

## Selecting Degenerate Multiplex PCR Primers<sup>\*</sup>

Richard Souvenir<sup>1</sup>, Jeremy Buhler<sup>1,2</sup>, Gary Stormo<sup>2,1</sup> and Weixiong Zhang<sup>1,2,\*\*\*</sup>

<sup>1</sup>Department of Computer Science and

<sup>2</sup>Department of Genetics

Washington University in St. Louis

St. Louis, MO 63130, USA

**Abstract.** Single Nucleotide Polymorphism (SNP) Genotyping is an important molecular genetics technique in the early stages of producing results that will be useful in the medical field. One of the proposed methods for performing SNP Genotyping requires amplifying regions of DNA surrounding a large number of SNP loci. In order to automate a portion of this method and make the use of SNP Genotyping more widespread, it is important to select a set of primers for the experiment. Selecting these primers can be formulated as the *Multiple Degenerate Primer Design (MDPD)* problem. An iterative beam-search algorithm, *Multiple, Iterative Primer Selector (MIPS)*, is presented for MDPD. Theoretical and experimental analyses show that this algorithm performs well compared to the limits of degenerate primer design and the number of spurious amplifications should be small. Furthermore, MIPS outperforms an existing algorithm which was designed for a related degenerate primer selection problem.

An implementation of the MIPS algorithm is available for research purposes from the website <http://www.cse.wustl.edu/~zhang/software/mips>.

### 1 Introduction

Single Nucleotide Polymorphisms (SNPs) are individual base differences in DNA sequences between individuals. It is estimated that there are roughly three million SNPs in the human genome [12]. Association studies between SNPs and various diseases, as well as differences in how individuals respond to common therapies, promise to revolutionize medical science in the coming years [2]. Recent work suggests there may be only a few hundred thousand "blocks" of SNPs that recombine to provide most of the variability seen in human populations [5]. However, it is still a daunting task to identify the specific genetic variations occurring in specific individuals in order to determine their associations with important phenotypes. Currently, there are many proposed techniques for determining the SNP composition of a given genome. However, in order for

\*\*\* Corresponding to: [zhang@cse.wustl.edu](mailto:zhang@cse.wustl.edu), phone: (314)935-8788.

\* We thank Pui Kwok for describing the problem and useful discussions. RS was supported by NIH training grant GM08802 to Washington University Medical School and NSF grant ITR/EIA-0113618, GS was funded by NIH grant HG00249, and WZ was supported by NSF grants IIS-0196057 and ITR/EIA-0113618.

these assaying techniques to be effective in large-scale genetic studies of hundreds or thousands of SNPs, they must be scalable, automated, robust, and inexpensive [9].

One technique involves the use of multiplex PCR (MP-PCR) to amplify the regions around the SNP. Multiplex PCR is a variation of PCR where multiple DNA fragments are replicated simultaneously. MP-PCR, like all PCR variations, makes use of oligonucleotide primers to define the boundaries of amplification. For each region of DNA that is to be amplified, two primers, generally referred to as the forward and reverse primers, are needed. In MP-PCR, it is necessary to select a forward and reverse primer for each of the regions to be replicated, and for the large-scale amplification required in SNP Genotyping, there can be hundreds, or perhaps thousands, of those regions. The process of selecting such a large set of primers by current methods, including trial-and-error [9], can be time-consuming and difficult.

There are two similar problems in primer selection, the Primer Selection Problem and the Degenerate Primer Design Problem. The Primer Selection Problem [15] involves minimizing the number of primers needed to amplify regions of DNA in a set of sequences. It has been shown that this is an NP-hard problem [6] in reductions from other hard problems, including SET-COVER and GRAPH-COLORING [3]. There have been a number of proposed heuristics to solve this problem, including a branch-and-bound search algorithm [14]. Also, algorithms have been proposed which incorporate biological data about the primers into the search [13, 4].

In an MP-PCR experiment where the number of primers is not optimized, the number of primers needed is equal to twice the number of sequences in the input set. In general, the algorithms mentioned above reduce the number of primers needed to 25-50% of this value, which can still be rather high for the large-scale amplification needed for SNP Genotyping. This leads to the use of degenerate primers. Degenerate primers [10] are primers that make use of degenerate nucleotides. For example, consider this degenerate primer,  $ACMCM$ , where  $M$  is a degenerate nucleotide which represents either of the bases,  $A$  or  $C$ . This degenerate primer is actually representative of the set of 4 primers  $\{ACACA, ACACC, ACCCA, ACCCC\}$ . The number of primers that a degenerate primer represents is referred to as its *degeneracy*. Degenerate primers are as easy to produce as regular primers, and therefore save the molecular biologist time during the primer design phase of the experiment. The use of degenerate primers introduces two new problems. First, the effective concentration of the desired primers is decreased by the presence of undesired primers. Second, the presence of undesired primers can lead to erroneous amplification. Therefore, it is important to use primers of relatively low degeneracy to realize the inherent benefits of degenerate primer design while minimizing the effects of these two problems.

The Degenerate Primer Design Problem (DPD) is the second related problem, however it makes use of degenerate primers. DPD is the decision problem of determining whether or not there exists a single degenerate primer below some given threshold which can amplify regions of DNA for some number of a set of input sequences. There are two variations of DPD. MAXIMUM COVERAGE DPD (MC-DPD) is the related maximization problem where the goal is to find the maximum number of sequences that can be amplified by a degenerate primer whose degeneracy falls below some threshold. MINIMUM DEGENERACY DPD (MD-DPD) is the second variation of DPD whose

goal is to find the degenerate primer of minimum degeneracy that amplifies all of the input sequences. Both MC-DPD and MD-DPD have been shown to be NP-Hard problems [11].

In this paper, we describe the Multiple Degenerate Primer Design Problem (MDPD), present an algorithm to solve this problem and describe how the results can be used to select a large set of primers that can be used in MP-PCR for SNP Genotyping. The details of the protocol for applying degenerate primers for genotyping using SNPs can be found in [9]. The basic problem uses a given collection of DNA sequences from genomic regions known to contain SNPs. The regions are chosen such that primers selected from them, one on each side, are not closer to the SNP than a fixed amount and no farther away than a specified distance. We proceed as follows: We first present two variants of MDPD problems. We then describe the *Multiple, Iterative Primer Selector (MIPS)* algorithm and show how MIPS performs relative to another solution in the domain, and the theoretical limits of the problem. We also discuss the issue of erroneous amplification. We specifically study the relationship among the probability of unexpected priming events, the length of degenerate primers and their degeneracy. Finally, we show the results of MIPS on different datasets.

## 2 Problem Description

Some of the notation from [11] is used to describe the MDPD problem. To maintain consistency, lower-case symbols (i.e.  $l, b, i$ ) represent numerical values, counting variables, or individual characters in a sequence. Upper-case symbols (i.e.  $P, S$ ) denote primers, sequences, or subsequences. Finally, calligraphy symbols (i.e.  $\mathcal{S}, \mathcal{C}$ ) represent sets of sequences or primers.

Let  $\Sigma = \{A, C, G, T\}$  which is the finite fixed alphabet of DNA. A *degenerate primer* is a string  $P$  with several possible characters at each position, i.e.,  $P = p_1 p_2 \cdots p_l$ , where  $p_i \subseteq \Sigma, p_i \neq \emptyset$ .  $l$  is the length of primer  $P$ . The *degeneracy* of  $P$  is  $d(P) = \prod_{i=1}^l |p_i|$ . Consider the degenerate primer  $P' = \{A\}\{A, C\}\{A, C\}\{C\}$ . The length of  $P'$  is 4 and  $d(P') = 4$ . For the sake of clarity, we use the IUPAC symbols for degenerate nucleotides to represent degenerate primers. Therefore,  $P'$  can be represented as  $AMMC$  where  $M$  is the degenerate nucleotide which represents  $\{A, C\}$ . Degenerate primers can be constructed by *primer addition*. For any two primers,  $P^1$  and  $P^2$ , their sum,  $P^3$  equals  $(p_1^1 \cup p_1^2)(p_2^1 \cup p_2^2) \cdots (p_l^1 \cup p_l^2)$ .

For any sequence  $S_i$  in an input set  $\mathcal{S}$ , we say that a degenerate primer  $P$  *covers*  $S_i$  if there is a substring  $F$  of length  $l$  in  $S_i$  where for each character  $f_i$  in  $F$ ,  $f_i \in p_i$ .

There are two variants of the MULTIPLE DEGENERATE PRIMER DESIGN problem: PRIMER-THRESHOLD MDPD (PT-MDPD) and TOTAL-THRESHOLD MDPD (TT-MDPD). For both, we are given  $n$  sequences,  $\mathcal{S} = \{S_1, S_2, \cdots, S_n\}$ , and a maximum degeneracy bound  $\alpha$ . For PT-MDPD, the goal is to find a set of degenerate primers,  $\mathcal{P}$ , of minimum size that covers every sequence in  $\mathcal{S}$  where, for each degenerate primer,  $P_i \in \mathcal{P}$ ,  $d(P_i) \leq \alpha$ . For TT-MDPD, the goal is to find a set of degenerate primers,  $\mathcal{P}$ , of minimum size that covers every sequence in  $\mathcal{S}$  where  $\sum d(P_i) \leq \alpha$ .

Both PT-MDPD and TT-MDPD are NP-hard [6]. The NP-hardness of PT-MDPD can be shown based on the observation that the Primer Selection Problem (PSP) [15]

is a special case of PT-MDPD, where the degeneracy threshold is set to one. The NP-hardness of TT-MDPD can be shown by a reduction from the Weighted Set Covering problem.

### 3 MIPS: Multiple, Iterative Primer Selector

To overcome the difficulty caused by the NP-hardness of MDPD problems, we propose an iterative beam search algorithm to make a tradeoff between optimality and tractability. In order to solve PT-MDPD and TT-MDPD, MIPS can run in either of two modes, MIPS-PT and MIPS-TT, respectively. This section focuses on MIPS-TT. However, we will highlight how MIPS-PT operates differently.

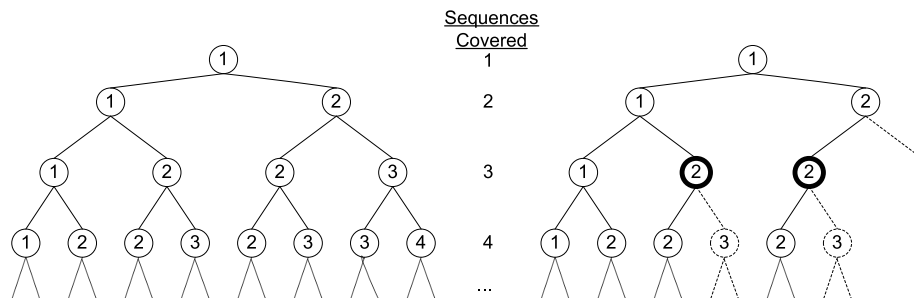
MIPS progressively constructs a set of primers that covers all the input sequences. Define a  $k$ -*primer* to be a degenerate primer that covers  $k$  input sequences. The basic algorithm first generates a set of candidate 2-primers, each having some degeneracy value, then iteratively extends all candidate  $k$ -primers into  $(k + 1)$ -primers by generalizing them to cover an additional sequence. Generalization stops when no primer can be extended without exceeding the degeneracy threshold  $\alpha$ . At this point, all remaining primers cover  $k_{last}$  sequences, so we retain the primer of minimum degeneracy, remove the input sequences it covers from consideration, and repeat the algorithm until all sequences are covered.

To guide the search, MIPS uses the degeneracy of a primer as a scoring function. The set of primers that are stored for extension are known as a *beam*. Beam Search [1] differs from greedy or best-first search in that multiple nodes, degenerate primers in this case, are saved for extension instead of just one. This model of progressively adding to a beam of degenerate primers and updating the scoring function is similar to the CONSENSUS motif-finding model [8].

It is important to note that the degeneracy of a given  $k$ -primer increases or remains the same, with the addition of additional sequence fragments. This observation permits us to employ a strategy which ignores degenerate primers with high degeneracy, in order to speed up the algorithm. Therefore, the search is restricted only to the primers with the lowest degeneracy. In this algorithm, the number of the candidate primers to restrict the search to at each level can be specified. This constant,  $b$ , describes the number of  $k$ -primers to save for each level. Increasing  $b$  can possibly improve the quality of the solution, but lengthens the running time of the algorithm. In the results section, we examine the effect of this parameter,  $b$ , on MIPS.

The pairwise comparison of two fragments is the dominating operation and a rate-limiting step of the algorithm. A majority of these comparisons are between two fragments that share few, if any, nucleotides. To avoid comparisons between dissimilar fragments, the exhaustive pairwise comparison is replaced with a similarity lookup. All of the primer candidates are added to a FASTA-style lookup table. In general, for DNA, a FASTA table fragment length of 6 is recommended [7]. Using the table, each fragment is compared only to the other fragments that are returned.

The constructive search continues until one of two cases occurs. In the first case, all sequences are covered by a single  $n$ -primer, where  $n$  is the number of sequences in the input set. The algorithm then terminates with that primer as the result. In the second



**Fig. 1.** For these graphs, the depth of a node represents the number of sequences from the input set covered and the number in a node represents the number of degenerate primers that will be used to cover those sequences. Each node can be expanded into two child nodes. The left child represents covering an additional sequence using an existing degenerate primer and the right child represents covering an additional sequence using a new degenerate primer. Graph (a) shows a full search. Graph (b) shows the pruning that takes place in MIPS-TT during the backtracking phase. Consider the two bold nodes. Both of these cover the same number of sequences with the same number of primers. MIPS-TT will therefore only expand the node whose total score is better. This avoids the exponential expansion seen in (a)

case, no  $k$ -primer can be extended to a  $(k + 1)$ -primer without exceeding the degeneracy threshold and there exists at least one sequence uncovered. At this point,  $k_{last}$  sequences have been covered. The algorithm chooses the best degenerate ( $k_{last}$ )-primer,  $P_0$ , from the set  $\mathcal{P}$  of primers sorted by degeneracy value. The problem then reduces to a smaller instance where the input set is the original set of sequences minus those covered by  $P_0$ . In MIPS-PT, the degeneracy threshold for this subproblem is equivalent to the original threshold,  $\alpha$ . In MIPS-TT, the degeneracy threshold is reduced by the degeneracy of  $P_0$ . *The algorithm then restarts on the reduced problem.*

For MIPS-PT, iteratively applying this procedure will eventually return a set of primers to cover the set of input sequences. However, this is not the case for MIPS-TT. After  $P_0$  is discovered and its sequences are removed from consideration, the new threshold may be too low to cover the rest of the sequences. In this case, MIPS-TT *backtracks* to the previous level,  $k_{last} - 1$ , and selects the next best primer  $P'_0$  for removal. Again, MIPS restarts on the sequences that  $P'_0$  has not covered and with a degeneracy limit that is the original  $\alpha$  minus the degeneracy of  $P'_0$ . Figure 1 shows, schematically, the execution of MIPS-TT.

A pseudo-code description of MIPS is given in Algorithms 1-3.

## 4 Analysis

We now examine the theoretical bounds of MIPS and degenerate primer design in general, and investigate the issue of erroneous amplification. In the following, let  $n$  be the number of sequences in the input set,  $m$  the average length of each sequence,  $b$  the beam size, and  $l$  the length of the primers.

---

**Algorithm 1** MIPS( $\mathcal{S}, \alpha$ )

---

- 1: *Initialize Global variables* (2-D matrices): *BEST* - candidate fragments; *COVERED* - sequences covered; *ALLOWABLE* - remaining degeneracy,  $ALLOWABLE(0, 0) = \alpha$ .
  - 2: **for**  $p = 1$  to the number of degenerate primers that will be used **do**
  - 3:   Let  $c =$  the maximum number of sequences that the ( $p-1$ ) primers covered
  - 4:   **while**  $c > 0$  **do**
  - 5:     MIPS\_SEARCH( $\mathcal{S} - COVERED(p - 1, c), ALLOWABLE(p - 1, c), p, c$ )
  - 6:     if this search covers  $\mathcal{S}$ , print solution and exit
  - 7:     else  $c=c-1$
- 

---

**Algorithm 2** MIPS\_SEARCH( $\mathcal{S}, \alpha, p, c$ )

---

- 1: **Input:** Sequence set  $\mathcal{S}$ , degeneracy bound  $\alpha$ , primer number  $p$ , sequences covered  $c$ .
  - 2: **Output:** total number of sequences covered
  - 3: Initialize priority queue  $Q$  of size  $b$ ;
  - 4: Perform pair-wise comparisons.
  - 5: **for all** sequence  $S_i \in \mathcal{S}$  **do**
  - 6:   **for all** substring  $S_i[j, l]$  **do**
  - 7:     Let  $\mathcal{C} = \{ x | \langle f, x \rangle \in T \text{ and } f \text{ is a } k\text{-length substring of } S_i[j, l] \}$
  - 8:     **for all** fragment  $C_k \in \mathcal{C}$  **do**
  - 9:        $D = S_i[j, L] + C_k$
  - 10:      Insert  $D$  into queue  $Q$
  - 11: Let  $c' = c$
  - 12: **while** queue  $Q$  is not empty **do**
  - 13:   Let  $P =$  the best element of  $Q$
  - 14:   **if** degeneracy( $P$ ) < degeneracy( $BEST(p, c)$ ) **then**
  - 15:      $BEST(p, c') = P$
  - 16:      $ALLOWABLE(p, c') = \alpha - \text{degeneracy}(P)$
  - 17:      $COVERED(p, c') = COVERED(p - 1, c) \cup \text{covers}(P, \mathcal{S})$
  - 18:      $Q = ONE\_PASS(Q, \mathcal{S}, \alpha)$
  - 19:      $c' = c' + 1$
  - 20: return ( $c' + 1$ )
- 

---

**Algorithm 3** ONE\_PASS( $Q, \mathcal{S}, \alpha$ )

---

- 1: **Input:** Priority queue  $Q$ , set of sequences  $\mathcal{S}$ , degeneracy bound  $\alpha$ .
  - 2: **Output:** Priority queue  $Q'$
  - 3: **for all** primer  $P \in Q$  **do**
  - 4:   **for all** sequence  $S_i \in \mathcal{S}$  **do**
  - 5:     **if**  $S_i \notin \text{covers}(P)$  **then**
  - 6:       **for all** substring  $S_i[j, l]$  **do**
  - 7:          $D = S_i[j, l] + P$
  - 8:         Insert  $D$  into queue  $Q'$
  - 9: return  $Q'$
-

#### 4.1 Algorithm complexity

**Space** From the input set, each primer is stored individually which requires space  $O(nml)$ . In the implementation, there are  $4n \times n$  matrices that are needed for backtracking and storing degenerate primers that could eventually become part of the final solution. This adds an additional  $O(n^2)$  of storage. Therefore, the total amount of space is  $O(n^2 + nml)$ .

**Time** The time complexity is analyzed in a bottom-up fashion. The procedure of comparing the fragments in the beam to the remaining sequences makes  $O(bnm)$  primer additions since there are  $O(nm)$  total fragments and  $b$  fragments in the beam. Each primer addition requires comparing every character in each of the two primer. Therefore, this portion requires  $O(bnml)$  time.

The process of generating new beams of  $k$ -primers, for increasing  $k$ , is called MIPS\_SEARCH. MIPS\_SEARCH uses the above procedure to build new beams, and could, in the worst case, build  $n$  beams. Therefore, the overall time complexity is  $O(bn^2ml)$ . The number of times MIPS\_SEARCH is executed depends on the amount of backtracking. This is directly related to the number of primers in the final solution. In the best case, if the solution only requires 1 primer, there will be only one call to MIPS\_SEARCH. In the worst case, if the solution requires  $n$  primers (one primer for each input sequence) there will be  $n^2/2$  calls to MIPS\_SEARCH. Let  $p$  be the number of primers in the final solution. The best approximation to the number of MIPS\_SEARCH calls is  $O(pn)$ . This brings the overall time complexity to  $O(bn^3mlp)$ . Currently, we are working on a method to reduce the time complexity of comparing the degenerate primers in the beam to the remaining sequences in order to speed up the entire algorithm.

#### 4.2 Limits of degenerate primer design

Multiplex primer design demands that many input sequences share sites complementary to some common (possibly degenerate) primer. The sequences to be co-amplified are not in general homologous, so their complementarity to a common primer is largely a matter of chance. We therefore explored the chance-imposed limits of multiplexing, that is, how many unrelated DNA sequences are likely to be covered by a single PCR primer of a given degeneracy?

Let  $\mathcal{S}$  be a collection of  $n$  DNA sequences of common length  $m$ . Call a primer  $P$  an  $(l, \alpha, k)$ -primer for  $\mathcal{S}$  if it has length  $l$  and degeneracy at most  $\alpha$  and covers at least  $k$  sequences of  $\mathcal{S}$ . A natural way to quantify the limits of multiplexing is to compute the probability that an  $(l, \alpha, k)$ -primer exists for  $\mathcal{S}$ . However, this probability is difficult to compute, even assuming that  $\mathcal{S}$  consists of i.i.d. random DNA with equal base frequencies. We instead compute the *expected* number of  $(l, \alpha, k)$ -primers for  $\mathcal{S}$ . If this expectation is much less than one, Markov's inequality implies that  $\mathcal{S}$  is unlikely to contain any such primer.

We count not the total number of  $(l, \alpha, k)$ -primers for  $\mathcal{S}$  but only the number of *maximal* primers. A primer  $P$  of degeneracy at most  $\alpha$  is said to be *maximal* if increasing  $P$ 's degeneracy at any position would cause its total degeneracy to exceed  $\alpha$ . The

expected number of maximal  $(l, \alpha, k)$ -primers for  $\mathcal{S}$  is in general less than the total number of  $(l, \alpha, k)$ -primers, but a primer of this type exists for  $\mathcal{S}$  iff a maximal primer exists. Hence, the former expectation is more useful than the latter for bounding the probability that at least one  $(l, \alpha, k)$ -primer exists.

**Occurrence probability for one fixed primer** Let  $P$  be a primer of length  $l$ , such that the  $j$ th position of  $P$  permits  $|p_j|$  different bases. Let  $\mathcal{S}$  be a collection of  $n$  i.i.d. random DNA sequences of common length  $m$  with equal base frequencies, and let  $T$  be a single  $l$ -mer at a fixed position in some sequence  $S_i \in \mathcal{S}$ . Say that  $P$  matches  $T$  if  $P$  would hybridize to  $T$ . We have that

$$\begin{aligned} \Pr[P \text{ matches } T] &= \prod_{j=1}^l \frac{|p_j|}{4} \\ &= \frac{d(P)}{4^l}. \end{aligned}$$

The probability that  $P$  covers  $S_i$ , i.e. that it matches *at least one*  $l$ -mer of  $S_i$ , depends in a complicated way on  $P$ 's overlap structure, but if  $S_i$  is not too short and  $d(P)/4^l \ll 1$  (both of which are typically true), then using Poisson approximation [16],

$$\Pr[P \text{ occurs in } S_i] \approx 1 - e^{-\frac{d(P)}{4^l}(m-l+1)}.$$

Let  $q$  be the probability that  $P$  matches somewhere in a single sequence of length  $m$ , and let  $c(P)$  be  $P$ 's coverage of  $\mathcal{S}$ , i.e. the number of sequences of  $\mathcal{S}$  in which  $P$  matches at some position. Because the sequences of  $\mathcal{S}$  are independent, the probability that  $P$  matches in at least  $k$  sequences given by the binomial tail probability

$$\Pr[c(P) \geq k] = 1 - \Pr[B(n, q) < k],$$

where  $B(n, q)$  is the sum of  $n$  independent Bernoulli random variables, each with probability  $q$  of success.

**Computing the expectation** Let  $\Pi(l, \alpha)$  be the set of all maximal primers of length  $l$  and degeneracy at most  $\alpha$ . To count the expected number  $E_{l, \alpha, k}$  of  $(l, \alpha, k)$ -primers for  $\mathcal{S}$ , observe that

$$E_{l, \alpha, k} = \sum_{P \in \Pi(l, \alpha)} \Pr[c(P) \geq k].$$

Enumerating all  $P \in \Pi(l, \alpha)$  to compute this expectation would be computationally expensive, but this enumeration is not needed for i.i.d. sequences with equal base frequencies. Given these assumptions about  $\mathcal{S}$ 's sequences, the probability that  $P$  matches a given  $l$ -mer does not change if we rearrange its positions (e.g. “*AMC*” versus “*MCA*”) or change the precise nucleotides matched (e.g. “*RTG*” versus “*MCA*”). Let  $W$  be a multiset of  $l$  values drawn from  $\{1, 2, 3, 4\}$  that lists the degeneracies  $n_j$  (in any order) of a primer from  $\Pi(l, \alpha)$ . Then every primer described by the same  $W$  has the same

probability of covering at least  $k$  sequences in  $\mathcal{S}$ . Hence, the desired expectation is given by

$$E_{l,\alpha,k} = \sum_W \#(W) \Pr[c(P) \geq k \mid P \text{ described by } W].$$

where the sum ranges over all feasible  $W$  for  $\Pi(l, \alpha)$  and  $\#(W)$  denotes the number of degenerate primers described by  $W$ . The probability is computed as described above, so we need only describe how to compute  $\#(W)$ .

Let  $W$  be a multiset with  $n_1$  1's,  $n_2$  2's,  $n_3$  3's, and  $n_4$  4's. If we fix *which* positions in  $P$  permit 1, 2, 3, and 4 nucleotides respectively, then there are  $4^{n_1} \times 6^{n_2} \times 4^{n_3}$  ways of assigning nucleotide sets to these positions. Hence,

$$\#(W) = \binom{l}{n_1 \ n_2 \ n_3} 4^{n_1+n_3} 6^{n_2}.$$

Enumerating all feasible  $W$  for  $\Pi(l, \alpha)$  is straightforward, so the expectation can be computed.

### 4.3 Mispriming

It is possible that a pair of primers binds to an undesired location and results in an erroneous amplification. *Mispriming* is the occurrence of this event where the unwanted PCR product is indistinguishable, by size, from the targeted products.

Suppose we design a set of degenerate primers with length  $l$ , such that the *total degeneracy* of the set is  $\alpha$ . We wish to estimate the expected number of mispriming events when our primer set is applied to a genome of length  $g$ . For simplicity, we assume that the genome is an i.i.d. random DNA sequence with equal base frequencies, and that a pair of  $l$ -mers cause a mispriming event iff they bind to the genome within  $\delta$  bases of each other, in the appropriate orientations to permit amplification of the sequence between them.

Let  $i$  index the positions of the genome on its forward strand. Let the 0-1 random variable  $x_i$  indicate the event that an  $l$ -mer from our primer set is complementary to the forward strand at position  $i$ , and let  $\bar{x}_i$  be the event that an  $l$ -mer is complementary to the reverse-complement strand at  $i$ . For any  $i$ , we have that

$$E[x_i] = E[\bar{x}_i] = \frac{\alpha}{4^l}.$$

We say that a mispriming event occurs at  $i$  if

$$\bar{x}_i \cap \bigcup_{j=i}^{i+\delta-1} x_j = 1.$$

Denote this event by the 0-1 indicator  $M_i$ . The total number of mispriming events  $M$  is simply  $\sum M_i$ , for  $i = 1, 2, \dots, g$ .

Note that the two matching events are independent in an i.i.d. random DNA sequence when the two primers do not overlap. To simplify our calculations, we ignore

the effect of overlapping primer boundaries. Using Poisson approximation to estimate the probability of the matching event on the forward strand, we have that

$$\begin{aligned} E[M_i] &= E[\bar{x}_i \cap \bigcup_{j=i}^{i+\delta-1} x_j] \\ &= E[\bar{x}_i] E\left[\bigcup_{j=i}^{i+\delta-1} x_j\right] \\ &\approx E[\bar{x}_i] \left(1 - e^{-\sum_{j=i}^{i+\delta-1} E[x_j]}\right). \end{aligned}$$

Finally, setting  $\rho = \alpha/4^l$ , we derive the expected mispriming rate as

$$\begin{aligned} E[M] &= \sum_{i=1}^g E[M_i] \\ &\approx g\rho (1 - e^{-\delta\rho}). \end{aligned}$$

Using  $g = 3 \times 10^9$  for the human genome and  $\delta = 500$  bases, we find that a design using 50 degenerate primers of length 20 and average degeneracy 10000 yields about 0.31 expected mispriming events in the genome. The mispriming rate scales linearly with the genome size and roughly quadratically with  $\rho$ .

## 5 Results

MIPS has been applied to both human DNA sequences and randomly generated datasets. We used a dataset containing regions of human DNA sequences surrounding 95 known SNPs. The sequences varied in length from a few hundred nucleotides to well over one thousand. The location of a SNP on a sequence was marked in order to provide a reference for the forward and reverse primers. To ensure effective PCR product analysis, each primer could not be located within 10 bases of the SNP and the entire PCR product length could not exceed 400 bases.

First, we show how MIPS performed relative to the theoretical limits previously discussed. Then, we show how various parameters, such as the beam size and degeneracy threshold, affect the performance, including the number of primers and running time. We then show some results of MIPS on the human dataset. Finally, we compare MIPS to an algorithm designed to solve a similar DPD problem considered in [11].

### 5.1 Comparison to theoretical limits

The theoretical estimates of section 4.2 can be used to evaluate whether a particular primer-design algorithm performs well on the MC-DPD problem, that is, whether it finds degenerate primers with coverage close to the maximum predicted for a given set of input sequences. We evaluated the MIPS algorithm's performance on MC-DPD by comparing the primers it found in random DNA with those expected to exist in theory.

degeneracy $\alpha$	Avg Coverage	Max Predicted
1000	6.30	7
10000	10.55	12
100000	19.30	26

**Table 1.** Actual and predicted coverage of 20-mer primers found on sets of 190 random sequences of length 211. Avg Coverage: average coverage of primer found over 20 random trials. Max Predicted: largest coverage  $m$  such that  $E_{20,\alpha,m} > 1$ .

For these experiments, we generated test sets of i.i.d. random DNA sequences with equal base frequencies with  $n = 190$ , and  $m = 211$ , so that the number and average length of the test sequences roughly matched those of the human DNA test sequences.

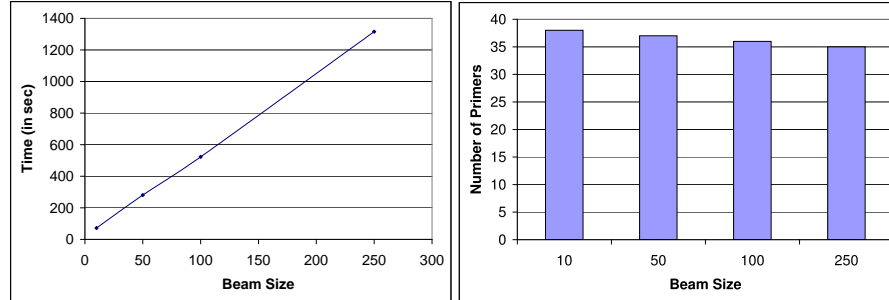
We used MIPS to find a single primer of length  $l = 15$  with maximum coverage in each test set, subject to varying degeneracy bounds  $\alpha$ . Table 1 compares the average coverage of primers found by MIPS in 20 trials to the largest coverage  $k$  such that  $E_{l,\alpha,k}$  for test sets of the specified size is  $> 1$ . Primers with coverage exceeding this value of  $k$  are not expected to occur in the test sets, while primers with slightly smaller coverage may or may not occur frequently.

MIPS proved adept at finding primers close to the maximum predicted coverage for relatively small degeneracies ( $\alpha \leq 10000$ ). We therefore have considerable confidence in its ability to find high-coverage primers if they are present. The gap between the best primers found by MIPS and those predicted to occur in theory grows with the degeneracy bound, but we cannot say with certainty whether this fact represents a limitation of the algorithm or of the theoretical estimates, since primers with expectation greater than one may with significant probability still fail to occur. Moreover, the high degeneracies where MIPS might perform poorly are of less practical interest, since single primers with such high degeneracies are experimentally more difficult to work with.

Overall, MIPS appears to be operating close to the theoretical limit for MC-DPD problems of small degeneracy. Although our analysis does not directly address the MDPD problems, any large gap between the most efficient design and the designs produced by MIPS is unlikely to arise from failure to find single high-coverage primers when they exist.

## 5.2 Performance

Figure 2 shows the effect of beam size on the solution quality, or number of primers. Figure 2a shows that increasing the beam size linearly increases the running time of the algorithm. These figures show the trade-off between the quality of the solution and the running time of the algorithm. For this particular dataset, there was a decrease of two degenerate primers in the final solution when the beam size was increased from 10 to 100. Moreover, only a slightly better solution was discovered when the beam size was increased to 250. For the average desktop computer, beam sizes larger than a few hundred result in impractical running times. For the input set we used, which contained 95 human DNA sequences, using a beam size of 100 produced a strong answer while keeping the running time reasonable. In general, the beam size should be close to the number of sequences in the input set.



**Fig. 2.** These graphs shows the effect of beam size on the (a) running time of the algorithm and (b) number of length 20 primers discovered. The algorithm was run on a dataset of 95 sequences which are regions surrounding known SNPs.

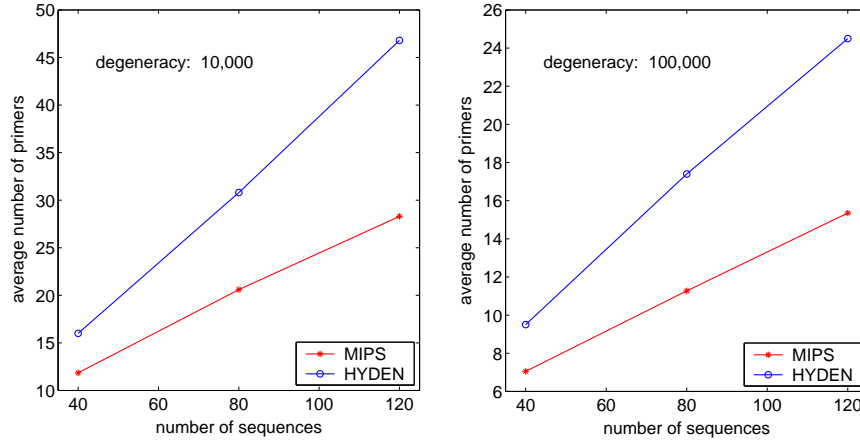
<i>PT-MDPD</i>		<i>TT-MDPD</i>	
<i>Degeneracy</i>	<i># Primers</i>	<i>Degeneracy</i>	<i># Primers</i>
$4^6 \approx 4K$	53	$4^9 \approx 262K$	44
$4^7 \approx 16K$	44	$4^{10} \approx 1M$	37
$4^8 \approx 64K$	36	$4^{11} \approx 4M$	30
$4^9 \approx 262K$	29	$4^{12} \approx 16M$	23

**Table 2.** Results on a dataset of 95 human SNP regions using primers of length 20 with default settings.

In an unpublished laboratory experiment, a set of degenerate primers of length 20 was manually constructed where each primer was a mixture of 8 specific bases and 12 fully degenerate nucleotides (*e.g.* *AGTCGGTANNNNNNNNNNNN*.) For this experiment, the total degeneracy would be  $\approx 4^{12}$ . MIPS was designed to automate this procedure and, possibly, reduce the total degeneracy and/or number of primers used. In practice the desired accuracy in the experiment determines the actual parameter values used for MIPS. Table 2 shows the results a large-scale MP-PCR experiment using primers of length 20. For 95 sequences, 190 primers would be needed in the general case. MIPS-PT decreased the total number of primers to 15% of this unoptimized value for a degeneracy limit of 262,144. Table 2 includes the similar results for PT-MDPD and TT-MDPD.

### 5.3 Comparison to HYDEN

The HYDEN algorithm [11] is a heuristic designed for finding approximate solutions to the DPD problems. Recall that DPD is a set of problems where the general goal is to find a *single* degenerate primer that either covers the most sequences while having a degeneracy value less than a specified threshold or covers all of the sequences with minimum degeneracy. The DPD problem is the most closely related one to our MDPD problem, and HYDEN is the only published algorithm for DPD that we are aware of.



**Fig. 3.** Both HYDEN and MIPS were tested on 20 randomly-generated datasets in order to solve the PT-MDPD problem. The tests were conducted with different degeneracy thresholds (a) 10,000 and (b) 100,000. The graph shows the number of degenerate primers selected.

HYDEN can solve the PT-MDPD problem indirectly by iteratively solving the MC-DPD problem on smaller and smaller sets. After selecting a pair of degenerate primers under a given bound that covers a certain subset of the sequences in an input set, the algorithm runs again on the remaining sequences. For the reasons described below, iteratively solving MC-DPD is not the most effective way to solve the PT-MDPD problem. However, this was the most reasonable comparison that was possible given the implementation available to us at the time of testing. The graphs in Figure 3 shows the number of primers that each algorithm found from a randomly generated set of sequences of varying lengths with varying degeneracy thresholds. They are uniformly-distributed i.i.d. sequences of equal length. Each program searched for degenerate primers of length 15 without allowing any mismatches at any positions.

In general, HYDEN always produced more primers than MIPS in attempting to solve PT-MDPD. For a primer degeneracy value of 100,000 and over 100 sequences, the difference was as large as 60% more primers. These results can be partially explained by the differing design requirements of the DPD and MDPD problems. Even when applied iteratively, the goal of the DPD problems is to have a result which could be divided into distinct PCR experiments. The goal of the MDPD problems is to have a set of primers for one large-scale PCR experiment. Specifically, to solve the DPD problem, the HYDEN algorithm must ensure that for any given degenerate forward primer that is discovered, exactly one degenerate reverse primer is used to cover the sequences covered by the forward primer. Therefore, a given degenerate forward primer is restricted to which sequences it is reported to cover based on the presence of a suitable degenerate reverse primer, and vice-versa. Moreover, the HYDEN algorithm has an additional restriction in which any given degenerate primer is limited to either covering a set of forward or reverse primers, but not both.

## 6 Conclusions

We have discussed a problem that arises in large-scale, high-throughput multiplex PCR experiments for SNP Genotyping. We developed an iterative beam-search heuristic, MIPS, for this problem which can be used to select a set of degenerate primers for a given set of sequences. This algorithm compares favorably to an existing algorithm for similar problems. Finally, using both theoretical calculations and experimental analysis, we have shown that MIPS is neither time nor memory intensive and could conceivably be used as a desktop tool for SNP Genotyping. The overall effectiveness of this algorithm will ultimately be determined by the application of the resulting primers in biological experiments, which is our next focus for this research.

## References

1. R. Bisiani. Search, beam. In S. C. Shapiro, editor, *Encyclopedia of Artificial Intelligence*, pages 1467–1468. Wiley-Interscience, New York, NY, 2nd edition, 1992.
2. F. Collins and V. McKusick. Implications of the human genome project for medical science. *JAMA*, 285:2447–2448, 2001.
3. K. Doi and H. Imai. A greedy algorithm for minimizing the number of primers in multiple pcr experiments. *Genome Informatics*, pages 73–82, 1999.
4. K. Doi and H. Imai. Complexity properties of the primer selection problem for pcr experiments. In *Proceedings of the 5th Japan-Korea Joint Workshop on Algorithms and Computation*, pages 152–159, 2000.
5. S. Gabriel, S. Schaffner, H. Nguyen, J. Moore, J. Roy, B. Blumenstiel, J. Higgins, M. DeFelice, A. Lochner, M. Faggart, S. Liu-Cordero, C. Rotimi, A. Adeyemo, R. Cooper, R. Ward, E. Lander, M. Daly, and D. Altshuler. The structure of haplotype blocks in the human genome. *Science*, 296:2225–2229, 2002.
6. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. Freeman, New York, NY, 1979.
7. D. Gusfield. *Algorithms on Strings, Trees, and Sequences: Computer Science and Computational Biology*, chapter 15, page 377. Press Syndicate of the University of Cambridge, 1997.
8. G. Hertz and G. Stormo. Identifying dna and protein patterns with statistically significant alignments of multiple sequences. *Bioinformatics*, 15:563–577, 1999.
9. P. Kwok. Methods for genotyping single nucleotide polymorphisms. *Annu. Rev. Genomics Human Genetics*, 2:235–58, 2001.
10. S. Kwok, S. Chang, J. Sninsky, and A. Wang. A guide to the design and use of mismatched and degenerate primers. *PCR Methods and Appl.*, 3:S39–S47, 1994.
11. C. Linhart and R. Shamir. The degenerate primer design problem. *Bioinformatics*, 18, Suppl. 1:S172–S180, 2002.
12. G. Marth, R. Yeh, M. Minton, R. Donaldson, Q. Li, S. Duan, R. Davenport, R. Miller, and P. Kwok. Single-nucleotide polymorphisms in the public domain: how useful are they? *Nature Genetics*, 27, 2001.
13. P. Nicodeme and J.-M. Steyaert. Selecting optimal oligonucleotide primers for multiplex PCR. In *Proceedings of Fifth Conference on Intelligent Systems for Molecular Biology ISMB97*, pages 210–213, 1997.
14. W. Pearson, G. Robins, D. Wrege, and T. Zhang. A new approach to primer selection problem in polymerase chain reaction experiments. In *Third International Conference on Intelligent Systems for Molecular Biology*, pages 285–291. AAAI Press, 1995.

15. W. Pearson, G. Robins, D. Wrege, and T. Zhang. On the primer selection problem in polymerase chain reaction experiments. *Discrete Applied Mathematics*, 71, 1996.
16. M. S. Waterman. *Introduction to Computational Biology*. Chapman & Hall, 1995.