Discrete Comput Geom
DOI 10.1007/s00454-008-9110-x

# Fast Dimension Reduction Using Rademacher Series on Dual BCH Codes

**Nir Ailon · Edo Liberty**

Received: 8 November 2007 / Revised: 9 June 2008 / Accepted: 4 September 2008


**Abstract** The Fast Johnson–Lindenstrauss Transform (FJLT) was recently discovered by Ailon and Chazelle as a novel technique for performing fast dimension reduction with small distortion from $\ell_2^d$ to $\ell_2^k$ in time $O(\max\{d\log d, k^3\})$. For $k$ in $[\Omega(\log d), O(d^{1/2})]$, this beats time $O(dk)$ achieved by naive multiplication by random dense matrices, an approach followed by several authors as a variant of the seminal result by Johnson and Lindenstrauss (JL) from the mid 1980s. In this work we show how to significantly improve the running time to $O(d\log k)$ for $k = O(d^{1/2-\delta})$, for any arbitrary small fixed $\delta$. This beats the better of FJLT and JL. Our analysis uses a powerful measure concentration bound due to Talagrand applied to Rademacher series in Banach spaces (sums of vectors in Banach spaces with random signs). The set of vectors used is a real embedding of dual BCH code vectors over $GF(2)$. We also discuss the number of random bits used and reduction to $\ell_1$ space.

The connection between geometry and discrete coding theory discussed here is interesting in its own right and may be useful in other algorithmic applications as well.

**Keywords** Fast dimension reduction · Measure concentration · Probability in Banach spaces · Error correcting codes

---

N. Ailon supported by the National Science Foundation under agreement No. DMS-0111298P.
E. Liberty supported by AFOSR, and NGA.

N. Ailon
Institute for Advanced Study, Princeton, NJ, USA

N. Ailon (✉)
Google Research, New York, NY, USA
e-mail: nailon@google.com

E. Liberty
Yale University, New Haven, CT, USA
e-mail: edo.liberty@yale.edu

## 1 Introduction

Applying random matrices is by now a well-known technique for reducing dimensionality of vectors in Euclidean space while preserving certain properties (most notably distance information). Beginning with the classic work of Johnson and Lindenstrauss [1], who used projections onto random subspaces, other variants of the technique using different distributions are known [2–5] and have been used in many algorithms [6–12]. In all variants of this idea, a fixed unit length vector $x \in \mathbf{R}^d$ is mapped onto $\mathbf{R}^k$ ($k < d$) via a random linear mapping $\Phi$ from a carefully chosen distribution. A measure concentration principle is used to show that the distribution of the norm estimator error $|\|\Phi x\|_2 - 1|$ in a small neighborhood of 0 is dominated by a Gaussian of standard deviation $O(k^{-1/2})$, uniformly for all $x$ and independent of $d$. The distribution of $\Phi$ need not even be rotationally invariant. When used in an algorithm, $k$ is often chosen as $O(\varepsilon^{-2} \log n)$ so that a union bound ensures that the error is smaller than a fixed $\varepsilon$ simultaneously for all $n$ vectors in some fixed input set. Noga Alon [13] proved that this choice of $k$ is essentially optimal and cannot be significantly reduced.

It makes sense to abstract the definition of a distribution of mappings that can be used for dimension reduction in the above sense. We will say that such a mapping has the Johnson–Lindenstrauss property (JLP), named after the authors of the first such construction (we make an exact definition of this property in Sect. 2). In view of Ailon and Chazelle's FJLT result [2], it is natural to ask about the computational complexity of applying a mapping drawn from a JLP distribution on a vector. The resources considered here are time and randomness. Ailon et al. showed that reduction from $d$ dimensions to $k$ dimensions can be performed in time $O(\max\{d \log d\}, k^3)$, beating the naïve $O(kd)$ time implementation of JL for $k$ in $\omega(\log d)$ and $o(d^{1/2})$. Similar bounds were found in [2] for reducing onto $\ell_1$ (Manhattan) space, but with quadratic (not cubic) dependence on $k$. From recent work by Matousek [5] it can be shown, by replacing Gaussian distributions with $\pm 1$'s, that Ailon and Chazelle's algorithm for the Euclidean case requires $O(\max\{d, k^3\})$ random bits in the Euclidean case.

### 1.1 Our Results

This work contains several contributions. We summarize them for the Euclidean case in Table 1 for convenience. The first (in Sect. 7) is a simple trick that can be used to reduce the running time of FJLT [2] to $O(\max\{d \log k\}, k^3)$, hence making it better than the naïve algorithm for small $k$ (first row in the table). In typical applications, the running time translates to $O(d \log\log n)$, where $n$ is the number of points we simultaneously want to reduce (assuming $n = 2^{O(d^{1/3})}$).

The main contribution (Sects. 5–6) is improving the case of "large $k$" (bottom row in the Table 1). We use tools from the theory of probability and norm interpolation in Banach spaces (Sect. 3) and the theory of error correcting codes (Sect. 4) to construct a distribution on matrices satisfying JLP that can be applied in time $O(d \log d)$ (note that, in this case, $\log d = O(\log k)$). Our construction takes advantage of ideas from different classical theories. These ideas provide a new algorithmic application of error correcting codes, an extremely useful tool in theoretical computer science

**Table 1** Schematic comparison of asymptotic running time of this work, Ailon and Chazelle's work [2] (FJLT) and a naïve implementation of Johnson–Lindenstrauss (JL), or variants thereof

| | Fast - - - - - - - - - - - → | | Slow |
|---|---|---|---|
| $k$ in $o(\log d)$ | This work | JL | FJLT |
| $k$ in $\omega(\log d)$ and $o(\text{poly}(d))$ | This work | FJLT | JL |
| $k$ in $\Omega(\text{poly}(d))$ and $o((d\log(d)^{1/3})$ | This work FJLT | | JL |
| $k$ in $\omega((d\log d)^{1/3})$ and $O(d^{1/2-\delta})$ | This work | FJLT | JL |

with applications in both complexity and algorithms (a good overview can be found in [14]; some other recent examples in [15, 16]).

A note on "large $k$": As stated above, $k$ is typically $O(\varepsilon^{-2}\log n)$, where $\varepsilon$ is a desired distortion bound, and $n$ is the number of vectors we seek to reduce. Although $\log n$ is typically small (logarithmic in input size), in various applications, especially in scientific computation, $\varepsilon^{-2}$ may be large. This case is therefore important to study.

It is illustrative to point out an apparent weakness in [2] that was a starting point of our work. The main tool used there was to multiply the input vector $x$ by a random sign change matrix followed by a Fourier transform, resulting in a vector $y$. It is claimed that $\|y\|_\infty$ is small (in other words, the "information" is spread out evenly among the coordinates). By a convexity argument the "worst case" $y$ (assuming only the $\ell_\infty$ bound) is a *uniformly supported* vector in which the absolute values of the coordinates in its (small) support are all equal. Intuitively, such a vector is extremely unlikely. In this work we consider other norms.

It is likely that the techniques we develop here can be used in conjunction with very recent research on explicit embeddings of $\ell_2$ in $\ell_1$ [17–19] and research on fast approximate linear algebraic scientific computation [11, 20–25].

## 2 Preliminaries

We use $\ell_p^d$ to denote the $d$-dimensional real space equipped with the norm $\|x\| = \|x\|_p = (\sum_{i=1}^d |x_i|^p)^{1/p}$, where $1 \le p < \infty$ and $\|x\|_\infty = \max\{|x_i|\}$. The dual norm index $q$ is defined by the solution to $1/q + 1/p = 1$. We remind the reader that $\|x\|_p = \sup_{\substack{y\in\ell_q^d\\ \|y\|=1}} x^T y$. For a real $k\times d$ matrix $A$, the matrix norm $\|A\|_{p_1\to p}$ is defined as the operator norm of $A: \ell_{p_1}^d \to \ell_p^k$ or

$$\|A\|_{p_1\to p} = \sup_{\substack{x\in\ell_{p_1}^d\\ \|x\|=1}} \|Ax\|_p = \sup_{\substack{y\in\ell_q^k\\ \|y\|=1}} \sup_{\substack{x\in\ell_{p_1}^d\\ \|x\|=1}} y^T Ax.$$

In what follows we use $d$ to denote the original dimension and $k < d$ the target (reduced) dimension. The input vector will be $x = (x_1, \dots, x_d)^T \in \ell_2^d$. Since we only consider linear reductions, we will assume without loss of generality that $\|x\|_2 = 1$.

**Definition 2.1** A distribution $\mathcal{D}(d,k)$ on $k \times d$ real matrices ($k \le d$) has the Johnson–Lindenstrauss property (JLP) with respect to a norm index $p$, if for any *unit* vector $x \in \ell_2^d$ and $0 \le \varepsilon < 1/2$,

$$\Pr_{A \sim \mathcal{D}_{d,k}} \left[ \left| \|Ax\|_p - 1 \right| > \varepsilon \right] \le c_1 e^{-c_2 k \varepsilon^2} \tag{2.1}$$

for some global $c_1, c_2 > 0$.

(A similar definition was given in [11].) In this work, we study the cases $p = 1$ (*Manhattan* JLP) and $p = 2$ (*Euclidean* JLP). We make a few technical remarks about Definition 2.1:

- For most dimension reduction applications, $k = \Omega(\varepsilon^{-2})$, so the constant $c_1$ can be "swallowed" by $c_2$, but we prefer to keep it here to avoid writing $O(e^{-\Omega(k\varepsilon^2)})$ and for generality.
- The definition is robust with respect to bias of $O(k^{-1/2})$. More precisely, if we prove $\Pr[\mu - \varepsilon \le \|Ax\|_p \le \mu + \varepsilon] \ge 1 - c_1 e^{-c_2 k \varepsilon^2}$ for some $\mu$ satisfying $|\mu - 1| = O(k^{-1/2})$, then this would imply (2.1), with possibly different constants. We will use this observation in what follows.

Recall that a Walsh–Hadamard matrix $H_d$ is a $d \times d$ orthogonal matrix with $H_d(i,j) = 2^{-d/2}(-1)^{\langle i,j \rangle}$ for all $i, j \in [0, d-1]$, where $\langle i, j \rangle$ is the dot product (over $\mathbb{F}_2$) of $i, j$ viewed as $(\log d)$-bit vectors. The matrix encodes the Fourier transform over the binary hypercube. It is well known that $x \mapsto H_d x \in \ell_2^d$ can be computed in time $O(d \log d)$ for any $x \in \ell_2^d$ and that the mapping is isomorphic.

**Definition 2.2** A matrix $A \in \mathbf{R}^{m \times d}$ is a *code* matrix if every row of $A$ is equal to some row of $H_d$ multiplied by $\sqrt{d/m}$.

The normalization is chosen so that columns have Euclidean norm 1.

## 2.1 Statement of our Theorems

The main contribution is in Theorem 2.2 below.

**Theorem 2.1** *For any code matrix $A$ of size $k \times d$ for $k < d$, the mapping $x \mapsto Ax$ can be computed in time $O(d \log k)$.*

Clearly this theorem is interesting only for $\log k = o(\log d)$, because otherwise the Walsh–Hadamard transform followed by projection onto a subset of the coordinates can do this in time $O(d \log d)$, by the definition of a code matrix. As a simple corollary, the running time of the algorithms in [2] can be reduced to $O(\max\{d \log k, k^3\})$, because effectively what they do is multiply the input $x$ (after a random sign change) by a code matrix of size $O(k^3) \times d$ and then manipulate the outcome in time $O(k^3)$. This gives the left column of Table 1. We refer the reader to [2, 5] for details on the original $O(\max\{d \log d, k^3\})$-time algorithm, from which, together with Theorem 2.1, the improvement is obvious.

The proof of Theorem 2.1 is done by a careful pruning of the execution tree of the original Walsh–Hadamard transform.

**Theorem 2.2** *Let $\delta > 0$ be some arbitrarily small constant. For any $d, k$ satisfying $k \le d^{1/2-\delta}$, there exists an algorithm constructing a random matrix $A$ of size $k \times d$ satisfying JLP, such that the time to compute $x \mapsto Ax$ for any $x \in \mathbf{R}^d$ is $O(d \log k)$. The construction uses $O(d)$ random bits and applies to both the Euclidean and the Manhattan cases.*

We will prove a slightly weaker running time of $O(d \log d)$ below and provide a sketch for reducting it to $O(d \log k)$, where the full details of the improvement are deferred to Sect. 8. This improvement is interesting for small $k$ and provides a unified solution for all $k \le d^{1/2-\delta}$, though the small $k$ case can also be taken care of using Theorem 2.1 above in conjunction with FJLT [2]. The main contribution of Theorem 2.1, of course, is in getting rid of the term $k^3$ in the running time of FJLT.

The matrix $A$ from Theorem 2.2 will be constructed by composing a constant number of fast transformations, each of which is a composition of a pair of matrices: a code matrix and a random diagonal matrix. Each pair is responsible for controlling the norm of the transformed input vector in a carefully chosen Banach space. The main tool that will be used for analyzing the matrix pair output is the theory of Rademacher random variables in Banach spaces.

## 3 Tools from Banach Spaces

The following is known as an interpolation theorem in the theory of Banach spaces. For a proof, refer to [26].

**Theorem 3.1** (Riesz–Thorin) *Let $A$ be an $m \times d$ real matrix, and assume that $\|A\|_{p_1 \to r_1} \le C_1$ and $\|A\|_{p_2 \to r_2} \le C_2$ for some norm indices $p_1, r_1, p_2, r_2$. Let $\lambda$ be a real number in the interval $[0, 1]$, and let $p, r$ be such that $1/p = \lambda(1/p_1) + (1-\lambda)(1/p_2)$ and $1/r = \lambda(1/r_1) + (1-\lambda)(1/r_2)$. Then $\|A\|_{p \to r} \le C_1^{\lambda} C_2^{1-\lambda}$.*

**Theorem 3.2** (Hausdorff–Young) *For norm index $1 \le p \le 2$, $\|H\|_{p \to q} \le d^{-1/p+1/2}$, where $q$ is the dual norm index of $p$.*

(This theorem is usually stated with respect to the Fourier operator for functions on the real line or on the circle and is a simple application of Riesz–Thorin by noticing that $\|H\|_{2 \to 2} = 1$ and $\|H\|_{1 \to \infty} = d^{-1/2}$.)

Let $M$ be a real $m \times d$ matrix, and let $z \in \mathbf{R}^d$ be a random vector with each $z_i$ distributed uniformly and independently over $\{\pm 1\}$. The random vector $Mz \in \ell_p^m$ is known as a *Rademacher* random variable. A nice exposition of concentration bounds for Rademacher variables is provided in Chap. 4.7 of [27] for more general Banach spaces. For our purposes, it suffices to review the result for finite-dimensional $\ell_p$ space. Consider the norm $Z = \|Mz\|_p$ (we say that "$Z$ is the norm of a Rademacher random variable in $\ell_p^d$ corresponding to $M$"). We associate two numbers with $Z$,

- the deviation $\sigma$, defined as $\|M\|_{2\to p}$, and
- the median $\mu$ of $Z$.

**Theorem 3.3** *For any* $t \geq 0$, $\Pr[|Z-\mu| > t] \leq 4e^{-t^2/(8\sigma^2)}$.

The theorem is a simple consequence of a powerful theorem of Talagrand (Chap. 1 of [27]) on measure concentration of functions on $\{-1,+1\}^d$ extendable to convex functions on $\ell_2^d$ with bounded Lipschitz norm.

## 4 Tools from Error Correcting Codes

Let $A$ be a code matrix, as defined above. The columns of $A$ can be viewed as vectors over $\mathbb{F}_2$ under the usual transformation $((+) \to 0, (-) \to 1)$. Clearly, the set of vectors thus obtained are closed under addition and hence constitute a linear subspace of $\mathbb{F}_2^m$. Conversely, any linear subspace $V$ of $\mathbb{F}_2^m$ of dimension $\nu$ can be encoded as an $m \times 2^\nu$ code matrix (by choosing some ordered basis of $V$). We will borrow well-known constructions of subspaces from coding theory, hence the terminology. Incidentally, note that $H_d$ encodes the Hadamard code, equivalent to a dual BCH code of designed distance 3.

**Definition 4.1** A code matrix $A$ of size $m \times d$ is *a-wise independent* if for all $1 \leq i_1 < i_2 < \cdots < i_a \leq m$ and $(b_1, b_2, \ldots, b_a) \in \{+1,-1\}^a$, the number of columns $A^{(j)}$ for which $(A_{i_1}^{(j)}, A_{i_2}^{(j)}, \ldots, A_{i_a}^{(j)}) = m^{-1/2}(b_1, b_2, \ldots, b_a)$ is exactly $d/2^a$.

**Lemma 4.1** *There exists a* 4*-wise independent code matrix of size* $k \times f_{\mathrm{BCH}}(k)$, *where* $f_{\mathrm{BCH}}(k) = \Theta(k^2)$.

The family of matrices is known as binary dual BCH codes of designed distance 5. Details of the construction can be found in [28].

## 5 Reducing to Euclidean Space for $k \leq d^{1/2-\delta}$

Assume that $\delta > 0$ is some arbitrarily small constant. Let $B$ be a $k \times d$ matrix with Euclidean unit length columns, and $D$ a random $\{\pm 1\}$ diagonal matrix. Let $Y = \|BDx\|_2$. Our goal is to get a concentration bound of $Y$ around 1. Notice that $E[Y^2] = 1$. In order to use Theorem 3.3, we let $M$ denote the $k \times d$ matrix with its $i$th column $M^{(i)}$ being $x_i B^{(i)}$, where $B^{(i)}$ denotes the $i$th column of $B$. Clearly $Y$ is the norm of a Rademacher random variable in $\ell_2^k$ corresponding to $M$. We estimate the deviation $\sigma$ and median $\mu$, as defined in Sect. 3.

$$\sigma = \|M\|_{2\to 2} = \sup_{\substack{y \in \ell_2^k \\ \|y\|=1}} \left\| y^T M \right\|_2 = \sup \left( \sum_{i=1}^d x_i^2 \left( y^T B^{(i)} \right)^2 \right)^{1/2}$$

$$\leq \|x\|_4 \sup\left(\sum_{i=1}^{d}\left(y^T B^{(i)}\right)^4\right)^{1/4} = \|x\|_4 \left\|B^T\right\|_{2\to 4}. \tag{5.2}$$

(The inequality is Cauchy–Schwarz.) To estimate the median, $\mu$, we compute

$$E\left[(Y-\mu)^2\right] = \int_0^\infty \Pr\left[(Y-\mu)^2 > s\right]ds \leq \int_0^\infty 4e^{-s/(8\sigma^2)}\,ds = 32\sigma^2.$$

The inequality is an application of Theorem 3.3. Recall that $E[Y^2]=1$. Also, $E[Y]=E[\sqrt{Y^2}] \leq \sqrt{E[Y^2]} = 1$ (by Jensen). Hence $E[(Y-\mu)^2] = E[Y^2] - 2\mu E[Y] + \mu^2 \geq 1 - 2\mu + \mu^2 = (1-\mu)^2$. Combining, $|1-\mu| \leq \sqrt{32}\sigma$. We conclude,

**Corollary 5.1** *For any* $t \geq 0$,

$$\Pr\left[|Y-1| > t\right] \leq c_3 \exp\left\{-c_4 t^2 / \left(\|x\|_4^2 \left\|B^T\right\|_{2\to 4}^2\right)\right\}$$

*for some global* $c_3, c_4 > 0$.

In order for the distribution of $BD$ to satisfy JLP, we need to have $\sigma = O(k^{-1/2})$. This requires controlling both $\|B^T\|_{2\to 4}$ and $\|x\|_4$. We first show how to design a matrix $B$ that is both efficiently computable and has a small norm. The latter quantity is adversarial and cannot be directly controlled, but we are allowed to manipulate $x$ by applying a (random) orthogonal matrix $\Phi$ without losing any information.

## 5.1 Bounding $\|B^T\|_{2\to 4}$ Using BCH Codes

**Lemma 5.1** *Assume that* $B$ *is a* $k \times d$ 4*-wise independent code matrix. Then* $\|B^T\|_{2\to 4} \leq (3d)^{1/4} k^{-1/2}$.

*Proof* For $y \in \ell_2^k$, $\|y\| = 1$,

$$\begin{aligned}
\left\|y^T B\right\|_4^4 &= d E_{j\in[d]}\left[\left(y^T B^{(j)}\right)^4\right] \\
&= dk^{-2} \sum_{i_1,i_2,i_3,i_4=1}^{k} E_{b_{i_1},b_{i_2},b_{i_3},b_{i_4}}\left[y_{i_1} y_{i_2} y_{i_3} y_{i_4} b_{i_1} b_{i_2} b_{i_3} b_{i_4}\right] \\
&= dk^{-2}\left(3\|y\|_2^4 - 2\|y\|_4^4\right) \leq 3dk^{-2},
\end{aligned} \tag{5.3}$$

where $b_{i_1}$ through $b_{i_k}$ are independent random $\{+1,-1\}$ variables. We now use the BCH codes. Let $B_k$ denote the $k \times f_{\mathrm{BCH}}(k)$ matrix from Lemma 4.1 (we assume here that $k = 2^a - 1$ for some integer $a$; this is harmless because otherwise we can reduce onto some $k' = 2^a - 1$ such that $k/2 \leq k' \leq k$ and pad the output with $k - k'$ zeros). In order to construct a matrix $B$ of size $k \times d$ for $k \leq d^{1/2-\delta}$, we first make sure that $d$ is divisible by $f_{\mathrm{BCH}}(k)$ (by at most multiplying $d$ by a constant factor and padding with zeros) and then define $B$ to be $d/f_{\mathrm{BCH}}(k)$ copies of $B_k$ side by side. Clearly $B$ remains 4-wise independent. Note that $B$ may no longer be a code matrix, but $x \mapsto Bx$ is computable in time $O(d \log k)$ by performing $d/f_{\mathrm{BCH}}(k)$ Walsh transforms on blocks of size $f_{\mathrm{BCH}}(k)$. □

## 5.2 Controlling $\|x\|_4$ for $k < d^{1/2-\delta}$

We define a randomized orthogonal transformation $\Phi$ that is computable in $O(d \log d)$ time and succeeds with probability $1 - O(e^{-k})$ for all $k < d^{1/2-\delta}$. Success means that $\|\Phi x\|_4 = O(d^{-1/4})$. (Note: Both big-$O$s hide factors depending on $\delta$). Note that this construction gives a running time of $O(d \log d)$. We discuss later how to do this for arbitrarily small $k$ with running time $O(d \log k)$.

The basic building block is the product $HD'$, where $H = H_d$ is the Walsh–Hadamard matrix, and $D'$ is a diagonal matrix with random i.i.d. uniform $\{\pm 1\}$ on the diagonal. Note that this random transformation was the main ingredient in [2]. Let $H^{(i)}$ denote the $i$th column of $H$.

We are interested in the random variable $X = \|HD'x\|_4$. We define $M$ as the $d \times d$ matrix with the $i$th column $M^{(i)}$ being $x_i H^{(i)}$, we let $p = 4$ ($q = 4/3$), and notice that $X$ is the norm of the Rademacher random variable in $\ell_4^d$ corresponding to $M$ (using the notation of Sect. 3). We compute the deviation $\sigma$,

$$\sigma = \|M\|_{2\to 4} = \left\|M^T\right\|_{4/3\to 2} = \sup_{\substack{y\in\ell_{4/3}^k\\ \|y\|_{4/3}=1}} \left(\sum_i x_i^2 \left(y^T H^{(i)}\right)^2\right)^{1/2}$$

$$\le \left(\sum x_i^4\right)^{1/4} \sup\left(\sum_i \left(y^T H^{(i)}\right)^4\right)^{1/4} = \|x\|_4 \left\|H^T\right\|_{\frac{4}{3}\to 4}. \tag{5.4}$$

(Note that $H^T = H$.) By the Hausdorff–Young theorem, $\|H\|_{\frac{4}{3}\to 4} \le d^{-1/4}$. Hence, $\sigma \le \|x\|_4 d^{-1/4}$. We now get by Theorem 3.3 that for all $t \ge 0$,

$$\Pr\left[\left|\left\|HD'x\right\|_4 - \mu\right| > t\right] \le 4e^{-t^2/(8\|x\|_4^2 d^{-1/2})}, \tag{5.5}$$

where $\mu$ is a median of $X$.

**Claim 5.1** $\mu = O(d^{-1/4})$.

*Proof* To see the claim, notice that for each separate coordinate, $E[(HD'x)_i^4] = O(d^{-2})$ and then use linearity of expectation to get $E[\|HD'x\|_4^4] = O(d^{-1})$. By Jensen's inequality, $E[\|HD'x\|_4^b] \le E[\|HD'x\|_4^4]^{b/4} = O(d^{-b/4})$ for $b = 1, 3$. Now

$$\begin{aligned} E\left[\left(\left\|HD'x\right\|_4 - \mu\right)^4\right] &= \int_0^\infty \Pr\left[\left(\left\|HD'x\right\|_4 - \mu\right)^4 > s\right] ds \\ &\le \int_0^\infty 4e^{-s^{1/2}/(8\|x\|_4^2 d^{-1/2})}\, ds \\ &= O(d^{-1}). \end{aligned}$$

This implies by multiplying the LHS out that $-\gamma_1 d^{-3/4}\mu - \gamma_2 d^{-1/4}\mu^3 + \mu^4 \le \gamma_3 d^{-1}$, where $\gamma_i > 0$ are global constants for $i = 1, 2, 3$. The statement of the claim immediately follows. □

Let $c_9$ be such that $\mu_4 \le c_9 d^{-1/4}$. We weaken inequality (5.5) using the last claim to obtain the following convenient form:

$$\Pr\left[\left\|HD'x\right\|_4 > c_9 d^{-1/4} + t\right] \le 4e^{-t^2/(8\|x\|_4^2 d^{-1/2})}. \tag{5.6}$$

In order to get a desired failure probability of $O(e^{-k})$, set $t = c_8 k^{1/2}\|x\|_4 d^{-1/4}$. For $k < d^{1/2-\delta}$, this gives $t < c_8 d^{-\delta/2}\|x\|_4$. In other words, with probability $1 - O(e^{-k})$ we get

$$\left\|HD'x\right\|_4 \le c_9 d^{-1/4} + c_8 d^{-\delta/2}\|x\|_4.$$

Now compose this $r$ times: Take independent random diagonal $\{\pm 1\}$ matrices $D' = D^{(1)}, D^{(2)}, \dots, D^{(r)}$ and define $\Phi_d^{(r)} = HD^{(r)}HD^{(r-1)}\cdots HD^{(1)}$. Using a union bound on the conditional failure probabilities, we get that with probability at least $1 - O(re^{-k})$,

$$\left\|\Phi_d^{(r)}x\right\|_4 \le d^{-1/4}c_9 c_8^r + c_8^r d^{-\delta r/2}\|x\|_4. \tag{5.7}$$

(We assumed $c_8 > 2$.) Hence, under our assumption that $\delta$ is constant, recalling that $\|x\|_4 \le \|x\|_2 = 1$, we get by plugging into (5.7):

**Lemma 5.2** ($\ell_4$ reduction for $k < d^{1/2-\delta}$) *With probability of success* $1 - O(e^{-k})$,

$$\left\|\Phi^{(r)}x\right\|_4 = O\left(d^{-1/4}\right) \tag{5.8}$$

*for* $r = \lceil 1/2\delta \rceil$.

(Note that the constant hiding in the bound (5.8) is exponential in $1/\delta$.)

Conditioned on the success event from Lemma 5.2, we can take $\Phi_d^{(r)}x$ as input to the random transformation $BD$, and combining with Corollary 5.1 we conclude that the random transformation $A = BD\Phi^{(r)}$ has Euclidean JLP for $k < d^{1/2-\delta}$ and can be applied to a vector in time $O(d\log d)$. This proves the Euclidean case of Theorem 2.2.

## 5.3 Reducing the Running Time to $O(d\log k)$

We now explain how to reduce the running time to $O(d\log k)$, using the new tools developed here. This provides a unified solution to the problem of designing efficient Johnson–Lindenstrauss projections for all $k$ up to $d^{1/2-\delta}$. Recall that in the construction of $B$ we placed $d/f_{\mathrm{BCH}}(k)$ copies of the same code matrix $B_k$ of size $k \times f_{\mathrm{BCH}}(k)$ side by side. It turns out that we can apply this "decomposition" of coordinates to $\Phi^{(r)}$. Indeed, let $I_j \subseteq [d]$ denote the $j$th block of $\beta = f_{\mathrm{BCH}}(k)k^\delta$ consecutive coordinates (assume that $\beta$ is an integer that divides $d$). For a vector $y \in \ell_p^d$, let $y_{I_j} \in \ell_p^\beta$ denote the projection of $y$ onto the set of coordinates $I_j$. Now, instead of using $\Phi^{(r)} = \Phi_d^{(r)}$ as above, we use a block-diagonal $d \times d$ matrix comprised of $d/\beta$ $\beta \times \beta$ blocks each drawn from the same distribution as $\Phi_\beta^{(r)}$. Clearly the running time of the block-diagonal matrix is $O(d\log k)$, by applying the Walsh transform independently on each block (recall that $\beta = f_{\mathrm{BCH}}(k)k^\delta = O(k^{2+\delta})$).

In order to see why this still works, one needs to repeat the above proofs using the family of norms $\|\cdot\|_{(p_1,p_2)}$ indexed by two norm indices $p_1$, $p_2$ and defined as

$$\|x\|_{(p_1,p_2)} = \left(\sum_{j=1}^{d/\beta} \|x_{I_j}\|_{p_1}^{p_2}\right)^{1/p_2}.$$

We defer the proofs to Sect. 8 below.

## 6 Reducing to Manhattan Space for $k < d^{1/2-\delta}$

We sketch this simpler case. As we did for the Euclidean case, we start by studying the random variable $W \in \ell_1^k$ defined as $W = \|k^{1/2}BDx\|_1$ for $B$ as described in Sect. 5 and $D$ a random $\pm 1$-diagonal matrix. In order to characterize the concentration of $W$ (the norm of a Rademacher r.v. in $\ell_1^k$), we compute the deviation $\sigma$ and estimate the median $\mu$. As before, we set $M$ to be the $k \times d$ matrix with the $i$th column being $k^{1/2}B^{(i)}x_i$.

$$\begin{aligned}\sigma = \sup_{\substack{y\in\ell_\infty^k\\ \|y\|=1}} \left\|y^T M\right\|_2 &= \sup\left(k\sum_{i=1}^d x_i^2\left(y^T B^{(i)}\right)^2\right)^{1/2}\\ &\le \sup k^{1/2}\|x\|_4 \left\|y^T B^{(i)}\right\|_4 = k^{1/2}\|x\|_4\left\|B^T\right\|_{\infty\to 4}. \qquad (6.9)\end{aligned}$$

Using the tools developed in the Euclidean case, we can reduce $\|x\|_4$ to $O(d^{-1/4})$ with probability $1 - O(e^{-k})$ using $\Phi_r(d)$, in time $O(d\log d)$ (in fact, $O(d\log k)$ using the improvement from Sect. 8). Also we already know from Sect. 5.1 that $\|B^T\|_{2\to 4} = O(d^{1/4}k^{-1/2})$ if $B$ is comprised of $k \times f_{\mathrm{BCH}}(k)$ dual BCH codes (of designed distance 5) matrices side by side (assume $f_{\mathrm{BCH}}(k)$ divides $d$). Since $\|y\|_\infty \ge k^{-1/2}\|y\|_2$ for any $y \in \ell_k$, we conclude that $\|B^T\|_{\infty\to 4} = O(d^{1/4})$. Combining, we get $\sigma = O(k^{1/2})$. We now estimate the median $\mu$ of $W$.

In order to calculate $\mu$ we first calculate $E(W) = kE[|P|]$ where $P$ is any single coordinate of $k^{1/2}BDx$. We follow (almost exactly) a proof by Matousek in [5], where he uses a quantitative version of the Central Limit Theorem by König, Schütt, and Tomczak [29].

**Lemma 6.1** (König–Schütt–Tomczak) *Let $z_1 \dots z_d$ be independent symmetric random variables with $\sum_{i=1}^d E[z_i^2] = 1$, let $F(t) = \Pr[\sum_{i=1}^d z_i < t]$, and let $\overline{\varphi}(t) = \frac{1}{2\pi}\int_{-\infty}^t e^{-x^2/2}dx$. Then*

$$\left|F(t) - \overline{\varphi}(t)\right| \le \frac{C}{1+|t|^3}\sum_{i=1}^d E\left[|z_i|^3\right]$$

*for all $t \in \mathbf{R}$ and some constant $C$.*

Clearly we can write $P = \sum_{i=1}^d z_i$ where $z_i = D'_i x_i$ and each $D'_i$ is a random $\pm 1$. Note that $\sum_{i=1}^d E[|z_i|^3] = \|x\|_3^3$. Let $\beta$ be the constant $\int_{-\infty}^{\infty} |t| d\overline{\varphi}(t)$ (the expectation of the absolute value of a Gaussian).

$$\begin{aligned}\big|E\big[|P|\big] - \beta\big| &= \left|\int_{-\infty}^{\infty} |t|\, dF(t) - \int_{-\infty}^{\infty} |t|\, d\overline{\varphi}(t)\right| \\ &\le \int_{-\infty}^{\infty} \big|F(t) - \overline{\varphi}(t)\big|\, dt \le \|x\|_3^3 \int_{-\infty}^{\infty} \frac{C}{1+|t|^3}\, dt.\end{aligned}$$

We claim that $\|x\|_3^3 = O(k^{-1})$. To see this, recall that $\|x\|_2 = 1$, $\|x\|_4 = O(d^{-1/4})$. Equivalently, $\|x^T\|_{2\to 2} = 1$ and $\|x^T\|_{4/3\to 2} = O(d^{-1/4})$. By applying Riesz–Thorin we get that $\|x\|_3 = \|x^T\|_{3/2\to 2} = O(d^{-1/6})$, hence $\|x\|_3^3 = O(d^{-1/2})$. Since $k = O(d^{1/2})$, the claim is proved.

By linearity of expectation we get $E(W) = k\beta(1 \pm O(k^{-1}))$. We now bound the distance of the median from the expected value:

$$\begin{aligned}\big|E(W) - \mu\big| &\le E\big[|W-\mu|\big] \\ &= \int_0^{\infty} \Pr\big[|W-\mu| > t\big]\, dt \le \int_0^{\infty} 4e^{-t^2/(8\sigma^2)} dt = O\big(k^{1/2}\big)\end{aligned}$$

(we used our estimate $\sigma = O(k^{1/2})$ above). We conclude that $\mu = k\beta(1 + O(k^{-1/2}))$. This clearly shows that (up to normalization) the random transformation $BD\Phi^{(r)}$ (where $r = \lceil 1/\delta \rceil$) has the JL property with respect to embedding into Manhattan space. The running time is $O(d \log d)$.

## 7 Trimmed Walsh–Hadamard Transform

We prove Theorem 2.1. For simplicity, let $H = H_d$. It is well known that computing the Walsh–Hadamard transform $H\mathbf{x}$ requires $O(d \log d)$ operations. It turns out that it is possible to compute $PH\mathbf{x}$ with $O(d \log k)$ operations, as long as the matrix $P$ contains at most $k$ nonzeros. This will imply Theorem 2.1, because code matrices of size $k \times d$ are a product of $PH_d$, where $P$ contains $k$ rows with exactly one nonzero in each row. To see this, we remind the reader that the Walsh–Hadamard matrix (up to normalization) can be recursively described as

$$H_1 = \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \qquad H_q = \begin{pmatrix} H_{q/2} & H_{q/2} \\ H_{q/2} & -H_{q/2} \end{pmatrix}.$$

We define $\mathbf{x}_1$ and $\mathbf{x}_2$ to be the first and second halves of $\mathbf{x}$. Similarly, we define $P_1$ and $P_2$ as the left and right halves of $P$, respectively.

$$\begin{aligned} PH_q\mathbf{x} &= \begin{pmatrix} P_1 & P_2 \end{pmatrix} \begin{pmatrix} H_{q/2} & H_{q/2} \\ H_{q/2} & -H_{q/2} \end{pmatrix} \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix} \\ &= P_1 H_{q/2}(\mathbf{x}_1 + \mathbf{x}_2) + P_2 H_{q/2}(\mathbf{x}_1 - \mathbf{x}_2). \end{aligned} \tag{7.10}$$

$P_1$ and $P_2$ contain $k_1$ and $k_2$ nonzeros, respectively, $k_1 + k_2 = k$, giving the recurrence relation $T(d,k) = T(d/2, k_1) + T(d/2, k_2) + d$ for the running time. The base cases are $T(d, 0) = 0$ and $T(d, 1) = d$. We use induction to show that $T(d,k) \le 2d \log(k+1)$:

$$\begin{aligned} T(d,k) &= T(d/2,k_1) + T(d/2,k_2) + d \\ &\le d \log(2(k_1+1)(k_2+1)) \\ &\le d \log((k_1+k_2+1)^2) \quad \text{for } k_1+k_2 = k \ge 1 \\ &\le 2d \log(k+1). \end{aligned}$$

The last sequence of inequalities, together with the base cases, clearly also gives an algorithm and proves Theorem 2.1.

Since in [2] both Hadamard and Fourier transforms were considered, we also shortly describe a simple trimmed Fourier transform. In order to compute $k$ coefficients from a $d$-dimensional Fourier transform on a vector $\mathbf{x}$, we divide $\mathbf{x}$ into $L$ blocks of size $d/L$ and begin with the first step of the Cooley Tukey algorithm which performs $d/L$ FFTs of size $L$ between the blocks (and multiplies them by twiddle factors). In the second step, instead of computing FFTs inside each block, each coefficient is computed directly, by summation, inside its block. These two steps require $(d/L) \cdot L \log(L)$ and $kd/L$ operations, respectively. By choosing $k/\log(k) \le L \le k$ we achieve a running time of $O(d \log(k))$.

## 8 Reducing the Running Time to $O(d \log k)$ for Small $k$

Recall the construction in Sect. 5: $\delta > 0$ is an arbitrarily small constant, we assume that $k \le d^{1/2-\delta}$, that $k^\delta$ is an integer, and that $\beta = f_{\mathrm{BCH}}(k)k^\delta$ divides $d$ (all these requirements can be easily satisfied by slightly reducing $\delta$ and at most doubling $d$). The matrix $B$ is of size $k \times d$ and was defined as follows:

$$B = \begin{pmatrix} B_k & B_k \cdots B_k \end{pmatrix},$$

where $B_k$ is the $k \times f_{\mathrm{BCH}}(k)$ code matrix from Lemma 4.1. Let $\hat{B}$ denote $k^\delta$ copies of $B_k$, side by side. So $\hat{B}$ is of size $k \times \beta$, and $B$ consists of $d/\beta$ copies of $\hat{B}$. As in Sect. 5, we start our construction by studying the distribution of the $\ell_2$ estimator $Y = \|BDx\|_2$, where $D$ is our usual random $\pm 1$ diagonal matrix. Going back to (5.2) (recall that $M$ is the matrix whose $i$th column $M^{(i)}$ is $x_i B^{(i)}$), we recompute the deviation $\sigma$:

$$\begin{aligned} \sigma = \|M\|_{2\to 2} &= \sup_{\substack{y \in \ell_2^k \\ \|y\|=1}} \left\| y^T M \right\|_2 \\ &= \sup \left( \sum_{i=1}^{d} x_i^2 \left( y^T B^{(i)} \right)^2 \right)^{1/2} = \sup \left( \sum_{j=1}^{d/\beta} \sum_{i \in I_j} x_i^2 \left( y^T B^{(i)} \right)^2 \right)^{1/2}, \end{aligned}$$

where $I_j$ is the $j$th block of $\beta$ consecutive integers between 1 and $d$. Applying Cauchy–Schwarz, we get

$$\begin{aligned}\sigma &\le \sup_{\substack{y\in\ell_2^k\\ \|y\|=1}} \left(\sum_{j=1}^{d/\beta} \|x_{I_j}\|_4^2 \|y^T\hat{B}\|_4^2\right)^{1/2}\\ &= \left(\sup \|y^T\hat{B}\|_4\right)\|x\|_{(4,2)} = \|\hat{B}^T\|_{2\to 4}\|x\|_{(4,2)}, \qquad (8.11)\end{aligned}$$

where $\|\cdot\|_{(p_1,p_2)}$ is defined by

$$\|x\|_{(p_1,p_2)} = \left(\sum_{j=1}^{d/\beta} \|x_{I_j}\|_{p_1}^{p_2}\right)^{1/p_2},$$

and $x_{I_j} \in \ell_{p_1}^{\beta}$ is the projection of $x$ onto the set of coordinates $I_j$. Our goal, as in Sect. 5, is to get $\sigma = O(k^{-1/2})$. By the properties of dual BCH code matrices (Lemma 5.1), we readily have that $\|\hat{B}^T\|_{2\to 4} = O((f_{\mathrm{BCH}}(k)k^{\delta})^{1/4}k^{-1/2})$, which is $O(k^{\delta/4})$ by our construction. We now need to somehow "ensure" that $\|x\|_{(4,2)} = O(k^{-1/2-\delta/4})$ in order to complete the construction.

As before, we cannot directly control $x$ (and its norms), but we can multiply it by random orthogonal matrices without losing $\ell_2$ information. Let $H'$ be a block diagonal $d \times d$ matrix with $d/\beta$ blocks of the Walsh–Hadamard matrix $H_\beta$:

$$H' = \begin{pmatrix} H_\beta & & & \\ & H_\beta & & \\ & & \ddots & \\ & & & H_\beta \end{pmatrix}.$$

Let $D'$ be a random diagonal $d \times d$ matrix over $\pm 1$. The random matrix $H'D'$ is orthogonal. We study the random variable $X' = \|H'D'x\|_{(4,2)}$. Let $M'$ be the matrix with the $i$th column $M'^{(i)}$ defined as $x_i H'^{(i)}$. We notice that $X'$ is the norm of the Rademacher random variable in $\ell_{(4,2)}^d$ corresponding to $M$.

*Remark* The results on Rademacher random variables, presented in Sect. 3, apply also to "nonstandard" norms such as $\|\cdot\|_{(p_1,p_2)}$. The dual of $\|\cdot\|_{(p_1,p_2)}$ is $\|\cdot\|_{(q_1,q_2)}$, where $q_1, q_2$ are the usual dual norm indices of $p_1, p_2$, respectively. It is an exercise to check that $\|x\|_{(p_1,p_2)} = \sup_{\|y\|_{(q_1,q_2)}=1} x^T y$. We compute the deviation $\sigma'$ and a median $\mu'$ of $X'$ (as we did in (5.4)):

$$\begin{aligned}\sigma' &= \|M\|_{2\to(4,2)} = \|M^T\|_{(4/3,2)\to 2}\\ &= \sup_{\substack{y\in\ell_{(4/3,2)}^k\\ \|y\|=1}} \left(\sum_i x_i^2 \left(y^T H^{(i)}\right)^2\right)^{1/2}\\ &= \sup\left(\sum_{j=1}^{d/\beta}\sum_{i\in I_j} x_i^2 \left(y^T H'^{(i)}\right)^2\right)^{1/2}\end{aligned}$$

$$
\begin{aligned}
&\le \sup\left(\sum_{j=1}^{d/\beta}\|x_{I_j}\|_4^2\|y_{I_j}^T H_\beta\|_4^2\right)^{1/2}\\
&\le \sup\left(\sum_{j=1}^{d/\beta}\|x_{I_j}\|_4^2\|y_{I_j}\|_{4/3}^2\|H_\beta^T\|_{4/3\to 4}^2\right)^{1/2}\\
&= \|H_\beta\|_{4/3\to 4}\sup\left(\sum_{j=1}^{d/\beta}\|x_{I_j}\|_4^2\|y_{I_j}\|_{4/3}^2\right)^{1/2},
\end{aligned}
$$

where the first inequality is Cauchy–Schwarz. By the inequality $(\sum_j A_j)^{1/2} \le \sum_j A_j^{1/2}$ holding for all nonnegative $A_1, A_2, \ldots$, we get

$$
\begin{aligned}
\sigma' &\le \|H_\beta\|_{4/3\to 4} \sup_{\substack{y\in\ell^k_{(4/3,2)}\\ \|y\|=1}} \sum_{j=1}^{d/\beta}\|x_{I_j}\|_4\|y_{I_j}\|_{4/3}\\
&\le \|H_\beta\|_{4/3\to 4}\|x\|_{(4,2)}.
\end{aligned}
$$

(The rightmost inequality is from the fact that $\sum_{j=1}^{d/\beta}\|y_{I_j}\|_{4/3}^2 = 1$ and the definition of $\|x\|_{(4,2)}$.) By Hausdorff–Young, $\|H_\beta\|_{4/3\to 4} \le \beta^{-1/4} = O(k^{-1/2-\delta/4})$, hence $\sigma' = O(k^{-1/2-\delta/4}\|x\|_{(4,2)})$. Any median $\mu'$ of $X'$ is $O(k^{-1/2-\delta/4})$ (details omitted). Applying Theorem 3.3, we get that for all $t \ge 0$,

$$
\Pr\bigl[X' > \mu' + t\bigr] \le 4e^{-t^2/(8\sigma'^2)} \le \hat{c}_1 \exp\bigl\{-\hat{c}_2 t^2 k^{1+\delta/2}/\|x\|_{(4,2)}^2\bigr\},
$$

for some global $\hat{c}_1, \hat{c}_2 > 0$. Setting $t = \Theta(\|x\|_{(4,2)}k^{-\delta/4})$, we get that

$$
\Pr\bigl[\|H'D'x\|_{(4,2)} > \mu' + t\bigr] = O\bigl(e^{-k}\bigr).
$$

Similarly to the arguments leading to Lemma 5.2 and with possible readjustment of the parameter $\delta$, we get using a union bound:

**Lemma 8.1** ($\ell_{(4,2)}$ reduction for $k < d^{1/2-\delta}$) *Let $H', D'$ be as above, and let $\Phi' = H'D'$. Define $\Phi'^{(r)}$ to be a composition of $r$ i.i.d. matrices, each drawn from the same distribution as $\Phi'$. Then with probability* $1 - O(e^{-k})$,

$$
\bigl\|\Phi'^{(r)}x\bigr\|_{(4,2)} = O\bigl(k^{-1/2-\delta/4}\bigr)
$$

*for* $r = \lceil 1/2\delta\rceil$.

Combining the above, the random transformation $A = BD\Phi'^{(r)}$ has the $JL$ Euclidean property for $k < d^{1/2-\delta}$ and can be applied to a vector in time $O(d\log k)$, as required. Indeed, multiplying by $\Phi'$ is done by doing a Walsh transform on $d/\beta$ blocks of size $\beta$ each, resulting in time $O(d\log k)$. Clearly the number of random bits used in choosing $A$ is $O(d)$.

## 9 Future Work

- *Lower bounds.* A lower bound on the running time of applying a random matrix with a JL property on a vector would be extremely interesting. Any nontrivial (superlinear) bound for the case $k = d^{\Omega(1)}$ will imply a lower bound on the time to compute the Fourier transform, because the bottleneck of our constructions is a Fourier transform.
- *Going beyond $k = d^{1/2-\delta}$.* As part of our work in progress, we are trying to push the result to higher values of the target dimension $k$ (the goal is a running time of $O(d \log d)$). We conjecture that this is possible for $k = d^{1-\delta}$ and have partial results in this direction.

**Acknowledgements** We thank Bernard Chazelle and Mark W. Tygert for helpful discussions or dimension reduction and Tali Kaufman for sharing her expertise in error correcting codes.

## References

1. Johnson, W.B., Lindenstrauss, J.: Extensions of Lipschitz mappings into a Hilbert space. Contemp. Math. **26**, 189–206 (1984)
2. Ailon, N., Chazelle, B.: Approximate nearest neighbors and the fast Johnson–Lindenstrauss transform. In: Proceedings of the 38st Annual Symposium on the Theory of Compututing (STOC), pp. 557–563. Seattle, WA (2006)
3. Frankl, P., Maehara, H.: The Johnson–Lindenstrauss lemma and the sphericity of some graphs. J. Combin. Theory Ser. A **44**, 355–362 (1987)
4. Indyk, P., Motwani, R.: Approximate nearest neighbors: Towards removing the curse of dimensionality. In: Proceedings of the 30th Annual ACM Symposium on Theory of Computing (STOC), pp. 604–613 (1998)
5. Matousek, J.: On variants of the Johnson–Lindenstrauss lemma. Private communication (2006)
6. Kushilevitz, E., Ostrovsky, R., Rabani, Y.: Efficient search for approximate nearest neighbor in high dimensional spaces. SIAM J. Comput. **30**(2), 457–474 (2000)
7. Littlestone, N.: Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. Mach. Learn. **2**(4), 285–318 (1988)
8. Arriaga, R.I., Vempala, S.: An algorithmic theory of learning: Robust concepts and random projection. In: FOCS '99: Proceedings of the 40th Annual Symposium on Foundations of Computer Science, p. 616. Washington, DC, USA, IEEE Computer Society (1999)
9. Indyk, P.: On approximate nearest neighbors in non-Euclidean spaces. In: Proceedings of the 39th Annual IEEE Symposium on Foundations of Computer Science (FOCS), pp. 148–155 (1998)
10. Vempala, S.: The Random Projection Method. DIMACS Series in Discrete Mathematics and Theoretical Computer Science (2004)
11. Sarlós, T.: Improved approximation algorithms for large matrices via random projections. In: Proceedings of the 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS). Berkeley, CA (2006)
12. Har-Peled, S.: A replacement for Voronoi diagrams of near linear size. In: Proceedings of the 42nd Annual IEEE Symposium on Foundations of Computer Science (FOCS), pp. 94–103. Las Vegas, Nevada, USA (2001)
13. Alon, N.: Problems and results in extremal combinatorics I. Discrete Math. **273**(1–3), 31–53 (2003)
14. Sudan, M.: Essential coding theory (class notes)
15. Khot, S.: Hardness of approximating thee shortest vector problem in lattices. In: Proceedings of the 45th Annual IEEE Symposium on Foundations of Computer Science (FOCS) (2004)
16. Ailon, N., Chazelle, B.: Lower bounds for linear degeneracy testing. J. ACM **52**(2), 157–171 (2005)
17. Razborov, A.A.: Expander codes and somewhat Euclidean sections in $\ell_1^n$. ECCC (2007)
18. Indyk, P.: Uncertainty principles, extractors, and explicit embeddings of l2 into l1. In: Proceedings of the 39th Annual ACM Symposium on the Theory of Computing (2007)

19. Artstein-Avidan, S., Milman, V.: Logarithmic reduction of the level of randomness in some probabilistic geometric constructions. SIAM J. Comput. **1**(34), 67–88 (2004)
20. Frieze, A.M., Kannan, R., Vempala, S.: Fast Monte-Carlo algorithms for finding low-rank approximations. In: IEEE Symposium on Foundations of Computer Science, pp. 370–378 (1998)
21. Drineas, P., Kannan, R.: Fast Monte-Carlo algorithms for approximate matrix multiplication. In: IEEE Symposium on Foundations of Computer Science, pp. 452–459 (2001)
22. Drineas, P., Kannan, R., Mahoney, M.: Fast Monte Carlo algorithms for matrices II: Computing a low-rank approximation to a matrix (2004)
23. Drineas, P., Kannan, R., Mahoney, M.: Fast Monte Carlo algorithms for matrices III: Computing a compressed approximate matrix decomposition (2004)
24. Woolfe, F., Liberty, E., Rokhlin, V., Tygert, M.: A fast randomized algorithm for the approximation of matrices. Yale Computer Science Technical Reports, YALE/DCS/TR1380 (2007)
25. Drineas, P., Mahoney, M.W., Muthukrishnan, S., Sarlos, T.: Faster least squares approximation. http://arxiv.org/abs/0710.1435 (2007)
26. Bergh, J., Lofstrom, J.: Interpolation Spaces. Springer, Berlin (1976)
27. Ledoux, M., Talagrand, M.: Probability in Banach Spaces: Isoperimetry and Processes. Springer, Berlin (1991)
28. MacWilliams, F.J., Sloane, N.J.A.: The Theory of Error Correcting Codes. North-Holland, Amsterdam (1983)
29. König, C.S.H., Jaegermann, N.T.: Projection constants of symmetric spaces and variants of Khintchine's inequality. J. Reine Angew. Math. **511**, 1–42 (1999)