Subexponential algorithm for unique games

Claire Mathieu

March 11, 2011

Abstract

A summary of the Arora-Barak-Steurer 2010 subexponential algorithm for unique games – without proofs.

1 Problem and result

Here is the *unique games* problem. Fix an integer $p > 0$.
Input: A multigraph G, each edge being labeled by an equation. Edge $\{i, j\}$ is labeled by an equation of the form: $x_i - x_j = a \bmod p$.
Output: An assignment of values $\{x_1, \dots, x_n\} \to \{0, 1, \dots, p-1\}$.
Objective: Maximize the fraction of edges (equations) that are satisfied.
Here is the result.

THEOREM 1.1. *There is an algorithm for the unique games problem that takes as input an instance of value at least* $1 - \epsilon^6$ *and outputs an assignment with value at least* $1 - O(\epsilon \log(1/\epsilon))$*. The runtime is* $2^{n^{O(1/\epsilon)}}$.

The algorithm is in three steps: preprocessing, partition, exhaustive search.

2 Preprocessing

(This algorithm is sketched in Appendix A of the paper.)

1. Scale the input, duplicating every edge of G enough times so that the minimum degree is $\geq d_1$.
2. Create a new graph G' as follows:
   For each vertex v of G
   (a) Let d denote the degree of v. Replace v by d new vertices $v_0, v_1, \dots, v_{d-1}$, each connected to one of v's old neighbors (self-loops become un-looped).
   (b) Add new edges forming an expander graph over vertices $\{v_0, v_1, \dots, v_{d-1}\}$, regular of degree $d_0 \geq 9$, for example as follows: choose a random involution σ over dd_0, and if $\sigma(i) = j$ then add an edge from $v_{\lfloor i/d_0 \rfloor}$ to $v_{\lfloor j/d_0 \rfloor}$.
   (c) Add $d_0 + 1$ self-loops at each vertex.
3. All new edges are labeled by equality constraints, $x_i - x_j = 0 \bmod p$.

By $(G'_{i,j})$, we denote the scaled adjacency matrix of G' such that each row sums to 1 and each column sums to 1: G' is *stochastic*. Additionally, G' is *lazy*, meaning that $G'_{i,i} \geq 1/2$ for all i.

LEMMA 2.1. *(1) An assignment of G with value $1 - \epsilon$ yields an assignment of G' with value at least $1 - \epsilon/20$.*
(2) An assignment of G' with value $1 - \epsilon/10$ yields an assignment of G with value at least $1 - \epsilon$.
(3) G' is stochastic and lazy.

Henceforth, we will assume that the input graph is stochastic and lazy.

3 Small set expansion subroutine

The second step requires solving the following problem, called the *small set expansion* problem. The *expansion* of a set S of vertices is $|E(S, V-S)|/(2|E(S)| + |E(S, V-S)|)$. In other words, first pick a random vertex $i \in S$, then pick a random neighbor j of i: the expansion of S is the probability that j is in S.

Fix $\epsilon > 0$.
Input: A stochastic lazy multigraph G.
Output: A subset S of vertices with size at most $n^{1-\epsilon}$.
Objective: Minimize the expansion of S.
(This algorithm is implicit in the proof of Lemma B.3 in Appendix B of the paper.)

Small set expansion subroutine

1. For each number of steps $s = 1, 2, \dots, O(\log n)$,
   (a) Compute G^s
   (b) For each vertex i and for each threshold t,
   (c) Let $S = \{j : G^s_{i,j} > t\}$.
2. Output the S of minimum expansion, among all the ones with size less than or equal to $n^{1-\epsilon}$.

The analysis relies on a new definition. The *τ-threshold rank* of a stochastic graph G is the number of eigenvalues of the adjacency matrix that are greater than or equal to τ.

LEMMA 3.1. *If G is stochastic, lazy, and has $(1-\epsilon^5)$-threshold rank at least $n^{100\epsilon}$, then the small set expansion algorithm returns a set with expansion $O(\epsilon^2)$.*

4 Partition

(This algorithm is sketched in Section 4 of the paper.)

1. $\mathcal{S} = \{V\}$ and $\mathcal{P} = \emptyset$.
2. For $t = 1, 2, \dots (10/\epsilon)\log(1/\epsilon)$
   (a) For each set A in $\mathcal{S}$:
   While the $(1-\epsilon^5)$-threshold rank of A is greater than $n^{100\epsilon}$
      i. $S \leftarrow$ output of the **small set expansion subroutine**, executed on A. Add S to $\mathcal{S}'$.
      ii. $A \leftarrow A - S$. Add selfloops to A to make it stochastic
   Add A to $\mathcal{P}$.
   (b) $\mathcal{S} \leftarrow \mathcal{S}'$.
3. Output $\mathcal{P} \cup \mathcal{S}$.

LEMMA 4.1. *If G is stochastic and lazy, then the Partition algorithm outputs a partition of V such that the fraction of edges across parts is at most $1 - O(\epsilon\log(1/\epsilon))$, and such that each part has $(1-\epsilon^5)$-threshold rank less than $n^{100\epsilon}$.*

5 Non-expanding set enumeration subroutine

The third step requires solving the following subproblem, called the *non-expanding set enumeration* problem.
Input: A stochastic lazy multigraph G.
Output: A collection $\mathcal{C}$ of subsets of vertices.
Objective: For every subset S^* of V with expansion less than ϵ, some set $S \in \mathcal{C}$ should have small symmetric difference with S^*.
(This algorithm is sketched in Section 2.1 of the paper.)

Non-expanding-set enumeration subroutine.

1. Compute the subspace U spanned by eigenvectors with eigenvalues greater than or equal to $1-\eta$. Let U_1 denote the unit sphere of U.

2. Let N be a collection of points of U_1 such that $U_1 \subseteq \cup_{x \in N} B(x, \sqrt{\epsilon/\eta})$. For example, N could be a random subset of U_1 of size $O((\eta/\epsilon)^{r-1})$, where r denotes the dimension of U_1.

3. For $\delta = 1, 2, \ldots, n$, for each $x \in N$,
put in $\mathcal{C}$ the set $S = \{i : x_i \geq 1/(2\sqrt{\delta})\}$.

LEMMA 5.1. *Let $\mathcal{C}$ denote the output of the enumeration algorithm. Then, for every S^* of V with expansion less than ϵ, some set $S \in \mathcal{C}$ has symmetric difference with S_0 less than or equal to $8(\epsilon/\eta)|S^*|$. The runtime is $O(e^r poly(n))$, where r denotes the $(1-\eta)$-threshold rank of G.*

6 Exhaustive search

(This algorithm is presented in Section 5 of the paper.)

1. Let $\widehat{G}$ denote the graph obtained from G by replacing each vertex i by p vertices $i_0, i_1, \ldots, i_{p-1}$, and each edge $\{i, j\}$ with label $x_i - x_j = a \bmod p$ by p edges between i_k and $j_{k-a \bmod p}$.

2. Let $\mathcal{C} \leftarrow$ output of the **non-expanding-set enumeration subroutine** on $\widehat{G}$ with $\eta = \epsilon^5$.

3. For each $S \in \mathcal{C}$,

   Define an assignment of $V(G)$ as follows: For each $i \in V$, let $x_i = 0$ if S contains none of $\{i_0, i_1, \ldots, i_{p-1}\}$, and $x_i = k$ if k is minimum such that $i_k \in S$.

4. Ouput the best assignment

LEMMA 6.1. *Let G be stochastic, lazy, and with $(1-\epsilon^5)$-threshold rank less than $n^{100\epsilon}$. If the unique games on G has value at least $1-\epsilon$, then the above algorithm outputs an assignment with value at least $1 - 20\epsilon/\eta$.*

References

[1] *Subexponential Algorithms for Unique Games and Related Problems*, Sanjeev Arora, Boaz Barak and David Steurer, IEEE FOCS 2010.