Proof: First observe that for each vertex $y \in Y$ and for each path P from r to y in D, the subpath P' of P from w to y is disjoint from R. Thus it suffices to find an arc yx from Y to R for which there is a path from x to w. Let T denote the set of the heads of the arcs that connect Y to R, and let N denote the set of in-neighbours of w in D. As we have observed above, T is non-empty and $N \subseteq R$. Let T^+ denote the set of vertices reachable from T in $D - Y$. If T^+ intersects N, we are done. If $T^+ \cap N = \emptyset$, then consider the w-based subpartition $(\mathcal{P}, \mathcal{Q})$, in which $\mathcal{P}$ consists of the singletons of T^+ and $\mathcal{Q} = \{Y\}$. Then $\text{dem}(\mathcal{P}, \mathcal{Q}) = 2|T^+| + 1$ and $s_G(\mathcal{P}, \mathcal{Q}) = \sum_{v \in T^+} \rho_D(v) = 2|T^+|$, which contradicts (5). $\square$

Let C be a proper cycle and suppose that there is a vertex $v \in R$ which is not reachable from r in $D' - w$. Then there is a set $X \subseteq V - r - w$ with $\rho_{D'}(X) = 0$ and $R \cap X \neq \emptyset$. The above discussion and the fact that D' is obtained from D by a cycle reversal implies:

Claim 25 *Let $X \subseteq V - r - w$ with $\rho_{D'}(X) = 0$ and $R \cap X \neq \emptyset$. Then in D*
(i) there is exactly one arc which leaves X, this arc belongs to C, and enters w,
(ii) there is exactly one arc which enters X, this arc belongs to C, and comes from $R - X$,
(iii) $X \cap Y = \emptyset$,
(iv) there is no arc from Y to X.

We say that a set $X \subseteq R - r$ is *dangerous*, if it satisfies the following properties: there is a unique incoming arc into X from $R - X$, there is no arc from Y to X, and there is an arc from X to w. Notice that for a dangerous set X the arc from X to w is unique: if there exist at least two arcs from X to w, then $\rho_D(w) = 2$ gives $\rho_D(X \cup Y \cup \{w\}) = 1$, contradicting the rooted 2-edge connectivity of D. The above discussion implies:

Claim 26 *Let R' be the set of vertices reachable from r in $D' - w$. Then $R - R' \neq \emptyset$ if and only if there is a dangerous set X such that C contains the arc from X to w.*

Thus if there is no dangerous set in D, then we are done. In the proof of the next claim we use the well-known submodular inequality of the in-degree function.

Claim 27 *Let $X, Z \subseteq R - r$ be dangerous sets with $X \cap Z \neq \emptyset$. Then $X \cup Z$ is also dangerous.*

Proof: By the definition of dangerous sets and R, we have

$$1 + 1 = \rho_{D-w}(X) + \rho_{D-w}(Z) \geq \rho_{D-w}(X \cap Z) + \rho_{D-w}(X \cup Z) \geq 1 + 1,$$

which implies that $\rho_{D-w}(X \cup Z) = 1$ must hold. It is clear that there is no arc from Y to $X \cup Z$ and there is an arc from $X \cup Z$ to w. $\square$

It follows from Claim 27 that the maximal (with respect to inclusion) dangerous sets $X_1, X_2, \ldots, X_z$ are pairwise disjoint. We may assume that $z \geq 1$.

First suppose that $z \geq 2$. Let $\overline{R} = R - r - \cup_1^z X_i$. Consider the w-based subpartition $(\mathcal{P}, \mathcal{Q})$ in which $\mathcal{P}$ consists of the singletons of $\overline{R}$, and $\mathcal{Q}$ consists of the sets X_i, $1 \leq i \leq z$, and Y. Then

$$\text{dem}(\mathcal{P}, \mathcal{Q}) = 2|\overline{R}| + z + 1.$$

Furthermore, since $\rho_D(w) = 2$ and each X_i sends an arc to w, we have $z = 2$ and each arc that leaves $\overline{R}$ enters some set X_i. The arcs leaving the X_i's or Y enter $\overline{R}$ or w. Therefore

$$s_G(\mathcal{P}, \mathcal{Q}) = \sum_{v \in \overline{R}} \rho_D(v) + \sum_1^z \rho_{D-w}(X_j) = 2|\overline{R}| + z,$$

which contradicts (5).

Next suppose that $z = 1$. Let X be the unique maximal dangerous set and let aw be the unique arc from X to w. Thus for the two arcs aw, bw that enter w we must have $b \in V - Y - X - w$. We shall prove that there is a proper cycle C which contains bw. By the maximality of X and Claim 26, by reversing cycle C we obtain a digraph D' with $R \subseteq R'$, as desired.

Let T be the set of the heads of the arcs that go from Y to R. We have seen that T is non-empty. It suffices to prove that there is a path P from some vertex $t \in T$ to b in $D - w$. To see this consider an arc yt with $y \in Y$ and observe that P, together with a path P' from w to y with $V(P') \subseteq Y + w$, and the arcs yt, bw, form a proper cycle.

So let us suppose that there is no path from T to b. Let T^+ be the set of vertices reachable from T in $D-w$. Let cx be the unique arc that enters X. First suppose that $a \notin T^+$. Let $\mathcal{P}$ be the subpartition consisting of the singletons of T^+ and let $\mathcal{Q} = \{Y\}$. Then the w-based subpartition $(\mathcal{P}, \mathcal{Q})$ satisfies

$$\mathrm{dem}(\mathcal{P}, \mathcal{Q}) = 2|T^+| + 1.$$

Furthermore, since each edge of G incident with a vertex v of T^+ contributes to the in-degree of v in D (as $a, b \notin T^+$ and $\rho_D(w) = 2$), and every arc from Y to $V - Y - \{w\}$ enters T^+, we have

$$s(\mathcal{P}, \mathcal{Q}) = \sum_{v \in T^+} \rho_D(v) = 2|T^+|,$$

which contradicts (5).

Next suppose that $a \in T^+$. Then T^+ intersects X and contains c. Observe that $X - T^+ = \emptyset$, for otherwise $\rho_{D-w}(X - T^+) = 0$, a contradiction (since $X - T^+ \subseteq R$). Hence $X \subseteq T^+$. Let $\overline{T} = T^+ - X$. Note that $\overline{T} \neq \emptyset$ (as $T \subseteq T^+$ and $T \cap X = \emptyset$).

Let $\mathcal{P}$ be the subpartition consisting of the singletons of $\overline{T}$ and let $\mathcal{Q} = \{X, Y\}$. Then the w-based subpartition $(\mathcal{P}, \mathcal{Q})$ satisfies

$$\mathrm{dem}(\mathcal{P}, \mathcal{Q}) = 2|\overline{T}| + 2.$$

Furthermore, since each edge of G incident with a vertex v of $\overline{T}$, except for cx, contributes to the in-degree of a vertex in $\overline{T}$ in D, every arc from X to $R - X$ leads to a vertex in $\overline{T}$, and every arc from Y to $V - Y - \{w\}$ enters $\overline{T}$, we have

$$s_G(\mathcal{P}, \mathcal{Q}) = \sum_{v \in \overline{T}} \rho_D(v) + 1 = 2|\overline{T}| + 1,$$

which contradicts (5). This completes the proof. □

We remark that the above proof is algorithmic. It gives rise to a polynomial time algorithm which can either find the required orientation or a w-based subpartition that violates (5) .

References

[1] O. Durand de Gevigney, Graphs Orientations: structures and algorithms, PhD thesis, Grenoble, 2006.

[2] O. Durand de Gevigney, On Frank's conjecture on k-connected orientations, *J. Comb. Theory*, Ser. B, 141: 105-114, 2020.

[3] A. Frank, On disjoint trees and arborescences, in: Algebraic methods in graph theory, Coll. Math. Soc. J. Bolyai, 25 (1978) 159-169.

[4] A. Frank, Connections in Combinatorial Optimization, Oxford University Press, 2011.

[5] D. Garamvölgyi, T. Jordán, Cs. Király, S. Villányi, Highly connected orientations from edge-disjoint rigid subgraphs, *Forum of Mathematics, Pi*, Vol. 13, 2025, e11.

[6] S.L. Hakimi, On the degrees of the vertices of a directed graph, *Journal of the Franklin Institute*, vol 279, no 4, pages 290–308, 1965.

[7] F. Hörsch, Z. Szigeti, The complexity of 2-vertex-connected orientations in mixed graphs, *Discrete Optimization*, vol. 48, 2023.

[8] T. Jordán, On the existence of k edge-disjoint 2-connected spanning subgraphs, *J. Comb. Theory*, Ser. B., Vol. 95, 257-262, 2005.

[9] T. Jordán, Combinatorial rigidity: graphs and matroids in the theory of rigid frameworks. In: Discrete Geometric Analysis, *MSJ Memoirs*, vol. 34, pp. 33-112, 2016.

[10] L. Lovász and Y. Yemini, On generic rigidity in the plane, *SIAM Journal on Algebraic Discrete Methods*, Vol. 3, Iss. 1 (1982).

[11] C. St. J. A. Nash-Williams, On Orientations, Connectivity and Odd-Vertex-Pairings in Finite Graphs, *Canadian J. Math.*, Vol. 12 , 1960 , pp. 555 - 567.

[12] C. Thomassen, Strongly 2-connected orientations of graphs. *J. Comb. Theory*, Ser.B, 110: 67–78, 2015.

l_1-Balanced Submodular Flows

Alpár Jüttner*

Department of Operations Research
Eötvös Loránd University,
Budapest, Hungary
alpar.juttner@ttk.elte.hu

Eszter Szabó*

Department of Operations Research
Eötvös Loránd University,
Budapest, Hungary
szaboeszti@student.elte.hu

Abstract: Balanced optimization problems aim to find the most equitable distribution of resources. One of these problems is the Balanced Submodular Flow Problem. This paper introduces the l-Balanced Submodular Flow problem with respect to a norm l. In particular, it is shown that the l_1-Balanced Submodular Flow problem is solvable in strongly polynomial time.

Keywords: submodular flow, balanced optimization

1 Introduction

In balanced optimization problems, the aim is to find the most equitable distribution of resources. Several problems have been analysed in the literature such as the balanced spanning tree problem [1], the balanced assignment problem [3]. The Balanced Submodular Flow Problem seeks a feasible submodular flow minimizing the difference between the largest and smallest flow values along the arcs. In [2], it is shown that this problem can be solved with $\mathcal{O}(m^2)$ submodular function minimizations and a strongly polynomial algorithm is presented for the weighted version of the problem.

In this paper, we introduce the l-Balanced Submodular Flow problem with respect to a norm l. This problem aims to calculate a real value α, such that there is a feasible submodular flow minimizing the difference between the flow values and the calculated α value w.r.t. the norm l. Note, that the l_∞-Balanced Submodular Flow problem is equivalent to the Balanced Submodular Flow problem mentioned above.

As a new algoritmic result, a strogly polynomial algorithm is given to the l_1-Balanced Submodular Flow problem — that is for the problem of finding a real number of α along with a submodular flow that minimizes the sum of the difference between the flow values and the value of α.

2 Preliminaries

Definition 2.1. *For an underlying set V, let $\mathcal{P}(V)$ denote the power set of V, i. e. the set of all subsets of V. The set functions $b, p : \mathcal{P}(V) \to \mathbb{R}$ are called* submodular *or* supermodular *if*

$$b(X) + b(Y) \geq b(X \cup Y) + b(X \cap Y) \tag{1}$$

or

$$p(X) + p(Y) \leq p(X \cup Y) + p(X \cap Y) \tag{2}$$

holds for each subsets $X, Y \subseteq V$, respectively. A function is called modular *if it is both sub- and supermodular.*

Theorem 2.2 ([4])**.** *Assuming that the value of a submodular function b can be computed for any $X \subseteq V$ in time T, then the value of $\min\{b(X) : X \subseteq V\}$ can be computed in time $\mathcal{O}(n^5T + n^6)$.*

Whenever a submodular function minimization is used as a subroutine, its running time will be denoted by Υ for the sake on simplicity.

*Research is supported in part by the Hungarian National Research, Development and Innovation Office grant NKFI-132524.

Definition 2.3. *For a directed graph $G = (V, A)$ and a subset of vertices $X \subseteq V$, let $\tilde{\varrho}(X)$ and $\tilde{\delta}(X)$ denote the set of edges entering X and leaving X, respectively. For a vector $x \in \mathbb{R}^A$, let*

$$\varrho_x(X) := \sum_{e\in\tilde{\varrho}(X)} x(e), \quad \delta_x(X) := \sum_{e\in\tilde{\delta}(X)} x(e) \quad \text{and} \quad \partial_x(X) := \varrho_x(X) - \delta_x(X). \tag{3}$$

Furthermore, let $\varrho(X)$, $\delta(X)$ and $\partial(X)$ denote the number of edges entering X, leaving X, and their difference, respectively.

It is straightforward to check that $\varrho_x(X)$ and $\delta_x(X)$ are submodular functions for any nonnegative vector x. If $l, u \in \mathbb{R}^A$ and $l \leq u$, then $\varrho_u(X) - \delta_l(X)$ is submodular and $\varrho_l(X) - \delta_u(X)$ is supermodular.

Definition 2.4. *Let us given a directed graph $G = (V, A)$ and a submodular function $b : \mathcal{P}(V) \to \mathbb{R}$. A vector $x \in \mathbb{R}^A$ is called a* submodular flow *if*

$$\varrho_x(X) - \delta_x(X) \leq b(X) \tag{4}$$

holds for each $X \subseteq V$.

For vectors $l, u \in \mathbb{R}^A$, a submodular flow x is called (l, u)-bounded *if $l \leq x \leq u$.*

Theorem 2.5. *For lower and upper bounds $l, u \in \mathbb{R}^A$, there exists an (l, u)-bounded submodular flow if and only if $l \leq u$ and*

$$\varrho_l(X) - \delta_u(X) \leq b(X) \tag{5}$$

holds for each $X \subseteq V$.

A general aim in balanced optimization is to minimize the difference between the values present in a solution, intuitively resulting in the following definition of spread.

Definition 2.6. *The* spread *of a vector $x \in \mathbb{R}^A$ is the value*

$$\sigma(x) := \max_{a\in A} x(a) - \min_{a\in A} x(a)$$

.

The *Balanced Submodular Flow Problem* is to find a submodular flow of minimum spread.

Problem 2.7. *Given a directed graph $G = (V, A)$ and a submodular function $b : \mathcal{P}(V) \to \mathbb{R}$. Find*

$$\sigma^* := \min \left\{\sigma(x) : x \in \mathbb{R}^A,\ \varrho_x(X) - \delta_x(X) \leq b(X) \quad \forall X \subseteq V\right\} \tag{6}$$

along with a minimizing vector x^*.

Theorem 2.8 ([2])**.** *Assuming that the value of b can be computed for any $X \subseteq V$ in time T, then Problem 2.7 is solvable in time $\mathcal{O}(m^7 T + n^6)$.*

3 l-balanced Submodular Flows

In the followings, a modified definition of the spread is considered with a parameter referred to as α.

Definition 3.1. *For a given norm $||\cdot|| : \mathbb{R}^A \longrightarrow \mathbb{R}$, the* α-spread *of a vector $x \in \mathbb{R}^A$ is the value*

$$\sigma_\alpha(x) := ||x(a) - \alpha\mathbb{1}||$$

.

The α-l-Balanced Submodular Flow Problem w.r.t. the norm $||\cdot||_l : \mathbb{R}^A \longrightarrow \mathbb{R}$ is defined as follows.

Problem 3.2. *Let us given a directed graph $G = (V, A)$, a submodular function $b : \mathcal{P}(V) \to \mathbb{Z}$ and an $\alpha \in \mathbb{R}$. Find*

$$\sigma^*_\alpha := \min \left\{||x(a) - \alpha\mathbb{1}||_l : x \in \mathbb{R}^A \quad \varrho_x(X) - \delta_x(X) \leq b(X) \quad \forall X \subseteq V\right\} \tag{7}$$

along with a minimizing vector x^*.

Then the l-Balanced Submodular Flow Problem w.r.t. the norm $||\cdot||_l : \mathbb{R}^A \longrightarrow \mathbb{R}$ is the following.

Problem 3.3. *Let us given a directed graph $G = (V, A)$, a submodular function $b : \mathcal{P}(V) \to \mathbb{Z}$. Find*

$$\sigma^* := \min_{\alpha\in\mathbb{R}} \sigma^*_\alpha \tag{8}$$

Note, that in case of maximum norm (i.e. $||x|| := \max\{x_a : a \in A\}$) the optimal value to Problem 3.3 is twice to optimal spread of the Balanced Submodular Flow problem.

4 l_1-balanced Submodular Flows

In this section, let us consider l-Balanced Submodular Flow Problem.

Given a specific value of α, it is natural to ask how to compute a feasible submodular flow with minimum α-l_1-spread. The subsequent claim provides the solution to this question.

Claim 4.1. *For any given value of α, a submodular flow with the minimal α-$\sum$-spread can be obtained by a single minimum cost submodular flow computation. Namely, the following value can be computed.*

$$\sigma^*_{\alpha-\sum} := \min\left\{\sigma_{\alpha-\sum}(x) : x \in \mathbb{R}^A \quad \varrho_x(X) - \delta_x(X) \leq b(X) \quad \forall X \subseteq V\right\} \tag{9}$$

Let us consider the following LP model of the l_1-balanced Submodular Flow Problem. Let G' be the graph obtained from G by duplicating every edge in the opposite direction.

$$\begin{aligned} &\min(\mathbb{1}x') \\ &\varrho^{G'}_{x'}(U) - \delta^{G'}_{x'}(U) + \alpha\partial^G(U) \leq b(U) && \forall U \subseteq V \\ &x'(a') \geq 0 && \forall a' \in A' \end{aligned}$$

The dual problem is the following.

$$\begin{aligned} &\min\Big(\sum_{U\subseteq V} y(U)b(U)\Big) \\ &\sum_{U:a'\in\varrho^{G'}(U)} y(U) - \sum_{U:a'\in\delta^{G'}(U)} y(U) \geq -1 && \forall a' \in A' \\ &\sum_{U\subseteq V} y(U)\partial(U) = 0 \end{aligned}$$

In order to solve problem **??**, the idea is to compute the dual optimum as follows. We start with an arbitrary dual feasible solution and check whether a primal solution satisfying all the complementary slackness conditions exists. If such a solution exists, then the optimal solution is found. Otherwise, we improve the current dual solution and repeat the procedure.

Let y be an arbitrary dual solution. The following notation will be useful.

$$\begin{aligned} \mathcal{U} &= \{U \subseteq V \mid y(U) > 0\} \\ \mathcal{A}' &= \{a' \in A' \mid \sum_{a'\in\varrho^{G'}(U)} y(U) - \sum_{a\in\delta^{G'}(U)} y(U) > -1\} \\ b_\alpha(U) &= b(U) - \alpha\partial^G(U) \quad \forall U \subseteq V \end{aligned}$$

Let us first fix a value of α and then seek a solution x' that fulfills the conditions of primal feasibility and complementary slackness. That is we are looking for x such that the followings hold. Due to the fact that b is submodular, we can assume that $\mathcal{U}$ is a chain. We seek for a vector x' that holds the followings.

$$\varrho^{G'}_{x'}(U) - \delta^{G'}_{x'}(U) \leq b_\alpha(U) \qquad \forall U \subseteq V \tag{10}$$

$$\varrho^{G'}_{x'}(U) - \delta^{G'}_{x'}(U) = b_\alpha(U) \qquad \forall U \in \mathcal{U} \tag{11}$$

$$x'(a') \geq 0 \qquad \forall a' \in A' \tag{12}$$

$$x'(a') = 0 \qquad \forall a' \in \mathcal{A}' \tag{13}$$

Condition (13) can easily be guaranteed by deleting all the edges in $\mathcal{A}'$ from the graph. The following straightforward claim show that (10)-(12) defines a submodular flow polyhedron, thus we can compute a feasible vector x'.

Claim 4.2. *Let $G = (V, A)$ be a directed graph, b a submodular function and $\mathcal{U}$ is a family of subsets of V. If $\mathcal{U} = \{U_1, U_2 \ldots U_k\}$, where $\emptyset = U_0 \subset U_1 \subset U_2 \subset \cdots \subset U_k \subset U_{k+1} = V$ is a chain, then a solution of*

$$\begin{aligned} &\varrho^G_x(U) - \delta^G_x(U) \leq b(U) && \forall U \subseteq V \\ &\varrho^G_x(U) - \delta^G_x(U) = b(U) && \forall U \in \mathcal{U} \end{aligned}$$

is a submodular flow in G with the following submodular function

$$b^{\mathcal{U}}(X) = \sum_{i=0,1,2\ldots k} \Big(b(X \cap U_{i+1} \cup U_i) - b(U_i)\Big). \tag{14}$$

Algorithm 1 Minimum spread calculation

$\mathcal{U} = \emptyset,\ \alpha = 0$

$b_{\mathcal{U}}(X) = \sum_{i=1,2\ldots k} \Big(b(X \cap U_{i-1} \cup U_i) - b(U_i) \Big)$

$C = \arg\min \Big(b_{\mathcal{U}}(X) - \alpha \partial_A(X) \mid \delta_{A' \setminus \mathcal{A}'}(X) = 0 \Big)$

$X = \emptyset, Y = \emptyset$

while $b_{\mathcal{U}}(C) - \alpha \partial_A(C) < 0$ **do**

  if $\partial_A(C) > 0$ **then**

    $\alpha = \frac{b_{\mathcal{U}}(C)}{\partial_A(C)}$

    $X = C$

  else if $\partial_A(C) < 0$ **then**

    $\alpha = \frac{b_{\mathcal{U}}(C)}{\partial_A(C)}$

    $Y = C$

  else if $X \neq \emptyset$ and $Y \neq \emptyset$ **then**

    Add X, Y to $\mathcal{U}$

    $\alpha = 0,\ X = \emptyset, Y = \emptyset$

  else

    Add C to $\mathcal{U}$

    $\alpha = 0,\ X = \emptyset, Y = \emptyset$

The procedure is outlined in Algorithm 1.

In the following, we show that how to modify the value of α or the dual vector y when there is no vector x satisfying (10)-(13). In this case we can either correct the value of α or add one or two new sets to $\mathcal{U}$. According to Claim 4.2, there is a set C that satisfies the followings.

$$b_\alpha^{\mathcal{U}}(C) < 0 \tag{15}$$

$$\delta^{A' \setminus \mathcal{A}'}(U) = \emptyset \tag{16}$$

In other words

$$\sum_{i=0,1,2\ldots k} \Big(b_\alpha(C \cap U_{i+1} \cup U_i) - b_\alpha(U_i) \Big) < 0$$

$$\sum_{i=0,1,2\ldots k} b_\alpha(C \cap U_{i+1} \cup U_i) < \sum_{i=0,1,2\ldots k} b_\alpha(U_i)$$

$$\sum_{i=0,1,2\ldots k} b(C \cap U_{i+1} \cup U_i) - \alpha \partial^G(C \cap U_{i+1} \cup U_i) < \sum_{i=0,1,2\ldots k} b(U_i) - \alpha \partial^G(U_i)$$

Note that

$$\partial^G(C \cap U_{i+1} \cup U_i) = \partial^G(C \cap U_{i+1}) + \partial^G(U_i) - \partial^G(C \cap U_{i+1} \cap U_i) =$$
$$\partial^G(C \cap U_{i+1}) + \partial^G(U_i) - \partial^G(C \cap U_i)$$

Thus

$$\sum_{i=0,1,2\ldots k} b(C \cap U_{i+1} \cup U_i) - \sum_{i=0,1,2\ldots k} \alpha \partial^G(C \cap U_{i+1}) - \sum_{i=0,1,2\ldots k} \alpha \partial^G(U_i) + \sum_{i=0,1,2\ldots k} \alpha \partial^G(C \cap U_i) <$$
$$\sum_{i=0,1,2\ldots k} b(U_i) - \sum_{i=0,1,2\ldots k} \alpha \partial^G(U_i)$$

$$\sum_{i=0,1,2\ldots k} b(C \cap U_{i+1} \cup U_i) - \sum_{i=0,1,2\ldots k} \alpha \partial^G(C \cap U_{i+1}) + \sum_{i=0,1,2\ldots k} \alpha \partial^G(C \cap U_i) < \sum_{i=0,1,2\ldots k} b(U_i)$$

$$\sum_{i=0,1,2\ldots k} b(C \cap U_{i+1} \cup U_i) - \alpha \partial^G(C \cap U_{k+1}) + \alpha \partial^G(C \cap U_0) < \sum_{i=0,1,2\ldots k} b(U_i)$$

$$\sum_{i=0,1,2\ldots k} b(C \cap U_{i+1} \cup U_i) - \alpha \partial^G(C) < \sum_{i=0,1,2\ldots k} b(U_i)$$

If $\partial^G(C) > 0$, this is equivalent to

$$\frac{\sum_{i=0,1,2\ldots k} \Big(b(C \cap U_{i+1} \cup U_i) - b(U_i) \Big)}{\partial^G(C)} < \alpha.$$

By choosing the value of α to be equal to the left side of the above inequality, we get that C is no longer satisfies (15). Similar statement holds if $\partial^G(C) < 0$. Let us consider the case when there are two sets X, Y, such that $\partial^G(X) > 0$, $\partial^G(Y) < 0$ and

$$\frac{\sum_{i=0,1,2\ldots k} \Big(b(X \cap U_{i+1} \cup U_i) - b(U_i) \Big)}{\partial^G(X)} < \frac{\sum_{i=0,1,2\ldots k} \Big(b(Y \cap U_{i+1} \cup U_i) - b(U_i) \Big)}{\partial^G(Y)}. \tag{17}$$

Let us define the following value.

$$\epsilon(a') = 1 + \sum_{vu \in \varrho^{G'}(U_i)} y(U_i) - \sum_{vu \in \delta^{G'}(U_i)} y(U_i)$$

Therefore, we get a dual solution with a smaller cost if we can perform the following changes on the dual vector y

$$\begin{aligned}
& y'(U_i) = y(U_i) - \lambda \left(\partial^G(X) + |\partial^G(Y)| \right) && \forall i \in \{1, 2 \ldots k\} \\
& y'((X \cap Y) \cap U_{i+1} \cup U_i) = \lambda \min \left(\partial^G(X), |\partial^G(Y)| \right) && \forall i \in \{1, 2 \ldots k\} \\
& y'((X \cup Y) \cap U_{i+1} \cup U_i) = \lambda \min \left(\partial^G(X), |\partial^G(Y)| \right) && \forall i \in \{1, 2 \ldots k\} \\
& y'(X \cap U_{i+1} \cup U_i) = \lambda \left(|\partial^G(Y)| - \partial^G(X) \right) && \text{if } |\partial^G(Y)| > \partial^G(X), i \in \{1, 2 \ldots k\} \\
& y'(Y \cap U_{i+1} \cup U_i) = \lambda \left(\partial^G(X) - |\partial^G(Y)| \right) && \text{if } |\partial^G(Y)| < \partial^G(X), i \in \{1, 2 \ldots k\},
\end{aligned}$$

where the value of λ is the minimum of the followings.

$$\min_{a' \in \delta(X)} \frac{\epsilon(a')}{\partial^G(X)} \tag{18}$$

$$\min_{a' \in \delta(Y)} \frac{\epsilon(a')}{|\partial^G(Y)|} \tag{19}$$

$$\min_{a' \in \delta(X \cap Y)} \frac{\epsilon(a')}{\min\left(\partial^G(X), |\partial^G(Y)|\right)} \tag{20}$$

$$\min_{a' \in \delta(X \cup Y)} \frac{\epsilon(a')}{\min\left(\partial^G(X), |\partial^G(Y)|\right)} \tag{21}$$

$$\min_{i=1,2\ldots k} \frac{y(U_i)}{\partial^G(X) + |\partial^G(Y)|} \tag{22}$$

Finally, let us consider the case when $\partial^G(C) = 0$. Then,

$$\sum_{i=0,1,2\ldots k} b(C \cap U_{i+1} \cup U_i) - \alpha \partial^G(C) < \sum_{i=0,1,2\ldots k} b(U_i)$$

holds for any α and the dual solution is improved as follows. Let us define the following new dual solution.

$$\begin{aligned}
& y'(U_i) = y(U_i) - \lambda && \forall i \in \{1, 2 \ldots k\} \\
& y'(C \cap U_{i+1} \cup U_i) = \lambda && \forall i \in \{1, 2 \ldots k\}
\end{aligned}$$

where the value of λ is the minimum of the followings.

$$\min_{a' \in \delta(C)} \epsilon(a') \tag{23}$$

$$\min_{i=1,2\ldots k} y(U_i) \tag{24}$$

The following claims hold in any cases described above.

Claim 4.3. *The modified dual vector y' is dual-feasible.*

Claim 4.4.

$$\sum_{U \subseteq V} y'(U)b(U)) < \sum_{U \subseteq V} y(U)b(U))$$

We omit the detailed technical proofs of the lemmas above.

Therefore, the modified dual vector is feasible and its cost is smaller than the cost of the previous dual solution.

Note that if the Algorithm execute line 7 in two consecutive iterations with C_1 and C_2, then $\partial(C_1) > \partial(C_2)$. Similarly, if the Algorithm execute line 10 in two consecutive iterations with C_1 and C_2, then $\partial(C_1) < \partial(C_2)$. Thus, Algorithm 1 takes at most m consecutive iteration, where the dual solution does not change. (This means that there is no new set added to $\mathcal{U}$ by the algorithm.) The following claim ensures that the procedure finishes after strongly polynomial iterations.

Claim 4.5. *Let us define the following sets.*

$$\mathcal{U} = \{U \subseteq V \mid y(U) > 0\}$$
$$\mathcal{U}' = \{U \subseteq V \mid y'(U) > 0\}$$

Then

$$\sum_{U \in \mathcal{U}} |U| < \sum_{U' \in \mathcal{U}'} |U'|$$

To sum up everything, we get the following theorem.

Theorem 4.6. *Algorithm 1 solves the l_1-balanced Submodular Flow Problem in $\mathcal{O}(n^2 m \Upsilon)$ time, where Υ denotes the time complexity of a submodular function minimization.*

References

[1] Paolo M. Camerini et al. "Most and least uniform spanning trees". In: *Discrete Applied Mathematics* 15 (2-3 1986), pp. 181–197. DOI: https://doi.org/10.1016/0166-218X(86)90041-7.

[2] Alpár Jüttner and Eszter Szabó. *Balanced Submodular Flows*. 2023. arXiv: 2308.12404 [math.OC]. URL: https://arxiv.org/abs/2308.12404.

[3] Silvano Martello et al. "Balanced optimization problems". In: *Operations Research Letters* 3.5 (1984), pp. 275–278. ISSN: 0167-6377. DOI: https://doi.org/10.1016/0167-6377(84)90061-0.

[4] James Orlin. "A Faster Strongly Polynomial Time Algorithm for Submodular Function Minimization". In: vol. 118. May 2009, pp. 237–251. DOI: https://doi.org/10.1007/s10107-007-0189-2.

QUBO formulation of some spanning tree related problems

Katalin Friedl*

Department of Computer Science and Information Theory
Budapest University of Technology and Economics
H-1111 Budapest, Műegyetem rakpart 3., Hungary
friedl@cs.bme.hu

László Kabódi*

Department of Computer Science and Information Theory
Budapest University of Technology and Economics
H-1111 Budapest, Műegyetem rakpart 3., Hungary
kabodil@cs.bme.hu

Abstract: The appearance of quantum annealers, like the D-Wave machines, made it interesting to formulate problems as Quadratic Unconstrained Binary Optimization tasks. In this paper we show two new QUBO formulations to the spanning tree problem, and use it to find a minimum weight spanning tree, a minimum weight spanning tree with a maximal degree constraint and a Steiner tree. We show that variants of the classical Prüfer sequences can be used for this purpose and they are as good as earlier results. Our network flow based approach improves on the earlier results when the number of edges is at most $O(\frac{n^2}{\log n})$.

Keywords: QUBO, spanning tree, Steiner tree

1 Introduction

Quantum computers promise significant speed ups in some problem classes. One of these classes is Quadratic Unconstrained Binary Optimization (QUBO), which is implemented, for example, in D-Wave quantum annealers. An interesting area of research is to create QUBO formulations for well-known classical problems. In this paper we show two new QUBO formulations to the spanning tree problem, and use them to find a minimum weight spanning tree, a minimum weight spanning tree with a maximal degree constraint and a Steiner tree. In Section 2 we give a brief introduction to QUBO, and show some basic tools that can be used when formulating a task in QUBO. In Section 3 we give a formulation for finding an arbitrary spanning tree in a given graph. Section 4 shows how to extend this to find a minimum weight spanning tree. Section 5 further extends this to a bounded degree minimum weight spanning tree. In Section 6 we show how to use the same approach to find an arbitrary Steiner tree. Section 7 contains our conclusions.

2 QUBO

A quantum annealer, like the D-Wave machines, finds the minimum of a given quadratic function over binary variables. In quantum computing, the function to be minimized is also called the Hamiltonian, and the binary variables are called qubits.

Formally, a QUBO task is to find the minimum of a function $f(x_1, \dots, x_n) = \sum_{i=1}^{n} \alpha_i x_i + \sum_{i=1}^{n} \sum_{j=1}^{n} \beta_{i,j} x_i x_j$ over the binary variables $x_1, \dots, x_n$. Creating a QUBO formulation for a problem means mapping binary variables to the parameters of the task, and giving a function, the minimum of which is where the value of the variables gives the solution of the original task.

2.1 Example: unary encoding

A number $1 \leq k \leq n$ can be encoded into n qubits $x_1, \dots, x_n$ in such a way, that $x_k = 1$ and $x_i = 0$ for all $i \neq k$. The Hamiltonian for this encoding is $H(x_1, \dots, x_n) = (1 - \sum_{i=1}^{n} x_i)^2$. The minimum value of the

*This research was partially supported by the Ministry of Culture and Innovation and the National Research, Development and Innovation Office within the Quantum Information National Laboratory of Hungary (Grant No. 2022-2.1.1-NL-2022-00004).