Available online at www.sciencedirect.com
ScienceDirect
Electronic Notes in Discrete Mathematics 30 (2008) 303–308
Electronic Notes in DISCRETE MATHEMATICS
www.elsevier.com/locate/endm

Combinatorial flexibility problems and their computational complexity [1]

Néstor E. Aguilera

CONICET and UNL. IMAL/INTEC, Güemes 3450, 3000 Santa Fe, Argentina.

Valeria A. Leoni [2], Graciela L. Nasini

CONICET and UNR. Depto de Matemática, Fac. de Cs. Exactas, Ingeniería y Agrimensura, Av. Pellegrini 250, 2000 Rosario, Argentina.

Abstract

The concept of flexibility—originated in the context of heat exchanger networks—is associated with a substructure which guarantees the *performance* of the original structure, in a given range of possible states. We extend this concept to combinatorial optimization problems, and prove several computational complexity results in this new framework.

Under some monotonicity conditions, we prove that a combinatorial optimization problem polynomially transforms to its associated flexibility problem, but that the converse need not be true.

In order to obtain polynomial flexibility problems, we have to restrict ourselves to combinatorial optimization problems on matroids. We also prove that, when relaxing in different ways the matroid structure, the flexibility problems become *NP*-complete. This fact is shown by proving the *NP*-completeness of the flexibility problems associated with the Shortest Path, Minimum Cut and Weighted Matching problems.

Keywords: combinatorial problems, flexibility, computational complexity

doi:10.1016/j.endm.2008.01.052

1 Flexibility and combinatorial optimization problems

The concept of flexibility arose from chemical engineering problems in the design of heat exchanger networks (see, for example [2]). This concept is associated with a substructure which guarantees the *performance* of the original structure in a given range of possible states. In the context of heat exchanger networks, this performance is defined by the value of a Maximum Flow-Minimum Cut. This problem motivated us to extend the flexibility concept to general combinatorial optimization problems.

Following [5], in a combinatorial optimization problem we deal with a finite set A, a vector of costs $\mathbf{c} \in \mathbb{R}^A$ and a family $\mathcal{F}(A)$ of subsets of A. The cost of $F \subseteq A$ will be indicated by $\mathbf{c}(F) = \sum_{a \in F} c_a$, and the optimal value by

$$\xi_A^{\mathcal{F}}(\mathbf{c}) = \min(\max)\left\{\mathbf{c}(F) : F \in \mathcal{F}(A)\right\}.$$

If $\mathcal{F}(A) = \emptyset$, we set $\xi_A^{\mathcal{F}}(\mathbf{c}) = +\infty$ and $\xi_A^{\mathcal{F}}(\mathbf{c}) = -\infty$, respectively. We say that $F \in \mathcal{F}(A)$ is $\mathbf{c}$-*optimal* if $\mathbf{c}(F) = \xi_A^{\mathcal{F}}(\mathbf{c})$.

We will identify a particular combinatorial optimization problem by the oracle algorithm $\mathcal{F}$ which decides, in constant time, whether a given subset of A belongs to the family $\mathcal{F}(A)$.

Working on flexibility problems, we consider a family of instances determined by a given set A and vectors $\mathbf{c}^-, \mathbf{c}^+ \in \mathbb{Z}_+^A$, defining the *state set* $S = \left\{\mathbf{c} \in \mathbb{R}^A : \mathbf{c}^- \leq \mathbf{c} \leq \mathbf{c}^+\right\}$. We also consider a *substructure* given by $B \subseteq A$. For simplicity we will indicate by $\xi_B^{\mathcal{F}}(\mathbf{c})$, the optimal value corresponding to the restriction of $\mathbf{c}$ to $\mathbb{R}^B$.

Given a state set S and $W \subseteq S$, we will say that B is $\mathcal{F}$-*flexible* in W if $\xi_B^{\mathcal{F}}(\mathbf{c}) = \xi_A^{\mathcal{F}}(\mathbf{c})$, for all $\mathbf{c} \in W$. When $W = S$, we just say that B is $\mathcal{F}$-*flexible.*

The $\mathcal{F}$-*flexibility problem* ($\mathcal{F}$-flex) is formulated as follows:

INSTANCE: A finite set A; $B \subseteq A$; $\mathbf{c}^+, \mathbf{c}^- \in \mathbb{Z}_+^A$.
QUESTION: Is there $\mathbf{c}$ with $\mathbf{c}^- \leq \mathbf{c} \leq \mathbf{c}^+$ and $\xi_A^{\mathcal{F}}(\mathbf{c}) \neq \xi_B^{\mathcal{F}}(\mathbf{c})$?

Notice that $\mathcal{F}$-flex consists of answering whether B is not $\mathcal{F}$-flexible.

From now on, we restrict ourselves to combinatorial optimization problems verifying some kind of monotonicity under inclusion on the optimal value. We say that $\mathcal{F}$ is *monotone increasing (decreasing)* if for all $\mathbf{c} \in \mathbb{R}^A$ and $B \subseteq A$ we have $\xi_B^{\mathcal{F}}(\mathbf{c}) \leq \xi_A^{\mathcal{F}}(\mathbf{c})$ ($\xi_B^{\mathcal{F}}(\mathbf{c}) \geq \xi_A^{\mathcal{F}}(\mathbf{c})$).

In fact, most of the combinatorial optimization problems of interest are monotone. In particular, between the problems considered here, the *Weighted*

[1] Partially supported by grants of CONICET-PIP 5810 and ANPCyT-PICT2005 38036
[2] Email: valeoni@fceia.unr.edu.ar

Matching problem and the *Maximum Weight Forest problem* are monotone increasing maximization problems, whereas the *Shortest Path problem* is a monotone increasing minimization problem and the *Minimum Cut problem* is monotone decreasing.

The monotonicity imposed on the optimal value leads us to prove that solving $\mathcal{F}$-flex is equivalent to asking whether a given element $a \in A$ is *useful* for S, following the terminology introduced in [1] in the context of the Maximum Flow problem. Formally, we prove:

Lemma 1.1 *$\mathcal{F}$-flex may be reduced to the family of instances given by $B = A \setminus \{a\}$, for $a \in A$.*

The key of the proof is the fact that the answer corresponding to an instance of $\mathcal{F}$-flex given by $(A, B, \mathbf{c}^+, \mathbf{c}^-)$ is YES if and only if, for any B' with $B \subseteq B' \subseteq A$, at least one of the instances $(A, B', \mathbf{c}^+, \mathbf{c}^-)$ and $(B', B, \mathbf{c}^+, \mathbf{c}^-)$ has also an affirmative answer.

In the next section we show that solving $\mathcal{F}$-flex is always at least as hard as solving $\mathcal{F}$, and try to find families of optimization problems with polynomial complexity associated flexibility problems.

The proofs of all computational complexity results are strongly based on the fact that, when checking flexibility, it is enough to do so on a *finite* subset of states, called *test set*. In particular, for $F \subseteq A$ we define the *F-state* $\mathbf{c}^F \in \mathbb{Z}_+^A$ by:

- if $\mathcal{F}$ is a minimization problem, $c_a^F = c_a^-$ if $a \in F$ and $c_a^F = c_a^+$ if $a \notin F$;
- if $\mathcal{F}$ is a maximization problem, $c_a^F = c_a^+$ if $a \in F$ and $c_a^F = c_a^-$ if $a \notin F$.

We prove that, for a given $F \in \mathcal{F}(A)$, the F-state is that state for which F has the greatest possibility of being optimal in the sense that, if F is a $\mathbf{c}$-optimal element for some $\mathbf{c} \in S$, then F is $\mathbf{c}^F$-optimal.

Moreover:

Lemma 1.2 *Let A, $B \subseteq A$, $\mathbf{c}^-$ and $\mathbf{c}^+$ defining an instance of $\mathcal{F}$-flex. If $\mathcal{F}$ is a monotone increasing minimization problem or a monotone decreasing maximization problem, then $\{\mathbf{c}^F : F \in \mathcal{F}(A)\}$ is a test set. If $\mathcal{F}$ is a monotone increasing maximization problem or a monotone decreasing minimization problem, then $\{\mathbf{c}^F : F \in \mathcal{F}(B)\}$ is a test set.*

Let us point out that, for a general combinatorial optimization problem $\mathcal{F}$, the cardinalities of the test sets given by Lemma 1.2 are non polynomial in the size of A.

2 Looking for polynomial flexibility problems

In order to compare the computational complexities of $\mathcal{F}$ and $\mathcal{F}$-flex, we need to take into account the relationship between $\mathcal{F}(B)$ and $\mathcal{F}(A)$, when $B \subseteq A$. Let us impose the following monotonicity conditions on the set of feasible solutions

(P1) for every $E \in \mathcal{F}(B)$, there exists $F \in \mathcal{F}(A)$ such that $E \subseteq F$,

(P2) if $E \in \mathcal{F}(A)$ and $E \subseteq B$ then $E \in \mathcal{F}(B)$.

Under these conditions we can prove:

Theorem 2.1 *$\mathcal{F}$ may be polynomially reduced to its corresponding flexibility problem, $\mathcal{F}$-flex.*

Let us observe that, for the Weighted Matching, the Maximum Weight Forest and the Shortest Path problems, it holds that

(P3) $\mathcal{F}(B) = \{F \in \mathcal{F}(A) : F \subseteq B\}$,

whereas, for the Minimum Cut problem it holds that

(P4) $\mathcal{F}(B) = \{F \cap B : F \in \mathcal{F}(A)\}$.

Each of the properties (P3) and (P4) imply (P1) and (P2). Conditions (P1) and (P2) are the weakest we may impose for proving Theorem 2.1. In fact, when finding the optimal value of $\mathcal{F}$ we just use (P1), and then (P2) is necessary in order to find an optimal element.

From Theorem 2.1, if we want to find polynomial flexibility problems, we should reduce our search to the family of polynomial optimization problems.

An *exchanger network* is a digraph $D = (V_1 \cup V_2, E)$ with $V_1 \cap V_2 = \emptyset$, $E \subseteq V_1 \times V_2$ and a vector $\mathbf{c} \in \mathbb{R}^{V_1 \cup V_2}$ (for $i \in V_1$, c_i is the *supply* of i and for $j \in V_2$, c_j is the *demand* of j). The maximum *exchange* in this class of networks can be modeled as a Maximum Flow-Minimum Cut problem in a certain st-network.

In [1] and [4] we may find two independent proofs of the NP-completeness of the Minimum Cut flexibility problem (FF), even on instances with $\mathbf{c}^- = \mathbf{0}$. However, considering those instances of FF corresponding to exchanger networks (FT), the problem becomes polynomial when $\mathbf{c}^- = \mathbf{0}$.

Nevertheless,

Theorem 2.2 *FT is NP-complete.*

The proof is based on the reduction of the Balanced Complete Bipartite Graph problem ($BCBG$). $BCBG$ consists in deciding whether there is a complete bipartite balanced subgraph of certain size in a given bipartite graph. The proof of its *NP*-completeness may be found for example in [3, p. 196].

We wonder if tightening conditions (P1) and (P2)—imposing, for example (P3)—we may establish the converse of Theorem 2.1. However, we prove:

Theorem 2.3 *The Shortest Path flexibility problem (FP) is NP-complete.*

In this case, we reduce *DVDP2* to the Shortest Path problem. Given a digraph G and nodes s, r, t, w of G, *DVDP2* consists in deciding whether there exist vertex disjoint st- and rw-paths.

A combinatorial optimization problem $\mathcal{F}$ is *hereditary* if for all A, $F \in \mathcal{F}(A)$ and $F' \subseteq F$ it holds that $F' \in \mathcal{F}(A)$. Since we deal with non negative states, hereditary combinatorial optimization problems become relevant when they are maximization problems.

Once again, instances with $\mathbf{c}^- = \mathbf{0}$ lead us to guess that hereditary optimization problems could have "easy" associated flexibility problems.

Lemma 2.4 *Let $\mathcal{F}$ be a hereditary problem satisfying condition (P2), $\mathbf{c}^- = \mathbf{0}$ and $\mathbf{c}^+$ with $c_a^+ > 0$ for some $a \in A$. Then, deciding if a is useful can be done in constant time.*

The key of the proof is to show that $a \in A$ is useful if and only if $\{a\} \subseteq \mathcal{F}(A)$.

We now consider the Weighted Matching problem, which satisfies (P3) and is also *hereditary*:

Theorem 2.5 *The Weighted Matching flexibility problem (FBM) is NP-complete, even on instances corresponding to bipartite graphs.*

The proof is based on the reduction of *FP* to *FBM* by using some known transformation between shortest paths and maximum weighted matchings in bipartite graphs.

Finally, hereditary optimization problems defined on matroids seem to be the "best candidates" when looking for polynomial flexibility problems, because of the characterization of matroids through greedy algorithms. In this case, this assertion can be confirmed by the following result:

Theorem 2.6 *The Maximum Independent Set (in a matroid) flexibility problem is polynomial.*

For the proof we use some previous results which allow us to show that, for an instance given by a matroid A, $\{\mathbf{c}^{\{a\}} : a \in A\}$ is a test set.

Finally, let us observe that the family of matchings in a bipartite graph is the intersection of two matroids. Hence, Theorem 2.5 implies that:

Theorem 2.7 *The Two Matroid Intersection flexibility problem is NP-complete.*

This last result leads us to guess that a "matroid structure" is the "weakest" from which we can obtain polynomial flexibility problems.

References

[1] Biedl, T. C., B. Brejová and T. Viñar, *Simplifying Flow Networks. M. Nielsen and B. Rovan (Eds.)*, MFCS 2000, LNCS 1893. (2000), 192–201.

[2] Cerdá, J., M. R. Galli, N. Camussi and M. A. Isla, *Synthesis of Flexible Heat Exchanger Networks-I. Convex Networks*, Comp. Chem. Engng. **14** (1990), 197–211.

[3] Garey, M. R. and D. Johnson, "Computers and Intractability: A Guide to the Theory of *NP*-Completeness", W. H. Freeman & Co., 1979.

[4] Nasini, G. L., "Sobre la Flexibilidad en Problemas de Optimización", Tesis Doctoral, Universidad Nacional de Rosario, Argentina, 1997.

[5] Nemhauser G. and L. Wolsey, "Integer and Combinatorial Optimization", Wiley, 1988.