Multiplexing Algorithms for High-Throughput Genomic Based Assays

Claudia Prăjescu

B.S., A.I.Cuza University, Iaşi, Romania, 2004

A Thesis

Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Science

at the

University of Connecticut

2005

APPROVAL PAGE

Master of Science Thesis

Multiplexing Algorithms for High-Throughput Genomic Based Assays

Presented by

Claudia Prăjescu

Major Advisor: _____

Ion Mandoiu

Associate Advisor: _____

Laurent Michel

Associate Advisor: _____

Alexander Russell

University of Connecticut

2005

Acknowledgements

I most especially thank my advisor, Ion Măndoiu, without whom this work would not have been done. For giving me the opportunity to study here, for all his tireless help, for invaluable advice and suggestions, I am most grateful to him.

I also want to express my gratitude to my former advisor at the A.I.Cuza University of Iaşi, Ferucio Laurenţiu Ţiplea, for believing in me.

Additionally, I would like to thank my associate advisors, Laurent Michel and Alexander Russell, for serving on my examination committee and also my colleagues, Jin Jun and Bogdan Paşaniuc, for kindly reviewing part of this thesis.

Finally, I acknowledge the constant support and encouragements I have received from my family and friends.

Abstract

After the completion of the Human Genome Project has provided a blueprint of the DNA present in each human cell, genomics research is now focusing on the study of DNA variations that occur between individuals, seeking to understand how these variations confer susceptibility to common diseases such as diabetes or cancer. The most common form of genomic variation are the so called *single nucleotide polymorphisms* (SNPs), i.e., the presence of different DNA nucleotides, or *alleles*, at certain chromosomal locations. Determining the identity of the alleles present in a DNA sample at a given set of SNP loci is called *SNP genotyping*.

Among emerging genotyping technologies, one of the most promising is the use of universal tag arrays, which provide unprecedented assay customization flexibility while maintaining a high degree of multiplexing and low unit cost. In the first part of this thesis we study methods for improving the multiplexing rate (defined as the average number of reactions assayed per array) in SNP genotyping assays involving multiple universal tag arrays. In general, it is not possible to use all tags in an array experiment due to, e.g., unwanted hybridizations. An assay specific optimization that determines the multiplexing rate (and hence the number of required arrays for a large assay) is the *tag assignment problem*, whereby individual tags are assigned to the primers used to genotype each SNP. We observe that significant improvements in multiplexing rate can be achieved by combining primer selection with tag assignment. For most tag array applications there are multiple primers with the desired functionality; for example in SNP genotyping one can choose the corresponding primer from either the forward or reverse strands. Since different primers hybridize to different sets of tags, a higher multiplexing rate is achieved by integrating primer selection with tag assignment. This integrated optimization is shown to lead to a reduction of up to 50% in the number of required arrays.

In the second part of the thesis, we propose a new genotyping assay architecture combining multiplexed solution-phase single-base extension (SBE) reactions with sequencing

by hybridization (SBH) using universal DNA arrays such as all k-mer arrays. In addition to PCR amplification of genomic DNA, SNP genotyping using SBE/SBH assays involves the following steps: (1) Synthesizing primers complementing the genomic sequence immediately preceding SNPs of interest; (2) Hybridizing these primers with the genomic DNA; (3) Extending each primer by a single base using polymerase enzyme and dideoxynucleotides labeled with 4 different fluorescent dyes; and finally (4) Hybridizing extended primers to a universal DNA array and determining the identity of the bases that extend each primer by hybridization pattern analysis. Under the assumption of perfect hybridization, unambiguous genotyping of a set of SNPs requires selecting primers upstream of the SNPs such that each primer hybridizes to at least one array probe that hybridizes to no other primer that can be extended by a common base. Our contributions include a study of multiplexing algorithms for SBE/SBH genotyping assays and preliminary experimental results showing the achievable tradeoffs between the number of array probes and primer length on one hand and the number of SNPs that can be assayed simultaneously on the other. We prove that the problem of selecting a maximum size subset of SNPs that can be unambiguously genotyped in a single SBE/SBH assay is NP-hard, and propose efficient heuristics with good practical performance. Our heuristics take into account the freedom of selecting primers from both strands of the genomic DNA. In addition, our heuristics can enforce user-specified redundancy constraints facilitating reliable genotyping in the presence of hybridization errors. Simulation results on datasets both randomly generated and extracted from the NCBI dbSNP database suggest that the SBE/SBH architecture provides a flexible and cost-effective alternative to genotyping assays currently used in the industry, enabling genotyping of up to hundreds of thousands of user-specified SNPs per assay.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Genetic Variation between Individuals/ SNPs

After the completion of the Human Genome Project has provided us with the blueprint of the DNA present in the cells of each human [9, 10], genomics research has focused on the study of variations that occur between individuals. These variations comprise 0.1% of the genome and are factors that confer susceptibility or resistance to disease, and influence the severity or progression of disease.

Genetic variations are also important in the field of pharmacogenomics, which uses genome-wide approaches to elucidate the inherited basis of differences between persons in the response to drugs. It is estimated that genetics can account for 20 to 95 percent of variability in drug disposition and effects. There are genetic variations which have already been associated with substantial changes in the metabolism or effects of medications, and some are now being used to predict clinical response [16].

The study of genetic variations is of interest in biological anthropology as well, since it may reveal human migration patterns.

While ancestral mutations propagated over generations are an important cause for variation, much of the observed genomic diversity is caused by recombination events, whereby the chromosome transmitted to a gamete is obtained during meiosis by combining segments taken alternatively from the two parent chromosomes.

There are several types of genetic variations, also called genetic markers:

Restriction fragment length polymorphisms (RFLP) are caused by the inactivation of a particular restriction endonuclease cleavage site in a subpopulation.

Variable number of tandem repeats polymorphisms (VNTR) consist of tandem repetitions of DNA segments occurring at numerous locations throughout the human genome. Based on the number of bases in the repeated block, VNTRs are classified in **minisatellites**, which have a repeated block of 14-500 bp, and **microsatellites**, which have a repeated block of 1-13 bp.

Single-nucleotide polymorphisms (SNP) are variations in the identity of the base appearing at a particular position in the genome and represent the most frequent form of genetic variation. In general, there are only two possible nucleotides, also called *alleles*, that can appear at a SNP position. Determining the alleles present at a SNP locus in the genetic DNA of an individual is called *SNP genotyping*.

1.2 Genotyping Methods

With a number of 10.4 million human SNPs available in public databases [32], efficient, flexible and affordable genotyping technologies are needed. There are a multitude of SNP genotyping methods, combining a variety of allele discrimination techniques, reaction formats and detection mechanisms, but further improvements in their throughput, flexibility and cost are desirable. In the following we briefly summarize the existing options for the three aspects of every SNP genotyping assay, namely, the allele discrimination technique, the reaction format, and the detection strategy. For comprehensive descriptions, see [24, 44].

Allelic discrimination

There are four main mechanisms for allelic discrimination, each with its advantages and disadvantages: allele-specific hybridization, allele-specific primer extension, allele-specific oligonucleotide ligation and allele-specific invasive cleavage.

In the *hybridization* approach, two probes are designed, each containing the complement of one of the SNP variants in the middle. The design of the probes must ensure that only the probe corresponding to the allele present at the SNP locus will hybridize to the target DNA sequence and that the one-base mismatch in the other probe will prevent it from forming a stable duplex with the target sequence.

Because it does not involve any enzyme, the allele-specific hybridization method is

the simplest one for allelic discrimination. At the same time, the reaction conditions that ensure the optimal distinction between the alleles depend on the target sequence and so, they must be determined separately for every SNP that has to be genotyped. This makes the design of multiplex assays based on hybridization a difficult task. One way of achieving this is the use of microarrays that contain multiple probes for each SNP. This method is used in the GeneChip HuSNP Mapping Array produced by Affymetrix [3] and described in more detail in the next section.

The *primer extension* method is based on the high accuracy of nucleotide incorporation by the DNA polymerase enzyme, instead of on differences in the thermal stability between mismatched and perfectly matched hybrids, as the hybridization approach does. There are two types of reactions that use the primer extension principle. In the first one, called minisequencing, a primer that hybridizes with the target sequence immediately preceding the SNP site is extended by the DNA polymerase with a single nucleotide, which will be the complement of the present allele. The identity of the incorporated base can be detected by different methods, like mass spectrometry [7, 42], enzyme-mediated luminometric detection of pyrophosphate [4, 35, 41] or fluorescence [36].

The second type of reaction uses two primers which perfectly hybridize to the target sequence immediately preceding the SNP site and which contain the complements of the two SNP variants at their $3'$ end. Only the primer corresponding to the present allele will perfectly match at its $3'$ end and, therefore, will be extended by the DNA polymerase. Monitoring the primer extension event allows one to infer the allele found in the DNA sample.

The main advantages of the methods based on primer extension are their robustness and the small number of primers required.

Another enzyme assisted method for allelic discrimination is the allele-specific oligonucleotide *ligation* method. Two pairs of probes are synthesized, such that one of the probes matches the genomic sequence immediately preceding the SNP locus and contains the complement of one of the SNP variants at its $3'$ end, and the other probe matches the genomic sequence following the polymorphic site. The DNA ligase enzyme will join one such pair of probes only if they perfectly hybridize to the target sequence. Thus, by

determining whether ligation has occurred or not, one can infer the identity of the base present at the polymorphic site. Ligation is highly specific, but it is a slow reaction and demands the largest number of synthesized probes, compared to the other allelic discrimination methods.

Finally, the *invasive cleavage* approach also requires two pairs of probes. The first probe in a pair, called invader probe, perfectly matches the DNA target sequence upstream the SNP site. The other probe is an allele-specific oligonucleotide, which contains the complement of one of the variants in the middle and its $3'$ part matches the DNA sequence following the SNP site, while its $5'$ part is unrelated to the target. When the allele-specific probe hybridizes to the target, the invader probe will displace it at the polymorphic site and the formed structure is recognized by a flap endonuclease enzyme, which cleaves the $5'$ part of the allele-specific probe. The purity of the allele-specific probes must be extremely high, in order to ensure the specificity of the reaction.

Reaction formats

Based on the reaction format, there are genotyping methods that use *homogeneous reactions* or *solid phase reactions*. The homogeneous reactions are done entirely in solution. For this reason, they are robust and highly flexible, but allow only a limited amount of multiplexing. In the solid phase reactions, solid supports, like latex beads, glass slides, silicon chips or the walls of a microtiter well, are used. The solid support can contain allele-specific oligonucleotides, as is the case with the GeneChip HuSNP Mapping Array [3], or generic oligonucleotides, as is the case of tag arrays, described in more details in the next section. The major advantage of performing genotyping reactions on solid supports is that many SNPs can be interrogated at the same time, saving time and reagents. The drawback is that the design of the assays and the optimization of the multiplex reactions require substantial capital and time investment [24].

Detection mechanisms

Detection of a positive allelic discrimination reaction can be done by monitoring the light emitted by the products, by measuring the mass of the products or by detecting a change in the electrical property when the products are formed.

Monitoring light emission is the most widely used detection mechanism in genotyping

and it uses properties of light as luminescence, fluorescence, time-resolved fluorescence, fluorescence resonance energy transfer (FRET) and fluorescence polarization (FP).

Mass spectrometry (MS) needs no label, since it measures the molecular weight of the products formed. High resolution MS can easily distinguish between DNA molecules that differ by only one base, which makes MS particularly useful as a detection mechanism for genotyping methods based on primer extension. The difficulty with MS is that the primer extesion products need to be rigorously purified before measurement to avoid background from biological material present in the sample [44].

Electrical detection is a new and promising detection method, which offers sensitivity and low cost. Numerous approaches to electrochemical detection have been developed, including direct electrochemistry of DNA, electrochemistry at polymer-modified electrodes, electrochemistry of DNA-specific redox reporters, electrochemical amplifications with nanoparticles, and electrochemical devices based on DNA-mediated charge transport chemistry (see [14] for a review).

1.3 Microarray Based Methods

A DNA microarray consists of a solid surface on which DNA oligonucleotides, called probes, are chemically bonded. With current technologies, up to 10^6 probes can be immobilized on a one cm^2 surface, which makes microarrays a great tool for performing multiplex genotyping assays. In this section we describe three existing microarray based SNP genotyping techniques, namely the GeneChip mapping arrays produced by Affymetrix, which are based on hybridization, the arrayed primer extension (APEX) method, based on minisequencing, and the universal tag arrays. The latter one also uses single base primer extension but, unlike the first two, it is a generic method, i.e., it allows the genotyping of a custom set of SNPs.

GeneChip Mapping Arrays

The GeneChip HuSNP Mapping Array produced by Affymetrix [3] contains multiple allele-specific probes for each SNP to be genotyped. The probes include all possible sequences containing a stretch of nucleotides flanking the polymorphic site. After hybridization, a computer algorithm is used to interpret the complex fluorescence patterns

formed by the multiple probes and to assign the genotypes of each SNP.

The current throughput achieved by this technology is 250,000 SNPs per array [3]. The major drawback is that the set of SNPs that can be genotyped using these arrays is selected by the manufacturer. Genotyping a similar number of user-specified set of SNPs would require the complete re-design of array probes as well as a difficult re-engineering of the primer-ligation amplification protocol.

Moreover, it is a common experience that about 20% of the SNPs on the GeneChip array do not yield confident results [24].

Arrayed primer extension (APEX)

APEX [45] is a genotyping method based on minisequencing, using solid support and fluorescence detection. More specifically, one primer for each SNP to be genotyped is immobilized via its $5'$ end on a glass surface. The primer must complement the DNA sequence upstream the polymorphic site. The PCR products containing the SNP sequences, together with a DNA polymerase and the four dideoxynucleotides fluorescently labeled with four different dyes, are brought into contact with the array. The primers on the array will get extended with the Watson-Crick complements of the bases present at the SNP loci. The array will be scanned and the genotypes determined by observing the colors at the various spots on the array.

The APEX assay is quite robust and can be multiplexed. Furthermore, a universal master mix containing the four dye-labeled terminators and DNA polymerase is used for all SNPs, making it a very simple reaction to set up. The challenge is that thermal cycling is generally not easily achieved in solid phase reactions, so single-stranded templates are needed for robust primer extension. This requires a larger amount of PCR products as target and a strand separation step that increases the cost of the reaction. In addition, placing SNP-specific probes on the solid support decreases the flexibility of the approach [24].

A method based on APEX, but which uses a set of generic probes (all k-mers) on the array was recently proposed in [43].

Tag Arrays

Among technologies that allow genotyping of custom sets of SNPs the most successful

one is the use of DNA tag arrays [8, 17, 19, 26]. DNA tag arrays consist of a set of DNA strings called *tags*, which are designed such that, together with their *antitags* (their Watson-Crick complements), satisfy the following hybridization constraints:

(H1) Every antitag hybridizes strongly to its tag;

(H2) No antitag hybridizes to a tag other than its complement; and

(H3) There is no antitag-to-antitag hybridization (including hybridization between two copies of the same tag and self-hybridization).

The flexibility of tag arrays comes from combining solid-phase hybridization with the high sensitivity of single-base extension reactions. A typical assay based on tag arrays performs SNP genotyping using the following steps [6, 19]: (1) A set of *reporter probes* is synthesized by ligating antitags to the $5'$ end of primers complementing the genomic sequence immediately preceding the SNPs of interest. (2) Reporter probes are hybridized in solution with the genomic sample. (3) The hybridized $3'$ (primer) end of reporter probes is extended by a single base in a reaction using the polymerase enzyme and dideoxynucleotides fluorescently labeled with 4 different dyes. (4) Reporter probes are separated from the template DNA and hybridized to a tag array. (5) Finally, fluorescence levels are used to determine the identity of the extending dideoxynucleotides.

Commercially available tag arrays have between 2,000 and 10,000 tags (see, e.g., GenFlex and ParAllele TrueTag arrays from Affymetrix [1, 2]). The number of SNPs that can be genotyped per array is typically smaller than the number of tags since some of the tags must remain unassigned due to cross-hybridization with the primers [6, 28].

Another factor limiting the throughput of tag arrays is the high synthesis cost of reporter probes, which have a typical length of 40 nucleotides.

1.4 Contributions

In the first part of this thesis we study methods for improving the multiplexing rate (defined as the average number of reactions assayed per array) in SNP genotyping assays involving multiple universal tag arrays. In general, it is not possible to use all tags in an array experiment due to, e.g., unwanted hybridizations. An assay specific optimization

that determines the multiplexing rate (and hence the number of required arrays for a large assay) is the *tag assignment problem*, whereby individual tags are assigned to the primers used to genotype each SNP. We observe that significant improvements in multiplexing rate can be achieved by combining primer selection with tag assignment. For most universal tag array applications there are multiple primers with the desired functionality; for example in SNP genotyping one can choose the corresponding primer from either the forward or reverse strands. Since different primers hybridize to different sets of tags, a higher multiplexing rate is achieved by integrating primer selection with tag assignment. This integrated optimization is shown to lead to a reduction of up to 50% in the number of required arrays.

In the second part of the thesis, we propose a new genotyping assay architecture combining multiplexed solution-phase single-base extension (SBE) reactions with sequencing by hybridization (SBH) using universal DNA arrays such as all k-mer arrays. In addition to PCR amplification of genomic DNA, SNP genotyping using SBE/SBH assays involves the following steps: (1) Synthesizing primers complementing the genomic sequence immediately preceding SNPs of interest; (2) Hybridizing these primers with the genomic DNA; (3) Extending each primer by a single base using polymerase enzyme and dideoxynucleotides labeled with 4 different fluorescent dyes; and finally (4) Hybridizing extended primers to a universal DNA array and determining the identity of the bases that extend each primer by hybridization pattern analysis. Under the assumption of perfect hybridization, unambiguous genotyping of a set of SNPs requires selecting primers upstream of the SNPs such that each primer hybridizes to at least one array probe that hybridizes to no other primer that can be extended by a common base. Our contributions include a study of multiplexing algorithms for SBE/SBH genotyping assays and preliminary experimental results showing the achievable tradeoffs between the number of array probes and primer length on one hand and the number of SNPs that can be assayed simultaneously on the other. We prove that the problem of selecting a maximum size subset of SNPs that can be unambiguously genotyped in a single SBE/SBH assay is NP-hard, and propose efficient heuristics with good practical performance. Our heuristics take into account the freedom of selecting primers from both strands of the genomic

DNA. In addition, our heuristics can enforce user-specified redundancy constraints facilitating reliable genotyping in the presence of hybridization errors. Simulation results on datasets both randomly generated and extracted from the NCBI dbSNP database suggest that the SBE/SBH architecture provides a flexible and cost-effective alternative to genotyping assays currently used in the industry, enabling genotyping of up to hundreds of thousands of user-specified SNPs per assay.

Most of the results presented in this thesis appear in:

1. I.I. Mandoiu, C. Prajescu, and D. Trinca. Improved Tag Set Design and Multiplexing Algorithms for Universal Arrays. *LNCS Transactions on Computational Systems Biology*, volume II (LNBI 3680), pages 124–137, 2005 [28]. A preliminary version of this paper has appeared in *Proc. 5th International Conference on Computational Science (ICCS 2005)/ 2005 International Workshop on Bioinformatics Research and Applications (IWBRA)*, LNCS 3515, pages 994–1002, 2005 [29].

2. N. Hundewale, I.I. Măndoiu, C. Prăjescu, and A. Zelikovsky. Integrated design flow for universal DNA tag arrays. In *9th Annual International Conference on Research in Computational Molecular Biology (RECOMB) Poster Book*, pages 141–142, 2005 [21].

3. I.I. Măndoiu and C. Prăjescu. High-throughput SNP genotyping by SBE/SBH. ACM Computing Research Repository, cs.DS/0512052, 2005 [27].

Chapter 2

Multiplexing Algorithms for Tag Arrays

In this chapter we study methods for improving the multiplexing rate (defined as the average number of reactions assayed per array) in large-scale genomic assays involving multiple tag arrays. In general, it is not possible to assign all tags to primers in an array experiment due to unwanted cross-hybridizations. For example, if a tag is assigned to a primer p, then tags hybridizing with p can no longer be assigned to other primers. An assay specific optimization that determines the multiplexing rate (and hence the number of required arrays for a large assay) is the *tag assignment problem*, whereby individual (anti)tags are assigned to each primer.

The rest of the chapter is organized as follows. Section 2.1 formalizes the Tag Array Multiplexing Problem and presents a review of the previous work. In Section 2.2 we observe that significant improvements in multiplexing rate can be achieved by combining primer selection with tag assignment. For most tag array applications there are pools containing multiple primers with a desired functionality; for example in the SNP genotyping assay described in Section 1.3 one can choose the primer from either the forward or reverse strands. Since different primers hybridize to different sets of tags, a higher multiplexing rate is achieved by integrating primer selection with tag assignment. Section 2.3 formalizes the problem of finding a maximum assignable pool set. Results on simulated data are presented in Section 2.4 and an application to gene expression in the Herpes B virus is described in Section 2.5.

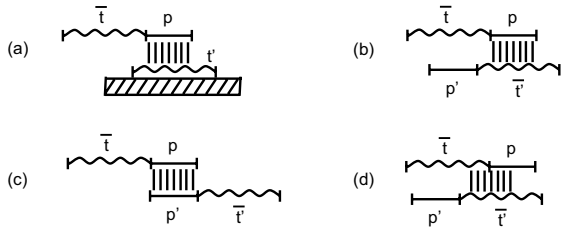


Figure 2.1: Four types of undesired hybridizations, caused by the formation of nucleation complexes between (a) a primer and a tag other than the complement of the ligated antitag, (b) a primer and an antitag, (c) two primers, and (d) two reporter probe substrings, at least one of which stradles a ligation point.

2.1 Problem Formalization and Review of Previous Work

Even when tags satisfy properties (H2)-(H3) in Section 1.3, which prevent unintended antitag-to-tag and antitag-to-antitag hybridizations, the formation of nucleation complexes involving portions of the primers may still lead to undesired hybridization between reporter probes and tags on the array (Fig. 2.1(a)), or between two reporter probes (Fig. 2.1(b)-(d)). The formation of these duplexes must be avoided as it leads to false primer extensions, extension misreporting, and/or reduced effective reporter probe concentration available for hybridization to the template DNA and to the tags on the array [6]. This can be done by assaying the primers using a sufficiently large number of arrays, and assigning antitags to primers such that the following constraints are satisfied:

(A1) If primer p forms the configuration in Fig. 2.1(a), then antitag $\bar{t}'$ is not assigned to any primer in array experiments in which p is assayed, unless it is assigned to p itself.

(A2) If primer p forms the configuration in Fig. 2.1(b), then antitag $\bar{t}'$ is not assigned to any primer in array experiments in which p is assayed (this time assigning $\bar{t}'$ to p is not allowable).

(A3) If primers p and p' form the configuration in Fig. 2.1(c), then they are assayed on different array experiments.

(A4) Antitag $\bar{t}$ is never assigned to primer p if they form the configuration in Fig. 2.1(d) with $t' = t$.

(A5) If antitag $\bar{t}$ is assigned to primer p in an array experiment, and the resulting reporter probe forms the configuration in Fig. 2.1(d) with $t' \neq t$, then antitag $\bar{t'}$ is not assigned to any primer in that experiment.

As in [6], we focus on preventing primer-to-tag hybridizations (A1). Our algorithms can be easily extended to prevent primer-to-antitag hybridizations (A2); a simple practical solution for preventing the other (less-frequent) unwanted hybridizations is to reassign offending primers in a post-processing step.

Let $\mathcal{P} = \{p_1, \ldots, p_m\}$ be a set of primers and $\mathcal{T} = \{t_1, \ldots, t_n\}$ a set of tags. Following [6], we define the *conflict graph* of $\mathcal{P}$ and $\mathcal{T}$ to be the bipartite graph whose vertices are primers $\mathcal{P}$ and tags $\mathcal{T}$ and which has an edge between primer p and tag t if they hybridize.

A set $\mathcal{P}$ of primers is called *assignable* to a set $\mathcal{T}$ of tags if there is a one-to-one mapping $a : \mathcal{P} \to \mathcal{T}$ such that, for every tag t hybridizing to a primer $p \in \mathcal{P}$, either $t \notin a(\mathcal{P})$ or $t = a(p)$. An assignable set of primers induces a balanced subgraph of maximum degree 1 in the conflict graph [6].

Tag Array Multiplexing Problem: *Given primers $\mathcal{P} = \{p_1, \ldots, p_m\}$ and tag set $\mathcal{T} = \{t_1, \ldots, t_n\}$, find a partition of $\mathcal{P}$ into the minimum number of assignable sets.*

The problem is shown to be NP-complete by reduction from Set Cover in [6]. Ben-Dor et al. also observe in [6] that the problem remains NP-complete even if the number of tags is required to be greater than or equal to the number of primers.

Defining X as the number of primers p for which there exists a tag t hybridizing with p and with no other primer and Y as the number of tags that do not hybridize with any primer, Ben-Dor et al. have proved in [6] the following proposition:

Proposition 1 ([6]) *A set of primers $\mathcal{P}$ is assignable if and only if*

$$X + Y \geq |\mathcal{P}|.$$

Proof. If $\mathcal{P}$ is assignable then there is a one-to-one mapping $a : \mathcal{P} \to \mathcal{T}$ such that, for every primer $p \in \mathcal{P}$, the tag $t = a(p)$ either does not hybridize to any primer (it contributes a value of 1 to Y), or it hybridizes to p and to no other primer (it contributes a value of 1 to X). This implies that $X + Y \geq |\mathcal{P}|$. Conversely, if $X + Y \geq |\mathcal{P}|$ then

`Input:` Primer set $\mathcal{P} = \{p_1, \ldots, p_m\}$ and tag set $\mathcal{T}$
`Output:` A partition ε of $\mathcal{P}$ into assignable subsets of primers

---

1. $\varepsilon \leftarrow \emptyset$
2. $P' \leftarrow P$
3. `While` P' is not assignable `do`
   Remove a primer of maximum potential from P'
   `End While`
4. $\varepsilon \leftarrow \varepsilon \cup \{P'\}$ (add P' to the cover)
5. Update $P \leftarrow P \backslash P'$
6. `If` $P = \emptyset$ `then` halt `else` go to 2

Figure 2.2: Algorithm B in [6].

there are more than $|\mathcal{P}|$ tags that hybridize to at most one primer, i.e., enough tags to assign to every primer. □

Using Proposition 1 to test whether a set of primers is assignable or not, [6] presents a simple heuristic algorithm for solving the Univeral Array Multiplexing Problem (see Figure 2.2). In each iteration, the algorithm checks whether the remaining set of primers P' is assignable. If not, it removes a primer that has the greatest potential of creating useful tags by its removal. Each tag t is assigned a potential of becoming useful, equal to 2^{-w}, where w is the number of primers that cross-hybridize with t (the greater w is, the lower is t's potential of becoming useful).

Ben-Dor et al. also formulate in [6] a simple stochastic model for the cross-hybridization matrix A, as follows:. Let $n(A)$ denote the minimum number of assignable primer sets and let $D(m, n, p)$ be a probability distribution of $m \times n$ matrices, where each matrix entry independently is equal to 1 with probability p and to 0 with probability $1 - p$. Using Chernoff bound, Ben-Dor et al. prove a lower bound for $n(A)$ for matrices drawn from $D(m, n, p)$:

Theorem 1 ([6]) *Let matrix A be drawn from a probability distribution $D(m, n, p)$. For a positive integer t, define $h = \lceil \frac{m}{t} \rceil$ and $x = n(1-p)^{h-1}(1-p+hp)$. Then the following bound holds for all t such that $h > x$:*

$$Prob[n(a) \leq t] \leq \frac{t^m}{t!} \left(\frac{x e^{\frac{h-x}{h}}}{h} \right)^{ht}.$$

2.2 Improved Multiplexing by Integrated Primer Selection and Tag Assignment

For most tag array applications there are multiple primers with the desired functionality, e.g., for the SNP genotyping assay described in Section 1.3 one can choose the primer from either the forward or reverse strands. Since different primers have different hybridization patterns, a higher multiplexing rate can in general be achieved by integrating primer selection with tag assignment.

A similar integration has been recently proposed in [22] between probe selection and physical DNA array design, with the objective of minimizing unintended illumination in photo-lithographic manufacturing of DNA arrays. The idea in [22] is to modify probe selection tools to return *pools* containing all feasible candidates, and let subsequent optimization steps select the candidate to be used from each pool. We use a similar approach. We say that a set of primer pools is *assignable* if we can select a primer from each pool to form an assignable set of primers.

Pooled Tag Array Multiplexing Problem: *Given primer pools* $\mathcal{P} = \{P_1, \ldots, P_m\}$ *and tag set* $\mathcal{T} = \{t_1, \ldots, t_n\}$*, find a partition of* $\mathcal{P}$ *into the minimum number of assignable sets.*

Theorem 2 *The Pooled Tag Array Multiplexing Problem is NP-complete.*

Proof. Ben-Dor et al. have proved in [6] that the problem is NP-complete for the case in which every pool has size 1 by reduction from the well-known Set Cover Problem. The NP-completeness for the general case follows from that. □

Let $\mathcal{P}$ be a set of primer pools and $\mathcal{T}$ a tag set. For a primer p (tag t), $\mathcal{T}(p)$ (resp. $\mathcal{P}(t)$) denotes the set of tags (resp. primers of $\bigcup_{P\in\mathcal{P}} P$) hybridizing with p (resp. t). Let $X(\mathcal{P}) = \{P \in \mathcal{P} \ : \ \exists p \in P,\ t \in \mathcal{T}$ s.t. $t \in \mathcal{T}(p)$ and $\mathcal{P}(t) \subseteq P\}$ and $Y(\mathcal{P}) = \{t \in \mathcal{T} \ : \ \mathcal{P}(t) = \emptyset\}$. Clearly, in every pool of $X(\mathcal{P})$ we can find a primer p that hybridizes to a tag t which is not cross-hybridizing to primers in other pools, and therefore assigning t to p will not violate (A1). Furthermore, any primer can be assigned to a tag in $Y(\mathcal{P})$ without violating (A1). Thus, a set $\mathcal{P}$ with $|X(\mathcal{P})| + |Y(\mathcal{P})| \geq |\mathcal{P}|$ is always assignable. The converse, and hence the corresponding Proposition 1 for the

Pooled Tag Array Multiplexing Problem, is not necessarily true: Figure 2.3 shows two pools that are assignable although $|X(\mathcal{P})| + |Y(\mathcal{P})| = 0$.

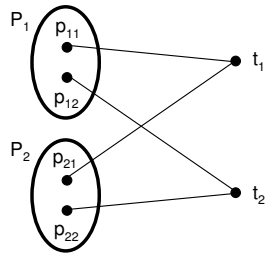


Figure 2.3: Two assignable pools for which $|X(\mathcal{P})| + |Y(\mathcal{P})| = 0$.

Our primer pool assignment algorithm (Primer-Del) (see Figure 2.4) is a generalization to primer pools of Algorithm B in [6]. In each iteration, the algorithm checks whether $|X(\mathcal{P}')| + |Y(\mathcal{P}')| \geq |\mathcal{P}'|$ for the remaining set of pools $\mathcal{P}'$. If not, a primer of maximum *potential* is deleted from the pools. As in [6], the potential of a tag t with respect to $\mathcal{P}'$ is $2^{-|\mathcal{P}'(t)|}$, and the potential of a primer p is the sum of potentials for the tags in $\mathcal{T}(p)$. If the algorithm deletes the last primer in a pool P, then P is itself deleted from $\mathcal{P}'$; deleted pools are subsequently assigned to new arrays using the same algorithm.

Figure 2.5 presents a variant of the algorithm described above, called Primer-Del+, which is shown to give better results in Section 2.4. In Primer-Del+, primers in pools of size 1 are omitted – unless all pools have size 1 – when selecting the primer with maximum potential for deletion.

We have also considered two simple heuristics (Min-Pot - Figure 2.6 and Min-Deg - Figure 2.7) that first select from each pool the primer of minimum potential, respectively minimum degree, and then run the iterative primer deletion algorithm on the resulting pools of size 1.

2.3 Maximum Assignable Pool Set Problem

Note that all algorithms presented above work by iteratively finding assignable pools sets. This justifies formalizing the problem of finding a maximum assignable pool set.

Maximum Assignable Pool Set Problem (MAP): *Given primer pools* $\mathcal{P} = \{P_1, \ldots, P_m\}$

```
Input: Primer pools P = {P_1, ..., P_m} and tag set T
Output: Triples (p_i, t_i, k_i), 1 ≤ i ≤ m, where p_i ∈ P_i is the selected primer for pool i,
t_i is the tag assigned to p_i, and k_i is the index of the array on which p_i is assayed
---------------------------------------------------------------------------
k ← 0
While P ≠ ∅ do
      k ← k + 1;  P' ← P
      While |X(P')| + |Y(P')| < |P'| do
         Remove the primer p of maximum potential from the pools in P'
         If p's pool becomes empty then remove it from P'
      End While
      Assign pools in P' to tags on array k
      P ← P \ P'
End While
```

Figure 2.4: The iterative primer deletion algorithm.

and tag set $\mathcal{T} = \{t_1, \ldots, t_n\}$, find a maximum assignable subset of $\mathcal{P}$.

Theorem 3 *The Maximum Assignable Pool Set Problem is NP-hard.*

Proof. Ben-Dor et al. have proved in [6] the NP-hardness for the case in which every pool has size 1 by reduction from the complete balanced bipartite subgraph problem. The NP-hardness for the general case follows from that. □

Ben-Dor et al. also note in [6] that MAP can be solved in polynomial time in the case that every pool has size 1 and every primer has degree at most 1. Figure 2.8 presents a polynomial algorithm that solves MAP for this case.

We will denote with p_0 the number of primers of degree 0 and with t_0 the number of tags of degree 0.

Lemma 1 ([6]) *The MAP algorithm described in Figure 2.8 gives a maximum assignment when the pool size is 1 and each primer has degree at most 1.*

Proof. In order to prove the optimality of our algorithm, we will first state and prove several claims.

Claim 1: By condition (A1) in Section 2.1, if a tag t with degree greater than 0 is chosen in an assay, then at most one of its neighbor primers can be assigned in that assay and it has to be assigned to t.

```
Input: Primer pools P = {P_1, ..., P_m} and tag set T
Output: Triples (p_i, t_i, k_i), 1 ≤ i ≤ m, where p_i ∈ P_i is the selected primer for pool i,
t_i is the tag assigned to p_i, and k_i is the index of the array on which p_i is assayed
---------------------------------------------------------------------------
k ← 0
While P ≠ ∅ do
      k ← k + 1;  P' ← P
      While |X(P')| + |Y(P')| < |P'| do
         P+ = {P ∈ P' s.t. |P| > 1}
         If |P+| > 0
         Then remove the primer p of maximum potential from the pools in P+
         Else
            Remove the primer p of maximum potential from the pools in P'
            Remove p's pool from P'
         End If
      End While
      Assign pools in P' to tags on array k
      P ← P \ P'
End While
```

Figure 2.5: Primer-Del+ algorithm.

Claim 2: There is an optimum solution to MAP in which every tag t of degree greater than 0 is assigned to one of its neighbor primers or not assigned at all. Otherwise, consider an optimum solution in which a tag t of degree greater than 0 is assigned to a primer p different from all its neighbor primers, which are left unassigned. Then, by leaving p unassigned and assigning t to one of its neighbor primers, an optimum solution is obtained.

Claim 3: There is an optimum solution to MAP that uses every tag of degree 0. Otherwise, consider an optimum assignment in which a tag t of degree 0 is left unassigned. Then, replacing any assigned tag with t does not create any conflicts, thus the newly constructed assignment forms an optimum solution.

Claim 4: There is an optimum solution to MAP which assigns $\min(p_0, t_0)$ primers of degree 0 to tags of degree 0. Otherwise, consider a maximum assignment in which a primer p of degree 0 is left unassigned and a tag t of degree 0 is assigned to a primer $p' \neq p$ of degree 1. Then, by assigning p to t and leaving p' unassigned, one gets an optimum solution.

If $p_0 \geq t_0$, then, from Claims 3 and 4, there is an optimum solution in which all the

`Input:` Primer pools $\mathcal{P} = \{P_1, \ldots, P_m\}$ and tag set $\mathcal{T}$
`Output:` Triples (p_i, t_i, k_i), $1 \leq i \leq m$, where $p_i \in P_i$ is the selected primer for pool i, t_i is the tag assigned to p_i, and k_i is the index of the array on which p_i is assayed

---

$P' \leftarrow \emptyset$
`For` all pools $P \in \mathcal{P}$ `do`
    Find a primer $p \in P$ of minimum potential
    $P' \leftarrow P' \cup \{p\}$
`End For`
Run Primer-Del on P'

Figure 2.6: Min-Pot algorithm.

`Input:` Primer pools $\mathcal{P} = \{P_1, \ldots, P_m\}$ and tag set $\mathcal{T}$
`Output:` Triples (p_i, t_i, k_i), $1 \leq i \leq m$, where $p_i \in P_i$ is the selected primer for pool i, t_i is the tag assigned to p_i, and k_i is the index of the array on which p_i is assayed

---

$P' \leftarrow \emptyset$
`For` all pools $P \in \mathcal{P}$ `do`
    Find a primer $p \in P$ of minimum degree
    $P' \leftarrow P' \cup \{p\}$
`End For`
Run Primer-Del on P'

Figure 2.7: Min-Deg algorithm.

tags of degree 0 are assigned to primers of degree 0. Since every tag of degree greater than 0 can be assigned to one of its neighbor primers, the size of the optimum solution in this case will be the total number of tags, $OPT = n$.

If $p_0 < t_0$, then, from Claims 3 and 4, there is an optimum solution in which p_0 tags of degree 0 are assigned to the degree 0 primers and the remaining $t_0 - p_0$ tags of degree 0 are assigned to primers of degree 1, thus leaving unassigned a number k of tags of degree greater than 0, $t_1, \ldots, t_k$. Then the following inequalities hold:

$$\sum_{i=1}^{k-1} \deg(t_i) < t_0 - p_0 \leq \sum_{i=1}^{k} \deg(t_i).$$

From this and from Claim 2, it follows that the size of the optimum solution in this case is $OPT = n - k$. Since OPT is maximized, i.e., k is minimized, the k tags that are left unassigned in an optimum solution are the k highest degree tags in $\mathcal{T}$.

This proves that the MAP algorithm constructs a maximum assignment. □

The algorithm described above can be extended to solve MAP when each *pool* P has

```
Input: Primers P = {p_1, ..., p_m} and tags T = {t_1, ..., t_n}
Output: P' ⊆ P a maximum assignable set of primers and assigments (p, t), for
every p ∈ P'
```

```
If p_0 ≥ t_0
Then
      Assign all tags of degree 0 to primers of degree 0
      Assign each tag of degree greater than 0 to an arbitrary neighbor primer
Else
      Assign p_0 tags of degree 0 to the p_0 primers of degree 0
      While there are unassigned tags of degree 0 do
         Remove t, an unassigned tag of maximum degree
         Assign as many as possible of t's neighbor primers to tags of degree 0
      End While
      Assign each left tag of degree greater than 0 to an arbitrary neighbor primer
End If
```

Figure 2.8: MAP algorithm.

degree at most 1, i.e., primers in P hybridize to at most one tag.

Lemma 2 *MAP can be solved in polynomial time when each pool has degree at most 1.*

Proof. Let $\mathcal{P}$ be a set of pools. Note that all primers of degree 1 in pool P are equivalent with regard to the generated conflicts, since they hybridize to the same tag, and, obviously, all primers of degree 0 in P are also equivalent. We will choose the *representative* of a pool P to be an arbitrary primer of degree 0 from P, if it exists, otherwise an arbitrary primer of degree 1 from P.

Let R be the set of representative primers for all pools in $\mathcal{P}$. We claim that the size of a maximum assignable subset of pools in $\mathcal{P}$ (denoted by $OPT(\mathcal{P})$) is equal to the size of a maximum assignable subset of primers in R (denoted by $OPT(R)$).

A maximum assignable subset of representatives $R' \subseteq R$ induces an assignable subset of pools (for each representative primer in R', its corresponding pool is assignable). So $OPT(\mathcal{P}) \geq OPT(R)$. Moreover, let $\mathcal{P}' \subseteq \mathcal{P}$ be a maximum assignable subset of pools and let R' be the set of the representative primers of pools in $\mathcal{P}'$. For every pool $P \in \mathcal{P}'$, let $p \in P$ be the chosen primer from P and t the tag assigned to p. If r, the representative primer of P, has degree 0, then r can be assigned to t. If r has degree 1, then it can also

be assigned to t, because r and p hybridize to the same tag. This means that R' is an assignable subset of R, so $OPT(\mathcal{P}) \leq OPT(R)$.

Thus, applying the MAP algorithm in Figure 2.8 on the set of representatives of each pool will give a maximum assignable pool set. □

In the case where each primer has degree at most 1, we do not know whether or not MAP is NP-hard, but we can decide if the size of a maximum assignable pool set is the number of tags using the maximum matching problem.

In the following we present an integer programming formulation for the Maximum Assignable Pool Set Problem. For every primer $p \in \bigcup_{P \in \mathcal{P}} P$ and every tag $t \in \mathcal{T}$, consider a binary variable z_{pt}. $z_{pt} = 1$ if and only if t is assigned to p.

$$\text{Maximize} \quad \sum_{p \in \bigcup P} \sum_{t \in \mathcal{T}} z_{pt} \tag{2.1}$$

Subject to

$$\sum_{p \in P_i} \sum_{t \in \mathcal{T}} z_{pt} \leq 1, \qquad 1 \leq i \leq m \tag{2.2}$$

$$\sum_{p:(p,t) \notin E} z_{pt} + \sum_{p:(p,t) \in E} \sum_{t' \in \mathcal{T}} z_{pt'} \leq 1, \qquad \forall t \in \mathcal{T} \tag{2.3}$$

$$z_{pt} \in \{0,1\}, \qquad \forall p \in \bigcup_{P \in \mathcal{P}} P, \forall t \in \mathcal{T} \tag{2.4}$$

Lemma 3 *The above integer program solves the Maximum Assignable Pool Set Problem.*

Proof. We have to show that our ILP formulation enforces condition (A1), i.e., the primers for which there is a tag t with $z_{pt} = 1$ form an assignable set. Conversely, suppose that there is a primer p and a tag t that cross-hybridize and primer p is assigned a tag $t' \neq t$ and tag t is assigned to a primer $p' \neq p$. Then $z_{pt'} = 1$ and $z_{p't} = 1$, which is not consistent with constraint (2.3), so we have a contradiction. □

The integer program described above can be used to solve the Pooled Tag Array Multiplexing Problem by iteratively finding maximum assignable pool sets (see Figure 2.9). While this method does not guarantee an optimum solution for the Multiplexing Problem, the experiments presented in Section 2.4 show that it performs slightly better than the heuristics presented in Section 2.2.

Input: Primer pools $\mathcal{P} = \{P_1, \ldots, P_m\}$ and tag set $\mathcal{T}$
Output: Triples (p_i, t_i, k_i), $1 \leq i \leq m$, where $p_i \in P_i$ is the selected primer for pool i, t_i is the tag assigned to p_i, and k_i is the index of the array on which p_i is assayed

---

$k \leftarrow 0$
While $\mathcal{P} \neq \emptyset$ do
    $k \leftarrow k + 1$
    Find the maximum assignable subset $\mathcal{P}' \subseteq \mathcal{P}$ using ILP
    Assign pools in $\mathcal{P}'$ to tags on array k
    $\mathcal{P} \leftarrow \mathcal{P} \setminus \mathcal{P}'$
End While

Figure 2.9: The iterative ILP algorithm.

2.4 Results on Simulated Data

We ran all four algorithms proposed in Section 2.2 on data sets with between 1000 to 5000 pools of up to 5 randomly generated primers. As in [6], we varied the number of tags between 500 and 2000.

For instance size, we report the number of arrays and the average tag utilization (computed over all arrays except the last) obtained by (a) algorithm B in [6] run using a single primer per pool, (b) the four pool-aware assignment algorithms run with 1 additional candidate in each pool, and (c) the four pool-aware assignment algorithms run with 4 additional candidates in each pool. Scenario (b) models SNP genotyping applications in which the primer can be selected from both strands of the template DNA, while scenario (c) models applications such as gene transcription monitoring, where significantly more than 2 gene specific primers are typically available.

We extracted tag sequences from the tag set of the commercially available GenFlex Tag Arrays. All GenFlex tags have length 20; primers used in our experiments are 20 bases long as well. Primer-to-tag hybridizations were assumed to occur between primers and tags containing complementary c-tokens with $c = 11$ (Table 2.1), respectively $c = 12$ (Table 2.2). The results show that significant improvements in multiplexing rate – and a corresponding reduction in the number of arrays – are achieved by the pool-aware algorithms over the Algorithm B in [6]. For example, assaying 5000 reactions on a 1000-tag array requires 13 arrays using the method in [6] for $c = 11$, compared to only 10 (respectively 8) if 2 (respectively 5) primers per pool are available. In these experiments,

the Primer-Del+ algorithm dominates in solution quality the Primer-Del, while Min-Deg dominates Min-Pot. Neither Primer-Del+ nor Min-Deg consistently outperforms the other over the whole range of parameters, which suggests that a good practical meta-heuristic is to run both of them and pick the best solution obtained.

Table 2.1: Multiplexing results for $c = 11$ (averages over 10 test cases).

# pools	Pool size	Algorithm	500 tags		1000 tags		2000 tags	
			#arrays	% Util.	#arrays	% Util.	#arrays	% Util.
1000	1	Algorithm B	4.6	52.6	3.0	41.5	3.0	24.3
	2	Primer-Del	4.0	61.9	3.0	44.1	3.0	24.6
	2	Primer-Del+	4.0	63.9	3.0	47.2	2.0	37.2
	2	Min-Pot	4.0	62.9	3.0	46.1	2.0	36.5
	2	Min-Deg	4.0	64.4	3.0	47.3	2.0	37.9
	5	Primer-Del	3.5	75.8	3.0	48.8	2.2	36.4
	5	Primer-Del+	3.0	86.2	2.0	72.2	2.0	44.4
	5	Min-Pot	3.0	85.7	2.0	71.0	2.0	42.4
	5	Min-Deg	3.0	84.2	2.0	75.8	2.0	46.3
2000	1	Algorithm B	8.0	55.3	6.0	38.8	4.4	27.5
	2	Primer-Del	7.0	65.3	5.0	47.0	4.0	31.5
	2	Primer-Del+	7.0	66.2	5.0	48.2	4.0	32.9
	2	Min-Pot	7.0	64.7	5.0	46.8	4.0	31.9
	2	Min-Deg	7.0	65.9	5.0	48.6	4.0	32.8
	5	Primer-Del	6.0	79.2	4.1	61.2	3.6	37.6
	5	Primer-Del+	5.0	88.0	4.0	64.0	3.0	47.0
	5	Min-Pot	6.0	78.9	4.0	61.4	3.0	44.8
	5	Min-Deg	6.0	79.6	4.0	64.8	3.0	48.1
5000	1	Algorithm B	17.6	59.7	13.0	41.7	9.0	30.1
	2	Primer-Del	15.0	71.0	11.0	49.9	8.0	35.2
	2	Primer-Del+	15.0	71.1	10.0	53.3	8.0	35.5
	2	Min-Pot	15.2	69.0	11.0	49.0	8.0	34.3
	2	Min-Deg	15.0	70.9	10.1	53.0	8.0	35.5
	5	Primer-Del	12.6	85.4	8.2	68.0	6.0	48.4
	5	Primer-Del+	12.0	88.2	8.0	69.5	6.0	49.0
	5	Min-Pot	14.0	76.3	9.0	60.7	6.9	42.1
	5	Min-Deg	12.1	86.5	8.0	69.3	6.0	49.4

We have also implemented the iterative ILP algorithm described in Figure 2.9 and compared it with the other algorithms. The data sets contained between 200 and 1000 pools with up to 5 randomly generated primers of length 20. For these experiments we used 100 GenFlex tags and considered a 11-token hybridization model. The results in Tables 2.3 and 2.4 show that the Iterative ILP algorithm slightly outperforms the algorithms described in Section 2.2.

Table 2.2: Multiplexing results for $c = 12$ (averages over 10 test cases).

# pools	Pool size	Algorithm	500 tags #arrays	500 tags % Util.	1000 tags #arrays	1000 tags % Util.	2000 tags #arrays	2000 tags % Util.
1000	1	Algorithm B	3.0	80.4	2.0	69.9	2.0	43.4
	2	Primer-Del	3.0	86.0	2.0	71.9	2.0	45.1
	2	Primer-Del+	3.0	89.1	2.0	79.6	2.0	49.5
	2	Min-Pot	3.0	89.3	2.0	79.1	2.0	48.7
	2	Min-Deg	3.0	87.9	2.0	80.9	2.0	49.3
	5	Primer-Del	3.0	95.5	2.0	85.5	2.0	47.7
	5	Primer-Del+	3.0	97.8	2.0	91.6	1.0	50.0
	5	Min-Pot	3.0	97.9	2.0	91.0	1.0	50.0
	5	Min-Deg	3.0	89.9	2.0	90.7	1.0	50.0
2000	1	Algorithm B	6.0	75.8	4.0	61.2	3.0	45.9
	2	Primer-Del	5.9	80.9	4.0	65.3	3.0	48.0
	2	Primer-Del+	5.0	89.5	3.0	79.3	2.0	68.2
	2	Min-Pot	5.0	88.8	3.3	75.5	2.3	62.4
	2	Min-Deg	5.0	87.5	3.0	80.1	2.0	69.9
	5	Primer-Del	5.0	96.6	3.0	89.4	2.9	52.4
	5	Primer-Del+	5.0	97.6	3.0	91.4	2.0	80.9
	5	Min-Pot	5.0	97.7	3.0	91.5	2.0	79.9
	5	Min-Deg	5.0	90.0	3.0	87.0	2.0	83.7
5000	1	Algorithm B	13.4	78.8	9.0	61.7	6.0	47.4
	2	Primer-Del	12.0	89.0	8.0	71.2	5.3	55.6
	2	Primer-Del+	12.0	89.5	7.0	77.9	5.0	59.0
	2	Min-Pot	12.0	88.5	8.0	71.1	5.0	57.5
	2	Min-Deg	12.0	88.3	7.1	76.5	5.0	59.2
	5	Primer-Del	11.0	97.9	7.0	83.3	4.2	74.2
	5	Primer-Del+	11.0	98.0	6.0	92.9	4.0	78.5
	5	Min-Pot	11.0	97.6	6.0	91.9	4.0	76.2
	5	Min-Deg	11.0	93.9	6.0	88.8	4.0	77.7

2.5 Application to Gene Expression in the Herpes B Virus

In this section we describe an integrated design flow for genomic assays based on universal tag arrays and we use it to design assays for measuring Herpes B viral gene expression in cells derived from macaque and human hosts. After defining a "B virus molecular signature", the assay can provide a sensitive tool for early B virus infection diagnosis and differentiation between B herpes and closely related herpes simplex viruses.

The steps of the proposed design flow are given in Figure 2.10. With small modifications, the flow is appropriate for a wide range of genomic analyses, including gene expression, single nucleotide polymorphism (SNP) genotyping, and micro-organism identification via string barcoding [11]. Below we detail the necessary steps and present experimental results for designing a universal tag array-based assay for studying gene expression in the Herpes B virus.

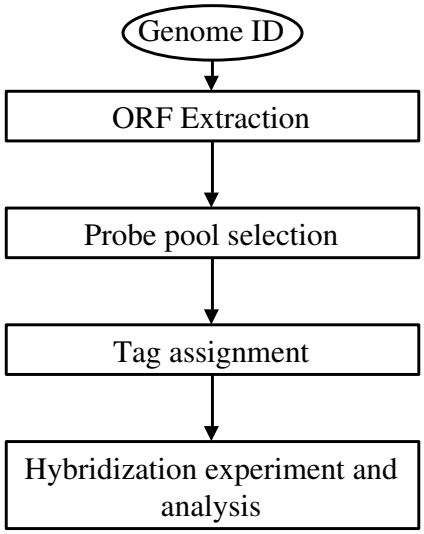


Figure 2.10: DNA Universal Tag Array Design Flow

Reading Genomic Data and Open Reading Frame (ORF) Extraction: ORFs are regions of genetic material beginning with a start codon and ending with a stop codon that might code for a protein. We use ORF extraction programs to extract the set of ORFs relevant to the application. Because of the differences between prokaryotic and eukaryotic transcription systems there are two types of ORFs. There are two approaches to accomplish this. The first one is ORF-Finder [34]. ORFs can be extracted by means of the genome's sequence or id using ORF Finder. It uses the prokaryotic approach, there are limitations in gene prediction using ORF finder. A second approach is to use GenMark [33], which provides identification of protein coding, uses both prokaryotic and eukaryotic; and it has several functions. It uses statistical methods to indicate the true beginning of the ORF and mean coding range of the ORF. GenMark extracts very specific ORFs compared to that of ORF finder.

Probe Pool Selection: The probe pool selection step is responsible for implementing the desired functionality of the DNA array. We use the Promide [40] algorithm to select a large number of possibly overlapping oligonucleotide probes (25-mers in our experiments) from every extracted ORF.[1] Promide uses a suffix array with additional

[1]The probe pool selection step is application dependent, e.g., probe candidates are chosen from the whole genomic sequence in string barcoding applications [11], or immediately preceding the target SNP on the sense or antisense strands in SNP genotyping.

information to rank all candidate oligos according to their hybridization specificity. It also introduce the concept of master sequences to describe the sequences from which oligos are to be selected. Constraints such as oligo length, melting temperature, and self-complementarily are incorporated in the master sequence at a preprocessing stage and thus kept separate from the main selection problem.

Tag Assignment: For the Tag Assignment step we integrated in our flow both Algorithm B from [6] (which is using only one candidate from each pool) and the pool-aware Primer-Del+ algorithm presented in Section 2.2.

We used our flow to design an assay for studying the expression of 78 genes in the Herpes B virus. We varied the prescribed temperature for the hybridization experiment between 60 and 70 degrees Celcius, and selected approximately 20 probes per gene in each case. In a first set of experiments we used 500, 1000, respectively 2000 tags of length 20 from the GenFlex Tag Array (see Table 2.5). We compared the results obtained in these experiments with the ones obtained when using the same number of so-called periodic tags generated using the Cycle Packing Algorithm in [30] (Table 2.6). The two tables report the number of arrays and the multiplexing rate (defined as average tag utilization computed over all arrays except the last) for both Algorithm B and Primer-Del+ algorithm. We observe that the results are significantly better when using periodic tags. This happens because the periodic tags, by construction, contain fewer c-tokens than the tags on the GenFlex Tag Array, so they generate a sparser conflict graph.

Table 2.3: Multiplexing results for 200–500 pools with $c = 11$.

# pools	Pool size	Algorithm	100 tags #arrays	100 tags % Util.
200	1	Iterative-ILP	3.0	91.0
	1	Algorithm B	3.0	89.0
	2	Iterative-ILP	3.0	98.5
	2	Primer-Del	3.0	94.0
	2	Primer-Del+	3.0	96.0
	2	Min-Pot	3.0	95.0
	2	Min-Deg	3.0	93.0
	5	Iterative-ILP	2.0	100.0
	5	Primer-Del	3.0	99.5
	5	Primer-Del+	2.0	100.0
	5	Min-Pot	2.0	100.0
	5	Min-Deg	3.0	93.5
300	1	Iterative-ILP	4.0	92.7
	1	Algorithm B	4.0	86.7
	2	Iterative-ILP	4.0	98.0
	2	Primer-Del	4.0	92.7
	2	Primer-Del+	4.0	94.0
	2	Min-Pot	4.0	94.7
	2	Min-Deg	4.0	90.7
	5	Iterative-ILP	3.0	100.0
	5	Primer-Del	4.0	98.7
	5	Primer-Del+	4.0	99.7
	5	Min-Pot	4.0	99.0
	5	Min-Deg	4.0	91.3
400	1	Iterative-ILP	5.0	92.8
	1	Algorithm B	5.0	87.2
	2	Iterative-ILP	5.0	98.8
	2	Primer-Del	5.0	95.0
	2	Primer-Del+	5.0	95.8
	2	Min-Pot	5.0	96.2
	2	Min-Deg	5.0	91.8
	5	Iterative-ILP	4.0	100.0
	5	Primer-Del	5.0	99.0
	5	Primer-Del+	5.0	99.8
	5	Min-Pot	5.0	99.8
	5	Min-Deg	5.0	92.2
500	1	Iterative-ILP	6.0	93.0
	1	Algorithm B	7.0	82.5
	2	Iterative-ILP	6.0	98.4
	2	Primer-Del	6.0	94.0
	2	Primer-Del+	6.0	94.4
	2	Min-Pot	6.0	94.2
	2	Min-Deg	6.0	91.2
	5	Iterative-ILP	5.0	100.0
	5	Primer-Del	6.0	99.2
	5	Primer-Del+	6.0	99.6
	5	Min-Pot	6.0	99.8
	5	Min-Deg	6.0	92.0

Table 2.4: Multiplexing results for 600–1000 pools with $c = 11$.

# pools	Pool size	Algorithm	100 tags #arrays	100 tags % Util.
600	1	Iterative-ILP	7.0	94.0
	1	Algorithm B	8.0	84.3
	2	Iterative-ILP	7.0	98.3
	2	Primer-Del	7.0	94.3
	2	Primer-Del+	7.0	94.8
	2	Min-Pot	7.0	94.7
	2	Min-Deg	7.0	91.7
	5	Iterative-ILP	6.0	100.0
	5	Primer-Del	7.0	99.3
	5	Primer-Del+	6.0	100.0
	5	Min-Pot	7.0	99.8
	5	Min-Deg	7.0	92.5
700	1	Iterative-ILP	8.0	94.0
	1	Algorithm B	9.0	86.0
	2	Iterative-ILP	8.0	98.6
	2	Primer-Del	8.0	94.6
	2	Primer-Del+	8.0	94.9
	2	Min-Pot	8.0	94.4
	2	Min-Deg	8.0	92.4
800	1	Iterative-ILP	9.0	94.2
	1	Algorithm B	10.0	86.4
	2	Iterative-ILP	9.0	98.5
	2	Primer-Del	9.0	95.0
	2	Primer-Del+	9.0	94.8
	2	Min-Pot	9.0	94.6
	2	Min-Deg	9.0	92.8
900	1	Iterative-ILP	11.0	89.9
	1	Algorithm B	11.0	86.6
	2	Iterative-ILP	10.0	98.7
	2	Primer-Del	10.0	95.0
	2	Primer-Del+	10.0	94.9
	2	Min-Pot	10.0	94.4
	2	Min-Deg	10.0	92.8
1000	1	Iterative-ILP	12.0	90.8
	1	Algorithm B	12.0	87.7
	2	Iterative-ILP	11.0	98.7
	2	Primer-Del	11.0	95.5
	2	Primer-Del+	11.0	95.3
	2	Min-Pot	11.0	94.9
	2	Min-Deg	11.0	93.3

Table 2.5: Results for GenFlex tags

T_m	# pools	Pool size	Algorithm	500 tags		1000 tags		2000 tags	
				#arrays	% Util.	#arrays	% Util.	#arrays	% Util.
60	1446	1	Algorithm B	5	68.50	3	56.45	2	46.25
60	1446	5	Primer-Del+	5	71.85	3	60.20	2	49.75
67	1560	1	Algorithm B	6	62.20	4	48.90	3	37.52
67	1560	5	Primer-Del+	5	72.25	4	51.90	2	49.15
70	1522	1	Algorithm B	5	70.45	3	58.80	2	49.50
70	1522	5	Primer-Del+	5	75.15	3	63.15	2	54.25

Table 2.6: Results for periodic tags

T_m	# pools	Pool size	Algorithm	500 tags		1000 tags		2000 tags	
				#arrays	% Util.	#arrays	% Util.	#arrays	% Util.
60	1446	1	Algorithm B	4	91.07	2	92.00	1	72.30
60	1446	5	Primer-Del+	4	93.40	2	95.30	1	72.30
67	1560	1	Algorithm B	4	93.40	2	95.20	1	78.00
67	1560	5	Primer-Del+	4	95.80	2	97.20	1	78.00
70	1522	1	Algorithm B	4	94.13	2	96.40	1	76.10
70	1522	5	Primer-Del+	4	95.80	2	97.80	1	76.10

Chapter 3

High-Throughput SNP Genotyping by SBE/SBH

3.1 Introduction

In the k-mer array format [13], all 4^k DNA probes of length k are spotted or synthesized on the solid array substrate (values of k of up to 10 are feasible with current high-density in-situ synthesis technologies). This format was originally proposed for performing *sequencing by hybridization (SBH)*, which seeks to reconstruct an unknown DNA sequence based on its k-mer spectrum [38]. However, the sequence length for which unambiguous reconstruction is possible with high probability is surprisingly small [39], and, despite several suggestions for improvement, such as the use of gapped probes [18] and pooling of target sequences [20], the SBH scheme has not become practical so far.

In this chapter we propose a new genotyping assay architecture combining multiplexed solution-phase single-base extension (SBE) reactions with sequencing by hybridization (SBH) using universal DNA arrays such as all k-mer arrays. SNP genotyping using SBE/SBH assays requires the following steps (see Figure 3.1): (1) Synthesizing primers complementing the genomic sequence immediately preceding SNPs of interest; (2) Hybridizing primers with the genomic DNA; (3) Extending each primer by a single base using polymerase enzyme and dideoxynucleotides labeled with 4 different fluorescent dyes; and finally (4) Hybridizing extended primers to a universal DNA array and determining the identity of the bases that extend each primer by hybridization pattern analysis.

To the best of our knowledge the combination of the two technologies in the context

of SNP genotyping has not been explored thus far. The most closely related genotyping assay is the generic Polymerase Extension Assay (PEA) recently proposed in [43]. In PEA, short amplicons containing the SNPs of interest are hybridized to an all k-mers array of *primers* that are subsequently extended via single-base extension reactions. Hence, in PEA the SBE reactions take place on solid support, similar to *arrayed primer extension* (APEX) assays which use SNP specific primers spotted on the array [45].

As in [20], the SBE/SBH assay leads to high array probe utilization since we hybridize to the array a large number of short extended primers. However, the main power of the method lies in the fact that the sequences of the labeled oligonucleotides hybridized to the array are a priori known (up to the identity of extending nucleotides). While genotyping with SBE/SBH assays uses similar general principles as the PEA assays proposed in [43], there are also significant differences. A major advantage of SBE/SBH is the much shorter length of extended primers compared to that of PCR amplicons used in PEA. A second advantage is that *all* probes hybridizing to an extended primer are informative in SBE/SBH assays, regardless of array probe length (in contrast, only probes hybridizing with a substring containing the SNP site are informative in PEA assays). As shown by the experimental results in Section 3.4 these advantages translate into an increase by orders of magnitude in multiplexing rate compared to the results reported in [43]. We further note that PEA's effectiveness crucially depends on the ability to amplify very short (preferably 40bp or less) genomic fragments spanning the SNP loci of interest. This limits the achievable degree of multiplexing in PCR amplification [23], making PCR amplification the main bottleneck for PEA assays. Full flexibility in picking PCR primers is preserved in SBE/SBH assays.

The rest of the chapter is organized as follows. In Section 3.2 we formalize two problems that arise in genotyping large sets of SNPs using SBE/SBH assays: the problem of partitioning a set of SNPs into the minimum number of "decodable" subsets, i.e., subsets of SNPs that can be unambiguously genotyped using a single SBE/SBH assay, and that of finding a maximum decodable subset of a given set of SNPs. We also establish hardness results for the latter problem. In Section 3.3 we propose several efficient heuristics. Finally, in Section 3.4 we present experimental results on both randomly generated

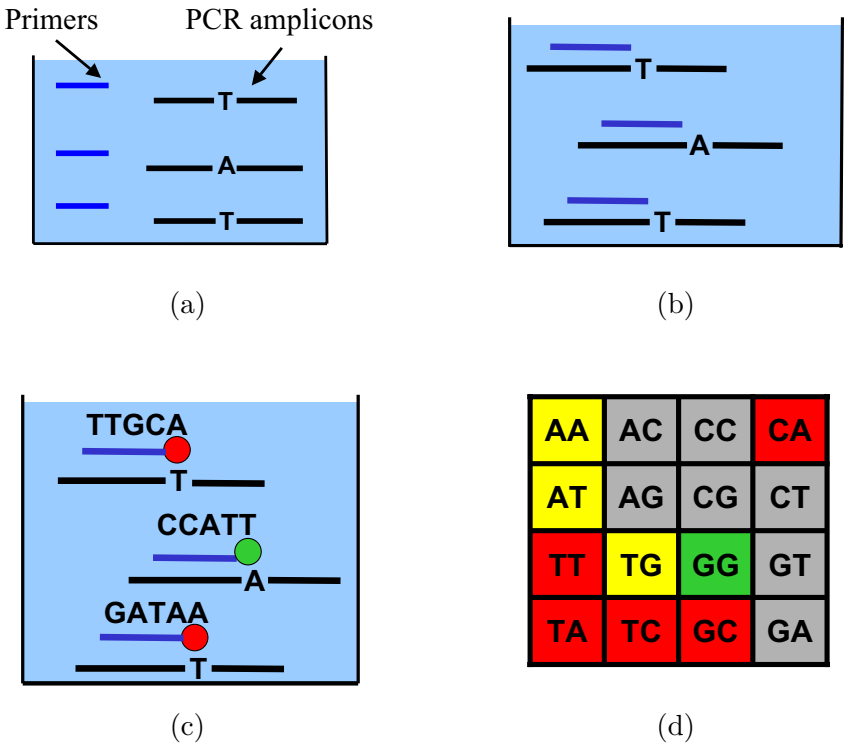


Figure 3.1: SBE/SBH assay: (a) Primers complementing genomic sequence upstream of each SNP locus are mixed in solution with the genomic DNA sample. (b) Temperature is lowered allowing primers to hybridize to the genomic DNA. (c) Polymerase enzyme and dideoxynucleotides labeled with 4 different fluorescent dyes are added to the solution, causing each primer to be extended by a nucleotide complementing the SNP allele. (d) Extended primers are hybridized to a universal DNA array (an all k-mer array for k=2 is shown) and SNP genotypes are determined by analyzing the resulting hybridization pattern. Under the assumption of perfect hybridization, unambiguous genotyping of the SNPs requires that each primer hybridizes to at least one array probe that hybridizes to no other primer that can be extended by a common base.

datasets and instances extracted from the NCBI dbSNP database, exploring achievable tradeoffs between the type/number of array probes and primer length on one hand and number of SNPs that can be assayed per array on the other. Our results suggest that the SBE/SBH architecture provides a flexible and cost-effective alternative to genotyping assays currently used in the industry, enabling genotyping of up to hundreds of thousands of user-selected SNPs per assay.

3.2 Problem Formulations and Complexity

A set of SNP loci can be unambiguously genotyped by SBE/SBH if every combination of SNP genotypes yields a different hybridization pattern (defined as the vector of dye colors observed at each array probe). To formalize the requirements of unambiguous genotyping, let us first consider a simplified SBE/SBH assay consisting of four parallel *single-color* SBE/SBH reactions, one for each possible SNP allele. Under this scenario, only one type of dideoxynucleotide is added to each SBE reaction, corresponding to the complement of the tested SNP allele. Therefore, a primer is extended in such a reaction if the tested allele is present at the SNP locus probed by the primer, and is left un-extended otherwise.

Let $\mathcal{P}$ be the set of primers used in a single-color SBE/SBH reaction involving dideoxynucleotide $e \in \{\text{A,C,G,T}\}$. From the resulting hybridization pattern we must be able to infer for every $p \in \mathcal{P}$ whether or not p was extended by e. The extension of p by e will result in a fluorescent signal at all array probes that hybridize with pe. However, some of these probes can give a fluorescent signal even when p is not extended by e, due to hybridization to other extended primers. Since in the worst case *all* other primers are extended, it must be the case that at least one of the probes that hybridize to pe does not hybridize to any other extended primer.

Formally, let $X \subset \{A, C, G, T\}^*$ be the set of array probes. For every string $y \in \{A, C, G, T\}^*$, let the *spectrum of y in X*, denoted $Spec_X(y)$, be the set of probes of X that hybridize with y. Under the assumption of perfect hybridization, $Spec_X(y)$ consists of those probes of X that are Watson-Crick complements of substrings of y. Then, a set of primers $\mathcal{P}$ is said to be *decodable* with respect to extension e if and only if, for every $p \in \mathcal{P}$,

$$Spec_X(pe) \setminus \bigcup_{p' \in \mathcal{P} \setminus \{p\}} Spec_X(p'e) \neq \emptyset \tag{3.1}$$

Decoding constraints (3.1) can be directly extended to 4-color SBE/SBH experiments, in which each type of extending base is labeled by a different fluorescent dye. As before, let $\mathcal{P}$ be the set of primers, and, for each primer $p \in \mathcal{P}$, let $E_p \subseteq \{A, C, G, T\}$ be the set of possible extensions of p, i.e., Watson-Crick complements of corresponding SNP alleles.

If we assume that any combination of dyes can be detected at an array probe location, unambiguous decoding is guaranteed if, for every $p \in \mathcal{P}$ and every extending nucleotide $e \in E_p$,

$$Spec_X(pe) \setminus \bigcup_{p' \in \mathcal{P} \setminus \{p\}, e \in E_{p'}} Spec_X(p'e) \neq \emptyset \tag{3.2}$$

In the following, we refine (3.2) to improve practical reliability of SBE/SBH assays. More precisely, we impose additional constraints on the set of probes considered to be *informative* for each SNP allele. First, to enable reliable genotyping of genomic samples that contain SNP alleles at very different concentrations (as a result of uneven efficiency in the PCR amplification step or of pooling DNA from different individuals), we require that a probe that is informative for a certain SNP locus must not hybridize to primers corresponding to different SNP loci, *regardless of their extension*. Second, since recent studies by Naef et al. [31] suggest that fluorescent dyes can significantly interfere with oligonucleotide hybridization on solid support, possibly destabilizing hybridization to a complementary probe on the array, we use a conservative approach and require that each probe that is informative for a certain SNP allele must hybridize to a strict substring of the corresponding primer. On the other hand, informative probes are still required not to hybridize with any other extended primer, even if such hybridizations involve fluorescently labeled nucleotides. Finally, we introduce a *decoding redundancy* parameter $r \geq 1$, and require that each SNP have at least r informative probes, i.e., probes that hybridize to the corresponding primer but do not hybridize to any other extended primer. Such a redundancy constraint facilitates reliable genotype calling in the presence of hybridization errors. Clearly, the larger the value of r, the more hybridization errors that can be tolerated. If a simple majority voting scheme is used for making allele calls, the assay can tolerate up to $\lfloor r/2 \rfloor$ hybridization errors involving the r informative probes of each SNP. Furthermore, since the informative probes of a SNP are required to hybridize *exclusively* with the primer corresponding to the SNP, the redundancy requirement provides a powerful mechanism for detecting and gauging the extent of hybridization errors. Indeed, each unintended hybridization at an informative probe for a bi-allelic SNP has a dye complementary to one of the SNP alleles with probability of 1/2, and the probability that k such errors pass undetected decreases exponentially in k.

The refined set of constraints is captured by the following definition, where, for every primer $p \in \{A, C, G, T\}^*$ and set of extensions $E \subseteq \{A, C, G, T\}$, we let

$$Spec_X(p, E) = \bigcup_{e \in E} Spec_X(pe)$$

Definition 1 *A set of primers $\mathcal{P}$ is said to be* strongly r-decodable *with respect to extension sets E_p, $p \in \mathcal{P}$, if and only if, for every $p \in \mathcal{P}$,*

$$\left| Spec_X(p) \setminus \bigcup_{p' \in \mathcal{P} \setminus \{p\}} Spec_X(p', E_{p'}) \right| \geq r \tag{3.3}$$

Note that testing whether or not a given set of primers is strongly r-decodable can be easily accomplished in time linear in the total length of the primers.

Genotyping a large set of SNPs will, in general, need more than one SBE/SBH assay. This rises the problem of partitioning a given set of SNPs into the smallest number of subsets that can each be genotyped using a single SBE/SBH assay. For each SNP locus there are typically two different primers that can be used for genotyping. As shown in [28] for the case of SNP genotyping using tag arrays, exploiting this degree of freedom significantly increases achievable multiplexing rates. Therefore, we next extend our definitions to capture this degree of freedom.

Let P_i be the *pool of primers* that can be used to genotype the SNP at locus i. Similarly to Definition 1, we have:

Definition 2 *A set of primer pools $\mathcal{P} = \{P_1, \ldots, P_n\}$ is said to be* strongly r-decodable *if and only if there is a primer p_i in each pool P_i such that $\{p_1, \ldots, p_n\}$ is strongly r-decodable with respect to the respective extension sets E_{p_i}, $i = 1, \ldots, n$.*

Primers $p_1, p_2, \ldots, p_n$ above are called the *representative primers* of pools $P_1, P_2, \ldots, P_n$, respectively.

The SNP partitioning problem can then be formulated as follows:

Minimum Pool Partitioning Problem (MPPP): *Given primer pools $\mathcal{P} = \{P_1, \ldots, P_n\}$, associated extension sets E_p, $p \in \cup_{i=1}^{n} P_i$, probe set X, and redundancy r, find a partitioning of $\mathcal{P}$ into the minimum number of strongly r-decodable subsets.*

A natural strategy for solving MPPP, similar to the greedy algorithm for the well-known set cover problem, is to find a maximum strongly r-decodable subset of pools, remove it from $\mathcal{P}$, and then repeat the procedure until no more pools are left in $\mathcal{P}$. This greedy strategy for solving MPPP has been shown to empirically outperform other algorithms for solving the similar partitioning problem for PEA assays [43]. In the case of SBE/SBH, the optimization involved in the main step of the greedy strategy is formalized as follows:

Maximum r-Decodable Pool Subset Problem (MDPSP): *Given primer pools $\mathcal{P} = \{P_1, \ldots, P_n\}$, associated extension sets E_p, $p \in \cup_{i=1}^{n} P_i$, probe set X, and redundancy r, find a strongly r-decodable subset $\mathcal{P}' \subseteq \mathcal{P}$ of maximum size. In addition, for each pool $P_i \in \mathcal{P}'$, find its representative primer.*

Unfortunately, as shown in next theorem, MDPSP is NP-hard even for the case when the redundancy parameter is 1 and each pool has exactly one primer.

Theorem 4 *MDPSP is NP-hard, even when restricted to instances with $r = 1$ and $|P| = 1$ for every $P \in \mathcal{P}$.*

Proof. We will use a reduction from the *maximum induced matching* problem in bipartite graphs, which is defined as follows:

Maximum Induced Matching (MIM) Problem in Bipartite Graphs: *Given a bipartite graph $G = (U \cup V, E)$, find maximum size subsets $U' \subseteq U$, $V' \subseteq V$, with $|U'| = |V'|$ such that the subgraph of G induced by $U' \cup V'$ is a matching.*

The MIM problem in bipartite graphs is known to be NP-hard even for graphs with maximum degree 3 [25]. Let $G = (U \cup V, E)$ be such a bipartite graph with maximum degree 3. Without loss of generality we may assume that every vertex in G has degree at least 1. We will denote by $N(u)$ the *neighborhood* of vertex $u \in U \cup V$, i.e., the set of vertices adjacent with u in G.

We construct an instance of MDPSP as follows: Let $r = 1$ and $l = \lceil \log_2 |V| \rceil$. For every $v \in V$ we add to X a distinct probe $x_v \in \{\mathrm{A,T}\}^l$; note that this can be done

since $|\{\text{A,T}\}^l| = 2^l > |V|$ by our choice of l. For every $u \in U$, with neighborhood $N(u) = \{v_1, v_2, v_3\}$, we construct a primer $p_u = x_{v_1}\text{C}x_{v_2}\text{C}x_{v_3}$ and set $P_u = \{p_u\}$. We use a similar construction for vertices $u \in U$ with only 1 or 2 neighbors. Note that in each case the pool P_u consists of a single primer p_u of length at most $3l + 2$. For each constructed primer p, the set of possible extensions is defined as $E_p = \{\text{G,C}\}$. Since the probes of X contain only A's and T's, for every primer p_u, $u \in U$,

$$Spec_X(p_u, E_{p_u}) = Spec_X(p_u) = \{x_v \in X |\ v \in N(u)\}$$

Let $U' \subseteq U$, $V' \subseteq V$, $|U'| = |V'|$, be subsets of vertices such that $U' \cup V'$ induces a matching in G. Let $\mathcal{P}' = \{P_u |\ u \in U'\}$. For every $u \in U'$, exactly one of u's neighbors, denoted v_u, appears in V', because $U' \cup V'$ induces a matching. Furthermore, for each $u' \in U' \setminus \{u\}$, $(u', v_u) \notin E$, and therefore $x_{v_u} \notin Spec_X(p_{u'}, E_{p_{u'}})$. Thus, for every $u \in U'$,

$$x_{v_u} \in Spec_X(p_u) \setminus \bigcup_{\{p_{u'}\} \in \mathcal{P}' \setminus \{p_u\}} Spec_X(p_{u'}, E_{p_{u'}})$$

which means that $\mathcal{P}'$ is a strongly 1-decodable subset of pools of the same size as the induced matching of G.

Conversely, let $\mathcal{P}'$ be a strongly 1-decodable subset of $\mathcal{P}$, and let $U' = \{u \in U |\ \{p_u\} \in \mathcal{P}'\}$. Since $\mathcal{P}'$ is 1-decodable, for every primer p_u with $\{p_u\} \in \mathcal{P}'$, there must exist a probe $x \in X$ such that $x \in Spec_X(p_u)$ and $x \notin Spec_X(p_{u'}, E_{p_{u'}})$ for every $\{p_{u'}\} \in \mathcal{P}' \setminus \{p_u\}$. Because $Spec_X(p_u) = \{x_v \in X |\ v \in N(u)\}$, it follows that every vertex $u \in U'$ has a neighbor $v \in V$ that is not a neighbor of any other $u' \in U' \setminus \{u\}$. Let v_u be such a neighbor (pick v_u arbitrarily if more than one vertex in V satisfies above property), and let $V' = \{v_u |\ u \in U'\}$. It is clear that $U' \cup V'$ induce a matching of size $|\mathcal{P}'|$ in G.

Thus, for every integer k, there is a one-to-one correspondence between induced matchings of size k in G and strongly 1-decodable subsets of k pools in the constructed instance of MDPSP, and NP-hardness of MDPSP follows. □

The reduction in the proof of Theorem 4 preserves the size of the optimal solution, and therefore any hardness of approximation result for the MIM in bipartite graphs will also hold for MDPSP, even when restricted to instances with $r = 1$ and $|P| = 1$ for every $P \in \mathcal{P}$. Since Duckworth et al. [15] proved that it is NP-hard to approximate MIM in bipartite graphs with maximum degree 3 within a factor of 6600/6659, we get:

```
Input: Pools P = {P_1, ..., P_n}, extension sets E_p, p ∈ ∪_{i=1}^n P_i, probe set X, and
redundancy r
Output: Strongly r-decodable subset of pools P' ⊆ P and set R of representative
primers for the pools in P'
-----------------------------------------------------------------------
0. P' ← ∅,  R ← ∅
1. For each P ∈ P do
2.    For each p ∈ P do
3.       If R ∪ {p} satisfies (3.3)
         Then
4.          P' ← P' ∪ P
4.          R ← R ∪ {p}
5.          Exit inner For
         End If
     End For
End For
```

Figure 3.2: The Sequential Greedy algorithm.

Theorem 5 *It is NP-hard to approximate MDPSP within a factor of 6600/6659, even when restricted to instances with* $r = 1$ *and* $|P| = 1$ *for every* $P \in \mathcal{P}$.

3.3 Algorithms

In this section we describe three heuristic approaches to MDPSP. The first one is a naive greedy algorithm that sequentially evaluates the primers in the given pools in an arbitrary order. The algorithm picks a primer p to be the representative of pool $P \in \mathcal{P}$ if p together with the representatives already picked satisfy condition (3.3). The pseudocode of this algorithm, which we refer to as Sequential Greedy, is given in Figure 3.2.

The next two algorithms are inspired by the MinGreedy algorithm in [15], which approximates MIM in d-regular graphs within a factor of $d - 1$. For the MIM problem, the MinGreedy algorithm picks at each step a vertex u of minimum degree and a vertex v, which is a minimum degree neighbor of u. All the neighbors of u and v are deleted and the edge (u, v) is added to the induced matching. The algorithm stops when the graph becomes empty.

Each instance of MDPSP can be represented as a bipartite *hybridization graph* $G = ((\bigcup_{i=1}^{n} P_i) \cup X, E)$, with the left side containing all primers in the given pools and the right side containing the array probes, i.e., X. There is an edge between primer p and

probe $x \in X$ iff $x \in Spec_X(p, E_p)$. As discussed in Section 3.2, we need to distinguish between the hybridizations that involve fluorescently labeled nucleotides and those that do not. Thus, for every primer p, we let $N^+(p) = Spec_X(p)$ and $N^-(p) = Spec_X(p, E_p) \setminus Spec_X(p)$. Similarly, for each probe $x \in X$, we let $N^+(x) = \{p|\ x \in N^+(p)\}$ and $N^-(x) = \{p|\ x \in N^-(p)\}$.

We considered two versions of the MinGreedy algorithm when run on the bipartite hybridization graph, depending on the side from which the minimum degree vertex is picked. In the first version, referred to as MinPrimerGreedy, we pick first a minimum degree node from the primers side, while in the second version, referred to as MinProbeGreedy, we pick first a minimum degree node from the probes side. Thus, MinPrimerGreedy greedy picks at each step a minimum degree primer p and pairs it with a minimum degree probe $x \in N^+(p)$. MinProbeGreedy greedy, selects at each step a minimum degree probe x and pairs it with a minimum degree primer p in $N^+(x)$. In both algorithms, all neighbors of p and x and their incident edges are removed from G. Also, at each step, the algorithms remove all vertices u, for which $N^+(u) = \emptyset$. These deletions ensure that the primers p selected at each step satisfy condition (3.3). Both algorithms stop when the graph becomes empty.

As described so far, the MinPrimerGreedy and MinProbeGreedy algorithms work when each pool contains only one primer and when the redundancy is 1. We extended the two variants to handle pools of size greater than 1 by simply removing from the graph all primers $p' \in P \setminus \{p\}$ when picking primer p from pool P. If the redundancy r is greater than 1, then whenever we pick a primer p, we also pick it's r probe neighbors from $N^+(p)$ with the smallest degrees (breaking ties arbitrarily). The primer neighbors of all these r probes will then be deleted from the graph. Moreover, at each step, all primers p for which $|N^+(p)| < r$ are removed from the graph. Thus, the algorithm maintains the invariant that $|N^+(p)| \geq r$ for every primer p and $|N^+(x)| \geq 1$ for every probe x. Figures 3.5 and 3.6 give the pseudocode for the MinPrimerGreedy, respectively the MinProbeGreedy greedy algorithms. For the sake of clarity, they use two subroutines for removing a primer vertex, respectively a probe vertex, which are described in Figures 3.3 and 3.4.

```
remove-primer (p)
---------------------------------------------------
Begin
     For all x ∈ N+(p) do
        N+(x) ← N+(x) \ {p}
        If |N+(x)| = 0
        Then remove-probe (x)
        End If
     End For
     For all x ∈ N−(p) do
        N−(x) ← N−(x) \ {p}
     End For
     Delete vertex p from graph G
End
```

Figure 3.3: The `remove-primer` subroutine.

Algorithms MinPrimerGreedy and MinProbeGreedy can be implemented efficiently using a Fibonacci heap for maintaining the degrees of primers, respectively of probes. Let N be the total number of primers in the n pools, m be the number of probes in X, and k be the size of the r-decodable set returned by the algorithm. Since each primer has bounded degree, the sorting of probe degrees requires $O(k)$ total time. The total number of edges in the hybridization graph is $O(N + m)$. Since by using a Fibonacci heap, finding a minimum degree primer (probe) can be done in $O(\log N)$ (respectively $O(\log m)$) and each primer degree update can be done in amortized $O(1)$ time, it follows that the total runtime for MinPrimerGreedy algorithm is $O(k \log N + N + m)$, and the total runtime for MinProbeGreedy algorithm is $O(k \log m + N + m)$.

3.4 Experimental Results

We considered two types of data sets:

- Randomly generated datasets containing between 1,000 to 200,000 pools with 1 or 2 primers of length between 10 and 30.

- Two-primer pools representing over 9 million reference SNPs in human chromosomes 1-22, X, and Y extracted from the NCBI dbSNP database build 125. We disregarded reference SNPs for which available flanking sequence was insufficient

```
remove-probe (x)
---------------------------------------------------
Begin
     For all p ∈ N+(x) do
        N+(p) ← N+(p) \ {x}
        If |N+(p)| < r
        Then remove-primer (p)
        End If
     End For
     For all p ∈ N−(x) do
        N−(p) ← N−(p) \ {x}
     End For
     Delete vertex x from graph G
End
```

Figure 3.4: The `remove-probe` subroutine.

for determining two non-degenerate primers of desired length (due, e.g., to the presence of degenerate bases near the SNP locus).

We used two types of array probe sets. First, we used probe sets containing all k-mers, for k between 8 and 10. All k-mer arrays are well studied in the context of sequencing by hybridization. However, a major drawback of all k-mer arrays is that the k-mers have a wide range of melting temperatures, making it difficult to ensure reliable hybridization results. For short oligonucleotides, a good approximation of the melting temperature is obtained using the simple 2-4 rule of Wallace [46], according to which the melting temperature of a probe is approximately twice the number of A and T bases, plus four times the number of C and G bases. As in [5], we define the *weight* of a DNA string to be the number of A and T bases plus twice the number of C and G bases. For a given integer c, a DNA string is called a c-token if it has a weight c or more and all its proper suffixes have weight strictly less than c. Since the weight of a c-token is either c or $c + 1$, it follows that the 2-4 rule computed melting temperature of all c-tokens varies in a range of about 4°C. In our experiments we used probe sets consisting of all c-tokens, with c varying between 11 and 13. The considered values of k and c were picked such that the resulting number of probes is representative of current array manufacturing technologies: there are roughly 65,000 8-mers, 262,000 9-mers, 1 million

```
Input: Pools 𝒫 = {P_1, ..., P_n}, extension sets E_p, p ∈ ∪_{i=1}^n P_i, probe set X, and
redundancy r
Output: Strongly r-decodable subset of pools 𝒫' ⊆ 𝒫 and set R of representative
primers for the pools in 𝒫'
---------------------------------------------------------------------------
Construct hybridization graph G
𝒫' ← ∅
R ← ∅
While G is not empty do
      Find a minimum degree primer p, and let P be the pool of p
      𝒫' ← 𝒫' ∪ {P}
      R ← R ∪ {p}
      For each (p') ∈ P \ {p} do
         remove-primer(p')
      End For
      Let |N+(p)| = k and let {x_1, ..., x_k} be the probes in N+(p), indexed
      in increasing order of their degrees
      For each x ∈ {x_1, ..., x_r} do
         For each (p') ∈ N+(x) ∪ N−(x) do
            remove-primer(p')
         End For
         Delete vertex x from G
      End For
      For each x ∈ {x_{r+1}, ..., x_k} ∪ N−(p) do
         remove-probe(x)
      End For
End While
```

Figure 3.5: MinPrimerGreedy greedy algorithm.

10-mers, 86,000 11-tokens, 236,000 12-tokens, and 645,000 13-tokens – the smaller probe sets can be spotted using current oligonucleotide printing robots, while the larger probe sets can be synthesized in situ using photolithographic techniques.

3.4.1 Results on Synthetic Datasets

In a first set of experiments on the randomly generated datasets we compared the three MDPSP algorithms on instances with primer length set to 20, which is the typical length used, e.g., in genotyping using tag arrays. In these experiments the set of possible extensions was considered to be {A,C,T,G} for all primers. Such a conservative choice gives an estimate of multiplexing rates achievable by SBE/SBH assays in more demanding genomic analyses such as microorganism identification by DNA barcoding [12], in which

```
Input: Pools P = {P_1, ..., P_n}, extension sets E_p, p ∈ ∪_{i=1}^n P_i, probe set X, and
redundancy r
Output: Strongly r-decodable subset of pools P' ⊆ P and set R of representative
primers for the pools in P'
---------------------------------------------------------------------------
Construct hybridization graph G
P' ← ∅
R ← ∅
While G is not empty do
      Find a minimum degree probe x
      Find a minimum degree primer p in N+(x), and let P be the pool of p
      P' ← P' ∪ {P}
      R ← R ∪ {p}
      For each p' ∈ P \ {p} do
         remove-primer(p')
      End For
      Let |N+(p)| = k and let {x_1, ..., x_k} be the probes in N+(p), indexed
      in increasing order of their degrees
      For each x ∈ {x_1, ..., x_r} do
         For each p' ∈ N+(x) ∪ N-(x) do
            remove-primer(p')
         End For
         Delete vertex x from G
      End For
      For each x ∈ {x_{r+1}, ..., x_k} ∪ N-(p) do
         remove-probe(x)
      End For
End While
```

Figure 3.6: MinProbeGreedy greedy algorithm.

a primer (typically referred to as a *distinguisher* in this context) may be extended by any of the DNA bases in different microorganisms. The results of these experiments for all k-mer and all c-token probe sets are presented in Tables 3.1 and 3.2, respectively. The results show that using the flexibility of picking primers from either strand of the genomic sequence yields an improvement of up to 10% in the number of r-decodable pools. The MinProbeGreedy algorithm typically produces better results compared to the MinPrimerGreedy variant. On the other hand, neither Sequential Greedy nor MinProbeGreedy dominates the other algorithms for all range of instance parameters – Sequential Greedy generally gives the better results for k-mer experiments with high redundancy values, while MinProbeGreedy generally gives better results for k-mer experiments with

large number of pools and low redundancy and for c-token experiments.

In the second set of experiments we ran the three MDPSP algorithms on datasets with the same primer length of 20, pool size of 2, and with the number of possible extensions of each primer set to 4 as in DNA-barcoding applications, and to 2 as in SNP genotyping. The results for all k-mer and all c-token probe sets are given in Tables 3.3 and 3.4. The relative performance of the algorithms is similar to that observed in the first set of experiments. As expected, taking into account the reduced number of possible extensions increases the size of computed decodable pool subsets, often by more than 5%.

In the third set of experiments we explored the degree of freedom given by the primer length. For any fixed array probe set and redundancy requirement, we need a minimum primer length to be able to satisfy constraints (3.3). Increasing the primer length beyond this minimum primer length is often beneficial, as it increases the number of array probes that hybridize with the primer. However, if primer length increases too much, an increasing number of these probes become non-specific, and the multiplexing rate starts to decline. Figure 3.7 gives the tradeoff between primer length and the size of the strongly r-decodable pool subsets computed by the three MDPSP algorithms for pools with 2 primers, 2 possible extensions per primer and all 10-mers, respectively all 13-tokens, as array probes. We notice that the optimal primer length increases with the redundancy parameter.

3.4.2 Results on dbSNP Data

To stress-test our methods, we extracted a total of over 9 million 2-primer pools corresponding to reference SNPs in human chromosomes 1-22, X, and Y in the NCBI dbSNP database build 125. We constructed a dataset for each of the 24 chromosomes by creating a 2-primer pool for each reference SNP for which dbSNP contains at least 20 non-degenerate base pairs of flanking sequence on both sides (the number of reference SNPs and extracted pools for each chromosome are given in Table 3.5). Since these large sets of pools must be partitioned between multiple SBE/SBH experiments, we used a simple MPPP algorithm which iteratively finds maximum r-decodable pool subsets using the sequential greedy algorithm.

Figures 3.8 and 3.9 give the cumulative coverage percentage for the first 50 arrays of all 10-mers, respectively all 13-tokens, on the set of pools extracted from the human chromosome 1. In these experiments we used redundancy between 1 and 5, and primer length 14 or 20. While the MDPSP size in the first few iterations of our MPPP algorithm is comparable to those reported for randomly generated datasets in Section 3.4.1, the number of SNPs assayed per array decreases constantly with array number – as we need to assay more and more "difficult" SNPs. Somehow unexpectedly, the results also suggest using primers of different lengths in different SBE/SBH experiments: while a primer length of 14 seems to be optimal for the first few arrays, longer primers improve the degree of multiplexing when only hard to differentiate SNPs remain, especially for high redundancy.

Finally, in Table 3.5 we give the number of arrays (containing either all 10-mers or all 13-tokens) required to cover 90%, respectively 95% of the extracted reference SNPs, when using primers of length 20. In practical association studies a much lower SNP coverage (and hence much fewer arrays) would be required due to the high degree of linkage disequilibrium between the SNPs in the human population [37].

Table 3.1: Size of the strongly r-decodable pool subset computed by the three MDPSP algorithms for primer length 20 and set of possible extensions {A,C,T,G}, with redundancy $r \in \{1, 2, 5\}$ and all k-mer probe sets for $k \in \{8, 9, 10\}$ (averages over 10 test cases).

r	# pools	Algorithm	k=8		k=9		k=10	
			1 primer	2 primers	1 primer	2 primers	1 primer	2 primers
1	1000	Sequential	1000	1000	1000	1000	1000	1000
		MinPrimer	1000	1000	1000	1000	1000	1000
		MinProbe	1000	1000	1000	1000	1000	1000
	2000	Sequential	2000	2000	2000	2000	2000	2000
		MinPrimer	2000	2000	2000	2000	2000	2000
		MinProbe	2000	2000	2000	2000	2000	2000
	10000	Sequential	7740	8574	9991	10000	10000	10000
		MinPrimer	7714	8319	9991	9999	10000	10000
		MinProbe	7768	8803	9991	10000	10000	10000
	20000	Sequential	9967	11071	19436	19948	19999	20000
		MinPrimer	9889	10999	19447	19745	19999	20000
		MinProbe	9886	11107	19458	19989	19999	20000
	100000	Sequential	12486	12656	43279	47688	93632	98630
		MinPrimer	13864	15324	42980	48021	93642	96712
		MinProbe	13993	15672	43273	48418	93837	99601
	200000	Sequential	12635	12658	49062	51646	140820	157908
		MinPrimer	15476	17010	50347	56017	139787	154028
		MinProbe	15822	17630	50459	56676	141614	160532
2	1000	Sequential	1000	1000	1000	1000	1000	1000
		MinPrimer	1000	1000	1000	1000	1000	1000
		MinProbe	1000	1000	1000	1000	1000	1000
	2000	Sequential	1997	2000	2000	2000	2000	2000
		MinPrimer	1997	2000	2000	2000	2000	2000
		MinProbe	1997	2000	2000	2000	2000	2000
	10000	Sequential	6210	6901	9934	9999	10000	10000
		MinPrimer	6002	6463	9932	9977	10000	10000
		MinProbe	6174	6890	9938	9998	10000	10000
	20000	Sequential	7463	8192	17948	19274	19992	20000
		MinPrimer	7052	7662	17812	18455	19992	20000
		MinProbe	7435	8068	18004	19288	19993	20000
	100000	Sequential	9254	9644	31845	34855	82315	90627
		MinPrimer	8917	9605	30043	32700	81056	85852
		MinProbe	9404	10273	31805	34481	82522	90935
	200000	Sequential	9674	9953	35514	37891	109450	122470
		MinPrimer	9658	10333	33479	36247	104891	114624
		MinProbe	10326	11246	35228	38498	109252	122986
5	1000	Sequential	995	1000	1000	1000	1000	1000
		MinPrimer	995	999	1000	1000	1000	1000
		MinProbe	995	1000	1000	1000	1000	1000
	2000	Sequential	1872	1973	1998	2000	2000	2000
		MinPrimer	1860	1898	1998	2000	2000	2000
		MinProbe	1866	1946	1998	2000	2000	2000
	10000	Sequential	3745	4161	8674	9483	9972	10000
		MinPrimer	3376	3635	8484	8881	9969	9998
		MinProbe	3480	3845	8564	9233	9970	10000
	20000	Sequential	4289	4705	12204	13750	19498	19967
		MinPrimer	3748	4029	11393	12360	19435	19804
		MinProbe	3943	4286	11680	12960	19468	19931
	100000	Sequential	5241	5520	17920	19612	52078	59021
		MinPrimer	4450	4726	15580	16781	47922	52711
		MinProbe	4818	5171	16521	17990	49329	55573
	200000	Sequential	5534	5775	19767	21251	62791	70334
		MinPrimer	4724	4990	16959	18116	56160	61406
		MinProbe	5177	5531	18175	19757	58565	65344

Table 3.2: Size of the strongly r-decodable pool subset computed by the three MDPSP algorithms for primer length 20 and set of possible extensions {A,C,T,G}, with redundancy $r \in \{1, 2, 5\}$ and all c-token probe sets for $c \in \{11, 12, 13\}$ (averages over 10 test cases).

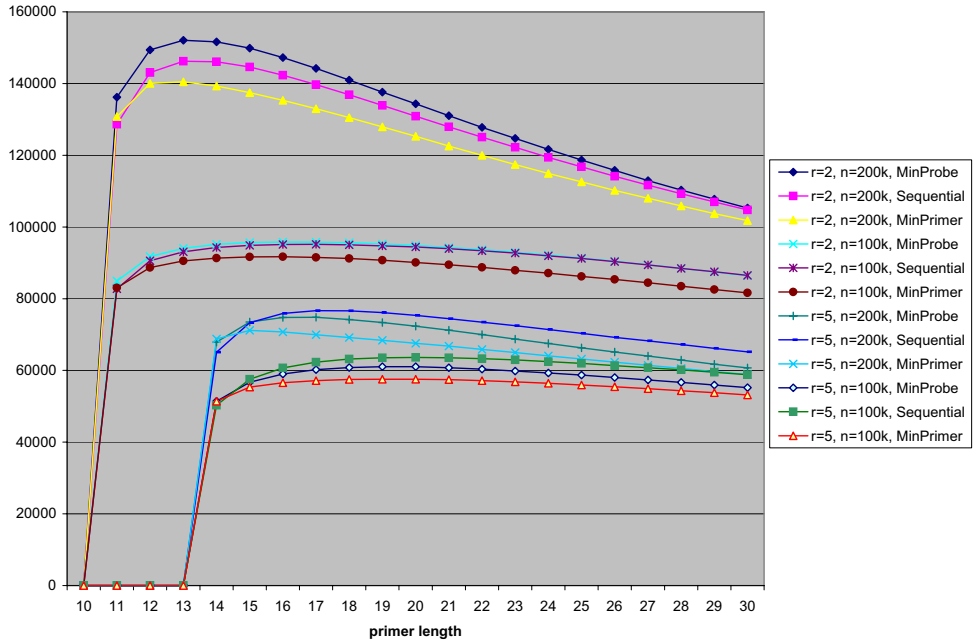
r	# pools	Algorithm	c=11		c=12		c=13	
			1 primer	2 primers	1 primer	2 primers	1 primer	2 primers
1	1000	Sequential	991	1000	999	1000	1000	1000
		MinPrimer	992	999	999	1000	1000	1000
		MinProbe	993	1000	999	1000	1000	1000
	2000	Sequential	1881	1982	1986	2000	1999	2000
		MinPrimer	1890	1959	1987	1998	1999	2000
		MinProbe	1906	1994	1988	2000	1999	2000
	10000	Sequential	5745	6993	8006	9218	9420	9927
		MinPrimer	5556	6401	8005	8782	9472	9801
		MinProbe	6385	7972	8436	9688	9550	9980
	20000	Sequential	7968	9733	12458	15191	16656	18931
		MinPrimer	7490	8798	12242	14080	16673	18204
		MinProbe	9190	11548	13684	17094	17430	19613
	100000	Sequential	13708	16042	26407	32202	45064	56064
		MinPrimer	12564	14736	24482	29336	42824	51540
		MinProbe	16820	20277	31414	39202	51448	65877
	200000	Sequential	16241	18516	33278	39552	61351	76037
		MinPrimer	14967	17278	30762	36618	57530	70048
		MinProbe	20574	24329	40580	49300	72230	91488
2	1000	Sequential	965	998	997	1000	1000	1000
		MinPrimer	965	986	997	999	1000	1000
		MinProbe	972	998	997	1000	1000	1000
	2000	Sequential	1711	1905	1940	1995	1995	2000
		MinPrimer	1697	1815	1942	1981	1995	2000
		MinProbe	1766	1948	1951	1997	1996	2000
	10000	Sequential	4216	5107	6578	7891	8616	9611
		MinPrimer	3926	4571	6344	7252	8572	9214
		MinProbe	4876	6059	7138	8610	8896	9783
	20000	Sequential	5482	6589	9450	11615	14060	16839
		MinPrimer	5024	5901	8919	10551	13699	15613
		MinProbe	6635	8151	10796	13540	15152	17980
	100000	Sequential	8587	9839	17469	20811	32223	39839
		MinPrimer	7897	9071	16133	19192	30138	36595
		MinProbe	10990	12695	21738	26341	38246	48131
	200000	Sequential	9899	11114	21192	24696	41783	50811
		MinPrimer	9149	10418	19730	23155	39125	47357
		MinProbe	12782	14541	26957	31714	51198	63112
5	1000	Sequential	787	906	947	992	992	1000
		MinPrimer	767	837	941	971	992	999
		MinProbe	794	905	947	990	992	1000
	2000	Sequential	1187	1433	1646	1870	1914	1991
		MinPrimer	1112	1284	1600	1753	1903	1960
		MinProbe	1204	1437	1652	1856	1914	1986
	10000	Sequential	2262	2713	4046	4988	6284	7662
		MinPrimer	2067	2467	3732	4495	5939	6976
		MinProbe	2363	2875	4154	5118	6324	7651
	20000	Sequential	2779	3279	5347	6540	9139	11399
		MinPrimer	2553	2998	4908	5956	8504	10308
		MinProbe	2957	3562	5520	6808	9222	11530
	100000	Sequential	4020	4536	8753	10211	17580	21359
		MinPrimer	3738	4250	8122	9494	16252	19645
		MinProbe	4509	5208	9284	11078	18048	22119
	200000	Sequential	4538	5035	10286	11738	21762	25859
		MinPrimer	4264	4749	9609	11054	20226	24058
		MinProbe	5221	5926	11149	12986	22602	27186

Table 3.3: Size of the strongly r-decodable pool subset computed by the three MDPSP algorithms for primer length 20 and 2 primers per pool, with number of possible extensions $|E_p| \in \{2, 4\}$, redundancy $r \in \{1, 2, 5\}$ and all k-mer probe sets for $k \in \{8, 9, 10\}$ (averages over 10 test cases).

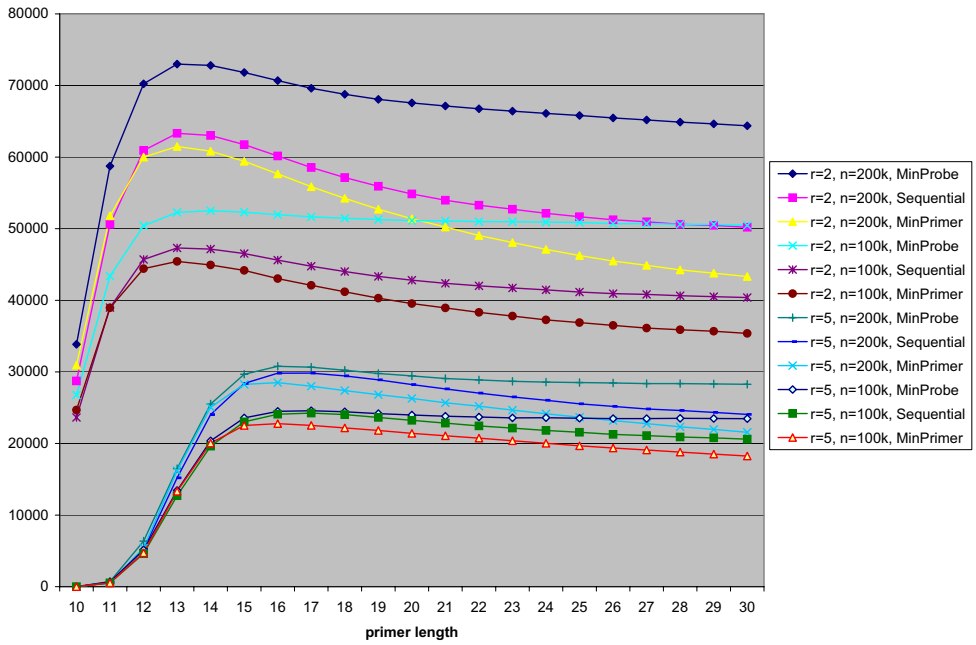
r	# SNPs	Algorithm	k=8		k=9		k=10	
			$\|E_p\| = 4$	$\|E_p\| = 2$	$\|E_p\| = 4$	$\|E_p\| = 2$	$\|E_p\| = 4$	$\|E_p\| = 2$
1	1000	Sequential	1000	1000	1000	1000	1000	1000
		MinPrimer	1000	1000	1000	1000	1000	1000
		MinProbe	1000	1000	1000	1000	1000	1000
	2000	Sequential	2000	2000	2000	2000	2000	2000
		MinPrimer	2000	2000	2000	2000	2000	2000
		MinProbe	2000	2000	2000	2000	2000	2000
	10000	Sequential	8574	8950	10000	10000	10000	10000
		MinPrimer	8319	8752	9999	10000	10000	10000
		MinProbe	8803	9358	10000	10000	10000	10000
	20000	Sequential	11071	11673	19948	19981	20000	20000
		MinPrimer	10999	11898	19745	19873	20000	20000
		MinProbe	11107	12051	19989	19998	20000	20000
	100000	Sequential	12656	13813	47688	50643	98630	99478
		MinPrimer	15324	16551	48021	52263	96712	98209
		MinProbe	15672	16800	48418	52712	99601	99885
	200000	Sequential	12658	13890	51646	55694	157908	166796
		MinPrimer	17010	18216	56017	60962	154028	164696
		MinProbe	17630	18783	56676	61488	160532	173910
2	1000	Sequential	1000	1000	1000	1000	1000	1000
		MinPrimer	1000	1000	1000	1000	1000	1000
		MinProbe	1000	1000	1000	1000	1000	1000
	2000	Sequential	2000	2000	2000	2000	2000	2000
		MinPrimer	2000	2000	2000	2000	2000	2000
		MinProbe	2000	2000	2000	2000	2000	2000
	10000	Sequential	6901	7325	9999	10000	10000	10000
		MinPrimer	6463	6977	9977	9993	10000	10000
		MinProbe	6890	7443	9998	9999	10000	10000
	20000	Sequential	8192	8639	19274	19670	20000	20000
		MinPrimer	7662	8348	18455	18988	20000	20000
		MinProbe	8068	8808	19288	19661	20000	20000
	100000	Sequential	9644	10175	34855	36886	90627	94420
		MinPrimer	9605	10398	32700	35771	85852	90098
		MinProbe	10273	11093	34481	37743	90935	94868
	200000	Sequential	9953	10535	37891	40060	122470	130911
		MinPrimer	10333	11143	36247	39619	114624	125287
		MinProbe	11246	12068	38498	41857	122986	134342
5	1000	Sequential	1000	1000	1000	1000	1000	1000
		MinPrimer	999	1000	1000	1000	1000	1000
		MinProbe	1000	1000	1000	1000	1000	1000
	2000	Sequential	1973	1989	2000	2000	2000	2000
		MinPrimer	1898	1933	2000	2000	2000	2000
		MinProbe	1946	1975	2000	2000	2000	2000
	10000	Sequential	4161	4405	9483	9722	10000	10000
		MinPrimer	3635	3970	8881	9211	9998	9999
		MinProbe	3845	4204	9233	9546	10000	10000
	20000	Sequential	4705	4924	13750	14739	19967	19985
		MinPrimer	4029	4391	12360	13378	19804	19905
		MinProbe	4286	4690	12960	14110	19931	19973
	100000	Sequential	5520	5727	19612	20634	59021	63631
		MinPrimer	4726	5114	16781	18352	52711	57521
		MinProbe	5171	5581	17990	19741	55573	61043
	200000	Sequential	5775	5970	21251	22193	70334	75361
		MinPrimer	4990	5375	18116	19732	61406	67565
		MinProbe	5531	5939	19757	21555	65344	72313

Table 3.4: Size of the strongly r-decodable pool subset computed by the three MDPSP algorithms for primer length 20 and 2 primers per pool, with number of possible extensions $|E_p| \in \{2, 4\}$, redundancy $r \in \{1, 2, 5\}$ and all c-token probe sets for $c \in \{11, 12, 13\}$ (averages over 10 test cases).

r	# SNPs	Algorithm	c=11		c=12		c=13	
			$\|E_p\| = 4$	$\|E_p\| = 2$	$\|E_p\| = 4$	$\|E_p\| = 2$	$\|E_p\| = 4$	$\|E_p\| = 2$
1	1000	Sequential	1000	1000	1000	1000	1000	1000
		MinPrimer	999	999	1000	1000	1000	1000
		MinProbe	1000	1000	1000	1000	1000	1000
	2000	Sequential	1982	1990	2000	2000	2000	2000
		MinPrimer	1959	1968	1998	1998	2000	2000
		MinProbe	1994	1998	2000	2000	2000	2000
	10000	Sequential	6993	7324	9218	9412	9927	9953
		MinPrimer	6401	6776	8782	9034	9801	9866
		MinProbe	7972	8280	9688	9782	9980	9990
	20000	Sequential	9733	10358	15191	15843	18931	19197
		MinPrimer	8798	9489	14080	14797	18204	18573
		MinProbe	11548	12187	17094	17599	19613	19746
	100000	Sequential	16042	17216	32202	34459	56064	59498
		MinPrimer	14736	15817	29336	31608	51540	55031
		MinProbe	20277	21599	39202	41665	65877	69188
	200000	Sequential	18516	19789	39552	42556	76037	81443
		MinPrimer	17278	18483	36618	39500	70048	75470
		MinProbe	24329	25757	49300	52534	91488	97154
2	1000	Sequential	998	998	1000	1000	1000	1000
		MinPrimer	986	990	999	1000	1000	1000
		MinProbe	998	999	1000	1000	1000	1000
	2000	Sequential	1905	1931	1995	1998	2000	2000
		MinPrimer	1815	1852	1981	1986	2000	2000
		MinProbe	1948	1962	1997	1999	2000	2000
	10000	Sequential	5107	5431	7891	8231	9611	9716
		MinPrimer	4571	4924	7252	7621	9214	9381
		MinProbe	6059	6372	8610	8833	9783	9851
	20000	Sequential	6589	7036	11615	12312	16839	17409
		MinPrimer	5901	6388	10551	11255	15613	16231
		MinProbe	8151	8674	13540	14184	17980	18396
	100000	Sequential	9839	10552	20811	22486	39839	42814
		MinPrimer	9071	9819	19192	20864	36595	39542
		MinProbe	12695	13562	26341	28190	48131	51125
	200000	Sequential	11114	11894	24696	26659	50811	54858
		MinPrimer	10418	11212	23155	25122	47357	51390
		MinProbe	14541	15467	31714	34015	63112	67567
5	1000	Sequential	906	932	992	996	1000	1000
		MinPrimer	837	868	971	981	999	999
		MinProbe	905	928	990	994	1000	1000
	2000	Sequential	1433	1497	1870	1896	1991	1995
		MinPrimer	1284	1350	1753	1800	1960	1974
		MinProbe	1437	1511	1856	1885	1986	1990
	10000	Sequential	2713	2944	4988	5343	7662	8000
		MinPrimer	2467	2668	4495	4825	6976	7324
		MinProbe	2875	3081	5118	5436	7651	7988
	20000	Sequential	3279	3552	6540	7040	11399	12143
		MinPrimer	2998	3273	5956	6424	10308	11007
		MinProbe	3562	3817	6808	7314	11530	12240
	100000	Sequential	4536	4912	10211	11140	21359	23232
		MinPrimer	4250	4610	9494	10352	19645	21421
		MinProbe	5208	5602	11078	11932	22119	23977
	200000	Sequential	5035	5443	11738	12809	25859	28234
		MinPrimer	4749	5128	11054	12022	24058	26297
		MinProbe	5926	6363	12986	13987	27186	29439



(a)



(b)

Figure 3.7: Size of the strongly r-decodable pool subset computed by the three MDPSP algorithms as a function of primer length, for pools with 2 primers, 2 possible extensions per primer, and array probes consisting of all 4^{10} 10-mers (a), respectively all 645,376 13-tokens (b) (averages over 10 test cases).

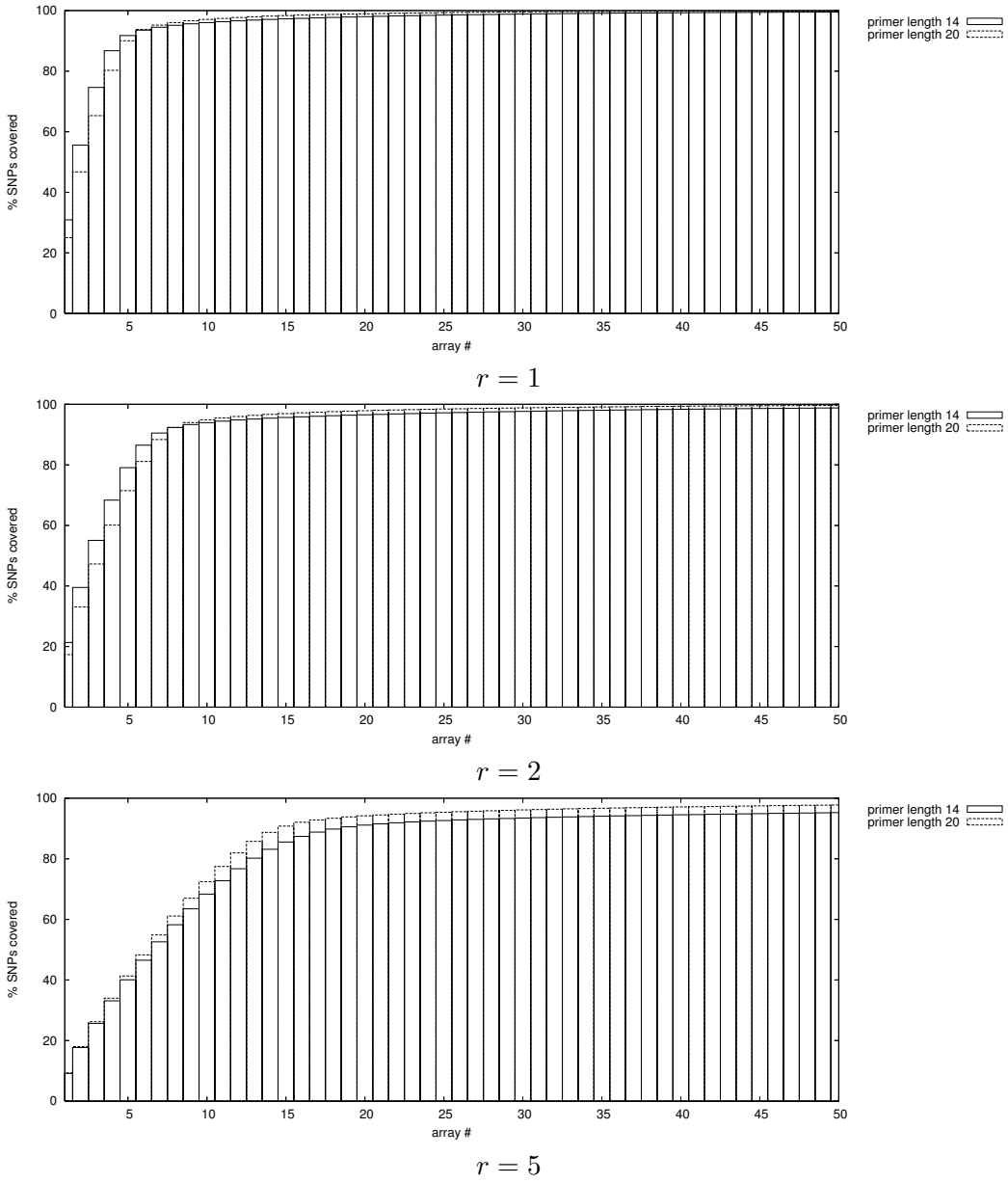


Figure 3.8: Cumulative coverage rates for the first 50 10-mers arrays used to decode the SNPs in Chromosome 1 with primer length 14 or 20 and redundancy $r \in \{1, 2, 5\}$.

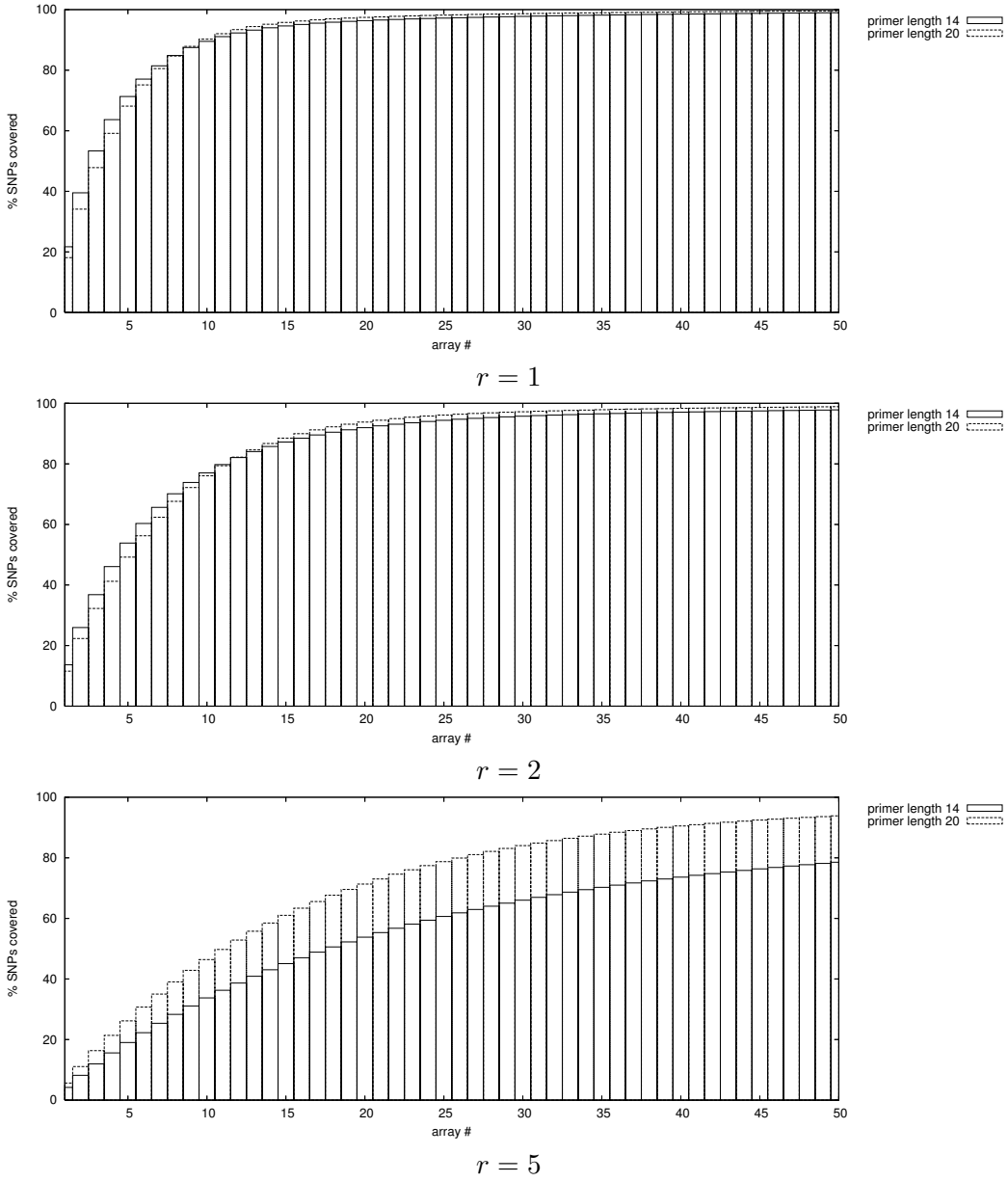


Figure 3.9: Cumulative coverage rates for the first 50 13-tokens arrays used to decode the SNPs in Chromosome 1 with primer length 14 or 20 and redundancy $r \in \{1, 2, 5\}$.

Table 3.5: Number of arrays needed to cover $90-95\%$ of the reference SNPs that have unambiguous primers of length 20.

Chr ID	# Ref. SNPs	# Extracted Pools	# 10-mer arrays						# 13-token arrays					
			r=1		r=2		r=5		r=1		r=2		r=5	
			90%	95%	90%	95%	90%	95%	90%	95%	90%	95%	90%	95%
1	786058	736850	5	7	8	11	15	24	10	14	17	23	39	56
2	758368	704415	5	6	7	9	14	18	9	12	14	18	32	42
3	647918	587531	5	6	7	8	13	16	8	10	12	15	26	35
4	690063	646534	5	6	7	9	14	17	8	10	12	15	26	34
5	590891	550794	5	6	6	8	12	16	7	10	12	15	26	34
6	791255	742894	10	20	14	29	30	54	15	29	23	38	49	73
7	666932	629089	6	9	8	12	16	25	10	15	16	22	36	48
8	488654	456856	4	5	5	7	10	12	7	8	10	13	22	29
9	465325	441627	4	6	6	8	11	17	7	10	11	16	26	36
10	512165	480614	4	6	6	8	11	16	8	10	12	16	27	38
11	505641	476379	4	6	6	8	11	15	8	10	12	15	26	35
12	474310	443988	4	6	6	8	11	18	7	10	11	15	25	36
13	371187	347921	3	4	5	6	9	11	5	7	8	10	16	22
14	292173	271130	3	4	4	5	7	10	5	7	8	10	16	23
15	277543	258094	3	4	4	5	7	11	5	7	8	10	17	24
16	306530	288652	4	6	5	9	9	18	7	10	11	15	25	35
17	269887	249563	3	5	4	8	9	18	7	10	11	15	25	37
18	268582	250594	3	3	4	5	7	9	4	6	6	8	14	18
19	212057	199221	4	6	5	9	11	21	8	11	12	17	29	43
20	292248	262567	3	4	4	5	7	11	6	8	9	12	20	27
21	148798	138825	2	3	3	3	5	6	3	4	5	6	10	13
22	175939	164632	3	4	3	6	6	13	6	8	9	12	21	29
X	380246	362778	4	6	6	8	10	15	6	9	9	13	19	26
Y	50725	49372	2	2	2	2	3	3	2	2	2	3	4	5

Chapter 4

Conclusions

In the first part of this thesis we have studied methods for improving the multiplexing rate in SNP genotyping assays involving multiple universal tag arrays. We observed that significant improvements in multiplexing rate can be achieved by combining primer selection with tag assignment, since different primers hybridize to different sets of tags. This integrated optimization is shown to lead to a reduction of up to 50% in the number of required arrays. Our heuristics currently focus on preventing primer-to-tag hybridizations, but they can be extended to avoid other undesired hybridizations, such as primer-to-antitag hybridizations. In ongoing work we seek to establish non-trivial approximation guarantees for algorithms solving the pooled tag array multiplexing problem.

In the second part of the thesis, we have proposed a new genotyping assay architecture combining multiplexed solution-phase single-base extension (SBE) reactions with sequencing by hybridization (SBH) using universal DNA arrays such as all k-mer arrays. Our contributions include a study of multiplexing algorithms for SBE/SBH genotyping assays and preliminary experimental results showing the achievable tradeoffs between the number of array probes and primer length on one hand and the number of SNPs that can be assayed simultaneously on the other. We proved that the problem of selecting a maximum size subset of SNPs that can be unambiguously genotyped in a single SBE/SBH assay is NP-hard, and proposed efficient heuristics with good practical performance. Our heuristics take into account the freedom of selecting primers from both strands of the genomic DNA. In addition, our heuristics can enforce user-specified redundancy constraints facilitating reliable genotyping in the presence of hybridization errors. Simulation results on datasets both randomly generated and extracted from the

NCBI dbSNP database suggest that the SBE/SBH architecture provides a flexible and cost-effective alternative to genotyping assays currently used in the industry, enabling genotyping of up to hundreds of thousands of user-specified SNPs per assay.

Establishing the complexity of MPPP and finding approximation algorithms for MPPP and MDPSP are interesting open problems.

Bibliography

[1] Affymetrix, Inc. GeneFlex Tag Array Technical Note No. 1, available online at http://www.affymetrix.com/support/technical/technotes/genflex_technote.pdf, 2001.

[2] Affymetrix, Inc. Custom and application-specific genotyping with the Affymetrix GeneChip MegAllele System, available online at http://www.affymetrix.com/support/technical/other/parallele_brochure.pdf, 2005.

[3] Affymetrix, Inc. GeneChip Human Mapping 500K Array Set, available online at http://www.affymetrix.com/support/technical/datasheets/500k_datasheet.pdf, 2005.

[4] A. Alderborn, A. Kristofferson, and U. Hammerling. Determination of single-nucleotide polymorphisms by realtime pyrophosphate DNA sequencing. In *Genome Research*, volume 10, pages 1249–1258, 2000.

[5] A. Ben-Dor, R. Karp, B. Schwikowski, and Z. Yakhini. Universal DNA tag systems: a combinatorial design scheme. *Journal of Computational Biology*, 7(3-4):503–519, 2000.

[6] A. BenDor, T. Hartman, B. Schwikowski, R. Sharan, and Z. Yakhini. Towards optimally multiplexed applications of universal DNA tag systems. In *Proc. 7th Annual International Conference on Research in Computational Molecular Biology*, pages 48–56, 2003.

[7] A. Braun, D.P. Little, and H. Koster. Detecting CFTR gene mutations by using primer oligo base extension and mass spectrometry. In *Clinical Chemistry*, volume 43, pages 1151–1158, 1997.

[8] S. Brenner. Methods for sorting polynucleotides using oligonucleotide tags. *US Patent 5,604,097*, 1997.

[9] International Human Genome Sequencing Consortium. Initial sequencing and analysis of the human genome. *Nature*, 409:860–921, 2001.

[10] International Human Genome Sequencing Consortium. Finishing the euchromatic sequence of the human genome. *Nature*, 431:931–945, 2004.

[11] B. DasGupta, K.M. Konwar, I.I. Mandoiu, and A.A. Shvartsman. Highly scalable algorithms for robust string barcoding. In *Proc. 2005 International Workshop on Bioinformatics Research and Applications (IWBRA'05)*, 2005.

[12] B. DasGupta, K.M. Konwar, I.I. Măndoiu, and A.A. Shvartsman. Highly scalable algorithms for robust string barcoding. *International Journal of Bioinformatics Research and Applications*, 1(2):145–161, 2005.

[13] R. Dramanac and R. Crkvenjakov. DNA sequencing by hybridization. *Yugoslav patent application*, 1987.

[14] T.G. Drummond, M.G. Hill, and J.K. Barton. Electrochemical DNA sensors. In *Nature Biotechnology*, volume 21, pages 1192–1199, 2003.

[15] W. Duckworth, D.F. Manlove, and M. Zito. On the approximability of the maximum induced matching problem. In *Journal of Discrete Algorithms*, volume 3, pages 79–91, 2005.

[16] W.E. Evans and H.L. McLeod. Pharmacogenomics - drug disposition, drug targets, and side effects. In *The New England Journal of Medicine*, volume 348, pages 538–549, 2003.

[17] N.P. Gerry, N.E. Witowski, J. Day, R.P. Hammer, G. Barany, and F. Barany. Universal DNA microarray method for multiplex detection of low abundance point mutations. *J. Mol. Biol.*, 292(2):251–262, 1999.

[18] S.A. Heath and F.P. Preparata. Enhanced sequence reconstruction with DNA microarray application. In *Proc. 7th Annual International Conference on Computing and Combinatorics (COCOON)*, pages 64–74, 2001.

[19] J.N. Hirschhorn, P. Sklar, K. Lindblad-Toh, Y.-M. Lim, M. Ruiz-Gutierrez, S. Bolk, B. Langhorst, S. Schaffner, E. Winchester, and E. Lander. SBE-TAGS: An array-based method for efficient single-nucleotide polymorphism genotyping. *PNAS*, 97(22):12164–12169, 2000.

[20] E. Hubbell. Multiplex sequencing by hybridization. *Journal of Computational Biology*, 8(2):141–149, 2001.

[21] N. Hundewale, I.I. Măndoiu, C. Prăjescu, and A. Zelikovsky. Integrated design flow for universal DNA tag arrays. In *9th Annual International Conference on Research in Computational Molecular Biology (RECOMB) Poster Book*, pages 141–142, 2005.

[22] A.B. Kahng, I.I. Mandoiu, S. Reda, X. Xu, , and A. Zelikovsky. Design flow enhancements for DNA arrays. In *Proc. IEEE International Conference on Computer Design (ICCD)*, pages 116–123, 2003.

[23] K.M. Konwar, I.I. Măndoiu, A.C. Russell, and A.A. Shvartsman. Improved algorithms for multiplex PCR primer set selection with amplification length constraints. In Y.-P. Phoebe Chen and L. Wong, editors, *Proc. 3rd Asia-Pacific Bioinformatics Conference (APBC)*, pages 41–50, London, 2005. Imperial College Press.

[24] P.Y. Kwok. Methods for genotyping single nucleotide polymorphisms. *Annual Review of Genomics and Human Genetics*, 2:235–258, 2001.

[25] V.V. Lozin. On maximum induced matchings in bipartite graphs. In *Infomation Processing Letters*, volume 81, pages 7–11, 2002.

[26] M.S. Morris, D.D. Shoemaker, R.W. Davis, and M.P. Mittmann. Selecting tag nucleic acids. *U.S. Patent 6,458,530 B1*, 2002.

[27] I.I. Măndoiu and C. Prăjescu. High-throughput SNP genotyping by SBE/SBH. ACM Computing Research Repository, cs.DS/0512052, 2005.

[28] I.I. Măndoiu, C. Prăjescu, and D. Trincă. Improved tag set design and multiplexing algorithms for universal arrays. *LNCS Transactions on Computational Systems Biology*, II(LNBI 3680):124–137, 2005.

[29] I.I. Măndoiu, C. Prăjescu, and D. Trincă. Improved tag set design and multiplexing algorithms for universal arrays. In *Proceedings of the 1st International Workshop on Bioinformatics Research and Applications (IWBRA)*, volume 3515, pages 994–1002. Lecture Notes in Computer Science, Springer-Verlag, 2005.

[30] I.I. Măndoiu and D. Trincă. Exact and approximation algorithms for DNA tag set design. ACM Computing Research Repository, 2005.

[31] F. Naef and M.O. Magnasco. Solving the riddle of the bright mismatches: Labeling and effective binding in oligonucleotide arrays. In *Physical Review E.*, volume 68, pages 11906–11910, 2003.

[32] NIH. dbSNP, http://www.ncbi.nlm.nih.gov/projects/snp/.

[33] NIH. GeneMark, http://opal.biology.gatech.edu/genemark.

[34] NIH. ORF Finder, http://www.ncbi.nih.gov/gorf/gorf.html.

[35] P. Nyren, B. Pettersson, and M. Uhlen. Solid phase DNA minisequencing by an enzymatic luminometric inorganic pyrophosphate detection assay. In *Analytical Biochemistry*, volume 208, pages 171–175, 1993.

[36] T. Pastinen, J. Partanen, and AC. Syvanen. Multiplex, fluorescent, solid-phase minisequencing for efficient screening of DNA sequence variation. In *Clinical Chemistry*, volume 42, pages 1391–1397, 1996.

[37] N. Patil et al. Blocks of limited haplotype diversity revealed by high-resolution scanning of human chromosome 21. *Science*, 294:1719–1723, 2001.

[38] P.A. Pevzner. *Computational Molecular Biology: An Algorithmic Approach*. MIT Press, 2000.

[39] P.A. Pevzner and R.J. Lipshutz. Towards DNA sequencing chips. In *Proc. 19th Int. Conf. on Mathematical Foundations of Computer Science*, pages 143–158, 1994.

[40] S. Rahmann. Rapid large-scale oligonucleotide selection for microarrays. In *Proc. IEEE Computer Society Bioinformatics Conference (CSB'02)*, 2002.

[41] M. Ronaghi, S. Karamohamed, B. Pettersson, M. Uhlen, and P. Nyren. Real-time DNA sequencing using detection of pyrophosphate release. In *Analytical Biochemistry*, volume 242, pages 84–89, 1996.

[42] P.L. Ross, K. Lee, and P. Belgrader. Discrimination of singlenucleotide polymorphisms in human DNA using peptide nucleic acid probes detected by MALDI-TOF mass spectrometry. In *Analytical Chemistry*, volume 69, pages 4197–4202, 1997.

[43] R. Sharan, J. Gramm, Z. Yakhini, and A. Ben-Dor. Multiplexing schemes for generic SNP genotyping assays. *Journal of Computational Biology*, 12(5):514–533, 2005.

[44] AC. Syvanen. Accessing genetic variation: genotyping single nucleotide polymorphisms. In *Nature Reviews Genetics*, volume 2, pages 930–942, 2001.

[45] N. Tonisson, A. Kurg, E. Lohmussaar, and A. Metspalu. Arrayed primer extension on the DNA chip - method and application. In Mark Schena, editor, *Microarray Biochip Technology*, pages 247–263. Eaton Publishing, 2000.

[46] R.B. Wallace, J. Shaffer, R.F. Murphy, J. Bonner, T. Hirose, and K. Itakura. Hybridization of synthetic oligodeoxyribonucleotides to phi chi 174 DNA: the effect of single base pair mismatch. *Nucleic Acids Res.*, 6(11):6353–6357, 1979.