Tensor Decomposition and Approximation Schemes for Constraint Satisfaction Problems *

W. Fernandez de la Vega † Ravi Kannan ‡

Marek Karpinski § Santosh Vempala ¶

Abstract

The only general class of MAX-rCSP problems for which Polynomial Time Approximation Schemes (PTAS) are known are the dense problems. In this paper, we give PTAS's for a much larger class of weighted MAX-rCSP problems which includes as special cases the dense problems and, for $r = 2$, all metric instances (where the weights satisfy the triangle inequality) and quasimetric instances; for $r > 2$, our class includes a generalization of metrics. Our algorithms are based on low-rank approximations with two novel features: (1) a method of approximating a tensor by the sum of a small number of "rank-1" tensors, akin to the traditional Singular Value Decomposition (this might be of independent interest) and (2) a natural and simple way of scaling the weights. Besides MAX-rCSP problems, we also give PTAS's for problems with a constant number of global constraints such as maximum weighted graph bisection and some generalizations.

*To appear in Proc. 37th ACM STOC (2005).

†LRI, Université Paris-Sud, Orsay, Paris. Supported in part by PROCOPE Project and by IST Grant 1496 (RAND-APX). Email **lalo@lri.lri.fr**

‡Computer Science, Yale, New Haven, CT 06520. Supported in part by NSF Grant CCR-9820850. Email **kannan@cs.yale.edu**

§Dept. of Computer Science, University of Bonn. Supported in part by DFG Grants, Max-Planck Research Prize, and by IST Grant 14036 (RAND-APX). Email **marek@cs.uni-bonn.de**

¶Mathematics, MIT, Cambridge, MA 02139. Supported by NSF award CCR-0307536 and a Sloan foundation fellowship. Email **vempala@math.mit.edu**

1 Introduction

The singular value decomposition is a useful tool in the design of efficient algorithms for a variety of problems (e.g., [FKV04, Mc01]). In this paper, motivated by boolean constraint satisfaction problems (CSP's) with r variables per constraint, we propose an extension of low-rank approximation to *tensors*, i.e., r-dimensional real arrays. We give an efficient algorithm for finding such an approximation and apply it to weighted MAX-rCSP, i.e., the problem of finding a boolean assignment that maximizes the total weight of satisfied constraints. As a consequence, for any MAX-rCSP that satisfies a certain density condition, we obtain a polynomial-time approximation scheme. In the past, there has been much progress on special cases; in particular, there are polynomial-time approximation schemes for *dense* unweighted problems [AKK95, F96, FK96, GGR96, FK00, AFKK02], and several cases of MAX-2CSP with metric weights including maxcut and partitioning [FK98, I99, FKKR03, FKK04]. We will show that our density condition captures all known special cases for which PTAS's exist as well as the metric MAX-2CSP (for which no PTAS was known before) and some natural generalizations.

A MAX-rCSP problem can be formulated as a problem of maximizing a homogeneous degree r polynomial in the variables $x_1, x_2, \dots x_n, (1-x_1), (1-x_2), \dots (1-x_n)$ (see e.g. [AFKK02].) Let

$$\mathbf{S} = \{y = (x_1, \dots x_n, (1-x_1), \dots (1-x_n)) : x_i \in \{0,1\}\}$$

be the solution set. Then the problem is

$$\mathrm{Max}_{y \in \mathbf{S}} \sum_{i_1, i_2, \dots i_r = 1}^{2n} A_{i_1, i_2, \dots i_r} y_{i_1} y_{i_2} \dots y_{i_r}.$$

where A is a given nonnegative symmetric r-dimensional array i.e.,

$$A_{i_1, i_2, \dots i_r} = A_{i_{\sigma(1)}, i_{\sigma(2)}, \dots i_{\sigma(r)}}$$

for any permutation σ. The entries of the r-dimensional array A can be viewed as the weights of an r-uniform hypergraph on $2n$ vertices. Throughout, we assume that r is fixed.

Our main tool to solve this problem is a generalization of low-rank matrix approximation. A rank-1 tensor is the *outer product* of r vectors

$x^{(1)}, \ldots x^{(r-1)}, x^{(r)}$, given by the r-dimensional array whose $(i_1, \ldots i_r)$'th entry is $x^{(1)}_{i_1} x^{(2)}_{i_2}, \ldots x^{(r)}_{i_r}$; it is denoted $x^{(1)} \otimes x^{(2)} \otimes \ldots x^{(r)}$. We will show the following:

1. For any r-dimensional array A, there exists a good approximation by the sum of a small number of rank-1 tensors.

2. We can algorithmically find such an approximation.

In the case of matrices, traditional Linear Algebra algorithms find good approximations. Indeed, we can find the *best* approximations under both the Frobenius and L_2 norms using the Singular Value Decomposition. Unfortunately, there is no such theory (or algorithm) for r-dimensional arrays when $r \geq 2$. Here, we will develop sampling-based algorithms for finding low-rank approximations which serve our purpose. These claims are formalized in the next lemma and theorem (see Section 3 for the definition of the generalized norms).

Lemma 1. *For any tensor A, and any $\epsilon > 0$, there exist $k \leq 1/\epsilon^2$ rank-1 tensors, $B_1, B_2, \ldots B_k$ such that*

$$||A - (B_1 + B_2 + \ldots B_k)||_2 \leq \epsilon ||A||_F.$$

Theorem 2. *For any tensor A, and any $\epsilon > 0$, we can find k rank-1 tensors $B_1, B_2, \ldots B_k$, where $k \leq 4/\epsilon^2$, in time $(n/\epsilon)^{O(1/\epsilon^4)}$ such that with high probability at least* $3/4$ *we have*

$$||A - (B_1 + B_2 + \ldots B_k)||_2 \leq \epsilon ||A||_F.$$

The proofs and the algorithm for low-rank tensor approximation are given in Section 3. For $r = 2$, the running time is a fixed polynomial in n and exponential only in ϵ.

Next, we give a density condition so that if a MAX-rCSP viewed as a weighted r-uniform hypergraph satisfies this condition, then there is a PTAS for the problem. This condition provides a unified framework for a large class of weighted MAX-rCSP's.

Define the node weights $D_1, \ldots, D_n$ of A and their average as

$$D_i = \sum_{i_2, i_3, \ldots i_r \in V} A_{i_1, i_2, \ldots i} \qquad \bar{D} = \frac{1}{2n} \sum_{i=1}^{n} D_i.$$

Note that when $r = 2$ and A is the adjacency matrix of a graph, the D_i are the degrees of the vertices and $\bar{D}$ is the average degree.

Definition 1. *The* core-strength *of a weighted r-uniform hypergraph given by an r-dimensional tensor A is*

$$\left(\sum_{i=1}^{2n} D_i\right)^{r-2} \sum_{i_1,i_2,\ldots,i_r} \frac{A^2_{i_1,\ldots,i_r}}{\prod_{j=1}^r (D_{i_j} + \bar{D})}$$

We say that a class of weighted hypergraphs (MAX-rCSP's) is *core-dense* if the core-strength is $O(1)$ (i.e., independent of A, n).

To motivate the definition, first suppose the class consists of unweighted hypergraphs. Then if a hypergraph in the class has E as the edge set with m edges, the condition says that

$$m^{r-2} \sum_{(i_1,\ldots,i_r)\in E} \frac{1}{\prod_{j=1}^r (D_{i_j} + \bar{D})} = O(1). \tag{1}$$

Note that here the D_i's are the degrees of the hypergraph vertices in the usual sense of the number of edges incident to the vertex. It is easy to see this condition is satisfied for dense hypergraphs, i.e., for $r-$ uniform hypergraphs with $\Omega(n^r)$ edges, because in this case, $\bar{D} \in \Omega(n^{r-1})$.

The condition can be specialized to the case $r = 2$, where it says that

$$\sum_{i,j} \frac{A^2_{ij}}{(D_i + \bar{D})(D_j + \bar{D})} = O(1). \tag{2}$$

We will show that all metrics satisfy this condition. Also, so do *quasimetrics*. These are weights that satisfy the triangle inequality up to a constant factor (e.g., powers of a metric) and arise in clustering applications [FKKR03, SS73, FK00]. So, as a special case of our main result, we get PTAS's for metrics and quasimetrics. (While PTAS's were known for the dense case, they were not known previously for the metric case.) Our main algorithmic result is the following.

Theorem 3. *There is a PTAS for any core-dense weighted MAX-rCSP.*

The algorithm and proof are given in Section 4. We will also show (in Section 5) that a generalization of the notion of metric for higher r also satisfies our core-dense condition.

Theorem 4. *Suppose for a MAX-rCSP, the tensor A satisfies the following local density condition:*

$$\forall\, i_1, \ldots, i_r, \quad A_{i_1,\ldots,i_r} \le \frac{c}{n^{r-1}} \sum_{j=1}^{r} D_{i_j}$$

where c is a constant. Then there is a PTAS for the MAX-rCSP defined by A.

The condition in the theorem says that no entry of A is "wild" in that it is at most a constant times the average entry in the r "planes" passing through the entry. The reason for calling such tensors "metric tensors" will become clear when we show in Section 5 that for $r = 2$, metrics do indeed satisfy this condition. When the matrix A is the adjacency matrix of a graph, then the condition says that for any edge, one of its end points must have degree $\Omega(n)$. This is like the "everywhere" dense condition in [AKK95]. Theorem 4 has the following corollary for "quasi-metrics", where the triangle inequality is only satisfied within constant factors - $A_{ik} \le c(A_{ij} + A_{jk})$.

Corollary 5. *There exists a PTAS for metric and quasimetric instances of MAX-CSP.*

2 The 2-dimensional case

In this section, we prove Theorem 3 in the case $r = 2$. This case already contains the idea of scaling which we will use for the case of higher r. But, as mentioned earlier, this case does not need special algorithms for finding low-rank approximations - they are already available from Linear Algebra.

Recall that we want to find

$$\mathrm{Max}_{y \in \mathbf{S}} A_{ij} y_i y_j = y^T A y,$$

where $\mathbf{S} = \{y = (x_1, x_2, \ldots x_n, (1 - x_1), (1 - x_2), \ldots (1 - x_n)), x_i \in \{0, 1\}\}$ is the solution set. We will describe in this section an algorithm to solve this problem to within additive error $O(\epsilon n \bar{D})$, under the assumption that that the core-strength of A is at most a constant c. The algorithm will run in time polynomial in n for each fixed $\epsilon > 0$. Note that $\mathrm{Max}_{y \in \mathbf{S}} y^T A y \ge E(y^T A y) = \frac{1}{2} n \bar{D}$, where E denotes expectation over uniform random choice of $x \in \{0, 1\}^n$. Thus, this will prove Theorem (3) for this case (of $r = 2$).

The algorithm first scales the matrix A to get a matrix B given by :

$$B = D^{-1}AD^{-1}$$

where, D is the diagonal matrix with $D_{ii} = \sqrt{D_i + \bar{D}}$. The scaling $B_{ij} = \frac{A_{ij}}{\sqrt{D_i}\sqrt{D_j}}$ is very natural and has been used in other contexts (for example when A is the transition matrix of a Markov Chain). This scaling unfortunately scales up "small degree" nodes too much for our purpose and so we use the modified scaling given here; we will see that while the addition of $\bar{D}$ does not increase the error in our approximation algorithms, it helps by modulating the scaling up of low degree nodes. Clearly,

Claim 1. *$||B||_F^2$ is the core-strength of the matrix A.*

By carrying out the standard Singular Value Decomposition (SVD) of the matrix B, we can find in polynomial-time, for any $\epsilon > 0$, a matrix $\hat{B}$ of rank $l \leq 4/\epsilon^2$ such that

$$||B - \hat{B}||_2 \leq \frac{\epsilon}{2}||B||_F.$$

In fact, as shown in [FKV04], such a matrix $\hat{B}$ can be computed in linear in n time with ϵ twice as large. We now let

$$\hat{A} = D\hat{B}D.$$

Note that the rank of $\hat{A}$ equals the rank of $\hat{B}$. We then solve the following problem approximately to within additive error $O(\epsilon n\bar{D})$.

$$\max_{y \in \mathbf{S}} y^T \hat{A} y \tag{3}$$

We will show how to do this approximate optimization presently. First, we analyze the error caused by replacing A by $\hat{A}$:

$$\begin{aligned}
\mathrm{Max}_{y \in \mathbf{S}} |y^T(A - \hat{A})y| &= \mathrm{Max}_{y \in \mathbf{S}} |y^T D(B - \hat{B})Dy| \\
&\leq \mathrm{Max}_{y \in \mathbf{S}} |Dy|^2 ||B - \hat{B}||_2 \\
&\leq \epsilon \sum_i (D_i + \bar{D})||B||_F \\
&\leq 4\epsilon n\bar{D}(\text{core-strength of } A)^{1/2},
\end{aligned}$$

the last because of Claim 1 and the fact that $\sum_i D_i = 2n\bar{D}$.

Now for solving the non-linear optimization problem (3), we proceed as follows : suppose the SVD of $\hat{B}$ expressed $\hat{B}$ as $U\Sigma V$, where the U is a $2n \times l$ matrix with orthonormal columns, Σ is a $l \times l$ diagonal matrix with the singular values of $\hat{B}$ and V is a $l \times 2n$ matrix with orthonormal rows. Now we write

$$y^T \hat{A} y = (y^T DU)\Sigma(VDy) = u^T \Sigma v$$
$$\text{where, } u^T = y^T DU \quad \text{and} \quad v = VDy$$

are two $l-$ vectors. This implies that there are really only $2l$ "variables" - u_i, v_i in the problem (and not the n variables - $y_1, y_2, \dots y_n$). This is the idea we will exploit. Note that for $y \in \mathbf{S}$, we have (since U, V have orthonormal columns, rows reaspectively)

$$|u|^2 \leq |y^T D|^2 \leq \sum_i (D_i + \bar{D}) \leq 4n\bar{D}.$$

Similarly, $|v|^2 \leq 4n\bar{D}$. So letting

$$\alpha = \sqrt{n\bar{D}},$$

we see that the the vectors u, v live in the rectangle

$$R = \{(u, v) : -2\alpha \leq u_i, v_j \leq +2\alpha\}.$$

Also, the gradient of the function $u^T \Sigma v$ with respect to u is Σv and with respect to v is $u^T \Sigma$; in either case, the length of the gradient vector is at most $2\alpha\sigma_1(\hat{B}) \leq 2\alpha\sqrt{c}$. We now divide up R into small cubes; each small cube will have side

$$\eta = \frac{\epsilon\alpha}{20\sqrt{l}},$$

and so there will be $\epsilon^{-O(l)}$ small cubes. The function $u^T \Sigma v$ does not vary by more than $\epsilon n\bar{D}\sqrt{c}/10$ over any small cube. Thus we can solve (3) by just enumerating all the small cubes in R and for each determining whether it is feasible (i.e., whether there exists a 0-1 vector x such that for some (u, v) in this small cube, we have $u^T = y^T Du, v = VDy$, for $y = (x, \underline{1} - x)$.)

For each small cube C in R, this is easily formulated as an integer program in the n 0,1 variables $x_1, x_2, \dots x_n$ with $4l$ constraints (arising from the upper and lower bounds on the coordinates of u, v which ensure that (u, v) is in the small cube.)

For a technical reason, we have to define a D_i to be "exceptional" if $D_i \geq \epsilon^6 n\bar{D}/10^6$; also call an i exceptional if either D_i or D_{i+n} is exceptional. Clearly, the number of exceptional D_i is at most $2 \times 10^6/\epsilon^6$ and we can easily identify them. We enumerate all possible sets of $2^{O(1/\epsilon^6)}$ 0,1 values of the exceptional x_i and for each of these set of values, we have an Integer Program again, but now only on the non-exceptional variables.

We consider the Linear Programming (LP) relaxation of each of these Integer Programs obtained by relaxing $x_i \in \{0, 1\}$ to $0 \leq x_i \leq 1$. If one of these LP's has a feasible solution, then, it has a basic feasible solution with at most $4l$ fractional variables, Rounding all these fractional variables to 0 changes Dy by a vector of length at most

$$\sqrt{4l\epsilon^6 n\bar{D}/10^6} \leq \eta.$$

Thus, the rounded integer vector y gives us a (u, v) in the small cube C enlarged (about its center) by a factor of 2 (which we call $2C$). Conversely, if none of these LP's has a feasible solution, then clearly neither do the corresponding Integer Programs and so the small cube C is infeasible. Thus, for each small cube C, we find (i) either C is infeasible or (ii) $2C$ is feasible. Note that $u^T \Sigma v$ varies by at most $\epsilon n\bar{D}/5$ over $2C$. So, it is clear that returning the maximum value of $u^T \Sigma v$ over all centers of small cubes for which (ii) holds suffices. This is what the algorithm does.

Remark We could have carried this out with any "scaling'. The current choice turns out to be useful for the two important special cases here. Note that we are able to add the $\bar{D}$ almost "for free" since we have $\sum_i D_i + \bar{D} \leq 2\sum D_i$.

2.1 Maximum Weighted Bisection and other problems

The maximum weighted bisection problem in an undirected graph is to split the vertices into equal parts so as to maximize the total weight of edges from one part to the other. We will show that this problem has a PTAS for the case of core-dense weights. In fact, we will show something more general : consider a family of problems of the form :

$$\text{Max}_{y \in \mathbf{S}} y^T A y \text{ subject to } Cx \leq d \qquad x_i \in \{0, 1\},$$

where

(i) the number of constraints in $Cx \leq d$ is $O(1)$,

(ii) for every solution of $Cx \leq d \quad ; \quad 0 \leq x_i \leq 1$, we can round only the fractional valued variables to integer values to get a solution to $Cx \leq d \quad , \quad x_i\{0,1\}$ and

(iii) the family has a core-dense weights matrix (A).

Our result is that any such family admits a PTAS. The argument proceeds the same way as when there are no "side-constraints" $Cx \leq d$. But we note that using (i), there are still only $O(l)$ fractional variables in a basic feasible solution of every LP. By (ii), we can round them to produce an integral solution with the same error bounds (within constant factors) as we get for the problem with no side-constraints.

Note that for the maximum weighted bisection problem, $Cx \leq d$ has just two constraints - $\sum_i x_i \leq n/2$ and $\sum_i x_i \geq n/2$ and (ii) is easily seen to be valid. Indeed, more generally, we may also have node weights and require that we split into two parts of equal node weight, as long as (ii) is valid. More generally, we can also require some $O(1)$ subsets of vertices must all be bisected etc.

3 Fast tensor approximation via sampling

Corresponding to A, there is an r-linear form which for a set of r vectors $x^{(1)}, x^{(2)}, \ldots x^{(r-1)}, x^{(r)}$, is defined as

$$A(x^{(1)}, x^{(2)}, \ldots x^{(r)}) = \sum_{i_1, i_2, \ldots i_r} A_{i_1, i_2, \ldots i_{r-1}, i_r} x^{(1)}_{i_1} x^{(2)}_{i_2}, \ldots x^{(r)}_{i_r}.$$

We will use the following two norms of r-dimensional arrays corresponding to the Frobenius norm and L_2 norm for matrices.

$$\begin{aligned} ||A||_F &= \left(\sum A^2_{i_1, i_2, \ldots i_r}\right)^{\frac{1}{2}} \\ ||A||_2 &= \max_{x^{(1)}, x^{(2)}, \ldots x^{(r)}} \frac{A(x^{(1)}, x^{(2)}, \ldots x^{(r-1)}, x^{(r)})}{|x^{(1)}||x^{(2)}| \ldots}. \end{aligned}$$

We begin with a proof of Lemma 1 about the existence of a low-rank tensor decomposition.

Proof. If $||A||_2 \leq \epsilon||A||_F$, then we are done. If not, there are vectors $x^{(1)}, x^{(2)}, \ldots, x^{(r)}$, all of length 1 such that

$$A(x^{(1)}, x^{(2)}, \ldots, x^{(r)}) \geq \epsilon||A||_F.$$

Now consider the $r-$dimensional array

$$B = A - (A(x^{(1)}, x^{(2)}, \ldots, x^{(r)}))x^{(1)} \otimes x^{(2)} \otimes \ldots x^{(r)}.$$

It is easy to see that

$$||B||_F^2 = ||A||_F^2 - (A(x, y, z, \ldots)^2).$$

We can repeat on B and clearly this process will only go on for at most $1/\epsilon^2$ steps. $\square$

From the proof of Lemma 1, it suffices to find $x^{(1)}, x^{(2)}, \ldots, x^{(r)}$ all of length 1, maximizing $A(x^{(1)}, x^{(2)}, \ldots, x^{(r)})$ to within *additive error* $\epsilon||A||_F/2$. We will give an algorithm to solve this problem. We need a bit more notation. For any $r-1$ vectors $x^{(1)}, x^{(2)}, \ldots x^{(r-1)}$, we define $A(x^{(1)}, x^{(2)}, \ldots x^{(r-1)}, \cdot)$ as the vector whose i'th component is

$$\sum_{i_1, i_2, \ldots i_{r-1}} A_{i_1, i_2, \ldots i_{r-1}, i} x^{(1)}_{i_1} x^{(2)}_{i_2}, \ldots x^{(r-1)}_{i_{r-1}}.$$

Tensor decomposition

Set $\eta = \epsilon^2/100r\sqrt{n}$ and $s = 10^5 r/\epsilon^2$.

1. Pick s random $(r-1)$-tuples $(i_1, i_2, \ldots i_{r-1})$ with probabilities proportional to the sum of squared entries on the line defined by it:

$$p(i_1, i_2, \ldots i_{r-1}) = \frac{\sum_i A^2_{i_1, i_2, \ldots i_{r-1}, i}}{||A||_F^2}.$$

   Let I be the set of s $r-1$ tuples picked.

2. For each $i_1, i_2, \ldots i_{r-1} \in I$, enumerate all possible values of $z^{(1)}_{i_1}, z^{(2)}_{i_2}, \ldots z^{(r-1)}_{i_{r-1}}$ whose coordinates are in the set

$$J = \{-1, -1+\eta, -1+2\eta, \ldots 0, \ldots 1-\eta, 1\}^{s(r-1)}.$$

   (a) For each set of $\hat{z}^{(t)}$, for each $i \in V_r$, compute

$$y_i = \sum_{(i_1, \ldots i_{r-1}) \in I} A(i_1, \ldots i_{r-1}, i) \hat{z}^{(1)}_{i_1} \ldots \hat{z}^{(r-1)}_{i_{r-1}}.$$

   and normalize the resulting vector y to be a unit vector (a candidate for $z^{(r)}$).

   (b) Consider the $(r-1)$-dimensional array $A(y)$ defined by

$$(A(y))_{i_1, i_2, \ldots i_{r-1}} = \sum_i A_{i_1, i_2, i_3 \ldots i_{r-1}, i} \quad y_i$$

   and apply the algorithm recursively to find the optimum

$$A(y)(x^{(1)}, x^{(2)}, \ldots x^{(r-1)})$$

   with $|x^{(1)}| = \ldots |x^{(r-1)}| = 1$ to within additive error $\epsilon ||A(y)||_F/2$. (Note that $||A(y)||_F \leq ||A||_F$ by Cauchy-Schwartz).

3. Output the set of vectors that given the maximum among all these candidates.

Here is the idea behind the algorithm. Suppose $z^{(1)}, z^{(2)}, \ldots z^{(r)}$ are the (unknown) unit vectors that maximize $A(x^{(1)}, x^{(2)}, \ldots)$. Since

$$A(z^{(1)}, \ldots z^{(r-1)}, z^{(r)}) = z^{(r)} \cdot A(z^{(1)}, \ldots z^{(r-1)}, \cdot),$$

we have

$$z^{(r)} = \frac{A(z^{(1)}, z^{(2)}, \ldots z^{(r-1)}, \cdot)}{|A(z^{(1)}, z^{(2)}, \ldots z^{(r-1)}, \cdot)|}.$$

Thus, if we had $z^{(1)}, z^{(2)}, \ldots z^{(r-1)}$, then we could find $z^{(r)}$. In fact, we can estimate the components of $z^{(r)}$ if we had sufficiently many random terms in the sum $A(z^{(1)}, \ldots z^{(r-1)}, \cdot)$. It turns out that we need only $s = O(1/\epsilon^2)$ terms for a good estimate. Now we do not need to know the $z^{(1)}, z^{(2)}, \ldots, z^{(r-1)}$ completely; only $s(r-1)$ of their coordinates in total are needed for the estimate. We enumerate all possibilities for the values of these coordinates (in steps of a certain size) and one of the sets of coordinates we enumerate will correspond to the optimal $z^{(1)}, z^{(2)}, \ldots z^{(r-1)}$, whence we get the an estimate of $z^{(r)}$. For each candidate $z^{(r)}$, we can reduce the problem to maximizing an $(r-1)$-dimensional tensor and we solve this recursively.

We will now analyze the algorithm and consequently prove Theorem 2. We begin by showing the discretization does not cause any signicant loss.

Lemma 6. *Let $z^{(1)}, z^{(2)}, \ldots z^{(r-1)}$ be the optimal unit vectors. Suppose $w^{(1)}, w^{(2)}, \ldots w^{(r-1)}$ are obtained from the $z^{(t)}$'s by rounding each coordinate down to the nearest integer multiple of η. Then,*

$$\left|A(z^{(1)}, \ldots z^{(r-1)}, \cdot) - A(w^{(1)}, \ldots w^{(r-1)}, \cdot)\right| \leq \frac{\epsilon^2}{100}||A||_F.$$

Proof. We may write

$$\begin{aligned} &\left|A(z^{(1)}, z^{(2)}, \ldots z^{(r-1)}, \cdot) - A(w^{(1)}, w^{(2)}, \ldots w^{(r-1)}, \cdot)\right| \\ &\leq \left|A(z^{(1)}, z^{(2)}, \ldots z^{(r-1)}, \cdot) - A(w^{(1)}, z^{(2)}, \ldots z^{(r-1)}, \cdot)\right| + \\ &\left|A(w^{(1)}, z^{(2)}, \ldots z^{(r-1)}, \cdot) - A(w^{(1)}, w^{(2)}, z^{(3)}, \ldots z^{(r-1)}, \cdot)\right| \ldots \end{aligned}$$

A typical term above is

$$\begin{aligned} &|A(w^{(1)}, \ldots w^{(t)}, z^{(t+1)}, \ldots z^{(r-1)}, \cdot) \\ &-A(w^{(1)}, \ldots w^{(t)}, w^{(t+1)}, z^{(t+2)}, \ldots z^{(r-1)}, \cdot)| \\ &\leq \left|B(z^{(t+1)} - w^{(t+1)})\right| \\ &\leq ||B||_2 |z^{(t+1)} - w^{(t+1)}| \\ &\leq ||B||_F \eta\sqrt{n} \leq ||A||_F \eta\sqrt{n}. \end{aligned}$$

Here, B is the matrix defined as the matrix whose ij'th entry is

$$\sum_{j_1,\ldots j_t,j_{t+2}\cdots j_{r-1}} A_{j_1,\ldots j_t,i,j_{t+2},\ldots j_{r-1},j} w_{j_1}^{(1)} \ldots w_{j_t}^{(t)} z_{j_{t+2}}^{(t+2)} \ldots z_{j_{r-1}}^{(r-1)}$$

The claim follows. $\square$

Next, we analyze the error incurred by sampling.

Consider an $(r-1)$-tuple $(i_1, i_2, \ldots i_{r-1}) \in I$ and define the random variables variables X_i for i by

$$X_i = \frac{A_{i_1,i_2,\ldots i_{r-1},i} w_{1_1}^{(1)} w_{i_2}^{(2)} \ldots w_{i_{r-1}}^{(r-1)}}{p(i_1, i_2, \ldots i_{r-1})}.$$

It follows that

$$E(X_i) = A(w^{(1)}, w^{(2)} \ldots w^{(r-1)}, \cdot)_i.$$

We estimate the variance:

$$\begin{aligned}
\sum_i \mathrm{Var}(X_i) &\leq \sum_i \sum_{i_1,i_2,\ldots} \frac{A^2_{i_1,i_2,\ldots i_{r-1},i}(w_{i_1}^{(1)} \ldots w_{i_{r-1}}^{(r-1)})^2}{p(i_1, i_2, \ldots)} \\
&\leq \sum_{i_1,i_2,\ldots} \frac{(z_{i_1}^{(1)} \ldots z_{i_{r-1}}^{(r-1)})^2}{p(i_1, i_2, \ldots)} \sum_i A^2_{i_1,i_2,\ldots i_{r-1},i} \\
&\leq ||A||_F^2.
\end{aligned}$$

Consider the y_i computed by the algorithm when all $\hat{z}_{i_t}^{(t)}$ are set to $w_{i_t}^{(t)}$. This will clearly happen sometime during the enumeration. This y_i is just the sum of s i.i.d. copies of X_i, one for each element of I. Thus, we have that

$$E(y) = sA(w^{(1)}, w^{(2)} \ldots w^{(r-1)}, \cdot)$$

and

$$\mathrm{Var}(y) = E(|y - E(y)|^2) \leq s||A||_F^2.$$

We will sketch the rest of the argument. Define

$$\zeta = A(z^{(1)}, z^{(2)}, \ldots z^{(r-1)}, \cdot) \quad \text{and} \quad \Delta = y - s\zeta.$$

From the above, it follows that with probability at least $1-(1/10r)$, we have

$$|\Delta| \leq 10r\sqrt{s}||A||_F.$$

Using this,

$$\begin{aligned}
\left|\frac{y}{|y|} - \frac{\zeta}{|\zeta|}\right| &= \frac{|(y|\zeta| - \zeta|y|)|}{|y||\zeta|} \\
&= \frac{1}{|y||\zeta|}\left|(\Delta + s\zeta)|\zeta| - \zeta(|y| - s|\zeta| + s|\zeta|)\right| \\
&\leq \frac{2|\Delta|}{(s|y|)} \leq \frac{\epsilon}{50},
\end{aligned}$$

assuming $|y| \geq \epsilon||A||_F/100$. If this assumption does not hold, we know that the $|\zeta| \leq \epsilon||A||_F/20$ and in this case, the all-zero tensor is a good approximation to the optimum. From this, it can be shown that

$$||A(\frac{y}{|y|}) - A(\frac{\zeta}{|\zeta|})||_F \leq \frac{\epsilon}{10}||A||_F.$$

Thus, for any $r-1$ unit length vectors $a^{(1)}, a^{(2)}, \ldots a^{(r-1)}$, we have

$$\left|A(a^{(1)}, \ldots a^{(r-1)}, \frac{y}{|y|}) - A(a^{(1)}, \ldots a^{(r-1)}, \frac{\zeta}{|\zeta|})\right| \leq \frac{\epsilon}{10}||A||_F.$$

This implies that the optimal set of vectors for $A(y/|y|)$ are nearly optimal for $A(\zeta/|\zeta|)$. Since $z^{(r)} = \zeta/|\zeta|$, the optimal vectors for the latter problem are $z^{(1)}, \ldots, z^{(r-1)}$.

The running time of algorithm is dominated by the number of candidates we enumerate, and is

$$\mathrm{poly}(n)\left(\frac{1}{\eta}\right)^{s^2 r} = \left(\frac{n}{\epsilon}\right)^{O(1/\epsilon^4)}.$$

4 Approximation schemes for core-dense MAX-rCSP's

In this section, we give a PTAS for core-dense weighted MAX-rCSP's proving Theorem 3. For this, we now only need to describe the scaling (which is a direct generalization of the case $r = 2$) and how to optimize in the case where the coefficient tensor is the sum of a small number of rank-1 tensors. First we describe the scaling.

We wish to solve the problem

$$\max_{y \in \mathbf{S}} A(y, y, \ldots, y).$$

The algorithm first scales the entries of A to get an r-dimensional tensor B, as follows :

$$B_{i_1,\ldots,i_r} = \frac{A_{i_1,\ldots,i_r}}{\prod_{j=1}^r \alpha_{i_j}}$$

where $\alpha = (\alpha_1, \ldots, \alpha_n) \in \boldsymbol{R}^n$ is defined by $\alpha_j = \sqrt{\bar{D} + D_j}$.

Note that again for any $y \in \mathbf{S}$, using the substitution, $z_j = y_j \alpha_j$, we get

$$A(y, \ldots, y) = B(z, \ldots, z).$$

Then, applying the sampling algorithm from Section 3 to get a tensor $\hat{B}$ of rank at most k satisfying

$$||B - \hat{B}||_2 \leq \frac{\epsilon}{2} ||B||_F.$$

We then solve the following problem approximately to within additive error $\epsilon |\alpha|^r ||B||_F / 2$.

$$\max_{z : y_j \in \mathbf{S}_1} \hat{B}(z, z, \ldots, z).$$

The error of approximating B by $\hat{B}$ is bounded by

$$\begin{aligned}
&\max_{z \in \mathbf{S}_1} |(B - \hat{B})(z, \ldots, z)| \\
&\leq \max_{z : |z| \leq |\alpha|\}} |(B - \hat{B})(z, \ldots, z)| \\
&\leq |\alpha|^r ||B - \hat{B}||_2 \\
&\leq \epsilon |\alpha|^r ||B||_F \\
&\leq \epsilon (\sum_{i=1}^n (\bar{D} + D_i))^{r/2} \left(\sum_{i_1,\ldots,i_r} \frac{A^2_{i_1,\ldots,i_r}}{\prod_{j=1}^r D_{i_j}} \right)^{1/2} \\
&\leq \epsilon 2^{r/2} c (\sum_{i=1}^n D_i)
\end{aligned}$$

where c is the bound on the core-strength, noting that $\sum_i (\bar{D} + D_i) = 2 \sum_i D_i$.

4.1 Optimizing constant-rank tensors

From the above, it now suffices to deal with the case of a tensor A of constant rank. Let us see that this can be done very similarly as in the 2-dimensional case.

Let A be a tensor of dimension r and rank ℓ, say:

$$A = \sum_{1 \le j \le \ell} A^{(j)}$$

with

$$A^{(j)} = a_j x^{(j,1)} \otimes x^{(j,2)} ... \otimes x^{(j,r)}$$

where the $x^{(j,i)} \in \mathbf{R}^{2n}$ are length one vectors and moreover we have that $||A^{(j))}||_F \le ||A||_F$ and $\ell = O(\epsilon^{-2})$. We want to maximize approximately $A(y)$, i.e., we want to maximize approximately the sum $\sum_{j=1}^{\ell} A^{(j)}(y)$ where

$$A^{(j)}(y) = a_j \prod_{k=1}^{r} \left(x^{(j,k)} \cdot y \right)$$

over the set of vectors y satisfying $y_i \in \{0, \alpha_i\}, y_{n+i} = \alpha_i - y_i, \ 1 \le i \le n$ where α is a given n-dimensional vector. Let us define the vectors $z^{(j,k)}$ by

$$z_i^{(j,k)} = \alpha_i x_i^{(j,k)} \qquad\qquad z_{n+i}^{(j,k)} = \alpha_i x_{n+i}^{(j,k)}$$

for $1 \le i \le n$. If we define the tensor $\hat{A}$ by

$$\hat{A} = \sum_{j=1}^{\ell} \hat{A}^{(j)}$$

with

$$\hat{A}^{(j)} = a_j z^{(j,1)} \otimes z^{(j,2)} ... \otimes z^{(j,r)}, j = 1, 2, ..., \ell,$$

then we can equivalently maximize approximately

$$\hat{A}(y) = \sum_{j=1}^{\ell} a_j \prod_{k=1}^{r} \left(z^{(j,k)} \cdot y \right) \tag{4}$$

for y now in $\mathbf{S}$.

Thus, similarly as in the 2-dimensional case, $\hat{A}(y)$ depends really only on the ℓr variables $u_{j,i}$, say, where $u_{j,i} = z^{(j,i)} \cdot y, j = 1, 2, ..., \ell, i = 1, 2, ..., r$, and the values of each of these products are confined to the interval $[-2|\alpha|, +2|\alpha|]$. Then, exactly similarly as in the 2-dimensional case, we can get, in polynomial time, approximate values within $\epsilon|\alpha|^r$ for each of the optimal $u_{j,i} = z^{(j,i)} \cdot y$. (Actually, we enumerate a certain set of values of the vector $u_{j,i} \in \mathbf{R}^{r\ell}$ which need contain this nearly optimal value.) Inserting then these values in (4) gives an approximation of $\max \hat{A}(y)$ with additive error $O\left(\epsilon|\alpha|^r ||A||_F\right)$.

5 Metric tensors

Lemma 7. *Let A be an r-dimensional tensor satisfying the following local density condition:*

$$\forall\, i_1, \ldots, i_r \in V, \quad A_{i_1,\ldots,i_r} \leq \frac{c}{rn^{r-1}} \sum_{j=1}^{r} D_{i_j}$$

where c is a constant. Then A is a core-dense hypergraph with core-strength c.

Proof. We need to bound the core-strength of A. To this end,

$$\begin{aligned}
&\sum_{i_1,i_2,\ldots,i_r \in V} \frac{A^2_{i_1,\ldots,i_r}}{\prod_{j=1}^{r}(D_{i_j} + \bar{D})} \\
&\leq \frac{c}{rn^{r-1}} \sum_{i_1,i_2,\ldots,i_r \in V} \frac{A_{i_1,\ldots,i_r} \sum_{j=1}^{r} D_{i_j}}{\prod_{j=1}^{r}(D_{i_j} + \bar{D})} \\
&\leq \frac{c}{rn^{r-1}} \sum_{i_1,i_2,\ldots,i_r \in V} A_{i_1,\ldots,i_r} \sum_{j=1}^{r} \frac{1}{\prod_{k \in \{1,\ldots,r\} \setminus j}(D_{i_k} + \bar{D})} \\
&\leq \frac{c}{rn^{r-1}} \left(\sum_{i_1,i_2,\ldots,i_r \in E} A_{i_1,\ldots,i_r} \right) \frac{r}{\bar{D}^{r-1}} \\
&= \frac{c}{(\sum_{i=1}^{n} D_i)^{r-2}}.
\end{aligned}$$

Thus, the core-strength is at most

$$(\sum_{i=1}^{n} D_i)^{r-2} \sum_{i_1,i_2,\ldots,i_r \in E} \frac{A^2_{i_1,\ldots,i_r}}{\Pi_{j=1}^{r}(D_{i_j} + \bar{D})} \leq c.$$

□

Theorem 4 follows directly from Lemma 7 and Theorem 3. We next prove Corollary 5 for metrics.

Proof. (of Corollary 5) For $r = 2$, the condition of Theorem 4 says that for any $i, j \in V$,

$$A_{i,j} \leq \frac{c}{2n}(D_i + D_j).$$

We will verify that this holds for a metric MAX-2CSP with $c = 2$. When the entries of A form a metric, for any i, j, k, we have

$$A_{i,j} \leq A_{i,k} + A_{k,j}$$

and so

$$\begin{aligned} A_{i,j} &\leq \frac{1}{n}\left(\sum_{k=1}^{n} A_{i,k} + \sum_{k=1}^{n} A_{j,k}\right) \\ &= \frac{1}{n}(D_i + D_j). \end{aligned}$$

□

We are going to extend now the notion of a metric. A nonnegative real function d defined on $M \times M$ is called *quasimetric* (cf. [MS79], [S03]; [MP00]) if $d(x, y) = 0$ when$x = y$, $d(x, y) = d(y, x)$ and $d(x, z) \leq C(d(x, y) + d(y, z))$, the last for some positive real number C, and all $x, y, z \in M$. Thus if it holds with $C = 1$, then d is a metric on M. The proof of Corollary 5 easily extends to quasimetrics. An interesting property of a quasimetric $d(x, y)$ is that $d(x, y)^a$ is also a quasimetric for every positive real number a (cf.[MS79]). Thus this notion encompasses a large number of interesting distance functions which are not metrics, like the squares of Euclidean distances used in clustering applications.

5.1 Core-dense graphs

We now confine attention to the case of graphs. As we saw already, dense graphs are core-dense graphs, but the converse is not in general true. One simple example is a graph consisting of a dense graph on $\Omega(n^{3/4})$ vertices, up to $O(n)$ edges in the subraph defined by the rest of the vertices and up to $O(n^{5/4})$ edges from high-degree vertices of the dense subgraph to the rest. We show below that as in this example, in fact there are always "large" dense subgraphs in a core-dense graph.

Theorem 8. *A core-dense graph with m edges contains a dense induced subgraph with $\theta(\sqrt{m})$ vertices.*

Proof. Since G is core-dense, we have

$$\sum_{i,j \in E} \frac{1}{(d_i + \bar{d})(d_j + \bar{d})} \leq c$$

for some c.

We assume that $m < n^2/16c$; otherwise, G itself is a dense graph.

We partition the vertices of the graph into 3 subsets R, S, T according to their degrees:

$$\begin{aligned} R &= \{i \in V \,:\, d_i \geq 8\sqrt{m}\} \\ S &= \{i \in V \,:\, \frac{\sqrt{m}}{64c} \leq d_i < 8\sqrt{m}\} \\ T &= \{i \in V \,:\, d_i < \frac{\sqrt{m}}{64c}\} \end{aligned}$$

We will prove that $|S| \geq \sqrt{m}/8$. Suppose not for a contradiction.

Using the density condition, the number of edges in the subgraph induced by T is at most

$$c\left(\frac{\sqrt{m}}{64c} + \frac{m}{n}\right)^2 < \frac{m}{16}.$$

Similarly, the number of edges between S and T is at most

$$c(8\sqrt{m} + \frac{m}{n})(\frac{\sqrt{m}}{64c} + \frac{m}{n}) < \frac{m}{8}.$$

Next, the number of vertices in R is at most $2m/8\sqrt{m} = \sqrt{m}/4$. Thus the total number of edges in the graph induced by R is at most $m/32$. Also, the number of edges between R and S is at most

$$|S|\frac{\sqrt{m}}{4} < \frac{m}{32}.$$

Addding up these bounds, the total number of edges in G not in the subgraph induced by S is at most $m/2$. Therefore, the number of vertices in S is at least

$$\frac{m}{8\sqrt{m}} = \frac{\sqrt{m}}{8}$$

which contradicts our assumption.

Thus G contains an induced subgraph with $\sqrt{m}/8$ vertices and minimum degree $\sqrt{m}/64c$. $\square$

References

[AFKK02] N. Alon, W. Fernandez de la Vega, R. Kannan and M. Karpinski, *Random Sampling and MAX-CSP Problems*, Proc. 34th ACM STOC (2002), pp. 232-239.

[AKK95] S. Arora, D. Karger and M. Karpinski, *Polynomial Time Approximation Schemes for Dense Instances of NP-Hard Problems*, Proc. 27th STOC (1995), pp. 284-293; J. Computer and System Sciences 58 (1999), pp. 193-210.

[F96] W. Fernandez de la Vega, *MAX-CUT has a Randomized Approximation Scheme in Dense Graphs*, Random Structures and Algorithms 8 (1996), pp. 187-198.

[FK00] W. Fernandez de la Vega and M. Karpinski, *Polynomial time approximation of dense weighted instances of MAX-CUT*, Random Structures and Algorithms 16 (2000), pp. 314-332.

[FK98] W. Fernandez de la Vega and C. Kenyon, *A randomized approximation scheme for metric MAX-CUT*, Proc. 39th IEEE FOCS (1998), pp. 468-471, final version in J. Computer and System Sciences 63 (2001), pp. 531-541.

[FKKR03] W. Fernandez de la Vega, M. Karpinski, C. Kenyon and Y. Rabani, *Approximation schemes for clustering problems*, Proc. 35th ACM STOC (2003), pp. 50-58.

[FKK04] W. Fernandez de la Vega, M. Karpinski and C. Kenyon, *Approximation Schemes for Metric Bisection and Partitioning*, Proc. 15th ACM-SIAM SODA (2004), pp. 499-508.

[FK96] A. M. Frieze and R. Kannan, *The Regularity Lemma and Approximation Schemes for Dense Problems*, Proc. 37th IEEE FOCS (1996), pp. 12-20.

[FK97] A. M. Frieze and R. Kannan, *Quick Approximation to Matrices and Applications*, Combinatorica 19 (2) (1999), pp. 175-120.

[FKV04] A. M. Frieze, R. Kannan and S. Vempala, *Fast Monte-Carlo Algorithms for Finding Low-Rank Approximations*, J. of the ACM 51(6) (2004), pp. 1025-1041.

[GGR96] O. Goldreich, S. Goldwasser and D. Ron, *Property Testing and its Connection to Learning and Approximation*, Proc. 37th IEEE FOCS (1996), pp. 339-348; J. ACM 45 (1998), pp. 653-750.

[I99] P. Indyk, *A Sublinear Time Approximation Scheme for Clustering in Metric Spaces*, Proc. 40th IEEE FOCS (1999), pp. 154-159.

[MS79] R. Macias and C. Segovia, *Lipschitz functions on spaces of homogenous type*, Advances in Mathematics 33 (1979), pp. 257-270.

[Mc01] F. McSherry, *Spectral Partitioning of Random Graphs*, FOCS 2001, pp.529-537

[MP00] R. R. Mettu and C. G. Plaxton, *The Online Median Problem*, Proc. 41st IEEE FOCS (2000), pp. 339-348.

[S03] S. Semmes, *A brief introduction to Gromov's notion of hyperbolic groups*, Mathematics, arXiv: math CA/021341 (2003), pp. 1-10.

[SS73] P. H. Sneath and R. R. Sokal, *Numerical Taxonomy*, Freeman, London, 1973.