Learning a Hidden Matching

Combinatorial Identification of Hidden Matchings with Applications to Whole Genome Sequencing

Noga Alon[*] Richard Beigel[†] Simon Kasif [‡] Steven Rudich [§] Benny Sudakov [¶]

May 8, 2002

Abstract

We consider the problem of learning a matching (i.e., a graph in which all vertices have degree 0 or 1) in a model where the only allowed operation is to query whether a set of vertices induces an edge. This is motivated by a problem that arises in molecular biology. In the deterministic nonadaptive setting, we prove a $(\frac{1}{2}+o(1))\binom{n}{2}$ upper bound and a nearly matching $0.32\binom{n}{2}$ lower bound for the minimum possible number of queries. In contrast, if we allow randomness then we obtain (by a randomized, nonadaptive algorithm) a much lower $O(n \log n)$ upper bound, which is best possible (even for randomized fully adaptive algorithms).

1 Introduction

This paper is motivated by an important and timely problem in computational biology that arises in whole-genome shotgun sequencing. Shotgun sequencing is a high throughput technique that has resulted in the sequencing of a large number of bacterial genomes (over 70 at the time of this writing), as well as Drosophila (fruit fly) and Mouse and the celebrated Human genome (at Celera). In all such projects, we are left with a collection of contigs (long DNA sequences) that for various biological or computational reasons cannot be assembled with even the best sequence assembly algorithms. The contigs must be ordered and oriented and the *gaps* between them must be sequenced using slower, more tedious methods. When the number of gaps is small (e.g., less than ten) biologists often use

[*] Institute for Advanced Study, Princeton, NJ 08540, USA and Department of Mathematics, Tel Aviv University, Tel Aviv 69978, Israel. Email: nogaa@post.tau.ac.il. Research supported in part by a State of New Jersey grant, by a USA Israeli BSF grant, by a grant from the Israel Science Foundation and by the Hermann Minkowski Minerva Center for Geometry at Tel Aviv University.

[†] School of Mathematics, Institute for Advanced Study, Princeton NJ 08540, USA and Temple University. Research supported in part by the National Science Foundation under grants CCR-0049019 and CCR-9877150. Email: beigel@cis.temple.edu. Home page http://www.cis.temple.edu/∼beigel

[‡] Boston University

[§] Carnegie–Mellon University

[¶] Department of Mathematics, Princeton University, Princeton, NJ 08540, USA and Institute for Advanced Study, Princeton, NJ 08540, USA. Email: bsudakov@math.princeton.edu. Research supported in part by NSF grants DMS-0106589, CCR-9987845 and by the State of New Jersey.

combinatorial PCR. This technique initiates a set of “bi-directional molecular walks” along the gaps in the sequence; these walks are facilitated by PCR.

In order to initiate the molecular walks biologists use primers.[1] Primers are designed so that they bind to unique (with respect to the entire DNA sequence) templates occurring at the end of each contig. A primer (at the right temperature and concentration) anneals to the designated unique DNA substring and promotes copying of the template starting from the primer binding site, initiating a one-directional walk along the gap in the DNA sequence. A PCR reaction occurs, and can be observed as a DNA ladder, when two primers that bind to positions on two ends of the same gap are placed in the same test tube.

If we are left with N contigs, the combinatorial (exhaustive) PCR technique tests all possible pairs (quadratically many) of $2N$ primers by placing two primers per tube with the original uncut DNA strand. PCR products can be detected using gels or they can be read using sequencing technology or DNA mass-spectometry. When the number of gaps is large, the quadratic number of PCR experiments is prohibitive, so primers are pooled using $K > 2$ primers per tube; this technique is called multiplex PCR.[2] Our paper provides optimal strategies for pooling the primers to minimize the number of biological experiments needed in the gap-closing process.

Our gap-closing problem can be stated more generally as follows. We are given a set of chemicals, a guarantee that each chemical reacts with at most one of the others (because only primers on opposite sides of the same gap create a reaction), and an experimental mechanism to determine whether a reaction occurs when several chemicals are combined in a test tube. We wish to determine which pairs of chemicals react with each other with a minimum number of experiments.

Our problem can be modeled as the problem of identifying or learning a hidden matching given a vertex set and an allowed query operation ([6, 2], see [4, 5] for an alternative formulation). A vertex will represent a chemical, an edge of the matching will represent a reaction, and a query will represent checking for a reaction when a set of chemicals are combined in a test tube. Let $V = \{1, 2, \ldots, n\}$. We wish to identify an unknown (not necessarily perfect) matching M on V by asking a small number of queries of the form

$$Q_F : \qquad \text{does } F \text{ contain at least one edge of } M \text{ ?} \tag{1}$$

where F is a subset of V. This problem is of interest even in the deterministic, fully nonadaptive case. We say that a family $\mathcal{F}$ of subsets of V *solves the matching problem* on V if for any two distinct matchings M_1 and M_2 on V there is at least one $F \in \mathcal{F}$ that contains an edge of one of the matchings and does not contain any edge of the other. Obviously, any such family enables us to learn an unknown matching deterministically and non-adaptively, by asking the questions Q_F defined in (1) for each $F \in \mathcal{F}$.

[1] Primers are short preconstructed single-stranded polynucleotide chains to which new deoxyribonucleotides can be “appended” by DNA polymerase.

[2] The earliest reference to multiplex PCR is [3]. Since then hundreds of papers report using the multiplex PCR technique to answer a diverse set of questions in molecular biology. Multiplex PCR using a simple, nonoptimal pooling strategy has recently been applied successfully at The Institute for Genomic Research (TIGR) to close gaps in a number of genomes including Streptococcus [6].

Our objective is to estimate the minimum possible cardinality of a family that solves the matching problem on a set of n vertices. Toward this end, we will generalize the matching problem to finding matchings contained in a graph H (not necessarily complete), and produce an H for which we can solve the matching problem with a family of size roughly half the number of edges of H. By applying a partitioning theorem of Wilson, we can then solve the matching problem on n vertices with a family of size $(\frac{1}{2} + o(1))\binom{n}{2}$.

We show that our construction is tight up to a constant factor, as stated in the following theorem.

Theorem 1.1 *For every $n > 2$, every family $\mathcal{F}$ that solves the matching problem on n vertices satisfies*

$$|\mathcal{F}| \geq \frac{49}{153}\binom{n}{2}.$$

The proof of the lower bound is presented in Sections 3 and 4.

Next we consider randomized nonadaptive algorithms. In contrast to the 1-round deterministic case we produce, somewhat surprisingly, an $O(n \log n)$ solution in this model. This solution is asymptotically optimal up to a constant factor, because of the information-theoretic $\Omega(n \log n)$ lower bound, even if we do not restrict the number of rounds. We believe that the sharp difference between deterministic and randomized nonadaptive algorithms here is remarkable; while one can hardly beat the trivial $\binom{n}{2}$ bound in the deterministic case, the randomized fully nonadaptive algorithm is already as efficient (up to a constant factor) as the best possible fully adaptive algorithm for the problem. Moreover, the same technique shows that a hidden copy of any sparse graph, that is, a graph with a linear number of edges in which all degrees are $o(\sqrt{n})$, can be found, with high probability, in a one-round randomized algorithm making only $O(n \log n)$ queries.

Finally we present deterministic k-round algorithms that make $O(kn^{1+1/(2(k-1))}\text{polylog}n)$ queries. Our deterministic 2-round algorithm asks $\frac{5}{4}n^{3/2}(1 + o(1))$ queries of size at most $n^{1/4}$ each. This is optimal up to a factor of 5/4 among *all* algorithms that make queries of size at most $n^{1/4}$, which may be useful in view of practical limitations on multiplexing. For $k \geq 3$ our algorithms are based on a coloring lemma for projective planes that may be interesting in its own right.

Our techniques combine combinatorial and probabilistic tools with results about graph decomposition and about the existence of certain designs. Throughout the paper, we omit all floor and ceiling signs, whenever these are not crucial. All logarithms are in base 2, unless otherwise specified.

2 Related Work

In an earlier paper with Fortnow and Apaydin [2] we obtained a randomized, adaptive algorithm that solves the matching problem in 8 rounds with an expected number of approximately $0.72n \log_2 n$ queries. Our results here improve the number of rounds to 1 in the randomized case (at the cost of doubling the number of queries. If we allow 2 rounds we can, in fact, keep the total number of queries to be roughly $0.72n \log_2 n$). We further show here that in the one round, deterministic case, far more queries are needed, though some saving over the trivial algorithm is possible.

Grebinski and Kucherov [4, 5] consider the problem of finding a Hamilton cycle. They obtain an $O(n \log n)$ *adaptive* algorithm. They also have an $O(n)$ purely nonadaptive solution using more powerful queries (i.e., queries that report the *number* of edges induced by a set of vertices). Using our methods here we can show that $\Omega(n^2)$ queries are needed for finding a Hamilton cycle in the deterministic nonadaptive case in our model. A similar $\Omega(n^2)$ lower bound can be proved for the problem of determining the *number* of edges of a hidden matching, as well as for the problem of finding a hidden copy of any given bounded degree graph with $\Omega(n)$ edges.

3 Sparse families

A family of sets $\mathcal{A} = \{A_1, \ldots, A_k\}$ is *sparse* if there is a collection of pairwise disjoint pairs of members of $V = \bigcup_{i=1}^{k} A_i$ such that each A_i contains at least one of the pairs. Therefore, $\mathcal{A}$ is sparse iff there is a matching on V such that the answer to each question Q_A for $A \in \mathcal{A}$ is "yes." It is easy to see that any set of more than $(p^2 + p + 1)/2$ lines in a projective plane of order p (in which each line is of size $p + 1$) is not sparse, and our results here will imply that every family consisting of at most $0.32\binom{m+2}{2}$ sets, each of size at least m, is sparse.

For a family $\mathcal{F}$ of subsets define the *t-weight* of the family, denoted $w_t(\mathcal{F})$, as follows:

$$w_t(\mathcal{F}) = \sum_{F \in \mathcal{F}} \frac{1}{\binom{|F|+t}{2}}.$$

The 2-weight is simply called the *weight* and is denoted, for short, by $w(\mathcal{F})$. The main lemma of this section is the following.

Lemma 3.1 *Every family $\mathcal{F}$ of sets whose weight is at most* $49/153$*, is sparse.*

Proof: If $\mathcal{F}$ contains a set of size 1 then $w(\mathcal{F}) \geq 1/3 > 49/153$. Thus we may and will assume that all sets in $\mathcal{F}$ are of size at least 2. Let $M \in \mathcal{F}$ be a set of minimum cardinality, $|M| = m$.

For a pair of distinct elements p, q, define

$$\mathcal{F}(p,q) = \{F - \{p,q\} : F \in \mathcal{F}, \{p,q\} \not\subseteq F\}.$$

Note that if we pick the pair $\{p, q\}$ as a member of the matching we are trying to construct to show that $\mathcal{F}$ is sparse, then the members of $\mathcal{F}(p, q)$ are precisely those that will have to be handled by the rest of the matching. This suggests to prove the following claim:

Claim: There exists a pair of distinct elements p, q of F such that $w(\mathcal{F}(p, q)) \leq w(\mathcal{F})$.

To prove the claim we choose p, q randomly and uniformly among all pairs of members of M and show that the expected value $E(w(\mathcal{F}(p, q)))$ is at most $w(\mathcal{F})$.

Henceforth F will denote an element of $\mathcal{F} - \{M\}$. Let $\kappa(F)$ denote $|F \cap M|$. We have

$$E(w(\mathcal{F}(p,q))) \quad = \quad w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \sum_{F \neq M} \left(\frac{\kappa(F)(m-\kappa(F))}{\binom{m}{2}} \left(\frac{1}{\binom{|F|+1}{2}} - \frac{1}{\binom{|F|+2}{2}} \right) - \frac{\binom{\kappa(F)}{2}}{\binom{m}{2}} \frac{1}{\binom{|F|+2}{2}} \right)$$

$$\begin{aligned}
&= w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \sum_{k<m}\sum_{\kappa(F)=k}\left(\frac{k(m-k)}{\binom{m}{2}}\left(\frac{1}{\binom{|F|+1}{2}} - \frac{1}{\binom{|F|+2}{2}}\right) - \frac{\binom{k}{2}}{\binom{m}{2}}\frac{1}{\binom{|F|+2}{2}}\right) \\
&= w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m}{2}}\sum_{k<m}\sum_{\kappa(F)=k}\left(k(m-k)\left(\frac{\binom{|F|+2}{2}}{\binom{|F|+1}{2}} - 1\right) - \binom{k}{2}\right)\frac{1}{\binom{|F|+2}{2}} \\
&= w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m}{2}}\sum_{k<m}\sum_{\kappa(F)=k}\left(\frac{2k(m-k)}{|F|} - \binom{k}{2}\right)\frac{1}{\binom{|F|+2}{2}} \\
&\le w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m}{2}}\sum_{k<m}\sum_{\kappa(F)=k}\left(\frac{2k(m-k)}{m} - \binom{k}{2}\right)\frac{1}{\binom{|F|+2}{2}} \\
&= w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m}{2}}\sum_{k<m}\left(\frac{2k(m-k)}{m} - \binom{k}{2}\right)\sum_{\kappa(F)=k}\frac{1}{\binom{|F|+2}{2}} \\
&= w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m}{2}}\sum_{k<m}\mu(m,k)\sum_{\kappa(F)=k}\frac{1}{\binom{|F|+2}{2}},
\end{aligned}$$

where we define $\mu(m,k) = \frac{2k(m-k)}{m} - \binom{k}{2}$. For all m, we have $\mu(m,0) = 0$, $\mu(m,1) = 2 - 2/m$, $\mu(m,2) = 3 - 8/m$, and $k \ge 2 \Rightarrow \mu(m,k) \le \mu(m,2)$. Thus $\mu(m,k)$ is maximized at $k = 1$ or $k = 2$. Define $\mu(m) = \max_{k<m}\mu(m,k)$. Now we have

$$\begin{aligned}
E(w(\mathcal{F}(p,q))) &\le w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m}{2}}\sum_{k<m}\mu(m,k)\sum_{\kappa(F)=k}\frac{1}{\binom{|F|+2}{2}} \\
&\le w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m}{2}}\sum_{k<m}\mu(m)\sum_{\kappa(F)=k}\frac{1}{\binom{|F|+2}{2}} \\
&= w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m}{2}}\mu(m)\sum_{k<m}\sum_{\kappa(F)=k}\frac{1}{\binom{|F|+2}{2}} \\
&= w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m}{2}}\mu(m)\sum_{F\ne M}\frac{1}{\binom{|F|+2}{2}} \\
&= w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m}{2}}\mu(m)\left(w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}}\right) \\
&= w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}} + \frac{1}{\binom{m+2}{2}}\frac{(m+2)(m+1)}{m(m-1)}\mu(m)\left(w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}}\right).
\end{aligned}$$

Thus it suffices to prove that

$$\frac{(m+2)(m+1)}{m(m-1)}\mu(m)\left(w(\mathcal{F}) - \frac{1}{\binom{m+2}{2}}\right) \le 1$$

or equivalently that

$$w(\mathcal{F}) \le \frac{m(m-1)}{(m+2)(m+1)\mu(m)} + \frac{1}{\binom{m+2}{2}}.$$

As noted above, $\mu(m)$ is either $\mu(m,1)$ or $\mu(m,2)$. In the first case, we have

$$\frac{m(m-1)}{(m+2)(m+1)\mu(m)} + \frac{1}{\binom{m+2}{2}} = \frac{m^2+4}{2(m+2)(m+1)} \geq \frac{13}{40}$$

for $m > 1$ (with equality at $m = 3$). In the second case, we have

$$\frac{m(m-1)}{(m+2)(m+1)\mu(m)} + \frac{1}{\binom{m+2}{2}} = \frac{1}{3}\frac{m^2(m-1)+6m-16}{(m+2)(m+1)(m-8/3)} \geq \frac{49}{153}$$

for $m > 2$ (with equality at $m = 16$). By assumption, $w(\mathcal{F}) \leq 49/153 < 13/40$, completing the proof of the claim.

By repeatedly applying the claim we get smaller and smaller families of sets whose weights remain bounded by 49/153. This process must terminate with a matching that captures all members of $\mathcal{F}$, showing that $\mathcal{F}$ is sparse and completing the proof of the lemma. □

4 The proof of the main result for the fully nonadaptive case

In this short section we present the proof of Theorem 1.1. We need the following simple fact.

Lemma 4.1 *Let $\mathcal{F}$ be a family of subsets of $V = \{1, 2, \ldots, n\}$ that solves the matching problem on V. Then, for every two distinct $a, b \in V$, the family $\mathcal{F}_{a,b} = \{F - \{a,b\} : F \in \mathcal{F} \text{ and } \{a,b\} \subseteq F\}$ is not sparse.*

Proof: Assume this is false, and $\mathcal{F}_{a,b}$ is sparse for some $a, b \in V$. Then, there is a matching M in $V - \{a,b\}$ so that each member of $\mathcal{F}_{a,b}$ contains an edge of M. But then the answers to each question Q_F with $F \in \mathcal{F}$ for the two matchings M and $M \cup \{a,b\}$ are identical, contradicting the fact that $\mathcal{F}$ solves the matching problem. □

Proof of Theorem 1.1: Let $\mathcal{F}$ be a family of subsets of $V = \{1, 2, \ldots, n\}$ that solves the matching problem on V. Let a, b be any pair of distinct vertices in V. We know by Lemma 4.1 that the family $\mathcal{F}_{a,b} = \{F - \{a,b\} : F \in \mathcal{F} \text{ and } \{a,b\} \subseteq F\}$ is not sparse. Therefore, by Lemma 3.1,

$$\sum_{F \in \mathcal{F}, \{a,b\} \subseteq F} \frac{1}{\binom{|F|}{2}} = \sum_{F' \in \mathcal{F}_{a,b}} \frac{1}{\binom{|F'|+2}{2}} > \frac{49}{153}.$$

We can now assign, for each $F \in \mathcal{F}$ a weight of $\frac{1}{\binom{|F|}{2}}$ to each pair of distinct elements $a, b \in F$. The total weight distributed in this way is precisely $|\mathcal{F}|$, as the total contribution of each $F \in \mathcal{F}$ is 1. On the other hand, the total weight assigned to each pair $a, b \in V$ is at least $\frac{49}{153}$, implying that $|\mathcal{F}| \geq \frac{49}{153}\binom{n}{2}$, as needed. □

Note that the same proof supplies a $\frac{49}{153}\binom{n}{2}$ lower bound for the number of queries in any one-round deterministic algorithm that determines the **number** of edges of a hidden matching on n vertices.

5 Other hidden graphs

In this section we show how to extend the methods described in the previous two sections and obtain a lower bound for the number of queries needed to find a hidden copy of a member of a given family of graphs with certain properties. Throughout the section, we make no attempt to optimize the absolute constants in our various estimates.

Let $\mathcal{H}$ be a family of labelled graphs on the set $V = \{1, 2, \ldots, n\}$, and suppose $\mathcal{H}$ is closed under isomorphism. Thus, for example, $\mathcal{H}$ may be the set of all Hamilton cycles on V, or all matchings on V, or all perfect matchings on V. Our objective is to learn a hidden copy of some member of $\mathcal{H}$ by asking a small number of queries Q_F as given in (1). We say that a family $\mathcal{F}$ solves the $\mathcal{H}$-problem if for any two distinct members H_1 and H_2 in $\mathcal{H}$ there is at least one $F \in \mathcal{F}$ that contains an edge of one of the graphs H_i and does not contain any edge of the other. Obviously, any such family enables us to learn an unknown member of $\mathcal{H}$ deterministically and non-adaptively, by asking the questions Q_F defined in (1) for each $F \in \mathcal{F}$.

Theorem 5.1 *There exists an absolute constant $c > 0$ such that the following holds.*
Let $\mathcal{H}$ be a family of graphs on V, closed under isomorphism, and suppose that there are two distinct graphs $H_1, H_2 \in \mathcal{H}$ and a set of vertices $D \subset V$, $|D| = d$ satisfying the following:
(i) The graphs obtained from H_1 and from H_2 by omitting all edges connecting two vertices of D are identical, and
(ii) There is a matching of at least pn edges in H_1 which contains no vertices of D (clearly this matching is also a matching in H_2).
Then, if $1/p > d$, every family $\mathcal{F}$ that solves the $\mathcal{H}$-problem satisfies $|\mathcal{F}| \geq c\frac{p^2}{d^2}\binom{n}{2}$.

Note that this result provides an $\Omega(n^2)$ lower bound for the problem of learning a **perfect** matching or a Hamilton cycle, and, more generally, the problem of learning a hidden copy of any fixed, bounded-degree graph with $\Omega(n)$ edges. It also provides an $\Omega(n^2)$ lower bound for the problem of finding a hidden copy of a vertex disjoint union of a clique of size $n-3$ and a single edge, but **not** for the problem of finding a hidden copy of a vertex disjoint union of a clique of size $n-2$ and a single edge (and indeed it is easy to see that $O(n)$ queries suffice for the latter problem).

A family $\mathcal{F}$ of subsets of V is *p-sparse* if there is a collection of at most pn pairwise disjoint pairs of members of V such that each $F \in \mathcal{F}$ contains at least one of the pairs. Therefore, $\mathcal{F}$ is p-sparse iff there is a matching on V consisting of at most pn edges such that the answer to each query Q_F for $F \in \mathcal{F}$ is "yes."

Lemma 5.2 *There is an absolute constant $c_1 > 0$ such that every family $\mathcal{F}$ of subsets of V of weight at most $c_1 p^2$ is p-sparse.*

Proof: Let V_1 be a random subset of V obtained by picking each $v \in V$, randomly and independently, to lie in V_1 with probability p. As the expected size of V_1 is pn, it follows that with probability at least a half, its size is at most $2pn$. For each set $F \in \mathcal{F}$, the expected size of $F \cap V_1$ is clearly $p|F|$, and as this size is a binomial random variable it follows, by the standard estimates for Binomial distributions (see, e.g., [1], Appendix A, Theorem A.1.13), that the probability that it is not at least

$p|F|/2$ does not exceed $e^{-p|F|/8}$. Since the weight of $\mathcal{F}$ is at most c_1p^2 for some (small) c_1, we conclude that each set in $\mathcal{F}$ is of size at least, say, $100/p$. As $e^{-px/8} < \frac{1}{p^2\binom{x+2}{2}}$ for all $x > 100/p$, it follows that the probability that there is some set $F \in \mathcal{F}$ such that $|F \cap V_1| < p|F|/2$ is smaller than $\frac{w(\mathcal{F})}{p^2} \leq c_1 < 1/2$. Therefore, there exists a set $V_1 \subset V$ of cardinality at most $2pn$ so that $|F \cap V_1| \geq p|F|/2$ for all $F \in \mathcal{F}$. As $|F| > 100/p$ for each $F \in \mathcal{F}$, this implies that the weight of the family $\mathcal{F}_1 = \{F \cap V_1 : F \in \mathcal{F}\}$ is at most, say, $\frac{8}{p^2}w(\mathcal{F}) \leq 8c_1 < 49/153$ (assuming c_1 is sufficiently small). By Lemma 3.1, $\mathcal{F}_1$ is sparse, and as $|V_1| \leq 2pn$, it follows that $\mathcal{F}$ is p-sparse. □

Lemma 5.3 *Let $\mathcal{H}$, V, n, d, and p be as in Theorem 5.1. Let $\mathcal{F}$ be a family of subsets of $V = \{1, 2, \ldots, n\}$ that solves the $\mathcal{H}$-problem. Then, for every subset $D \subset V$, $|D| = d$, the family $\mathcal{F}_D = \{F - D : F \in \mathcal{F}, |F \cap D| \geq 2\}$ is not p-sparse.*

Proof: Assume this is false, and suppose $\mathcal{F}_D$ is p-sparse for some $D' \subset V$, $|D'| = d$. Then, there is a matching M of size at most pn in $V - D'$ so that each member of $\mathcal{F}_{D'}$ contains an edge of M. The matching M can be completed to a graph with no edges in D' which is isomorphic to the graph obtained from H_1 (or H_2) by omitting the edges inside D, where the isomorphism maps D onto D'. It is now possible to extend this graph to a copy of H_1, or to a copy of H_2, by only adding edges inside D'. But then the answers to each question Q_F with $F \in \mathcal{F}$ for these two distinct members of $\mathcal{H}$ are identical, contradicting the fact that $\mathcal{F}$ solves the $\mathcal{H}$-problem. □

Proof of Theorem 5.1: Let $\mathcal{F}$ be a family of subsets of $V = \{1, 2, \ldots, n\}$ that solves the $\mathcal{H}$-problem. By Lemma 5.2 and Lemma 5.3, for every set D consisting of d vertices of V, the weight of the family $\mathcal{F}_D = \{F - D : F \in \mathcal{F}, |F \cap D| \geq 2\}$ is at least c_1p^2. We claim that

$$\sum_{F \in \mathcal{F}, |F \cap D| \geq 2} \frac{1}{\binom{|F|}{2}} \geq c_2p^2,$$

for every D as above (and for an appropriately chosen $c_2 > 0$). Indeed, if there is a set $F \in \mathcal{F}$ of size at most, say, $10/p$, that intersects D in at least 2 elements, this follows immediately. Otherwise, by the assumption that $1/p > d$,

$$\frac{1}{\binom{|F|-d+2}{2}} \leq 2\frac{1}{\binom{|F|}{2}},$$

and the claim follows from the fact that the weight of $\mathcal{F}_D$ is at least c_1p^2.

If D is a random subset of d vertices of V, then for every $F \in \mathcal{F}$, the probability that $|D \cap F| \geq 2$ is at most $\binom{d}{2}\binom{|F|}{2}/\binom{n}{2}$. It follows that the expected value of the random variable

$$\sum_{F \in \mathcal{F}, |F \cap D| \geq 2} \frac{1}{\binom{|F|}{2}}$$

is at most $\binom{d}{2}|\mathcal{F}|/\binom{n}{2}$, and as this random variable is always at least c_2p^2, the desired result follows. □

6 An upper bound for the fully nonadaptive case

In this section we show how to design families of size $(\frac{1}{2}+o(1))\binom{n}{2}$ to solve the matching problem on n vertices. It will be helpful if we first generalize the matching problem. We say that a family $\mathcal{F}$ of cliques contained in G *solves the matching problem* on G if for any two distinct matchings contained in G, there is at least one clique in $\mathcal{F}$ that contains an edge of one of the matchings and does not contain any edge of the other. (Note, for example, that if G is triangle free then $\mathcal{F}$ must be a set of edges.) The matching problem on n vertices is thus the same as the matching problem on K_n. Let $f(G)$ denote the size of the smallest family that solves the matching problem on G.

Throughout this section, let $E(H)$ denote the edge set of a graph H, and let $\gcd(H)$ denote the greatest common divisor of the degrees of all vertices in H.

Theorem 6.1 (Wilson [7]) *For every graph H there exists a constant N such that for all $n \geq N$, K_n is the union of $\binom{n}{2}/|E(H)|$ pairwise edge-disjoint graphs isomorphic to H if and only if $\binom{n}{2}$ is divisible by $|E(H)|$ and $n-1$ is divisible by* $\gcd(H)$.

Corollary 6.2 *For every fixed graph H,*

$$f(K_n) \leq \frac{f(H)}{|E(H)|}\binom{n}{2} + O(n).$$

Furthermore, for fixed H, the solution to the matching problem for K_n is constructive.

Proof: By Wilson's theorem, K_n is the union of $\binom{n}{2}/|E(H)|$ graphs isomorphic to H. Solve the matching problem in each of those graphs with a family of size $f(H)$. □

We say that a family $\mathcal{F}$ determines the status of an edge e if for every pair of matchings M_1, M_2 such that $e \in M_1$ and $e \notin M_2$ there is at least one clique $\mathcal{F}$ that contains an edge of one of the matchings and does not contain any edge of the other.

The reader may easily verify the following lemma:

Lemma 6.3 (Two Thirds) *Let a, b, c, x be four distinct vertices. Assume that $\mathcal{F}$ determines the status of $\{a, b\}$ and $\{b, c\}$. If $\mathcal{F}$ contains the two triangles $\{a, b, x\}$ and $\{b, c, x\}$, then $\mathcal{F}$ also determines the status of $\{a, x\}$, $\{b, x\}$, and $\{c, x\}$.*

We note in passing that one may easily apply the Two Thirds lemma to obtain a solution of size $\frac{2}{3}\binom{n}{2} + O(n)$.

Definition 6.4 (HEX_s^+) *Let $s \geq 1$. Tile a hexagon having side length s with unit equilateral triangles. Add one more vertex Z and edges from Z to every vertex in the tiling. Call the resulting graph* HEX_s^+.

The tiling above contains $v = 3s^2 + 3s + 1$ vertices, $e = 9s^2 + 3s$ edges, and $f = 6s^2$ triangles. Therefore the graph HEX_s^+ contains $v + e = 12s^2 + 6s + 1$ edges. We solve the matching problem on HEX_s^+ with the following tests:

Tetrahedra $T \cup \{Z\}$, for every triangle T in the tiling,

Boundary every boundary edge of the tiling, and every edge from Z to a point on the boundary.

As a warmup, let us see why these tests suffice, assuming that Z is unmatched. In this case, the tetrahedra queries are equivalent to triangles, and we just apply the two-thirds lemma repeatedly, starting at the boundary of the tiling.

Now let us see why these tests suffice in general. Try the following cases in order.

Case 1: For some Y on the boundary of the tiling, $ZY \in M$. This edge is tested, so we know $ZY \in M$. Finish as in the warmup.

Case 2: For some Y in the interior of the tiling, $ZY \in M$. In this case all six tests containing Y say yes. Call that the 6-triangle property for Y. If only one point has the 6-triangle property then we know that that point is matched to Z. If three distinct points have the 6-triangle property, it is easy to check that they must be adjacent in a straight line, and the middle one must be matched to Z. Consequently, no more than three points can have the 6-triangle property. In either of those subcases we finish as in the warmup.

If exactly two points have the 6-triangle property, then they must be adjacent; we proceed as in the warmup until we reach the 10 triangles containing those two points. With the aid of the tests already performed, a simple case analysis tells us which point is matched to Z.

Case 3: For all Y in the tiling, $ZY \notin M$. Then no point has the 6-triangle property, so we know that Z is unmatched. Proceed as in the warmup.

The total number of tests is $f + 12s = 6s^2 + 12s$. The ratio of tests to edges is

$$(6s^2 + 12s)/(12s^2 + 6s + 1) = \frac{1}{2} + \frac{18s - 1}{24s^2 + 12s + 2}.$$

Combining this with the Corollary of Wilson's Theorem, we see that $f(K_n) = (1/2 + O(1/s))\binom{n}{2}$. Letting s be a slowly growing function of n, we obtain

Corollary 6.5

$$f(K_n) \le \left(\frac{1}{2} + o(1)\right)\binom{n}{2}.$$

7 Probabilistic nonadaptive algorithms

In this section we present a very efficient randomized algorithm for the matching problem. The simplest version of the algorithm queries $bn \log n$ random subsets of size $c\sqrt{n}$ each. The analysis below shows that for an appropriate choice of b and c the algorithm solves, with high probability, the matching problem in one round. Since we believe that this algorithm and some of its variants may be of practical interest, we do make here some efforts to optimize the absolute constant obtained in the estimate for the total number of queries. It turns out that in order to improve the constant, it is better to ask the queries according to randomly shifted (modified) projective planes. The details follow.

Procedure RPP: Testing according to a random projective plane We assume now that $n = p^2 + p + 1$ for some prime p. Testing according to a random projective plane consists of the following: Randomly permute our n vertices and identify them with the points of the projective plane P of order p. Perform one test for each line.

Now consider a pair (x, y). Exactly one line in P contains x and y. The probability that that line contains no (other) edge of M is at least the value of this probability when the matching is perfect and xy is not a matching edge. It is not difficult to see that in this case, the probability is precisely:

$$\frac{(n-4)(n-6)\ldots(n-2p)}{(n-2)(n-3)\ldots(n-p)}.$$

Indeed, the number of ways to choose an ordered set of the other $p-1$ points of the line (besides x and y) without containing any matching edge is the numerator, as after i points (including x and y) have already been chosen, there are $n-2i$ possibilities for choosing the next point, which has to be different from the chosen points and their mates. The denominator is the total number of possibilities for choosing an ordered set of $p-1$ points. The last expression is at least

$$e^{-(1-o(1))\frac{p^2}{2n}} = e^{-1/2}(1-o(1)).$$

Testing according to $d \log n$ random projective planes Perform Procedure RPP $d \log n$ times independently in parallel, for some real number d. The probability that every line containing x and y contains an edge of M (other than possibly (x, y)) is at most $\pi(d) = \left(\left(1 - e^{-1/2}\right)^d (1 + o(1))\right)^{\log n}$. If we choose $d = (1 + o(1)) \ln 2 / \ln(1/(1 - e^{-1/2})) \approx 0.74$, then $\pi(d) \leq 1/n$.

Consequently, those tests suffice to identify all but $n/2$ non-matching edges on average. Those remaining nonedges and all matching edges can be identified in a second round with only n tests. Thus we have a 2-round algorithm for the matching problem that makes an expected number of approximately $0.74n \log n$ tests and makes no errors.

If we choose d twice as large, i.e., $d = 2 \ln 2 / \ln(1/(1 - e^{-1/2})) + \epsilon$ for some arbitarily small ϵ (say $d \approx 1.49$), then $\pi(d) = o(1/n^2)$. Consequently, those tests suffice to identify all non-matching edges with high probability. Once we have identified all nonedges, the same reasoning shows that all the matching edges are identified with high probability as well. Thus we have a 1-round algorithm for the matching problem that makes an expected number of approximately $1.49n \log n$ tests and makes no errors.

Point doubling These constants can be improved. When every vertex is in the same number of tests, the ideal test size is approximately $\sqrt{(2 \ln 2)n}$, but there are no designs like the projective plane with sets of size greater than $\sqrt{n}$. Fortunately, we do not need every pair of points to belong to exactly one set. It suffices to construct designs in which every pair of points belongs to at least one set, provided that we do not generate too many sets in the process. This can be accomplished rather easily by randomly "doubling" some of the points in the projective plane. We double a point x by adding a new point x' to each line that contains x.

To be precise, we assume now that $n = \lceil (2\ln 2)m \rceil$, where $m = p^2 + p + 1$ and p is prime. We start with the projective plane of order p and double $\lceil (2\ln 2 - 1)m \rceil$ randomly chosen points. This results in n points. We still have $m \approx n/(2\ln 2)$ lines. By the law of large numbers each of those lines has approximately $2\ln 2\sqrt{m} \approx \sqrt{(2\ln 2)n}$ points with high probability.

Now, let us look at a single design and a single pair of points x, y. Consider a "line" containing x and y. (If x and y are not the duplicates of a single point, there is one such line, else, there are $p+1$ such lines, and then it suffices to consider one of them.) Let $t \approx \sqrt{(2\ln 2)n}$ be the number of points on this line. The probability that it contains no (other) edge of M, besides, possibly, xy, is, by the same reasoning described above, at least

$$\frac{(n-4)(n-6)\ldots(n-2t+2)}{(n-2)(n-3)\ldots(n-t+1)} = e^{-(1+o(1))\frac{t^2}{2n}} = (1/2)(1+o(1)).$$

Take, now $d\log n$ random projective planes with doubled points of the type above. The probability that every "line" containing x and y contains an edge of M (other than possibly (x,y)) is at most $\pi'(d) = \Big((1/2)^d\Big(1+o(1)\big)\big)^{\log n}$. Thus $\pi'(1+o(1)) = 1/n$ and $\pi'(2+o(1)) = 1/n^2$. Since each design contains approximately $n/(2\ln 2)$ "lines," we obtain the following.

Theorem 7.1 *The matching problem on n vertices can be solved by probabilistic algorithms with the following parameters:*

- *2 rounds and $(1/(2\ln 2))n\log n(1+o(1)) \approx 0.72n\log n$ tests*
- *1 round, and $(1/\ln 2)n\log n(1+o(1)) \approx 1.44n\log n$ tests.*

Note that the algorithms make no errors in the sense that when we get the answers we know which edges are matching edges and which are not. With high probability, we get all the information in the 1-round algorithm; in the rare event we do not, we know it, and can make an additional set of queries for all the edges whose status has not been determined. In the 2-round algorithm we always get all the information, but with positive probability we will have to ask more than n queries in the second round.

Note also that the algorithms described here can be easily modified to find a hidden copy of any graph with $O(n)$ edges and with maximum degree $O(\sqrt{n})$ in one randomized round, using $O(n\log n)$ queries.

8 Deterministic k-round algorithms

In this section we present *deterministic* k-round algorithms that make at most $O(n^{1+1/(2(k-1))}\text{polylog}n)$ queries per round. In the special case $k=2$, we do not need the polylog$n$ factor. All of our deterministic algorithms are constructive.

A lemma about the chromatic number of the graph consisting of all edges contained in half the lines of a projective plane will allow us to reduce the general problem to a bipartite matching problem. The lemma, which may be interesting in its own right, is proved by considering the eigenvalues of the plane's incidence matrix.

Lemma 8.1 (Coloring) *Let P be a finite projective plane with n points. Obtain Q by deleting at least $n/2$ of P's lines. Let $G = (V, E)$ where V is P's point set and E consists of all pairs (x, y) such that there is a line in Q containing both x and y. Then G is $\sqrt{n}\ln n(1 + o(1))$ colorable using color classes of size less than $\sqrt{n}$. Furthermore such a coloring can be found in time polynomial in n.*

The basic idea in the proof of this lemma is as follows. Consider a set B of points. On average, a line will contain about $|B|/\sqrt{n}$ points. We will show that if B is not very small then most lines contain at least half that number of points. This will allow us to greedily choose our color classes from among the lines that were deleted from P. We need the following Lemma, whose proof is essentially identical to that of Lemma 9.2.4 of [1].

Lemma 8.2 *Let $G = (U, V; E)$ be a d-regular bipartite graph with classes of vertices U and V of size n each. Let $A = (A_{u,v} : u \in U, v \in V)$ be the (bipartite) adjacency matrix of G given by $A_{u,v} = 1$ iff $uv \in E$ and $A_{u,v} = 0$ otherwise. Suppose, further, that every eigenvalue of A^tA except the largest (which is d^2) is at most λ^2. Then, for every $B \subseteq V$,*

$$\sum_{u\in U}\left(|N(u)\cap B| - d\frac{|B|}{n}\right)^2 \leq \lambda^2|B|\left(1-\frac{|B|}{n}\right).$$

Let P be a projective plane of order p. Thus it has $n = p^2 + p + 1$ points. Let G be the incidence graph of P, i.e., the bipartite graph with classes of vertices U and V, with $|U| = |V| = n$, in which V is the set of points and U is the set of lines, where uv is an edge iff the line u contains the point v. If A is the adjacency matrix of G, then A^tA is a matrix in which all diagonal entries are $p + 1$ and all other entries are 1. Consequently, the largest eigenvalue of A^tA is $(p + 1)^2$ and all its other eigenvalues are equal to p. It follows that for every set of points $B \subset U$, we can bound the number of lines v containing less than $\frac{d|B|}{2n}$ points of B by the above lemma. Namely

$$\begin{aligned}\left|\left\{v \in V : |N(v)\cap B| < \frac{d}{2}\frac{|B|}{n}\right\}\right| &< \frac{4\lambda^2}{d^2}\frac{n^2}{|B|}\\ &= \frac{4p}{(p+1)^2}\frac{n^2}{|B|}\\ &\leq \frac{4n^{3/2}}{|B|}.\end{aligned}$$

Therefore, if $|B| > 10\sqrt{n}$ then every set consisting of $0.4n$ lines contains a line that contains at least $\frac{\sqrt{n}}{2}\frac{|B|}{n} = \frac{|B|}{2\sqrt{n}}$ elements of B. Now we are in a position to prove

Corollary 8.3 *Every set S consisting of at least $0.4n$ lines contains a subset consisting of at most $\sqrt{n}\ln n$ lines covering all but at most $10\sqrt{n}$ points.*

Proof: Initially, let $B = V$. As long as $|B| \geq 10\sqrt{n}$, we may choose a line in S that contains at least a $\frac{1}{2\sqrt{n}}$ fraction of the points in B, and then remove those points from B. After at most $2\sqrt{n}\ln[n/(10\sqrt{n})] < \sqrt{n}\ln n$ iterations, we will have reduced B to size at most $10\sqrt{n}$. □

To complete the proof of the Coloring Lemma, we take our set of lines to be the ones deleted from P. Our color classes are the $\sqrt{n}\ln n$ lines promised by the preceding corollary as well as the $10\sqrt{n}$ singletons not covered by those lines. If a point belongs to more than one of those lines, then we can choose its color class arbitrarily from among them. □

Lemma 8.4 (First Bipartite) *Assume that M is a nonempty matching on V. Let V be the disjoint union of L and R where $|R| \geq 2$, and assume we know that neither L nor R contains an edge of M. We can learn M with a k-round algorithm that makes at most $|M||L|^{1/k}\log|R|$ tests per round.*

Proof: The proof is by induction on k. If $k = 1$, then we can perform a parallel binary search for each element of L to find its match, if any, in R. The number of tests performed is $|L|\log|R|$.

Now let $k \geq 2$ and assume we have a $(k-1)$-round algorithm that makes at most $|M||L|^{1/(k-1)}\log|R|$ tests per round. Let $t = |L|^{1/k}$. Partition L into t pieces $L_1, \ldots, L_t$ of size $|L|/t$. In round one, test $L_i \cup R$ for each i. At most $|M|$ of those sets can contain an edge, say $L_{i_1} \cup R, \ldots, L_{i_m} \cup R$ where $m \leq |M|$. Apply the inductive hypothesis to the matching problems on those m sets. Let e_j denote the number of edges in L_{i_j}. The number of tests per round is at most

$$\sum_j e_j(|L|/t)^{1/(k-1)}\log|R| = |M||L|^{1/k}\log|R|$$

□

Lemma 8.5 (Second Bipartite) *Assume that M is a nonempty matching on V. Let V be the disjoint union of L and R, and assume we know that neither L nor R contains an edge of M. Let c be a real number such that $0 < c < 1$, $c|L|^{1/k} \geq 1$. Let $k \geq 2$. We can learn M with a k-round algorithm that makes at most $c|L|^{1/k}$ tests in the first round and at most $|M||L|^{1/k}\log|R|/c^{1/(k-1)}$ tests in each subsequent round.*

Proof: Let $t = c|L|^{1/k}$. Partition L into t pieces $L_1, \ldots, L_t$ of size $|L|/t$. In round one, test $L_i \cup R$ for each i. At most $|M|$ of those sets can contain an edge, say $L_{i_1} \cup R, \ldots, L_{i_m} \cup R$ where $m \leq |M|$. Apply the First Bipartite Lemma to the matching problems on those m sets. Let e_j denote the number of edges in L_{i_j}. The number of tests performed in round one is $t = c|L|^{1/k}$ and in each subsequent round it is at most

$$\sum_j e_j(|L|/t)^{1/(k-1)}\log|R| = |M||L|^{1/k}\log|R|/c^{1/(k-1)}$$

□

Theorem 8.6 *For $3 \leq k \leq \log n$, there is a deterministic k-round algorithm for the matching problem that asks $O(n^{1+1/(2(k-1))}(\log n)^{1+1/(k-1)})$ queries per round.*

Proof: After adding $o(n)$ virtual unmatched points we may assume that n is of the form p^2+p+1 where p is prime; these virtual points will be omitted from any actual tests. In round one, construct a projective plane with n points, and perform one test for each line. Delete every line that contains no edge of the matching. Construct G and its color classes as in the Coloring lemma. If $(x,y) \in M$ then x and y must belong to distinct color classes of G. For each pair of color classes apply the Bipartite Lemma with $c = \log n^{1/(k-1)}/\log n$ and the number of rounds $= k-1$. The number of tests in round two is at most

$$O(\binom{\sqrt{n}\log n}{2} c\sqrt{n}^{1/(k-1)}) = O(n^{1+1/(2(k-1))}(\log n)^{1+1/(k-1)}).$$

The number of tests performed in any of the rounds 3 through k is bounded by

$$O((n/2)\sqrt{n}^{1/(k-1)} \log \sqrt{n}/c^{1/(k-2)}) = O(n^{1+1/(2(k-1))}(\log n)^{1+1/(k-1)}).$$

□

Our 2-round deterministic algorithm uses finite projective spaces of dimension 4 in a somewhat different way. It has the advantage of using queries whose size is approximately $n^{1/4}$ or less.

Theorem 8.7 *There is a deterministic 2-round algorithm that asks $\frac{5}{4}n^{3/2}(1+o(1))$ queries of size at most $n^{1/4}$ each.*

Proof: Choose $m \approx n^{1/4}$ such that K_n is the disjoint union of approximately $n^{3/2}$ copies of K_m. (Use a projective or affine space of dimension 4 where each line has length m.) (1) Ask one query for each copy of K_m. At most $n/2$ of them can contain an edge. (2) Use brute force to find those edges. □

9 Open Problems

- Determine the smallest possible constant c such that there is a deterministic nonadaptive algorithm for the matching problem on n vertices that makes $c\binom{n}{2}(1+o(1))$ queries.
- Find more efficient deterministic k-round algorithms or prove lower bounds for the number of queries in such algorithms.

Acknowledgments. We would like to thank Lance Fortnow, László Lovász and Oded Regev for helpful discussions. The second author gratefully acknowledges a Temple University study leave.

References

[1] N. Alon and J. H. Spencer, **The Probabilistic Method**, Second Edition, Wiley, New York, 2000.

[2] R. Beigel, N. Alon, M. S. Apaydin, L. Fortnow and S. Kasif, An optimal procedure for gap closing in whole genome shotgun sequencing, Proc. 2001 RECOMB, ACM Press, pp. 22–30.

[3] M. Claustres, P. Kjellberg, M. Desgeorges, H. Bellet, P. Sarda, H. Bonnet, C. Boileau, Detection of deletions by the amplification of exons (multiplex PCR) in Duchenne muscular dystrophy, J. Genet. Hum. 1989 37 (3): 251-257, (in French).

[4] V. Grebinski and G. Kucherov, Optimal Query Bounds for Reconstructing a Hamiltonian Cycle in Complete Graphs, Proc. 5th Israeli Symposium on Theoretical Computer Science (1997), pp. 166–173.

[5] V. Grebinski and G. Kucherov, Reconstructing a Hamiltonian Cycle by Querying the Graph: Application to DNA Physical Mapping, *Discrete Applied Math.* 88 (1998), 147–165.

[6] H. Tettelin, D. Radune, S. Kasif, H. Khouri, and S. Salzberg. Pipette Optimal Multiplexed PCR: Efficiently Closing Whole Genome Shotgun Sequencing Project, *Genomics*, Vol. 62, pp. 500–507, 1999.

[7] R. M. Wilson, Decomposition of complete graphs into subgraphs isomorphic to a given graph, *Congressus Numerantium* **XV** (1975), 647–659.