Sketching via hashing: from heavy hitters to compressed sensing to sparse fourier transform

Citation	Indyk, Piotr. “Sketching via Hashing: from heavy hitters to compressed sensing to sparse fourier transform.” in Proceedings of the 32nd Symposium on Principles of Database Systems - PODS ’13 (2013). June 22–27, 2013, New York, New York, USA. ACM. p.87-90.
As Published	http://dx.doi.org/10.1145/2463664.2465217
Publisher	Association for Computing Machinery
Version	Author's final manuscript
Accessed	Wed Nov 21 01:42:53 EST 2018
Citable Link	http://hdl.handle.net/1721.1/86898
Terms of Use	Creative Commons Attribution-Noncommercial-Share Alike
Detailed Terms	http://creativecommons.org/licenses/by-nc-sa/4.0/

Sketching via Hashing: from Heavy Hitters to Compressive Sensing to Sparse Fourier Transform

Piotr Indyk
MIT
indyk@mit.edu

Categories and Subject Descriptors

F.2.1 [**Analysis of Algorithms and Problem Complexity**]: Numerical Algorithms and Problems—*Computations on matrices; Computation of transforms (e.g., fast Fourier transform)*

General Terms

Algorithms

Keywords

Dimensionality reduction, hashing, sparse Fourier transform, sparsity

1. INTRODUCTION

Sketching via hashing is a popular and useful method for processing large data sets. Its basic idea is as follows. Suppose that we have a large multi-set of elements $S = \{a_1, \ldots a_s\} \subset \{1 \ldots n\}$, and we would like to identify the elements[1] that occur "frequently" in S. The algorithm starts by selecting a hash function h that maps the elements into an array $c[1 \ldots m]$. The array entries are initialized to 0. Then, for each element $a \in S$, the algorithm increments[2] $c[h(a)]$. At the end of the process, each array entry $c[j]$ contains the count of all data elements $a \in S$ mapped to j. It can be observed that if an element a occurs frequently enough in the data set S, then the value of the counter $c[h(a)]$ must be large. That is, "frequent" elements are mapped to "heavy" buckets. By identifying the elements mapped to heavy buckets and repeating the process several times, one can efficiently recover the frequent elements, possibly together with a few extra ones (false positives).

Variants of this method have originated in several fields, including databases [FSGM+98, CM03a, CM03b], computer networks [FCAB98, EV02] (cf. [BM04]) and theoretical computer science [CCF02, GGI+02b, CM04]. One of the key features of this method is that it allows to *approximate* the counts of the elements using very limited storage while making only a single pass over the data. As a result, the method has become one of the staples in the field of *data stream computing* [Mut05]. However, this was just the beginning. Over the last decade, this approach has been used to design improved algorithms for remarkably diverse tasks such as *compressive sensing*, *dimensionality reduction* and *sparse Fourier transforms.* In this survey we give a brief overview of how hashing is used in the aforementioned applications.

In order to apply the approach to those tasks, the first step is to view the hashing process as a linear mapping of the characteristic vector x of the set S to the vector c. Specifically, for any $j = 1 \ldots m$, $c[j] = \sum_{a:h(a)=j} x_a$. This can be written as $c = Ax$ where A is a sparse binary $m \times n$ matrix. The algorithmic benefit of using such mappings is due to the sparsity of the matrix A (which makes it easy to perform various tasks such as matrix-vector multiplication efficiently) as well as the overall simplicity of the hashing process.

2. COMPRESSED SENSING

In compressed sensing [Don06, CRT06] one is given the vector Ax and the goal is to recover an approximation x' to x that is k-sparse, i.e., that has at most k non-zero entries. The approximation should (approximately) minimize the error $\|x' - x\|_p$ for some choice of the ℓ_p norm. Note that for any value of p, the error $\|x - x'\|_p$ is minimized when the approximation x' consists of the k largest (in magnitude) coefficients of x. This problem has numerous applications in signal processing or imaging, where signals are quite sparse, possibly after applying an appropriate change-of-basis transform. In those applications compressed sensing allows one to recover a good approximation to a signal x from only few "measurements" Ax. In particular, the result of [CRT06] shows that one can recover a k-sparse approximation to x using only $m = O(k \log(n/k))$ measurements, and it is known that this bound cannot be improved [Don06, DIPW10, FPRU10]. The bound is achieved using matrices A with random i.i.d. Gaussian or Bernoulli entries. Unfortunately, any operation on such matrices takes $O(nm)$

[1] These elements are often referred to as *heavy hitters* or *elephants.*

[2] Typically, the value is incremented by 1. However, some algorithms such as Count Sketch [CCF02] or the pre-identification procedure of [GGI+02b] use randomly chosen increments.

PODS'13, June 22–27, 2013, New York, New York, USA.

time, which makes the recovery algorithms somewhat slow for high values of n.[3]

It was observed in [CM06] (cf. [GI10]) that the algorithm of [CCF02] yields a recovery procedure and a matrix A with $O(k \log n)$ measurements, which is not too far from the optimal bound (although the recovery procedure is only correct with high probability). At the same time, thanks to the sparsity of the matrix A, the approximation x' can be computed in only $O(n \log n)$ time. Compressive sensing via sparse matrices has attracted a considerable interest in the literature, see e.g., [SBB06, CM06, WGR07, GSTV07, XH07, SBB10, WWR10, KDXH11, Ind08, LMP$^+$08, BGI$^+$08, IR08, BIR08, JXHC09, GLPS10, GM11, PR12, BJCC12] or a survey [GI10]. In particular, the results of [BGI$^+$08, IR08, BIR08, GLPS10] show that sparse matrices can match the optimal $O(k \log(n/k))$ measurement bound achieved via fully random matrices while supporting faster algorithms, albeit in some cases providing somewhat weaker approximation guarantees.

3. DIMENSIONALITY REDUCTION

A mapping from x to Ax can be also used to reduce the dimensionality of general (non-sparse) vectors x, as per the Johnson-Lindenstrauss theorem [JL84]. The original theorem used random dense matrices, which necessitated $O(nm)$ matrix-vector multiplication time. Faster dimensionality reduction is possible by using structured matrices that support much faster matrix-vector multiplication procedures [AC10, AL11, KW11, NPW12], but the reduced dimension is either sub-optimal or restricted. Moreover, the running times of those procedures do not scale with the the sparsity of the vector x. In contrast, the line of research on sparse dimensionality reduction matrices [SPD$^+$09, WDL$^+$09, DKS10, BOR10, KN12] has led to matrices with optimal reduced dimension bounds that are supported by algorithms with runtime $O(\epsilon km)$, where k is the number of non-zero entries in x and $\epsilon > 0$ is an approximation parameter that is arbitrarily close to 0. Using such matrices, [CW13] (see also [NN12, MM13]) recently showed almost linear time approximate algorithms for sparse regression and low-rank approximation, the key problems in numerical linear algebra.

4. SPARSE FOURIER TRANSFORM

The Discrete Fourier Transform (DFT) maps an n-dimensional signal x sampled in time domain into an n-dimensional spectrum $\hat{x}$. The widely used Fast Fourier Transform algorithm performs this task in $O(n \log n)$ time. It is not known whether this algorithm can be further improved. However, it is known that one can compute DFT significantly faster for signals whose spectrum is (approximately) sparse. Such sparsity is common for many data sets occurring in signal processing, imaging and communication. For such signals, one may hope for faster algorithms.

The first algorithms of this type were designed for the Hadamard Transform, i.e., the Fourier transform over the Boolean cube [KM91, Lev93] (cf. [GL89, Gol99]). Soon, algorithms for the complex Fourier transform were discovered as well [Man92, GGI$^+$02a, AGS03, GMS05, Iwe10, Iwe12, Aka10, HIKP12b, HIKP12a, LWC12, BCG$^+$12, GHI$^+$13, HKPV13]. In particular, the algorithm given in [HIKP12a] computes the DFT of a signal with k-sparse spectrum in $O(k \log n)$ time. Note that this running time improves over the FFT as long as $k = o(n)$. In fact, for low values of k the running time is *sub-linear* in n, i.e., the algorithm does not even read its input. Instead, it infers the large Fourier coefficients by randomly sampling the signal x.

Perhaps surprisingly, most of the aforementioned algorithms (notably [Lev93, GGI$^+$02a, GMS05, Iwe10, Iwe12, Aka10, HIKP12b, HIKP12a, LWC12, BCG$^+$12, GHI$^+$13, HKPV13])[4] use sketching via hashing, albeit in the frequency domain. Specifically, the algorithms utilize multiple band-pass filters which bin the spectrum coefficients into a number of "buckets". The process is randomized to ensure that each coefficient is mapped to a "random" bucket and that two large coefficients are not likely to collide. A somewhat distinctive feature of this process is that can yield "leaky" buckets, where a large coefficient affects not only the bucket it is mapped into, but also the nearby ones. Fortunately, thanks to a careful filter design, the leakage can be made negligible [HIKP12b, HIKP12a, BCG$^+$12, HKPV13] or even be completely eliminated [Iwe10, Iwe12, LWC12, GHI$^+$13].

For further overview of recent work on sub-linear algorithms for sparse Fourier transform as well their applications, see [GIKR13].

5. REFERENCES

[AC10] Nir Ailon and Bernard Chazelle. Faster dimension reduction. *Communications of the ACM*, 53(2):97–104, 2010.

[AGS03] A. Akavia, S. Goldwasser, and S. Safra. Proving hard-core predicates using list decoding. *FOCS*, 44:146–159, 2003.

[Aka10] A. Akavia. Deterministic sparse Fourier approximation via fooling arithmetic progressions. *COLT*, pages 381–393, 2010.

[AL11] Nir Ailon and Edo Liberty. An almost optimal unrestricted fast Johnson-Lindenstrauss transform. *SODA*, pages 185–191, 2011.

[BCG$^+$12] P. Boufounos, V. Cevher, A. C. Gilbert, Y. Li, and M. J. Strauss. What's the frequency, kenneth?: Sublinear Fourier sampling off the grid. *RANDOM/APPROX*, 2012.

[BGI$^+$08] R. Berinde, A. Gilbert, P. Indyk, H. Karloff, and M. Strauss. Combining geometry and combinatorics: a unified approach to sparse signal recovery. *Allerton*, 2008.

[BIR08] R. Berinde, P. Indyk, and M. Ruzic. Practical near-optimal sparse recovery in the l_1 norm. *Allerton*, 2008.

[BJCC12] Mayank Bakshi, Sidharth Jaggi, Sheng Cai, and Minghua Chen. Sho-fa: Robust compressive sensing with order-optimal complexity, measurements, and bits. *arXiv preprint arXiv:1207.2335*, 2012.

[BM04] Andrei Broder and Michael Mitzenmacher. Network applications of bloom filters: A

[3] This issue can be alleviated by using random *structured* matrices, which support matrix-vector product in $O(n \log n)$ time [CT06, RV06, CGV13, NPW12]. However, even the best construction due to [NPW12] requires the number of measurements to be $O(\log^2 n)$ times larger than optimal, at least in theory.

[4] See also [GOS$^+$11] which applied a similar method to the related problem of testing Fourier sparsity.

survey. *Internet Mathematics*, 1(4):485–509, 2004.

[BOR10] Vladimir Braverman, Rafail Ostrovsky, and Yuval Rabani. Rademacher chaos, random eulerian graphs and the sparse Johnson-Lindenstrauss transform. *arXiv preprint arXiv:1011.2590*, 2010.

[CCF02] M. Charikar, K. Chen, and M. Farach-Colton. Finding frequent items in data streams. *ICALP*, 2002.

[CGV13] Mahdi Cheraghchi, Venkatesan Guruswami, and Ameya Velingker. Restricted isometry of Fourier matrices and list decodability of random linear codes. *SODA*, 2013.

[CM03a] Saar Cohen and Yossi Matias. Spectral bloom filters. *SIGMOD*, pages 241–252, 2003.

[CM03b] Graham Cormode and S. Muthukrishnan. What's hot and what's not: tracking most frequent items dynamically. *PODS*, pages 296–306, 2003.

[CM04] G. Cormode and S. Muthukrishnan. Improved data stream summaries: The count-min sketch and its applications. *LATIN*, 2004.

[CM06] G. Cormode and S. Muthukrishnan. Combinatorial algorithms for Compressed Sensing. *Proc. 40th Ann. Conf. Information Sciences and Systems*, Mar. 2006.

[CRT06] E. J. Candès, J. Romberg, and T. Tao. Stable signal recovery from incomplete and inaccurate measurements. *Comm. Pure Appl. Math.*, 59(8):1208–1223, 2006.

[CT06] Emmanuel J Candes and Terence Tao. Near-optimal signal recovery from random projections: Universal encoding strategies? *Information Theory, IEEE Transactions on*, 52(12):5406–5425, 2006.

[CW13] Kenneth L Clarkson and David P Woodruff. Low rank approximation and regression in input sparsity time. *STOC*, 2013.

[DIPW10] K. Do Ba, P. Indyk, E. Price, and D. Woodruff. Lower bounds for sparse recovery. *SODA*, 2010.

[DKS10] Anirban Dasgupta, Ravi Kumar, and Tamás Sarlós. A sparse Johnson-Lindenstrauss transform. *STOC*, pages 341–350, 2010.

[Don06] D. L. Donoho. Compressed Sensing. *Information Theory, IEEE Transactions on*, 52(4):1289–1306, Apr. 2006.

[EV02] Cristian Estan and George Varghese. New directions in traffic measurement and accounting. *SIGCOMM*, 2002.

[FCAB98] Li Fan, Pei Cao, Jussara M. Almeida, and Andrei Z. Broder. Summary cache: A scalable wide-area web cache sharing protocol. *SIGCOMM*, pages 254–265, 1998.

[FPRU10] S. Foucart, A. Pajor, H. Rauhut, and T. Ullrich. The gelfand widths of lp-balls for $0 < p \leq 1$. *preprint*, 2010.

[FSGM$^+$98] Min Fang, Narayanan Shivakumar, Hector Garcia-Molina, Rajeev Motwani, and Jeffrey D Ullman. Computing iceberg queries efficiently. *VLDB*, 1998.

[GGI$^+$02a] A. Gilbert, S. Guha, P. Indyk, M. Muthukrishnan, and M. Strauss. Near-optimal sparse Fourier representations via sampling. *STOC*, 2002.

[GGI$^+$02b] A. C. Gilbert, S. Guha, P. Indyk, Y. Kotidis, S. Muthukrishnan, and M. J. Strauss. Fast, small-space algorithms for approximate histogram maintenance. *STOC*, 2002.

[GHI$^+$13] Badih Ghazi, Haitham Hassanieh, Piotr Indyk, Dina Katabi, Eric Price, and Lixin Shi. Sample-optimal average-case sparse fourier transform in two dimensions. *arXiv preprint arXiv:1303.1209*, 2013.

[GI10] A. Gilbert and P. Indyk. Sparse recovery using sparse matrices. *Proceedings of IEEE*, 2010.

[GIKR13] Anna Gilbert, Piotr Indyk, Dina Katabi, and Ramesh Raskar. Workshop on sparse Fourier transform etc. *http://groups.csail.mit.edu/netmit/sFFT/workshop.html*, 2013.

[GL89] O. Goldreich and L. Levin. A hard-corepredicate for all one-way functions. *STOC*, pages 25–32, 1989.

[GLPS10] A. C. Gilbert, Y. Li, E. Porat, and M. J. Strauss. Approximate sparse recovery: optimizing time and measurements. *STOC*, pages 475–484, 2010.

[GM11] Michael T Goodrich and Michael Mitzenmacher. Invertible bloom lookup tables. In *Allerton*, pages 792–799, 2011.

[GMS05] A. Gilbert, M. Muthukrishnan, and M. Strauss. Improved time bounds for near-optimal space Fourier representations. *SPIE Conference, Wavelets*, 2005.

[Gol99] O. Goldreich. Modern cryptography, probabilistic proofs and pseudorandomness. *Algorithms and Combinatorics*, 17, 1999.

[GOS$^+$11] Parikshit Gopalan, Ryan O'Donnell, Rocco A. Servedio, Amir Shpilka, and Karl Wimmer. Testing fourier dimensionality and sparsity. *SIAM J. Comput.*, 40(4):1075–1100, 2011.

[GSTV07] A. C. Gilbert, M. J. Strauss, J. A. Tropp, and R. Vershynin. One sketch for all: fast algorithms for compressed sensing. *STOC*, pages 237–246, 2007.

[HIKP12a] H. Hassanieh, P. Indyk, D. Katabi, and E. Price. Near-optimal algorithm for sparse Fourier transform. *STOC*, 2012.

[HIKP12b] H. Hassanieh, P. Indyk, D. Katabi, and E. Price. Simple and practical algorithm for sparse Fourier transform. *SODA*, 2012.

[HKPV13] Sabine Heider, Stefan Kunis, Daniel Potts, and Michael Veit. A sparse prony fft. 2013.

[Ind08] P. Indyk. Explicit constructions for compressed sensing of sparse signals. *SODA*, 2008.

[IR08] P. Indyk and M. Ruzic. Near-optimal sparse recovery in the l_1 norm. *FOCS*, 2008.

[Iwe10] M. A. Iwen. Combinatorial sublinear-time Fourier algorithms. *Foundations of*

Computational Mathematics, 10:303–338, 2010.

[Iwe12] M.A. Iwen. Improved approximation guarantees for sublinear-time Fourier algorithms. *Applied And Computational Harmonic Analysis*, 2012.

[JL84] W. B. Johnson and J. Lindenstrauss. Extensions of Lipschitz mapping into Hilbert space. *Conf. in modern analysis and probability*, 26:189–206, 1984.

[JXHC09] S. Jafarpour, W. Xu, B. Hassibi, and A. R. Calderbank. Efficient and robust compressed sensing using high-quality expander graphs. *Information Theory, IEEE Transactions on*, 55(9):4299–4308, 2009.

[KDXH11] M. A. Khajehnejad, A. G. Dimakis, W. Xu, and B. Hassibi. Sparse recovery of positive signals with minimal expansion. *Signal Processing, IEEE Transactions on*, 59(1):196–208, 2011.

[KM91] E. Kushilevitz and Y. Mansour. Learning decision trees using the Fourier spectrum. *STOC*, 1991.

[KN12] Daniel M Kane and Jelani Nelson. Sparser Johnson-Lindenstrauss transforms. *SODA*, pages 1195–1206, 2012.

[KW11] Felix Krahmer and Rachel Ward. New and improved Johnson-Lindenstrauss embeddings via the restricted isometry property. *SIAM Journal on Mathematical Analysis*, 43(3):1269–1281, 2011.

[Lev93] L.A. Levin. Randomness and non-determinism. *J. Symb. Logic*, 58(3):1102–1103, 1993.

[LMP$^+$08] Y. Lu, A. Montanari, B. Prabhakar, S. Dharmapurikar, and A. Kabbani. Counter braids: A novel counter architecture for per-flow measurement. *SIGMETRICS*, 2008.

[LWC12] D. Lawlor, Y. Wang, and A. Christlieb. Adaptive sub-linear time Fourier algorithms. *arXiv:1207.6368*, 2012.

[Man92] Y. Mansour. Randomized interpolation and approximation of sparse polynomials. *ICALP*, 1992.

[MM13] Xiangrui Meng and Michael W Mahoney. Low-distortion subspace embeddings in input-sparsity time and applications to robust linear regression. *STOC*, 2013.

[Mut05] S Muthukrishnan. *Data streams: Algorithms and applications*. Now Publishers Inc, 2005.

[NN12] Jelani Nelson and Huy L Nguyên. Osnap: Faster numerical linear algebra algorithms via sparser subspace embeddings. *arXiv preprint arXiv:1211.1002*, 2012.

[NPW12] Jelani Nelson, Eric Price, and Mary Wootters. New constructions of RIP matrices with fast multiplication and fewer rows. *CoRR*, abs/1211.0986, 2012.

[PR12] Sameer Pawar and Kannan Ramchandran. A hybrid dft-ldpc framework for fast, efficient and robust compressive sensing. In *Allerton*, pages 1943–1950, 2012.

[RV06] M. Rudelson and R. Veshynin. Sparse reconstruction by convex relaxation: Fourier and Gaussian measurements. *Proc. 40th Ann. Conf. Information Sciences and Systems*, Mar. 2006.

[SBB06] S. Sarvotham, D. Baron, and R. G. Baraniuk. Sudocodes - fast measurement and reconstruction of sparse signals. *IEEE International Symposium on Information Theory*, 2006.

[SBB10] S. Sarvotham, D. Baron, and R. G. Baraniuk. Bayesian compressive sensing via belief propagation. *Signal Processing, IEEE Transactions on*, 58(1):269–280, 2010.

[SPD$^+$09] Qinfeng Shi, James Petterson, Gideon Dror, John Langford, Alex Smola, and SVN Vishwanathan. Hash kernels for structured data. *The Journal of Machine Learning Research*, 10:2615–2637, 2009.

[WDL$^+$09] Kilian Weinberger, Anirban Dasgupta, John Langford, Alex Smola, and Josh Attenberg. Feature hashing for large scale multitask learning. *ICML*, pages 1113–1120, 2009.

[WGR07] Wei Wang, Minos Garofalakis, and Kannan Ramchandran. Distributed sparse random projections for refinable approximation. *IPSN*, pages 331–339, 2007.

[WWR10] W. Wang, M. J. Wainwright, and K. Ramchandran. Information-theoretic limits on sparse signal recovery: Dense versus sparse measurement matrices. *Information Theory, IEEE Transactions on*, 56(6):2967–2979, 2010.

[XH07] W. Xu and B. Hassibi. Efficient compressive sensing with determinstic guarantees using expander graphs. *IEEE Information Theory Workshop*, 2007.