18.438 Advanced Combinatorial Optimization October 29, 2009

Lecture 13

Lecturer: Michel X. Goemans *Scribe: Alan Deckelbaum*

In today's lecture, we cover an algorithm for matroid union, discuss some applications, and analyze Shannon switching games.

1 Matroid Union

For $i = 1, \dots, k$, let $M_i = (S_i, \mathcal{I}_i)$ be a matroid. Recall that we define the matriod union to be

$$M = M_1 \vee M_2 \vee \cdots \vee M_k = (\cup_{i=1}^{k} S_i, \mathcal{I} = \{\cup_{i=1}^{k} I_i | I_i \in \mathcal{I}\}).$$

Last class, we showed that M is a matroid, and derived the rank function

$$r_M(U) = \min_{T \subseteq U} \left[|U \setminus T| + \sum_{i=1}^{k} r_{M_i}(T \cap S_i) \right].$$

We now discuss the question of given I, how can we efficiently check if $I \in \mathcal{I}$.

1.1 Testing for independence

There are two main ways for checking if a given set is independent in the matroid union.

1.1.1 Matroid Intersection

We discuss two ways to determine, given $I \subset S$, whether $I \in \mathcal{I}$. The first method is based on the previous lecture, where we considered the set $\hat{S} = \dot{\cup} \hat{S}_i$, a union of disjoint copies of the S_i. Formally, we write $\hat{S} = \{(e, i) : e \in S_i\}$. There is a natural mapping $f : \hat{S} \to \cup S_i$ which maps (e, i) to e. We can now define a partition matroid M_p over the ground set $\hat{S}$, where $\mathcal{I}(M_p) = \left\{ I \subseteq \hat{S} \ s.t. \ \forall e \in S : |I \cap \{(e, i) : e \in S_i\}| \leq 1 \right\}$. (Thus, an independent set I in M_p does not contain both (e, i) and (e, j) for any $e \in S, i \neq j$.) We can now test for independence in the matroid union by looking at $f^{-1}(U)$ and using our matroid intersection algorithm with the partition matroid to look for a maximal such subset of $f^{-1}(U)$. A potential difficulty of this approach is that $\hat{S}$ can be large. (If, for example, all M_i are defined over the same ground set, then $\hat{S}$ is k times as large as this ground set.)

1.1.2 Matroid "Partition"

Because of the above difficulty with the matroid intersection approach, we will frequently use a different technique for testing whether a set is independent in the matroid union. This approach is called matroid "partition." Given some independent set $I \in \mathcal{I}(M_1 \vee M_2 \vee \cdots \vee M_k)$ and given a partition of $I = I_1 \dot{\cup} I_2 \dot{\cup} \cdots \dot{\cup} I_k$ with $I_i \in \mathcal{I}_i$, and given some $s \notin I$, we wish to determine whether $I + s \in \mathcal{I}$. (Clearly, if we have such an algorithm, then we can test independence of our set by adding elements one at a time and testing for independence as each element is added.)

For each matroid M_i, we create a bipartite directed graph $D_{M_i}(I_i)$. The vertex set of $D_{M_i}(I_i)$ correspond to the elements in S. The left vertices of this graph are the elements of I_i, and the right vertices are the elements of $S_i \setminus I_i$. We put a directed edge left-to-right from $y \in I_i$ to $x \in S_i \setminus I_i$ if $I_i - y + x \in \mathcal{I}_i$.

We now combine these graphs D_{M_i} into a single graph $D = \cup D_{M_i}(I_i)$, which is the superposition of the edges from all of the graphs. For each i, we set F_i to be the set of vertices in $S_i \backslash I_i$ which can be inserted into I_i such that $I_i + x$ is independent in M_i. Formally, we have

$$F_i = \{x \in S_i \backslash I_i : I_i + x \in \mathcal{I}_i\}.$$

We set $F = \cup F_i$, and note that F is the set of elements in $S \backslash I$ which can be inserted into I (any such element might already be in I) and have the set remain independent in the union matroid. Recall that our goal is to determine whether $I + s \in \mathcal{I}$, given some set I which is independent in the union matroid. We now prove the following theorem, which provides a complete answer to this question.

Theorem 1 *For any $s \in S \backslash I$, we have $I + s \in \mathcal{I} \Leftrightarrow$ there exists a directed path from F to s in D.*

Proof: I will first show the ($\Leftarrow$) direction. Suppose that there exists a directed path from F to s, and let P be a shortest such path. Write the vertices of P as $\{s_0, s_1, \ldots, s_p\}$. We know that $s_0 \in F$, so WLOG we may assume $s_0 \in F_1$. For each $j = 1, \ldots, k$, we now define the set of vertices $S_j = \{s_i, s_{i+1} : (s_i, s_{i+1}) \in D_{M_j}(I_j)\}$. (Thus, S_j contains the endpoints of the edges of P which belong to $D_{M_j}(I_j)$.)

We now set $I_1' = (I_1 \triangle S_1) \cup \{s_0\}$. For each $j > 1$, we set $I_j' = I_j \triangle S_j$. It is clear that $\cup_j I_j' = I + s$. Thus, to show that $I + s$ is independent in the matroid union, it suffices to show that $I_j' \in \mathcal{I}_j$ for each j. Notice that, since we chose P to be a shortest path, for each $j > 1$ there is a unique matching in $D_{M_j}(I_j)$ between the elements we removed and the elements we added to construct $I_j' = I_j \triangle S_j$. Thus, since this matching is unique, from our earlier analysis with matroid intersection we know that the resulting set $I_j' \in \mathcal{I}_j$. Similarly, since $s_0 \in F_1$, we see that $I_1' \in \mathcal{I}_1$ (see our analysis for matroid intersection). Therefore we see that $I + s$ is independent in the matroid union. (Note that the total number of call to an independence oracle in one iteration of this algorithm is $\sum_i (|I_i|)(|S_i| - |I_i|)$ (since we must construct each of the $D_{M_i}(I_i)$ graphs) and thus this algorithm runs in polynomial time.)

I will now show the ($\Rightarrow$) direction of the proof. Suppose that there is no directed path from F to s in D. We now set T to be the set of vertices of D from which we can reach s. That is, $T = \{x | \exists \text{ a directed path from } x \text{ to } s\}$. By assumption, we have $F \cap T = \emptyset$.

I claim now that for all i, we have $|I_i \cap T| = r_i(T \cap S_i)$. (This statement implies that $I_i \cap T$ is a maximal subset of T which is independent in M_i.) Suppose for the sake of contradiction that this were not the case. Since $|I_i \cap T| = r_i(I_i \cap T) \le r_i(T \cap S_i)$ is clear, the only remaining possibility is that $|I_i \cap T| < r_i(T \cap S_i)$. This means that there exists an $x \in T \cap (S_i \setminus I_i)$ such that $(I_i \cap T) + x \in \mathcal{I}_i$. But $x \notin F$ (since $x \in T$ and $T \cap F = \emptyset$) and thus $I_i + x \notin \mathcal{I}_i$. Thus (since $I_i + x$ contains a unique circuit), it must be the case that there exists a $y \in I_i - T$ such that $I_i + x - y \in \mathcal{I}_i$. However, this means that (y, x) is an edge in $D_{M_i}(I_i)$, which implies that $y \in T$, contradicting the choice of $y \in I_i - T$. Therefore, we know that for all i, we have $|I_i \cap T| = r_i(T \cap S_i)$.

Notice that, since $s \in T$, we have:

$$(I + s) \cap T = (\cup I_i + s) \cap T = \cup (I_i \cap T) + s.$$

From the rank function of a matroid union, we know that

$$r_M(I + s) \le \left[|(I + s) \setminus T| + \sum_{i=1}^{k} r_i(T \cap S_i) \right]$$

$$r_M(I + s) \le |(I + s) \setminus T| + \sum_{i=1}^{k} |I_i \cap T| = |I \setminus T| + \sum_{i=1}^{k} |I_i \cap T| = |I| < |(I + s)|$$

and thus $(I + s) \notin \mathcal{I}$, as desired. This completes the proof.

$\square$

2 Basis Exchange and Other Applications of Matroid Union

We will now show several applications of matroid union. We begin by looking at a theorem concerning "basis exchange."

Theorem 2 *Let $M = (S, \mathcal{I})$ be any matroid, and let $\mathcal{B}$ be the collection of all bases of M. Let $B_1, B_2 \in \mathcal{B}$, and let $B_1 = X_1 \dot\cup Y_1$ be an arbitrary partition of B_1 into two disjoint subsets. Then there is a partition of $B_2 = X_2 \dot\cup Y_2$ such that $X_1 \cup Y_2 \in \mathcal{B}$ and $X_2 \cup Y_1 \in \mathcal{B}$.*

Proof: The proof follows from matroid union. Let $M_1 = M/Y_1$ be M contracted by Y_1, and let $M_2 = M/X_1$ be M contracted by X_1. Then we observe that $X_1 \in \mathcal{I}_1$ and $Y_1 \in \mathcal{I}_2$. Therefore, we see that B_1 is independent in the matroid union $M_1 \vee M_2 = (S, \mathcal{I}_1 \vee \mathcal{I}_2)$. All that remains is to show that $B_2 \in \mathcal{I}_1 \vee \mathcal{I}_2$, since this will provide an appropriate way of partitioning B_2.

We now compute

$$r_{M_1 \vee M_2}(B_2) = \min_{U \subseteq B_2} \left[|B_2 \setminus U| + r_{M_1}(U - Y_1) + r_{M_2}(U - X_1) \right]$$

Notice that for any set $A \subseteq S - Y_1$, we have $r_{M_1}(A) = r_M(A \cup Y_1) - r_M(Y_1)$. Thus, we compute

$$r_{M_1 \vee M_2}(B_2) = \min_{U \subseteq B_2} \left[|B_2 \setminus U| + r_M(U \cup Y_1) - r_M(Y_1) + r_M(U \cup X_1) - r_M(X_1) \right].$$

By using submodularity on $r_M(U \cup Y_1) + r_M(U \cup X_1)$, we conclude

$$r_{M_1 \vee M_2}(B_2) \geq \min \left[|B_2 \setminus U| + r_M(U \cup X_1 \cup Y_1) + r_M(U) - |Y_1| - |X_1| \right].$$

Since $r_M(U) = |U|$ and $r_M(U \cup X_1 \cup Y_1) = |X_1| + |Y_1|$ (as $B_1 \in \mathcal{B}$), we conclude that

$$r_{M_1 \vee M_2}(B_2) \geq |B_2|$$

and hence $r_{M_1 \vee M_2}(B_2) = |B_2|$, so $B_2 \in \mathcal{I}_1 \vee \mathcal{I}_2$, as desired. Therefore B_2 can be partitioned as described in the statement of the theorem.

□

We now look at the special case of matroid union where we take the union of k copies of the same matroid: $M^k = M \vee M \vee \cdots \vee M$. From our standard rank equation for matroid union, we have

$$r_{M^k}(U) = \min_{T \subseteq U} \left[|U \setminus T| + k r_M(T) \right].$$

Notice that in the above equation, it suffices to minimize only over "flat" sets $T \subseteq U$, where T is a "flat" if $span(T) = T$. This is because we only make the above quantity smaller if we can increase $|T|$ without also increasing $r_M(T)$.

Since $r_{M^k}(U)$ is simply the maximum size of the union of subsets $U_1, U_2, \ldots U_k \subseteq U$ with each $U_i \in \mathcal{I}$, we can obtain several immediate results by using the above formula.

Theorem 3 (Matroid base covering) *A matroid M can be covered by k bases if and only if $\forall T \subseteq S : |T| \leq k r_M(T)$.*

The above theorem is obtained by noting that M can be covered by k bases if and only if $r_{M^k}(S) = |S|$.

Theorem 4 (Matroid base packing) *A matroid M contains k disjoint bases if and only if $\forall T \subseteq S : |S \setminus T| \geq k[r_M(S) - r_M(T)]$.*

The above theorem is obtained by noting that M contains k disoint bases if and only if $r_{M^k}(S) = kr_M(S)$.

We can obtain two more results by specializing to the case where M is a graphic matroid. Notice that a flat in a graphic matroid corresponds to picking some disjoint sets of vertices $U_1, U_2, \ldots, U_l$ of V (these sets need not partition V) and then taking all edges which lie entirely within a U_i.

Theorem 5 (Nash-Williams) *A graph G can be covered by k spanning trees if and only if*

$$\forall U \subseteq V : |E(U)| \leq k(|U| - 1).$$

Theorem 6 (Tutte, Nash-Williams) *A graph G contains k edge-disjoint spanning trees if and only if for all partitions $(V_1, \ldots, V_p)$ of V:*

$$|\delta(V_1, \ldots, V_p)| \geq k(p - 1),$$

where $\delta(V_1, \ldots, V_p)$ denotes the edges between different members of the partition.

3 Shannon Switching Game

The Shannon Switching Game is played on a graph $G = (V, E)$ with two identified vertices $u, v \in V$. There are two players, referred to as "cut" and "join." Turns in the game alternate between players. On cut's turn, cut deletes an edge from G. On join's turn, join fixes an edge so that it cannot be deleted. Join wins the game if join can fix edges connecting u to v. (Join may also fix additional edges.) Cut wins the game if join doesn't. Notice that in this game, it is never disadvantageous to move. (Cut can be no worse off if the graph has one fewer edge, and join can be no worse off if an additional edge is fixed.) Therefore, any game falls into one of three categories:

- Join game: Join wins, even if join plays second.
- Cut game: Cut wins, even if cut plays second.
- Neutral game: Whichever player moves first wins.

Lehman [2] generalized this game to matroids and provided characterizations and winning strategies for all types of games. In the graphical case, he inserts an additional edge e into G which connects u and v. Neither player is allowed to move on this edge. With this new framework, join wants to create a circuit containing e in M_1, and cut wants to create a circuit in the dual M^* containing e:

$$M = (E + e, \mathcal{I}), M^* = (E + e, \mathcal{I}^*).$$

In this more general matroid setting, join wants to pick $A \subseteq E$ such that $A + e$ contains a circuit C of M with $e \in C$. Cut wants to pick $B \subseteq E$ such that $B + e$ contains a circuit C of M^* with $e \in C$. This will be discussed in more details in the next lecture.

Edmonds [1] looked at such games in a more general framework over an arbitrary matroid. In this setting, we have a matroid $M = (N \dot\cup K, \mathcal{I})$, where cut and join play on the set N. (The set K plays the role of $\{e\}$ in the above example. Join wins if he can get a set $A \subseteq N$ such that $span(A) \supseteq K$. Cut wins if join loses.

A special case of the above game is where K spans the entire ground set. In this case, join wants to get a basis, while cut wants to prevent join from getting a basis. For example, if our matroid is the graphic matroid, then join wants to get a spanning tree.

Theorem 7 *A matroid $M = (S, \mathcal{I})$ is a join game (for the above game, where join wants to get a basis) if and only if M contains 2 disjoint bases.*

Proof: We first prove the ($\Leftarrow$) direction. Suppose that M contains 2 disjoint bases. It suffices to give a winning strategy for join if join plays second. Let B_1, B_2 be disjoint bases of M. On cut's turn, cut deletes an element. If cut deletes an element which is in neither B_1 nor B_2, then join can play arbitrarily (or we can simply let cut go again.) Suppose that cut deletes an element $x \in B_1$. We want to fix a $y \in B_2$ such that $B_1 - x + y$ is a basis. This follows from the basis exchange axiom. In the next round of the game, join wants to build a basis in the matroid M/y (M with y contracted). We observe that $B_1 - x$, $B_2 - y$ are disjoint bases in M/y, and we're done by induction.

I now prove the ($\Rightarrow$) direction. Suppose that M doesn't contain 2 disjoint bases. To show that this is not a join game, I will show that cut can win if cut goes first. Since M doesn't contain 2 disjoint bases, from Theorem 4 we conclude that there exists a $T \subseteq S$ such that $|S \setminus T| < 2(r_M(S) - r_M(T))$. On cut's turn, cut always plays in $S \setminus T$. Since cut plays first, we see that cut will get to delete at least half of $S \setminus T$. Therefore join will be able to fix strictly less than $r_M(S) - r_M(T)$ elements of $S \setminus T$. Therefore, the rank of all of the elements fixed by join will be strictly less than $r_M(T) + (r_M(S) - r_M(T)) = r_M(S)$, and thus join will not be able to fix a basis. $\square$

In our original (G, u, v) game (where we had a graph G and vertices u, v given, and join wants to fix a path between u and v), a similar result can be obtained. It can be shown that (G, u, v) is a join game if and only if there exist 2 disjoint cotrees spanning u, v. (This means that there are two disjoint trees spanning the same set of vertices U, with $u, v \in U$.) It is clear that the existence of 2 disjoint cotrees implies that join has a winning strategy. The other direction is less obvious to prove. This will be further discussed in the next lecture.

References

[1] J. Edmonds, "Lehman's switching game and a theorem of Tutte and Nash-Williams, *Journal of Research of the National Bureau of Standards – B*, **69B**, 73–77, 1965.

[2] A. Lehman, "A solution to Shannon's switching game", *J. Soc. Indust. Appl. Math.*, **12**, 687–725, 1964.