Egerváry Research Group
on Combinatorial Optimization

Technical reportS

TR-2007-09. Published by the Egerváry Research Group, Pázmány P. sétány 1/C, H–1117, Budapest, Hungary. Web site: www.cs.elte.hu/egres . ISSN 1587–4451.

Degree constrained submodular flows

Tamás Király and Lap Chi Lau

July 2007

Degree constrained submodular flows

Tamás Király⋆ and Lap Chi Lau⋆⋆

Abstract

We consider the problem of finding a minimum cost $0-1$ submodular flow with the additional constraint that the sum of the incoming and outgoing flow at each node cannot exceed a given limit. We show that this problem is NP-hard, but it can be approximated in the following sense: we can find a submodular flow of cost not greater than the optimum which violates the additional constraints by at most 1 at every node.

1 Introduction and main result

Given a directed graph $D = (V, E)$, we use the following notation for the set of edges entering or leaving a node set:

$$\delta^{in}(X) = \{uv \in E : \ u \notin X, v \in X\},$$
$$\delta^{out}(X) = \{uv \in E : \ u \in X, v \notin X\},$$
$$\delta(X) = \delta^{in}(X) \cup \delta^{out}(X).$$

If $F \subseteq E$ is an edge set and $x : E \to \mathbb{R}$ is a function on the edges, then we use the notation $x(F) = \sum_{e\in F} x(e)$.

Given a digraph $D = (V, E)$, a crossing submodular set function $b : 2^V \to \mathbb{Z}\cup\{+\infty\}$, a node set $T \subseteq V$, and a function $g : T \to \mathbb{Z}_+$, a *degree-constrained* $0-1$ *submodular flow* is a vector $x : E \to \{0, 1\}$ with the following properties:

$$x(\delta^{in}(X)) - x(\delta^{out}(X)) \le b(X) \qquad \text{for every } X \subseteq V, \qquad (1)$$
$$x(\delta(v)) \le g(v) \qquad \text{for every } v \in T. \qquad (2)$$

If $T = \emptyset$, this is the standard submodular flow, introduced by Edmonds and Giles [1]. There are several efficient algorithms for finding a feasible submodular flow, or even a minimum cost submodular flow for a linear cost function. However, the addition

⋆MTA-ELTE Egerváry Research Group, Dept. of Operations Research, Eötvös Loránd University. Research supported by OTKA K60802, ADONET MCRTN 504438, and OMFB-01608/2006. `tkiraly@cs.elte.hu`

⋆⋆Dept. of Computer Science and Engineering, The Chinese University of Hong Kong. `chi@cse.cuhk.edu.hk`

of the degree constraints (2) makes the feasibility problem NP-complete, as we show in Section 2.

One way to define approximately optimal solutions in such a situation is to allow a slight violation of the degree constraints. More precisely: let $c : E \to \mathbb{Z}$ be a cost function, and let α be the optimum value of the LP relaxation of the problem (i.e. the minimum cost of a feasible vector $x : E \to [0, 1]$). Our aim is to find a $0-1$ submodular flow of cost no more than α that violates each degree constraint by at most 1. This approach is similar to the one used for degree-constrained spanning trees by Goemans [2] and Singh and Lau [6]. The main result of this note is that such a solution can be found in polynomial time.

Theorem 1.1. *Given an instance of the degree-constrained submodular flow problem and a cost function $c : E \to \mathbb{Z}$, let α be the optimum value of the LP relaxation. Then it is possible to find in polynomial time a $0-1$ submodular flow of cost no more than α that violates each degree constraint by at most one.*

Proof. We use a variant of the technique that was applied successfully to the degree-constrained spanning tree problem in [6]. In fact, the proof is even simpler in our case. The technique is inspired by the iterative rounding method of Jain [4] and the idea of iterative relaxation in [5].

First, we remove as many degree constraints as possible. Removal of a degree constraint at a node v is possible in the following 3 cases:

- If $|\delta(v)| \leq g(v) + 1$ then we can remove the degree constraint at v, since a solution of the resulting problem cannot violate the original degree constraint by more than 1.

- If $g(v) = 0$ then we can delete the edges incident to v and remove the degree constraint.

- If $g(v) = 1$ then we replace v by two nodes v_1 and v_2. An edge $uv \in E$ is replaced by uv_1, while edge $vu \in E$ is replaced by v_2u. The set function b is modified as follows:

$$b'(X) = \begin{cases} 1 & \text{if } X = v_1 \text{ or } X = V - v_2, \\ b(X) & \text{if } X \cap \{v_1, v_2\} = 0, \\ b(X - \{v_1, v_2\} + v) & \text{if } \{v_1, v_2\} \subseteq X, \\ \infty & \text{otherwise.} \end{cases}$$

  The set function b' is crossing submodular. No degree constraint is given for v_1 and v_2, i.e. $T' = T - v$. The definition of b' implies that $x(\delta(v_1)) \leq 1$ and $x(\delta(v_2)) \leq 1$ for any solution x. This means that the corresponding solution on the original digraph violates the degree constraint at v by at most 1.

After the above modifications, we may assume that $g(v) \geq 2$ and $|\delta(v)| \geq g(v) + 2$ for every $v \in T$.

Let x^* be an optimal basic solution of the linear programming relaxation. This can be obtained in polynomial time by the ellipsoid method. The number of edges can be reduced in two cases:

- If $x^*(e) = 0$ for some $e \in E$, then we delete the edge e from the digraph. A solution of the resulting problem is good for the original problem.
- If $x^*(e) = 1$ for some $e = uv \in E$, then we delete the edge e from the digraph, decrease $g(u)$ and $g(v)$ by 1, and change b as follows:

$$b'(X) = \begin{cases} b(X) - 1 & \text{if } u \notin X \text{ and } v \in X, \\ b(X) + 1 & \text{if } u \in X \text{ and } v \notin X, \\ b(X) & \text{otherwise.} \end{cases}$$

  The set function b' is also crossing submodular. If we have a solution x' for this modified problem, then we can obtain a solution for the original problem by setting $x'(e) = 1$.

In light of these observations we may assume that $0 < x^*(e) < 1$ for every $e \in E$, and $|\delta(v)| \geq g(v) + 2 \geq 4$ for every $v \in T$. In addition, we may assume that there are no isolated nodes (otherwise we could delete them and change the set function b accordingly). Since x^* is a basic solution, there is a system of linearly independent constraints which are tight at x^* for which x^* is the unique solution of the equation system given by these tight constraints. Let $\mathcal{F}^*$ be the family of sets corresponding to the submodular flow constraints in this system, and let T^* denote the set of nodes with degree constraints that are in the system.

Claim 1.2. *We may assume that the family $\mathcal{F}^*$ is cross-free.*

Proof. Let $\mathcal{F}^*$ be a possible family where the number of sets not crossing any other set is maximal. Suppose that $X \in \mathcal{F}^*$ and $Y \in \mathcal{F}^*$ are crossing. By the submodularity of b, the constraints given by $X \cap Y$ and $X \cup Y$ are also tight. Moreover, the constraint given by Y is a linear combination of the constraints given by $X, X \cap Y, X \cup Y$ (here we use the structure of flow constraints). This means that Y can be replaced in $\mathcal{F}^*$ by one of $X \cap Y$ and $X \cup Y$. Notice that if a set in $\mathcal{F}^*$ did not cross any other before the replacement, then it does not cross any set after it. If we repeat this for every set in $\mathcal{F}^*$ that crosses X, we obtain a system where the number of sets not crossing any other set is more, contradicting our initial assumption. □

Since x^* is the unique solution of the equation system defined by $\mathcal{F}^*$ and T^*, we have $|E| \geq |\mathcal{F}^*| + |T^*|$. We show, using a simple counting argument, that this is impossible.

We assign $2|E|$ tokens to the nodes by assigning 2 tokens for every edge in E to the two endnodes of the edge. The idea of the proof is to reassign these tokens to the members of $\mathcal{F}^*$ and T^* so that every member gets at least 2 tokens and at least one token is not assigned to any member. This would contradict $|E| \geq |\mathcal{F}^*| + |T^*|$.

Let $r \in V$ be an arbitrary node. We define the family

$$\mathcal{H}^* := \{X \subseteq V - r : X \in \mathcal{F}^*\} \cup \{X \subseteq V - r : V - X \in \mathcal{F}^*\}.$$

Notice that $\mathcal{H}^*$ is laminar. For a set $X \in \mathcal{H}^*$, we define $X' \in \mathcal{F}^*$ to be either X or $V - X$ (depending on which one is in $\mathcal{F}^*$). We will assign 2 tokens to each member of $\mathcal{H}^*$ so that every member gets tokens from its nodes, thus the tokens of r are not used.

A node $v \in T^*$ has at least 4 tokens since $|\delta(v)| \geq g(v) + 2 \geq 4$. We assign 2 of its tokens to v (as degree constraint) and 2 tokens to the smallest member of $\mathcal{H}^*$ containing v. If no member of $\mathcal{H}^*$ contains v, we have 2 unused tokens.

To assign tokens to the remaining members of $\mathcal{H}^*$, we proceed in an order compatible with the partial order of inclusion. Let $X \in \mathcal{H}^*$ be a set that has no tokens yet, and let $\{X_1, \dots, X_k\}$ be the maximal members of $\mathcal{H}^*$ inside X, which all have at least two tokens assigned to them. There must be an edge with an endnode in $X - \cup_{i=1}^k X_i$, otherwise the constraints corresponding to $X', X'_1, \dots, X'_k$ would be linearly dependent: the constraint for X' would be a ± 1 combination of the constraints for $X'_1, \dots, X'_k$, where the i-th coefficient depends on whether $X'_i = X_i$ or $X'_i = V - X_i$. Moreover, if only one such edge e existed, then $x^*(e)$ would be integer because it would be determined by an integer combination $b(X'), b(X'_1), \dots, b(X'_k)$. Since $0 < x^*(e) < 1$ for every edge, it follows that there are at least two edges with an endnode in $X - \cup_{i=1}^k X_i$, hence there are at least two tokens inside X that are not yet assigned to other sets. We assign these tokens to X.

At the end of this procedure, every member of $\mathcal{H}^*$ and T^* is assigned 2 tokens, and there is an unused token at r since it is not an isolated node. This contradicts the assumption that $|E| \geq |\mathcal{F}^*| + |T^*|$, so we proved that there must be at least one edge e with $x^*(e) = 1$. This concludes the proof. □

An application of Theorem 1.1 concerns the reorientation of digraphs to achieve high edge-connectivity. Given a digraph $D = (V, E)$ and a cost function $c : E \to \mathbb{Z}$, it is possible to find in polynomial time an edge set of minimum cost whose reversal makes the digraph k-edge-connected (if it exists). This can be done by considering the submodular flow problem defined by the set function $b(X) = |\delta^{in}(X)| - k$ $(\emptyset \neq X \subsetneq V)$ (see [3]). However, if we add the additional constraint for each node v that the number of reversed edges incident to v cannot exceed a given number $g(v)$, then the problem becomes NP-hard, as it is shown in Section 2.

Theorem 1.1 implies that it is possible to find in polynomial time a solution of cost no more than the optimum that violates each degree constraint by at most one.

We may also mention as a special case the degree-constrained directed cut cover problem: given a digraph $D = (V, E)$, a cost function $c : E \to \mathbb{Z}$, and a degree constraint $g : V \to \mathbb{Z}_+$, find an edge set $E' \subseteq E$ of minimum cost such that $|E' \cap \delta(X)| \geq k$ for every directed cut X and $|E' \cap \delta(v)| \leq g(v)$ for every $v \in V$. This is clearly a degree constrained submodular flow problem, so Theorem 1.1 can be applied to get a solution that violates each degree constraint by at most 1.

Finally, let us note that the proof presented here gives an alternative proof of the integrality of the submodular flow polyhedron.

2 Hardness of the feasibility problem

In this section we prove that a special case of the degree-constrained $0-1$ submodular flow problem is NP-complete. A subset of edges in a digraph is called *independent* if no two edges have a common node.

Theorem 2.1. *Given a digraph $D = (V, E)$ and a subset $F \subseteq E$ of edges, it is NP-complete to decide if it is possible to change the orientation of an independent subset of edges in F so that the resulting graph is strongly connected.*

Proof. We reduce SAT to this problem. Let us consider a SAT instance with variables $x_1, \ldots, x_n$ and clauses $c_1, \ldots, c_m$. We associate a digraph $D = (V, E)$ and an edge set $F \subseteq E$ to this instance using the following construction.

For the variable x_j, let m_j be the number of clauses that contain x_j or $\neg x_j$. We construct a cycle of length $4m_j$: the nodes are $u_i^j, v_i^j, w_i^j, z_i^j$ $(i = 1, \ldots, m_j)$, the oriented edges are $u_i^j v_i^j, w_i^j v_i^j, z_i^j w_i^j, z_i^j u_{i+1}^j$ $(i = 1, \ldots, m_j)$. The edge set F consists of all these edges.

In addition, we add a node t and nodes s_i $(i = 1, \ldots, m)$, and add edges $s_i t$ $(i = 1, \ldots, m)$. For a given variable x_j, suppose that c_i is the l-th clause that contains x_j or $\neg x_j$. If it contains x_j, then we add the edges $s_i u_l^j, u_l^j s_i, w_l^j t, t w_l^j$. If it contains $\neg x_j$, then we add the edges $s_i w_l^j, w_l^j s_i, u_l^j t, t u_l^j$. This finishes the construction of the digraph D.

Consider the cycle of length $4m_j$ associated to the variable x_j. The nodes v_i^j have out-degree 0, while the nodes z_i^j have in-degree 0 $(i = 1, \ldots, m_j)$. This means that we have to change the orientation of $2m_j$ independent edges in the cycle in order to get a strong orientation. Thus we have two possibilities: either we change the orientation of the edges $u_i^j v_i^j, z_i^j w_i^j$ $(i = 1, \ldots, m_j)$, or of the edges $w_i^j v_i^j, z_i^j u_{i+1}^j$ $(i = 1, \ldots, m_j)$. We say that the former corresponds to the 'true' value of x_j, while the later corresponds to the 'false' value.

In this way, there is a one-to-one correspondence between orientations of the above structure and possible evaluations of the variables. We claim that the orientation is strongly connected if and only if the corresponding evaluation satisfies the SAT formula. Suppose that the formula is not satisfied, i.e. there is a clause c_i containing only false literals. Consider the node set consisting of s_i and its neighbors of type u and w. By the construction, this set has in-degree 0 in the orientation corresponding to the evaluation. Therefore the orientation cannot be strongly connected.

Now suppose that an evaluation satisfies the formula. Then each node s_i $(i = 1, \ldots, m)$ can be reached from t by a path of length 4 (which correspond to the "true" literal in c_i). Since there is an edge from s_i to t for each s_i, and all other nodes obviously have paths to and from t or some s_i, the orientation is strongly connected. □

Corollary 2.2. *The feasibility problem for degree-constrained $0-1$ submodular flows is NP-complete.*

References

[1] J. Edmonds, R. Giles, *A min-max relation for submodular functions on graphs*, Ann. Discrete Math. 1 (1977), 185–204.

[2] M.X. Goemans, *Minimum bounded-degree spanning trees*, Proceedings of 47th IEEE FOCS (2006), 273–282.

[3] A. Frank, *An algorithm for submodular functions on graphs*, Ann. Discrete Math. 16 (1982), 97–120.

[4] K. Jain, *A factor 2 approxiamtion algorithm for the generalized Steiner network problem*, Combinatorica 21 (2001), 39–60.

[5] L.C. Lau, S. Naor, M. Salavatipour,M. Singh, *Survivable network design with degree or order constraints*, Proceedings of 39th ACM STOC (2007).

[6] M. Singh, L.C. Lau, *Approximating minimum bounded degree spanning trees to within one of optimal*, Proceedings of 39th ACM STOC (2007).