 is the densest subset in the universe U, while d^* is the density of a specific subset H^*.

In the following, we will have occasion to apply the following lemmas.

▶ **Lemma 7.** *Let $a, b, c, d, \theta \in \mathbb{R}^+$ be such that the inequalities $\frac{a}{b} \geqslant \theta$ and $\frac{c}{d} \geqslant \theta$ hold. Then it is true that $\frac{a+c}{b+d} \geqslant \theta$. Thus, if $\frac{a}{b} \geqslant \frac{c}{d}$, then $\frac{a+c}{b+d} \geqslant \frac{c}{d}$ (by setting $\theta = \frac{c}{d}$).*

Also,

▶ **Lemma 8.** *Let $a, b, c, d \in \mathbb{R}^+$ be real numbers such that $\frac{a}{b} \geqslant \frac{c}{d}$ holds.*

- *Suppose $a \geqslant c$, $b \geqslant d$. Then the inequality $\frac{a-c}{b-d} \geqslant \frac{a}{b}$ holds.*
- *Suppose $c \geqslant a$, $d \geqslant b$. Then the inequality $\frac{c}{d} \geqslant \frac{c-a}{d-b}$ holds.*

We make the following claim:

▶ Claim 2. The sequence of subsets $D_1, D_2, \cdots, D_L$ obeys the following ordering:

$$\frac{f(D_1)}{|D_1|} \geqslant \frac{f(D_2) - f(D_1)}{|D_2| - |D_1|} \geqslant \cdots \geqslant \frac{f(D_{i+1}) - f(D_i)}{|D_{i+1}| - |D_i|} \geqslant \cdots \geqslant \frac{f(D_L) - f(D_{L-1})}{|D_L| - |D_{L-1}|}$$

Proof. Consider any term in this sequence, say $\frac{f(D_{i+1}) - f(D_i)}{|D_{i+1}| - |D_i|}$. Note that H_{i+1} was chosen as arg max of $\frac{f(D_i + X) - f(D_i)}{|X|}$. Therefore, $\max_X \frac{f(D_i+X)-f(D_i)}{|X|} = \frac{f(D_{i+1})-f(D_i)}{|D_{i+1}|-|D_i|}$. Hence this quantity is larger than $\frac{f(D_{i+2})-f(D_i)}{|D_{i+2}|-|D_i|}$ (as long as D_{i+2} is well defined). Now from the second part of Lemma 8, we get that

$$\frac{f(D_{i+1}) - f(D_i)}{|D_{i+1}| - |D_i|} \geqslant \frac{f(D_{i+2}) - f(D_i)}{|D_{i+2}| - |D_i|} \geqslant \frac{f(D_{i+2}) - f(D_{i+1})}{|D_{i+2}| - |D_{i+1}|}$$

◀

Via an application of Lemma 7, we then have:

▶ Claim 3. Given any i $(1 \leqslant i \leqslant L)$, the following holds:

$$\frac{f(D_i)}{|D_i|} \geqslant \frac{f(D_i) - f(D_{i-1})}{|D_i| - |D_{i-1}|}$$

Proof. We will the prove the statement by induction.
Base Case: We implicitly assume that $D_0 = \emptyset$, and hence the case for $i = 1$ holds.
Induction Step: Assume the statement by induction for $i = k$, and we prove it for $i = k+1$. Thus, by hypothesis we have

$$\frac{f(D_k)}{|D_k|} \geqslant \frac{f(D_k) - f(D_{k-1})}{|D_k| - |D_{k-1}|}$$

Now by Claim 2 we have that

$$\frac{f(D_k) - f(D_{k-1})}{|D_k| - |D_{k-1}|} \geqslant \frac{f(D_{k+1}) - f(D_k)}{|D_{k+1}| - |D_k|}$$

Thus,

$$\frac{f(D_k)}{|D_k|} \geqslant \frac{f(D_{k+1}) - f(D_k)}{|D_{k+1}| - |D_k|}$$

Applying Lemma 7, we get:

$$\frac{f(D_{k+1})}{|D_{k+1}|} \geqslant \frac{f(D_{k+1}) - f(D_k)}{|D_{k+1}| - |D_k|}$$

Thus we have proven the Claim by induction. ◀

The analysis will be broken up into two parts. We will consider the set D_ℓ in the sequence $D_1, D_2, \cdots, D_L$ such that the following hold:

$$\frac{f(D_\ell) - f(D_{\ell-1})}{|D_\ell| - |D_{\ell-1}|} \geqslant \frac{d^*}{2}$$

but

$$\frac{f(D_{\ell+1}) - f(D_\ell)}{|D_{\ell+1}| - |D_\ell|} < \frac{d^*}{2}$$

Since $d(D_1) \geq d^*$ by Claim 1, such an ℓ will exist or $\ell = L$. If $\ell = L$, then we have a feasible solution D_L with the property that $\frac{f(D_L)-f(D_{L-1})}{|D_L|-|D_{L-1}|} \geqslant \frac{d^*}{2}$. Therefore, by Claim 3 we have that $d(D_L) \geqslant \frac{d^*}{2}$ and we are done in this case.

So we may assume that $\ell < L$ so that D_ℓ is *not* feasible. In this case, we will prove that D'_ℓ has the correct density, i.e. that $d(D'_\ell) \geqslant \frac{d^*}{2}$.

To this end, we will prove two facts about D_ℓ and that will yield the desired result:

▶ Claim 4.

$$f(D_\ell) - f(D_\ell \cap H^*) \geqslant \frac{d^*}{2}(|D_\ell| - |D_\ell \cap H^*|)$$

Proof. Note that $D_\ell = H_1 + H_2 + \cdots + H_\ell$. For brevity, for $1 \leqslant i \leqslant \ell$, denote $H_i \cap H^*$ as A_i (thus, $A_i \subseteq H_i$ for every i). Thus, $D_\ell \cap H^* = A_1 + A_2 + \cdots + A_\ell$.

We will prove the following statement by induction on i (for $1 \leqslant i \leqslant \ell$):

$$f(H_1+H_2+\cdots+H_i) - f(A_1+A_2+\cdots+A_i) \geqslant \frac{d^*}{2}(|H_1+H_2+\cdots+H_i| - |A_1+A_2+\cdots+A_i|)$$

Base Case: For $i = 1$, we have to prove that:

$$\frac{f(H_1) - f(A_1)}{|H_1| - |A_1|} \geqslant \frac{d^*}{2}$$

Since H_1 is the *densest* subset, we have

$$\frac{f(H_1)}{|H_1|} \geqslant \frac{f(A_1)}{|A_1|}$$

and we may apply (the first part of) Lemma 8 to obtain the desired.

Induction Step: Assume the statement to be true for i, and we will prove it for $i + 1$.

Consider the following chain:

$$\frac{f(H_1+\cdots+H_i+H_{i+1})-f(H_1+\cdots+H_i)}{|H_{i+1}|} \overset{H_{i+1}\arg\max}{\geqslant} \frac{f(H_1+\cdots+H_i+A_{i+1})-f(H_1+\cdots+H_i)}{|A_{i+1}|} \overset{supermodular}{\geqslant} \frac{f(A_1+\cdots+A_i+A_{i+1})-f(A_1+\cdots+A_i)}{|A_{i+1}|}$$

We would now like to apply Lemma 8 to the first and last terms in the above chain. To this end, let us check the preconditions:

$$f(H_1+\cdots+H_i+H_{i+1})-f(H_1+\cdots+H_i) \overset{monotone}{\geqslant} f(H_1+\cdots+H_i+A_{i+1})-f(H_1+\cdots+H_i) \overset{supermodular}{\geqslant} f(A_1+\cdots+A_i+A_{i+1})-f(A_1+\cdots+A_i)$$

Since, clearly $|H_{i+1}| \geqslant |A_{i+1}|$, the preconditions in Lemma 8 hold and we have:

$$\frac{f(H_1+\cdots+H_{i+1})-f(A_1+\cdots+A_{i+1})-f(H_1+\cdots+H_i)+f(A_1+\cdots+A_i)}{|H_{i+1}|-|A_{i+1}|} \geqslant \frac{f(H_1+\cdots+H_i+H_{i+1})-f(H_1+\cdots+H_i)}{|H_{i+1}|} \geqslant \frac{d^*}{2}$$

Applying Lemma 7 to the first term in the above chain and the induction statement for i, we obtain the desired result for $i+1$. Hence done. ◀

The next claim lower bounds the value $f(D_\ell \cap H^*)$.

Building up to the Claim, let us note that $D_\ell \cap H^* \neq \emptyset$. If the intersection were empty, then H^* is a subgraph of density d^*, and so $H_{\ell+1}$ would be a subgraph of density at least d^*. But then,

$$\frac{f(D_\ell+H_{\ell+1})-f(D_\ell)}{|H_{\ell+1}|} \overset{supermodular}{\geqslant} \frac{f(H_{\ell+1})}{|H_{\ell+1}|} \geqslant d^*$$

But this contradicts the choice of D_ℓ.

▶ Claim 5.

$$f(D_\ell \cap H^*) \geqslant \frac{d^*}{2}|D_\ell \cap H^*| + \frac{d^*}{2}|H^*|$$

Proof. Let $X = H^* - D_\ell \cap H^*$. Then, $X \cap D_\ell = \emptyset$ and $D_\ell + X = D_\ell \cup H^*$. Then by definition of D_ℓ, we know that $\frac{f(D_\ell+X)-f(D_\ell)}{|X|} \leqslant \frac{f(D_{\ell+1})-f(D_\ell)}{|D_{\ell+1}|-|D_\ell|} < d^*/2$. Thus, $f(D_\ell \cup H^*) - f(D_\ell) \leqslant \frac{d^*}{2}(|H^*| - |D_\ell \cap H^*|)$.

Therefore, $f(D_\ell \cup H^*) + f(D_\ell \cap H^*) \leqslant f(D_\ell) + f(D_\ell \cap H^*) + \frac{d^*}{2}(|H^*| - |D_\ell \cap H^*|)$.

Applying supermodularity we have that $f(D_\ell \cup H^*) + f(D_\ell \cap H^*) \geqslant f(D_\ell) + f(H^*)$. Thus, cancelling $f(D_\ell)$ gives us that $f(D_\ell \cap H^*) + \frac{d^*}{2}(|H^*| - |D_\ell \cap H^*|) \geqslant f(H^*)$. The claim follows by observing that $d^* = \frac{f(H^*)}{|H^*|}$. ◀

Note that this claim also implies that the density of the set $D_\ell \cap H^*$ is at least d^*. Intuitively, $D_\ell \cap H^*$ is a subset that has "enough f-value" as well as a "good" density.

We may now combine the statements of Claim 4 and Claim 5 to get the following chain of inequalities:

$$f(D_\ell) \overset{Claim\ 4}{\geqslant} f(D_\ell \cap H^*) + \frac{d^*}{2}|D_\ell| - \frac{d^*}{2}|D_\ell \cap H^*| \overset{Claim\ 5}{\geqslant} \frac{d^*}{2}|D_\ell| + \frac{d^*}{2}|H^*|$$

Consider D'_ℓ: this is obtained from D_ℓ by adding suitably many elements to make D_ℓ feasible. Let r be the minimum *number* of elements to be added to D_ℓ so as to make it feasible. Since H^* is a feasible solution too, clearly, $r \leqslant |H^*|$. With this motivation, we define the *Extension Problem* for a matroid $\mathcal{M}$. The input is a matroid $\mathcal{M} = (U, \mathcal{I})$ and a subset $A \subseteq U$. The goal is to find a subset T of minimum cardinality such that $A \cup T \in \mathcal{I}$. Lemma 9 shows that we can find such a subset T in polynomial time. Thus, we would have that:

$$d(D'_\ell) = \frac{f(D'_\ell)}{|D_\ell| + r} \geqslant \frac{f(D_\ell)}{|D_\ell| + r} \geqslant \frac{f(D_\ell)}{|D_\ell| + |H^*|} \geqslant d^*/2$$

and we are done with the proof of Theorem 1, modulo the proof of Lemma 9.

We proceed to present the lemma and its proof:

▶ **Lemma 9.** *The Extension Problem for matroid $\mathcal{M}$ and subset A can be solved in polynomial time.*

Proof. The proof considers the *base polyhedron* of the matroid (see the text by Schrijver [17]). We will have a variable x_i for each element $i \in U \setminus A$, where $x_i = 1$ would indicate that we pick the element i in our solution T. For brevity, we will also maintain a variable y_i that indicates whether i is *absent* from the solution T. Thus for every i, we will maintain that $x_i + y_i = 1$. Given an arbitrary set S, we will let $r(S)$ denote the *rank* of the subset S in the matroid $\mathcal{M}$.

The following is a valid integer program for the Extension Problem (where $y(S)$ is shorthand for $\sum_{i \in S} y_i$). The linear program to the right is the relaxation of the integer program, and with variables x_i eliminated.

$$\text{IP}_1: \quad \begin{array}{lll} \min & \displaystyle\sum_{i \in U} x_i & \\ \text{s.t.} & x_i + y_i = 1 & \text{for all } i \in U \\ & y(S) \leqslant r(S) & \text{for all } S \subseteq U \\ & x_i = 1 & \text{for all } i \in A \\ & x_i, y_i \in \{0,1\} & \text{for all } i \in U\ . \end{array} \qquad \text{LP}_1: \quad \begin{array}{lll} \min & \displaystyle\sum_{i \in U} (1 - y_i) & \\ \text{s.t.} & y(S) \leqslant r(S) & \text{for all } S \subseteq U \\ & y_i = 0 & \text{for all } i \in A \\ & y_i \geqslant 0 & \text{for all } i \in U\ . \end{array}$$

The linear program LP_1 can also be formulated as a maximization question. To be precise, let $\text{VAL}(\text{LP}_1)$ denote the *value* of the program LP_1. Then $\text{VAL}(\text{LP}_1) = |U| - \text{VAL}(\text{LP}_2)$, where LP_2 is as follows:

$$\text{LP}_2: \quad \begin{array}{lll} \max & \displaystyle\sum_{i \in U} y_i & \\ \text{s.t.} & y(S) \leqslant r(S) & \text{for all } S \subseteq U \\ & y_i = 0 & \text{for all } i \in A \\ & y_i \geqslant 0 & \text{for all } i \in U\ . \end{array}$$

Now, by folklore results in matroid theory (cf. [17]), we have that solutions to LP_2 are integral and can be found by a greedy algorithm. Thus, we can solve IP_1 in polynomial time, and this proves the statement of the Lemma. ◀

4 Proof of Theorem 3

We will present the proof for the case of the graph density function, i.e. where $f(S) = |E(S)|$. The proof for arbitrary f will require a passage to the Lovász Extension $\mathcal{L}_f(x)$ of a set function $f(S)$ and is deferred to the full version [6].

We will *augment* the LP that Charikar [7] uses to prove that graph density is computable in polynomial time. Given a graph $G = (V, E)$, there are edge variables y_e and vertex variables x_i in the LP. We are also given an auxiliary *dependency* digraph $D = (V, \vec{A})$ on the vertex set V. In the augmented LP, we also have constraints $x_i \leqslant x_j$ if there is an arc from i to j in the digraph $D = (V, \vec{A})$. The DENdep problem is modelled by the linear program LP_3.

$$\text{LP}_3: \quad \begin{aligned} \max \quad & \sum_{e \in E} y_e \\ \text{s.t.} \quad & \sum_i x_i = 1 \\ & y_e \leqslant x_i && \forall e \sim i, e \in E \\ & x_i \leqslant x_j && \forall (i,j) \in \vec{A} \\ & x_i \geqslant 0 && \forall i \in V(G) \end{aligned} \qquad \text{CP}_1: \quad \begin{aligned} \max \quad & \sum_{e=(i,j) \in E} \min\{x_i, x_j\} \\ \text{s.t.} \quad & \sum_i x_i = 1 \\ & x_i \leqslant x_j && \forall (i,j) \in \vec{A} \\ & x_i \geqslant 0 && \forall i \in V(G) \end{aligned}$$

Suppose we are given an optimal solution H^* to the DENdep problem. Let $\text{VAL}(\text{LP}_3)$ denote the feasible value of this LP: we will prove that $\text{VAL}(\text{LP}_3) = d(H^*)$.

$\text{VAL}(\text{LP}_3) \geqslant d(H^*)$:

We let $|H^*| = \ell$, and $x_i = 1/\ell$ for $i \in H^*$, and 0 otherwise. Likewise, we set $y_e = 1/\ell$ for $e \in E(H^*)$, and 0 otherwise. Note that H^* is feasible, so if $a \in H^*$ and $(a, b) \in \vec{A}$, then it also holds that $b \in H^*$. We may check that the assignment x and y is feasible for the LP. So, $d(H^*) = \frac{|E(H^*)|}{\ell}$ is achieved as the value of a feasible assignment to the LP.

$\text{VAL}(\text{LP}_3) \leqslant d(H^*)$:

In the rest of the proof, we will prove that there exists a subgraph H such that $\text{VAL} \leqslant d(H)$. First, it is easy to observe that in any optimal solution of the above LP, the variables y_e will take the values $\min\{x_i, x_j\}$ where $e = (i, j)$. Thus, we may eliminate the variables y_e from the program LP_3 to obtain the program CP_1. We claim that CP_1 is a convex program. Given two concave functions, the min operator preserves concavity. Thus, the objective function of the above modified program is concave. Hence we have a convex program: here, the objective to be *maximized* is concave, subject to linear constraints. We may solve the program CP_1 and get an output optimal solution x^*. Relabel the vertices of V such that the following holds: $x^*_1 \geqslant x^*_2 \geqslant \cdots \geqslant x^*_n$. If there are two vertices with (modified) indices a and b where $a < b$ and there is an arc $(a, b) \in \vec{A}$, then we have the equalities $x^*_a = x^*_{a+1} = \cdots = x^*_b$. We will replace the inequalities in the program CP_1 as follows:

$$\text{LP}_4: \quad \begin{aligned} \max \quad & \sum_{e=(i,j) \in E : i<j} x_j \\ \text{s.t.} \quad & \sum_i x_i = 1 \\ & x_i \geqslant x_{i+1} && \text{for all } i \in \{1, 2, \cdots, (n-1)\} \\ & x_n \geqslant 0 && . \end{aligned}$$

where some of the inequalities $x_i \geqslant x_{i+1}$ may be equalities if there is an index a with $a \leqslant i$ and an index b with $b \geqslant (i+1)$ such that $(a, b) \in \vec{A}$. Note also that because of the ordering of the variables of this LP, the objective function also simplifies and becomes a linear function. Clearly x^* is a feasible solution to this LP. Thus the value of this LP is no less than the

value of CP_1. Consider a BFS x to LP_4. The program LP_4 has $(n+1)$ constraints, and n variables. Given the BFS x, call a constraint *non-tight* if it does not hold with equality under the solution x. Thus, there may be at most one *non-tight* constraint in LP_4. In other words, there is at most one constraint $x_i \geqslant x_{i+1}$ that is a strict inequality. This, in turn, implies that all the non-zero values in x are equal. Let there be ℓ such non-zero values. From the equality $\sum_i x_i = 1$, we get that each non-zero $x_i = 1/\ell$. Let H denote the set of indices i that have $x_i > 0$. The objective value corresponding to this BFS x is $|E(H)|/\ell = d(H)$.

Thus we have proven that $d(H) \geqslant \text{VAL}(LP_4) \geqslant \text{VAL}(CP_1) = \text{VAL}(LP_3)$, as required. This completes the proof of Theorem 3.

Remarks about the proof:

- We remark that the objective in the convex program CP_1 is precisely the Lovász Extension $\mathcal{L}_f(x)$ for the specific function $f = |E(S)|$. Thus our proof shows that the LP provided by Charikar [7] is precisely the Lovász Extension for the supermodular function $|E(S)|$.
- Note that there are other proofs possible for this result. For instance, one can follow the basic argument of Charikar to show that LP_3 satisfies $d(H^*) = \text{VAL}(LP_3)$. The proof we provide above is new, and is inspired by the work of Iwata and Nagano [13].
- Via our proof, we also prove that any BFS for the basic graph density LP has the property that all the non-zero values are equal. This fact is not new: it was proven by Khuller and Saha [14] but we believe our proof of this fact is more transparent.

5 Acknowledgements

We gratefully acknowledge helpful discussions with Aman Dhesi, Raghav Kulkarni and Amit Kumar about the topic.

References

1 N. Alon and J. H. Spencer. *The Probabilistic Method*. Wiley, New York, 1992.
2 R. Andersen and K. Chellapilla. Finding dense subgraphs with size bounds. In *WAW*, pages 25–37, 2009.
3 A. Bhaskara, M. Charikar, E. Chlamtac, U. Feige, and A. Vijayaraghavan. Detecting high log-densities: an $o(n^{1/4})$ approximation for densest k-subgraph. In *STOC*, pages 201–210, 2010.
4 S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
5 G. Călinescu, C. Chekuri, M. Pál, and J. Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.*, 40(6):1740–1766, 2011.
6 V. T. Chakaravarthy, N. Modani, S. R. Natarajan, S. Roy, and Y. Sabharwal. Density functions subject to a co-matroid constraint. *CoRR*, abs/1207.5215, 2012.
7 M. Charikar. Greedy approximation algorithms for finding dense components in a graph. In *APPROX*, pages 84–95, 2000.
8 Y. Filmus and J. Ward. A tight combinatorial algorithm for submodular maximization subject to a matroid constraint. In *FOCS*, 2012.
9 A. Gajewar and A. Das Sarma. Multi-skill collaborative teams based on densest subgraphs. In *SDM*, pages 165–176, 2012.
10 M. X. Goemans and J. A. Soto. Symmetric submodular function minimization under hereditary family constraints. *CoRR*, abs/1007.2140, 2010.
11 A. V. Goldberg. Finding a maximum density subgraph. Technical report, UC Berkeley, 1984.
12 M. Grötschel, L. Lovász, and A. Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1(2):169–197, 1981.

13 S. Iwata and K. Nagano. Submodular function minimization under covering constraints. In *FOCS*, pages 671–680, 2009.

14 S. Khuller and B. Saha. On finding dense subgraphs. In *ICALP*, 2009.

15 B. Saha, A. Hoch, S. Khuller, L. Raschid, and X-N. Zhang. Dense subgraphs with restrictions and applications to gene annotation graphs. In *RECOMB*, pages 456–472, 2010.

16 A. Schrijver. A combinatorial algorithm minimizing submodular functions in strongly polynomial time. *J. Comb. Theory, Ser. B*, 80(2):346–355, 2000.

17 A. Schrijver. *Combinatorial Optimization - Polyhedra and Efficiency.* Springer, 2003.

18 D.M. Topkis. *Supermodularity and Complementarity.* Princeton University Press, 1998.