

---

**io-port 02199418****Ganguly, Sumit****Estimating frequency moments of data streams using random linear combinations.**

Jansen, Klaus (ed.) et al., Approximation, randomization, and combinatorial optimization. Algorithms and techniques. 7th international workshop on approximation algorithms for combinatorial optimization problems, APPROX 2004 and 8th international workshop on randomization and computation, RANDOM 2004, Cambridge, MA, USA, August 22-24, 2004. Proceedings. Berlin: Springer (ISBN 3-540-22894-2/pbk). Lecture Notes in Computer Science 3122, 369-380 (2004).

Summary: The problem of estimating the  $k^{\text{th}}$  frequency moment  $F_k$  for any non-negative  $k$ , over a data stream by looking at the items exactly once as they arrive, was considered in a seminal paper by *N. Alon*, *Y. Matias* and *M. Szegedy* [“The space complexity of approximating the frequency moments”, in: Proceedings of the 28th annual ACM symposium on the theory of computing (STOC). Philadelphia, PA, USA, May 22–24, 1996. New York, NY: ACM, 20–29 (1996; Zbl 0922.68057) and J. Comput. Syst. Sci. 58, 137–147 (1999; Zbl 0938.68153)]. The space complexity of their algorithm is  $\tilde{O}(n^{1-\frac{1}{k}})$ . For  $k > 2$ , their technique does not apply to data streams with arbitrary insertions and deletions. In this paper, we present an algorithm for estimating  $F_k$  for  $k > 2$ , over general update streams whose space complexity is  $\tilde{O}(n^{1-\frac{1}{k-1}})$  and time complexity of processing each stream update is  $\tilde{O}(1)$ . Recently, an algorithm for estimating  $F_k$  over general update streams with similar space complexity has been published by *D. Coppersmith* and *R. Kumar* [“An improved data stream algorithm for estimating frequency moments”, in: Proceedings of the 15th annual ACM-SIAM symposium on discrete algorithms, SODA 2004, New Orleans, LA, USA, January 11–14, 2004, 151–156 (2004)]. Our technique is, (a) basically different from the technique used by Coppersmith and Kumar [loc. cit.], (b) is simpler and symmetric, and, (c) is more efficient in terms of the time required to process a stream update ( $\tilde{O}(1)$  compared with  $\tilde{O}(n^{1-\frac{1}{k-1}})$ ).

doi:10.1007/b99805