CS 598CSC: Combinatorial Optimization	Lecture date: March 9, 2009
Instructor: Chandra Chekuri	Scribe: Sreeram Kannan

1 Maximum Weight Independent Set in a Matroid, Greedy Algorithm, Independence and Base Polytopes

1.1 More on Matroids

We saw the definition of base, circuit, rank, span and flat of matroids last lecture. We begin this lecture by studying some more basic properties of a matroid.

Exercise 1 *Show that a set $I \subseteq S$ is independent in a matroid $\mathcal{M}$ iff $\forall y \in I$, there exists a flat F such that $I - y \subseteq F$ and $y \notin F$.*

Definition 2 *A matroid $\mathcal{M} = (S, \mathcal{I})$ is defined as connected if $r_{\mathcal{M}}(U) + r_{\mathcal{M}}(S \backslash U) > r_{\mathcal{M}}(S)$ for each $U \subseteq S, U \neq \emptyset$. Equivalently, for each $s, t \in S, s \neq t$, there is a circuit containing both s, t.*

1.2 Operations on a Matroid

1.2.1 Dual

Given a matroid $\mathcal{M} = (S, \mathcal{I})$ its dual matroid $\mathcal{M}^* = (S, \mathcal{I}^*)$ is defined as follows:

$$\mathcal{I}^* = \{I \in S \mid S \backslash I \text{ is spanning in } \mathcal{M}, \text{ i.e., } r_{\mathcal{M}}(S \backslash I) = r_{\mathcal{M}}(S)\}.$$

Exercise 3 *Verify that $(S, \mathcal{I}^*)$ is indeed a matroid.*

The following facts are easy to prove:

1. $\mathcal{M}^{**} = \mathcal{M}$.
2. B is a base of $\mathcal{M}^*$ iff $S \backslash B$ is a base of $\mathcal{M}$.
3. $r_{\mathcal{M}^*}(U) = |U| + r_{\mathcal{M}}(S \backslash U) - r_{\mathcal{M}}(S)$

Remark 4 *Having an independence or rank oracle for $\mathcal{M}$ implies one has it for $\mathcal{M}^*$ too.*

Exercise 5 *Prove that $\mathcal{M}$ is connected if and only if $\mathcal{M}^*$ is.*

Hint: Use the third property from above.

1.2.2 Deletion

Definition 6 *Given a matroid $\mathcal{M} = (S, \mathcal{I})$ and $e \in S$, deleting e from $\mathcal{M}$ generates a new matroid*

$$\mathcal{M}' = \mathcal{M} \backslash e = (S - e, \mathcal{I}'), \tag{1}$$

where $\mathcal{I}' = \{I - e | I \in \mathcal{I}\}$. For $Z \subseteq S$, the matroid $\mathcal{M} \backslash Z$ is obtained similarly by restricting the matroid $\mathcal{M}$ to $S \backslash Z$.

1.2.3 Contraction

Definition 7 *Given a matroid $\mathcal{M} = (S, \mathcal{I})$ and $e \in S$, the contraction of the matroid with respect to e is defined as $\mathcal{M}/e = (\mathcal{M}^* \backslash e)^*$, i.e., it is obtained by deleting e in the dual and taking its dual. Similarly for a set $Z \subseteq S$, we can similarly define $\mathcal{M}/Z = (\mathcal{M}^* \backslash Z)^*$*

It is instructive to view the contraction operation from a graph theoretic perspective. If e is a loop, $\mathcal{M}/e = \mathcal{M} \backslash e$, else $\mathcal{M}/e = (S - e, \mathcal{I}')$ where

$$\mathcal{I}' = \{I \in S - e | I + e \in \mathcal{I}\}.$$

For the case of contracting a subset Z, we can take a base $X \subseteq Z$ and $M/Z = (S \backslash Z, \mathcal{I}')$, where

$$\mathcal{I}' = \{I \in S \backslash Z | I \cup Z \in \mathcal{I}\}.$$

Also

$$r_{\mathcal{M}/Z}(X) = r_{\mathcal{M}}(X \cup Z) - r_{\mathcal{M}}(Z)$$

Exercise 8 *Show that*

$$\mathcal{M}/\{e_1, e_2\} = (\mathcal{M}/e_1)/e_2 = (\mathcal{M}/e_2)/e_1. \tag{2}$$

1.2.4 Minor

Similar to graph minors, matroid minors can be defined, and they play an important role in characterizing the type of matroids.

Definition 9 *A matroid $\mathcal{M}'$ is a minor of a matroid $\mathcal{M}$ if $\mathcal{M}'$ is obtained from $\mathcal{M}$ by a sequence of contractions and deletions.*

1.3 Maximum Weight Independent Set in a Matroid

Matroids have some important algorithmic properties, the simplest one being that the problem of determining the maximum weight independent set in a matroid can be solved using a greedy algorithm. The maximum weight independent set problem is stated as follows: Given $\mathcal{M} = (S, \mathcal{I})$ and $w : S \to R$, output

$$\arg\max_{I \in \mathcal{I}} w(I). \tag{3}$$

1.3.1 Greedy Algorithm

The greedy algorithm can be stated as follows:

1. Discard all $e \in S$ where $w(e) \leq 0$ or e is a loop.
2. Let $S = \{e_1, ..., e_n\}$ such that $w(e_1) \geq w(e_2) ... \geq w(e_n)$.
3. $X \leftarrow \emptyset$.
4. For $i = 1$ to n, do
   if($X + e_i \in \mathcal{I}$) then $X \leftarrow X + e_i$.

5. Output X.

The above algorithm had to specifically take care of loops and edges with non-negative weights. An equivalent algorithm without this hassle is:

1. Let $S = \{e_1, ..., e_n\}$ such that $w(e_1) \geq w(e_2) ... \geq w(e_n)$.
2. $X \leftarrow \emptyset$.
3. For $i = 1$ to n, do
   if$(X + e_i \in \mathcal{I})$ and $w(X + e_i) \geq w(X)$, then $X \leftarrow X + e_i$.
4. Output X.

Theorem 10 *The greedy algorithm outputs an optimum solution to the maximum weight independent set problem.*

Proof: Without loss of generality, assume $w(e) > 0, \forall e \in S$ and that there are no loops.

Claim 11 *There exists an optimum solution that contains e_1.*

Assuming this claim is true, we can use induction to show that greedy algorithm has to yield an optimum solution. This is because the greedy algorithm is recursively finding an optimum solution in the matroid $\mathcal{M}' = (S - e, \mathcal{I}')$ where $\mathcal{I}' = \{I - e_1 | I \in \mathcal{I}'\}$.

To prove the claim, let I^* be an optimum solution. If $e_1 \in I^*$, we are done, else, we can see that $I^* + e_1$ is not independent, otherwise $w(I^* + e_1) > w(I^*)$. Thus $I^* + e_1$ contains a circuit, and hence $\exists e \in I^*$ such that $I^* - e + e_1 \in \mathcal{I}$ (See Corollary 22 in the last lecture). $w(I^* - e + e_1) \geq w(I^*)$ since $w(e_1)$ has the largest weight among all the elements in the set. Thus there is an optimum solution $I^* - e + e_1$ that contains e_1. □

Remark 12 *If all weights are non-negative then it is easy to see that the greedy algorithm outputs a base of $\mathcal{M}$. We can adapt the greedy algorithm to solve the maximum weight base problem by making all weights non-negative by adding a large constant to each of the weights. Thus max-weight base problem, and equivalently min-cost base problem can be solved (by taking the weights to be the negative of the costs).*

Remark 13 *Kruskal's algorithm for finding the maximum weight spanning tree can be interpreted as a special case of the greedy algorithm for matroids when applied to the graphic matroid corresponding to the graph.*

1.3.2 Oracles for a Matroid

Since the set of all independence sets could be exponential in $|S|$, it is infeasible to use this representation. Instead we resort to one of the two oracles in order to efficiently solve optimization problems:

- An independence oracle that given $A \subseteq S$, returns whether $A \in \mathcal{I}$ or not.
- A rank oracle that given $A \subseteq S$, returns $r_{\mathcal{M}}(A)$.

These two oracles are equivalent in the sense that one can be recovered from the other in polynomial time.

1.4 Matroid Polytopes

Edmonds utilized the Greedy algorithm in proving the following theorem:

Theorem 14 *The following polytope is the convex hull of the characteristic vectors of the independent sets of a matroid $\mathcal{M} = (S, \mathcal{I})$ with rank function $r_{\mathcal{M}} : 2^S \rightarrow \mathcal{Z}_+$,*

$$x(A) \leq r_{\mathcal{M}}(A) \quad \forall A \subseteq S,$$
$$x(A) \geq 0.$$

Also, the system of inequalities described above is TDI.

Proof: We will show that the above system of inequalities is TDI (Totally Dual Integral), which will in turn imply that the polytope is integral since $r_{\mathcal{M}}(.)$ is integer valued.

Let us consider the primal and dual linear programs for some integral weight vector $w : S \rightarrow \mathcal{Z}$. We will show that the solution picked by Greedy algorithm is the optimal solution for primal by producing a dual solution that attains the same value. Alternately we could show that the dual solution and the primal solution picked by Greedy satisfy complementary slackness.

$$\begin{aligned} \text{Primal:} \quad \max \sum_{e \in S} w(e)x(e) & \\ x(A) \leq r(A), & \quad \forall\ A \subseteq S \\ x \geq 0 & \\ \text{Dual:} \quad \min \sum_{A \in S} r(A)y(A) & \\ \sum_{A: e \in A} y(A) \geq w(e), & \quad \forall e \in S \\ y \geq 0 & \end{aligned}$$

Let $S = \{e_1, ..., e_n\}$ such that $w(e_1) \geq w(e_2) ... \geq w(e_n) \geq 0$, since setting $w(e_i) = 0$ whenever $w(e_i) < 0$ does not alter the solution of the primal or dual. Define $A_j = \{e_1, ..., e_j\}$ with $A_0 = \emptyset$. It is easy to see that $r(A_j) = r(A_{j-1}) + 1$ iff e_j is picked by Greedy. Consider the following dual solution

$$\begin{aligned} y(A_j) &= w(e_j) - w(e_{j+1}), j < n \\ &= w(e_n), j = n \\ y(A) &= 0, \text{ if } A \neq A_j \text{ for some } j \end{aligned}$$

Claim 15 *y is dual feasible.*

Clearly, $y \geq 0$. Since $w(e_j) \geq w(e_{j-1}) \forall j$, for any e_i,

$$\begin{aligned} \sum_{A: e_i \in A} y(A) &= \sum_{j \geq i} y(A_i) \\ &= \sum_{j=i}^{n-1} \{w(e_j) - w(e_{j+1})\} + w(e_n) \\ &= w(e_i) \end{aligned}$$

Define $I = \{i | e_i \text{ is picked by Greedy}\}$. As we noted earlier, $i \in I \iff r(A_i) = r(A_{i-1}) + 1$.

Claim 16

$$\sum_{i \in I} w(e_i) = \sum_{A \subseteq S} r(A) y(A)$$

$$\begin{aligned}
\sum_{i \in I} w(e_i) &= \sum_{i \in I} w(e_i)(r(A_i) - r(A_{i-1})) \\
&= \sum_{j=1}^{n} w(e_j)(r(A_j) - r(A_{j-1})) \\
&= w(e_n)y(A_n) + \sum_{j=1}^{n-1}(w(e_j) - w(e_{j+1}))r(A_j) \\
&= \sum_{j=1}^{n} y(A_j) r(A_j) \\
&= \sum_{A \subseteq S} r(A) y(A)
\end{aligned}$$

Thus y has to be dual optimal, and the solution produced by Greedy has to be primal optimal. This means that the dual optimal solution is integral whenever w is integral, and therefore the system is TDI.

□

Corollary 17 *The base polytope of $\mathcal{M} = (S, I)$, i.e., the convex hull of the bases of $\mathcal{M}$ is determined by*

$$\begin{aligned}
x(A) &\leq r(A), \forall A \subseteq S, \\
x(S) &= r(S) \\
x &\geq 0
\end{aligned}$$

1.4.1 Spanning Set Polytope

Another polytope associated with a matroid is the spanning set polytope, which is the convex hull of the incidence vectors of all spanning sets.

Theorem 18 *The spanning set polytope of a matroid $\mathcal{M} = (S, I)$ with rank function $r_{\mathcal{M}}$ is determined by*

$$\begin{aligned}
0 \leq x(e) \leq 1, &\quad \forall e \in S \\
x(U) \geq r_{\mathcal{M}}(S) - r_{\mathcal{M}}(S \backslash U), &\quad \forall U \subseteq S.
\end{aligned}$$

Proof: A given set $A \subseteq S$ is spanning in $\mathcal{M}$ iff $S \backslash A$ is independent in $\mathcal{M}^*$. Thus $x \in \mathcal{P}_{\text{spanning}}(\mathcal{M})$ iff $1 - x \in \mathcal{P}_{\text{independence}}(\mathcal{M}^*)$. Now, by the relation between the ranks of dual matroids,

$$r_{\mathcal{M}^*}(U) = |U| + r_{\mathcal{M}}(S \backslash U) - r_{\mathcal{M}}(S).$$

Thus $1 - x \in \mathcal{P}_{\text{independence}}(\mathcal{M}^*)$ iff

$$\begin{aligned} 1 - x &\geq 0, \\ |U| - x(U) &\leq r_{\mathcal{M}^*}(U) = |U| + r_{\mathcal{M}}(S \backslash U) - r_{\mathcal{M}}(S), \end{aligned}$$

which matches the statement of the theorem. □

1.4.2 Separation Oracle

We have now determined that the independence polytope of a matroid is given by the linear conditions $x \geq 0$ and $x(U) \leq r_{\mathcal{M}}(U), U \subseteq S$. The greedy algorithm allows us to optimize over the polytope and by the equivalence between optimization and separation, there is a polynomial time separation oracle for the polytope. It is instructive to consider it explicitly.

For the separation problem, given a test vector $w : S \to \mathbb{R}$, we need to find out if $w \in \mathcal{P}_{\text{independence}}(\mathcal{M})$. We can easily test for non-negativity. To test the second condition, it is sufficient to check that $\min_{A \subseteq S}(r_{\mathcal{M}}(A) - w(A)) \geq 0$. In fact any violated inequality in the linear system can be found by constructing the set

$$U = \arg\min_{A \subseteq S}(r_{\mathcal{M}}(A) - w(A)).$$

Define $f : 2^S \to \mathbb{R}$, where $f(A) = r_{\mathcal{M}}(A) - w(A)$. f is a submodular set function since $r(.)$ is the submodular rank function and $-w(.)$ is modular. Thus if we can minimize an arbitrary submodular function specified by a value oracle, we can use the same for separating over a matroid polytope. However, there is a more efficient algorithm for separating over the independence polytopes given by Cunningham. See [1] for details.

References

[1] Lex Schrijver, "Combinatorial Optimization: Polyhedra and Efficiency, Vol. B," Springer-Verlag 2003.