Chapter 45

Submodular function minimization

This chapter describes a strongly polynomial-time algorithm to find the minimum value of a submodular function. It suffices that the submodular function is given by a value giving oracle.
One application of submodular function minimization is optimizing over the intersection of two polymatroids. This will be discussed in Chapter 47.

45.1. Submodular function minimization

It was shown by Grötschel, Lovász, and Schrijver [1981] that the minimum value of a rational-valued submodular set function f on S can be found in polynomial time, if f is given by a value giving oracle and an upper bound B is given on the numerators and denominators of the values of f. The running time is bounded by a polynomial in $|S|$ and $\log B$. This algorithm is based on the ellipsoid method: we can assume that $f(\emptyset) = 0$ (by resetting $f(U) := f(U) - f(\emptyset)$ for all $U \subseteq S$); then with the greedy algorithm, we can optimize over EP_f in polynomial time (Corollary 44.3b), hence the separation problem for EP_f is solvable in polynomial time, hence also the separation problem for

(45.1) $\qquad P := EP_f \cap \{x \mid x \leq \mathbf{0}\},$

and therefore also the optimization problem for P. Now the maximum value of $x(S)$ over P is equal to the minimum value of f (by (44.8), (44.9), and (44.34)).

Having a polynomial-time method to find the minimum value of a submodular function, we can turn it into a polynomial-time method to find a subset T of S minimizing $f(T)$: For each $s \in S$, we can determine if the minimum value of f over all subsets of S is equal to the minimum value of f over subsets of $S \setminus \{s\}$. If so, we reset $S := S \setminus \{s\}$. Doing this for all elements of S, we are left with a set T minimizing f over all subsets of (the original) S.

Grötschel, Lovász, and Schrijver [1988] showed that this algorithm can be turned into a strongly polynomial-time method. Cunningham [1985b] gave a