Approximability of Dense Instances of Nearest Codeword Problem

Cristina Bazgan* W. Fernandez de la Vega† Marek Karpinski‡

Abstract. We give a polynomial time approximation scheme (PTAS) for dense instances of the Nearest Codeword problem.

*LAMSADE, Université Paris-Dauphine, 75775 Paris, `bazgan@lamsade.dauphine.fr`.
†LRI, Université Paris Sud, 91405 Orsay, `lalo@lri.fr`.
‡Dept. of Computer Science, University of Bonn, 53117 Bonn, `marek@cs.uni-bonn.de`.

1 Introduction

We follow [KST97] in defining the Nearest Codeword problem as the minimum constraint satisfaction problem for linear equations mod 2 with exactly 3 variables per equation. It is shown in [KST97] that the restriction imposed on the number of variables per equation (fixing it to be exactly 3) does not reduce approximation hardness of the problem. The problem is, for a given set of linear equations mod 2 to construct an asignment which minimizes the number of unsatisfied equations. We shall use in this paper clearly an equivalent formulation of the problem of minimizing the number of satisfied equations. Adopting the notation of [H97] we denote it also as the MIN-E3-LIN2 problem. MIN-Ek-LIN2 will stand for the k-ary version of the Nearest Codeword problem.

The *Nearest Codeword* problem arises in a number of coding theoretic, and algorithmic contexts, see, e.g., [ABSS93], [KST97], [DKS98], [DKRS00]. It is known to be exceedingly hard to approximate; it is known to be NP-hard to approximate to within a factor $n^{\Omega(1)/loglogn}$. In this paper we prove that, somewhat surprisingly, the Nearest Codeword problem on dense instances does have a PTAS. We call an instance of Nearest Codeword problem (MIN-E2-LIN2) problem *dense*, if the number of occurrences of each variable in the equations is $\Theta(n^2)$ for n the number of variables. We call an instance of Nearest Codeword (MIN-E2-LIN2) *dense in average* if the number of equations is $\Theta(n^3)$. Analogously, we define *density*, and *average density*, for MIN-Ek-LIN2 problems.

It is easy to be seen that the results of [AKK95] and [FdV96] on existence of PTASs for average dense maximum constraint satisfaction problems cannot be applied to their average dense minimum analogs (for a survey paper on approximability of some other dense optimization problems see also [K97]). This observation can be also strenghten for the dense instances of minimum constraint satisfaction by noting that dense instances of Vertex Cover can be expressed as dense instances of minimum constraint satisfaction problem for 2DNF clauses, i.e. conjunctions of 2 literals, and then applying the result of [CT96], [KZ97] to the effect that there are no PTAS for the dense Vertex Cover. In [FdVK99] it was also proven that the *dense* and *average dense* instances of MIN TSP(1,2) and LONGEST PATH problems do not have polynomial time approximation schemes.

In [AKK95] there were however two dense minimization problems identified as having PTASs, namely dense Bisection, and MIN-k CUT. This has lead us to investigate the approximation complexity of dense Nearest Codeword problem. Also recently, PTASs have been designed for dense MIN EQUIVALENCE and dense MIN-kSAT problems (cf. [BFdV99], [BFK00]). The main result of this paper is a proof of an existence of a PTAS for the dense Nearest Codeword problem.

The approximation schemes developed in this paper for the dense Nearest Codeword problem use some novel density sampler techniques for graphs, and k-uniform hypergraphs, and extend available up to now approximation techniques for attacking dense instances of minimum constraint satisfaction problems.

The NEAREST CODEWORD problem in its bounded arity (=3) form was proven to be approximation hard for its unbounded arity version in [KST97] (Lemma 37). This results in $n^{\Omega(1)/\log\log n}$ approximation lower bound for the NEAREST CODEWORD problem by [DKS98], [DKRS00], where n is the number of variables. No nontrivial approximation ratios are known for this problem, other than of order n, where n is the number of variables. It is also easy to show that NEAREST CODEWORD is hard to approximate to within a factor $n^{\Omega(1)/\log\log n}$ on average dense instances.

The paper is organized as follows. In Section 2 we give the necessary definitions and prove the NP-hardness of dense instances of MIN-E3-LIN2 in exact setting, and in Section 3 we give a polynomial time approximation scheme for the dense instances of MIN-E3-LIN2.

2 Preliminaries

We begin with basic definitions.

Approximability. A minimization problem has a *polynomial time approximation scheme* (a PTAS, in short) if, there exists a polynomial time approximation algorithm that gives for each instance x of the problem a solution y of value $m(x,y)$ such that $m(x,y) \le (1+\varepsilon)opt(x)$ for every constant $\varepsilon > 0$ where $opt(x)$ is the value of an optimum solution.

NEAREST CODEWORD Problem
(MIN-E3-LIN2)

Input: A set of m equations on boolean variables $x_1, \ldots, x_n$ where each equation has the form $x_{i_1} \oplus x_{i_2} \oplus x_{i_3} = 0$ or $x_{i_1} \oplus x_{i_2} \oplus x_{i_3} = 1$.
Output: An assignment to the variables that minimizes the number of equations satisfied.

Density. A set of instances of MIN-E3-LIN2 is δ-dense if for each variable x, the total number of occurrences of x is at least δn^2 in each instance. A class of instances of MIN-E3-LIN2 is *dense*, if there is a constant δ such that the class is δ-dense.

Let us show now that Dense MIN-E3-LIN2 is NP-hard in exact setting. The reduction is from MIN-E3-LIN2, which is approximation hard for a ratio $n^{\Omega(1)/\log\log n}$ [DKS98], [DKRS00], where n is the number of variables. Given an instance I of MIN-E3-LIN2 on n variables $X = \{x_1, \ldots, x_n\}$ with m equations $x_{t_1} \oplus x_{t_2} \oplus x_{t_3} = b$, where $b \in \{0,1\}$, we construct an instance I' of Dense MIN-E3-LIN2 as follows: we extend the set of variables X by two disjoint sets $Y = \{y_1, \ldots, y_n\}$ and $Z = \{z_1, \ldots, z_n\}$. I' contains aside from the equations of I, the equations of the form $x_i \oplus y_j \oplus z_h = 0$ and $x_i \oplus y_j \oplus z_h = 1$ for all $1 \le i,j,h \le n$. Note that the system I' is dense. We note also that exactly n^3 of the added equations are satisfied independently of the values of the variables in X, Y and Z. Thus $opt(I') = opt(I) + n^3$, proving the claimed reduction.

3 Dense MIN-E3-LIN2 has a PTAS

Let the system $\mathcal{S} = \{E_1, ..., E_m\}$ be a δ-dense instance of MIN-E3-LIN2, on a set X of n variables $\{x_1, \ldots, x_n\}$.

We will run two distinct algorithms, algorithm A and algorithm B, and select the solution with the smallest value. Algorithm A gives a good approximate solution for the instances whose minimum value is at least αn^3. Algorithm B gives a good approximate solution for the instances whose minimum value is less than αn^3, where α is a constant depending both on δ and the required accuracy ε.

3.1 Algorithm A

Algorithm A depends on formulating the problem as a Smooth Integer Program [AKK95] as follows.

A smooth degree-3 polynomial (with smoothness e) has the form

$$\sum a_{ijh} x_i x_j x_h + \sum b_{ij} x_i x_j + \sum c_i x_i + d$$

where each $|a_{ijh}| \le e$, $|b_{ij}| \le en$, $|c_i| \le en^2$, $|d| \le en^3$ (cf. [AKK95]).

For each equation $x_i \oplus y_i \oplus z_i = b_i$ in $\mathcal{S}$, we construct the smooth polynomial

$$P_i \equiv (1-x_i)(1-y_i)(1-z_i) + x_i y_i (1-z_i) + y_i z_i (1-x_i) + z_i x_i (1-y_i)$$

if $b_i = 0$, and

$$P_i \equiv x_i(1-y_i)(1-z_i) + y_i(1-x_i)(1-z_i) + z_i(1-x_i)(1-y_i) + x_i y_i z_i$$

if $b_i = 1$. We have then the Smooth Integer Program IP:

$$\begin{cases} \min \sum_{j=1}^{m} P_i \\ \text{s. t. } x_i, y_i, z_i \in \{0,1\}\ \forall i,\ 1 \le i \le n. \end{cases}$$

A result of [AKK95] can be used now to approximate in polynomial time the minimum value of IP with additive error εn^3 for every $\varepsilon > 0$. This provides an approximation ratio $1+\varepsilon$ whenever the optimum value is $\Omega(n^3)$.

3.2 Algorithm B

The algorithm B is guaranteed to give, as we will show, approximation ratio $1+\varepsilon$ for each fixed ε, whenever the optimum is at most αn^3 for a fixed α, depending on ε and on the density.

Algorithm B

Input: A dense system $\mathcal{S}$ of linear equations in GF[2] over a set X of n variables with exactly 3 variables per equation.

1. Pick two disjoint random samples $S_1, S_2 \subseteq X$ of size $m = \Theta\left(\frac{\log n}{\varepsilon^2}\right)$;

2. For each possible assignment $a \in \{0,1\}^{|S_1 \cup S_2|}$ for the variables y in $S_1 \cup S_2$ (y^a will stand for the boolean value of y for assignement a) do the following:

2.1 For each variable $x \notin S_1 \cup S_2$ do the following:

Let $H^a_{x,0}$ and $H^a_{x,1}$ be the bipartite graphs with common vertex set $V(H^a_{x,0}) = V(H^a_{x,1}) = S_1 \cup S_2$ and edge sets

$$E(H^a_{x,0}) = \{\{y,z\} : \chi_{S_1}(y) \oplus \chi_{S_1}(z) = 1 \wedge x \oplus y \oplus z = b \in \mathcal{S} \wedge y^a \oplus z^a = b\}$$

and

$$E(H^a_{x,1}) = \{\{y,z\} : \chi_{S_1}(y) \oplus \chi_{S_1}(z) = 1 \wedge x \oplus y \oplus z = b \in \mathcal{S} \wedge 1 \oplus y^a \oplus z^a = b\}$$

Let $m_0^a = |E(H_{x,0}^a)|$, $m_1^a = |E(H_{x,1}^a)|$.
If $m_0^a \geq \frac{2}{3}(m_0^a + m_1^a)$, then set x to 1.
If $m_1^a \geq \frac{2}{3}(m_0^a + m_1^a)$, then set x to 0.

Otherwise, set x to be *undefined.*

2.2 In this stage, we assign values to the variables which are undefined after the completion of stage 2.1. Let D^a be the set of variables assigned in stage 2.1, and let $U^a = S_1 \cup S_2 \cup D^a$. $V^a = X \setminus U^a$ denotes the set of undefined variables. For each undefined variable y, let S_y denote the set of equations which contain y and two variables in U^a. Let k_0^a (resp. k_1^a) denote the number of equations in S_y satisfied by a and by setting y to 0 (resp. to 1).

If $k_0^a \leq k_1^a$, then set y to 0. Else, set y to 1.

Let X^a denote the overall assignment produced by the end of this stage.

3. Among all the assignments X^a pick one which satisfies the minimum number of equations of $\mathcal{S}$.

Output that assignment.

4 Proof of the correctness of algorithm B when the value of the instance is "small"

We will use the following graph density sampling lemma. Recall that the density d of a graph $G = (V, E)$ is defined by

$$d = \frac{|E|}{\binom{|V|}{2}}.$$

Lemma 1 *Let d and ε be fixed and let the graph $G = (V, E)$ have $|V| = n$ vertices and density d. Let $m = \Theta(1/d\ \varepsilon^{-2} \log n)$. Let $X = \{x_1, ..., x_m\}$ and $Y = \{y_1, ..., y_m\}$ be two random disjoint subsets of $V(G)$ with $|X| = |Y| = m$ and let $e(X, Y)$ be the number of edges of G between X to Y. Then, for each sufficiently large n, we have*

$$\Pr[|e(X, Y) - m^2 d| \leq \varepsilon m^2 d] = 1 - o(1/n).$$

Proof: We will use the following inequality due to Hoeffding [H64]. Let $X_1, ..., X_m$ be independent and identically distributed. Let $\mu = E(X_1)$ and assume that X_1 satisfies $0 \leq X_1 \leq \Delta$. Let $S_m = \sum_{i=1}^m X_i$. Then:

$$\Pr(|S_m - \mu m| \geq \varepsilon \Delta m) \leq 2 \exp(-2\varepsilon^2 m). \tag{1}$$

Clearly

$$E(e(X, Y)) = m^2 d.$$

For each $z \in V \backslash X$, write

$$T_z = |\Gamma(z) \cap X|.$$

Let $T = \sum_{z \in V \backslash X} T_z$. Then, $T = T' + \Delta$ where $\Delta \leq m(m-1)/2$, and T' is the sum of m randomly chosen valencies from the set of valencies of G. Thus using (1),

$$\Pr[|T' - mnd| \leq \varepsilon mn + m(m-1)^2/2] \geq 1 - 2 \exp(-O(\varepsilon^2 m)).$$

Clearly,

$$\begin{aligned} e(X,Y) &= \sum_{z\in Y} T_z \\ &= \sum_{1\le i\le m} \delta_i \end{aligned}$$

say. Assume now, with negligible error, that the vertices of Y are produced by independent trials. Then, the δ_i are independent r.v.'s with the same distribution as δ_1, defined by

$$\Pr[\delta_1 = k] = \frac{1}{n-m}|\{z \in V(G)|T_z = k\}|,\ 0 \le k \le m.$$

Conditionally on θ where $\theta \in mnd(1 \pm \varepsilon)$ and $E(\delta_1) = \theta$, and using again (1),

$$\Pr[|e(X,Y) - \frac{m\theta}{n}| \le \varepsilon m^2] \ge 1 - 2\exp(-2\varepsilon^2 m)$$

or

$$\Pr[|e(X,Y) - \frac{m\theta}{n}| \le \varepsilon m^2 d] \ge 1 - 2\exp(-2\varepsilon^2 d^2 m).$$

The conditioning event has probability at least $1 - 2\exp(-2\varepsilon^2 m^2 d)$. We have thus, without any conditioning,

$$\begin{aligned} \Pr[|e(X,Y) - \frac{m\theta}{n}| \le \varepsilon m^2 d] &\ge 1 - 2\exp(-2\varepsilon^2 d^2 m) - 2\exp(-2\varepsilon^2 m^2 d) \\ &\ge 1 - 3\exp(-2\varepsilon^2 d^2 m). \end{aligned}$$

This completes the proof. □

We now return to our proof of correctness. We assume, as we can, that a is the restriction to $S_1 \cup S_2$ of an optimal assignment a^*. For each $y \in X$, we let y^{a^*} denote the value of y in a^*. Let $x \in X \setminus S_1 \cup S_2$.

Let $G_{x,0}$ and $G_{x,1}$ be the graphs with common vertex set $V(G_{x,0}) = V(G_{x,1}) = X$ and edge sets

$$E(G_{x,0}) = \{\{y,z\} : x \oplus y \oplus z = b \in \mathcal{S} \wedge y^{a^*} \oplus z^{a^*} = b\}$$

and

$$E(G_{x,1}) = \{\{y,z\} : x \oplus y \oplus z = b \in \mathcal{S} \wedge 1 \oplus y^{a^*} \oplus z^{a^*} = b\}$$

Let $n_0^{a^*} = |E(G_{x,0})|$, $n_1^{a^*} = |E(G_{x,1})|$, $n^{a^*} = n_0^{a^*} + n_1^{a^*}$. Also, let $m^a = m_0^a + m_1^a$.

Lemma 2 *(i) Assume that x is such that we have*

$$n_0^{a^*} \ge \frac{3(n_0^{a^*} + n_1^{a^*})}{4}.$$

Then, with probability $1 - o(1/n)$, x is assigned (correctly) to 1 in step 2.1 of algorithm B.

(ii) Assume that x is such that we have

$$n_1^{a^*} \ge \frac{3(n_0^{a^*} + n_1^{a^*})}{4}.$$

Then, with probability $1 - o(1/n)$, x is assigned (correctly) to 0 in step 2.1 of algorithm B.

(iii) With probability $1 - o(1)$, each variable $y \in D^a$ is assigned to its correct value y^{a^*} by the algorithm B.

Proof: We first prove (iii). Suppose that y is assigned to 1 in stage 2.1. The case where y is assigned to 0 is similar. We have to prove that $n_0^{a^*} \geq n_1^{a^*}$ with probability $1 - o(1/n)$ since if in an optimum solution $x_i = 1$ then $n_0^{a^*} \geq n_1^{a^*}$. Thus, Lemma 1 applied to the graph $G_{x,0}$ with $d = \frac{2n_0^{a^*}}{n(n-1)}$ and the samples S_1 and S_2 gives

$$\Pr\left(m_0^a \leq \frac{8 \cdot 2n_0^{a^*} m^2}{7n(n-1)}\right) = 1 - o(1/n),$$

and so,

$$\Pr\left(n_0^{a^*} \geq \frac{7m_0^a n(n-1)}{2 \cdot 8m^2}\right) = 1 - o(1/n).$$

Also, Lemma 1 applied to the union of the graphs $G_{x,0}$ and $G_{x,1}$ with $d = \frac{2n^{a^*}}{n(n-1)}$ and the samples S_1 and S_2 gives

$$\Pr\left(m^a \geq \frac{8 \cdot 2n^{a^*} m^2}{9n(n-1)}\right) = 1 - o(1/n),$$

and so,

$$\Pr\left(n^{a^*} \leq \frac{9m^a n(n-1)}{2 \cdot 8m^2}\right) = 1 - o(1/n).$$

Since y takes value 1 in stage 2.1 and $m_0^a \geq 2/3m^a$,

$$\Pr\left(\frac{n_0^{a^*}}{n^{a^*}} \geq \frac{7 \cdot 2}{9 \cdot 3}\right) = 1 - o(1/n),$$

and so ,

$$\Pr\left(\frac{n_0^{a^*}}{n^{a^*}} \geq \frac{1}{2}\right) = 1 - o(1/n).$$

Assertion (iii) follows.

Now we prove (i). The proof of (ii) is completely similar to that of (i). Lemma 1 applied to the graph $G_{x,0}$ with $d = \frac{2n_0^{a^*}}{n(n-1)}$ and the samples S_1 and S_2 gives

$$\Pr\left(m_0^a \geq (1-\varepsilon)\frac{2m^2}{n(n-1)} n_0^{a*}\right) = 1 - o(1/n).$$

Let $m^a = m_0^a + m_1^a$. We apply now Lemma 1 to the union of the graphs $G_{x,0}$ and $G_{x,1}$. This gives

$$\Pr\left(m^a \leq (1+\varepsilon)\frac{2m^2}{n(n-1)} n^{a^*}\right) = 1 - o(1/n).$$

Substraction gives

$$\Pr\left(m_0^a - \frac{2m^a}{3} \geq \frac{2m^2}{n(n-1)}((1-\varepsilon)n_o^{a^*} - (1+\varepsilon)\frac{2(n_0^{a^*} + n_1^{a^*})}{3})\right) = 1 - o(1/n).$$

Using the inequality $n_0^{a^*} + n_1^{a^*} \leq \frac{4n_0^{a^*}}{3}$, we obtain

$$\Pr\left(m_0^a - \frac{2m^a}{3} \geq \frac{2m^2}{n(n-1)}\frac{1-20\varepsilon}{9}n_o^{a^*}\right) = 1 - o(1/n),$$

and fixing $\varepsilon = 1/20$,

$$\Pr\left(m_0^a - \frac{2m^a}{3} \geq 0\right) = 1 - o(1/n),$$

concluding the proof.

□

Lemma 3 *With probability $1-o(1)$, the number of variables undefined after the completion of stage 2.1 satisfies*

$$|V^a| \leq \frac{4\ \mathrm{opt}}{\delta n^2}.$$

Proof: Assume that x is undefined. We have thus simultaneously $n_0^{a^*} < \frac{3}{4}(n_0^{a^*} + n_1^{a^*})$ and $n_1^{a^*} < \frac{3}{4}(n_0^{a^*} + n_1^{a^*})$ and so $n_1^{a^*} > \frac{1}{4}(n_0^{a^*} + n_1^{a^*})$ and $n_0^{a^*} > \frac{1}{4}(n_0^{a^*} + n_1^{a^*})$. Since x appears in at least δn^2 equations, $n_0^{a^*} + n_1^{a^*} \geq \delta n^2$. Thus,

$$opt \geq \min\{n_0^{a^*}, n_1^{a^*}\} \cdot |V^a| \geq \frac{\delta n^2}{4}|V^a|.$$

The assertion of the lemma follows. □

We can now complete the correctness proof. Let val denote the value of the solution given by our algorithm and let opt be the value of an optimum solution.

Theorem 1 *Let ε be fixed. If $opt \leq \alpha n^3$ where α is sufficiently small, then we have that $val \leq (1+\varepsilon)opt$.*

Proof: Let us write

$$\mathrm{val} = \mathrm{val}_1 + \mathrm{val}_2 + \mathrm{val}_3 + \mathrm{val}_4$$

where:

- val_1 is the number of satisfied equations with all variables in U^a
- val_2 is the number of satisfied equations with all variables in V^a
- val_3 is the number of satisfied equations with two variables in U^a and one in V^a
- val_4 is the number of satisfied equations with one variable in U^a and two in V^a.

With an obvious intended meaning, we write also

$$\mathrm{opt} = \mathrm{opt}_1 + \mathrm{opt}_2 + \mathrm{opt}_3 + \mathrm{opt}_4$$

We have clearly $\mathrm{val}_1 = \mathrm{opt}_1$ and $\mathrm{val}_3 \leq \mathrm{opt}_3$. Thus,

$$\begin{aligned}
\mathrm{val} &\leq \mathrm{opt} + \mathrm{val}_2 - \mathrm{opt}_2 + \mathrm{val}_4 - \mathrm{opt}_4 \\
&\leq \mathrm{opt} + \mathrm{val}_2 + \mathrm{val}_4 \\
&\leq \mathrm{opt} + \frac{|V^a|^3}{6} + n\frac{|V^a|^2}{2},
\end{aligned}$$

and, using Lemma 3,

$$\begin{aligned}
\text{val} &\leq \text{opt} + \frac{4^3\text{opt}^3}{6\delta^3 n^6} + n\frac{4^2\text{opt}^2}{2\delta^2 n^4} \\
&\leq \text{opt}\left(1 + \frac{32\text{opt}^2}{3\delta^3 n^6} + \frac{8\text{opt}}{\delta^2 n^3}\right).
\end{aligned}$$

Since opt$\leq \alpha n^3$ then,

$$\begin{aligned}
\text{val} &\leq \text{opt}\left(1 + \frac{32\alpha^2}{3\delta^3} + \frac{8\alpha}{\delta^2}\right) \\
&\leq \text{opt}(1+\varepsilon)
\end{aligned}$$

for $\alpha \leq \frac{\delta^2\varepsilon}{9}$ and sufficiently small ε. □

5 Extensions to Dense MIN-Ek-LIN2

We are able to extend our result to arbitrary k-ary versions of the problem, i.e. to Dense MIN-Ek-LIN2 for arbitrary k. This requires a bit more subtle construction, and the design of a density sampler for $(k-1)$-uniform hypergraphs. This extension will appear in the final version of the paper [BFK00].

Acknowledgments

The authors thank Madhu Sudan and Luca Trevisan for stimulating remarks and discussions.

References

[ABSS93] S. Arora, L. Babai, J. Stern and Z. Sweedyk, *The Hardness of Approximate Optima in Lattice, Codes, and Systems of Linear Equations*, Proc. of 34th IEEE FOCS, 1993, 724–733.

[AKK95] S. Arora, D. Karger and M. Karpinski, *Polynomial Time Approximation Schemes for Dense Instances of NP-hard Problems*, Proc. of 27th ACM STOC, 1995, 284–293; the full paper appeared in Journal of Computer and System Sciences 58(1999), 193–210.

[BFdV99] C. Bazgan and W. Fernandez de la Vega, *A Polynomial Time Approximation Scheme for Dense* MIN 2SAT, Proc. Fundamentals of Computation Theory, LNCS 1684, Springer, 1999, 91–99.

[BFK00] C. Bazgan, W. Fernandez de la Vega and M. Karpinski, *Polynomial Time Approximation Schemes for Dense Instances of Minimum Constraint Satisfaction*, 2000, in preparation.

[CT96] A.E.F. Clementi and L. Trevisan, *Improved Non-Approximability Results for Vertex Cover with Density Constraints*, Proc. of 2nd Conference on Computing and Combinatorics, 1996, Springer, 1996, 333–342.

[DKS98] I. Dinur, G. Kindler and S. Safra, *Approximating CVP to Within Almost Polynomial Factors is NP-Hard*, Proc. of 39th IEEE FOCS, 1998, 99–109.

[DKRS00] I. Dinur, G. Kindler, R. Raz and S. Safra, *An Improved Lower Bound for Approximating CVP*, 2000, submitted.

[FdV96] W. Fernandez de la Vega, *Max-Cut Has a Randomized Approximation Scheme in Dense Graphs*, Random Structures and Algorithms, 8(3) (1996), 187–198.

[FdVK99] W. Fernandez de la Vega and M. Karpinski, *On Approximation Hardness of Dense TSP and Other Path Problem*, Information Processing Letters 70, 1999, 53–55.

[H97] J. Hastad, *Some Optimal Inapproximability Results*, Proc. of 29th ACM STOC, 1997, 1–10.

[H64] W. Hoeffding, *Probability Inequalities for Sums of Bounded Random Variables*, Journal of the American Statistical Association, 58(301), 1964, 13–30.

[K97] M. Karpinski, *Polynomial Time Approximation Schemes for Some Dense Instances of NP-Hard Optimization Problems*, Randomization and Approximation Techniques in Computer Science, LNCS 1269, Springer, 1997, 1–14.

[KZ97] M. Karpinski and A. Zelikovsky, *Approximating Dense Cases of Covering Problems*, ECCC Technical Report TR 97-004, 1997, appeared also in DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 40, 1998, 169–178.

[KST97] S. Khanna, M. Sudan and L. Trevisan, *Constraint Satisfaction: The Approximability of Minimization Problems*, Proc. of 12th IEEE Computational Complexity, 1997, 282–296.