

Lecture 4: Pipage Rounding

Consider the maximum coverage problem: given a universe U and a collection of subsets of U : $\{S_1, \dots, S_m\}$, choose a collection of k sets so as to maximize the cardinality of their union.

One can consider the following greedy algorithm: pick the set which covers the maximum number of uncovered elements iteratively till you pick k sets. This algorithm achieves an a $(1 - (1 - 1/k)^k)$ approximation factor.

We now give another algorithm which achieves a better factor in certain cases. Consider the following linear program for the problem.

$$\begin{aligned} \max \quad & \sum_{i \in U} z_i && (x, z \geq 0) && (1) \\ \text{subject to} \quad & z_i \leq 1 && \forall i \in U && (2) \\ & z_i \leq \sum_{j: i \in S_j} x_j && \forall i \in U && (3) \\ & \sum_{j=1}^m x_j = k && && (4) \end{aligned}$$

Note that the above objective can be rewritten as $\max\{\sum_{i \in U} \min(1, \sum_{j: i \in S_j} x_j) : \sum_j x_j = k\}$. Denote $\{x \in [0, 1]^m : \sum_j x_j = k\}$ as \mathcal{P} and the function $L(x) := \sum_{i \in U} \min(1, \sum_{j: i \in S_j} x_j)$. Then from the above discussion (1) is equivalent to

$$\max \{L(x) : x \in \mathcal{P}\} \tag{5}$$

which can be solved in polytime and upper bounds the optimum.

Now consider the following *non-linear* function.

$$F(x) := \sum_{i \in U} (1 - \prod_{j: i \in S_j} (1 - x_j))$$

Observe that whenever $x \in \{0, 1\}^m$, $F(x)$ equals the value of the algorithm which picks sets with $x_j = 1$. Thus, the following *non-linear* program

$$\max \{F(x) : x \in \mathcal{P}\} \tag{6}$$

is also an upper bound on opt . It might seem pointless going to this non-linear program since we don't know how to solve this in polynomial time (even when the domain is $[0, 1]^m$). But what

makes this interesting is that (6) actually has an *integral* optimum unlike (1). More precisely, given any fractional solution to (6), there is a rounding algorithm which returns an integral solution of at least the value of the fractional solution. This is the so called *pipage rounding* algorithm which we will describe below.

To summarize, we know how to solve (5) but the solution is not integral. We don't know how to solve (6), but we know there is an integral optimum. To complete the picture, if we show $F(x) \geq \rho \cdot L(x)$ for all $x \in \mathcal{P}$ for some $\rho \leq 1$, then we have a ρ -approximation. To see this, solve (5) to get x^* . Run the pipage rounding algorithm to get x^{int} . We know that

$$\text{alg} = F(x^{\text{int}}) \geq F(x^*) \geq \rho \cdot L(x^*) \geq \rho \cdot \text{opt}$$

Before we describe the pipage rounding algorithm for (6), let us show that $\rho = (1 - (1 - 1/f)^f)$ suffices, where f is the maximum number of sets an element of U occurs in. So for the k -vertex coverage problem, we get a $3/4$ -approximation.

Claim 1. For all $x \in \mathcal{P}$, $F(x) \geq \rho \cdot L(x)$.

Proof. It suffices to show $1 - \prod_{j:i \in S_j} (1 - x_j) \geq \rho \cdot \min(1, \sum_{j:i \in S_j} x_j)$ for any $i \in U$. By the AM-GM inequality, the LHS is at least $1 - (1 - \frac{\sum_{j:i \in S_j} x_j}{f})^f$. If $\sum_{j:i \in S_j} x_j \geq 1$, then this is at least $1 - (1 - \frac{1}{f})^f$; therefore, we may assume $\sum_{j:i \in S_j} x_j < 1$.

Since the function $g(t) := 1 - (1 - \frac{t}{f})^f$ is concave in $t \in [0, 1]$, we have for $t < 1$, $g(t) \geq (1 - t)g(0) + tg(1) = t(1 - (1 - \frac{1}{f})^f)$. Thus,

$$1 - \prod_{j:i \in S_j} (1 - x_j) \geq 1 - \left(1 - \frac{\sum_{j:i \in S_j} x_j}{f}\right)^f \geq \left(\sum_{j:i \in S_j} x_j\right) \left(1 - \left(1 - \frac{1}{f}\right)^f\right)$$

□

Pipage Rounding

The setting where pipage rounding applies is more general than the one described above. Abstractly, suppose we want to maximize a function on m variables in $\{0, 1\}^m$ intersected with a polytope.

$$\max\{F(x) : x \in \mathcal{P} \cap \{0, 1\}^m\}$$

Now suppose for every non-integral $x \in \mathcal{P}$, there exists a vector v_x and $\alpha_x, \beta_x > 0$ such that both $x + \alpha_x v_x$ and $x - \beta_x v_x$ have strictly larger number of integral coordinates. Furthermore, for all $x \in \mathcal{P}$, also suppose the function $F()$ is *convex* in the direction v_x . That is, the one-dimensional function $g_x()$ defined as

$$g_x(t) := F(x + t v_x)$$

is a convex function. Then, the following abstract rounding procedure takes a feasible $x \in \mathcal{P}$, and returns an $\{0, 1\}^m$ -vector $x^{\text{int}} \in \mathcal{P}$ with $F(x^{\text{int}}) \geq F(x)$.

Procedure PIPAGE

Repeat till x is a $\{0, 1\}$ -vector: if $F(x + \alpha_x v_x) \geq F(x)$, $x = x + \alpha_x v_x$; else $x = x - \beta_x v_x$.

Since $F()$ is convex in direction of v_x , we know that $F(x) \leq \max(F(x + \alpha_x v_x), F(x - \beta_x v_x))$ (Because $g_x(0) \leq \frac{\beta_x}{\alpha_x + \beta_x} g_x(\alpha) + \frac{\alpha_x}{\alpha_x + \beta_x} g_x(-\beta_x)$). So the value of $F(x)$ never goes down. Furthermore, since the number of integral coordinates increases, the above procedure terminates in at most m steps.

Let us now move to concrete examples. To start with, let's take the coverage problem described above. Here, $\mathcal{P} := \{x \in [0, 1]^m : \sum_{j=1}^m x_j = k\}$. Suppose x is a non-integral vector in \mathcal{P} . Note that there are at least two fractional coordinates. Let it be x_p and x_q . Our vector v_x is then the vector $(e_p - e_q)$, where e_p is the vector with 1 in the p th coordinate and 0 elsewhere. $\alpha_x = \min(1 - x_p, x_q)$ and $\beta_x = \min(1 - x_q, x_p)$. Note that $x + \alpha_x v_x$ and $x - \beta_x v_x$ are vectors in \mathcal{P} with at least one less fractional coordinate.

$F(x) = \sum_{i \in U} (1 - \prod_{j: i \in S_j} (1 - x_j))$. For any $x \in \mathcal{P}$, and fractional coordinates x_p and x_q , note that the function $g_x(t)$ is also a sum of functions of t ; each function is either a constant (if $i \notin S_p$ and $i \notin S_q$), or a linear function (if i is in exactly one of S_p or S_q), or a quadratic in t with a positive coefficient for t^2 (if $i \in S_p$ and $i \in S_q$). All of these are convex, and thus $F()$ is convex in the direction of v_x . Therefore pipage rounding is applicable and we thus, as claimed before, given a fractional vector $x \in \mathcal{P}$, we can get an integral vector $x^{\text{int}} \in \mathcal{P}$ with $F(x^{\text{int}}) \geq F(x)$.

Let us summarize how pipage rounding can be used to obtain approximation algorithms for maximization problems (maximization is nothing special, the minimization case goes through with convexity replaced with concavity, etc). First we cast the problem as a problem of maximizing $\max\{F(x) : x \in \mathcal{P} \cap \{0, 1\}^m\}$. We need three things for all to work.

1. For any (non-integral) $x \in \mathcal{P}$, we need a vector v and $\alpha, \beta > 0$ such that $x + \alpha v$ or $x - \beta v$ have strictly more integral coordinates.
2. For all x , the function $F()$ needs to be convex in the direction of v . More precisely, the function $g_x(t) := F(x + tv)$ needs to be convex.
3. Finally, we need a *starting* fractional x with a guarantee that $F(x) \geq \rho \cdot \text{opt}$.

One way to do so is to get a *linear approximator* for F . That is, a linear function $L(x)$ such that $F(x) \geq \rho \cdot L(x)$ for all $x \in \mathcal{P}$. Then the starting x is the solution of the linear program $\max\{L(x) : x \in \mathcal{P}\}$ (assuming \mathcal{P} is a set of linear constraints). This was done in the application above. Another way could be just find an *approximate* solution to the non-linear program $\max\{F(x) : x \in \mathcal{P}\}$.

Remark 1. *If pipage rounding is applicable, then note that \mathcal{P} must be an integral polytope. Otherwise, given a fractional vertex $x \in \mathcal{P}$, one can never get the vector v and $\alpha, \beta > 0$ such that $x + \alpha v$ and $x - \beta v$ both lie in \mathcal{P} .*

Remark 2. *In fact, one doesn't require the first condition very strongly. It suffices if one can show $x + \alpha v$ and $x - \beta v$ "make progress" towards an integral solution. One possible measure of progress if both $x + \alpha v$ and $x - \beta v$ lie on a face of \mathcal{P} smaller dimension.*

Maximizing Monotone Submodular Functions over Matroid Constraints

Definitions. A set function $f : 2^U \rightarrow \mathbb{Z}_{\geq 0}$ is submodular if $f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$ for all $A, B \subseteq U$. It is monotone if for any pair of sets $A \subseteq B$, $f(A) \leq f(B)$. A set system $M := (U, \mathcal{I})$ form a matroid if it satisfies the following two conditions: a) if $B \in \mathcal{I}$ and $A \subseteq B$, then $A \in \mathcal{I}$, and b) for any two sets $A, B \in \mathcal{I}$, with $|A| < |B|$, there exists an element $i \in B$ such that $A \cup i \in \mathcal{I}$. The sets in \mathcal{I} are called *independent sets*.

Problem. Given a monotone submodular function f and a matroid $M := (U, \mathcal{I})$, find $S \in \mathcal{I}$ which maximizes $f(S)$.

This abstract problem above captures *a lot* of problems. In fact, sometimes it is not that easy to see that this is indeed the case. This is because the submodular function and the set system may have to be designed carefully. The coverage problem described above falls in this class. A few more will be covered in exercises.

Theorem 1 (Calinescu-Chekuri-Pál-Vondrák 2009). *There is a randomized $(1-1/e)$ approximation algorithm to the problem of maximizing a monotone submodular function over a matroid constraint.*

Matroid Polytope. Given a set $S \in \mathcal{I}$, we can define the characteristic vector of S as the $\{0, 1\}^n$ vector which has 1 in the coordinates corresponding to $i \in S$. The matroid polytope is defined as the convex hull of all the characteristic vectors. This can be represented as follows. We need a definition. Given a set $S \subseteq U$, the *rank* of S , $r_M(S)$, is the size of the largest independent subset of S . The matroid polytope is defined as

$$\mathcal{P} := \{x \in [0, 1]^n : \sum_{i \in S} x_i \leq r_M(S), \forall S \subseteq [n]\}.$$

The polytope has the following property. Given $x \in \mathcal{P}$, let \mathcal{L} denote a set of maximally linearly independent constraints which hold with equality. If x is not a basic feasible solution, we know that dimension of the space spanned by \mathcal{L} is strictly less than n . Then there exists i and j and positive α and β such that the number of linearly independent tight constraints which hold with equality with $x + \alpha(e_i - e_j)$ and $x - \beta(e_i - e_j)$ is strictly larger. This gives us the condition 1 for pipage rounding.

Let me give a sketch of how to find e_i, e_j, α, β given x . Call a set S tight if $\sum_{i \in S} x_i = r_M(S)$. Now the rank itself is a submodular function (why?), and this implies that if A and B are tight sets, so are their intersections and union (why?). Therefore, given i one can define the minimal tight set containing i . As a result, unless x is integral, one can find i, j such that the minimal tight set containing i coincides with the minimal tight set containing j . These are our i and j . Now consider increasing i and decreasing j ; we can keep doing this till a subset of this set becomes tight and this leads to a new linearly independent equality constraint, or till $x_j = 0$ in which case this is added as the linearly independent constraint. Whatever the value at which one of these happen gives α . The converse process gives β .

Submodular Function to Continuous Non-Linear Function. Given f , define $F : [0, 1]^n \rightarrow \mathbb{R}_{\geq 0}$ as follows:

$$F(x_1, \dots, x_n) := \sum_{S \subseteq [n]} \left(f(S) \prod_{i \in S} x_i \prod_{i \notin S} (1 - x_i) \right)$$

Note the following

1. $F(x_1, \dots, x_n)$ is the expected value of $f(S)$ where S is obtained by sampling each element i independently with probability x_i . This can be used to obtain an *estimate* of $F(x_1, \dots, x_n)$ – we might come back to this later in the class.
2. Monotonicity of f implies $\frac{\partial F}{\partial x_i} \geq 0$, for all i , for all x .
Submodularity implies $\frac{\partial^2 F}{\partial x_i \partial x_j} \leq 0$ for $i \neq j$, for all x .
3. For any two coordinates i, j ; let $v = e_i - e_j$. The function $g_{x,i,j}(t) = F(x + tv)$ is convex for all x, i, j . This is because

$$\frac{d^2 g_{x,i,j}(t)}{dt^2} = \sum_{p,q} \frac{\partial^2 F(x + tv)}{\partial x_p \partial x_q} v_p v_q = \frac{\delta^2 F(x + tv)}{\delta x_i^2} - 2 \frac{\delta^2 F(x + tv)}{\delta x_i \delta x_j} + \frac{\delta^2 F(x + tv)}{\delta x_j^2} \geq 0$$

Point 3 above gives us the condition 2 required by pipage rounding above. Point 1 and 2 above give the third condition for pipage rounding in form of the following theorem which we will not get into.

Theorem 2 (Vondrák 08). *If F satisfies point 2 above, and if ∇F can be estimated, and if given weights w_j the vector $v \in \mathcal{P}$ maximizing $\sum_j v_j w_j$ can be found in polynomial time, then there is a randomized $(1 - 1/e)$ -approximation to the problem of maximizing $F(x) : x \in \mathcal{P}$.*

Putting all together (and adding a bunch of details mainly arising out of the above theorem's implementation), we get the $(1 - 1/e)$ approximation to maximizing a monotone submodular function over a matroid constraint.