Contention Resolution Schemes

Bo Waggoner

March 13, 2015

1 Background and Recap

We wish to approximately maximize f, a nonnegative submodular function on 2^N. We consider the space $[0,1]^N$, where a set S corresponds to the indicator point x_S (with 1s at the coordinates of elements in S and 0s elsewhere). We can think of a point $x \in [0,1]^N$ as representing a distribution on 2^N, and let $R(x)$ denote a random set obtained by including each i independently with probability x_i.

The *multilinear extension* of f is denoted $F : [0,1]^N \to \mathbb{R}$, and is defined by $F(x) = \mathbb{E}_{S \sim x} f(S)$, where $S \sim x$ means to draw S randomly by including each element $i \in N$ independently with probability x_i. In other words, $F(x) = \sum_S f(S) \prod_{i \in S} x_i \prod_{i \notin S}(1 - x_i)$.

Last time, Eric covered the continuous greedy algorithm of Vondrák [2008], Calinescu et al. [2011]. In that setting, f was *monotone* and we had *matroid* constraints. The idea was to find a point x such that $F(x) \geq \left(1 - \frac{1}{e}\right) \max_y F(y)$, noting that $\max_y F(y) \geq \max_S f(S)$. We could then round this point x into a set S such that $f(S) \geq F(x)$ using the pipage rounding technique covered by Jean, of Ageev and Sviridenko [2004].

2 Overview: This Time

We now examine maximizing submodular functions f that may be *non-monotone* over *independence* constraints. These notes cover the paper introducing contention resolution schemes for this problem, Vondrák et al. [2011].

Setup: f is a nonnegative submodular function on 2^N, with multilinear extension F. $\mathcal{I}$ is a downward-closed set of subsets: If $A \in \mathcal{I}$ and $A' \subseteq A$, then $A' \in \mathcal{I}$. We wish to approximately solve $\max_{S \in \mathcal{I}} f(S)$, given access to *value queries* (specify S and receive $f(S)$).[1]

As in other papers recently, we take two steps: (1) approximately maximize the *multilinear extension* F of f; (2) round this solution to get an appoximately optimal set for f.

For our problem, the polytope over which we maximize F is $P_{\mathcal{I}} = Conv(\{x_S : S \in \mathcal{I}\})$, the convex hull of the set of indicators of the independent sets. Note that if our constraint is a matroid, this is the same polytope as $P(\mathcal{M}) = \{y \in [0,1]^N : \forall S, \sum_{i \in S} y_i \leq r_M(S)\}$ with r_M the rank function.

Step 1: An α-approximation algorithm for F. One shows that there is a polynomial-time algorithm returning some y with $F(y) \geq \alpha \max_{x \in P} F(x)$ for any polytope P that is "down-closed" ($x \in P \implies cx \in P$ for all $c \in [0,1]$) and "solvable" (one can optimize linear objectives over P in polynomial time).

In this paper, the first such (constant-factor) algorithm was given for multilinear extensions of non-monotone f over general downward-closed constraints (previous cases such as continuous greedy were for monotone f and/or matroids). The factor obtained is $\alpha = 0.325$ (with a complicated simulated annealing approach that appears only in the conference version of the paper; they also give essentially continuous-greedy approaches achieving 0.25 and 0.309). However, subsequent to this paper, Feldman et al. [2011] give a "unified continuous greedy" algorithm with at $\frac{1}{e} - \epsilon \approx 0.367$ approximation. So we will not cover the algorithms in this paper.

Step 2: "Contention resolution schemes" for rounding approximate maximizers of F. The algorithm from Step 1 returns some vector $x \in [0,1]^N$. The problem now is to round x to obtain a solution that is both feasible and has high value. This paper introduces *monotone c-balanced contention resolution schemes* for this purpose. The idea is to create a set S by including each element i with probability x_i independently, then to construct some final output as a subset I of S so that $I \in \mathcal{I}$ (so it is feasible) and there is a guarantee for each element of S on its probability of inclusion in I, in terms of c. It is then shown that such a scheme implies that the final set I satisfies $\mathbb{E} f(I) \geq cF(x)$.

The name "contention resolution" comes (I believe) from the idea that each element $i \in \text{support}(x)$ is "contending" to be included in the final solution, but this violates the constraints and so these contentions must be resolved. The term "balanced" comes from the c, which roughly ensures that each element has a "c chance" of being included.

Combining the steps. Given an α-approximation algorithm for F and a monotone c-balanced CR scheme, we have a αc approximation in expectation, since we have an output I satisfying

$$\mathbb{E} f(I) \geq cF(x) \geq c\left(\alpha \max_{x \in P_{\mathcal{I}}} F(x)\right) \geq c\alpha \max_{S \in \mathcal{I}} f(S).$$

(Recall that the optimum of the multilinear extension upper-bounds the optimum of f since any feasible solution S of f yields a feasible solution for F of the same value.)

Other contributions. The paper also characterizes when CR schemes can be constructed and traces a connection to the "correlation gap", which we will briefly cover; and shows that CR schemes can be constructed for matroids, knapsacks, and some other packing constraints. The actual approximation ratios were improved due to later improvements for approximating F, so I don't list the ratios here.

What are CR Schemes good for? (Note: subject to later improvements we've not yet covered....) If you want to maximize f over a matroid constraint, even if f is not monotone, then you should use pipage rounding instead of CR schemes and lose nothing in the rounding step — this is due to the appendix of Vondrák [2013] and is mentioned in Feldman et al. [2011], where together with their $\frac{1}{e}$ approximation for maximizing F it gives $\frac{1}{e}$ for maximizing any nonnegative submodular f over a matroid.

[1] The paper will not achieve this in polynomial time for all families $\mathcal{I}$, but for certain ones like matroid constraints and knapsack constraints.

But CR schemes are good for other types of constraints like knapsack, and especially good at composing (if we have a CR scheme for two different constraints, we have one for the intersection of their constraints).

3 Contention Resolution

Let $R(x)$ be a random set obtained by including each item i with probability x_i independently. The following definition and theorem are simplifications of those presented in the paper, but they capture the intuition.

Definition 1. Let $b, c \in [0,1]$. A (b,c)-*balanced Bo-CR scheme* π for $P_{\mathcal{I}}$ is a procedure that, for every $x \in bP_{\mathcal{I}} = \{b \cdot x : x \in P_{\mathcal{I}}\}$ and every $A \in 2^N$, returns a (random) set $I = \pi_x(A)$ where, first, $I \in \mathcal{I}$ and $I \subseteq A \cap \text{support}(x)$; and second, for each $i \in \text{support}(x) \cap A$, we have $\Pr_I[i \in I] \geq c$.

Why is the b useful? Because of continuous-greedy-type algorithms: We can always terminate them at time $b < 1$ to obtain a solution in $bP_{\mathcal{I}}$. So we include b in the definition and this is useful in some contexts.

Now we show why the definition is useful: If we have a CR scheme, then we can use it to round any x to get (in expectation) a c-approximation to $F(x)$. The idea is to draw $R(x)$ randomly by taking each i with probability x_i, then apply the CR scheme to $R(x)$.

Theorem 2. *If π is a monotone (b,c)-balanced Bo-CR scheme, then for any $x \in bP_{\mathcal{I}}$, letting the random output $I = \pi_x(R(x))$, we have $\mathbb{E} f(I) \geq cF(x)$.*

Proof. Note that $cF(x) = c\,\mathbb{E} f(R(x))$. Let $I_i = I \cap \{1,\dots,i\}$; that is, I when only considering the first i elements of N. Similarly, let $R = R(x)$ be the randomly drawn input and let $R_i = R \cap \{1,\dots,i\}$. Then using linearity of expectation,

$$\begin{aligned}
\mathbb{E} f(I) &= \mathbb{E}_{R \leftarrow R(x)} \sum_i \mathbb{E}_{I \leftarrow \pi_x(R)} \left[f(I_i) - f(I_{i-1})\right] \\
&= \mathbb{E}_{R \leftarrow R(x)} \sum_{i \in R} \mathbb{E}_{I \leftarrow \pi_x(R)} \left[f(I_i) - f(I_{i-1})\right] \\
&= \mathbb{E}_{R \leftarrow R(x)} \sum_{i \in R} \sum_{I \subseteq R : i \in I} \Pr[I = \pi_x(R)] \left(f_{I_{i-1}}(i)\right) \\
&\geq \mathbb{E}_{R \leftarrow R(x)} \sum_{i \in R} \sum_{I \subseteq R : i \in I} \Pr[I = \pi_x(R)] \left(f_{R_{i-1}}(i)\right) \\
&\geq \mathbb{E}_{R \leftarrow R(x)} \sum_{i \in R} c \left(f_{R_{i-1}}(i)\right) \\
&= c\,\mathbb{E}_{R \leftarrow R(x)} f(R).
\end{aligned}$$

Here, we used the notation $f_S(i) = f(S \cup \{i\}) - f(S)$. The first inequality followed by submodularity, because $I_{i-1} \subseteq R_{i-1}$ (since $I \subseteq R$). The second inequality followed by the definition of (b,c)-balanced CR schemes, because the total probability of picking an $I \subseteq R$ with $i \in I$ is at least c. $\square$

Now let's give the full definition:

Definition 3. Let $b, c \in [0,1]$. A (b,c)-*balanced CR scheme* π for $P_{\mathcal{I}}$ is a procedure that, for every $x \in bP_{\mathcal{I}} = \{b \cdot x : x \in P_{\mathcal{I}}\}$ and every $A \in 2^N$, returns a (random) set $I = \pi_x(A)$ where, first, $I \in \mathcal{I}$ and $I \subseteq A \cap \text{support}(x)$; and second, for each $i \in \text{support}(x)$, we have $\Pr_I[i \in \pi_x(R(x)) \mid i \in R(x)] \geq c$.

The scheme is *monotone* if $A_1 \subseteq A_2$ implies that $\Pr[i \in \pi_x(A_1)] \geq \Pr[i \in \pi_x(A_2)]$.

The paper proves the following:

Theorem 4. *Let η_f be the following "pruning function": $J = \eta(I)$ is constructed by sequentially adding each $i \in I$ to J only if it increases $f(J)$, or otherwise tossing out i.*

If π is a monotone (b,c)-balanced CR scheme, then for any $x \in bP_{\mathcal{I}}$, letting the random output $I = \pi_x(R(x))$, and $J = \eta_f(I)$, we have $\mathbb{E} f(J) \geq cF(x)$.

Some facts about composing or changing CR schemes:

Claim 5. *A (b,c)-balanced CR scheme π can be transformed into a $(1,bc)$-balanced CR scheme π'.*

The proof given by the authors is: define $\pi'_x(A)$ to simply remove each element of A independently with probability $1-b$, obtaining some set A', and return $\pi_x(A')$. But I am not completely clear on this, because it seems to me that they should return $\pi_{bx}(A')$.

Lemma 6. *If $\mathcal{I} = \bigcap_{i=1}^k \mathcal{I}_i$ and each $P_{\mathcal{I}_i}$ has a (b,c_i)-CR scheme, then $P_{\mathcal{I}}$ has a $(b,\prod_i c_i)$-CR scheme.*

The idea is to let $\pi_x(A) = \cap_i \pi_{i,x}(A)$. I think the intuition is that $\Pr[i \in \pi_x(A)] \geq \prod_i \Pr[i \in \pi_{i,x}(A)]$.

4 Existence of CR schemes

In this section, we use an LP to investigate existence and optimality of CR schemes. The LP is very large, so this won't always be useful for constructing a polynomial-time scheme (although sometimes it will), but will illuminate some structure.

Let us fix x and focus on computing the optimal scheme for x. A deterministic scheme is a mapping $\phi : 2^N$ to $\mathcal{I}$. Any randomized scheme can be written as a distribution λ over deterministic schemes, where with probability λ_ϕ we pick scheme ϕ and apply it. The key condition we need to satisfy is that $\Pr[i \in \pi_x(R(x)) \mid i \in R(x)] \geq c$. Using that π_x can be written as a distribution λ over ϕ, the left side is $\sum_\phi \lambda_\phi \Pr[i \in \phi(R(x)) \mid i \in R(x)]$. Thus the LP for maximizing c is

$$\begin{aligned}
\max \quad & c \\
\text{s.t.} \sum_\phi \lambda_\phi \Pr[i \in \phi(R(x)) \mid i \in R(x)] &\geq c \qquad \forall i \in [N] \\
\sum_\phi \lambda_\phi &= 1 \\
\lambda_\phi &\geq 0 \qquad \forall \phi.
\end{aligned}$$

The dual[2] is

$$\begin{aligned} &\min \quad \mu \\ \text{s.t.} &\sum_{i=1}^{N} z_i \Pr[i \in \phi(R(x)) \mid i \in R(x)] \le \mu \qquad \forall \phi \\ &\sum_{i=1}^{N} z_i = 1 \\ &z_i \ge 0 \qquad \forall i. \end{aligned}$$

We can think of this as a two-player game where the column player is choosing a distribution λ over ϕ and the row player is choosing a distribution z over i. When each player draws from her distribution, we get a ϕ and an i, and the column player's payoff is $\Pr[i \in \phi(R(x)) \mid i \in R(x)]$ while the row player's payoff is one minus this. The primal maximizes the amount the column player can guarantee herself, while the dual maximizes the amount the row player can guarantee herself.

4.1 Characterization of solution

An optimal solution, by strong duality, is the optimal value of the dual, which will be tight at some constraint:

$$\min_z \max_\phi \sum_{i=1}^{N} z_i \Pr[i \in \phi(R(x)) \mid i \in R(x)].$$

Noting that $\Pr[i \in \phi(R(x)) \mid i \in R(x)] = \Pr[i \in \phi(R(x))]/\Pr[i \in R(x)]$, and the denominator is x_i, let us make the change of variables $y_i = \frac{z_i}{x_i}$:

$$\begin{aligned} &\min_{y:\sum_i x_i y_i = 1} \max_\phi \sum_{i=1}^{N} y_i \Pr[i \in \phi(R(x))] \\ &= \min_{y:\sum_i x_i y_i = 1} \max_\phi \mathbb{E}_{S \leftarrow \phi(R(x))} \left[\sum_{i \in S} y_i \right]. \end{aligned}$$

The claim is that this is maximized by the ϕ that sets $\phi(R) = \arg\max_{S \subseteq R, S \in \mathcal{I}} \sum_{i \in S} y_i$. This is the "max-weight independent set problem". Define $\mathbf{r}_y(T) = \max_{S \subseteq T, S \in \mathcal{I}} \sum_{i \in S} y_i$ to be the "weighted rank function", or weight of the max-weight independent set for the problem. Then the optimal solution is

$$\begin{aligned} &\min_{y:\sum_i x_i y_i = 1} \mathbb{E}_{R \leftarrow R(x)} \mathbf{r}_y(R) \\ &= \min_{y \ge 0} \frac{\mathbb{E}_{R \leftarrow R(x)} \mathbf{r}_y(R)}{\sum_{i=1}^{N} x_i y_i} \end{aligned}$$

4.2 Efficiently solving this problem via the dual.

We can efficiently solve the dual if we have a separation oracle. To do this, given μ and z, we just need to check if there is any ϕ such that the corresponding constraint is violated, *i.e.* $\sum_i z_i \Pr[i \in \phi(R(x)) \mid i \in R(x)] > \mu$, and if so output a separating hyperplane.

By the previous subsection, if we have an algorithm for the max-weight independent set problem, then we can find this ϕ and thus this most-violated constraint. (The appendix describes how to find the separating hyperplane and handling details.) In addition, to estimate the violation of the constraint, we have to actually repeatedly randomly sample $R \leftarrow R(x)$ and compute $\phi(R)$ for the ϕ just defined (max-weight independent set).

The entire algorithm is therefore this: Run the ellipsoid algorithm using the above procedure as a separation oracle until we have found a polynomial number of constraints certifying that the dual optimum is at least μ where the true optimum μ^* satisfies $\mu^* \le \mu + \epsilon$. Then, solve the primal restricted to this polynomial number of variables corresponding to these constraints (again sampling is necessary). The paper also discusses how, if the max-weight independent set algorithm is monotone, then so is the resulting CR scheme.

Monotonicity. Luckily, it turns out that if the algorithm for max-weight independent set is monotone, then the resulting CR scheme is monotone.

5 Correlation Gap

Let f be a set function on $[N]$. For any distribution p on 2^N, let p_i be the marginal probability that i is included and let $\times_i p_i$ be the distribution on 2^N where i is included with probability p_i independently of all other elements. The *correlation gap* of f is

$$\inf_p \frac{\mathbb{E}_{S \sim \times_i p_i} f(S)}{\mathbb{E}_{S \sim p} f(S)}.$$

(Note that in mechanism design it is usually one over this value.)

We can (for fun) define the *concave closure* of a set function f; it is

$$f^+(x) = \max_p \left\{ \mathbb{E}_{S \sim p} f(S) \;:\; \sum_S p(S) \mathcal{I}_S = x \right\}$$

where $\mathcal{I}_S$ is the vector with ones at indices i in S and zeroes elsewhere. In other words, it is exactly the largest possible expected value of $f(S)$ over any distribution whose marginals are given by x. Thus the correlation gap can be written as

$$\inf_x \frac{F(x)}{f^+(x)}.$$

Now subject to a downward-closed constraint $\mathcal{I}$, define the correlation gap to be

$$\inf_{x \in P_\mathcal{I}} \frac{F(x)}{f^+(x)}.$$

Now in particular, if we consider the weighted rank function defined above, $\mathbf{r}_y(T) = \max_{S \subseteq T, S \in \mathcal{I}} \sum_{i \in S} y_i$, then the correlation gap is

$$\inf_{x \in P_\mathcal{I}, y \ge 0} \frac{\mathbb{E}\, \mathbf{r}_y(R(x))}{\mathbf{r}_y^+(x)}.$$

But what is $\mathbf{r}_y^+(x)$? Assuming that x is in the polytope, $x = \sum_{S \in \mathcal{I}} \alpha_S \mathbf{1}_S$, where $\mathbf{1}_S$ is the indicator vector for the set S. But if we take this distribution α, then we get each element i of weight y_i exactly x_i times, so we get $x \cdot y = \sum_i x_i y_i$. To make it more clear, we never have to drop any elements from the sets we pick, unlike when we draw independently from each coordinate. Thus, we have proven

Claim 7. *The correlation gap of the weighted rank function over $\mathcal{I}$ is the same as the maximum c such that $\mathcal{I}$ has a c-balanced CR scheme.*

[2] If reading the paper, notice that I have taken the change of variables $z_i = y_i x_i$.

We can also see this in the LP primal, as follows. Because $x \in P_{\mathcal{I}}$, x can be written as $\sum_S \alpha_S x_S$. Imagine that instead of drawing $R(x)$ by drawing each i independently with probability x_i, we drew $R(x)$ from the correlated distribution α (which still has marginals given by x). Then the identity function ϕ would be valid because $R(x)$ would always be in $\mathcal{I}$, and we would have $\Pr[i \in \phi(R(x)) \mid i \in R(x)] = 1$ for that choice of ϕ. So the value of the primal would be 1 if we allowed a correlated distribution on $R(x)$ with marginals x.

6 Constructing CRs

Theorem 8. *For any matroid on n elements and $b \in [0,1]$, there is a $\left(b, \frac{1-\left(1-\frac{b}{n}\right)^n}{b}\right)$-balanced contention resolution scheme for the associated polytope $P_{\mathcal{I}}$.*

The proof is essentially that, first, the greedy algorithm for max-weight independent sets works for matroids (and is monotone); and second, a correlation gap result.

The knapsack independence system is constructed as follows: The items have sizes $a_1, \dots, a_n \in [0,1]$ and we have a single knapsack constraint, so the resulting feasible sets are $\mathcal{F} = \{S : \sum_{i \in S} a_i \leq 1\}$. Naturally it is downward closed, and nicely the polytope $P_{\mathcal{F}} = \{x \in [0,1]^N : \sum_i a_i x_i \leq 1\}$.

Theorem 9. *There is a monotone $(b, 1-2b)$-balanced CR scheme for the knapsack polytope, and also some better schemes when given guarantees on the sizes of the items.*

The CI scheme is to randomly draw a set $R(x)$, order by size from largest to smallest, and output the longest prefix of this ordering that fits in the knapsack.

7 Unified Continuous Greedy

Let us briefly overview Feldman et al. [2011], "A Unified Continuous Greedy Algorithm for Submodular Maximization". The paper in a sense follows up on contention resolution, mainly by solving step (1) (optimize the multilinear extension) better.

The algorithm is called *measured continuous greedy*. The idea is to modify continuous greedy by shrinking the direction of movement according to the (discrete) gradient of F at the current point y. Specifically, given a polytope and a current timestep, we first greedily solve for $x = \arg\max\{w(y) \cdot x : x \in \mathbb{P}\}$ where $w(y)$ is a "discrete gradient", $w(y)_e = F(y \wedge \mathbf{1}_e) - F(y)$. Then each coordinate x_e is decreased to $x_e(1-y_e)$. You can think of this decreasing as reweighting w to look more like a gradient.

For a submodular f (not necessarily monotone) with a "down-closed solvable polytope" constraint, stopping at time $T = 1$ guarantees feasibility and has an approximation ratio of $1/e - o(1)$. If f is monotone, we get $1 - 1/e - o(1)$.

Note that this paper proves these approximation ratios in terms of $F(x)$ for the solution x we find, so for instance $F(x) \geq (1/e - o(1))f(OPT)$. One can then use contention resolution schemes, but for matroid constraints, it is better to use pipage rounding which suffers no loss at all. It turns out that pipage rounding works both for monotone and non-monotone f (due to Vondrák [2013]). So we get $f(S) \geq (1/e - o(1))f(OPT)$ and $f(S) \geq (1-1/e-o(1))f(OPT)$ respectively.

One interesting point is that with this method, we don't have to stop at time T, and the paper gives a lot of results for cases where we know something more about the polytope and can stop later to get a better solution.

References

Alexander A Ageev and Maxim I Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization*, 8(3):307–328, 2004.

Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011.

Moran Feldman, Joseph Naor, and Roy Schwartz. A unified continuous greedy algorithm for submodular maximization. In *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*, pages 570–579. IEEE, 2011.

Jan Vondrák. Optimal approximation for the submodular welfare problem in the value oracle model. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 67–74. ACM, 2008.

Jan Vondrák. Symmetry and approximability of submodular maximization problems. *SIAM Journal on Computing*, 42(1):265–304, 2013.

Jan Vondrák, Chandra Chekuri, and Rico Zenklusen. Submodular function maximization via the multilinear relaxation and contention resolution schemes. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 783–792. ACM, 2011.