Finding and counting cliques and independent sets in r-uniform hypergraphs

Raphael Yuster *
Department of Mathematics
University of Haifa
Haifa 31905, Israel

Abstract

We present a matrix multiplication based algorithm for counting the number of (induced) occurrences of a fixed r-uniform hypergraph in a larger hypergraph. In many cases, the running time is better than that of the naïve algorithm. We also present several useful applications of the algorithm, such as determining the dominant color among monochromatic simplices in a red-blue edge-colored hypergraph, approximating the number of independent simplices in a random hypergraph, and counting induced occurrences of a given 3-uniform k-vertex hypergraph in a larger k-clique free hypergraph.
Keywords: algorithms, hypergraphs, fast matrix multiplication

1 Introduction

Finding and counting independent sets or other types of induced graphs or hypergraphs are classical problems in complexity theory and algorithmic combinatorics. Finding a maximum independent set is NP-Hard, and also hard to approximate [7] even in random graphs [8]. This problem is also conjectured to be *not* fixed parameter tractable [4]. All known algorithms for finding and counting induced sub(hyper)graphs on k vertices in a given n vertex (hyper)graph have running time $n^{\Theta(k)}$. In this paper we consider the following problem.

The H counting problem:

Input: An r-uniform hypergraph G.

Output: The number of (induced) copies of H in G. If at least one copy of H exists in G, a witness certifying this fact should be exhibited.

If H has k vertices, the naïve algorithm can easily solve the problem (in both the induced or non-induced versions) in $O(n^k)$ time by exhaustive search.

*e-mail: raphy@research.haifa.ac.il

In the sequel we shall use n and m to denote the number of vertices and edges of an input hypergraph G. Nešetřil and Poljak [9] presented an algorithm for the H counting problem in the graph-theoretic case. Their algorithm uses fast matrix multiplication and its running time is $O(n^{\omega\lfloor k/3\rfloor+(k \bmod 3)})$ where k is the number of vertices of H and ω is the exponent of matrix multiplication. Coppersmith and Winograd proved in [3] that $\omega < 2.376$. In case $k \bmod 3 \neq 0$ it is possible to slightly improve the exponent using fast *rectangular* matrix multiplication [5]. Already for 3-uniform hypergraphs the problem becomes extremely difficult. The only known algorithm for the H counting problem is the naïve algorithm. In fact, even the very special case of deciding whether a given n-vertex 3-uniform hypergraph contains a K_4^3 (K_k^r denotes the complete r-uniform hypergraph with k vertices) is not known to be solved in $o(n^4)$ time.

Our first result is a generalization of the method of Nešetřil and Poljak [9] to a hypergraph setting. This generalization solves the H counting problem faster than the naïve algorithm for a certain class of labeled hypergraphs H, but not all possible ones. We present this generalization in Section 2. Coupled with several combinatorial identities, some linear algebraic facts and some probabilistic arguments, our new algorithm has several useful applications, which we now list.

Recall that a *simplex* of an r-uniform hypergraph is a complete subhypergraph on $r+1$ vertices. Similarly, an *independent simplex* is an independent set with $r+1$ vertices. Consider the following decision problem:

The dominant color of monochromatic simplices:

Input: A red-blue edge-colored K_n^r.

Question: Are there more red monochromatic simplices than blue ones?

In the graph-theoretic case $r = 2$ the question is to decide, in a given red-blue edge-colored K_n, whether there are more red triangles than blue ones. This problem is obviously not more difficult that the problem of counting the number of triangles in a given graph, and hence can be solved in $O(n^\omega)$ time. However, it is, essentially, also not easier. Goodman [6] observed that the number of monochromatic triangles in a red-blue colored K_n can be determined just by examining the degree sequence of the red subgraph. Thus, if we know the *difference* between the number of red triangles and blue triangles, we also know the number of triangles in each color. One consequence of our H counting algorithm is an algorithm for the dominant color of monochromatic simplices for $r = 3$.

Theorem 1.1 *There exists an algorithm that, given a red-blue edge-colored complete* 3*-uniform hypergraph with* n *vertices, decides whether there are more red monochromatic simplices than blue ones in* $O(n^{\omega+1})$ *time.*

It is interesting to note that, unlike the graph-theoretic case, we do not know whether this problem is as difficult as the problem of counting the number of K_4^3. As mentioned before, the only known algorithm for the latter problem is the naïve $O(n^4)$ algorithm. The proof of Theorem 1.1 is presented in Section 3.

Another consequence of our algorithm is an approximation algorithm for the number of independent simplices of a random r-uniform hypergraph. Recall that $G(n,p)$ denotes the random hypergraph on n vertices, where each edge is chosen with probability p. The model $G(n,p)$ very closely resembles the model $G(n,m)$ in which each possible r-uniform hypergraph with $m = p\binom{n}{r}$ edges is equally likely (see, e.g., [2]). Suppose the input distribution to our algorithm is $G(n,p)$. The expected number of independent simplices is $\binom{n}{r+1}(1-p)^{r+1}$. Thus, a straightforward approximation algorithm exists, whose additive error is derived from bounds on large deviations using the second moment method, and the sampling algorithm, that repeatedly selects random $(r+1)$-sets of vertices and checks whether they form an independent simplex or not, cannot do better. In fact, it is not difficult to show that the additive error of the sampling algorithm is $O(n^{(r+1)/2}p^{1/2})$ (see Section 4). The following algorithm outperforms these algorithms whenever $p = o(1)$.

Theorem 1.2 *There exists an $O(n^{\omega+r-2})$ algorithm that, given an r-uniform hypergraph chosen at random from $G(n,p)$, approximates the number of independent simplices. Asymptotically almost surely, the additive error is $O(n^{(r+1)/2}p^2)$.*

Consider, for example, a 3-uniform hypergraph drawn from $G(n, n^{-0.5})$. The expected number of 4-sets that are *not* independent simplices is $\Theta(n^{3.5})$ and any sampling algorithm can, therefore, approximate the number of independent simplices up to a $\Theta(n^{1.75})$ additive error. Our algorithm, on the other hand, guarantees an $O(n)$ additive error, and has running time $O(n^{3.376})$. Notice that the naïve algorithm that enumerates all the non-independent simplices by examining all mn possible pairs of edges and vertices has expected running time $O(mn) = O(n^{3.5})$. We present the proof of Theorem 1.2 in Section 4.

Let $\mathcal{H}_{k,r}$ be the family of all r-uniform hypergraphs with k vertices. Another consequence of our algorithm is the following result, whose proof appears in Section 5.

Theorem 1.3 *If G is a 3-uniform hypergraph without a clique of order k, then, for each 3-uniform hypergraph $H \in \mathcal{H}_{k,3}$, we can determine the number of induced copies of H in G in $O(n^{k-3+\omega})$ time. In particular, we can determine the number of independent k-sets in $O(n^{k-3+\omega})$ time.*

2 An H-counting algorithm

Let $H = (V_H, E_H)$ be an r-uniform hypergraph with k vertices, and assume that the vertices are labeled $\{1,\dots,k\}$. Given an r-uniform hypergraph $G = (V_G, E_G)$ with n vertices labeled $\{1,\dots,n\}$, a labeled copy of H in G is a one-to-one mapping σ from V_H to V_G such that if $(i_1,\dots,i_r) \in E_H$ then $(\sigma(i_1),\dots,\sigma(i_r)) \in E_G$. Similarly, a labeled *induced copy* is a one-to-one mapping σ from V_H to V_G such that $(i_1,\dots,i_r) \in E_H$ if and only if $(\sigma(i_1),\dots,\sigma(i_r)) \in E_G$. Let $c(G,H)$ be the number of labeled copies of H in G and let $c^*(G,H)$ be the number of induced labeled copies of H in G. Notice that $c(G,H)$ and $c^*(G,H)$ are independent of the actual labeling used.

For $0 \leq t \leq \lfloor k/3 \rfloor$ consider the partition of $\{1, \ldots, k\}$ into four parts $P_0 = \{1, \ldots, k-3t\}$, $P_i = \{k + (i-4)t + 1, \ldots, k + (i-3)t\}$ for $i = 1, 2, 3$. We say that a labeling of H is t-good if for each edge $e \in E_H$, there exists $i \in \{1, 2, 3\}$ so that $e \cap P_i = \emptyset$. Trivially, every labeling is 0-good. However, we can sometimes do better. Consider, for example, the labeled 3-uniform hypergraph with $V_H = \{1, 2, 3, 4\}$ and with $E_H = \{(123), (124), (134)\}$. The partition $P_i = \{i+1\}$ for $i = 0, 1, 2, 3$ is 1-good. Having these definitions, we can now prove the following.

Theorem 2.1 *Let H be a labeled r-uniform hypergraph with k vertices, and assume the labeling is t-good. Then, given a labeled n-vertex r-uniform hypergraph G we can compute $c(G, H)$ in $O(n^{k-(3-\omega)t})$. A copy of H in G, if exists, can also be produced.*

Proof: Let V_G^{k-3t} denote the family of all ordered subsets of $k - 3t$ vertices of G. Fix an ordered set $K \in V_G^{k-3t}$. Let F_K be the family of all ordered sets of t vertices from $V_G \setminus K$. Clearly, $|F_K| = (n-k+3t)(n-k+3t-1)\cdots(n-k+2t+1) < n^t$. We create three matrices, denoted A_{12}^K, A_{13}^K and A_{23}^K. The rows and columns of each of the matrices are indexed by F_K. For $U_i, U_j \in F_K$ we determine the value of $A_{ij}^K(U_i, U_j)$ as follows. For each edge $(w_1, \ldots, w_r) \in E_H$ for which $w_s \in P_0 \cup P_i \cup P_j$ for all $s = 1, \ldots, r$, we consider the mapping where w_s is mapped to $v_s \in V_G$ according to the following rule. If w_s is the ℓ'th vertex of P_0 then v_s is the ℓ'th vertex of K. If w_s is the ℓ'th vertex of P_i then v_s is the ℓ'th vertex of U_i. If w_s is the ℓ'th vertex of P_j then v_s is the ℓ'th vertex of U_j. Now, if for all edges $(w_1, \ldots, w_r) \in E_H$ for which $w_s \in P_0 \cup P_i \cup P_j$ for all $s = 1, \ldots, r$ the corresponding $(v_1, \ldots, v_r) \in E_G$ then $A_{ij}^K(U_i, U_j) = 1$. Otherwise, $A_{ij}^K(U_i, U_j) = 0$. Notice that the value of each cell of A_{ij}^K can be determined in constant time, using the usual adjacency representation of G as an r-dimensional array.

Consider the matrix product $A^K = A_{12}^K \times A_{23}^K$. Suppose $A^K(U_1, U_3) = p$ and also $A_{13}^K(U_1, U_3) = 1$. This means that there are p distinct elements of F_K, say, $U_2^1, \ldots, U_2^p$, so that for each $1 \leq q \leq p$, the mapping which orderly maps P_0 to K, P_1 to U_1, P_2 to U_2^q and P_3 to U_3 corresponds to a labeled copy of H in G (we use here the fact that no edge of H intersects P_1, P_2 and P_3 simultaneously). Clearly, each such labeled copy is counted precisely once in this way. It follows that

$$c(G, H) = \sum_{K \subset V_G^{k-3t}} \sum_{U_1 \in F_K} \sum_{U_3 \in F_K} A_{13}^K(U_1, U_3) A^K(U_1, U_3).$$

Since $|F| < n^t$, each matrix product can be performed in $O(n^{\omega t})$ time. As there are $|V_G^{k-3t}| < n^{k-3t}$ matrix products, we have that $c(G, H)$ can be computed in $O(n^{k-3t+\omega t})$ time. Finally, if $c(G, H) > 0$ then we *know* of at least one triple (K, U_1, U_3) for which $A_{13}^K(U_1, U_3) A^K(U_1, U_3) > 0$ and hence a corresponding witness can be produced in $O(n^t) < O(n^{k-3t+\omega t})$ additional time. ∎

3 Monochromatic simplices in a red-blue coloring of K_n^3

For $i = 0, \ldots, 4$ let H_i be the 3-uniform hypergraph with 4 vertices and i edges. Notice that H_0 is an independent simplex and H_4 is a simplex. As denoted in the previous section, given an n-vertex 3-uniform hypergraph G, let $c(G, H_i)$ (resp. $c^*(G, H_i)$) denote the number of (resp. induced) labeled copies of H_i in G. For simplicity, let $x_i = c^*(G, H_i)$ and let $y_i = c(G, H_i)$. Recall that for a hypergraph H, $aut(H)$ denotes the automorphism group of H. Specifically, $|aut(H_0)| = |aut(H_4)| = 4! = 24$, $|aut(H_1)| = |aut(H_3)| = 6$ and $|aut(H_2)| = 4$.

If H is a hypergraph and $\mathcal{H}$ is the set of pairwise non-isomorphic hypergraphs containing H on the same set of vertices then, clearly, the following combinatorial identity holds.

$$c(G, H) = \sum_{H' \in \mathcal{H}} \frac{c^*(G, H')}{|aut(H')|} c(H', H). \tag{1}$$

Theorem 1.1 is an immediate corollary of the following proposition.

Proposition 3.1 *Let G be an n-vertex 3-uniform hypergraph. Then, $c^*(G, H_4) - c^*(G, H_0)$ can be computed in $O(n^{\omega+1})$ time.*

Proof: Notice first that if $j \geq i$ then $H_j \in \mathcal{H}_i$. Also notice that $c(H_i, H_0) = 24$ and $c(H_4, H_i) = 24$. Also, $c(H_3, H_2) = 12$, $c(H_3, H_1) = 18$, $c(H_2, H_1) = 12$ and $c(H_i, H_i) = |aut(H_i)|$. Thus, the following four linear equalities result from (1).

$$\begin{aligned}
y_3 &= x_4 + x_3 \\
y_2 &= x_4 + 2x_3 + x_2 \\
y_1 &= x_4 + 3x_3 + 3x_2 + x_1 \\
y_0 &= x_4 + 4x_3 + 6x_2 + 4x_1 + x_0.
\end{aligned}$$

Now, notice that $4y_3 - 6y_2 + 4y_1 - y_0 = x_4 - x_0$. Thus, it remains to show that $y_i = c(G, H_i)$ can be computed in $O(n^{\omega+1})$ time for $i = 0, 1, 2, 3$. Indeed, each of the hypergraphs H_0, H_1, H_2, H_3 has a labeling which is 1-good. Note that H_3 has a 1-good labeling by considering the labeling $V_{H_3} = \{1, 2, 3, 4\}$ and $E_{H_3} = \{(123), (124), (134)\}$. Since H_i is a spanning subgraph of H_3 for $i = 0, 1, 2$ they also have a 1-good labeling. Now, by Theorem 2.1, when $i = 0, 1, 2, 3$, $c(G, H_i)$ can be computed in $O(n^{\omega+1})$ time, as required. ■

4 Approximating the number of independent simplices of $G(n, p)$

Generalizing the notations of the previous section, for $i = 0, \ldots, r + 1$ let H_i be the r-uniform hypergraph with $r + 1$ vertices and i edges. Similarly, let $x_i = c^*(G, H_i)$ and let $y_i = c(G, H_i)$.

Using (1) we get the following system of $r+2$ linear equations, as in the proof of Proposition 3.1.

$$y_i \;=\; \sum_{j=i}^{r+1} \binom{r+1-i}{j-i} x_j \qquad i = 0, \ldots, r+1. \tag{2}$$

Now, suppose that we are *given* $x_{i+1}, \ldots, x_{r+1}$. It follows from (2) that by computing $y_0, \ldots, y_i$ we can determine $x_0, \ldots, x_i$ as well.

Lemma 4.1 *The values of y_0, y_1, y_2, y_3 can be computed in $O(n^{\omega+r-2})$ time.*

Proof: Consider the labeling of H_3 with $V_{H_3} = \{1, \ldots, r+1\}$ and $E_{H_3} = \{(1, \ldots, r), (1, \ldots, r-1, r+1), (1, \ldots, r-2, r, r+1)\}$. This labeling is 1-good. Thus, by Theorem 2.1, y_3 can be computed in $O(n^{\omega+r-2})$ time. As H_2, H_1, H_0 are subgraphs of H_3, they also have a 1-good labeling. ■

Corollary 4.2 *If we know $x_4, \ldots, x_{r+1}$ then we can determine the number of independent simplices in $O(n^{\omega+r-2})$ time. If, for all $i = 4, \ldots, r+1$, we can approximate each x_i up to an additive $O(n^\alpha)$ error where $\alpha > 0$, then we can approximate the number of independent simplices up to an additive $O(n^\alpha)$ error.*

Proof: The number of independent simplices is $x_0/(r+1)!$ and x_0 can be determined from $x_4, \ldots, x_{r+1}$ and y_0, y_1, y_2, y_3. By Lemma 4.1 this can be done in $O(n^{\omega+r-2})$ time if we know $x_4, \ldots, x_{r+1}$. The second part of the corollary follows from the first by observing that the values of the coefficients and the number of equations in the system (2) are independent of n. ■

Proof of Theorem 1.2: We can clearly approximate the values of $x_4, \ldots, x_{r+1}$ of a hypergraph drawn from the probability space $G(n,p)$. The expected value of x_i precisely satisfies:

$$E[x_i] = (r+1)!\binom{n}{r+1} p^i (1-p)^{r+1-i} < n^{r+1} p^i.$$

Using the second moment method (see, e.g., [1] Chapter 4), we have, asymptotically almost surely, that $|x_i - E[x_i]| = O(n^{(r+1)/2} p^{i/2})$. Thus, for all $i \geq 4$, we have, asymptotically almost surely, that $E[x_i]$ approximates x_i up to an $O(n^{(r+1)/2} p^2)$ additive error. By Corollary 4.2, given an input hypergraph G drawn from $G(n,p)$ we can asymptotically almost surely determine the number of independent simplices up to an $O(n^{(r+1)/2} p^2)$ additive error. ■

The approximation obtained in Theorem 1.2 should be compared to the approximation obtained by using upper bounds on large deviations. The expected number of independent simplices in $G(n,p)$ is $\binom{n}{r+1}(1-p)^{r+1}$, and hence the expected number of $(r+1)$-sets that are *not* independent simplices is $\binom{n}{r+1}(1-(1-p)^{r+1})$. By the second moment method, this expectation approximates the number of $(r+1)$-sets that are *not* independent simplices, and hence also the number of independent simplices, up to an $O(n^{(r+1)/2} p^{1/2})$ additive error.

5 Counting induced 3-uniform hypergraphs

In this section we prove Theorem 1.3. First observe that if $H \in \mathcal{H}_{k,3}$ and $H \neq K_k^3$ then H has a 1-good labeling. Indeed, this follows from the obvious fact that we can always label H so that $(k-2, k-1, k)$ is *not* an edge of H. As in Section 3, let $\mathcal{H}$ be the subset of $\mathcal{H}_{k,3}$ consisting of all the hypergraphs that contain H. The following lemma is immediate from (1).

Lemma 5.1 *If we know $c^*(G, H')$ for all $H' \in \mathcal{H}_{k,3}$ that properly contain H, and if we also know $c(G, H)$ then we can determine $c^*(G, H)$ as well.* ■

Proof of theorem 1.3: We show how to compute $c^*(G, H)$ for each $H \in \mathcal{H}_{k,3}$. The proof proceeds by inverse induction on the number of edges of H. If $H = K_k^3$ then $c^*(G, H) = 0$ by assumption. Assuming that we already computed $c^*(G, H')$ for all hypergraphs with more edges than H, then, in particular, we have computed $c^*(G, H')$ for all H' that properly contain H. We can also compute $c(G, H)$ in $O(n^{k-3+\omega})$ time by Theorem 2.1 since H has a 1-good labeling. Thus, by Lemma 5.1, we can determine $c^*(G, H)$ in $O(n^{k-3+\omega})$ time. ■

Notice that theorem 1.3 remains valid as long as we know in advance the value of $c(G, K_k^3)$. Furthermore, any approximation of the value of $c(G, K_k^3)$ up to an additive $O(n^\alpha)$ error yields an $O(n^{k-3+\omega})$ time algorithm that approximates $c^*(G, H)$ for each $H \in \mathcal{H}_{k,3}$ up to an $O(n^\alpha)$ additive error. Another useful corollary is the following.

Corollary 5.2 *Let G be a 3-uniform hypergraph with n vertices and m edges. The number of independent k-sets of G can be found in $O(m^{\lfloor k/3 \rfloor} n^{k \bmod 3} + n^{k-3+\omega})$ time.*

Proof: By considering all possible combinations of $\lfloor k/3 \rfloor$ independent edges and $k \bmod 3$ independent vertices (also independent from the edges) we can count the number of K_k^3. By Theorem 1.3 we can now also compute the number of independent k-sets in additional $O(n^{k-3+\omega})$ time. We also note that, as in Theorem 2.1, we can *find* a witness independent k-set within the same time bounds. ■

Corollary 5.2 is a bit surprising since sparse hypergraphs are likely to have more independent sets, and computing the precise number seems, at first glance, to be more difficult than in the dense case. For example if G is a 3-uniform hypergraph with $m = O(n^{2.376})$ edges, we can compute the number of independent simplices in $O(n^{3.376})$ time. We cannot say the same if, e,g., $m = \Theta(n^3)$.

6 An open problem

We have demonstrated that fast matrix multiplication algorithms can be used to count and find certain types of subhypergraphs faster than the naïve method. We are currently unable to improve upon the naïve algorithm for *all* possible r-uniform hypergraphs. We therefore state the following intriguing open problem.

Problem 6.1 *Let r and k be fixed positive integers, $3 \leq r < k$. Is there an $o(n^k)$ algorithm that, given an r-uniform hypergraph G, computes $c^*(G, H)$ for each $H \in \mathcal{H}_{k,r}$? More specifically, is there an $o(n^{r+1})$ algorithm that, given an r-uniform hypergraph G, determines if G has a simplex?*

References

[1] N. Alon and J. Spencer, *The Probabilistic Method*, Wiley, New York, 1992.

[2] B. Bollobás, *Random Graphs*, Academic Press, London ; Orlando, 1985.

[3] D. Coppersmith and S. Winograd, *Matrix multiplication via arithmetic progressions*, J. Symbolic Comput. 9 (1990), no. 3, 251–280.

[4] R. G. Downey and M. R. Fellows, *Fixed-parameter tractability and completeness II. On completeness for W[1]*, Theoret. Comput. Sci. 141 (1995), no. 1-2, 109-131.

[5] F. Eisenbrand and F. Grandoni, *On the complexity of fixed parameter clique and dominating set*, Theoret. Comput. Sci. 326 (2004), no. 1-3, 57–67.

[6] A. W. Goodman, *On sets of acquaintances and strangers at any party*, Am. Math. Monthly 66 (1959), 778-783.

[7] J. Håstad, *Clique is hard to approximate within $n^{1-\epsilon}$*, Acta Math. 182 (1998), no. 1, 105-142.

[8] M. Jerrum, *Large cliques elude the metropolis process*, Random Structures and Algorithms 3 (1992), 347–359.

[9] J. Nešetřil and S. Poljak, *On the complexity of the subgraph problem*, Comment. Math. Univ. Carolin. 26 (1985), no. 2, 415–419.