Discrete Applied Mathematics 250 (2018) 252–261

Contents lists available at ScienceDirect

Discrete Applied Mathematics

journal homepage: www.elsevier.com/locate/dam

Complexity and approximations for submodular minimization problems on two variables per inequality constraints

Dorit S. Hochbaum

Department of Industrial Engineering and Operations Research, University of California, Berkeley, United States

ARTICLE INFO

Article history:
Received 1 April 2017
Received in revised form 28 March 2018
Accepted 6 April 2018
Available online 9 May 2018

Keywords:
Submodular optimization
Approximation algorithms
Submodular closure
Monotone constraints

ABSTRACT

We demonstrate here that submodular minimization (SM) problems subject to constraints containing up to two variables per inequality, SM2, are 2-approximable in polynomial time and a better approximation factor cannot be achieved in polynomial time unless NP=P. The 2-approximation holds when either the constraints have the round-up property (the rounding up of a feasible fractional solution is feasible) or, if the constraints do not have this property, for monotone submodular functions. The submodular minimization or supermodular maximization on constraints where the coefficients of the two variables in each constraint are of opposite signs (*monotone* constraints) is solvable in polynomial time. The 2-approximability and the polynomial time solvability for monotone constraints hold also for multi-sets that contain elements with integer multiplicity greater than 1, except that the running time is then pseudo-polynomial in that it depends on the range of the variables. This complexity cannot be improved unless NP=P.

Our results indicate that SM2 problems are not much harder than the respective linear integer problems on two variables per constraint: For monotone constraints both problems are polynomial time solvable, and for non-monotone NP-hard problems, both problems have 2-approximation algorithms. For SM2 problems the factor 2 approximation is provably best possible, whereas for the respective linear integer problems it has not been established that the factor 2 is best possible, but this has been conjectured. On the other hand, for SM2 problems where the two variables constraints' coefficients form a totally unimodular constraint matrix, the linear integer optimization problem is solved in polynomial time, whereas the submodular optimization is proved here to be NP-hard.

The submodular minimization NP-hard problems for which our general purpose 2-approximation algorithm applies include submodular-vertex cover, submodular-2SAT, submodular-min satisfiability, submodular-edge deletion for clique, submodular-node deletion for biclique and others.

1. Introduction

We demonstrate here that constrained submodular minimization problems, where each constraint has at most two variables, (SM2), are 2-approximable in polynomial time. This holds if either the constraints have the round-up property, or if the submodular function is monotone. This approximation factor of 2 for SM2 is provably best possible unless NP=P. This result is shown here to apply for multi-sets submodular minimization as well. For SM2 with the coefficients of the two variables in each constraint having opposite signs, monotone constraints, the submodular minimization problem is known

E-mail address: hochbaum@ieor.berkeley.edu.

https://doi.org/10.1016/j.dam.2018.04.012

to be solvable in (strongly) polynomial time. Multi-sets submodular minimization on monotone constraints are shown to be solvable in pseudopolynomial time that depends on the largest multiplicity of a set, and this running time cannot be improved (by removing this dependence) unless NP=P.

A nonnegative function f defined on the subsets of a set V is said to be *submodular* if it satisfies for all $X, Y \subseteq V$, $f(X) + f(Y) \geq f(X \cap Y) + f(X \cup Y)$. A submodular function f is said to be *monotone* if $f(S) \leq f(T)$ for any $S \subseteq T$. A binary vector of dimension $n = |V|$, $\mathbf{x} = \{x_i\}_{i=1}^n$, is associated with a corresponding subset of V, $X = \{i \in V | x_i = 1\}$. The vector $\mathbf{x}$ is then said to be the *characteristic vector* of the set X.

The SM2 problem addressed here is

$$\begin{array}{lll} & \min & f(X) \\ (\text{SM2}) & \text{subject to} & a_{ij}x_i + b_{ij}x_j \geq c_{ij} \text{ for all } (i,j) \in A \\ & & x_j \in \{0,1\} \quad \text{ for all } j \in V, \end{array}$$

where a_{ij}, b_{ij} and c_{ij} are any real numbers and A is a set of pairs (including singletons, and also allowing multiple copies of the same pair) defining the constraints. Our main results are that any SM2 problem, with constraints that satisfy the *round up* property or with monotone submodular objective function, is 2-approximable in polynomial time and this approximation factor cannot be improved unless NP=P. A set of constraints satisfies the round up property if any feasible half integer solution can be rounded *up* to an integer feasible vector. Vertex cover and all covering constraints satisfy the round up property, but non-covering problems, such as minimum (weighted) node deletion so remaining graph is a maximum clique, satisfy the round up property as well. The formulations and discussion of the properties of these and other SM2 problems are given in Section 3.1.

An inequality constraint in up to two variables, $a_{ij}x_i - b_{ij}x_j \geq c_{ij}$ is called *monotone* if a_{ij} and b_{ij} have the same signs. (This concept of monotonicity is unrelated to the monotonicity of a submodular function.) The feasible solutions of monotone constraints form a distributive lattice (a *ring family*), and submodular minimization over a ring family is known to be polynomial time solvable. Submodular minimization over a ring family was first shown to be solved in strongly polynomial time by Grötschel, Lovász, and Schrijver in [4]. Combinatorial strongly polynomial algorithms were given later by Schrijver, [21] and by Iwata, Fleischer, and Fujishige [13]. The current fastest strongly polynomial algorithms on a ring family were devised by Orlin [20], and later by Iwata and Orlin, [15].

In contrast to the polynomial solvability of submodular minimization over monotone constraints, the submodular minimization (or supermodular maximization) over constraints with *totally unimodular* constraints matrix is proved here to be NP-hard. This indicates that monotone constraints form a more significant structure than totally unimodular constraints, in terms of complexity, for submodular (supermodular) minimization (maximization).

The results here apply also to submodular minimization on *multi-sets*, (SM2-multi). These are submodular functions defined on sets containing elements with multiplicity greater than 1. A nonnegative integer vector $\mathbf{x} \in \mathcal{Z}^n$ is the characteristic vector of a multiset $X = \{(i, q_i) | x_i = q_i\}$, where $(i, q_i) \in X$ means that X contains element i q_i times, for positive integers q_i. All properties of submodular functions extend to multi-sets, with the generalized definition of containment, $X_1 \subseteq X_2$, meaning that for all $(i, q_i) \in X_1, (i, q_i') \in X_2$ with $q_i \leq q_i'$. The problem of constrained submodular minimization on multi-sets is then $\min\{f(X) | A\mathbf{x} \geq \mathbf{b},\ \mathbf{0} \leq \mathbf{x} \leq \mathbf{u}, \mathbf{x} \in \mathcal{Z}^n\}$ for $\mathbf{u}$ the vector of upper bounds on the multiplicities of the elements. Let the upper bound on the multiplicity of element i be u_i. The formulation of SM2-multi is then,

$$\begin{array}{lll} & \min & f(X) \\ (\text{SM2-multi}) & \text{subject to} & a_{ij}x_i + b_{ij}x_j \geq c_{ij} \text{ for all } (i,j) \in A \\ & & 0 \leq x_j \leq u_j \quad \text{ and integer, for all } j \in V. \end{array}$$

The respective 2-approximations or polynomial time algorithms for multi-sets are shown to be attained in time polynomial in $U = \max_{j=1,\ldots,n} u_j$. The dependence of the run time on U cannot be removed (to, say, logarithmic dependence) unless NP=P. This is because finding a feasible solution to a monotone integer linear program on constraints with up to two variables per inequality was shown to be NP-hard, [18]. The pseudopolynomial run time of the algorithms for integer linear optimization on monotone constraints, in [11], and for SM2-multi here, indicate that these two problems are in fact *weakly* NP-hard.

1.1. Related research

A prominent example of SM2 is the submodular vertex cover, SM-vertex cover, where the constraint matrix A contains exactly two 1s per row and $\mathbf{b}$ is a vector of 1s.

Approximating SM-vertex cover has been a subject of previous research work. Three different 2-approximation algorithms were devised for the problem: Koufogiannakis and Young [17] devised approximations for SM-“covering” problems with monotone submodular objective function. Their approach is based on the frequency technique (called maximal dual feasible technique in [7] Ch. 3). Their algorithm is a 2-approximation for the SM-vertex cover for *monotone* submodular objective function. Goel et al. [3] devised a 2-approximation algorithm for SM-vertex cover with monotone submodular function which involves solving a relaxation with the Ellipsoid method with a separation algorithm equivalent to a submodular minimization problem. Goel et al. further proved that submodular vertex cover is inapproximable within a factor better than 2. Iwata and Nagano in [14] presented a 2-approximation algorithm for the SM-vertex cover, and addressed the SM-set cover and the SM-edge cover. Their algorithm does not require the submodular function to be monotone. Iwata and Nagano’s technique relies on using Lovász extension of submodular minimization to convex minimization.

1.2. Contributions here

We devise here a unified framework for generating 2-approximation algorithms for all NP-hard SM2 and SM2-multi problems with constraints that have the round-up property, or, if round-up does not hold, for monotone submodular functions. Unlike previous results, these algorithms do not require solving a linear programming relaxation or using the Lovász extension convex optimization, yet run in strongly polynomial time (Theorem 1). In particular, our algorithm is a 2-approximation algorithm for the SM-vertex cover (without restriction of submodular function's monotonicity). Other NP-hard submodular minimization problems for which we derive 2-approximations include the following: the submodular min-2SAT, minimum node deletion biclique, minimum edge deletion clique, and min SAT. Among these only the SM-min-2SAT requires monotone submodular objective function.

Our results shed some light on the relationship between submodular minimization and integer linear minimization. Obviously submodular minimization can only be harder than integer linear minimization. Indeed, as we show in Section 5 Theorem 3, SM-vertex cover on bipartite graphs is an NP-hard problem, whereas the linear vertex cover on bipartite graphs is polynomial time solvable. This demonstrates that submodular minimization over a totally unimodular constraint matrix, is NP-hard, and hence strictly harder than the respective integer linear optimization. On the other hand, for two variables per inequality constraints, submodular minimization is not harder than the respective integer linear optimization in that, like in the respective linear case, there is a unified 2-approximation polynomial time algorithm for any NP-hard SM2. That 2-approximation algorithm is a generalization of the unified 2-approximation algorithm, devised in [10], for any NP-hard integer linear minimization on two variables per inequality constraints. Our results indicate that there is a difference in the approximability of the submodular versus the linear case: Hochbaum conjectured in [6] that the lower bound on approximability of the vertex cover problem is $2-\epsilon$. Yet, the tightest lower bound established to date on the approximability of vertex cover is 1.36 [1], whereas for the submodular monotone analog it is $2-\epsilon$ [3]. The latter closes the gap between the lower bound and the factor 2 approximation, conjectured in [6], only for submodular vertex cover but not for linear vertex cover. Since the vertex cover problem is as general as the entire class of minimization problems on two variables per inequality (a proof is provided in [8]), it follows that *for all* NP-hard submodular minimization problems on two variables per inequality, the polynomial time 2-approximation algorithms presented *cannot be improved*.

Summary of contributions.

1. We present here the first known polynomial time 2-approximation algorithms for a large family of constrained submodular minimization problems, SM2 and SM2-multi, with constraints that contain at most two variables per inequality. We provide a direct and simple proof of the 2-approximability using only the properties of submodular functions improving on past work for SM-vertex cover. A previous 2-approximation of SM-vertex cover, [14], used a construction based on Lovász convex extension of submodular functions which involved an "intermediate" convex formulation that is shown here to be unnecessary.
2. SM2-multi problems on monotone constraints are shown to be solved in polynomial time that depends on the multiplicity of the sets for either submodular minimization or supermodular maximization. This complexity cannot be improved, as the linear version of SM2 on monotone constraints is (weakly) NP-hard.
3. Submodular minimization over constraints with coefficients' matrix that is totally unimodular is proved to be NP-hard in Theorem 3. In particular, submodular vertex cover on bipartite graphs is an NP-hard problem. This proof provides additional evidence to the difficulty of generalizing linear optimization, or approximation, algorithms to the submodular context. But the closely related problems of bipartite-submodular vertex cover, and bipartite-supermodular independent set, on bipartite graphs, are shown to be polynomial time solvable. (Bipartite-submodular and bipartite-supermodular functions are defined in Section 2.)
4. The 2-approximation factor is shown to be best possible approximation factor for all SM2 problems. This follows from the lower bound proof on the approximability of SM-vertex cover of Goel et al. [3], proving that SM-vertex cover cannot be approximated in polynomial time within a factor of $2-\epsilon$, for any $\epsilon > 0$, unless NP=P.

2. Notations and preliminaries

Given an $m \times n$ real matrix $A^{(2)}$ where each row contains at most two non-zeros, the SM2 problem can be written as $\min\{f(X)|A^{(2)}\mathbf{x} \geq \mathbf{b},\ \mathbf{0} \leq \mathbf{x} \leq \mathbf{u}, \text{integer}\}$ for $u_j = 1$ for $j = 1, 2, \ldots, n$. For general positive values of u_js the problem is called SM2-multi.

An important class of SM2 has the two non-zeros in each row of $A^{(2)}$ of opposite signs in which case the constraints are said to be monotone. SM2 problems on monotone constraints are shown here to be polynomial time solvable.

A feasible SM2 (that has a feasible integer solution) is said to have the *round-up* property, if for any given feasible half integral solution vector $\mathbf{x}^{\frac{1}{2}}$ there exists an integer feasible solution $\mathbf{x}^{\text{int}}$ such that $\mathbf{x}^{\frac{1}{2}} \leq \mathbf{x}^{\text{int}}$. We refer to an SM2 with such set of constraints as *round-up-SM2*. Notice that the round-up property depends on the constraints only. All covering matrices, where the inequalities are $\geq$ constraints and all coefficients are non-negative, have the round-up property. But there are also non-covering matrices that have the round-up property.

For a directed graph $G = (V, A)$ a set of nodes $D \subseteq V$ is said to be *closed* if all the successors (or predecessors) of the nodes in D are also in D. In other words, the *transitive closure* of D, forming all the nodes reachable from nodes of D along a directed

Table 1
Examples of 2-approximable and polynomial time solvable SM-problems.

SM-problem name	Monotone constraints	Round-up property	Submodular objective $f()$	Aprrox factor
Vertex cover	No	Yes	any	2
Complement of max-clique	No	Yes	any	2
Node-deletion bi-clique	No	Yes	any	2
Min-satisfiability	No	Yes	any	2
Min-2SAT	No	No	monotone	2
SM-closure	Yes	NA	any	1

path in G, is equal to D. We call linear constraints on two variables of the form $x \leq y$ *closure constraints*. Closure constraints are obviously monotone.

A directed graph is said to be a *closure graph* if all arcs are of infinite capacity. An s, t-graph $G_{st} = (V \cup \{s, t\}, A \cup A_s \cup A_t)$ with A_s the set of arcs adjacent to source s and A_t a set of arcs adjacent to sink t, is called a *closure s, t-graph* if all finite capacity arcs are either in A_s or A_t.

A function f is said to be *supermodular* if for all $X, Y \subseteq V, f(X) + f(Y) \leq f(X \cap Y) + f(X \cup Y)$.

A function f defined on a bipartite graph $G = (V_1 \cup V_2, E)$ is *bipartite-submodular* if there exist two submodular functions f_1 and f_2 defined on V_1 and V_2, respectively, such that, for $X_1 = X \cap V_1, X_2 = X \cap V_2$,

$$f(X) = f_1(X_1) + f_2(X_2).$$

A bipartite-supermodular function is defined analogously.

3. Some of the submodular minimization problems solved here

Table 1 lists several SM2 submodular problems for which the algorithmic framework devised here applies. In the table it is noted, for each problem, whether it has the round-up property or not. Problems that are polynomial time solvable with the technique here are indicated with an approximation factor of 1.

We now provide the formulations and discussion of properties for each of the problems. The SM-closure problem is discussed in Section 4.2.

3.1. Formulations of several SM2 problems

Vertex cover. The vertex cover problem is to find a subset of nodes in a graph $G = (V, E)$ so that each edge in E has at least one endpoint in the subset.

$$\text{(SM-vertex-cover)} \quad \begin{array}{ll} \min & f(X) \\ \text{subject to} & x_i + x_j \geq 1 \;\text{ for all }\; [i, j] \in E \\ & x_i \;\text{ binary }\; i \in V. \end{array}$$

The submodular vertex cover problem was shown to have a 2-approximation by Iwata and Nagano, [14] for general submodular $f()$. SM-vertex cover obviously has the round-up property and therefore the 2 approximation described here applies to any general submodular objective function. When the graph $G = (V_1 \cup V_2, E)$ is bipartite, the SM-vertex-cover is still NP-hard (Section 5), but for a bipartite-submodular objective, $f(X_1 \cup X_2) = f_1(X_1) + f_2(X_2)$, for $X_i \subseteq V_i, i = 1, 2$, the problem is polynomial time solvable.

Complement of maximum clique. The maximum clique problem is a well known optimization problem that is notoriously hard to approximate, e.g. Hastad, [5]. The problem is to find in a graph the largest set of nodes that forms a clique—a complete subgraph.

An equivalent statement of the clique problem is to find the complete subgraph which maximizes the number (or more generally, sum of weights) of the edges in the subgraph. When the weight of each edge is 1, then there is a clique of size k if and only if there is a clique on $\binom{k}{2}$ edges. The inapproximability result for the node version extends trivially to this edge version as well.

The complement of this edge variant of the maximum clique problem is to find a minimum weight of edges to *delete* so the remaining subgraph induced on the non-isolated nodes is a clique. We define here the SM-edge deletion for clique. For a graph $G = (V, E)$, the submodular function $f(Z)$ is defined on the set of variables z_{ij} for all edges $[i, j] \in E$. Let x_j be a variable that is 1 if node j is in the clique, and 0 otherwise. Let z_{ij} be 1 if edge $[i, j] \in E$ is *deleted*.

$$\text{(SM-Clique-edge-delete)} \quad \begin{array}{ll} \min & f(Z) \\ \text{subject to} & 1 - x_i \leq z_{ij} \quad [i, j] \in E \\ & 1 - x_j \leq z_{ij} \quad [i, j] \in E \\ & x_i + x_j \leq 1 \quad [i, j] \notin E \\ & x_j \;\text{ binary }\; j \in V \\ & z_{ij} \;\text{ binary }\; [i, j] \in E. \end{array}$$

This formulation has two variables per inequality and therefore it is SM2. This SM2 is of covering-type because a fraction value of z_{ij} can be rounded up, (note that the respective x_i and x_j may be rounded up or down without affecting the objective function), and therefore the 2-approximation algorithm applies to any submodular function. The gadget and network for solving the monotonized SM-Clique-edge-delete problem are given in detail in [9].

Node deletion biclique. Here we consider the submodular minimization of node deletion in a bipartite graph $(V_1 \cup V_2, E)$ so that the subgraph induced on the remaining nodes forms a biclique (a complete bipartite graph). This problem is identical to the submodular vertex cover on a bipartite graph, proved in Section 5 to be NP-hard. The linear version of this problem is polynomial time solvable, [8]. In the formulation given below x_i assumes the value 1 if node i is deleted from the bipartite graph, and 0 otherwise.

$$\text{(SM-Biclique-node-delete)} \quad \begin{array}{ll} \min & f(X) \\ \text{subject to} & x_i + x_j \geq 1 \text{ for edge } \{i,j\} \notin E \; i \in V_1, j \in V_2 \\ & x_j \in \{0,1\} \text{ for all } j \in V_1 \cup V_2. \end{array}$$

Minimum satisfiability. In the problem of *minimum satisfiability*, MINSAT, we are given a CNF satisfiability formula. The aim is to find an assignment satisfying the smallest number of clauses, or the smallest weight collection of clauses. The MINSAT problem was introduced by Kohli et al. [16] and was further studied by Marathe and Ravi [19]. The problem is NP-hard even if there are only two variables per clause, [16].

The submodular minimum satisfiability SM-MINSAT problem can be formulated as SM2, and thus 2-approximable: Choose a binary variable y_j for each clause C_j and a binary variable x_i for each literal. Let $S^+(j)$ be the set of variables that appear unnegated and $S^-(j)$ those that are negated in clause C_j. The following formulation of MINSAT has two variables per inequality and is thus a special case of SM2:

$$\text{(SM-MINSAT)} \quad \begin{array}{lll} \min & f(Y) & \\ \text{subject to} & y_j \geq x_i & \text{for } i \in S^+(j) \text{ for clause } C_j \\ & y_j \geq 1 - x_i & \text{for } i \in S^-(j) \text{ for clause } C_j \\ & x_i, y_j \text{ binary for all } i, j. & \end{array}$$

It is interesting to note that the formulation is monotone when for all clauses C_j $S^+(j) = \emptyset$ or in all clauses $S^-(j) = \emptyset$. (In the latter case, we need to transform the x variables to x' with $x' = -x$.) Indeed in these instances the boolean expression is uniform and the problem is trivially solved setting all variables to FALSE in the first case, or to TRUE in the latter case.

Although SM-MINSAT is not of covering type, it is nevertheless a round-up SM2, as can be easily verified, and therefore there is no restriction on $f()$ for the 2-approximation algorithm to apply.

MIN-2SAT. The MIN-2SAT problem is defined for a 2SAT CNF with each clause containing at most two variables. The goal is to find a truth assignment, satisfying all clauses, with the least weight collection of variables that are set to true. Although finding a satisfying assignment to a 2SAT can be done in polynomial time (Even et al. [2]), finding a satisfying assignment that *minimizes* the number, or the weight, of the true variables is NP-hard.

Let X be the set of true variables, and $x_i = 1$ if the ith variable is set to true, and 0 otherwise.

$$\text{(SM-MIN-2SAT)} \quad \begin{array}{lll} \min & f(X) & \\ \text{subject to} & x_i + x_j \geq 1 & \text{for clause } (x_i \vee x_j) \\ & x_i - x_j \geq 0 & \text{for clause } (x_i \vee \bar{x}_j) \\ & x_i + x_j \leq 1 & \text{for clause } (\bar{x}_i \vee \bar{x}_j) \\ & x_i \text{ binary for all } i = 1, \ldots, n. & \end{array}$$

Each constraint here has up to two variables and thus this problem is in the class SM2. The SM-MIN-2SAT does *not* have the round-up property and therefore the resulting 2-approximation algorithm applies for $f()$ *monotone* submodular function.

Additional problems related to finding maximum biclique – a clique in a bipartite graph – are also formulated in two variables per inequality, in [8]. For these problems, all the corresponding submodular minimization problems are either monotone, and thus solved in polynomial time, or have a polynomial time 2-approximation.

4. 2-approximation for SM2 and SM2-multi

We show first, in Section 4.1, that the constraints of a general SM2 and SM2-multi can be transformed to monotone constraints. This transformation is a relaxation in the sense that any feasible solution to the original constraints is also feasible for the monotonized constraints, but a feasible solution to the monotonized constraints maps to a fractional solution and therefore will not satisfy the integrality requirement in the original system of constraints. Although fractional, the feasible solution to the original problem derived from the monotonized constraints is guaranteed to be an integer multiple of half.

Next we show, in Section 4.2, that SM2-multi on monotone constraints is equivalent to a problem we call SM-closure and then solved in (strongly) polynomial time in the size of the problem. The derivation of the equivalent SM-closure is done with an algorithm which is referred to as *binarizing*. Although SM2 on monotone constraints, and on binary variables, is known to be solved in polynomial time (it is submodular minimization over a ring family), the binarizing process converts the problem to SM-closure and in that highlights the link to the linear objective case, where the monotone problem is set to be equivalent to the (binary) closure problem, and solved with a minimum cut procedure.

4.1. Transforming general SM2 and SM2-multi to their monotonized version

General SM2s (and SM2-multi) are transformed to monotone SM2s using a process we refer to as *monotonizing*. The monotonizing process is described here for the constraints of general SM2 and SM2-multi: We first duplicate the set of elements V and their respective characteristic vectors $\mathbf{x}$ to $\mathbf{x}^+$ and $\mathbf{x}^-$, so that x_j^+ assumes values in $\{0, 1, \ldots, u_j\}$ and x_j^- assumes values in $\{-u_j, \ldots, -1, 0\}$. For given vectors $\mathbf{x}^+$ and $\mathbf{x}^-$ the corresponding multisets are

$$X^+ = \{(i, p_i)|x_i^+ = p_i,\ i \in V\},\ \text{and}$$
$$X^- = \{(j, q_j)|x_j^- = -q_j,\ j \in V\}.$$

Each non-monotone inequality $a_{ij}x_i + b_{ij}x_j \geq c_{ij}$ is replaced by the following two inequalities:

$$a_{ij}x_i^+ - b_{ij}x_j^- \geq c_{ij}$$
$$-a_{ij}x_i^- + b_{ij}x_j^+ \geq c_{ij}.$$

In case the set of constraints contains, in addition to non-monotone constraints, also a subset of monotone constraints, each monotone inequality $a'_{ij}x_i - b'_{ij}x_j \geq c'_{ij}$ is replaced by the two inequalities:

$$a'_{ij}x_i^+ - b'_{ij}x_j^+ \geq c_{ij}$$
$$-a'_{ij}x_i^- + b'_{ij}x_j^- \geq c_{ij}.$$

It is easy to verify that setting $x_j = \frac{x_j^+ - x_j^-}{2}$ is feasible for the original inequalities, and $x_j = x_j^+ - x_j^-$ is feasible to the original inequalities multiplied by 2, $2a_{ij}x_i + 2b_{ij}x_j \geq 2c_{ij}$. We refer to the latter as the *doubled* inequalities.

The objective function is $\frac{1}{2}(f(X^+) + f(X^-))$, and the constraints are the doubled inequalities and the lower and upper bounds on the variables. The objective function is submodular as a sum of two submodular functions. This formulation is therefore SM2-multi over monotone constraints.

4.2. A polynomial time algorithm for solving SM2-multi on monotone constraints

As pointed out above, monotone constraints on binary variables form a ring family. For binary variables a monotone constraint of the form $a_{ij}x_i - b_{ij}x_j \geq c_{ij}$ may imply that the problem is infeasible, or that one of the variables is of fixed value in any feasible solution. This constraint is of non-trivial interest only if it is equivalent to $x_i \geq x_j$. Namely, that $x_j = 1$ implies that $x_i = 1$. To see this notice that if $x_j = 0$ implies that $x_i = 1$ (this happens when $1 \geq \frac{c_{ij}}{a_{ij}} > 0$) then $x_i = 1$ in any solution and therefore can be fixed and eliminated. If $x_j = 1$ implies that $x_i > 1$ then x_j must be fixed at 0 for any feasible solution. The constraints of the type $x_i \geq x_j$ are *closure* constraints, known also as *partial order* constraints.

The set of monotone constraints considered here has general lower and upper bounds on the variables that could be negative:

$$\begin{array}{ll}(\text{monotone constraints}) & a_{ij}x_i - b_{ij}x_j \geq c_{ij} \quad \forall (i,j) \in A, \\ & \ell_j \leq x_j \leq u_j \ \text{ for all } \quad j \in V.\end{array}$$

We show next that a set of monotone constraints is equivalent to a set of closure constraints on binary variables.

The linear optimization (either minimization or maximization) over closure constraints is called the closure problem. Solving the closure problem is equivalent to solving linear optimization over a ring family, as explained next. Consider first the maximum weight closure problem where the closure is in terms of successors. Let x_j be a binary variable that is 1 if node j is in the closure, and 0 otherwise. Let w_j be the weight of node j. Note that the problem is trivial if all w_j are positive (optimal solution is V), or if all w_j are negative (optimal solution is $\emptyset$). The problem formulation for a directed graph $G = (V, A)$ is

$$\begin{array}{lll}(\text{max-closure}) \ \max & \displaystyle\sum_{j \in V} w_j x_j & \\ \quad \text{subject to} & x_i \leq x_j \quad \forall (i,j) \in A, & \\ & x_j \ \text{binary} \quad j \in V. & \end{array}$$

To solve this linear problem we construct an associated s, t closure graph: Consider the partition on V to $V^+ = \{v \in V | w_v > 0\}$ and $V^- = \{v \in V | w_v \leq 0\}$, and note that both sets are non-empty for non-trivial problems. We add to the graph a node s and a set of arcs $\{(s, j)|j \in V^+\}$, where arc (s, j) is of capacity w_j. Next we add a node t and a set of arcs $\{(i, t)|i \in V^-\}$, where arc (i, t) is of capacity $-w_i$. All arcs in A are assigned infinite capacity. It is easy to see that the source set S for any finite capacity s, t-cut (S, T) in this graph is a closed set, and that the minimum capacity cut has a source set of maximum weight. Respectively, the sink set of the minimum cut T is closed with respect to predecessors, and it is of total minimum weight. Additional details on the closure problem can be found, e.g. in [12].

We comment that in contrast to submodular minimization SM2-multi over closure constraints that is solved in pseudopolynomial time, the convex separable minimum closure problem on integers (non-binary) is solvable in polynomial time using a parametric cut algorithm as proved in Hochbaum and Queyranne [12].

We call the submodular minimization over closure constraints, and in binary variables, the *SM-closure* problem. This problem is formulated for a submodular function $f(X)$ where $X \subseteq V$ as follows:

$$\begin{array}{lll} \text{(SM-closure)} & \min & f(X) \\ & \text{subject to} & x_i \leq x_j \quad \forall (i,j) \in A, \\ & & x_j \text{ binary} \quad j \in V. \end{array}$$

Note that SM-closure is interesting, or non-trivial, for non-monotone submodular functions. Solving SM-closure is done with any algorithm that minimizes a submodular function over a ring family. We show next that any SM2-multi on monotone constraints is equivalent to an SM-closure problem on a graph on $\sum u_i$ nodes.

To reduce SM2-multi to SM-closure, we use a process, introduced by Hochbaum and Naor [11], that maps a monotone constraint in integers $a_{ij}x_i - b_{ij}x_j \geq c_{ij}$ into an equivalent collection of closure constraints on binary variables. We refer to this procedure as *binarizing*. A sketch of the procedure is as follows: The integer variables are first replace by binary variables. (This step in not required for SM2 where the variables are already binary.) For each element i and $p = \ell_i + 1, \ldots, u_i$ we let $x_i^{(p)} = 1$ only if $x_i \geq p$. In particular the variable x_i can be written as a summation of binary variables $x_i = \ell_i + \sum_{p=\ell_i+1}^{u_i} x_i^{(p)}$. The following restriction then holds for all $p = \ell_i + 2, \ldots, u_i$: $x_i^{(p)} = 1 \Longrightarrow x_i^{(p-1)} = 1$. Note that $x_i^{(\ell_i)} \equiv 1$.

For a monotone constraint $a_{ij}x_i - b_{ij}x_j \geq c_{ij}$, let $q(p) \equiv \lceil \frac{c_{ij}+b_{ij}p}{a_{ij}} \rceil$. Assuming that $q(p) \leq u_i$ (otherwise update the upper bound on x_j to be $u_j := p - 1$, and if $p - 1$ is less than the lower bound on x_j, the problem is infeasible), then the monotone constraints can be equivalently written as follows:

$$\begin{array}{lll} x_j^{(p)} \leq x_i^{(q(p))} & \forall\, (i,j) \in A & \text{for } p = 1, \ldots, u_j \\ x_i^{(p)} \leq x_i^{(p-1)} & \forall\, i \in V & \text{for } p = 1, \ldots, u_i \\ x_i^{(p)} \in \{0, 1\} & \forall\, i \in V & \text{for } p = 1, \ldots, u_i. \end{array}$$

The constraints listed are closure constraints in an s, t-graph where for each element i there are up to u_i nodes and for every constraint $(i,j) \in A$ in the original problem there are up to $\min\{u_i, u_j\}$ arcs. Hence any SM2-multi problem on monotone constraints is equivalent to a SM-closure problem on a graph with $O(\sum_{i \in V} u_i)$ nodes, and is solvable in polynomial time in the size of the graph. For general range of the variables, this run time is pseudopolynomial and, as discussed in Section 1 it cannot be improved unless NP=P.

4.3. The 2-approximation algorithm

Given a non-monotone SM2-multi problem, we first monotonize it to generate a SM2-multi on monotone inequalities, as described in Section 4.1. The next step is to convert the resulting problem to an SM-closure problem on binary variable, as detailed in Section 4.2. We refer to the resulting SM-closure problem as *relaxed* SM2 (or SM2-multi). The relaxed SM2-multi problem is defined on binary variables, represented as nodes, and each inequality of the form $y' \leq z'$ corresponds to an arc (y', z') in the set of arcs A'. We let the set of binary variables that are generated from $\mathbf{x}^+$ vectors be denoted by V^+ and those that are generated from $\mathbf{x}^-$ vectors be V^-. Each node in the graph $G = (V^+ \cup V^-, A')$ indicates that the respective variable value is at its upper bound. Recall that since the lower bound variable is not in the graph, as the variable value is always greater or equal to the lower bound, for binary variables in $\{0, 1\}$ the selection of a node in the graph is equivalent to setting its value to 1, and for binary variables in $\{-1, 0\}$ the selection of a node in the graph is equivalent to setting its value to 0. A node $j \in V^+$ is in a closed set S implies that the respective binary variable $x_j^+ = 1$, and a node $i \in V^-$ is in S implies that the respective binary variable $x_i^- = 0$.

We denote by $x_i' = -x_i^-$ for all $i \in V^-$ so that $\mathbf{x}'$ is a binary vector taking values in $\{0, 1\}$. Let $X^+ = \{j \in V^+ | x_j^+ = 1\}$ and $X^- = \{j \in V^- | x_j' = 1\}$. That is, the vectors $\mathbf{x}^+$ and $\mathbf{x}'$ are the characteristic vectors of the sets X^+ and X^-. Let S^* be an optimal set minimizing the function $f()$ for the (original) SM2 formulation, and let $\mathbf{x}^*$ be the associated characteristic vector. On the graph $G = (V^+ \cup V^-, A')$, let S^{*+} and S^{*-} be the copies of S^* in V^+ and V^-, respectively. Then this is a feasible solution for the relaxed problem since $S^{*+} \cup (V^- \setminus S^{*-})$ is a closed set, and the vectors $\mathbf{x}^{*+} = \mathbf{x}'^* = \mathbf{x}^*$ defined on V^+ and V^- are the characteristic vectors of S^{*+} and S^{*-}. Therefore, setting $x_i = x_i^{*+} + x_i'^*$ is a feasible solution $\mathbf{x}$ for the "doubled" inequality constraints.

The submodular function $f()$ is defined on subsets of V and therefore the objective function of the SM-closure problem defined on the constructed graph, $g(X^+ \cup X^-) = f(X^+) + f(X^-)$, is well defined and, $f(X^+)$, $f(X^-)$ and $g(X^+ \cup X^-)$ are submodular functions.

Theorem 1. *Let $\tilde{X}^+ \subseteq V^+$ and $\tilde{X}^- \subseteq V^-$ be the sets minimizing $g()$ among all feasible pairs of sets for the relaxed SM2-multi. Let S^* be an optimal set minimizing the function $f()$ in the (original) SM2-multi formulation. Then, $2f(S^*) \geq f(\tilde{X}^+ \cup \tilde{X}^-)$.*

Proof. Let S^{*+} and S^{*-} be the copies of S^* in V^+ and V^-, respectively. Then,

$$\begin{aligned} 2f(S^*) &= f(S^{*+}) + f(S^{*-}) \geq g(\tilde{X}^+ \cup \tilde{X}^-) = f(\tilde{X}^+) + f(\tilde{X}^-) \\ &\geq f(\tilde{X}^+ \cup \tilde{X}^-) + f(\tilde{X}^+ \cap \tilde{X}^-) \geq f(\tilde{X}^+ \cup \tilde{X}^-). \end{aligned}$$

The first inequality holds since $\tilde{X}^+ \cup \tilde{X}^-$ is an optimal solution to the relaxed SM2-multi. The second inequality follows from the submodularity of the function f. $f(\tilde{X}^+ \cup \tilde{X}^-)$ is the value of our solution where an element is included if either one of its two copies is in $\tilde{X}^+$ or in $\tilde{X}^-$. The third and last inequality follows since $f()$ takes non-negative values. □

2-*approximation for round-up SM2-multi.*

Theorem 1 leads immediately to the 2-approximation result for round-up SM2-multi:

Lemma 1. *For round-up SM2-multi and a **general** non-negative submodular objective function $f()$ there is a polynomial time 2-approximation algorithm.*

Proof. If for an index j both variables x_j^+, x_j' are of value 1, then we set the value of x_j to be equal to 1. Let $V^1 = \{j \in V | x_j^+ = x_j' = 1\}$ be the set of such variables. If both x_j^+, x_j^- are of value 0, we let $x_j = 0$, and $V^0 = \{j \in V | x_j^+ = x_j' = 0\}$ is the set of these variables. The set of remaining variables, which have exactly one of x_j^+, x_j' equal to 1 and the other one equal to 0, is called $V^{\frac{1}{2}}$.

For round-up SM2-multi, the rounded solution is the set $V^1 \cup V^{\frac{1}{2}} = \tilde{X}^+ \cup \tilde{X}^-$. From Theorem 1 this is a 2-approximate solution for SM2-multi. □

We prove next the approximation result for SM problems without the round up property. The rounding in this case is shown to be guided by a feasible solution found by an algorithm equivalent to finding a truth assignment, if exists, to a 2SAT expression. A truth assignment for 2SAT, or determining that there is none, can be found using the linear time algorithm of [2]. To show how to find a feasible solution to a set of constraints in two variables we first demonstrate that a SM2-multi is equivalent to a respective SM-MIN-2SAT problem. This proof is based on the approach of binarizing first the non-monotone system. This result is the same as was derived for linear integer programming on constraints with two variables per inequality, in Hochbaum et al. [10]:

Theorem 2. *The set of SM2-multi constraints is equivalent to the constraints of SM-MIN-2SAT on at most nU binary variables and $O(mU)$ constraints, for $U = \max_{i \in V} u_i$, in that both have the same sets of feasible solutions.*

Proof. For a general constraint of the form, $a_{ki}x_i + a_{kj}x_j \geq c_k$, consider the case where both a_{ki} and a_{kj} are positive, and assume without loss of generality that $0 < c_k < a_{ki}u_i + a_{kj}u_j$. The other cases where one coefficient is negative (and the constraint is monotone), or both are negative, are similarly "binarized".

For every ℓ ($\ell = 0, \ldots, u_i$), let $\alpha_{k\ell} = \left\lceil \frac{c_k - \ell a_{ki}}{a_{kj}} \right\rceil - 1$. For any integer solution $\mathbf{x}$, $a_{ki}x_i + a_{kj}x_j \geq c_k$ if and only if for every ℓ ($\ell = 0, \ldots, u_i - 1$), either $x_i > \ell$ or $x_j > \alpha_{k\ell}$, or, equivalently, either $x_i \geq \ell + 1$ or $x_j \geq \alpha_{k\ell} + 1$, which can be written as $x_{i,\ell+1} + x_{j,\alpha_{k\ell}+1} \geq 1$.

Obviously, if $\alpha_{k\ell} \geq u_j$, then we fix the variable, $x_{i,\ell+1} = 0$.

If the above transformation is applied to a monotone system of inequalities, then the resulting 2-SAT integer program is also monotone. More precisely, for a constraint of the form $a_{ki}x_i - a_{kj}x_j \geq c_k$ the set of binarized constraints are all of the form $x_{ip} \geq x_{jq}$, or the reverse inequality, for some values of p and q. To see that, note that if $x_j \geq \ell$ then $x_i \geq \left\lceil \frac{c_k + \ell a_{kj}}{a_{ki}} \right\rceil = \beta_{k\ell}$. For this condition to be satisfied $x_{j\ell} \leq x_{i,\beta_{k\ell}}$, which is a monotone, closure, constraint.

Altogether we have replaced one original constraint on x_i and x_j by at most $u_i + 1$ constraints on the variables $x_{i\ell}$ and $x_{j\ell}$. The other cases, corresponding to different sign combinations of a_{ki}, a_{kj}, and c_k, can be handled in a similar way. This completes the proof of Theorem 2. □

With Theorem 2, it is sufficient to show the respective result for SM-MIN-2SAT. It is shown next that the rounded (up or down) solution is $B \cup V^1$, for $B \subseteq V^{\frac{1}{2}}$. In contrast to round-up SM2-multi, here B may be a strict subset of $V^{\frac{1}{2}}$.

Lemma 2. *For a submodular monotone function $f()$, any feasible rounding (up or down) of the variables in $V^{\frac{1}{2}}$ yields a 2-approximate solution for (SM2).*

Proof. Since $V^1 \cup V^{\frac{1}{2}} = \tilde{X}^+ \cup \tilde{X}^-$ then for any $B \subseteq V^{\frac{1}{2}}$ it follows from the monotonicity of function f that, $f(V^1 \cup B) \leq f(\tilde{X}^+ \cup \tilde{X}^-)$. From Theorem 1 we conclude that $f(V^1 \cup B) \leq 2f(S^*)$ thus demonstrating a polynomial time 2-approximation algorithm for any SM2 optimization of a monotone submodular function. □

It remains to show that if there exists a feasible solution to SM2-multi then it is possible to find a rounding of the variables in $V^{\frac{1}{2}}$ that yields a feasible solution to SM2-multi. Furthermore, such a feasible solution, if exists, can be found in polynomial time, since a feasible solution is derived by identifying a truth assignment to a 2SAT problem. If there is no truth assignment to the 2SAT problem, then the respective SM2-multi problem is infeasible. Assume that the set of SM2 constraints has a feasible integer solution denoted by $z_1, \ldots, z_n$.

Let the optimal solution to the monotonized constraints problem be m_i^+ and m_i^-. The first quantity, m_i^+, is the number of variables $x_i^{(p)+}$ that belong to $\tilde{X}^+$. Recall that because of the constraints $x_i^{(p)+} \geq x_i^{(p+1)+}$, there will be a consecutive sequence of variables $x_i^{(p)+}$ that are equal to 1, for $p = 0, 1, \ldots, q_i$ followed by a sequence of 0s. Hence, $m_i^+ = q_i$. Similarly, $m_i^- = q_i'$ is the largest index p such that $x_i'^{(p)-} = 1$.

For $i = 1, \ldots, n$, let $m_i^* = \frac{1}{2}(m_i^+ - m_i^-)$. We define the following solution vector, denoted by $\boldsymbol{\ell} = (\ell_1, \ldots, \ell_n)$, where for $i = 1, \ldots, n$:

$$\ell_i = \begin{cases} \min\{m_i^+, -m_i^-\} & \text{if } z_i \le \min\{m_i^+, -m_i^-\}, \\ z_i & \text{if } \min\{m_i^+, -m_i^-\} \le z_i \le \max\{m_i^+, -m_i^-\}, \\ \max\{m_i^+, -m_i^-\} & \text{if } z_i \ge \max\{m_i^+, -m_i^-\}. \end{cases}$$

We now prove that the vector $\boldsymbol{\ell}$ is feasible:

Lemma 3. *The vector $\boldsymbol{\ell}$ is a feasible solution to the given SM2-multi problem.*

Proof. Let $ax_i + bx_j \ge c$ be an inequality where a and b are nonnegative. We check all possible cases. If ℓ_i is equal to z_i or $\min\{m_i^+, -m_i^-\}$, and ℓ_j is equal to z_j or $\min\{m_j^+, -m_j^-\}$, then clearly, $a\ell_i + b\ell_j \ge az_i + bz_j \ge c$. Suppose $\ell_i \ge z_i$ and $\ell_j = \max\{m_j^+, -m_j^-\}$. By construction, we know that $am_i^+ - bm_j^- \ge c$ and $-am_i^- + bm_j^+ \ge c$.
If $\ell_i \ge -m_i^-$, then, $a\ell_i + b\ell_j \ge -am_i^- + bm_j^+ \ge c$. Otherwise, $a\ell_i + b\ell_j \ge am_i^+ - bm_j^- \ge c$.
The last case is when $\ell_i = \max\{m_i^+, -m_i^-\}$, and $\ell_j = \max\{m_j^+, -m_j^-\}$. In this case,
$a\ell_i + b\ell_j \ge am_i^+ - bm_j^- \ge c$. The other types of inequalities are handled similarly. □

The feasibility of the vector $\boldsymbol{\ell}$ for the set of constraints implies the following.

Corollary 1. *There exists a "rounding" to a set W satisfying*

$$\tilde{X}^+ \cap \tilde{X}^- \subseteq W \subseteq \tilde{X}^+ \cup \tilde{X}.$$

Corollary 2. *If $m_i^+ = m_i^-$ then $w_i = m_i^+ = m_i^-$.*

5. Submodular minimization over totally unimodular constraints

5.1. The SM-vertex cover on bipartite graphs is NP-hard

Iwata and Nagano prove that Switching Submodular Function Minimization (SSFM) is NP-hard, [14], even for f strictly monotone. Let V and V' both consist of n elements, with element $i \in V$ corresponding to element $i' \in V'$, and a subset $X \subseteq V$ corresponding to $X' \subseteq V'$. Let a submodular function f be defined on subsets of $V \cup V'$. The problem SSFM is to find a bi-partition of V, $(X \cup Y)$ that minimizes $f(X \cup Y')$. The proof that the SM vertex cover on bipartite graph is NP-hard is by reduction from SSFM[1] : Given an instance of SSFM with a strictly monotone submodular function f. Construct a bipartite graph on the sets of nodes V and V', with one edge between each $i \in V$ and $i' \in V'$. An optimal vertex cover in this graph includes exactly one of each pair of i and i'. The set selected in V is X and its complement – the set selected in V' – is Y'. This selection gives the optimal value of $f(X \cup Y')$. This reduction is obviously approximation preserving.

Since the constraint matrix of bipartite vertex cover is totally unimodular, we conclude as follows.

Theorem 3. *SM minimization over a totally unimodular constraints matrix is NP-hard.*

5.2. The bipartite-submodular vertex cover on bipartite graph

To generate the intuition for the relationship between bipartite-submodular vertex cover and SM-closure consider first the vertex cover and closure problems on a bipartite graph $G = (V_1 \cup V_2, E)$. We replace the set of edges E by a set of arcs directed from V_1 to V_2. Given a feasible closed set S (w.r.t. successors), then $(V_1 \setminus S) \cup (V_2 \cap S)$ is a feasible vertex cover. Vice versa, given a feasible vertex cover D, then $(V_1 \setminus D) \cup (V_2 \cap D)$ is a closed set (w.r.t. successors) in G.

For a bipartite graph $G = (V_1 \cup V_2, E)$ the bipartite-submodular vertex cover is to minimize $f(D) = f_1(D \cap V_1) + f_2(D \cap V_2)$ for D a vertex cover. Since D is a vertex cover, then $(V_1 \setminus D) \cup (V_2 \cap D)$ is a closed set in G. Since f_1 is submodular, then $f'(D) = f_1(V_1 \setminus D)$ is submodular as well, and so is $g(D) = f'(D) + f_2(D)$. Since the minimum submodular closure problem $\min_{D \subseteq V_1 \cup V_2} g(D)$ is solved in strongly polynomial time, then so is the bipartite-submodular vertex cover. The analogous argument proves that the bipartite-supermodular independent set problem is also solved in strongly polynomial time.

[1] This is to thank Asaf Levin for devising this reduction.

6. Conclusions

We demonstrate here a unified technique to generate best possible 2-approximation algorithms for a family of constrained submodular optimization with two variables per inequality that are NP-hard. The results hold also for submodular minimization over multi-sets. This settles, for the first time, the approximation and complexity status of a number of submodular minimization problems including SM-2SAT, SM-min satisfiability, SM-edge deletion for clique and SM-node deletion for biclique. In this sense these submodular minimization problems have similar complexity to the respective linear integer minimization. We also establish that the problem of submodular vertex cover on bipartite graphs or over a totally unimodular constraint matrix is NP-hard. This demonstrates that submodular minimization is harder than the respective linear integer minimization, where the minimization (or maximization) over a totally unimodular constraint matrix is polynomial time solvable.

Acknowledgments

The author is grateful to the anonymous referees for their constructive comments that greatly improved the presentation, and to Asaf Levin for his contribution on the proof of NP-hardness of the SM bipartite vertex cover as well as useful discussions. The author research is supported in part by NSF award No. CMMI-1200592.

References

[1] I. Dinur, S. Safra, Approximating vertex cover within a factor better than 1.36 is NP-hard, in: STOC '02: Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of Computing. 33–42 , 2002.
[2] S. Even, A. Itai, A. Shamir, On the complexity of timetable and multicommodity flow problems, SIAM J. Comput. 5 (1976) 691–703.
[3] G. Goel, C. Karande, P. Tripathi, L. Wang, Approximability of combinatorial problems with multi-agent submodular cost functions, in: Proc. 50th IEEE Symp. on Foundations of Computer Science, 2009, pp. 755–764.
[4] M. Grötschel, L. Lovász, A. Schrijver, Geometric Algorithms and Combinatorial Optimization, Springer-Verlag, Berlin, 1988.
[5] J. Hastad, Clique is hard to approximate within $n^{1-\epsilon}$, in: Proc. 37th IEEE Symp. on Foundations of Computer Science, 1996, pp. 627–636.
[6] D.S. Hochbaum, Efficient bounds for the stable set, vertex cover and set packing problems, Discrete Appl. Math. 6 (1983) 243–254.
[7] D.S. Hochbaum, Approximating covering and packing problems: Set cover, vertex cover, independent set and related problems, in: D.S. Hochbaum (Ed.), Approximation Algorithms for NP-Hard Problems, PWS Boston, 1997, pp. 94–143. Chapter 3.
[8] D.S. Hochbaum, Approximating clique and biclique problems, J. Algorithms 29 (1998) 174–200.
[9] D.S. Hochbaum, Solving integer programs over monotone inequalities in three variables: A framework for half integrality and good approximations, European J. Oper. Res. 140 (2) (2002) 291–321.
[10] D.S. Hochbaum, N. Megiddo, J. Naor, A. Tamir, Tight bounds and 2-approximation algorithms for integer programs with two variables per inequality, Math. Program. 62 (1993) 69–83.
[11] D.S. Hochbaum, J. Naor, Simple and fast algorithms for linear and integer programs with two variables per inequality, SIAM J. Comput 23 (6) (1994) 1179–1192.
[12] D.S. Hochbaum, M. Queyranne, Minimizing a convex cost closure set, SIAM J. Discrete Math. 16 (2) (2003) 192–207.
[13] S. Iwata, L. Fleischer, S. Fujishige, A combinatorial strongly polynomial algorithm for minimizing submodular functions, J. ACM 48 (4) (2001) 761–777.
[14] S. Iwata, K. Nagano, Submodular function minimization under covering constraints, in: Proc. 50th IEEE Symp. on Foundations of Computer Science, 2009, pp. 671–680.
[15] S. Iwata, J.B. Orlin, A Simple Combinatorial Algorithm for Submodular Function Minimization, in: SODA, 2009, pp. 1230–1237.
[16] R. Kohli, R. Krishnamurti, P. Mirchandani, The minimum satisfiability problem, SIAM J. Discrete Math. 7 (2) (1994) 275–283.
[17] C. Koufogiannakis, N. Young, Greedy delta-approximation algorithm for covering with arbitrary constraints and submodular cost, in: ICALP 2009, Part I, LNCS 5555, 2009, pp. 634-652.
[18] J.C. Lagarias, The computational complexity of simultaneous diophantine approximation problems, SIAM J. Comput. 14 (1985) 196–209.
[19] M.V. Marathe, S.S. Ravi, On approximation algorithms for the minimum satisfiability problem, Inform. Process. Lett. 58 (1) (1996) 23–29.
[20] J.B. Orlin, A faster strongly polynomial time algorithm for submodular function minimization, Math. Program. 118 (2) (2009) 237–251.
[21] A. Schrijver, A combinatorial algorithm minimizing submodular functions in strongly polynomial time, J. Combin. Theory Ser. B 80 (2000) 346–355.

Further reading

[1] D.S. Hochbaum, Approximation algorithms for the weighted set covering and node cover problems, SIAM J. Comput. 11 (3) (1982) 555–556. Also, GSIA Working Paper No. 64-79-80, April 1980.