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 \( F : [0,1]^N \rightarrow \mathbb{R} \), 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, \( F(x) = \sum_S f(S) \prod_{i \in S} x_i \prod_{i \not\in S} (1-x_i) \).

Last time, Eric covered the continuous greedy algorithm of Vondrak [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 pipeage 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, Vondrak [2008], Calinescu 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 \( f(S) \)).

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 approximately optimal set for \( f \).

For our problem, the polytope over which we maximize \( F \) is \( P_{\mathcal{I}} = \text{Conv}(\{x_S : S \in \mathcal{I}\}) \), 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 \( \alpha \)-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” \((x \in P \implies cx \in P \text{ for all } c \in [0,1])\) 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 \( \frac{1}{3} - \epsilon \approx 0.367 \) 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 \( \alpha \)-approximation algorithm for \( F \) and a monotone \( c \)-balanced CR scheme, we have an \( \alpha c \) approximation in expectation, since we have an output \( I \) satisfying

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

(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 pipeage rounding instead of CR schemes and lose nothing in the rounding step — this is due to the appendix of Vondrak [2013] and is mentioned in Feldman et al. [2011], where together with their \( \frac{1}{3} \) approximation for maximizing \( F \) it gives \( \frac{1}{3} \) for maximizing any nonnegative submodular \( f \) over a matroid.

---

1The 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 \( \pi \) for \( P_\mathcal{I} \) is a procedure that, for every \( x \in b P_\mathcal{I} = \{ b \cdot x : x \in P_\mathcal{I} \} \) and every \( A \subseteq 2^N \), returns a (random) set \( I = \pi_x(A) \) where, first, \( I \subseteq \mathcal{I} \) and \( I \subseteq A \cap \text{support}(x) \); and second, for each \( i \in \text{support}(x) \), we have \( \Pr[I \in \pi_x(R(x)) | i \in R(x)] \geq c \).

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 \( b P_\mathcal{I} \). 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 \( \pi \) is a monotone \((b, c)\)-balanced Bo-CR scheme, then for any \( x \in b P_\mathcal{I} \), letting the random output \( I = \pi_x(R(x)) \), we have \( \mathbb{E} f(I) \geq c F(x) \).

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

\[
\mathbb{E} f(I) = \mathbb{E}_{R \leftarrow R(x)} \sum_i \mathbb{E}_{I \leftarrow \pi_x(R)} [f(I_i) - f(I_{i-1})]
\]

\[
= \mathbb{E}_{R \leftarrow R(x)} \sum_{i \in R} \mathbb{E}_{I \leftarrow \pi_x(R)} [f(I_i) - f(I_{i-1})]
\]

\[
= \mathbb{E}_{R \leftarrow R(x)} \sum_{i \in R} \sum_{I \subseteq R: I \subseteq I} \Pr[I = \pi_x(R)(I)] (f_{I_{i-1}}(i))
\]

\[
\geq \mathbb{E}_{R \leftarrow R(x)} \sum_{i \in R} \sum_{I \subseteq R: I \subseteq I} \Pr[I = \pi_x(R)] (f_{R_{i-1}}(i))
\]

\[
\geq \mathbb{E}_{R \leftarrow R(x)} c (f_{R_{i-1}}(i))
\]

\[
= c \mathbb{E}_{R \leftarrow R(x)} f(R).
\]

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 \( c \).

Now let’s give the full definition:

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

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 \( \eta \) 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 \( \pi \) is a monotone \((b, c)\)-balanced CR scheme, then for any \( x \in b P_\mathcal{I} \), letting the random output \( I = \pi_x(R(x)) \), and \( J = \eta(I) \), we have \( \mathbb{E} f(J) \geq c F(x) \).

Some facts about composing or changing CR schemes:

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

The proof given by the authors is: define \( \pi'(A) \) 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 \( \mathcal{I} = \bigcap_{i=1}^N \mathcal{I}_i \) and each \( P_{\mathcal{I}_i} \) has a \((b, c_i)\)-CR scheme, then \( P_{\mathcal{I}} \) has a \((b, c)\)-CR scheme.

The idea is to let \( \pi_x(A) = \bigcap_i \pi_{i,x}(A) \). 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 \( \phi : 2^N \rightarrow \mathcal{I} \). Any randomized scheme can be written as a distribution \( \lambda \) over deterministic schemes, where with probability \( \lambda_\phi \) we pick scheme \( \phi \) and apply it. The key condition we need to satisfy is that \( \Pr[i \in \pi_x(R(x)) | i \in R(x)] \geq c \). Using that \( \pi_x \) can be written as a distribution \( \lambda \) over \( \phi \), the left side is \( \sum_\phi \lambda_\phi \Pr[i \in \phi(R(x)) | i \in R(x)] \). Thus the LP for maximizing \( c \) is

\[
\begin{align*}
\text{max } c \\
\text{s.t. } \sum_\phi \lambda_\phi \Pr[i \in \phi(R(x)) | i \in R(x)] & \geq c & \forall i \in [N] \\
\sum_\phi \lambda_\phi &= 1 \\
\lambda_\phi &\geq 0 & \forall \phi.
\end{align*}
\]


The dual is
\[
\min \mu \\
\text{s.t. } \sum_{i=1}^{N} z_i \Pr[i \in \phi(R(x)) \mid i \in R(x)] \leq \mu \quad \forall \phi \\
\sum_{i=1}^{N} z_i = 1 \\
z_i \geq 0 \quad \forall i.
\]

We can think of this as a two-player game where the column player is choosing a distribution $\lambda$ over $\phi$ and the row player is choosing a distribution $z$ over $i$. When each player draws from her distribution, we get a $\phi$ 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:
\[
\min \max_{z} \sum_{i=1}^{N} z_i \Pr[i \in \phi(R(x)) \mid i \in R(x)].
\]

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 $y_i = \frac{z_i}{x_i}$:
\[
\min_{y \geq 0} \max_{\sum_{i} x_i y_i = 1} \sum_{i=1}^{N} y_i \Pr[i \in \phi(R(x))] \\
= \min_{y \geq 0} \max_{\sum_{i} x_i y_i = 1} \sum_{i \in S} y_i E_{S \leftarrow \phi(R(x))}. \sum_{i \in S} y_i.
\]

The claim is that this is maximized by the $\phi$ 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 $r_y(T) = \max_{S \subseteq T, S \in \mathcal{I}} \sum_{i \in S} y_i$ to be the “weighted rank function”, or weight of the max-weight independent set for the problem. Then the optimal solution is
\[
\min_{y \geq 0} \sum_{x_i y_i = 1} E_{R \leftarrow R(x)} r_y(R) \\
= \min_{y \geq 0} \sum_{x_i y_i} E_{R \leftarrow R(x)} r_y(R) \\
= \inf_{x \in \mathcal{P}_2, y \geq 0} \frac{F(x)}{r_y^+(x)}.
\]

Now subject to a downward-closed constraint $\mathcal{I}$, define the correlation gap to be
\[
\inf_{x \in \mathcal{P}_2} \frac{F(x)}{r^+(x)}.
\]

Now in particular, if we consider the weighted rank function defined above, $r_y(T) = \max_{S \subseteq T, S \in \mathcal{I}} \sum_{i \in S} y_i$, then the correlation gap is
\[
\inf_{x \in \mathcal{P}_2, y \geq 0} \frac{E_{r_y(R(x))}}{r_y^+(x)}.
\]

But what is $r_y^+(x)$? Assuming that $x$ is in the polytope, $x = \sum_{S \subseteq \alpha} a_S 1_S$, where $1_S$ is the indicator vector for the set $S$. But if we take this distribution $\alpha$, 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.
We can also see this in the LP primal, as follows. Because $x \in P_I$, $x$ can be written as $\sum \alpha_S x_S$. Imagine that instead of drawing $R(x)$ by drawing each $i$ independently with probability $x_i$, we drew $R(x)$ from the correlated distribution $\alpha$ (which still has marginals given by $x$). Then the identity function $\phi$ would be valid because $R(x)$ would always be in $I$, and we would have $Pr[i \in \phi(R(x)) | i \in R(x)] = 1$ for that choice of $\phi$. 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 $\left(1, \frac{1-\frac{1}{e}}{b}\right)$-balanced contention resolution scheme for the associated polytope $P_I$.

The proof is essentially that, first, the greedy algorithm for maximal 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 $a_1, \ldots, a_n \in [0,1]$ and we have a single knapsack constraint, so the resulting feasible set is $F = \{S : \sum_{i \in S} a_i \leq 1\}$. Naturally it is downward closed, and nicely the polytope $P_F = \{x \in [0,1]^n : \sum a_i x_i \leq 1\}$.

**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 $x = \arg \max \{w(y) \cdot x : x \in P\}$ where $w(y)$ is a “discrete gradient”, $w(y)_x = F(y \land 1_x) - F(y)$. Then each coordinate $x_e$ is decreased to $x_e \cdot (1 - y_e)$. 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 Vondrak [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


