Faster Greedy Algorithms for Multiple Degenerate Primer Selection

Sudha Balla, Sanguthevar Rajasekaran and Ion I. Măndoiu

Abstract— **To achieve increased throughput, amplification of multiple DNA sequences is often performed using a single Polymerase Chain Reaction (PCR) called Multiplex PCR (MP-PCR). Successful MP-PCR requires efficient methods for selecting sets of synthetic oligonucleotides called primers that collectively amplify all DNA loci of interest. Since the potential for forming primer-dimer pairs and unintended amplification products increases with the number of primers, a common optimization objective is to minimize the number of primers required to amplify all targets. Significant reductions in the number of required primers can be achieved by using primers with degenerate bases, or degenerate primers. The problem of selecting the minimum number of degenerate primers that amplify a given set of loci, referred to as the Multiple Degenerate Primer Selection Problem (MDPSP) has received much attention from researchers in the past few years. Since several variants of the problem have been proved to be NP-Complete, research has focused on heuristic algorithms that perform well on real biological data. In this paper, we present two new greedy algorithms for MDPSP, analyze their time and space complexities and compare their performance on random and real biological data with that of two previously reported algorithms. Our results show that the execution time and memory requirement of proposed algorithms is less than of existing algorithms, thus enabling the processing of larger input sets. Also, the new algorithms eliminate the dependency of the previous algorithms on an empirical input parameter that affects the runtime and quality of output. The software is downloadable at** http://www.engr.uconn.edu/~sub02005/software.html

I. INTRODUCTION

PCR is a fundamental technique in molecular biology used to amplify a given double-stranded DNA molecule into an exponential number of copies. The PCR reaction requires two synthetic oligonucleotides, called forward and reverse *primers*, typically 15-30 nucleotides in length, that are essentially substrings of the 5'-3' sequence upstream of the desired amplification locus on each of the two DNA strands. MP-PCR [3] is an advanced technique used to amplify several DNA loci in a single experiment. This requires the presence of forward and reverse primers for each of the DNA loci to be amplified, though primers can be shared between different loci. Undesired events such as mispriming and primer dimerization may occur in an MP-PCR experiment due to the presence of large numbers of primers, thus compromising the quality of PCR products. Hence, it is critical to minimize the number of primers employed in MP-PCR. This can be achieved by selecting primers that act as forward and/or reverse primers for multiple amplification targets. To further reduce the number of primers and increase the amount of primer sharing between target loci, an effective method is to use primers with multiple nucleotides at certain positions, commonly referred to as *degenerate primers* [4].

The *total degeneracy* of a degenerate primer p, denoted as d_p, is defined as the product of the numbers of nucleotides at each of its positions. Thus, if $p = p_1 \dots p_l$ is a degenerate primer of length l, the total degeneracy of p is $d_p = \Pi_{i=1}^{l} |p_i|$. Since highly degenerate primers have low specificity and can lead to a large number of mispriming events, it is common to impose an upper bound on the degeneracy of the primers to be used in a MP-PCR reaction. Therefore, the goal is to identify a set of degenerate primers, each of a specified length l and of total degeneracy at most d, that collectively *cover* all 5'-3' DNA sequences upstream of the given amplification targets. A degenerate primer p is said to *cover* a DNA sequence s if one of the d non-degenerate primers represented by p is a substring of s. Formally, the problem can be stated as follows:

Multiple Degenerate Primer Selection Problem (MDPSP): Given n DNA sequences $S = \{S_1, S_2, ..., S_n\}$, and two integers l and d, find a minimum cardinality set of degenerate primers P_d that covers S, such that each degenerate primer $p \in P_d$ is of length l and has total degeneracy at most d.

MDPSP was proved to be NP-Complete in [6]. Therefore, in this paper we focus on heuristic algorithms for MDPSP with good practical performance. The rest of the paper is organized as follows. Most of the MDPSP heuristics proposed in the literature select degenerate primers one at a time, attempting at each step to maximize the *coverage* of the selected degenerate primer, i.e., the number of not-yet-covered sequences that it covers. In Section II-A, we review existing heuristic algorithms for MDPSP, discussing the salient ones in more detail. In Section II-B, we describe the two new greedy algorithms for MDPSP and analyze their running time. Finally, we give experimental results comparing the solution quality and runtime of proposed algorithms to that of two best performing previous algorithms in Section III. Section IV concludes the paper.

II. METHODS

A. Algorithms for Degenerate Primer Selection

The first efforts of reducing the number of primers for multiple DNA sequence amplification by identifying common substrings in a subset of the input came from Pearson et al. in [8], where the authors designed a set of non-degenerate primers using a greedy set cover algorithm and an exact

This research has been supported in part by NSF awards ITR-0326155, IIS-0546457 and DBI-0543365.

S. Balla is a Research Fellow at the Sloan-Kettering Institute, Memorial Sloan-Kettering Cancer Center, 1275, York Avenue, New York, NY 10065, USA ballas@mskcc.org

S. Rajasekaran and I. I. Măndoiu are with the Department of Computer Science & Engineering, University of Connecticut, Storrs, CT 06269, USA rajasek@engr.uconn.edu, ion@engr.uconn.edu

branch-and-bound algorithm. Linhart and Shamir pioneered the work on degenerate primer design in [7], proposing the HYDEN algorithm for MDPSP. HYDEN designs every degenerate primer by repeatedly solving the Maximum Coverage Degenerate Primer Design (MC-DPD) problem on the uncovered set of input sequences. The authors showed that HYDEN performed well in practice when used to design degenerate primers for a set of Human Olfactory genes that in turn were used to extract genes of the same family from genome data. Their elaborate work in [6] formulates and proves several variants of Degenerate Primer Selection to be NP-Complete. Wei, Kuhn and Narasimhan [11] proposed the DePiCt algorithm that employs hierarchical clustering to group a set of given protein sequences based on similarity and designs degenerate primer pairs for each cluster from highly conserved regions of a multiple alignment of the sequences in the cluster.

The MIPS algorithm of Souvenir et al. [10] follows an iterative beam search technique to design degenerate primers. It starts with a set of primers (called *2-primers*) that cover two sequences from an input of n sequences. Then it extends the coverage of the primers in the candidate set by one additional sequence, introducing degeneracy in the primers if necessary, retains a subset of these primers (the number determined by an input parameter called *beam size b*) for the next iterative step until none of the primers can be extended further without crossing the target degeneracy. At this point, the primer with the lowest degeneracy is selected and the sequences that it covers are removed from the input set. The procedure is repeated until all the sequences are covered. MIPS has overall time complexity of $O(bn^3mp)$, where b is the beam size, n is the number of sequences, m is the sequence length, l is the primer length, and p is the cardinality of the final set of selected degenerate primers. Experimental results for varying number of input sequences and different target degeneracy, the sequences being uniformly distributed i.i.d. sequences of equal length, were reported in [10]. It was shown that MIPS always produced fewer primers than HYDEN.

The DPS algorithm in [1] follows the footsteps of MIPS but uses additional sorting and a new ranking metric called coverage-efficiency in each iterative step to order and select the b best primers, thus improving the worst-case time complexity of MIPS. DPS was shown to perform better in practice on real biological data both in quality and runtime when compared to MIPS.

A downside of both MIPS and DPS is that their runtime depends upon the empirical input parameter b; the authors of [10] suggest using a value of b close to the number of sequences in the input to achieve a good trade-off between solution quality and runtime. Although DPS was shown to achieve better quality of output for lesser values of b, its dependency on b still exists. In the following section we propose two new greedy approaches that eliminate the dependency on b, while improving the runtime and retaining the quality of output. The greedy approaches avoid the time consuming process of explicit generation of next generation candidates from substrings of length l from all the uncovered sequences as done by MIPS and DPS. Instead, in order to increase coverage they identify the best candidate to merge the primer with based on two measures, namely, the Hamming distance of possible candidates of the input from the primer and their potential to increase the degeneracy of the primer, explained in detail in the next section. Also, in order to speed up the generation of the initial set of 2-primers, MIPS and DPS adopt a FASTA look-up approach, which could skip some valid primer candidates because of the mismatches between two substrings of length l being distributed in such a way that they do not share a sufficiently long substring. This is overcome in our greedy algorithms by adopting an efficient technique to calculate the Hamming distances between substrings of length l in a sequence and those of other sequences of the input.

B. New Greedy Algorithms for MDPSP

In this section we propose two new greedy heuristics for MDPSP, called algorithm DPS-HD (stands for Degenerate Primer Selector by Hamming Distance) and algorithm DPS-DIP (stands for Degenerate Primer Selector by Degenerate Increase Potential). Similar to its predecessors, DPS-HD designs members of the output set P_d one at a time, attempting to select a degenerate primer P of maximum coverage for the set of input sequences to be covered yet. In order to select P, DPS-HD works as follows. Let us consider that we are selecting the first degenerate primer of P_d. Therefore, all the input sequences are alive (yet to be covered). An arbitrary sequence $S_i, 1 \leq i \leq n$ is selected; let it be S_k. For every u, a substring of length l (l-mer) of S_k, the following is performed to develop u from a non-degenerate primer that covers S_k to a degenerate primer of degeneracy at most d to cover as many uncovered sequences as possible. First, its Hamming distances (i.e., the number of mismatches) to every l-mer in S_i is calculated using the efficient technique described in [9]. Let c_u be a binary array of size n, whose k-th bit is set to 1 to indicate that u covers S_k. Let $D[0:l]$ be an array of linked lists, where $D[h], 0 \leq h \leq l$ represents the list that contains elements of the form (q, r), if the Hamming distance $dist(u, S_{q,r..(r+l-1)}) = h$. $D[0]$ contains those elements that are sequences already covered by u, therefore, its elements are examined one at a time, and the q-th bit of c_u is set to 1 for each additional sequence S_q covered by u. Then, an arbitrary element from the non-empty list of D with the lowest value of h is chosen and the l-mer represented by the element, say v, is merged with u. c_u is updated accordingly. Let $dist(u, v) = h_{uv}, 1 \leq h_{uv} \leq l$. The h_{uv} positions to which additional symbols are added to u and the corresponding symbol added are maintained in separate lists. For $D[h], h_{uv} \leq h \leq l$, the elements in the lists are processed, creating a next generation set of candidate sites, say D', by recalculating the Hamming distance of l-mers in sequences that are alive in $O(nmh_{uv})$ time. This procedure is continued until the degeneracy of u reaches the target degeneracy d or all input sequences are covered. The degenerate primer u with the maximum coverage is selected as P and added to P_d. The sequences covered by P are eliminated and the procedure is repeated until all input sequences are covered. The pseudocode of the algorithm is given below:

```
Algorithm DPS-HD {
    $P_d := null$;
    $R := \{1, 2, ..., n\}$; //sequences alive
    while($|R| > 0$) {
        $P := null$;
        $coverage_P := 0$; // number of sequences covered by $P$
        (1) Choose an arbitrary sequence $k$ from $|R|$;
        (2) Calculate the Hamming distance of all $l$-mers in $S_k$
        with the $l$-mers of other sequences alive;
        (3) for each $l$-mer $u \in S_k$ do {
            expand $u$ to cover additional sequences,
            by merging $u$ with a $v$ at a minimum
            Hamming distance from it, and recalculating
            Hamming distances of sites alive,
            until $d_u := d$ or $coverage_u := |R|$;

            if($coverage_u > coverage_P$) {
              $P := u$;
              $coverage_P := coverage_u$;
            }
        }
        (4) $P_d := P_d \cup P$;
        (5) Delete sequences covered by $P$ from $R$;
    }
    output $P_d$;
}
```

Step (2) of the algorithm above that calculates the Hamming distance of all the $O(m)$ l-mers in the chosen sequence takes $O(nm^2)$ time and $O(nm^2)$ space, using the technique described in [9]. A non-degenerate primer can be expanded into a degenerate primer of degeneracy at most d in $O(|\Sigma| \log_{|\Sigma|} d)$ iterations (as the number of symbols that can be added to achieve the degeneracy d is in the range $[\lfloor \log_2 d \rfloor : (|\Sigma| - 1)\lceil \log_{|\Sigma|} d \rceil]$). Each expansion iteration recalculates the Hamming distances of $O(nm)$ candidate sites that are alive. Therefore, the expansion of one non-degenerate primer takes $O(nm|\Sigma| \log_{|\Sigma|} d)$. Since $O(m)$ candidate l-mers are expanded in Step (3), selecting one primer takes $O(nm^2|\Sigma| \log_{|\Sigma|} d)$ time. If $|P_d| = p$, then the runtime of algorithm DPS-HD is $O(nm^2p|\Sigma| \log_{|\Sigma|} d)$ and its space requirement is $O(nm^2)$.

The greedy approach of algorithm DPS-DIP differs from DPS-HD by the criterion used to rank the possible primer sites that a given primer could be merged to expand its degeneracy and coverage, namely, the *degeneracy increase potential*, defined as follows:

The *degeneracy increase potential* (DIP) of a primer site v (a l-mer in a sequence yet to be covered) is the factor f of increase in degeneracy caused by merging v with candidate primer u. Thus, if $u' = u \cup v$, then $f = d_{u'}/d_u$.

Adopting DIP as the expansion site selection criterion is based on the intuition that two primer sites, say x and y, that are at the same Hamming distance from a primer u could result in primers of different resultant degeneracy when merged with u, depending on the positions of x and y that differ from u. If the number of symbols in a position i of u, $1 \leq i \leq l$ is 1, if $u[i] \neq x[i]$, the potential increase in degeneracy $x[i]$ would have is 2, if $|u[i]| = 2$, this value would be 1.5 and if $|u[i]| = 3$, it would be 1.33. The DIP factor f of a primer site x is the product of the potential increase of each of its positions with respect to the primer u. Therefore, we believe that selecting the site with the least DIP value increases u's degeneracy by the least amount and may yield improved overall coverage.

We use the same strategy of DPS-HD to calculate the DIP values of primer sites. Clearly, the DIP values of primer sites are real numbers in the range $[1, 2^l]$. Although the array of linked lists described above (D) cannot be used here, it can be seen that such an array is not critical for either DPS-HD or DPS-DIP. Since each iteration recomputes all $O(nm)$ DIP values, the site with the minimum DIP value can be identified during recomputation. A DIP value of 1 indicates the primer already covers a particular primer site (and in turn the sequence of the input that it is a substring of). Therefore, the runtime of DPS-DIP is still $O(nm^2p|\Sigma| \log_{|\Sigma|} d)$.

III. RESULTS

We have implemented algorithms DPS-HD and DPS-DIP in Java and tested their performance on random and real biological data, essentially the same datasets described in [1]. The MIPS algorithm, implemented in C++, was obtained from its authors and the DPS algorithm was implemented in Java. All the implementations were run on a PowerEdge 2600 Linux server with 4 GB of RAM and dual 2.8 GHz Intel Xeon CPUs - only one of which is used by these sequential implementations. A comparison of the performance of MIPS, DPS, DPS-HD and DPS-DIP is provided in Tables I-III. The HYDEN algorithm designs primers for a given input in the form of primer pairs, considering the 5' end and the 3' end of the input dataset separately, while algorithm MIPS, DPS and DPS-HD consider the 5' end sequences and the reverse complement of the 3' end sequences together. The authors of MIPS showed that this strategy reduced the number of primers in the output. For the above reason, it may not be appropriate to directly compare the performance of algorithm HYDEN with the other algorithms. Therefore, we do not include HYDEN in our analysis below.

Two real biological datasets were considered. The first dataset is a set of 95 DNA sequences on which the algorithm MIPS was tested in [10]. Each sequence in the dataset has an SNP in it and the goal of the MP-PCR is to amplify the regions of every sequence that would include the SNPs in the amplified products. Thus, the input to the algorithms was a dataset of 190 sequences, each input sequence having two representatives in the input set, the first being the sub-sequence from the start to one position before the SNP and the second being the reverse complement of the subsequence from one position after the SNP to the end of the sequence. The second dataset is a set of 50 human olfactory genes, which we received from the authors of algorithm HYDEN. Each sequence in this dataset was approximately 1 Kbp long. Taking the first 300 nucleotides and the reverse complement of the last 300 nucleotides of each gene generated the input

TABLE I

PERFORMANCE ON RANDOM DATASETS: $l = 15, d = 10000$, p = # OF PRIMERS, t = EXECUTION TIME IN SEC.

	MIPS		DPS		DPS-HD		DPS-DIP	
n	p	t	p	t	p	t	p	t
20	4	1.7	4	1.7	4	1.51	4	2.97
40	6.3	9	6	11.9	6	5.41	6	10.35
60	9	26	8.3	39.1	9	11.95	9	22.40
80	11	54.1	10.8	93.2	11	20.82	11	40.30
100	13.1	100.3	12.3	179.6	13.2	32.67	13	58.35
120	15.1	180.5	14.1	316.0	15	45.44	15	83.24
140	17	218.4	16.1	499.3	17	63.02	17	110.06
160	19.2	313.8	17.8	761.1	19	80.40	19	143.27
180	21	422.8	19.5	1103.0	21	125.47	20.7	172.62

TABLE II

PERFORMANCE ON RANDOM DATASETS: $l = 15, d = 100000$, p = # OF PRIMERS, t = EXECUTION TIME IN SEC.

	MIPS		DPS		DPS-HD		DPS-DIP	
n	p	t	p	t	p	t	p	t
20	3	1.7	2.9	2.5	2.8	1.47	2.3	2.77
40	4	8.0	4	18.3	4	5.15	4	9.26
60	5.6	22.3	5	57.4	5.6	11.26	5.2	19.55
80	7	47.6	6	131.1	7	19.13	7	32.33
100	8	88.8	7	251.7	8	28.13	7.9	47.96
120	9	139.6	8	431.9	9	40.99	9	68.12
140	10	217.1	8.9	679.8	10	53.93	10	87.82
160	10.9	326.6	9.8	1039.9	11	68.53	11	111.17
180	11.9	435.3	10.4	1530.8	12	85.10	12	137.31

TABLE III

PERFORMANCE ON BIOLOGICAL DATASETS: $l = 20$, p = # OF PRIMERS, t = EXECUTION TIME IN SEC.

Dataset	Degeneracy	MIPS		DPS		DPS-HD		DPS-DIP	
	(d)	p	t	p	t	p	t	p	t
Olfactory	4096	11	130	9	92.1	10	13.75	9	21.75
	16384	10	161	8	114.6	8	13.24	7	21.35
	65536	8	197	6	152.3	6	13.64	6	21.51
	262144	7	214	5	209.5	5	14.53	5	22.55
SNP	4096	55	716	53	637.5	55	28.44	54	63.88
	16384	45	803	43	628.8	45	24.48	45	52.37
	65536	37	882	35	755.9	38	21.77	37	45.38
	262144	31	789	29	966.2	31	18.00	30	38.47

set for the experiment. Thus, the input set consisted of 100 sequences each of length 300. The length of the primer searched was 20, the value of $b = n$ were adopted for MIPS and DPS and the degeneracy thresholds considered were 4096, 16384, 65536, and 262144 respectively.

10 random datasets from [5] were used (generated from the uniform distribution induced by assigning equal probabilities to each nucleotide for each value of n in 20, 40, 60, 80, 100, 120, 140, 160, 180). We recorded the average over all the 10 runs for each option of n sequences. The sequence length m was 300 for all random sequences. The value of $b = n$ was adopted for MIPS and DPS, the length of the primer searched for was 15 and the experiments were run for degeneracy of 10000 and 100000 respectively.

IV. CONSLUSION

In this paper, we proposed new greedy algorithms for the problem of selecting multiple degenerate primers for use in MP-PCR. The proposed algorithms eliminate the dependency of previously known algorithms on an empirical input parameter that affects the runtime and quality of output. The implementations of the proposed algorithms execute faster than the previously known algorithms providing the same quality of output on random and real biological data. We believe that the low memory requirement and fast execution of the proposed algorithms will be useful in processing very large input sets. Also, our algorithms are highly amenable to parallelization and we plan to test if parallel implementations lead to better output quality as the primers could be developed from more than one arbitrary uncovered input sequence. We are currently working on extending the heuristics proposed in this paper to address other variants of the degenerate primer design such as those in [2][6][7].

REFERENCES

[1] S. Balla, S. Rajasekaran, I. I. Mandoiu, "Efficient Algorithms for Degenerate Primer Search", *International Journal of Foundations of Computer Science* **18(4)**, 2007, pp 899-910.

[2] S. Balla, S. Rajasekaran, "An Efficient Algorithm for Minimum Degeneracy Primer Selection", *IEEE Transactions on Nanobioscience (IEEE-TNB)* **6(1)**, 2007, pp 12-17.

[3] J. S. Chamberlain, R. A. Gibbs, J. E. Rainer, P. N. Nguyen, C. T. Casey, "Deletion screening of the Duchenne muscular dystrophy locus via multiplex DNA amplification", *Nucleic Acids Research* **16**, 1988, pp 11141-11156.

[4] S. Kwok, S. Y. Chang, J. J. Sninsky, A. Wang, "A guide to the design and use of mismatched and degenerate primers", *PCR Methods and Applications* **3**, 1994, pp S39-S47.

[5] K.M. Konwar and I.I. Mandoiu and A. Russell and A. Shvartsman, "Algorithms for Multiplex PCR Primer Set Selection with Amplification Length Constraints", *In I.I. Mandoiu and A.Z. Zelikovsky, Bioinformatics Algorithms: Techniques and Applications, Wiley*, 2008, pp. 241-258

[6] C. Linhart, R. Shamir, "The degenerate primer design problem Theory and Applications", *Journal of Computational Biology* **12(4)**, 2005, pp 431-456.

[7] C. Linhart, R. Shamir, "The degenerate primer design problem", *Bioinformatics* **18(1)**, 2002, pp S172-S180.

[8] W. R. Pearson, G. Robins, D. E. Wrege, T. Zhang, "On the primer selection problem in polymerase chain reaction experiments", *Discrete Applied Mathematics* **71**, 1996, pp 231-246.

[9] P. Pevzner, S. Sze, "Combinatorial Approaches to Finding Subtle Signals in DNA Sequences", *Proc. of Eighth International Conference on Intelligent Systems for Molecular Biology*, 2000, pp 269-278.

[10] R. Souvenir, J. Buhler, G. Stormo, W. Zhang, "Selecting Degenerate Multiplex PCR Primers", *Proc. 3rd Intl. Workshop on Algorithms in Bioinformatics (WABI)*, 2003, pp 512-526.

[11] X. Wei, D. N. Kuhn, G. Narasimhan, "Degenerate Primer Design via Clustering", *Proc. of the 2003 IEEE Bioinformatics Conference (CSB)*, 2003, pp 75-83.