Directed Network Design with Orientation Constraints*

Sanjeev Khanna † Joseph (Seffi) Naor‡ F. Bruce Shepherd§

Abstract

We study directed network design problems with *orientation constraints*. An orientation constraint on a pair of nodes u and v states that a feasible solution may include at most one of the arcs (u, v) and (v, u). Such constraints arise naturally in many network design problems, since link or edge resources such as fibre can be used to support traffic in one of two possible directions only. Our first result is that the directed network design problem with orientation constraints can be solved in polynomial time in the case where the requirement function f is intersecting supermodular. (The case where there are no orientation constraints follows from work of Frank [6].) The second main result of the paper is a 4-approximation algorithm for the minimum cost strongly connected subgraph problem with orientation constraints. Our algorithm shows that the problem of enforcing orientation constraints can be reduced to the minimum cost 2-edge connected subgraph problem on undirected graphs. Finally, we study the problem for general crossing supermodular functions and show the following bi-criteria approximation result. Let k denote the maximum requirement of any set under the given requirement function f. We give a $2k$-approximation algorithm to construct a solution that satisfies a slightly weaker requirement function, namely, $f'(S) = \max\{f(S) - 1, 0\}$.

*An extended abstract appeared in the proceedings of the 11th Annual ACM-SIAM Symposium on Discrete Algorithms (2000), pp. 663–671.

†Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA 19104. This work was done when the author was at Bell Laboratories, Lucent Technologies, 700 Mountain Avenue, Murray Hill, NJ 07974. Email : `sanjeev@cis.upenn.edu`.

‡Computer Science Department, Technion, Haifa 32000, Israel. This work was done when the author was at Bell Laboratories, Lucent Technologies, 700 Mountain Avenue, Murray Hill, NJ 07974. E-mail: `naor@cs.technion.ac.il`.

§Bell Laboratories, Lucent Technologies, 700 Mountain Avenue, Murray Hill, NJ 07974. E-mail: `bshep@research.bell-labs.com`.

1 Introduction

We study directed network design problems with *orientation constraints*. Specifically, we study design problems in the following framework:

INPUT: We are given a directed graph $D = (V, A)$ with a cost function $c : A \rightarrow \mathcal{Z}$, a requirement function f defined over all subsets of V, and a disjoint collection of *constrained arc pairs*, each of which induces a *digon* (i.e. two arcs directed in opposite directions). Let $A_p \subseteq A$ denote the set of constrained arcs, and for any arc $a \in A_p$, let a^- denote the arc that a is paired with.

GOAL: Find an optimal solution to the integer program below; here $\delta^+(S)$ denotes the set of arcs leaving $S \subseteq V$.

$$(I) \qquad \min \quad \sum_{a \in A} c_a x_a$$

$$\begin{aligned} \sum_{a \in \delta^+(S)} x_a \geq f(S) & \quad \text{for each } S \subset V & (1) \\ x_a + x_{a^-} \leq 1 & \quad \text{for each } a \in A_p & (2) \\ x_a \in \{0, 1\} & \quad \text{for each } a \in A & (3) \end{aligned}$$

We refer to constraints (1.1) as the *cut* constraints, constraints (1.2) as the *orientation* constraints, and constraints (1.3) as the *integrality* constraints.

The above framework, without the orientation constraints, has been studied extensively and it captures a large number of fundamental combinatorial optimization problems. Some representative examples include minimum cost branchings, minimum cost k-strongly connected subgraphs, and the directed steiner network problem. Many of these problems are NP-hard and thus a large amount of research has focused on the design of polynomial-time approximation algorithms for these problems.

In the present paper, we restrict attention to *crossing supermodular* requirement functions f. That is, for every $X, Y \subseteq V$ such that $X \cap Y \neq \emptyset$ and $X \cup Y \neq V$, we have that

$$f(X) + f(Y) \leq f(X \cap Y) + f(X \cup Y).$$

Directed network design problems with a crossing supermodular requirement function remain NP-hard. An example of a crossing supermodular function is the function $f(S) = k$ for all subsets $S \subset V$; this is known as the minimum cost k-strongly connected subgraph problem. For this case, a simple 2-approximation algorithm is obtained by solving two minimum cost k-disjoint arborescence problems (one into and one out from) at an arbitrary node v [5].

Frank [6] showed that in the special case where the requirement function is also *intersecting supermodular*, i.e., the above inequality holds whenever X and Y intersect, the network design problem can be solved optimally in polynomial time. Melkonian and Tardos

[17] have recently shown that this result can be used to obtain a 2-approximation algorithm for any requirement function which is crossing supermodular.

In undirected graphs, weakly supermodular functions have been widely studied as they model a broad class of network design problems, including for instance, the generalized Steiner tree problem. Following a long line of work ([1, 10, 11, 20]), Jain [13] devised an ingenious 2-approximation algorithm for weakly supermodular functions. He proved that every basic feasible solution to the linear programming (LP) relaxation of the problem contains a variable of value at least a half. Jain's algorithm finds and rounds such a *large component* iteratively until a final integral solution is obtained.

Network design problems in undirected graphs are generally much better understood than their directed counterparts. In particular, techniques for network design problems on undirected graphs, e.g., the widely used primal-dual approach [1, 10], do not seem to be easily amenable to directed network design problems. A recent result of Melkonian and Tardos[17] gives a directed graph analog of Jain's result and requires significant further insight into the combinatorial structure of basic solutions. They show that every basic solution to an LP relaxation of the design problem contains a variable of value at least a quarter whenever the requirement function f is crossing supermodular. We note that the LP relaxation for (I) is polynomially solvable by the ellipsoid method [12].

1.1 Our Problems

We study directed network design problems with orientation constraints (1.2) (as specified in (I)). An orientation constraint on a pair of nodes u and v states that a feasible solution may include at most one of the arcs (u, v) and (v, u).[1] A case of special interest in our study is the design of strongly connected directed graphs with orientation constraints.

We can view our directed network design problem as a two-phase problem: finding a subgraph of an undirected graph and then orienting its edges so as to satisfy the cut constraints. The cost function associated with the orientation may in general be *asymmetric*, i.e., the cost of orienting an edge $e = uv$ from v to u is different from orienting it from u to v. (An edge can only be oriented in one direction.) The cost of an orientation is defined to be the sum of the costs of the orientations of the edges.

Orientation constraints arise in many network design problems, since link/edge resources such as fibre, are commonly unidirectional (i.e. they support traffic in only one of the two possible directions at a given time). Asymmetric costs may arise in many network routing problems. For instance, consider the setting where traffic demand is being incrementally introduced in an existing network. Load balancing constraints may favor forcing traffic in opposite directions between a given pair of switches. Hence, when routing new demands, costs on the directed links may increase proportionately to the amount of existing traffic. Asymmetric costs may also arise in network planning due to assorted line termination equipment; these are the costs associated with terminating the two ends of a link.

[1] We will actually address a more general problem which allows for many copies of arcs between pairs of nodes.

An interesting special case of the asymmetric orientation problem is when the constraint (1.2) in (I) is replaced by $x_a + x_{a^-} = 1$ for each arc $a \in A$. Equivalently, we are given an undirected graph and we need to find a minimum cost orientation that satisfies the requirement function f. Younger [21] had observed that for strongly connected orientations, a good characterization for this problem follows from the classical min-max theorem of Lucchesi and Younger [16]. The problem for general crossing supermodular requirement functions can also be solved in polynomial time via a reduction to submodular flows [9]. (See Appendix A for more details.)

Perhaps, the most basic orientation problem with asymmetric costs that involves both subgraph constraints and orientation constraints is finding among all subgraphs of G that admit a strong orientation, one that has a strong orientation of minimum cost. This problem generalizes two well known NP-hard problems. If the orientation cost function is symmetric, then the problem reduces to finding a minimum cost 2-edge connected subgraph of G. On the other hand, if there are no orientation constraints, then the problem reduces to finding a minimum cost strongly connected subgraph of a directed graph. We note that for both problems, 2-approximation algorithms are known.

Thus our network design problems combine constraints of two types, subgraph constraints, and orientation constraints. While each type of constraint has been well-studied separately, not much seems to be known for design problems that combine these two types of constraints simultaneously.

1.2 Our Results

Our first result is that the directed network design problem with orientation constraints can be solved in polynomial time in the case where the requirement function f is intersecting supermodular. We show that any basic solution for the relaxation of (I) in this case has integral components. This generalizes the work of Frank [6] who proved the same result for the variant with no orientation constraints. In fact, we establish our result for a generalization of (I) where arcs may have arbitrary integral capacities.

Our second result is that the minimum cost strongly connected subgraph problem with orientation constraints has a 4-approximation algorithm. We give a combinatorial approximation algorithm based on the idea that the problem of enforcing orientation constraints can be reduced to the minimum cost 2-edge connected subgraph problem. We start with any feasible solution to the minimum cost strongly connected subgraph problem and use the above reduction to modify the solution so as to satisfy the violated orientation constraints.

Finally, we study our problem for general crossing supermodular functions and show the following bi-criteria approximation result. Let k denote the maximum requirement of any set under the given requirement function f. We give a $2k$-approximation algorithm that constructs a solution satisfying a slightly weaker requirement function, namely, $f'(S) = \max\{f(S) - 1, 0\}$.

In Appendix A, we discuss in more detail the algorithms for the case where all orientation constraints are set at equality and the requirement function is crossing supermodular. This version of the problem can be solved optimally in polynomial time via work of Frank [9] and Edmonds and Giles [3]. We also describe a clever proof of Younger [21] which had not previously appeared.

2 Preliminaries

We denote a directed graph by $D = (V, A)$. For $S \subseteq V$, denote by $\delta^+(S)$ (resp. $\delta^-(S)$) the set of arcs with tail in S (resp. $V - S$) and head in $V - S$ (resp. S).

A pair of subsets X, Y, of a ground set V, is *intersecting* if $X \cap Y \neq \emptyset$. An intersecting pair X, Y of sets is *crossing* if $X \cup Y \neq V$ and X, Y are noncomparable. A family $\mathcal{F}$ of nonempty subsets of V is *intersecting* if we have $X \cap Y, X \cup Y \in \mathcal{F}$ for each intersecting pair $X, Y \in \mathcal{F}$. A family is *crossing* if $X \cap Y, X \cup Y \in \mathcal{F}$ for each crossing pair $X, Y \in \mathcal{F}$. A function $f : 2^V \to \mathcal{Z}_+$ is *crossing* (resp. *intersecting supermodular*) on a crossing (resp. intersecting) family $\mathcal{F}$, if:

1. $f(V) = f(\emptyset) = 0$.
2. For each crossing (resp. intersecting) pair $X, Y \in \mathcal{F}$ such that $f(X), f(Y) > 0$, $f(X) + f(Y) \leq f(X \cap Y) + f(X \cup Y)$.

We emphasize that we only require the inequality to hold for X, Y in the support of f.[2]

For any ordered pair (u, v) of nodes of V, we define an operator Φ_{uv} as follows. Given any $f : 2^V \to \mathcal{Q}_+$, $\Phi_{uv}(f)$ is a new function such that $\Phi_{uv}(f)(S) = f(S) - 1$ if $u \in S, v \notin S$ and $f(S) > 0$, and $\Phi_{uv}(f)(S) = f(S)$, otherwise. The following is shown in [6].

Lemma 1 *If f is crossing (resp. intersecting) supermodular on $\mathcal{F}$, then $\Phi_{uv}(f)$ is also crossing (resp. intersecting) supermodular.*

For an integer $k > 0$, we denote by $\Phi^k_{uv}(f)$ the function obtained by applying Φ_{uv} k times to f. When f is clear from the context, we denote by $\mathcal{F}(uv)$ the family obtained from $\mathcal{F}$ by removing all sets $S \neq V, \emptyset$ for which $\Phi_{uv}(f)(S) = 0$. One can easily verify that if f is crossing (intersecting) supermodular on the family $\mathcal{F}$ then it is also crossing (intersecting) supermodular on $\mathcal{F}(uv)$. We can also define $\mathcal{F}^k(uv)$ in an iterative manner.

A family $\mathcal{S} = \{S_i\}_{i=1}^m$ of proper, nonempty subsets of a finite ground set V is *laminar* if no pair of sets in $\mathcal{S}$ cross. The family is *strongly laminar* if for each pair S_i, S_j of distinct sets in $\mathcal{S}$, we have either $S_i \subseteq S_j, S_j \subseteq S_i$, or $S_i \cap S_j = \emptyset$.

3 Intersecting Supermodularity

In this section we study the polyhedron obtained by relaxing the integrality constraints (1.3) in (I), i.e., for each $a \in A$, we now require only $0 \leq x_a \leq 1$. In fact, we introduce a more

[2] This property has also been referred to as **weakly** crossing (intersecting) supermodularity in [6], whereas supermodularity referred to functions which satisfy condition (2) for all crossing (intersecting) pairs X, Y.

general integer program below. We assume throughout that f is crossing supermodular.

Let D be a digraph and $\mathcal{E}$ be a disjoint collection of arc pairs, a and a^-, each of which forms a *digon*, i.e., directed circuit of length two. Denote by $\mathcal{F}$ an intersecting family of subsets of V, and let f be an intersecting supermodular function on $\mathcal{F}$. A quadruple of the form above will be called simply an *f-connectivity problem.* This problem is an *intersecting* (*crossing*) supermodular f-connectivity if f is *intersecting* (*crossing*) supermodular. Denote by $P(D, f, \mathcal{F}, \mathcal{E})$ the polyhedron defined by the above relaxation.

A *capacitated* f-connectivity problem is a quintuple $(D, f, \mathcal{F}, \mathcal{E}, u)$ where $(D, f, \mathcal{F}, \mathcal{E})$ is an f-connectivity problem and $u : A \cup \mathcal{E} \to \mathcal{Z}_+$ is an assignment of capacities to the arcs and digons of $\mathcal{E}$. For such a problem, we amend the constraints of problem (I) so that $x_a \leq u_a$ for each arc a and $x_a + x_{a^-} \leq u_{a,a^-}$ for each digon $\{a, a^-\} \in \mathcal{E}$. We are again interested primarily in integer solutions to this capacitated f-connectivity problem and we denote by $P(D, f, \mathcal{F}, \mathcal{E}, u)$ the obvious linear relaxation.

Our goal is to prove the following.

Theorem 2 *If $(D, f, \mathcal{F}, \mathcal{E}, u)$ is an intersecting supermodular f-connectivity problem, then the extreme points of $P(D, f, \mathcal{F}, \mathcal{E}, u)$ are integral.*

The special case where there are no orientation constraints follows from Frank [6]. We note here that it is easy to construct examples such that there is an unbounded gap between the cost of optimal solutions with and without orientation constraints.

We will in fact prove a slightly stronger result than Theorem 2 which may be applicable to some instances of crossing f-connectivity problems. Henceforth we let $P(D, f, \mathcal{F}, \mathcal{E}, u)$, be a crossing supermodular f-connectivity polyhedron and x^* be an extreme point. A vector x^* is *slack* if for each arc a: $0 < x^*_a < u_a$. Note that any slack extreme point has a *defining system* determined by a pair $\mathcal{S}, \mathcal{R}$ where $\mathcal{S} \subseteq \mathcal{F}$, $\mathcal{R} \subseteq \mathcal{E}$ and $m = |A| = |\mathcal{S}| + |\mathcal{R}|$. That is, x^* is the unique solution (in R^A) to the system of equalities:

1. $x_a + x_{a^-} = u_{a,a^-}$, for each $\{a, a^-\} \in \mathcal{R}$
2. $x(\delta^+(S)) = f(S)$, for each $S \in \mathcal{S}$

and in particular, $\mathcal{S}, \mathcal{R}$ identify a set of linearly independent rows in the constraint matrix for the f-connectivity problem. An arc a is *restrained* if a is included in some digon of $\mathcal{R}$; otherwise, a is *free*. Let F, R denote the sets of free and restrained arcs respectively. For $S \in \mathcal{S}$, we denote by $F(S)$ the set of free arcs in $\delta^+(S)$. We define $R(S)$ similarly.

The next several lemmas are dedicated to the analysis of the structure of such an extreme point x^* and this (not necessarily unique) defining system.

The following result is similar in essence to the Lemma 2 in [17].

Lemma 3 *Any slack extreme point x^* has a defining system $\mathcal{S}, \mathcal{R}$ for which $\mathcal{S}$ is laminar.*

Proof: Consider a defining system $\mathcal{S}, \mathcal{R}$ such that there exist $X, Y \in \mathcal{S}$ with each of $X \cap Y, X - Y, Y - X, V - (X \cup Y)$ being nonempty. Let $x_{ab} = x^*([X - Y, Y - X])$, $x_{ba} = x^*([Y - X, X - Y])$, $x_{io} = x^*([X \cap Y, V - [X \cup Y])$, $x_{ia} = x^*([X \cap Y, X - Y])$,

$x_{ib} = x^*([X \cap Y, Y - X])$, $x_{b0} = x^*([Y - X, V - (X \cup Y)])$, $x_{a0} = x^*([X - Y, V - (X \cup Y)])$. Since $X \cup Y \neq V$, we have

$$\begin{aligned}
&(x_{ab} + x_{io} + x_{a0} + x_{ib}) + (x_{io} + x_{ba} + x_{b0} + x_{ia}) \\
&\quad = \quad f(X) + f(Y) \\
&\quad \leq \quad f(X \cap Y) + f(X \cup Y) \\
&\quad = \quad (x_{ib} + x_{ia} + x_{io}) + (x_{io} + x_{b0} + x_{a0})
\end{aligned}$$

from which we deduce that $X \cap Y, X \cup Y$ are also tight for x^*. In fact, we have $x_{ab} = x_{ba} = 0$ and hence the sum of the two constraints X, Y is identical to the sum of the constraints for $X \cap Y, X \cup Y$. Thus replacing X, Y in $\mathcal{S}$ by $X \cap Y, X \cup Y$ we obtain another defining system. Applying this procedure increases the value $\sum_{S \in \mathcal{S}} |S|^2$ and so we iteratively repeat this operation to obtain a laminar family. $\Box$

For any laminar family $\mathcal{S}$ we define an acyclic *laminar directed graph* $H(\mathcal{S})$ with a node v_S associated with each $S \in \mathcal{S}$ and where there is an arc (v_{S_i}, v_{S_j}) if S_i is a minimal set containing S_j. Let $\Phi : A \to V(H(\mathcal{S}))$ be a mapping. The *contribution* to v_S by Φ, denoted by $C(v)$, is

$$\sum_{a \in F, \Phi(a) = v_S} 1 \; + \sum_{a \in R, \Phi(a) = v_S} \frac{1}{2} \, .$$

The function Φ is a *legal mapping* for a subgraph $H' \subseteq H$ if for each $v_S \in V(H')$ we have:

- $C(v_S) \geq 1$.
- the tail of each arc of $\Phi^-(v_S)$ lies in S.

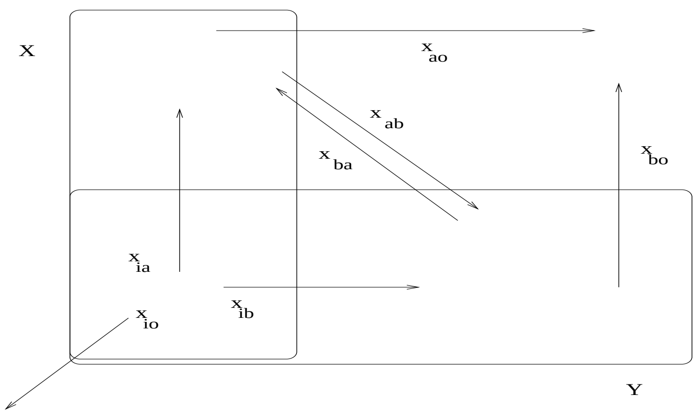

Figure 1: Proofs of Lemmas 3 and 4

Lemma 4 *If $(D, f, \mathcal{F}, \mathcal{E}, u)$ is an intersecting supermodular problem, then any slack extreme point x^* has a defining system such that $\mathcal{S}$ is strongly laminar.*

Proof: We may assume that $\mathcal{S}$ is laminar by Lemma 3. Now suppose that we have $X, Y \in \mathcal{S}$ such that $X \cap Y, X - Y, Y - X$ are all nonempty, and $X \cup Y = V$. We define the parameters

$x_{ab}, x_{ib}, x_{ba}, x_{ia}$ as in Lemma 3 and Figure 1. For example, $x_{io} = \sum_{a \in \delta^+(X \cap Y) \cap \delta^+(X \cup Y)} x_a$. Thus we have

$$\begin{aligned}
&(x_{ab} + x_{ib}) + (x_{ba} + x_{ia}) \\
&= f(X) + f(Y) \\
&\leq f(X \cap Y) + f(X \cup Y) \\
&= f(X \cap Y) \\
&= (x_{ib} + x_{ia})
\end{aligned}$$

from which we deduce that $x_{ab} = x_{ba} = 0$ and that $X \cap Y$ is again tight for x^*. In this case, we may replace the set Y in $\mathcal{S}$ by $X \cap Y$. Let a, b, c be the $0,1$ incidence vectors of $\delta^+(X), \delta^+(Y), \delta^+(X \cap Y)$ respectively. Note that we have $a + c = 2a + b$ and hence the resulting system will again be defining (i.e., induce a nonsingular matrix) for the vector x^*.

Applying this procedure decreases the value $\sum_{S \in \mathcal{S}} |S|$ and does not create any crossing pairs. Thus we may repeatedly apply this operation until we obtain the desired strongly laminar system. $\square$

The following lemma is in slightly more general form than we need; it guarantees an integral component in specially structured extreme points of crossing supermodular constrained polyhedra.

Lemma 5 *Let $x^* > 0$ be a vector that has a defining system $\mathcal{S}, \mathcal{R}$ for which $\mathcal{S}$ is strongly laminar. Then there is exists an arc a such that x^*_a is an integer.*

Proof: Suppose that we have that $x^*, \mathcal{S}, \mathcal{R}$ is a counterexample for a digraph D with a minimum number of arcs and subject to this minimizes $\sum_a x^*_a$. We first claim that without loss of generality

$$\text{Each arc appears in some cut } \delta^+(S) \text{ where } S \in \mathcal{S}. \tag{4}$$

Suppose this is not the case for some a, then clearly $a \in \mathcal{R}$ and the constraint $x_a + x_{a^-} \leq 1$ is the only constraint from the defining system which involves a. For any value z, let x^z be the vector obtained by replacing the value x^*_a by the value z. Let z_0 be the smallest value such that x^{z_0} is feasible. Note that if $z_0 > 0$, then a is in some tight cut $\delta^+(S)$ for x^{z_0}, and so we may add this cut to $\mathcal{S}$ and remove the orientation constraint for $\{a, a^-\}$ to obtain a linearly independent defining system for x^z. Thus $x^z > 0$ is another extreme point that contradicts the minimality of x^*. Finally, if $z_0 = 0$, then by suppressing a's component in x^* to obtain a new vector x', we obtain another completely fractional extreme point for $D - a$. We next prove the following claim.

Claim 6 *If $x^* > 0$ has no integral component, then for each subtree T rooted at a node v_S in $H(\mathcal{S})$ we may find a legal mapping such that $C(v_S) \geq 1 + |F(S)|/2$ and for each other node $v \in T$ we have $C(v) \geq 1$.*

Proof of Claim: Note that since $x^* > 0$, we must have that $|\delta^+(S)| \geq 2$ for each $S \in \mathcal{S}$. Hence the result holds in the case where $|V(H)| = 1$. Now suppose that $v_{S_1}, v_{S_2}, \ldots, v_{S_q}$ are the maximal descendents of v_S. By induction, for each i, there is a legal mapping Φ_i for the subgraph induced by the descendants of v_{S_i} such that the contribution to v_{S_i} is at least $1 + |F(S_i)|/2$. For each i, we first partition the arcs in $\delta^+(S_i)$ as follows. Let $F_i^+ = \delta^+(S_i) \cap F(S_i)$ and $R_i^+ = \delta^+(S_i) \cap R(S_i)$, and let $F_i^- = F(S_i) - F_i^+$ and $R_i^- = R(S_i) - R_i^+$. Finally, let $F' = F(S) - (\cup_i F_i^+)$ and $R' = R(S) - (\cup_i R_i^+)$. The cardinality of each of these sets will be denoted by changing the large capital letter to small, e.g., $f_i = |F_i|$.

We create a legal map Φ by mapping the arcs of F', R' directly to v_S and for each i, we "keep" a contribution of 1 for v_{S_i} and "pass on" a remaining contribution of $|F(S_i)|/2$ under Φ_i. Note that if this does not contribute $1 + |F(S)|/2$ to v_S, then we have

$$f' + \frac{r'}{2} + \frac{1}{2}\sum_{i=1}^{q}(f_i^+ + f_i^-) < 1 + \frac{f'}{2} + \frac{1}{2}\sum_{i=1}^{q} f_i^+.$$

This implies that $f' + r' + \sum_{i=1}^{q} f_i^- \leq 1$. Consider adding the constraint for S together with all orientation constraints $\{a, a^-\}$ for which both ends of a are contained in S. To this combination, subtract the equalities associated with the cuts S_i. We are left with the following equation:

$$\sum_{a \in F' \cup R'} x_a^* - \sum_{i=1}^{q} \sum_{a \in F_i^-} x_a^* = \gamma,$$

where γ is some integer. Note that we require (4) in order to deduce this fact. Thus if $f' = r' = \sum_{i=1}^{q} f_i^- = 0$, then we have a nontrivial combination of tight constraints which yields the zero vector, a contradiction. Thus exactly one of $f', r', f_1^-, f_2^-, \ldots, f_q^-$ is 1. But, in this case, the equality above shows that x^* has an integral component, a contradiction. $\square$

We now continue with the proof of the lemma. Suppose that $S_1, S_2 \ldots S_p$ are the maximal sets in $\mathcal{S}$ and apply the claim to the subtree rooted at each v_{S_i}; let Φ_i be the legal mapping thus found. Note first that the subtrees are disjoint for if this were not the case, then there would exist some node, say v_S of in-degree two or more. But in that case, any pair of parents S', S'' would be incomparable (by definition of an arc in H) and hence $S' - S''$, $S'' - S'$ and $S'' \cap S'$ would all be nonempty, contradicting strong laminarity.

Thus we can combine the mappings Φ_i into an overall mapping Φ such that the contribution to each node is at least 1 and $C(v_{S_i}) \geq 1 + |F(S_i)|/2$ for each $i = 1, 2, \ldots, p$. We thus have

$$|\mathcal{S}| \leq \sum_{v_S \in H} C(v_S) \leq \sum_{a \in A \cap F} 1 + \sum_{a \in A \cap R} \frac{1}{2} = |A| - |\mathcal{R}|.$$

But we know that $|\mathcal{S}| = |A| - |\mathcal{R}|$ and so all inequalities must be tight above. In particular, we must have $C(v_{S_i}) = 1$ for each $i = 1, 2, \ldots, p$. Hence $|F(S_i)| = 0$ for each i. But then we have a nontrivial combination of the zero vector by adding the cut inequalities for the maximal sets S_i and subtracting the orientation constraints corresponding to arcs appearing in some $\delta^+(S_i)$ (again using (4)). This final contradiction completes the proof. $\square$

Proof of Theorem 2 Suppose the statement is false and x^* is a non-integral extreme point for some $P(D, f, \mathcal{F}, \mathcal{E}, u)$. Moreover, choose such a counterexample for which $|A|$ is minimized. Suppose x^*_a is an integer, say k, for some arc a. Let u' be obtained by restricting u to the variables $V - \{a\}$. Also let $\mathcal{E}'$ be obtained from $\mathcal{E}$ by removing any digon of the form $\{a, a^-\}$ if it exists. Then we claim that suppressing a's component yields a smaller counterexample. Indeed, if $k = 0$, then the resulting non-integral vector is an extreme point of $P(D-a, f, \mathcal{F}, \mathcal{E}', u')$. If $k > 0$, then the new vector is an extreme point for $P(D, \Phi^k(f), \mathcal{F}^k(wv), \mathcal{E}', u')$, where $a = (w, v)$. Thus we may assume that every component of x^* is fractional, and in particular it is slack. This, however, contradicts Lemmas 4 and 5. $\square$

Corollary 7 *Let $x^* > 0$ be a an extreme point of a crossing f-connectivty polyedhron $P(D, f, \mathcal{F}, \mathcal{E}, u)$. If x^* has a defining system $\mathcal{S}, \mathcal{R}$ for which $\mathcal{S}$ is strongly laminar, then x^* is integral.*

Proof: We may now apply Theorem 2 to the intersecting problem obtained by restricting f's support to the intersecting family $\mathcal{S}$ and restricting $\mathcal{E}$ to $\mathcal{R}$. $\square$

We note that Theorem 2 can actually be strengthened as was indicated to us (independently) by J. Cheriyan and A. Frank (communicating ideas from T. Kiraly). In particular, the system of inequalities for an instance $(D, f, \mathcal{F}, \mathcal{E}, u)$ (for f intersecting supermodular) indeed yields integral dual solutions as well, i.e., it is "totally dual integral". Cheriyan and Kiraly exhibited this by showing that the corresponding constraint matrix is a network matrix, and hence totally unimodular.

4 Strong Connectivity

We present in this section a combinatorial 4-approximation algorithm for the problem of strong connectivity with orientation constraints. For clarity of presentation, we assume that the orientation constraints hold for every pair of vertices. However, our algorithm extends in a straightforward manner to the general case. In what follows, we assume that the input directed graph is $D = (V, A)$ and the optimal solution is a directed graph $D^* = (V, A^*)$. We use OPT to denote the cost of the optimal directed graph D^*. We say that an arc (u, v) in a directed graph $D = (V, A)$ is *simple* if $(v, u) \notin A$, and we say that it is *non-simple* otherwise. A directed cycle is called *non-trivial* if it is a simple cycle of length at least three.

At the center of our approach is a procedure that takes as input any strongly connected subgraph of D, possibly violating some orientation constraints, and reduces the problem of "amending" its violated orientation constraints to that of finding a minimum cost 2-edge connected subgraph in an undirected graph. For the latter problem, a 2-approximate algorithm [5, 14] is known. We now describe our algorithm in detail:

1. Pick any node r and compute a minimum cost in-branching to r, say T_1, as well as a

minimum cost out-branching from r, say T_2. Consider the directed graph $D_1 = (V, A_1)$ induced by $T_1 \cup T_2$. Clearly, D_1 is strongly connected and its cost is at most $2 \cdot \mathrm{OPT}$. Assume without loss of generality that D_1 is minimal.

2. The set of simple arcs $A' \subseteq A_1$ induces a collection of strongly connected components $C_1, C_2, ..., C_k$ (see Lemma 8). Shrink each component C_i to a single node x_i and construct a directed graph $D_2 = (X, A_2)$, where $X = \{x_1, ..., x_k\}$. An arc $(x_i, x_j) \in A_2$, if and only if D_1 contained some arc (u, v) with $u \in X_i, v \in X_j$. The minimality of D_1 implies that D_2 is minimally strongly connected as well.

3. Replacing each non-simple pair of arcs by an undirected edge, evidently results in a tree, by minimality. Thus D_2 has $k-1$ non-simple arc pairs $(a_1, b_1), ..., (a_{k-1}, b_{k-1})$. It is convenient to view D_2 as an undirected tree $T = (X, E_T)$, such that T contains an edge e_i for each pair (a_i, b_i). Let $(X_i, \bar{X}_i)$ be the partition of node set X induced by removal of a pair (a_i, b_i). We associate with any such partition a cut $(S_i, \bar{S}_i)$ in D_1, where $S_i = \bigcup_{x_j \in X_i} C_j$. We refer to these cuts as the *canonical cuts* of D_1. We say that an arc a *hits* a canonical cut if $a \in \delta^+(S_i) \cup \delta^+(\bar{S}_i)$. For each canonical cut, $A^* \setminus A_1$ contains an arc $a \in \delta^+(S_i) \cup \delta^+(\bar{S}_i)$ (see Lemma 9). Thus, the cost of a minimum cost set of arcs, say $A_3 \subseteq A \setminus A_2$, that hits all canonical cuts of D_1 (call it the *canonical cut cover* of D_1) is no more than OPT. Finding an optimal canonical cut cover is NP-hard, but a 2-approximation can be easily obtained (see Lemma 10). Since the cost of A_3 is at most OPT, the cost of the arcs that we add in this step is at most $2 \cdot \mathrm{OPT}$. Let $D_3 = (V, A_1 \cup A_3)$ be the directed graph obtained by adding arcs in A_3 to the directed graph D_1. The total cost of arcs in D_3 is at most $4 \cdot \mathrm{OPT}$.

4. The final step is to show that the directed graph D_3 above can be modified into another directed graph $D_4 = (V, A_4)$ such that (i) D_4 is strongly connected, (ii) $A_4 \subseteq (A_2 \cup A_3)$, and (iii) all arcs in D_4 are simple (see Lemma 11). The costs of steps (1) and (3) are no more than $2 \cdot \mathrm{OPT}$ each. Therefore we now have a 4-approximation to our problem.

Lemma 8 *In any minimally strongly connected directed graph H, the set of simple arcs induces a collection of strongly connected components.*

Proof: It suffices to show that every simple arc lies on a directed cycle that consists only of simple arcs. Consider any simple arc (u, v) and a path $P(v, u)$ from v to u in H. By the minimality of H, every arc on $P(v, u)$ must be simple. The lemma follows. □

Lemma 9 *$A^* \setminus A_1$ hits every canonical cut of D_1.*

Proof: Suppose not. Then there is a cut $(S_i, \bar{S}_i)$ such that A^* has at most one arc (from the pair $\{a_i, b_i\}$) that crosses the cut. This contradicts that D^* is strongly connected. □

Lemma 10 *There is a 2-approximation algorithm for the minimum cost canonical cut cover problem.*

Proof: We solve this problem by a reduction to the minimum cost 2-edge connected subgraph problem. Let B be the set of arcs $A \setminus A_2$. Consider the undirected graph $H = (X, E)$

obtained as follows. There is an edge $x_i x_j \in E$ if and only if there is an arc in B that connects some node in X_i to X_j or vice versa. Moreover, the cost of this edge is equal to the minimum cost such arc. Also, for each edge in E_T, we include an edge of cost 0 in H.

We now claim that the problem of finding a minimum cost canonical cut cover of D_1 is equivalent to finding a minimum cost 2-edge connected subgraph $H' = (X, E')$ of H. To see this, consider any canonical cut cover $\hat{A}$. Then the set of undirected edges in H, obtained from the arcs in $\hat{A}$ along with the edges in E_0, has the property that every cut in H has at least two edges crossing it. Moreover, the cost of this collection of edges is no more than the cost of the canonical cut cover $\hat{A}$. In the other direction, let us consider a 2-edge connected subgraph $\hat{E}$ of H. Since exactly one edge in E_0 crosses a cut in H, there must be at least one additional edge in $\hat{E}$ crossing any cut. Thus, the directed arcs corresponding to the edges in $\hat{E}$ form a canonical cut cover of D_1.

A 2-approximation algorithm is known for the minimum cost 2-edge connected subgraph problem[14]. It then suffices to use this algorithm to get a 2-approximation algorithm for the minimum cost canonical cut cover problem. □

Lemma 11 *The directed graph D_3 obtained at the end of Step (3) of the algorithm, can be modified into a directed graph D_4 such that D_4 is strongly connected, contained in D_3, and does not contain any non-simple arcs.*

Proof: The only non-simple arcs in D_3 are the ones corresponding to the pairs (a_i, b_i). We will now give a procedure to remove from D_3 exactly one arc from each such pair and still keeping it strongly connected. Consider a leaf x_i in T. There must be an arc in A_3 of the form (x_i, x_j) or (x_j, x_i) that hits the canonical cut $(x_i, X \setminus \{x_i\})$. Assume without loss of generality that it is of the form (x_i, x_j). Consider the directed path from x_j to x_i in D_2, consisting only of non-simple arcs, and remove every non-simple arc that is oriented in a direction opposite to this path. Contract the resulting cycle and repeat this procedure on a leaf of the resulting tree T'. It is easy to verify that the procedure continues until the resulting tree reduces to a single node, corresponding to the graph D_4 above. □

Boesch and Tindell [2] have given necessary and sufficient conditions for the existence of a strong orientation of a mixed graph. We remark that the above proof essentially shows that these conditions are met in our case.

5 Crossing Supermodularity

We now focus our attention on general crossing supermodular functions. While simple constant factor approximation algorithms are known in the absence of orientation constraints [17], the problem seems to become much harder in the presence of orientation constraints. Although we do not resolve this question here, we make some progress towards solving our original problem (I) for crossing supermodular functions. We will establish the following result.

Theorem 12 *Let* OPT *denote the optimal cost of a fractional solution to the problem (I) with some crossing supermodular function f and let $p = \max_S f(S)$ denote the maximum* *requirement of any set. Then we can find an integral solution of cost* $2p \cdot \mathrm{OPT}$ *that satisfies the slightly weaker requirement function* f' *defined as* $f'(S) = \max\{f(S) - 1, 0\}$.

An immediate corollary of the above theorem is that we can find a solution to a $(k-1)$-strong connectivity problem with orientation constraints at a cost that is bounded by $2k$ times the optimal cost for k-strong connectivity. We devote the rest of this section to the proof of Theorem 12.

Let D be our digraph and f be a crossing supermodular function on the family $\mathcal{F}$. Let $p = \max_S f(S) - 1$. Define $f'(S) = \lfloor \frac{f(S)}{1+1/p} \rfloor$ for each set S. It is easy to verify that $f'(S) = f(S) - 1$ if $f(S) > 0$, and that f' is also crossing supermodular on $\mathcal{F}$.

Let OPT denote the cost of a feasible solution x^* to the problem (I).

1. Define u_a to be $\lceil x^*(a)/(1/p) \rceil$. Clearly, $u_a + u_{a^-} \leq p+1$. Also, define $f_p(S) = p \cdot f(S)$ to be a new crossing supermodular function.

2. We now solve two seperate intersecting supermodular LPs with upper bounds just defined. These LP's are obtained by splitting the 'requiring' sets into $\mathcal{F}^1 = \{F \in \mathcal{F} : v \in F\}$ and $\mathcal{F}^2 = \mathcal{F} - \mathcal{F}^1$ where v is an arbitrarily chosen vertex. The first we may solve directly; the latter is not actually intersecting and so we work with the intersecting family $\{V - S : S \in \mathcal{F}^2\}$. For the second family we must also work with the function f' defined by $f'(S) = f(V - S)$ and use the digraph with the arcs reversed (see [17]). By Theorem 2, any basic solution for these LP's is integral. We may thus find two such vectors z^1, z^2 in polynomial time. Actually, since these LP's do not have orientation constraints, we can also find integral optimal solutions using Frank's approach ([6], see also [19]).

   We note here that since these LP's do not have orientation constraints, we can also find integral optimal solutions using Frank's approach [6]. Now define a new vector z by setting $z_a = \max\{z_a^1, z_a^2\}$ for each $a \in A$. Clearly z is integral, satisfies $p \cdot f$, and costs no more than $2p \cdot \mathrm{OPT}$.

3. By setting $y = \frac{1}{p} z$ we obtain a solution for the original function f which is $(1/p)$-integral and has cost at most 2OPT. The solution y violates any orientation constraint by at most a factor of $1 + 1/p$. We scale down all violating arc pairs to satisfy the constraint $x_a + x_{a^-} = 1$. We also uniformly scale up any arc pairs with $1/p \leq x_a + x_{a^-} < 1$ to satisfy $x_a + x_{a^-} = 1$. The resulting solution clearly satisfies the function f' and has a cost that is at most $2p \cdot \mathrm{OPT}$.

4. At this point, since all orientation constraints are tight, we may use the results of Appendix A to derive an integral solution of no greater cost.

Acknowledgements

We are grateful to Joseph Cheriyan and András Frank who independently described shorter and more general proofs for Theorem 2.

References

[1] A. Agrawal, P. Klein, and R. Ravi, An approximation algorithm for generalized Steiner tree problems in networks, *Proceedings of the 23rd ACM Symposium on Theory of Computing*, (1991) 134-144.

[2] F. Boesch and R. Tindell, Robbins's theorem for mixed multigraphs, *American Mathematical Monthly*, **87** (1980), pp. 716–719.

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

[4] K.P. Eswaran, R.E. Tarjan, Augmentation problems, *SIAM Journal on Computing,* **5** (1976), pp. 653–665.

[5] G. N. Frederickson and J. JáJá, Approximation algorithms for several graph augmentation problems, *SIAM Journal on Computing*, **10**(2) (1981), pp. 270–283.

[6] A. Frank, Kernel systems of directed graphs, *Acta Sci. Math (Szeged),* **41** (1979), pp. 63–76.

[7] A. Frank, How to make a digraph strongly connected, *Combinatorica,* **1**(2) (1981), pp. 145–153.

[8] A. Frank, Augmenting graphs to meet edge-connectivity requirements, *SIAM Journal on Discrete Math.,* **5**(1) (1992), pp. 25–53.

[9] A. Frank, Applications of submodular functions, *Surveys in Combinatorics, London Mathematical society Lecture Notes Series 187*, Cambridge University Press (Ed. K. Walker) (1993), pp. 85–136.

[10] M. Goemans and D. Williamson, A general approximation technique for constrained forest problems, *SIAM Journal on Computing,* **24** (1995), pp. 296–317.

[11] M. Goemans, A. Goldberg, S. Plotkin, D. Shmoys, É. Tardos, and D. Williamson, Approximation algorithms for network design problems, *Proceedings of the 5th Annual ACM-SIAM Symposium on Discrete Algorithms*, (1994), pp. 223–232.

[12] M. Grötschel, L. Lovász, and A. Schrijver, The ellipsoid method and its consequences in Combinatorial Optimization, *Combinatorica* **1** (1981), 169-197.

[13] K. Jain, A factor 2 approximation algorithm for the generalized steiner network problem, *Proceedings of the 39th Annual Symposium on the Foundations of Computer Science* (1998), pp. 448–457.

[14] S. Khuller and U. Vishkin, Biconnectivity approximations and graph carvings, *J. Assoc. Comput. Mach.,* **41** (1994), pp. 214–235.

[15] L. Lovász, On two min-max theorems in graph theory, *JCT Series B,* **21** (1976) 96-103.

[16] C.L. Lucchesi, D.H. Younger, A minimax relation for directed graphs, *J. London Math. Soc.,* **17**(2) (1978), pp. 369–374.

[17] V. Melkonian and É. Tardos, Approximation algorithms for a directed network design problem, *IPCO '99. Proceedings of the 7th International Integer Programming and Combinatorial Optimization Conference (IPCO)* (1999).

[18] A. Schrijver, Packing and covering crossing families of cuts, *JCT Series B*, **35** (1983), pp. 104–128.

[19] A. Schrijver, Total dual integrality from directed graphs, crossing families and sub- and supermodular functions, in: W.R. Pulleyblank, ed., *Progress in Combinatorial Optimization*, Academic Press, Toronto, 1984, 315-362.

[20] D. Williamson, M. Goemans, M. Mihail, and V. Vazirani, A Primal-dual approximation algorithm for generalized steiner network problems, *Combinatorica*, **15** (1995), pp. 435–454.

[21] D. Younger, Private Communication, (1985).

A Algorithms for Minimum Cost Orientations

In this appendix we explain how to optimally solve problem (I) in the case where all orientation constraints are *equalities* and f is a crossing supermodular function.

One way to solve this problem is as follows (see [9]). First, choose the cheaper arc from each pair of arcs appearing in an orientation constraint, yielding a directed graph denoted by $D_1 = (V, A_1)$. Then, find a minimum cost collection of arcs $(u, v) \in A_1$, such that if they are *flipped*, i.e., (u, v) is replaced by (v, u), then a directed graph satisfying the requirement function f is obtained. Call such a set of arcs an *f-flip set.* When $f \equiv 1$, we refer to this simply as a flip set.

Mininum cost f-flip sets can be optimally computed by using the work of Edmonds and Giles [3] on submodular flows. It is easily seen that a set A_2 is a f-flip set if and only if its incidence vector x satisfies for every proper subset S:

$$|\delta^+(S)| + x(\delta^-(S)) - x(\delta^+(S)) \geq f(S)$$

Define a new function g where

$$g(S) = -f(S) + |\delta^+(S)|.$$

Since f is crossing supermodular, we get that g is submodular. Therefore, finding a minimum cost f-flip set is equivalent to solving the submodular flow problem $\min\{cx : x \geq 0,\ x(\delta^+(S)) - x(\delta^-(S)) \leq g(S)$ for each proper subset $S\}$. In [3], such linear programs are shown to have integral optima which means that minimum cost f-flip sets can be computed in polynomial time.

There is an alternative approach in the case of strong connectivity. Namely, finding minimum cost flip sets can be reduced to an algorithmic form of the Lucchesi-Younger theorem as we will shortly describe. Define a *directed cut*, or *di-cut*, in a directed graph D to be a nonempty set of arcs of the form $\delta^+(S)$ for which $\delta^-(S) = \emptyset$. A *di-cut cover* is a set of arcs which intersects every di-cut.

Theorem 13 (Lucchesi-Younger [16]) *For any directed graph D, the maximum packing of arc-disjoint di-cuts is equal to the cardinality of a minimum di-cut cover.*

This theorem immediately applies to weighted di-cut covers by taking parallel copies of arcs. The bipartite version of this theorem had been studied earlier by McWhirter and Younger. A simple proof of the general theorem appeared in Lovász [15] and an algorithmic proof (using distinct techniques) can be found in Frank [7]. Apart from Edmonds and Giles [3], a further generalization appeared in Schrijver [18].

Evidently, if D_1 has a cut containing a single arc, then it has no flip set. On the other hand, if the underlying undirected graph of D_1 is 2-edge connected, then it has a flip set by Whitney's ear decomposition method. We note that the minimum flip set problem is similar to but distinct from the augmentation problem studied by Eswaran and Tarjan [4], as well as Frank [8], where one may include an augment arc (v, u) only if the arc (u, v) appeared in A_1. This latter problem is transparently equivalent to finding a minimum cardinality set of arcs whose contraction results in a strongly connected digraph, which in turn is equivalent to finding a minimum directed cut cover. Clearly, any flip set must be a di-cut cover, implying that the minimum size of a flip set is at least the size of a minimum di-cut cover. In fact we have the following.

Theorem 14 (Younger [21]) *Let $D = (V, A)$ be a directed graph where all cuts contain at least two arcs. The minimum cost of a di-cut cover is equal to the minimum cost of a flip set.*

This follows immediatley from the following unpublished result due to Younger [21].

Proposition 15 *If $A' \subseteq A$ is a minimal di-cut cover, then A' is a flip set for D.*

Proof: Suppose this is not the case and let $A' \subseteq A$ be a minimal directed cut cover which is not a flip set. Then there must exist a nonempty proper subset $S \subseteq V$ such that (1) $\emptyset \neq \delta^-(S) \subseteq A'$ and (2) $\delta^+(S) \cap A' = \emptyset$. Let $\delta^-(S) = \{a_1, \ldots a_k\}$ and choose S so that k is minimized ($k = 0$ is impossible as A' is a cut cover). For each i, let $S_i \subseteq V$ define a *justifying cut* for a_i, i.e., $\delta^+(S_i) \cap A' = \{a_i\}$. Now for some i, suppose $S_i \cap S \neq \emptyset$. Any arc in $\delta^+(S_i \cap S)$ is either in $\delta^+(S)$ or in $\delta^+(S_i) \setminus \{a_i\}$ and so $\delta^+(S_i \cap S) \cap A' = \emptyset$. Moreover, $\delta^-(S_i \cap S) \subseteq (\delta^-(S) - \{a_i\})$ which contradicts minimality. Thus $S_i \subseteq V - S$ for each i.

We claim that the S_i's are also disjoint. For suppose that $S_i \cap S_j \neq \emptyset$. First note that $\delta^+(S_j \cap S_i)$ cannot contain a_i or a_j, for then one of the two cuts includes $\{a_i, a_j\}$ contradicting justification. Hence, we may assume that $S_i - S_j$ and $S_j - S_i$ are not empty. But then $\delta^+(S_i \cap S_j)$ is a directed cut which contains no arc of A', a contradiction.

For any i, consider an arc $a'_i \neq a_i$ in $\delta^+(S_i)$; this arc exists by the absence of cut edges. Moreover, by disjointness of the S_i's, the head of a'_i must lie outside $T = S \cup S_1 \cup S_2 \ldots \cup S_k$ and so in particular we have $T \neq V$. Now any arc in $\delta^+(T)$ (respectively $\delta^-(T)$) must lie in

$\delta^+(S)$ or some $\delta^+(S_i) - \{a_i\}$ (respectively, in some $\delta^-(S_i)$). Thus $\delta^-(T) = \emptyset$ and, using (2), $\delta^+(T) \cap A' = \emptyset$. Thus $\delta^+(T)$ is a directed cut which is not covered by A', a contradiction. $\square$

This theorem extends to an alternative algorithm for finding minimum cost flip sets by way of an algorithm of Frank [7] for finding minimum cost di-cut covers.

We remark, that the second handling above amounts to dropping some of the constraints in our original subdmodular flow linear program. Namely, (in the case where $f \equiv 1$), the content of the lemma above is that we only require the submodular flow cut constraints on sets S, for which $g(S) > 0$. It would be interesting to know if this approach works for crossing supermodular functions in general. This would require a modification of the Lucchesi-Younger Theorem, distinct from submodular flows, where one is required to pack cuts for which $g(S) > 0$.