Recognizing polynomial time solvable integer programs and a general purpose technique for approximation algorithms

Dorit S. Hochbaum
UC Berkeley

LNMB, Jan 2019

A general (2-) approximations framework

1. Any integer program (on bounded variables) where there are two variables per inequality constraint, **IP2**, are 2-approximable in polynomial time with a minimum cut procedure.
2. Some easily recognizable integer programs on constraints with three variables per constraint, **IP3**, are 2-approximable in polynomial time.
3. The 2-approximation factor cannot be improved unless NP=P.
4. Submodular minimization on IP2 constraints are 2-approximable in polynomial time.
5. The "monotone" versions of IP2 and IP3 are integer programs that are solved in polynomial time.

The IP2 problem

For V a set of variables and a set of pairs A, the IP2 formulation is defined on the multi-graph G=(V,A)

$$
\begin{array}{lll}
 & \min & \sum_{j \in V} w_j x_j \\
\text{(IP2)} & \text{s.t.} & a_{ij} x_i + b_{ij} x_j \geq c_{ij} \text{ for all } (i,j) \in A \\
 & & x_j \in [\ell_j, u_j] \quad \text{integer, for all } j \in V,
\end{array}
$$

The formulation is **monotone** if the signs of the two coefficients in each constraint are opposite.

The IP3 problem

For integer variables z_{ij} appearing at most once in each constraint, and with $u_{ij} \geq 0$, the IP3 formulation is,

$$
\begin{array}{lll}
\text{(IP3)} & \min & \sum_{j \in V} w_j x_j + \sum u_{ij} z_{ij} \\
\text{s.t.} & & a_{ij} x_i + b_{ij} x_j \geq c_{ij} + z_{ij} \text{ for all } (i,j) \in A \\
& & x_j \in [\ell_j, u_j] \quad \text{integer, for all } j \in V \\
& & z_{ij} \geq 0 \quad \text{integer, for all } (i,j) \in A.
\end{array}
$$

The formulation is monotone if the signs of the two coefficients in each constraint are opposite.

Complement of max-clique: IP2 and 2-approx example

The problem is to minimize the number (or weight) of deleted edges so the remaining graph is a clique (NP-hard):

$$
\begin{array}{lll}
\min & \sum_{[i,j]\in E} w_{ij} z_{ij} & \\
\text{subject to} & 1 - x_i \le z_{ij} & [i,j] \in E \\
& 1 - x_j \le z_{ij} & [i,j] \in E \\
& x_i + x_j \le 1 & [i,j] \notin E \\
& x_j \ \text{ binary} & j \in V \\
& z_{ij} \ \text{ binary} & [i,j] \in E.
\end{array}
$$

The Technique

Apply a transformation to the constraints resulting in an integer Programming formulation that is totally unimodular.

Integer Programming $\xrightarrow{\text{2 – factor transformation}}$ totally unimodular minimization

Furthermore, the transformed integer program is solved with a minimum cut procedure on a respective graph.

Applications of the Technique 2-approximations include:

- Vertex cover [H83]
- Min 2SAT [HMNT93]
- Feasible cut [Hoc02]
- Minimum satisfiability
- Edge biclique, bipartite, and general [Hoc98]
- Node biclique, general
- Node separator to two cliques
- Generalized vertex cover
- Scheduling with precedence [CH99]
- Generalized 2SAT [HP00]
- Complement of maximum clique

Applications of the Technique to monotone integer programs:
Polynomial time algorithms

- Convex cost dual of minimum cost network flow [AHO03, AHO04]
- Image segmentation [Hoc01,Hoc13]
- Generalized vertex cover and independent set on bipartite graphs
- Forest Harvesting (with edge effects)
- HNC clustering with normalized cut variant (previously thought to be NP-hard) for image segmentation and data mining [Hoc10,Hoc13a]
- Security applications: identifying alert regions with likely threats.
- Co-segmentation [HSingh09]
- Neuroscience: identifying neurons and signals [2018]

Motivation: Vertex Cover

A graph G=(V,E), node weights c_j

$$\text{Min} \sum_{1}^{n} c_j x_j$$

$$x_i + x_j \geq 1 \qquad \forall (i,j) \in E$$

$$x_j \in \{0,1\} \qquad \forall j \in V$$

2-Approximation algorithm [H80]:

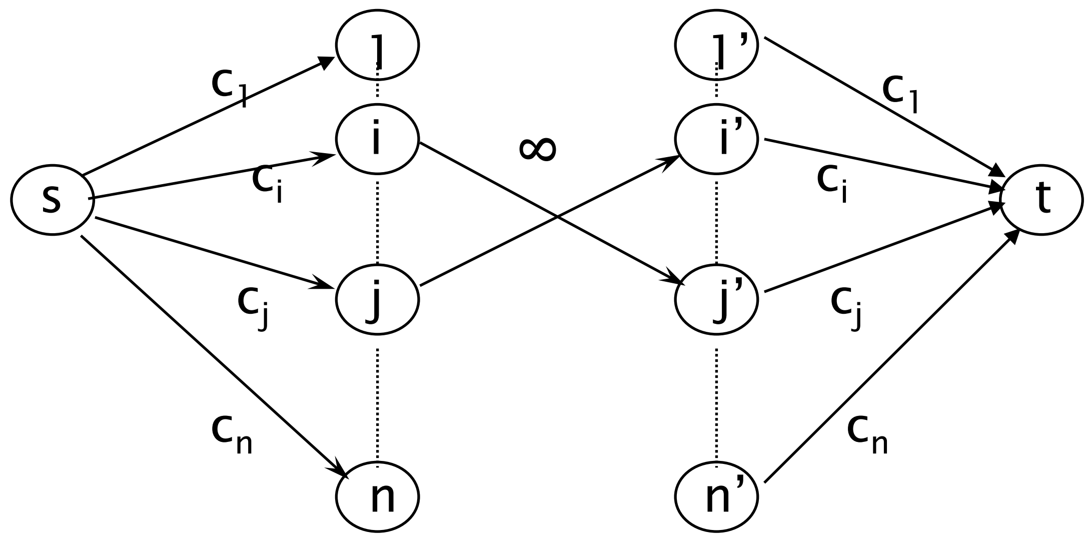
1. Solve the LP relaxation
2. Round up the half integral variables

Properties:

1. LP solution is half integral
2. LP can be solved instead by using minimum s,t cut
3. Persistency: Integer components retain their value in some optimal integer solution.

These properties hold for any IP2

Min s,t–cut for Solving VC Relaxation



The LP-relaxation of VC can be solved by finding an optimal cover C in a bipartite graph $G_b = (V_1 \cup V_2, E_b)$ having vertices $j \in V_1$ and $j' \in V_2$ of weight c_j for each vertex $j \in V$, and two edges $(i, j'), (j, i')$ for each edge $[i, j] \in E$. Given the optimal cover C on G_b, the optimal solution to the LP-relaxation of VC is given by:

$$x_j = \begin{cases} 1 & \text{if} \quad j \in C \text{ and } j' \in C, \\ \frac{1}{2} & \text{if} \quad j \in C \text{ and } j' \notin C, \text{ or } \quad j \notin C \text{ and } j' \in C, \\ 0 & \text{if} \quad j \notin C \text{ and } j' \notin C. \end{cases}$$

Recall

Vertex cover on bipartite graphs is solved via a **minimum s,t-cut** procedure on the respective s,t-network.

For a minimum cut $(S, \bar{S})$, the optimal vertex cover is:

$$(V_1 \cap \bar{S}) \cup (V_2 \cap S)$$

The constraints coefficients' matrix of vertex cover on bipartite graphs is **totally unimodular.**

Another Perspective on VC LP-relaxation

$$x_i \rightarrow \frac{x_i^+ - x_i^-}{2}$$

$$x_i + x_j \geq 1 \rightarrow \begin{cases} x_i^+ - x_j^- \geq 1 \\ -x_i^- + x_j^+ \geq 1 \end{cases}$$

Min $\sum_j c_j x_j^+ - \sum_j c_j x_j^-$

$$x_j^+ - x_i^- \geq 1$$

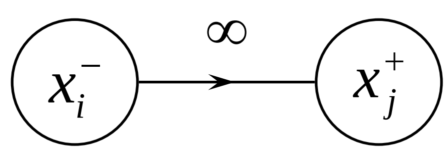
$$x_i^+ - x_j^- \geq 1$$

$$x_j^+ \in \{0,1\} \quad x_j^- \in \{-1,0\}$$

Representing the constraint as a cut problem on a graph:

$$x_i^+ \in \begin{cases} S & if \quad = 1 \\ \overline{S} & if \quad = 0 \end{cases}$$

$$x_i^- \in \begin{cases} S & if \quad = 0 \\ \overline{S} & if \quad = -1 \end{cases}$$



Generalizing this idea to "monotonize" IP2

"Monotonize" IP2 by replacing each variable x_i in the objective by $x_i = \frac{x_i^+ - x_i^-}{2}$ where $l_i \leq x_i^+ \leq u_i$ and $-u_i \leq x_i^- \leq -l_i$.

Each non-monotone inequality $a_{ij}x_i + b_{ij}x_j \leq c_{ij}$ is replaced by:

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

And each monotone inequality $a'_{ij}x_i - b'_{ij}x_j \leq c'_{ij}$ is replaced by:

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

Resulting monotone IP2'

With the monotonization we get the following problem:

$$\text{(IP2')} \min \quad \frac{1}{2}\left(\sum_{i\in V} w_i x_i^+ + \sum_{i\in V}(-w_i)x_i^-\right)$$

$$\begin{aligned}
\text{s.t.} \quad & a_{ij}x_i^+ - b_{ij}x_j^- \leq c_{ij} \quad \forall\ (i,j) \in E \\
& -a_{ij}x_i^- + b_{ij}x_j^+ \leq c_{ij} \quad \forall\ (i,j) \in E \\
& l_i \leq x_i^+ \leq u_i,\ integer \quad \forall\ i \in V \\
& -u_i \leq x_i^- \leq -l_i,\ integer \quad \forall\ i \in V
\end{aligned}$$

This problem is solved as minimum cut on a graph, as a closure problem, after applying "binarization":

“Binarization” of monotone inqequalities

For an integer variable x_i, $\ell_i \leq x_i \leq u_i$
replace it by binary variables
$x_i = \ell_i + \sum_{p=1}^{u_i - \ell_i} x_i^{(p)}$.
Given the monotone inequality $ax_i - bx_j \geq c$
Then, $x_j \geq p \Rightarrow x_i \geq \lceil \frac{bp+c}{a} \rceil = q(p)$.
Hence, $x_j^{(p)} = 1 \Rightarrow x_i^{q(p)} = 1$, or,

$$x_j^{(p)} \leq x_i^{q(p)}.$$

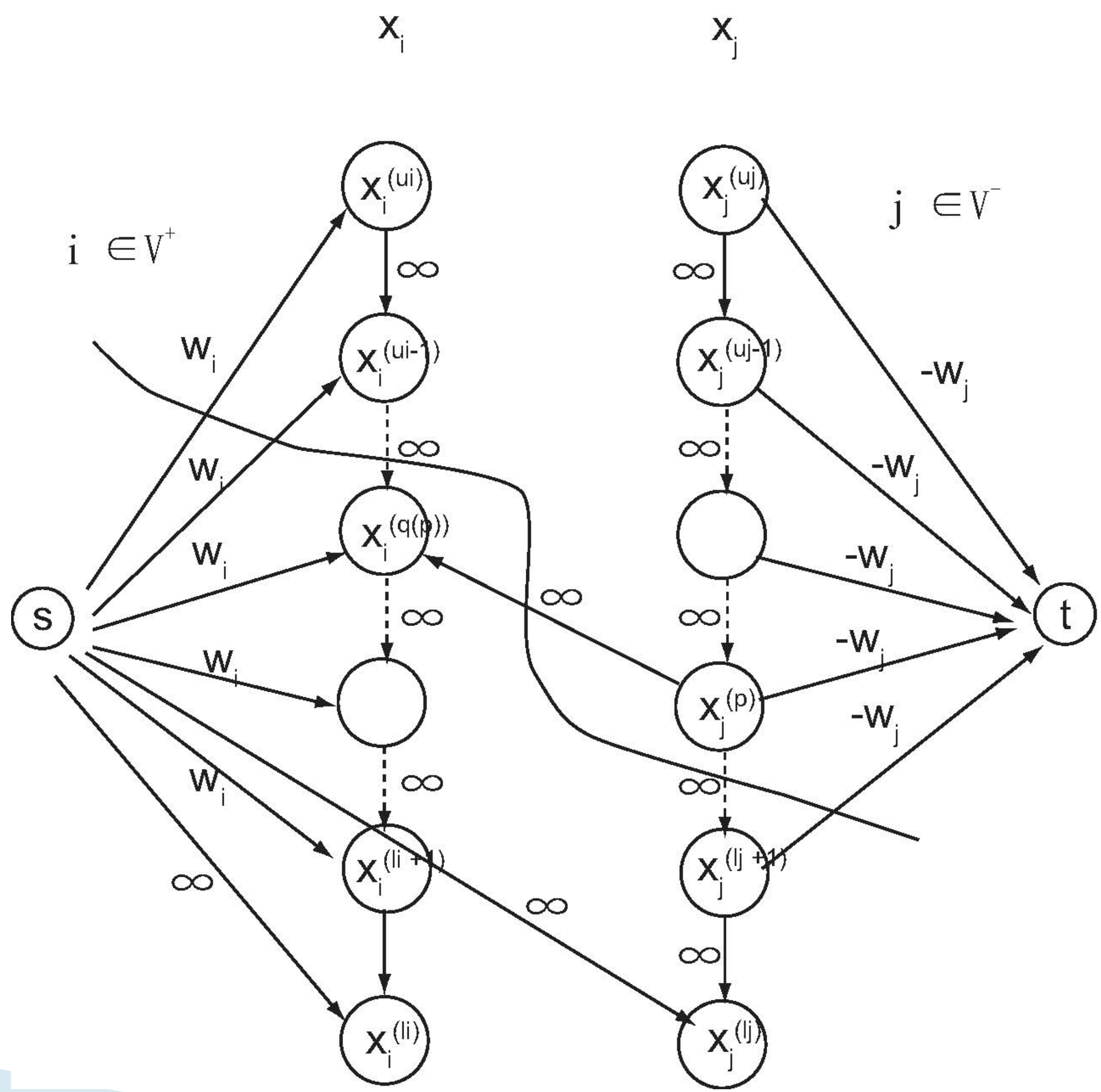
The problem then becomes a minimum closure problem

$$
\begin{array}{lll}
\text{(min closure)} & \min & \sum_{j \in V} w_j x_j \\
\text{subject to} & & x_i \le x_j \quad \text{for } (i,j) \in A \\
 & & x_j \in \{0,1\} \quad \text{for } j \in V.
\end{array}
$$

This problem is known (Picard 1979) to be solved as a minimum s,t-cut problem on an associated graph.

The cut graph for $ax_i - bx_j \le c$

Source set of a minimum cut solution is max/min closed set (that contains its successors) in the graph



Let the optimal solution to IP2' be $x_i^+ = m_i^+$ and $x_i^- = m_i^-$.

Attaining half integral super-optimal solution

$x_i = \frac{m_i^+ - m_i^-}{2}$ is half integral, feasible for the original inequalities and super optimal.

Rounding: There is always a feasible "rounding" to an integer 2-approximate solution for a feasible IP2. For IP3, the existence of such rounding is not guaranteed.

IP3 on Monotone Inequalities is solved in polynomial time as min cut

$$ax - by \le c + z$$

$$a \ge 0, \quad b \ge 0$$

After binarizarion the problem becomes the s-excess problem:

Binarized IP3 is the s-excess problem

$$
\begin{aligned}
\text{(s-excess)} \quad \min \quad & \sum_{j \in V} w_j x_j + \sum_{(i,j) \in A} u_{ij} z_{ij} \\
\text{subject to} \quad & x_i - x_j \le z_{ij} \quad \text{for } (i,j) \in A \\
& x_j \in \{0,1\} \quad \text{for } j \in V \\
& z_{ij} \in \{0,1\} \quad \text{for } (i,j) \in A.
\end{aligned}
$$

This problem is known (H01,H08) to be solved as a minimum s,t-cut problem on an associated graph.

If an IP3 has a feasible rounding, e.g. generalized vertex cover, then the solution is 2-approximate.

If an IP3 problem is monotone then it is poly time solvable. The following problem was thought to be NP-hard due to its similarity to normalized cut (Shi &Malik00):

$$\min_{\emptyset \subset S \subset V} \frac{C(S,\bar{S})}{C(S,S)}$$

Clustering with HNC: monotone IP3

For "seed" nodes s and t, find a cluster S:

$$\min_{S \subset V} \frac{C(S, \bar{S})}{C(S, S)}$$

The formulation is (for $x_s = 1$, $x_t = 0$):

$$
\begin{array}{lll}
\text{(HNC)} & \min & \frac{\sum w_{ij} z_{ij}}{\sum w'_{ij} y_{ij}} \\
\text{subject to} & & x_i - x_j \leq z_{ij} \quad \text{for all } [i,j] \in E \\
& & x_j - x_i \leq z_{ji} \quad \text{for all } [i,j] \in E \\
& & y_{ij} \leq x_i \quad \text{for all } [i,j] \in E \\
& & y_{ij} \leq x_j \\
& & x_j, z_{ij}, y_{ij} \text{ binary.}
\end{array}
$$

[H10,H13a]

Image segmentation, clustering, data mining, and pattern recognition with HNC has provided effective results and improvements over the spectral method and machine learning techniques

Additional interesting results

1. Any submodular minimization on constraints with up to two variables per inequality (SM2) has an immediate 2-approximation in poly time.
2. A SM2 on monotone constraints is solved in polynomial time. This holds also for multi-sets
3. The 2-approximation factor cannot be improved unless NP=P.

Selected References

- [H10] D. S. Hochbaum. Polynomial time algorithms for ratio regions and a variant of normalized cut. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32 (5), 889-898, May 2010.
- [HP00] D.S. Hochbaum and A. Pathria. Approximating a Generalization of MAX 2SAT and MIN 2SAT". *Discrete Applied Mathematics*, 107 (1-3) (2000) pp. 41-59.
- [HMNT93] D.S. Hochbaum, N. Megiddo, J.Naor and A.Tamir. Tight bounds and 2-approximation algorithms for integer programs with two variables per inequality. *Math. Programming*}, 62, 69-83,1993.
- [HN94] D.S. Hochbaum and J. Naor. Simple and fast algorithms for linear and integer programs with two variables per inequality. *SIAM Journal on Computing*, 23(6) 1179-1192, 1994.
- [H98] Dorit S. Hochbaum. "Approximating clique and biclique Problems". *Journal of Algorithms* 29, 1998, 174-200.
- [H02] Dorit S. Hochbaum. "Solving integer programs over monotone inequalities in three variables: A framework for half integrality and good approximations" *European Journal of Operational Research* 140/2, 2002, 291-321

The end

Questions/suggestions/comments?

hochbaum@ieor.berkeley.edu
http://www.ieor.berkeley.edu/~hochbaum/

Binarizing process

$$ax_i - bx_j \le c \rightarrow x_i^{(p_i)} - x_j^{(p_j)} \le 0$$

SM2 – submodular minimization over 2 variables per constraint

The problem of submodular minimization over constraints with up to two variables per inequality is, for a set of constraints associated with a collection of pairs A (multi-arcs/pairs allowed):

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

If the sign of the coefficient of one variable is opposite the sign of the second variable, then the constraint is called **monotone**.

SM2-multi – submodular multisets on 2 variables per constraint

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, $X_1 \subseteq X_2$ to mean that for all $(i, q_i) \in X_1$, $(i, q'_i) \in X_2$ with $q_i \leq q'_i$.

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

As for SM2, the constrains are monotone if the coefficients of the two variables are of opposite signs.

Summary of Results

- A poly time algorithm for any SM2 or SM2-multi on **monotone** constraints for u_j bounded by polynomial quantities. This run time dependence on u_j cannot be improved unless NP=P.
- A 2-approximation polynomial time algorithm for any SM2 or SM2-multi with either:
  1. Constraints with **round-up property** (easily recognizable) with respect to objective variables. If satisfied, the approximation algorithm holds for any submodular objective f().
  2. For constraints without **round-up property** the approximation algorithm holds for any **monotone submodular** objective f().

  Conditions 1 and 2 cannot be jointly relaxed unless NP=P!! (else max-clique is 2-approximable).
- The factor 2 approximation cannot be improved unless NP=P.
- A proof that submodular minimization over constraints with coefficients matrix that is totally unimodular is **NP-hard**, thus monotone constraints are easier than totally unimodular constraints for submodular minimization.

Examples of SM2 2-approximable and poly time solvable 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
SM-cut on closure graph	Yes	NA	any	1

FAQ: Is round-up property the same as covering constraints?

No. The complement of the max-clique problem is an illustration:

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

Here the Z variables are the only ones appearing in the objective. Therefore, round-up must apply only to Z-variables.

FAQ: Is the submodular objective necessarily non-negative?

No. The objective can assume negative values

FAQ: Can one use Lovasz extension instead?

The algorithm provided does not use a convex representation of the problem. The approximation Bound uses only basic properties of SM functions.

For technical reasons the direct use of the convex extension will NOT work, and can only increase the computational effort.

FAQ: For the respective linear problem HN94 provided a mapping to a totally unimodular constraints' matrix. Isn't it just the same for the submodular case?

Linear minimization in integers on totally unimodular constraints' matrix is poly time solvable.
Submodular minimization on totally unimodular constraints' matrix is proved here to be NP-hard. (Asaf Levin).

FAQ: Can one get a better algorithm that does not require either of the conditions to be satisfied?

Consider max independent set in minimization form (the negative of the objective) with submodular objective.

$$\begin{array}{lll} \min & f(X) & \\ \text{subject to} & x_i + x_j \leq 1 & [i,j] \in E \\ & x_j \text{ binary} & j \in V. \end{array}$$

Then:

1. Constraints do not have the round-up property.
2. The objective is NOT monotone.

Prior Results for SM-vertex cover

- Koufogiannakis-Young09 devised approximations for SM-``covering" problems with monotone submodular objective function. The approximation algorithm is based on the frequency technique (maximal dual feasible technique in [Hoc97]). Their algorithm is a 2-approximation for the SM-vertex cover for monotone submodular objective function.
- Goel et al.09 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-Nagano09 presented a 2-approximation algorithm for the SM-vertex cover without monotone restriction on f(). Their algorithm relies on Lovasz extension of submodular minimization to convex minimization.

A major subroutine: SM-cut and SM-closure

The SM-cut problem is defined on an s, t graph G_{st}. For $S \subseteq V$, the source set of the cut, the set of arcs from $\{s\} \cup S$ to $\bar{S} \cup \{t\}$, also denoted as, $(\{s\} \cup S, \bar{S} \cup \{t\})$, form the cut-set. The problem of finding such partition so that the submodular function value $f(S)$ of the associated cut-set is minimum, is solvable in polynomial time. To see that, observe that the source sets S form a ring: the intersection and union of source sets of cuts are source sets of cuts.

The SM-closure is written as:

$$
\begin{array}{llll}
\text{(SM-closure)} & \min & f(X) & \\
 & \text{subject to} & x_i - x_j \geq 0 & \forall (i,j) \in A, \\
 & & x_j \ \text{ binary} & j \in V.
\end{array}
$$

For a given cut-set in a closure graph, the source set S is a closed set. Hence the SM-cut on closure graph is equivalent to SM-closure, and both are solved in polynomial time.

The algorithm: Monotone SM2-multi is equivalent to SM-closure

Hochbaum-Naor94 algorithm finds a feasible and optimal solution for integer programming on monotone inequalities in pseudopoly time (in u_j), whereas finding a feasible solution to a system of monotone inequalities is NP-hard (Lagarias85), by mapping to an equivalent closure problem:

$x_i = \sum_{p=1}^{u_i} x_i^{(p)}$ where $x_i^{(p)} = 1$ only if $x_i \geq p$ and 0 otherwise, and $x_i^{(p)} = 1 \implies x_i^{(p-1)} = 1$.

For a monotone constraint (i,j), $a_{ij}x_i - b_{ij}x_j \geq c_{ij}$, let $q(p) \equiv \lceil \frac{c_{ij}+b_{ij}p}{a_{ij}} \rceil$.

The set of monotone constraints is equivalent to:

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

Monotone SM2-multi is in P

(for poly u_j)

Non-monotone SM2-multi: Monotonizing process

Duplicate V to V^+ and V^-;
the characteristic vector $\mathbf{x}$ to $\mathbf{x}^+$ and $\mathbf{x}^-$.
$x_j^+ \in \{0, 1, \ldots, u_j\}$ and $x_j^- \in \{-u_j, \ldots, -1, 0\}$.

A non-monotone inequality $a_{ij}x_i + b_{ij}x_j \geq c_{ij}$ is replaced by:

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

A monotone inequality $a'_{ij}x_i - b'_{ij}x_j \geq c'_{ij}$ is replaced by:

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

Setting $x_j = \frac{x_j^+ - x_j^-}{2}$ is feasible for the original inequalities.

The 2-approximation algorithm

SM2-multi is first monotonized, then the monotone relaxation is solved in polynomial time as an SM-closure problem.

Substituting, $x'_i = -x_i^-$ for all $i \in V^-$:
for any closed set S, $V^+ \cap S = \{j \in V^+ | x_j^+ = 1\}$ and $V^- \cap S = \{j \in V^- | x'_j = 0\}$.
Let $X^+ = \{j \in V^+ | x_j^+ = 1\}$ and $X^- = \{j \in V^- | x'_j = 1\}$
and $\mathbf{x}^+$ and $\mathbf{x}'$ are characteristic vectors of X^+ and X^-.
Define,

$$\begin{aligned} V^1 &= \{j | x_j = x'_j = 1\} \\ V^0 &= \{j | x_j = x'_j = 0\} \\ V^{\frac{1}{2}} &= \{j | x_j + x'_j = 1\}. \end{aligned}$$

Let $g(X^+ \cup X^-) = f(X^+) + f(X^-)$.
For $f()$ a submodular function $f^+(D) = f(V^+ \cap D)$ and $f^-(D) = f(V^- \setminus D)$ are submodular functions. Therefore, $f(X^+)$, $f(X^-)$ and $g(X^+ \cup X^-)$ are submodular functions.

2-approx theorem and corollary

Theorem 1: Let $X'^{+} \subseteq V^{+}$ and $X'^{-} \subseteq V^{-}$ 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, $2f(S^*) \geq f(X'^{+} \cup X'^{-})$.

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

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

The first inequality holds since $X'^{+} \cup X'^{-}$ is an optimal solution to the relaxed SM2-multi. The second inequality follows from the submodularity of the function f. $f(X'^{+} \cup X'^{-})$ is the value of our "round-up" solution where an element is included if is in $V^{1} \cup V^{\frac{1}{2}}$.

Corollary: For round-up SM2-multi and a **general** submodular objective function f() there is a polynomial time 2-approximation algorithm.

2-approximation for SM-problems without round-up property

We use the following result applying also to SM2-multi:

Theorem [HMNT93]: SM2-multi constraints are equivalent to the constraints of SM-MIN-2SAT on at most nU binary variables and O(mU) constraints, for U=$max_j u_j$, in that both have the same sets of feasible solutions.

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

A corollary for SM2-multi

SM2-multi problems on two variables per inequality constraints are 2-approximable in polynomial time using with either round-up constraints or monotone submodular function as determined by the equivalent SM-min-2SAT constraints.

Conclusions

1. SM2-multi problems on two variables per inequality constraints are 2-approximable in polynomial time using with either round-up constraints or monotone submodular function as determined by the equivalent SM-min-2SAT constraints.
2. The condition on either round-up or monotone SM cannot be removed unless NP=P.
3. The 2-approximation factor cannot be improved unless NP=P.
4. The dependence of the run time of SM2-multi on the upper bounds on the variables cannot be removed unless NP=P.

Thank you!