The Minimum k-Colored Subgraph Problem in Haplotyping and DNA Primer Selection

M.T. Hajiaghayi[*] K. Jain[†] K. Konwar[‡] L.C. Lau[§] I.I. Măndoiu[‡]
A. Russell[‡] A. Shvartsman[‡] V.V. Vazirani[¶]

Abstract

In this paper we consider the minimum k-colored subgraph problem (MkCSP), which is motivated by maximum parsimony based population haplotyping and minimum primer set selection for DNA amplification by multiplex Polymerase Chain Reaction, two important problems in computational biology. We use several new techniques to obtain improved approximation algorithms for both the general MkCSP and some important special cases. We also establish a novel relation between MkCSP and the densest k-subgraph problem, whose approximability is notoriously hard. This relation gives evidence that some of the approximation factors in this paper might be almost tight. Furthermore, this relation could shed light into a potential proof of polynomial inapproximability for the densest k-subgraph problem (using an inapproximability result that we present for a generalized version of the densest k-subgraph problem).

1 Introduction

The *minimum k-colored subgraph problem* (MkCSP) is defined as follows: given an undirected graph G, a color function that assigns to each edge one or more of n given colors, and an integer $k \leq n$, find a minimum set of vertices of G inducing edges of at least k colors. This problem – which has a surprising connection to the densest k-subgraph maximization problem (see Section 4) – is a common generalization of two important problems in computational biology: maximum parsimony based population haplotyping and minimum primer set selection for DNA amplification by multiplex Polymerase Chain Reaction (PCR). An important case of MkCSP happens when $k = n$; we refer to this special case as the *minimum multi-colored subgraph problem* (MMCSP).

1.1 Maximum parsimony based population haplotyping

A *Single Nucleotide Polymorphism*, or SNP, is a position in the genome at which exactly two of the possible four nucleotides occur in a large percentage of the population. SNPs account for most of the genetic variability between individuals, and mapping SNPs in human population has become the next high-priority in genomics after the completion of the Human Genome project. In diploid organisms such as humans, there are two non-identical copies of each chromosome. A description of the SNPs in each chromosome is called a *haplotype*, which can be viewed as a 0/1 vector, e.g., by representing the most frequent (dominant) SNP allele as a 0 and the alternate (minor) allele as a 1. At present, it is prohibitively expensive to directly determine the haplotypes

[*] MIT Computer Science and Artificial Intelligence Laboratory and Microsoft Research. E-mail: hajiagha@csail.mit.edu

[†] Microsoft Research. E-mail: kamalj@microsoft.com.

[‡] CSE Department, University of Connecticut. E-mail: {kishori,ion,acr,aas}@cse.uconn.edu.

[§] Department of Computer Science, University of Toronto and Microsoft Research. E-mail: chi@cs.toronto.edu

[¶] College of Computing, Georgia Institute of Technology. E-mail: vazirani@cc.gatech.edu

of an individual, but it is possible to obtain rather easily the conflated SNP information in the so called *genotype*. A genotype can be conveniently represented as a 0/1/2 vector, where 0 (1) means that both chromosomes contain the dominant (respectively minor) allele, and 2 means that the two chromosomes contain different alleles.

The *population haplotyping problem* (PHP) is to infer the haplotypes from the genotypes of a large population; see [1, 8, 9, 10] for recent surveys on computational methods for solving this problem. A particularly elegant approach to solving PHP is based on the principle of *maximum parsimony*, which postulates that the simplest solution that explains the observed data should be preferred. Adopting this principle leads to the following formulation for PHP [7, 15]: given a set of genotypes $\mathcal{G}$, find the smallest set of haplotypes $\mathcal{H}$ such that for every $g \in \mathcal{G}$ there exist $h, h' \in \mathcal{H}$ with $h + h' = g$, where $h + h'$ is the vector whose i-th component is equal to 2 when $h_i \neq h'_i$, and to the common value of h_i and h'_i when $h_i = h'_i$.

The maximum parsimony PHP can be viewed as a special case of the minimum multi-colored subgraph problem by associating a vertex to each candidate haplotype, and coloring every edge (h, h') by $h+h'$ whenever $h+h'$ is one of the given genotypes. Notice that in the resulting MMCSP instances each edge is assigned at most one color (in fact, color classes form a matching in the underlying graph). This property is no longer true for the more general version of PHP in which the input contains missing data, i.e., when the input consists of *partial genotypes* which are vectors over the alphabet $\{0, 1, 2, *\}$, and the goal is to resolve each "$*$" symbols into a 0, 1, or a 2, and find a smallest possible set of haplotypes that explain resolved haplotypes.

1.2 Primer set selection for multiplex PCR

A critical step in many high-throughput genomic protocols, including SNP genotyping, is the cost-effective amplification of DNA sequences containing loci of interest via biochemical reactions such as the *Polymerase Chain Reaction* (PCR) [13]. In its basic form, PCR requires a pair of short single-stranded DNA sequences called *primers* for each amplification target. More precisely, the two primers must be (perfect or near perfect) reversed Watson-Crick complements of the $3'$ ends of the forward and reverse strands in the double-stranded amplification target (see Figure 1). Typically there is significant freedom in selecting the exact ends of an amplification target, i.e., in selecting PCR primers. Consequently, primers can be optimized individualy with respect to various criteria affecting reaction efficiency, such as primer length, melting temperature, secondary structure, etc.

Multiplex PCR (MP-PCR) is a variation of PCR in which multiple DNA fragments are amplified simultaneously. Like the basic PCR, MP-PCR makes use of two oligonucleotide primers to define the boundaries of each amplification target, but a primer may participate in multiple amplification reactions. In addition to individual constraints on the biochemical properties of the primers, MP-PCR primer selection must ensure various *pairwise* compatibility constraints between selected primers. Since the efficiency of PCR amplification falls off exponentially as the length of the amplification product increases, an important practical constraint is that the two primer binding sites must be within a certain maximum distance of each other. Another common type of pairwise compatibility constraint is the requirement of unique amplification [5]: for every locus there should be a pair of primers that amplifies a DNA fragment surrounding it but no other genome fragment. Subject to these constraints, one would like to minimize the total number of primers required to amplify at least k of the given loci. MP-PCR primer set selection can be easily cast as an instance of MkCSP: each candidate primer becomes a graph vertex and each pair of primers that feasibly amplifies a desired locus becomes an edge colored by the respective locus number. The case in which we want to amplify all given loci is an instance of MMCSP [5].

1.3 Previous work

Gusfield [7] proposed an integer programming formulation for the maximum parsimony population haplotyping problem. He reports that the commercial integer programming solver CPLEX finds optimal solutions in practical running time for instances with up to 50 individuals and up to 30 SNP positions. For the same problem, Wang and Xu [20] recently proposed a greedy heuristic and an exact branch and bound algorithm. Lancia et al. [15] proved that maximum parsimony population haplotyping problem is APX-hard, and gave two straightforward algorithms with approximation factors of $\sqrt{n}$ and q, where n is the number of genotypes and q is the maximum number of haplotype pairs compatible with a genotype.[1] These results immediately imply APX-hardness of MMCSP and MkCSP (even when only one color can be assigned to each edge), and give approximation factors of $\sqrt{n}$ and $\mathfrak{m}$ for MMCSP with one color per edge, where n is the number of colors and $\mathfrak{m}$ is the maximum size of a color class.

Fernandes and Skiena [5] studied MMCSP with at most one color per edge in the context of multi-use primer selection for synthesis of spotted microarrays. They gave practical greedy and densest-subgraph based heuristics for the the problem and proved, by a direct reduction from set cover, that even this special case of MkCSP cannot be approximated within a factor better than $(1 - o(1)) \ln n - o(1)$, where again n is the number of colors.

1.4 Our results and techniques

In this paper we give several approximation algorithms for MkCSP and its important special case MMCSP. Our results provide surprising hardness for MkCSP and improve over the approximations given in [15] for MMCSP with at most one color per edge. The improved approximation factors hold for the general formulation of MkCSP, in which multiple colors can be assigned to an edge and we want to cover only k of the n color classes; no non-trivial approximations were previously known for this version. Our contributions are as follows:

- First, we present a non-trivial $\sqrt{k \ln \Delta}$ approximation algorithm for MkCSP using an algorithm of Slavik [18] for the partial set cover problem. Here Δ is the maximum number of colors assigned to an edge.

- Then, we present evidence of potential polynomial inapproximability for MkCSP problem by showing a novel reduction from the densest k-subgraph maximization problem to our minimization problem. We believe that our approach can serve as a general technique to reduce hardness from other budgeted graph-theoretic maximization problems to the corresponding minimization problems. This surprising relation between densest k-subgraph and MkCSP is of its own interest, since not only it gives evidence that most approximation factors established in this paper might be almost tight, but it could also shed light into a potential proof of polynomial inapproximability for the densest k-subgraph problem (using an inapproximability result that we present for the generalized version of densest k-subgraph, called *densest k-subhypergraph*, see Section 7).

- Next, we give an $O(\sqrt{\mathfrak{m}} \log n)$ approximation algorithm for MMCSP, where $\mathfrak{m}$ is the maximum size of a color class (i.e., the maximum number of edges sharing the same color) and n is the number of colors. In the context of PCR primer set selection with amplification length and uniqueness constraints $\mathfrak{m} = O(L^2)$, where L is the upperbound on the amplification length. Hence, our result implies an approximation factor of $O(L \log n)$, which asymptotically improves over the approximation factor of $\min\{\sqrt{n}, L^2\}$ implied by the results of [15]. Our approximation algorithm for MMCSP (see Section 5) is based on LP-rounding. We also show that our LP-rounding method is almost tight by showing a matching (up to the log factor) integrality gap for the LP.

- Last but not least, we show that minimum primer set selection can be approximated within a factor of $\ln(nL)$ when only amplification lengths are imposed. We obtain this result by modelling the problem as a string-pair generalization of the partial set cover problem [18].

[1] Note that $q = 2^{t-1}$, where t is the maximum number of 2's in a genotype.

Since the problem cannot be approximated within a factor of $(1 - o(1)) \ln n$, this implies that our approximation factor is optimal up to an additive term of $\ln L$. Our algorithm is a modified version of the classical greedy algorithm for set cover in which choices are made based on the change in an appropriate potential function. The idea of using a potential function to drive the greedy algorithm appears to be new and is probably of independent interest.

Preliminary empirical results [12] show that our greedy algorithm for primer selection with amplification constraints significantly outperforms previously published algorithms [17, 19] in solution quality and/or running time.

2 Notations and problem formulations

Let $G = (V, E)$ be an undirected graph and $\chi_1, \ldots, \chi_n \subset E$ a family of nonempty "color classes" of edges with the property that $\bigcup_i \chi_i = E$. The *minimum k-colored subgraph problem* (MkCSP) is to compute a minimum size set I of vertices inducing at least one edge of at least k of total n colors. The *minimum multi-colored subgraph problem* (MMCSP) is the special case of MkCSP in which $k = n$. Assigning $\mathcal{X} = (\chi_1, \ldots, \chi_n)$, we will denote by $\mathcal{I}(k, G, \mathcal{X})$ ($\mathcal{I}(G, \mathcal{X})$) the size of an optimal solution for MkCSP, respectively MMCSP. Note that $2 \le \mathcal{I}(k, G, \mathcal{X}) \le 2k$ and, as an edge may belong to several distinct color classes, both of these extreme values are in fact possible. We denote by Δ the maximum number of colors assigned to an edge.

3 General approximation algorithm for MkCSP

In this section we give the first non-trivial approximation algorithm for MkCSP.

Theorem 1 *There exists an approximation algorithm with factor $\sqrt{2kH(\Delta)} = O(\sqrt{k} \ln \Delta)$ for MkCSP.*

Proof. The algorithm is as follows. Let X be the set of selected vertices; initially empty. While the number of colors covered is less than k, we choose an edge with maximum number of uncovered colors and add both of its endpoints to X (if they are not already in X). Let I be the number of edges that we choose in this process. We know that $|X| \le 2i$. On the other hand, by a result of Slavik [18], we know that the above greedy algorithm for the partial set cover problem, i.e., finding the minimum number of sets to cover at least k elements, is an $H(\Delta)$ approximation algorithm. This means that the minimum number of edges needed to cover at least k colors is at least $i/H(\Delta)$. It is easy to see that, in order to induce at least $i/H(\Delta)$ edges, the optimum MkCSP solution should pick at least $\sqrt{2i/H(\Delta)}$ vertices. The approximation factor follows immediately by using this lower bound. ■

Remark. For the case when $k = n$ and $\Delta = 1$, i.e., for MMCSP with one color per edge, the above algorithm corresponds to the $\sqrt{k}$-approximation algorithm of [15]. It is also worth mentioning that using the approximation algorithm of Gandhi, Khuller and Srinvasan [6] for partial set cover in the proof of Theorem 1, we can obtain an $\sqrt{2k\mathfrak{m}}$ approximation algorithm for MkCSP, where $\mathfrak{m}$ is the maximum number of edges sharing the same color. In next section we establishing an interesting reduction from MkCSP to the densest k-subgraph problem, showing that the approximation factor in Theorem 1 cannot be easily improved.

4 Hardness result for MkCSP

In this section, we show an interesting relation between MkCSP and the densest k-subgraph problem. Formally, we show that if there is a polynomial time f-approximation algorithm $\mathcal{A}$ for MkCSP, then there is a polynomial time $2f^2$-approximation algorithm for the densest k-subgraph problem. Given a graph G and a parameter k, the densest k-subgraph problem is to find a set of k vertices with maximum number of induced edges. The densest k-subgraph problem is well-studied in the literature [4, 11]. The best known approximation factor for the densest k-subgraph problem is $O(n^{1/3-\epsilon})$ for some small $\epsilon > 0$ and improvement is known to be hard [2, 11]. The connection between MkCSP and the densest k-subgraph problem suggests that significant improvements in the approximation ratio for MkCSP would require substantially new ideas.

Theorem 2 *If there is a polynomial time f-approximation algorithm $\mathcal{A}$ for MkCSP, then there is a polynomial time $2f^2$-approximation algorithm for the densest k-subgraph problem.*

Proof. Given a graph G with m edges, we would like to find a set of k vertices with maximum number of edges in the subgraph induced by this set. We assign to each edge of G a different color and use $\mathcal{A}$ to find the approximate solutions for MkCSP on the resulting graph. Suppose l is the maximum color coverage requirement for which $\mathcal{A}$ outputs a solution Y with at most k vertices. That is, there are l colors assigned to the subgraph induced by Y, and the approximate solution returned by $\mathcal{A}$ when $l+1$ colors are required to be covered contains at least $k+1$ vertices. Let the optimal solution to the densest k-subgraph problem contain *opt* edges. We shall prove that $opt \leq 2f^2 l$ and thus Y is a solution to the densest k-subgraph problem which is within a factor of $\frac{1}{2f^2}$ to the optimal solution.

By our choice of l and the fact that $\mathcal{A}$ is an f-approximation algorithm, any $\frac{k}{f}$ vertices of G can induce at most l colors. Consider a subset X with k vertices. The total number of colors induced by all possible subsets of $\frac{k}{f}$ elements of X is at most $\binom{k}{\frac{k}{f}}l$. Notice that each edge is counted exactly $\binom{k-2}{\frac{k}{f}-2}$ times. So, the total number of edges in X is at most

$$\frac{\binom{k}{\frac{k}{f}}l}{\binom{k-2}{\frac{k}{f}-2}} = \frac{k(k-1)}{\frac{k}{f}(\frac{k}{f}-1)}l \leq f^2 l(\frac{k-1}{k-f}) < 2f^2 l$$

(The last inequality holds since we can assume without loss of generality that $k > 2f$, otherwise, any single edge is a $2f^2$-approximation). Since X is an arbitrary set with k vertices, $opt \leq 2f^2 l$ and this completes the proof. ■

5 LP-rounding based approximation

In this section we consider MMCSP, the important case of MkCSP when $n = k$, for which we present an improved approximation algorithm using LP-rounding techniques. In addition, we show that the approximation factor of the algorithm is almost tight (up to a logarithmic factor) by showing a matching integrality gap for our LP.

Theorem 3 *MMCSP can be approximated to within an approximation factor of $O(\sqrt{\mathfrak{m}}\log|\mathcal{X}|)$ in polynomial time, where $\mathfrak{m} = \max_{\chi\in\mathcal{X}} \mid \chi \mid$.*

Proof. We use the following integer program formulation of MMCSP:

$$\begin{aligned}
&\min \sum_v x_v, \text{ subject to} \\
&\qquad \forall \chi \in \mathcal{X}, \sum_{e \in \chi} y_e \geq 1 \ , \\
&\qquad \forall v \in V, \forall \chi \in \mathcal{X}, \sum_{v \in e \in \chi} y_e \leq x_v \ , \\
&\qquad \forall e \in E, y_e \geq 0, \forall v \in V, x_v \geq 0 \ .
\end{aligned}$$

Relaxing this formulation by allowing the variables x_v and y_e to take values in $[0,1]$ results in a linear program, the optimum value of which we denote by $\mathcal{I}_\ell(G,\mathcal{X})$. We begin by scaling the linear program to obtain the following new linear program:

$$\begin{aligned}
&\min \sum_v x_v, \text{ subject to} \\
&\qquad \forall \chi \in \mathcal{X}, \sum_{e \in \chi} y_e \geq \sqrt{\mathfrak{m}} \ , \\
&\qquad \forall v \in V, \forall \chi \in \mathcal{X}, \sum_{v \in e \in \chi} y_e \leq x_v \ , \\
&\qquad \forall e \in E, y_e \geq 0, \forall v \in V, x_v \geq 0 \ .
\end{aligned}$$

Let $\mathcal{I}_\ell^s(G,\mathcal{X})$ denote the optimum value for the scaled LP and let $x^* \in \mathbb{R}^V$ and $y^* \in \mathbb{R}^E$ denote an optimal solution. Clearly, $\mathcal{I}_\ell^s(G,\mathcal{X}) \leq \sqrt{\mathfrak{m}} \cdot \mathcal{I}_\ell(G,\mathcal{X})$. Based on the solution (x^*, y^*) above we define a family of (artificial) independent $\{0,1\}$-valued random variables $\{Z_{v,e} \mid v \in e, v \in V, e \in E\}$, where $\Pr[Z_{v,e} = 1] = p_e \triangleq \min(y_e^*, 1)$ for each $v \in e$. In terms of these variables, define, for each $v \in V$ and each $(u,v) = e \in E$, the variables $X_v = \bigvee_{v \in e \in E} Z_{v,e}$ and $Y_e = Z_{u,e} Z_{v,e}$. Finally, let variables X_v determine the random set of vertices $S = \{v \mid X_v = 1\}$. Our goal is to show that, for each color class χ, the set S is likely to induce an edge in χ.

Comment. Observe that indicator variable for the event that the set S induces the edge $e = (u,v)$ is $X_u X_v$ which dominates the variable $Y_{(u,v)}$. We focus on this second, less natural, set of variables because, unlike the variables $X_u X_v$, the $Y_{(u,v)}$ are independent.

With this in mind, note that $\Pr[Y_e = 1] = (p_e)^2$ and that for each v

$$\begin{aligned}
\Pr[v \in S] = \Pr[X_v = 1] &= \left(1 - \prod_{v \in e} \Pr[Z_{v,e} = 0]\right) = \left(1 - \prod_{v \in e}(1 - p_e)\right) \\
&\leq \left(1 - \left(1 - \sum_{v \in e} p_e\right)\right) \leq \sum_{v \in e} y_e^* \leq x_v^* \ .
\end{aligned}$$

Hence, by linearity of expectation

$$\mathrm{Exp}\left[|S|\right] = \mathrm{Exp}\left[\sum_v X_v\right] \leq \mathcal{I}_\ell^s(G,\mathcal{X}) \leq \sqrt{\mathfrak{m}} \cdot \mathcal{I}_\ell(G,\mathcal{X}) \leq \sqrt{\mathfrak{m}} \cdot \mathcal{I}(G,\mathcal{X}) \ .$$

We wish to upper bound, for each color class χ, the quantity

$$\Pr[\forall e \in \chi, Y_e = 0] = \Pr\left[S \text{ induces no edge from } \chi\right]$$

with the intention of showing that this selection S of vertices is likely to induce many color classes. So, consider now an arbitrary color class χ; then

$$\text{Exp}\left[\sum_{e\in\chi} X_u X_v\right] \geq \text{Exp}\left[\sum_{e\in\chi} Y_e\right] = \sum_{e\in\chi} p_e^2 \geq |\ \chi\ | \cdot \left(\frac{\sqrt{\mathfrak{m}}}{|\ \chi\ |}\right)^2 \geq 1\ ,$$

as $\sum_{e\in\chi} p_e \geq \sqrt{\mathfrak{m}}$ and the function $x \mapsto x^2$ is convex. Considering that the Y_e are independent, we compute

$$\begin{aligned}\Pr[\chi \text{ not induced by } S] &= \Pr[\forall(u,v)\in\chi, X_u X_v = 0] \leq \Pr\left[\forall e\in\chi, Y_e = 0\right]\\ &= \prod_{e\in\chi}(1-p_e^2) \leq \prod_{e\in\chi} e^{-p_e^2} = e^{-\sum_{e\in\chi} p_e^2} \geq e^{-1}\ .\end{aligned}$$

Evidently, selection of S as above "covers" any individual class χ with constant probability. So, finally, consider the set S of vertices obtained by repeating the above procedure $t = (\log k + 2)$ times and taking the union. Then

$$\text{Exp}[|\ S\ |] \leq \sqrt{\mathfrak{m}}(\log k + 2)\cdot \mathcal{I}(G,\mathcal{X})$$

so that by Markov's inequality, the probability that $|\ S\ |$ exceeds this value by a factor 3 is no more than 1/3. In addition, the probability that S fails to induce an edge in all of the color classes is

$$\Pr[\exists\chi\in\mathcal{X}, \text{no edge of } \chi \text{ induced by } S] \leq |\mathcal{X}|\cdot\left(e^{-1}\right)^{\log|\mathcal{X}|+2} = e^{-2} \leq 1/3\ .$$

Hence with constant probability this procedure results in a collection of vertices that induces at least one edge of each color class and has cardinality no more than $O(\sqrt{\mathfrak{m}}\log|\mathcal{X}|)\mathcal{I}(G,\mathcal{X})$, as desired. ■

We show below that the integrality gap of the LP defining $\mathcal{I}_\ell(G,\mathcal{X})$ is $\Omega(\sqrt{\mathfrak{m}})$ in general. This suggests that this particular LP formulation may have limited value in achieving approximation results beyond the $\sqrt{\mathfrak{m}}$ threshold.

Theorem 4 *For every $s \geq 0$ there is a pair $(G,\mathcal{X})$ for which $\mathfrak{m} = s$ and $\mathcal{I}(G,\mathcal{X}) \geq \Omega(\sqrt{\mathfrak{m}})\mathcal{I}_\ell(G,\mathcal{X})$.*

Proof. Consider the graph on $n \gg s$ vertices obtained by selecting, independently and uniformly at random, n matchings $\chi_1,\ldots,\chi_n$, each of size s, and assigning $E = \bigcup_{i=1}^n \chi_i$. Observe that the feasible solution obtained by setting $x_v = y_e = 1/s$ for all e and v implies that $\mathcal{I}_\ell(G,\mathcal{X}) \leq n/s$.

On the other hand, we show that with high probability, this random selection of matchings results in a graph for which the smallest integer solution has objective value at least $\ell \triangleq (n-1)/\sqrt{2s}$. Specifically, let $L \subset V$ be a fixed collection of ℓ vertices and note that the probability that any given edge induced by L is included in, e.g., χ_1 is $s/\binom{n}{2}$; hence the probability that L induces an edge of each color is no more than

$$\left(\frac{s}{\binom{n}{2}}\binom{\ell}{2}\right)^n \leq \left(\frac{s\ell^2}{(n-1)^2}\right)^n \leq \left(\frac{1}{2}\right)^n$$

Hence the probability that some set of ℓ vertices induces an edge of each color is no more than $\binom{n}{\ell}2^{-n} < 1$. Evidently, there exists a family of color classes $\mathcal{X} = (\chi_1,\ldots,\chi_n)$ for which $\mathcal{I}(G,\mathcal{X}) \geq \Theta(\sqrt{\mathfrak{m}})\mathcal{I}_\ell(G,\mathcal{X})$, as desired. ■

It is worth mentioning the integrality gap in Theorem 4 holds for maximum parsimony PHP as well. As mentioned in Subsection 1.1, in this case, the graph is more restricted, that is, each vertex is a 0/1 vector and each edge between vertices h and h' has a unique color $h + h'$ (which is a 0/1/2 vector). Still we can construct such a restricted graph which shows the integrality gap is the same as that of Theorem 4. In the interest of space we omit the details from this extended abstract.

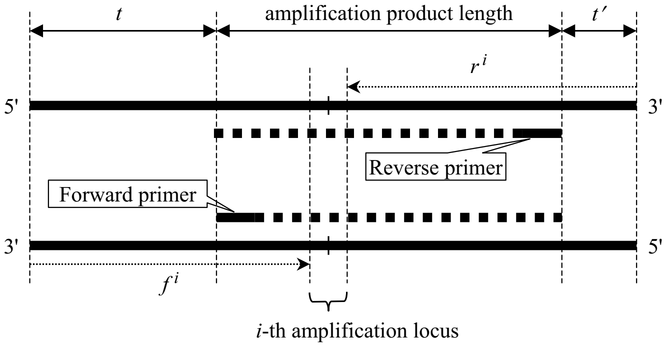

Figure 1: Strings f^i and r^i consist of the L DNA bases immediately preceding in $3' - 5'$ order the i-th amplification locus along the forward (respectively reverse) genomic sequence. If forward and reverse PCR primers cover f^i and r^i at positions t, respectively t', then the PCR amplification product length is $(2L + x) - (t + t')$, where x is the length of the amplification locus ($x = 1$ for SNP genotyping). Thus, amplification product length is at most $L + x$ iff $t + t' \geq L$.

6 Improved approximation for primer set selection with amplification length constraints

As discussed in Section 1.2, primer set selection for multiplex PCR with amplification length and uniqueness constraints is a special case of MMCSP. In this section we show that when only amplification length constraints are imposed we can obtain a significantly improved approximation factor.

Let $\Sigma = \{a, c, g, t\}$ be the DNA alphabet. We denote by Σ^* the set of strings over Σ, and by λ the empty string. Overloading notations, we use $|\cdot|$ to denote both the length of strings over Σ and the size of sets. For a string s and an integer $t < |s|$, we denote by $s[1..t]$ the prefix of length t of s. We denote by L the given threshold on the PCR amplification length, and by f^i (respectively r^i) the string consisting of the L DNA bases immediately preceding in $3' - 5'$ order the i-th amplification locus along the forward (respectively reverse) DNA genomic sequence (see Figure 1).

We say that primer $p = p_1 p_2 \dots p_l$ *covers* (or *hybridizes at*) position i of string $s = s_1 s_2 \dots s_m$ iff i is the largest index such that $s_i s_{i+1} \dots s_{i+l-1}$ is the reversed Watson-Crick complement of p, i.e., iff s_{i+j} is the Watson-Crick complement of p_{l-j} for every $0 \leq j \leq l - 1$.[2] A set of primers P is an *L-restricted primer cover* for the pairs of sequences $(f^i, r^i) \in \Sigma^L \times \Sigma^L$, $i = 1, \dots, n$, iff for every $i = 1, \dots, k$, there exist primers $p, p' \in P$, not necessarily distinct, and integers $t, t' \in \{1, \dots, L\}$, such that

1. p hybridizes at position t of f^i;
2. p' hybridizes at position t' of r^i; and
3. $t + t' \geq L$

The last constraint ensures that the PCR amplification product length is no more than $L + x$, where x is the length of the desired amplification target ($x = 1$ for SNP genotyping).

[2] A promising approach to further increasing MP-PCR efficiency is the use of *degenerate PCR primers* [14, 16, 19]. For simplicity, we consider only non-degenerate primers here, but note that our algorithm guarantees the same approximation factor for the problem of selecting a minimum set of degerate primers amplifying the given set of targets.

```
Input: Primer length $l$, amplification length upperbound $L$, and pairs of sequences $(f^i, r^i) \in \Sigma^L \times \Sigma^L$,
$i = 1, \ldots, n$
Output: $L$-restricted primer cover $P$ consisting of primers of length $l$
--------------------------------------------------------------------------------

Function $\Delta(p, i)$:
$\Delta \leftarrow 0$
If $|\overline{f}^i| + |\overline{r}^i| \geq L$ return 0
If $p$ covers $f^i$ at position $t > |\overline{f}^i|$, $\Delta \leftarrow \Delta + (t - |\overline{f}^i|)$
If $p$ covers $r^i$ at position $t > |\overline{r}^i|$, $\Delta \leftarrow \Delta + (t - |\overline{r}^i|)$
Return $\min\{\Delta, L - (|\overline{f}^i| + |\overline{r}^i|)\}$


$P \leftarrow \emptyset$; For every $i = 1, \ldots, n$, $\overline{f}^i \leftarrow \overline{r}^i \leftarrow \lambda$
While $\Phi(P) := \sum_{i=1}^n \min\{L, |\overline{f}^i| + |\overline{r}^i|\} < nL$ do
    Find the primer $p$ maximizing $\Delta\Phi = \sum_{i=1}^n \Delta(p, i)$
    For every $i = 1, \ldots, n$,
        If $p$ covers $f^i$ at position $t > |\overline{f}^i|$ then $\overline{f}^i \leftarrow f^i[1..t]$
        If $p$ covers $r^i$ at position $t > |\overline{r}^i|$ then $\overline{r}^i \leftarrow r^i[1..t]$
    $P \leftarrow P \cup \{p\}$
Return $P$
```

Figure 2: The greedy algorithm for MPSS-L

> **Minimum primer set selection problem with amplification length constraints (MPSS-L):** Given primer length l, amplification length upperbound L, and n pairs of sequences (f^i, r^i), $i = 1, \ldots, n$, find a minimum size L-restricted primer cover consisting of primers of length l.

MPSS-L can be viewed as a generalization of the partial set cover problem [18]. In the partial set cover problem one must cover with the minimum number of sets a given fraction of the total number of elements. In MPSS-L we can take the elements to be covered to be the non-empty prefixes of the $2n$ forward and reverse sequences; there are $2nL$ such elements. A primer p covers prefix $f^i[1..j]$ ($r^i[1..j]$) if it hybridizes to f^i (respectively r^i) at position $t \geq j$. The objective is to cover at least L (i.e., half) of the elements of $\{f^i[1..j], r^i[1..j] \mid 1 \leq j \leq L\}$ for every $i \in \{1, \ldots, n\}$.

For a set of primers P, let $\overline{f}^i$ and $\overline{r}^i$ denote the longest prefix of f^i, respectively r^i, covered by a primer in P. Note that $|\overline{f}^i| + |\overline{r}^i|$ gives the number of elements of $\{f^i[1..j], r^i[1..j] \mid 1 \leq j \leq L\}$ that are covered by P. Let $\Phi(P) := \min\{L, |\overline{f}^i| + |\overline{r}^i|\}$. Note that $\Phi(\emptyset) = 0$, $\Phi(P) = nL$ for every feasible MPSS-L solution, and that $\Phi(P) \leq \Phi(P')$ whenever $P \subseteq P'$.

Our greedy algorithm uses $\Phi(\cdot)$ as a measure of the progress made towards feasibility. The algorithm (see Figure 2) starts with an empty set of primers and iteratively selects primers which give the largest increase in Φ until reaching feasibility.

Theorem 5 *The greedy algorithm returns an L-restricted primer cover of size at most $\ln(nL)$ times larger than the optimum.*

Proof. Let OPT denote a minimum size L-restricted primer cover, and let $p_1, \ldots, p_g$ be the primers selected by the greedy algorithm. It can be verified that, for every A and B, $\Phi(A \cup B) \leq$

$\Phi(A) + \sum_{p \in B}[\Phi(A \cup \{p\}) - \Phi(A)]$. By using this claim with $A = \{p_1, \ldots, p_{i-1}\}$ and $B = OPT$, it follows that in the step when the greedy algorithm selects p_i, there is a primer in $\mathrm{OPT} \setminus \{p_1, \ldots, p_{i-1}\}$ whose selection increases Φ by at least $(nL - \Phi(P))/|\mathrm{OPT}|$. Hence, the selection of p_i must increase Φ by at least the same amount, i.e., reduce the difference between $\Phi(\mathrm{OPT})$ and $\Phi(P)$ by a factor of at least $(1 - 1/|\mathrm{OPT}|)$. By induction we get that

$$nL - \Phi(\{p_1, \ldots, p_i\}) \leq nL \left(1 - \frac{1}{|\mathrm{OPT}|}\right)^i \tag{1}$$

which implies that the number of primers selected by the greedy algorithm is at most $\ln(nL)$. ■

Remark. In [17] it is proved that the following primer cover problem is as hard to approximate as set cover: Given integer l and strings $s_1, \ldots, s_n$, find a minimum set of l-length primers covering all s_i's. A simple approximation preserving reduction of the primer cover problem to MPSS-L shows that the MPSS-L is also as hard to approximate as set cover. Hence, the approximation factor in Theorem 5 is tight up to an additive term of $\ln L$, unless $\mathrm{NP} \subseteq \mathrm{TIME}(n^{O(\log \log n)})$.

7 Further results and discussion

Motivated by Theorem 2, we investigate the hardness of the densest k-subgraph problem and show that the following closely related problem is hard to approximate to within a factor of $2^{(\log n)^\delta}$ for some $\delta > 0$ under the assumption that 3-SAT $\notin \mathrm{DTIME}(2^{n^{3/4+\epsilon}})$.

> **Densest k-subhypergraph:** Given a hypergraph $G = (V, E)$ and a parameter k, find a set of k vertices with maximum number of hyperedges in the subgraph induced by this set.

Recently, Khot [11] proved that there exists a constant ϵ such that it is hard to approximate the densest k-subgraph problem to within a $(1 + \epsilon)$-factor under the assumption that NP has no subexponential time algorithms. We show the hardness of the densest k-subhypergraph problem by a reduction from the maximum balanced complete bipartite subgraph problem.

> **Maximum Balanced Complete Bipartite Subgraph:** Given a bipartite graph $G = (X, Y, E)$, find a maximum balanced complete bipartite subgraph (i.e. with the maximum number of vertices). Here, a balanced complete bipartite subgraph H is a complete bipartite subgraph such that $|H \cap X| = |H \cap Y|$.

Theorem 6 *If there is a polynomial time f-approximation algorithm $\mathcal{A}$ for the densest k-subhypergraph problem, then there is a polynomial time f-approximation algorithm for the maximum balanced complete bipartite subgraph problem.*

Proof. To prove this theorem, we first transform the densest k-subhypergraph in the following way. Given a hypergraph $G = (V, E)$, we construct a bipartite graph $G' = (X, Y, E')$ so that $X = V$ and $Y = E$. A vertex $x \in X$ is adjacent to a vertex $y \in Y$ in G' if and only if the corresponding hyperedge y contains x. Now, the problem of finding a densest k-subhypergraph is equivalent to the following *bipartite k-coverage* problem: find a set S of k vertices in X in G' such that the cardinality of $Y(S)$ is maximized where $Y(S) = \{v \in Y \mid N(v) \subseteq S\}$. Note that the bipartite k-coverage problem can also be reduced to the densest k-subhypergraph problem by a reserve procedure, so $\mathcal{A}$ is a f-approximation algorithm on the bipartite k-coverage problem.

Now, given a bipartite graph G', we would like to find the maximum balanced complete bipartite graph in G'. In order to give a f-approximation for the maximum balanced complete bipartite subgraph problem, we shall use $\mathcal{A}$ in the following way. First, we construct a bipartite graph $\overline{G} = (X, Y, \overline{E})$ so that $x \in X$ is adjacent to $y \in Y$ in $\overline{G}$ if and only if $x \in X$ is not adjacent to $y \in Y$ in G'. So, a balanced complete bipartite subgraph in G' is a balanced bipartite independent set in $\overline{G}$. Then, we use $\mathcal{A}$ to solve the bipartite k-coverage problem in $\overline{G}$ with k from 1 to $|X|$. Let the output of the k-th run be S_k and $Y(S_k)$. Notice that $Y(S_k)$ and $X - S_k$ is a bipartite independent set in $\overline{G}$ and so we have a balanced bipartite independent set of size $\min\{|Y(S_i)|, |X - S_i|\}$ in $\overline{G}$. Finally, we return the maximum balanced bipartite independent set in $\overline{G}$ amongst the $|X|$ runs of $\mathcal{A}$. Let *opt* be the size of the optimal solution to the maximum balanced complete bipartite subgraph problem in G'. We shall show that the approximation solution we returned has size at least $\frac{opt}{f}$.

To see this, consider the iteration when $\mathcal{A}$ is run with $k^* = |X| - opt$. In this instance, we know that there is a $S \subseteq X$ of size k^* and a $Y(S) \subseteq Y$ of size at least *opt* such that $N(y) \subseteq S$ for $y \in Y(S)$; otherwise there is no balanced bipartite independent set of size *opt* in $\overline{G}$ and thus no balanced complete bipartite subgraph in G'. Since $\mathcal{A}$ is a f-approximation algorithm, it will output a set $S_{k^*} \subseteq X$ of size k^* and a set $Y(S_{k^*}) \subseteq Y$ of size at least $\frac{opt}{f}$ such that $N(y) \subseteq S_{k^*}$ for $y \in Y(S_{k^*})$. This implies that $Y(S_{k^*})$ and $X - S_{k^*}$ form a bipartite independent set. From $Y(S_{k^*})$ and $X - S_{k^*}$, we can obtain a balanced bipartite independent set in $\overline{G}$ of size at least $\min\{|Y(S_{k^*})|, |X - S_{k^*}| \geq \frac{opt}{f}$ (recall that $k^* = |X| - opt$), and hence a complete bipartite complete subgraph of size at least $\frac{opt}{f}$ in G'. This completes the proof of the theorem. ■

Theorem 7 *The densest k-subhypergraph problem is hard to approximate within a factor of* $2^{(\log n)^\delta}$ *for some $\delta > 0$ under the (plausible) assumption that 3-SAT $\notin$ DTIME*$(2^{n^{3/4+\epsilon}})$.

Proof. The corollary follows from Theorem 6 and the hardness result for the maximum complete bipartite subgraph problem in [3]. ■

References

[1] P. Bonizzoni, G. D. Vedova, R. Dondi, and J. Li, *The haplotyping problem: An overview of computational models and solutions*, Journal of Computer Science and Technology, 18 (2003), pp. 675–688.

[2] U. Feige, *Relations between average case complexity and approximation complexity*, in Proceedings of the thiry-fourth annual ACM symposium on Theory of computing, ACM Press, 2002, pp. 534–543.

[3] U. Feige and S. Kogan, *Hardness of approximation of the balanced complete bipartite subgraph problem*, Technical report MCS04-04, Department of Computer Science and Applied Math., The Weizmann Institute of Science, (2004).

[4] U. Feige, G. Kortsarz, and D. Peleg, *The dense k-subgraph problem*, Algorithmica, 29 (2001), pp. 410–421.

[5] R. Fernandes and S. Skiena, *Microarray synthesis through multiple-use PCR primer design*, Bioinformatics, 18 (2002), pp. S128–S135.

[6] R. Gandhi, S. Khuller, and A. Srinivasan, *Approximation algorithms for partial covering problems*, in ICALP'01, 2001, pp. 225–236.

[7] D. Gusfield, *Haplotype inference by pure parsimony*, Tech. Report Technical Report CSE-2003-2, Department of Computer Science, University of California at Davis, 2003.

[8] ———, *An overview of combinatorial methods for haplotype inference*, in Proc. of the DIMACS/RECOMB Satellite Workshop on Computational Methods for SNPs and Haplotype Inference, vol. 2983 of Lecture Notes in Bioinformatics, Berlin, 2004, Springer-Verlag, pp. 9–25.

[9] B. Halldorsson, V. Bafna, N. Edwards, R. Lippert, S. Yooseph, and S. Istrail, *Combinatorial problems arising in SNP and haplotype analysis*, in Proc. DMTCS 2003, vol. 2731 of Lecture Notes in Computer Science, Berlin, 2003, Springer-Verlag, pp. 26–47.

[10] ——, *A survey of computational methods for determining haplotypes*, in Proc. of the DIMACS/RECOMB Satellite Workshop on Computational Methods for SNPs and Haplotype Inference, vol. 2983 of Lecture Notes in Bioinformatics, Berlin, 2004, Springer-Verlag, pp. 26–47.

[11] S. Khot, *Ruling out PTAS for graph min-bisection, densest subgraph and bipartite clique*, in FOCS'04, 2004. To appear.

[12] K. Konwar, I. Mandoiu, A. Russell, and A. Shvartsman, *Approximation algorithms for minimum pcr primer set selection with amplification length and uniqueness constraints.* ACM Computing Research Repository, cs.DS/0406053, 2004.

[13] P. Kwok, *Methods for genotyping single nucleotide polymorphisms*, Annual Review of Genomics and Human Genetics, 2 (2001), pp. 235–258.

[14] S. Kwok, S. Chang, J. Sninsky, and A. Wong, *A guide to the design and use of mismatched and degenerate primers*, PCR Methods and Appl., 3 (1994), pp. S539–S547.

[15] G. Lancia, C. Pinotti, and R. Rizzi, *Haplotyping populations: complexity and approximations*, Tech. Report Technical Report DIT-02-0080, University of Trento, 2002.

[16] C. Linhart and R. Shamir, *The degenerate primer design problem*, Bioinformatics, 18 (2002), pp. S172–S181.

[17] W. Pearson, G. Robins, D. Wrege, and T. Zhang, *On the primer selection problem for polymerase chain reaction experiments*, Discrete and Applied Mathematics, 71 (1996), pp. 231–246.

[18] P. Slavik, *Improved performance of the greedy algorithm for partial cover*, Information Processing Letters, 64 (1997), pp. 251–254.

[19] R. Souvenir, J. Buhler, G. Stormo, and W. Zhang, *Selecting degenerate multiplex PCR primers*, in Proc. 3rd Intl. Workshop on Algorithms in Bioinformatics (WABI), 2003, pp. 512–526.

[20] L. Wang and Y. Xu, *Haplotype inference by maximum parsimony*, Bioinformatics, 19 (2003), pp. 1773–1780.